IBM Developer SKILLS NETWORK

# IBM DATA SCIENCE

## APPLIED DATA SCIENCE CAPSTONE PROJECT

# Space X to Space Y

AUTHOR

**SHUBHAM MEHAR**

DATE- 22/01/2024

# REPORT OUTLINE



[GitHub Repository Link]

# EXECUTIVE SUMMARY

## SUMMARY OF METHADOLOGIES

- Data Collection involves using the SpaceX REST API and web scraping techniques.
- Data Wrangling is implemented to create a success/failure outcome variable.
- Exploratory Data Analysis considers payload, launch site, flight number, and yearly trends using data visualization techniques.
- Data is analyzed with SQL, calculating statistics like total payload, payload range for successful launches, and the total number of successful and failed outcomes.
- Launch site success rates and proximity are explored using Folium.
- Launch sites with the most success and successful payload ranges are visualized using Plotly and Dash to create an interactive dashboard.
- Model Building utilizes logistic regression, support vector machine (SVM), decision tree, and K-nearest neighbor (KNN) to predict landing outcomes.

# EXECUTIVE SUMMARY

## SUMMARY OF RESULTS

- Success rates improve as flight numbers increase from launch sites, with an initially poor rate.
- Higher payload correlates with a higher success rate.
- CCAFS SLC 40 launches most payloads weighing over 12000 kg.
- VAFB-SLC has no launches for heavy payloads (>10000 kg).
- ES-L1, GEO, HEO, and SSO show a perfect 100% success rate.
- SO exhibits a 0% success rate. GTO, ISS, LEO, MEO, and PO have success rates between 50% and 80%.
- Decision Tree algorithm emerges as the best predictor, showcasing superior accuracy on validation data among various algorithms.

# INTRODUCTION

## PROJECT BACKGROUND

The commercial space age is here, companies are making space travel affordable for everyone. Virgin Galactic is providing suborbital spaceflights. Rocket Lab is a small satellite provider. Blue Origin manufactures sub-orbital and orbital reusable rockets. Perhaps the most successful is SpaceX.

SpaceX's accomplishments include:

Sending spacecraft to the International Space Station. Starlink, a satellite internet constellation providing satellite Internet access. Sending manned missions to Space. One reason SpaceX can do this is the rocket launches are relatively inexpensive. SpaceX advertises Falcon 9 rocket launches on its website with a cost of 62 million dollars; other providers cost upwards of 165 million dollars each, much of the savings is because SpaceX can reuse the first stage.

## TO FIND

In this capstone, you will take the role of a data scientist working for a new rocket company. Space Y that would like to compete with SpaceX founded by Billionaire industrialist Allon Musk. Your job is to determine the price of each launch.

# SPACE X

Successfully Launch

Successfully Landed

Unsuccessfully Landed

# METHODOLOGY

## EXECUTIVE SUMMARY

### ❑ DATA COLLECTION METHODOLOGY

- Collecting data with an API
- Web Scrapping -using the Python BeautifulSoup package to web scrape some HTML tables that contain valuable Falcon 9 launch records from WikiPiedia.

❑ Data Wrangling- Filtering data for Falcon 9 Launches, handling missing data and Feature Encoding.

❑ Exploratory data analysis (EDA) using visualization libraries Matplotlib and Seaborn and SQL.

❑ Interactive visual analytics using Folium and Plotly Dash.

❑ Predictive analysis using classification models

- Train Test Split
- Train the chosen model on the training dataset.
- Adjust hyperparameters to optimize performance.
- Evaluate the model's performance on the testing dataset using appropriate metrics

# DATA COLLECTION

## Collecting the Data with an API

- ❑ SpaceX launch data that is gathered from an API, specifically the SpaceX REST API. This API will give us data about launches, including information about the rocket used, payload delivered, launch specifications, landing specifications, and landing outcome.
- ❑ Request and parse the SpaceX launch data using the GET request.
- ❑ The response content as a Json using .json() and turn it into a Pandas dataframe using .json_normalize().
- ❑ Create dataframe from the dictionary
- ❑ Filter dataframe to contain only Falcon 9 launches
- ❑ Replace missing values of Payload Mass with calculated .mean()
- ❑ Export data to csv file
- ❑ [Data collection from API file link]

# DATA COLLECTION

## Collecting the Data with an API Flowchart

Request SpaceX Launch Data

↓

Parse API Response as JSON

↓

Create DataFrame using .json_normalize()

↓

Filter Falcon 9 Launches

↓

Replace Missing Payload Mass

↓

Export to CSV

# DATA COLLECTION

## Web scraping

- Using the Python BeautifulSoup package to web scrape some HTML tables that contain valuable Falcon 9 launch records from Wiki Pages.
- Request the Falcon9 Launch Wiki page from its URL
- Create a BeautifulSoup object from the HTML response.
- Extract all column/variable names from the HTML table header
- Create a data frame by parsing the launch HTML tables
- Fill in the parsed launch record values into launch_dict
- Export it to a CSV
- Web Scrapping file link

# DATA COLLECTION

## Web scraping Flowchart

Import the necessary libraries

↓

Send a GET request to Falcon 9 Launch Wiki URL

↓

Create a BeautifulSoup object from HTML response

↓

Extract all column/variable names from the HTML table header

↓

Create an empty dictionary (launch_dict)

↓

Parse the launch HTML tables and fill in launch_dict

↓

Create a Pandas DataFrame from launch_dict

↓

Export the DataFrame to a CSV file

# DATA WRANGLING

- ❑ Exploratory Data Analysis (EDA) to find some patterns in the data and determine what would be the label for training supervised models.
- ❑ Calculate the number of launches on each site
- ❑ Calculate the number and occurrence of each orbit
- ❑ Calculate the number and occurence of mission outcome of the orbits
- ❑ Create a landing outcome label from Outcome column
    - Using the Outcome, create a list where the element is zero if the corresponding row in Outcome is in the set bad_outcome; otherwise, it's one. Then assign it to the variable landing_class:
- ❑ Export it to a CSV
- ❑ Data Wrangling File Link

# DATA WRANGLING

## Exploratory Data Analysis Using 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 average payload mass carried by booster version F9 v1.1
- ❑ List the date when the first succesful landing outcome in ground pad was acheived.
- ❑ List the names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000
- ❑ List the total number of successful and failure mission outcomes
- ❑ List the names of the booster_versions which have carried the maximum payload mass. Use a subquery
- ❑ List the records which will display the month names, failure landing_outcomes in drone ship ,booster versions, launch_site for the months in year 2015.
- ❑ Rank 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.
- ❑ [SQL FILE LINK]

# DATA WRANGLING

## Exploratory Data Analysis Using Visualization

Matplotlib and Seaborn python libraries are used for EDA using Visualizations.

- ❑ Visualize the relationship between Flight Number and Launch Site
- ❑ Plot a scatter point chart with x axis to be Flight Number and y axis to be the launch site, and hue to be the class value
- ❑ Visualize the relationship between Payload and Launch Site
- ❑ Plot a scatter point chart with x axis to be Pay Load Mass (kg) and y axis to be the launch site, and hue to be the class value
- ❑ Visualize the relationship between success rate of each orbit type
- ❑ check if there are any relationship between success rate and orbit type, use groupby method on Orbit column and get the mean of Class column.
- ❑ Visualize the relationship between FlightNumber and Orbit type
- ❑ Plot a scatter point chart with x axis to be FlightNumber and y axis to be the Orbit, and hue to be the class value
- ❑ Visualize the relationship between Payload and Orbit type
- ❑ Plot a scatter point chart with x axis to be Payload and y axis to be the Orbit, and hue to be the class value
- ❑ Visualize the launch success yearly trend
- ❑ Plot a line chart with x axis to be the extracted year and y axis to be the success rate

❑ [EDA Visualization File Link]

# FEATURE ENGINEERING

- some preliminary insights about how each important variable would affect the success rate, we will select the features that will be used in success prediction in the future module.
- Create dummy variables to categorical columns
  - Use the function get_dummies and features dataframe to apply OneHotEncoder to the column Orbits, LaunchSite, LandingPad, and Serial. Assign the value to the variable features_one_hot, display the results using the method head. Your result dataframe must include all features including the encoded ones.
- Cast all numeric columns to `float64`
- features_one_hot dataframe only contains numbers cast the entire dataframe to variable type float64, use astype function
- Export it to a CSV

# Interactive Map with Folium

## ❑ Mark all launch sites on a map

- Using the launch site name, coordinates (i.e., latitude and longitude) to mark all launch sites on an interactive map. Add a circle marker on the launch site with the launch site name using an icon label.

## ❑ Mark the success/failed launches for each site on the map

- To enhance the map by adding the launch outcomes for each site, and see which sites have high success rates. Recall that data frame spacex_df has detailed launch records, and the class column indicates if this launch was successful or not
- create markers for all launch records. If a launch was successful (class=1), then we use a green marker and if a launch was failed, we use a red marker (class=0)
- Marker clusters can be a good way to simplify a map containing many markers having the same coordinate.
- Create a new column in launch_sites dataframe called marker_color to store the marker colors based on the class value

## ❑ Calculate the distances between a launch site to its proximities

- MousePosition on the map to get coordinate for a mouse over a point on the map. As such, while you are exploring the map, you can easily find the coordinates of any points of interests (such as railway)
- Launch site and explore its proximity to see if you can easily find any railway, highway, coastline, etc. Move your mouse to these points and mark down their coordinates (shown on the top-left) in order to the distance to the launch site.

[Interactive Map with Folium File Link]

# Interactive Dashboard with Plotly Dash

- Building a Plotly Dash application for users to perform interactive visual analytics on SpaceX launch data in real-time.
- This dashboard application contains input components such as a dropdown list and a range slider to interact with a pie chart and a scatter point chart.
- Add a Launch Site Drop-down Input Component
  - To select launch site from drop down
- Add a callback function to render success-pie-chart based on selected site dropdown
  - It will help to visualize the pie chart for all launch site or for particular launch site.
- Add a Range Slider to Select Payload
  - Slider is used to check payload between a range.
- Add a callback function to render the success-payload-scatter-chart scatter plot
  - It will change according to slider range and dropdown selected launch site.

[Interactive Dashboard with plotly dash file link]

# Predictive Analysis

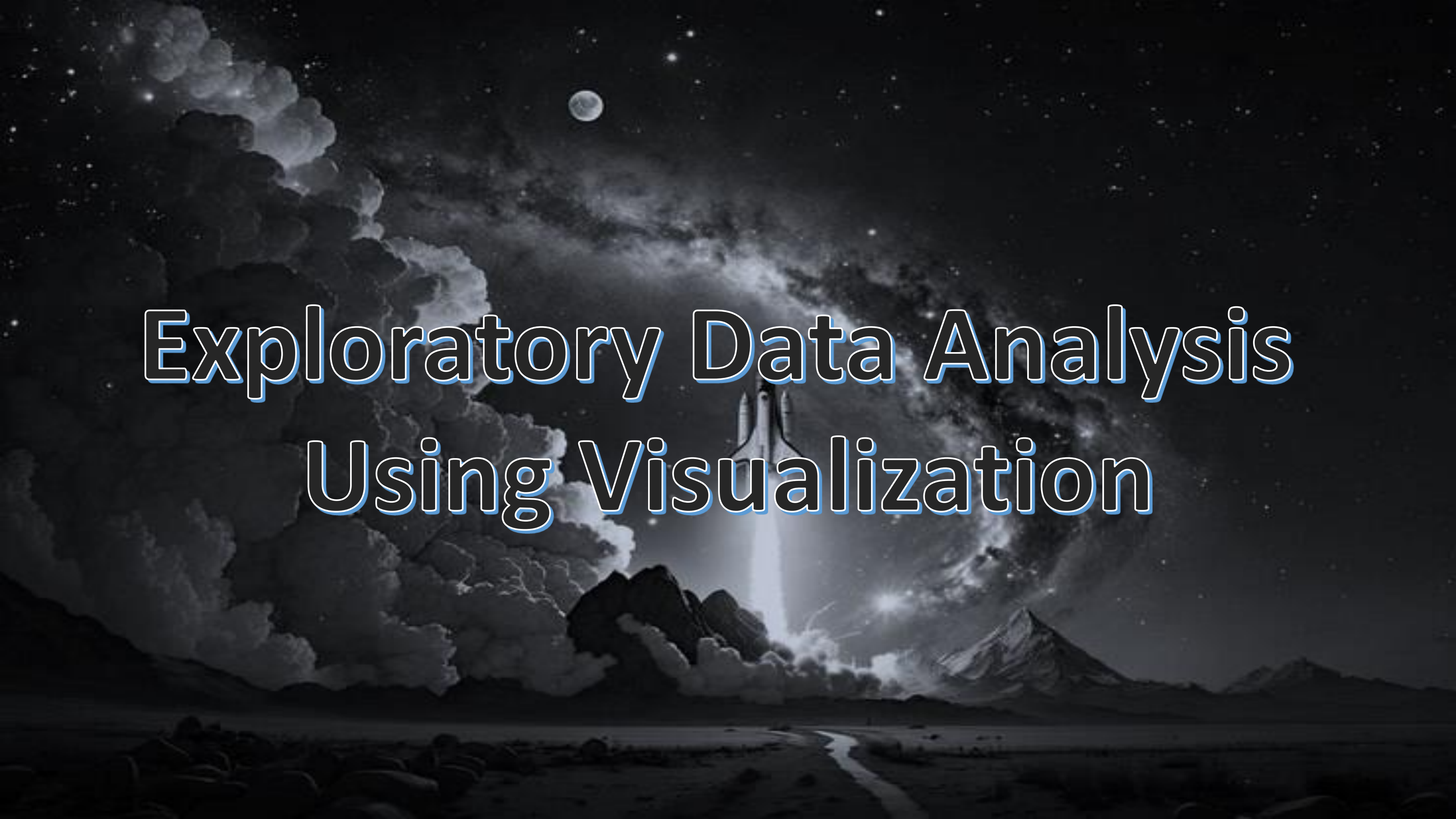
- Create a NumPy array from the column Class in data
- Standardize the data in X then reassign it to the variable X using the transform
- train_test_split to split the data X and Y into training and test data. Set the parameter test_size to 0.2 and random_state to 2.
- Create a GridSearchCV object for classification algorithm
- The GridSearchCV object for classification algorithms. We display the best parameters using the data attribute best_params_ and the accuracy on the validation data using the data attribute best_score_.
- Calculate cc is a table that is often used to evaluate the performance of a classification algorithm. It summarizes the results of a classification problem, showing the number of true positive (TP), true negative (TN), false positive (FP), and false negative (FN) predictions.

- Predictive analysis (Machine Learning) File Link

# RESULTS

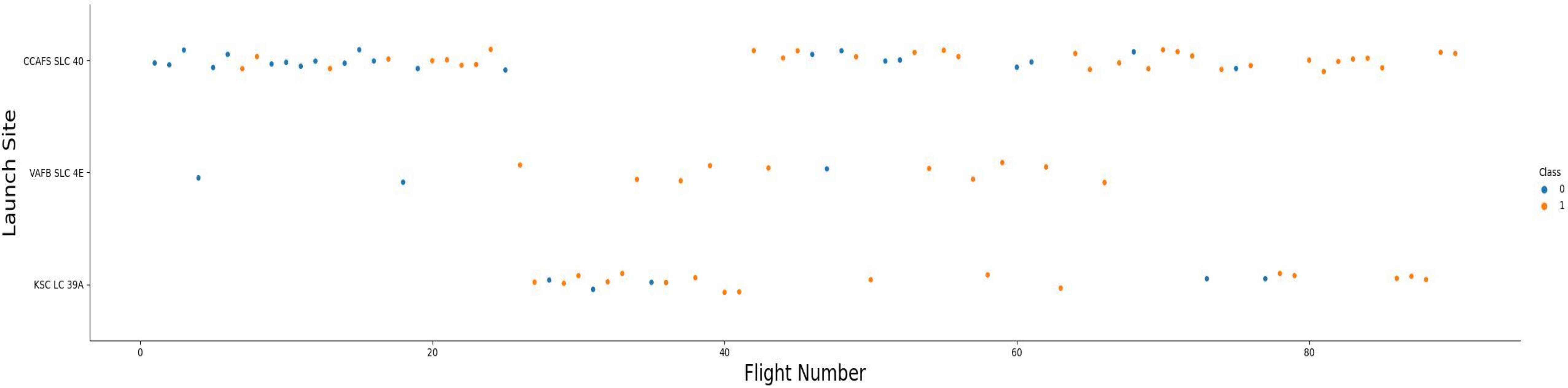
- ❑ Exploratory data analysis results
- ❑ Interactive analytics demo in screenshots
- ❑ Predictive analysis results

# Exploratory Data Analysis Using Visualization



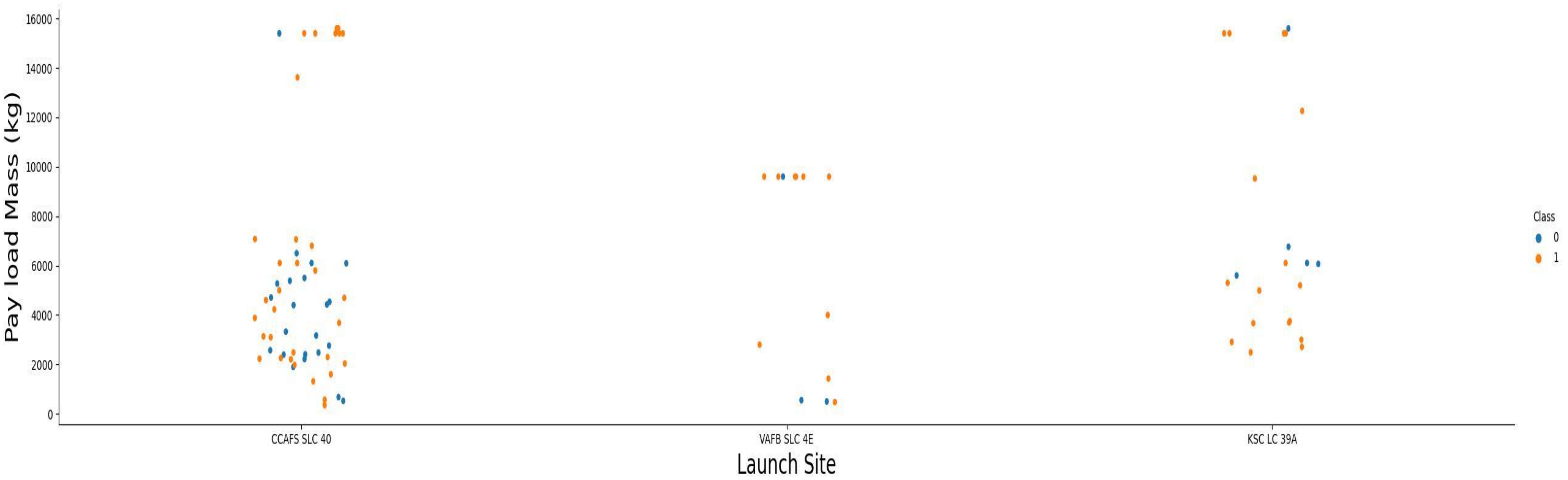
# The relationship between Flight Number and Launch Site

| Class | Landing Result | Color Code |
|---|---|---|
| 0 | Fail | Blue |
| 1 | Success | Orange |



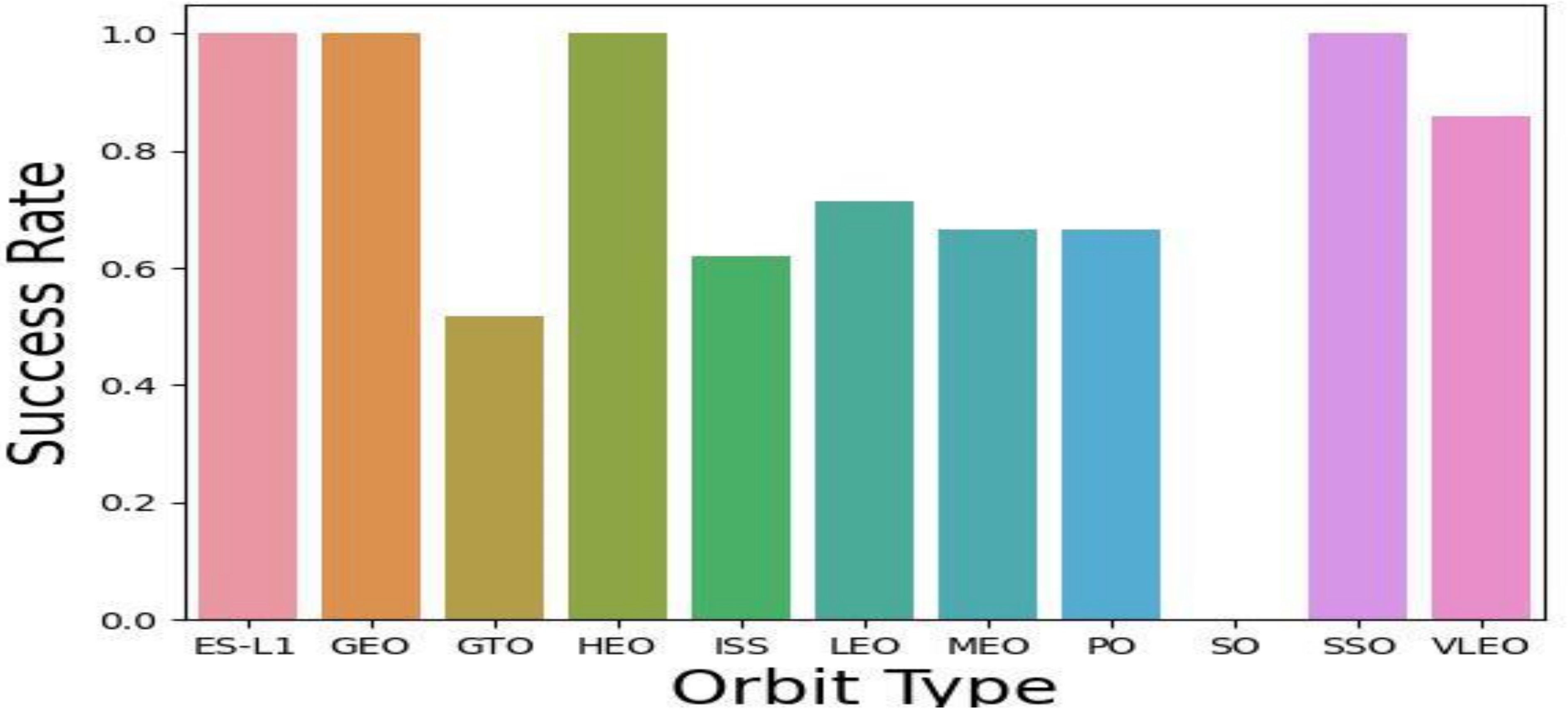

- As the flight number increases from the launch sites, the success rate also improves. Initially, the success rate is poor.

# The relationship between Payload and Launch Site



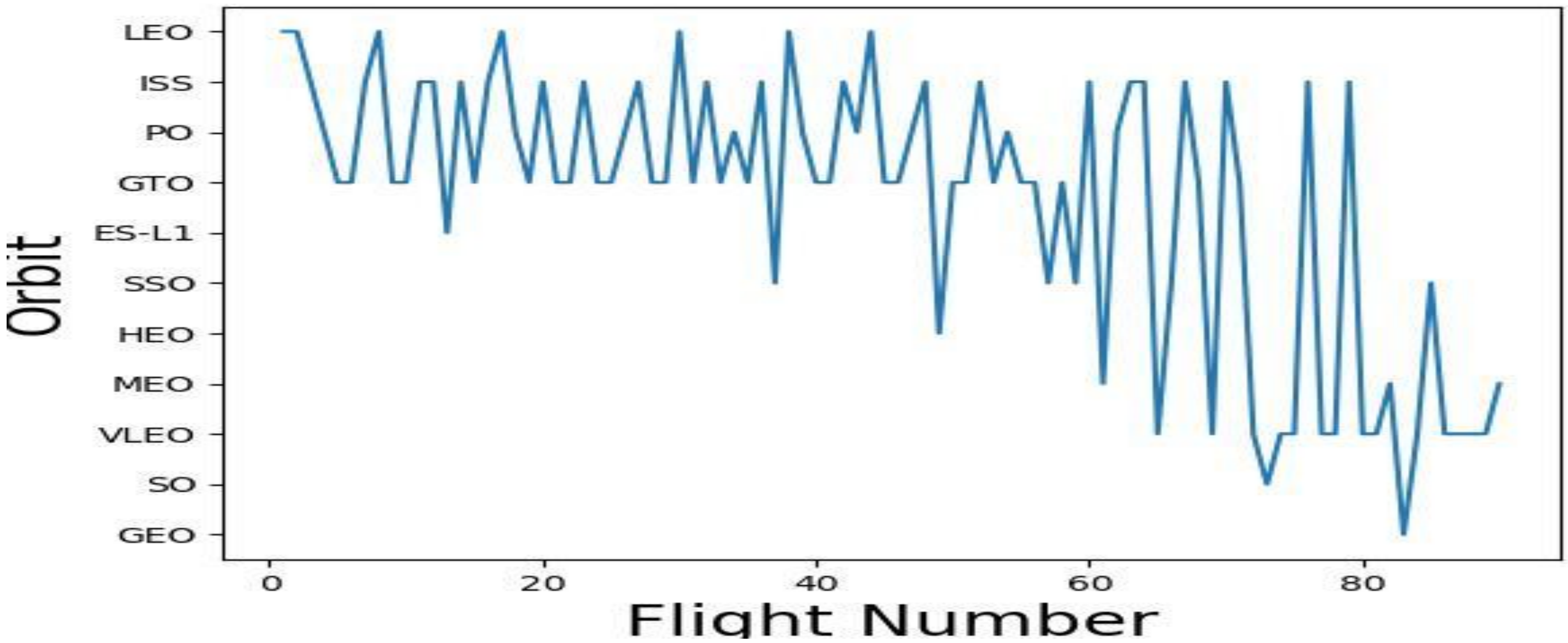

- The higher the payload, the higher the success rate.
- Most payloads weighing more than 12000 kg are launched from CCAFS SLC 40 launch site.
- VAFB-SLC launch site there are no rockets launched for heavy payload mass(greater than 10000).

# The relationship between success rate of each orbit type



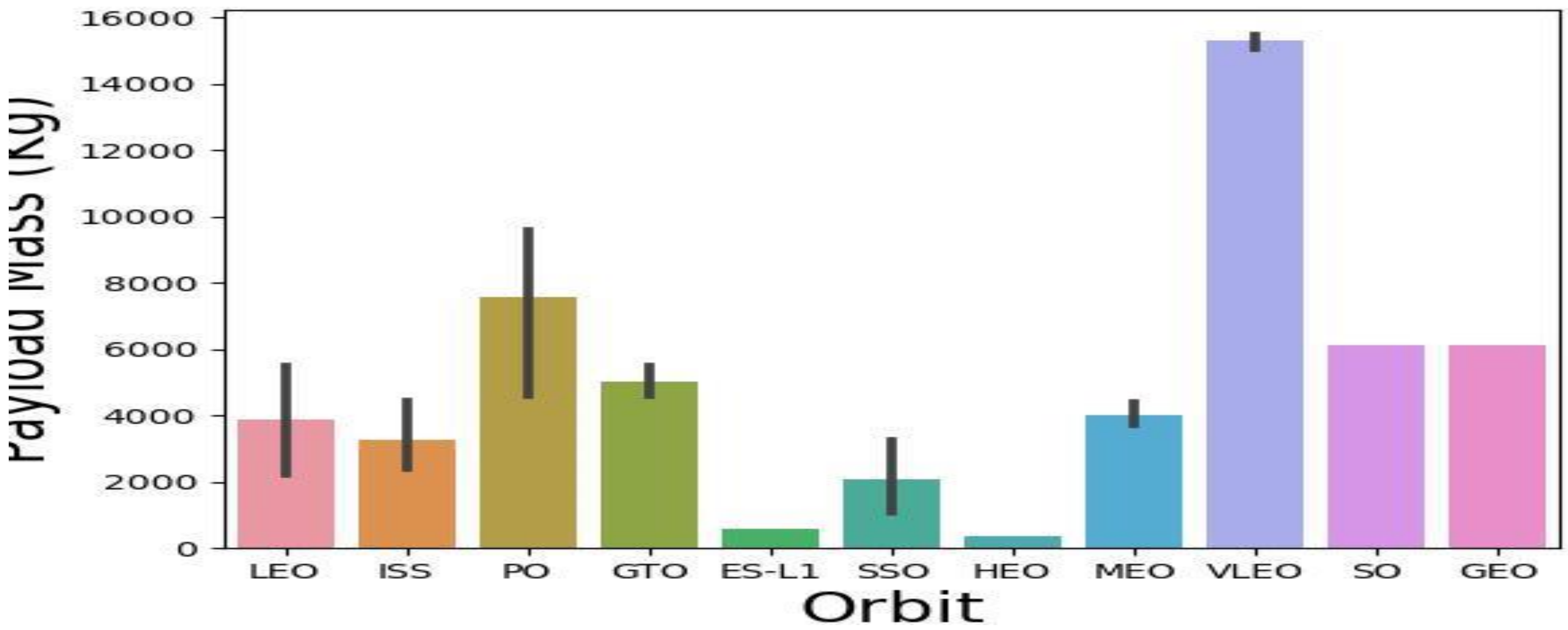

- ES-L1, GEO, HEO, and SSO have a 100% success rate.
- SO has a 0% success rate.
- GTO, ISS, LEO, MEO, and PO have success rates between 50% and 80%.

# The relationship between FlightNumber and Orbit type



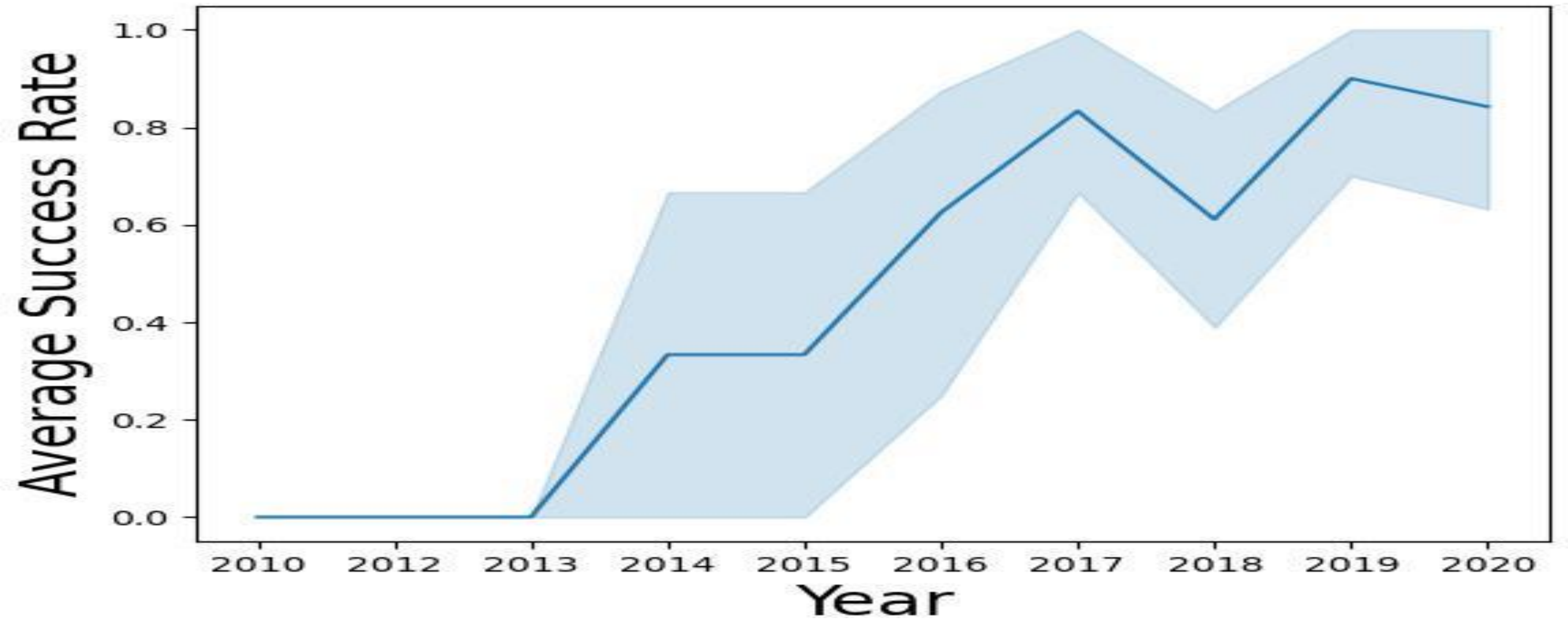

- The LEO orbit the Success appears related to the number of flights; on the other hand, there seems to be no relationship between flight number when in GTO orbit.

# The relationship between Payload and Orbit type



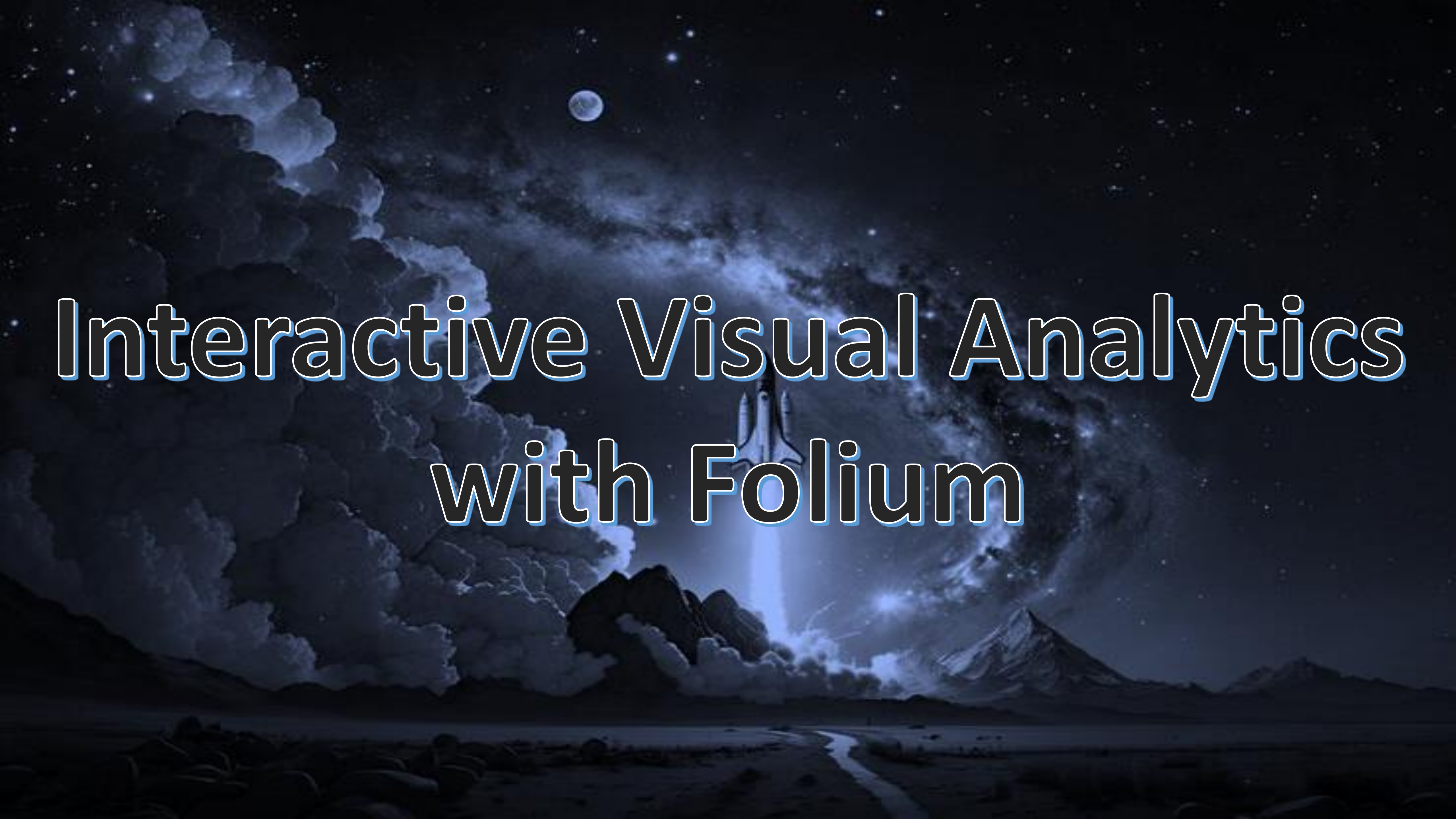

- With heavy payloads the successful landing or positive landing rate are more for Polar,LEO and ISS.
- However for GTO we cannot distinguish this well as both positive landing rate and negative landing(unsuccessful mission) are both there here.

# The launch success yearly trend

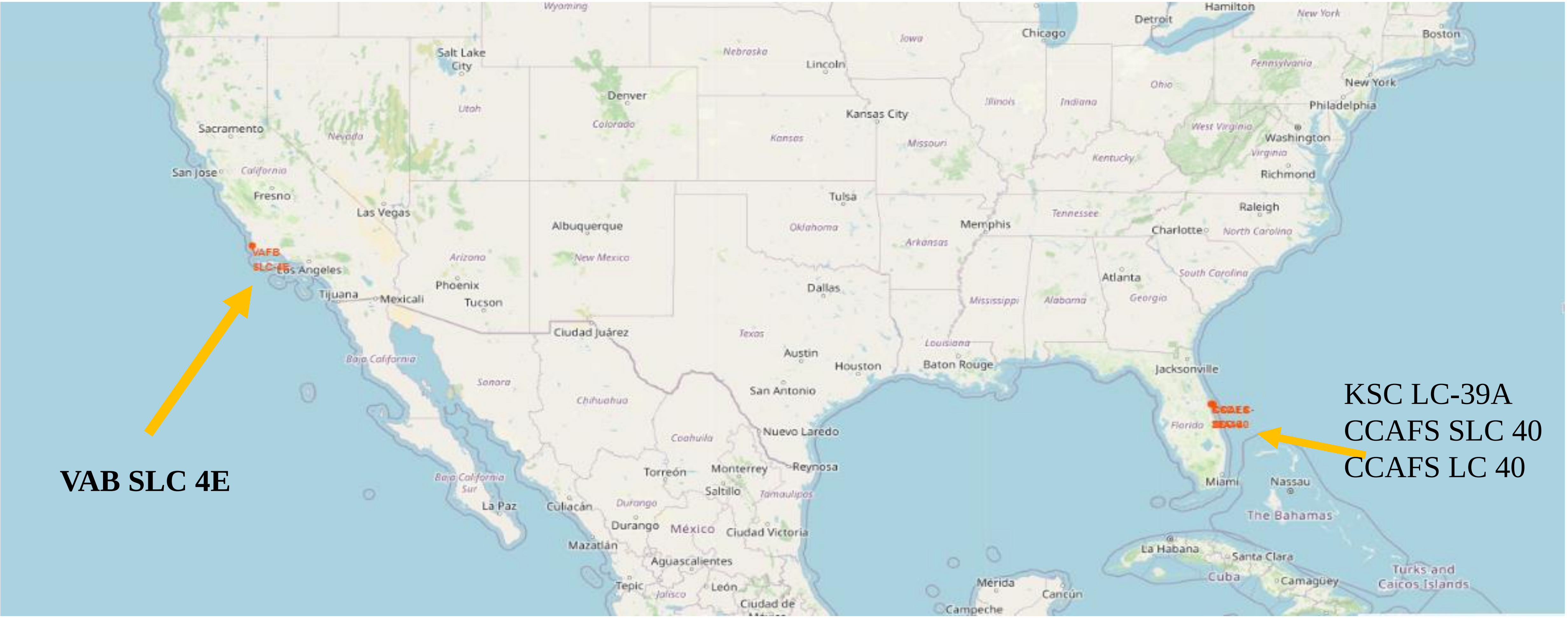



- The success rate since 2013 kept increasing till 2017 (stable in 2014) and after 2015 it started increasing.

# Interactive Visual Analytics with Folium



# All launch sites on a map



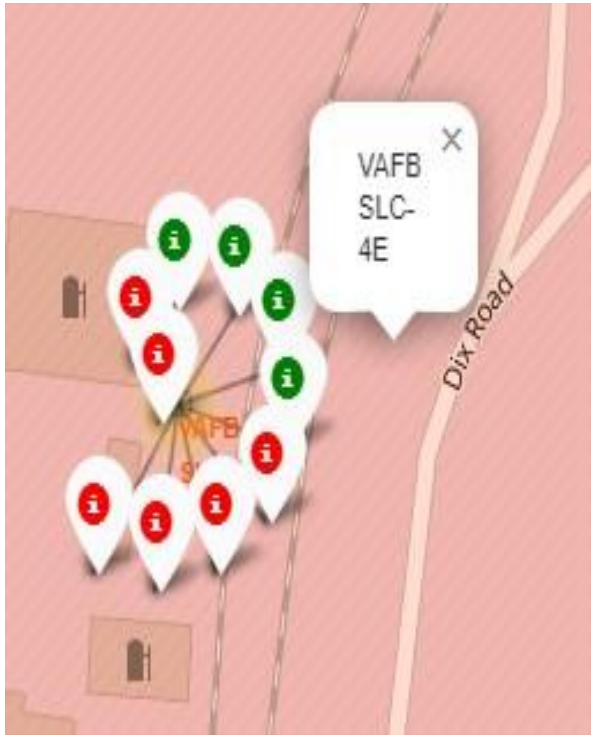

Q. Are all launch sites in proximity to the Equator line?

Ans- Yes, Space launch sites are situated near the equator to leverage the Earth's higher rotational velocity, reducing the energy required for launches. This efficiency allows rockets to carry heavier payloads and reach various orbital inclinations. Proximity to the equator also offers safety benefits by directing launch debris over open ocean areas.

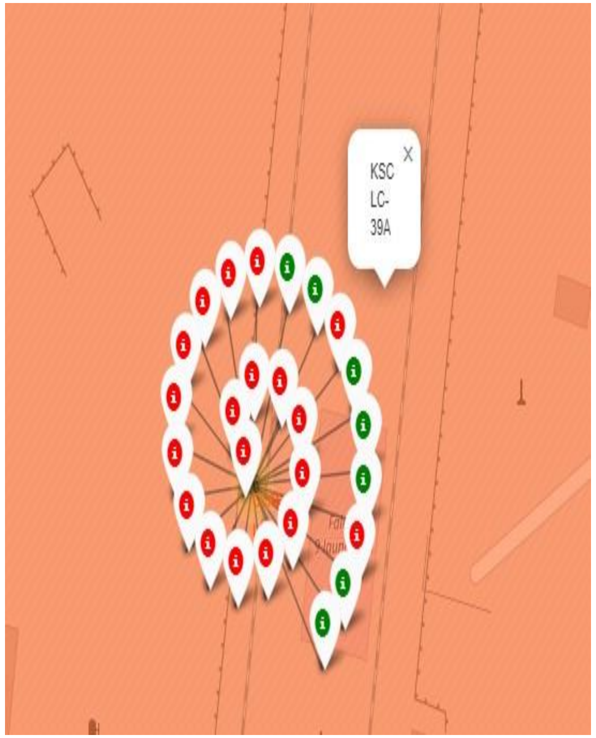
Q. Are all launch sites in very close proximity to the coast?

Ans- Yes, Space launch sites near coasts provide clear, unpopulated paths over the ocean, minimizing risks in case of launch failures. Coastal locations aid logistical efficiency by allowing transport of rockets and components by sea.

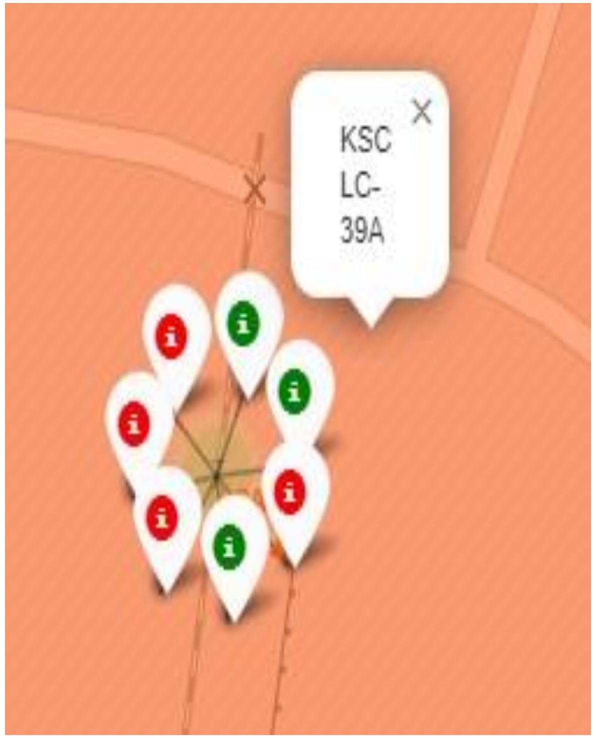
# Mark the success/failed launches for each site on the map




VAFB SLC-40




CCAFS LC 40



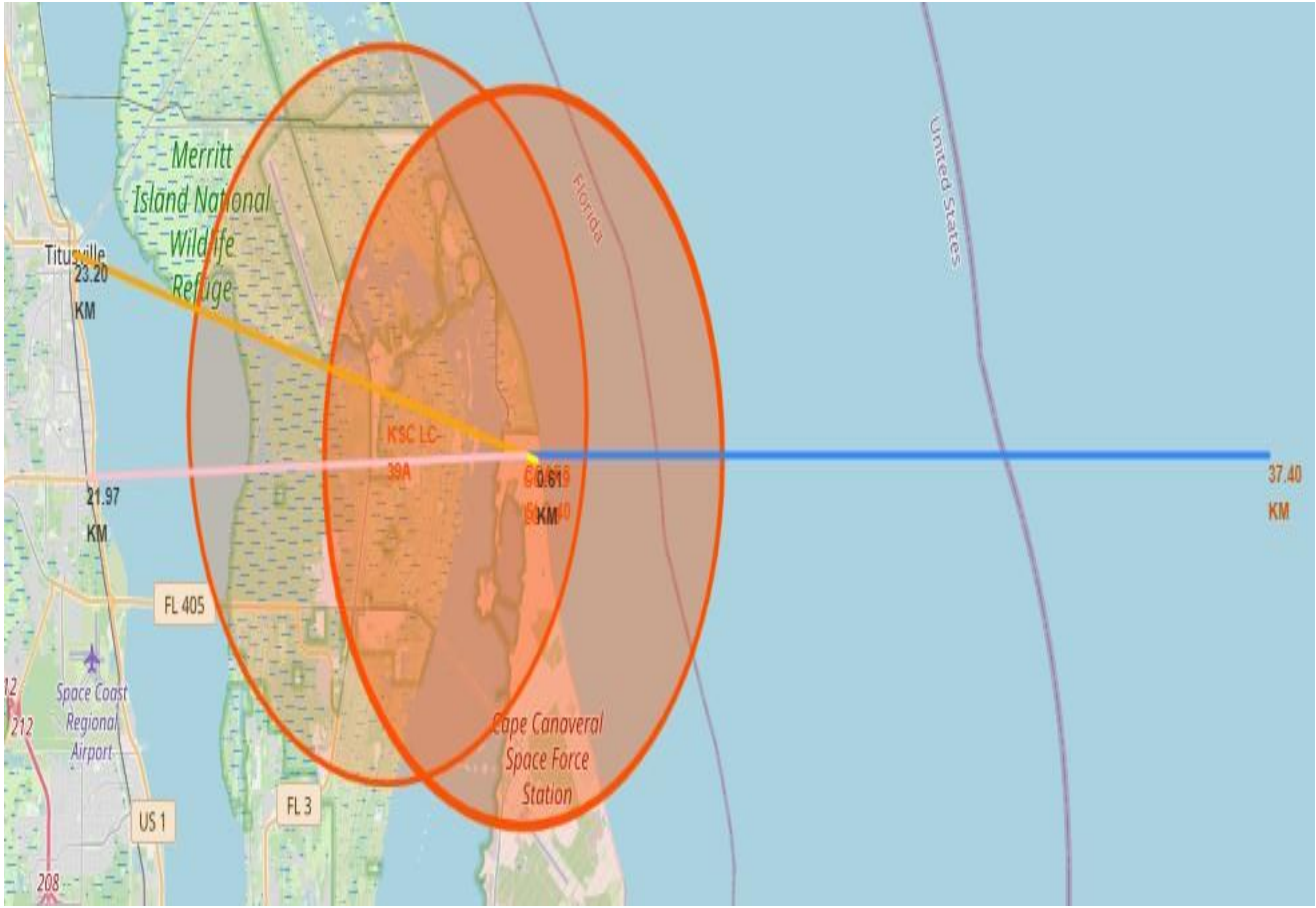

CCAFS SLC 40


KSC LC-39A

Note-

- The red marker failed launches and Green markers are Successful launches.
- Because of overlapping marker circles the marker may show wrong launch site name.

# The distances between a launch site to its proximities




| CCAFS SLC – 40 Launch Site to | Colour Line | Distance from Launch Site |
|---|---|---|
| Sea Cost | Blue Line | 0.86Km |
| Highway | Yellow Line | 0.61Km |
| City | Orange Line | 23.20Km |
| Railway Route | Pink Line | 21.97 Km |

# The distances between a launch site to its proximities

After you plot distance lines to the proximities, you can answer the following questions easily:

- 1. Are launch sites in close proximity to railways?

Ans- No, `21.97km Floroda East Cost Railway` from CCAFS SLC-40 launch Site.

- 2. Are launch sites in close proximity to highways?

Ans- Yes, `0.61Km  Samuel C Phillips Parkway` from CCAFS SLC-40 launch Site.

- 3. Are launch sites in close proximity to coastline?

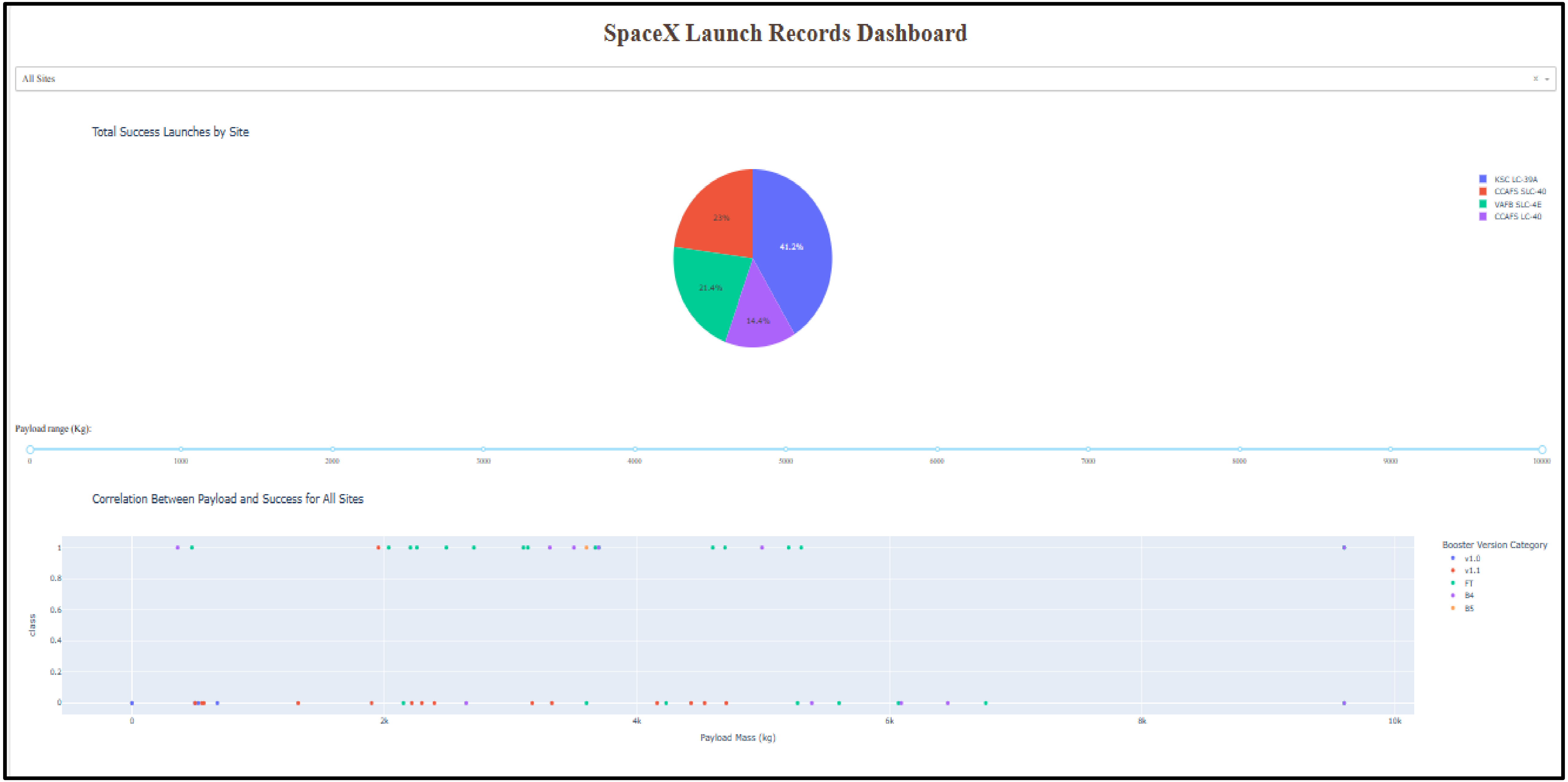
Ans- Yes, `0.86Km  Coastline` from CCAFS SLC-40 launch Site.

- 4. Do launch sites keep certain distance away from cities?

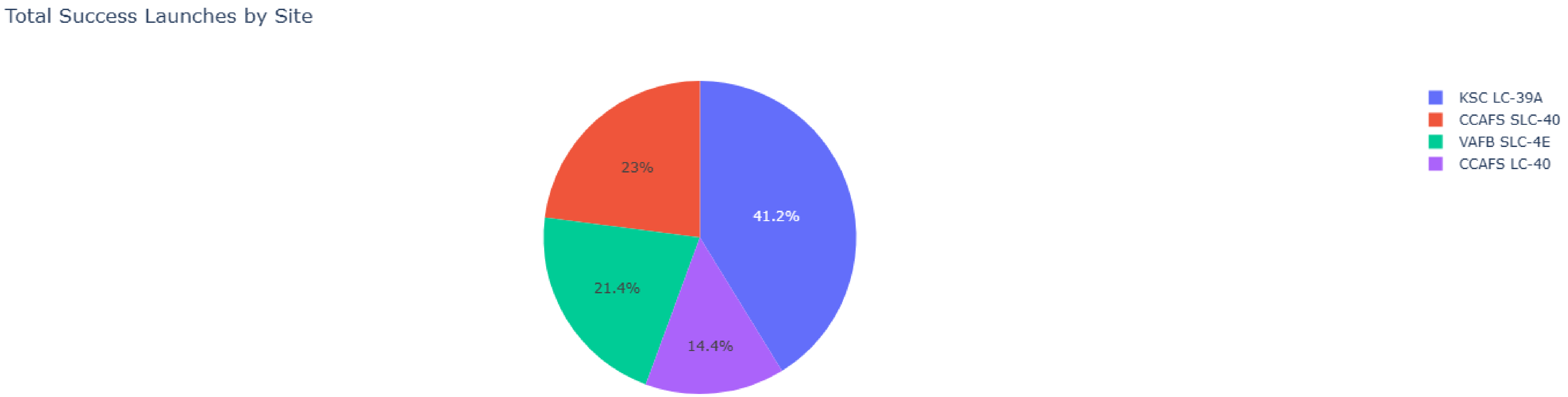
Ans- Yes, `23.20km Titusville` from CCAFS SLC-40 launch Site.

# Interactive Dashboard with Plotly and Dash

# SpaceX Launch Records Dashboard



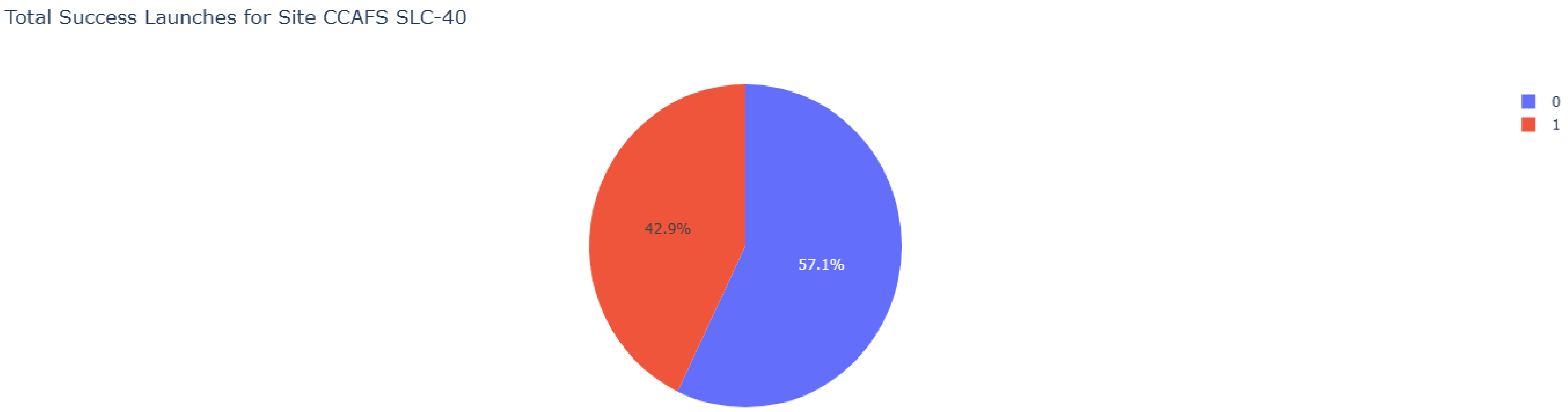
# The largest successful launches




Q. Which site has the largest successful launches?

Ans- KSC LC-39A

# The highest launch success rate




Q. Which site has the highest launch success rate?

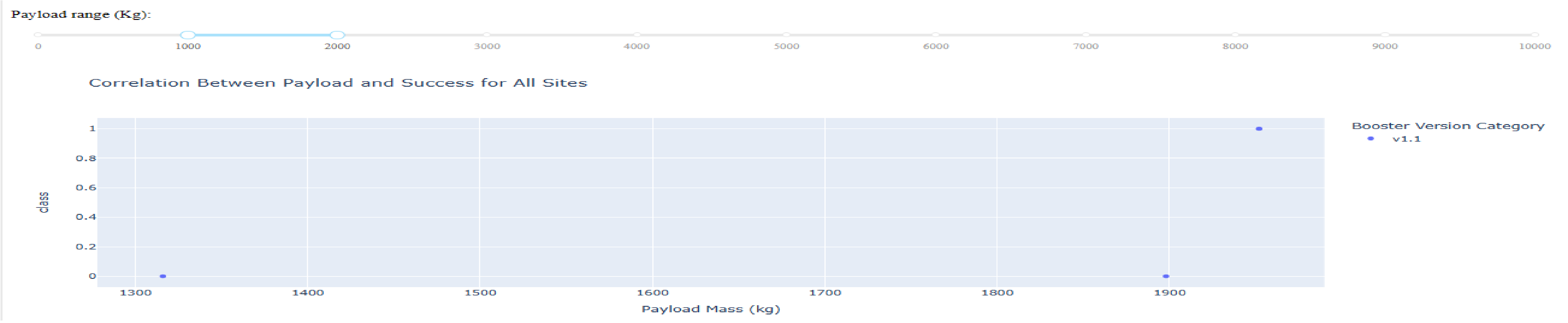
Ans- CCAFS SLC-40 has the highest launch success rate 42.9%.

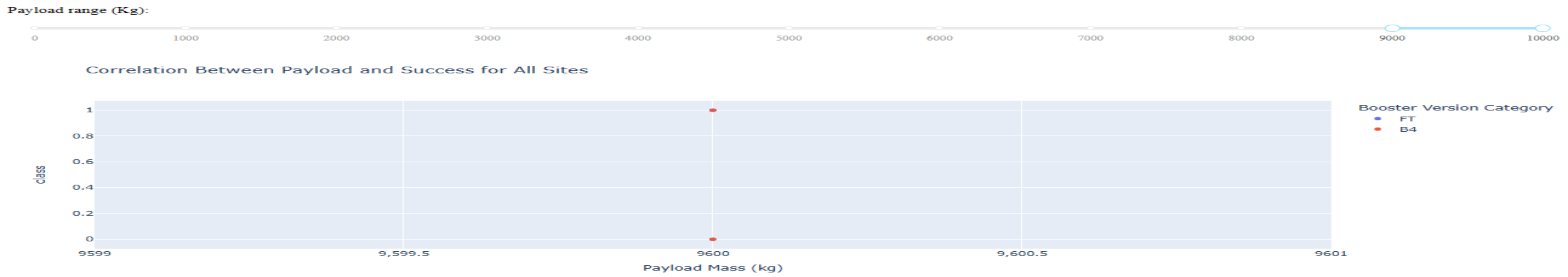
# Payload range has the highest launch success rate


Q. Which payload range(s) has the highest launch success rate?

Ans-Payload Range between 3000-4000 (kg) has the highest launch success rate with 7 successful launch.

# Payload range(s) has the lowest launch success rate

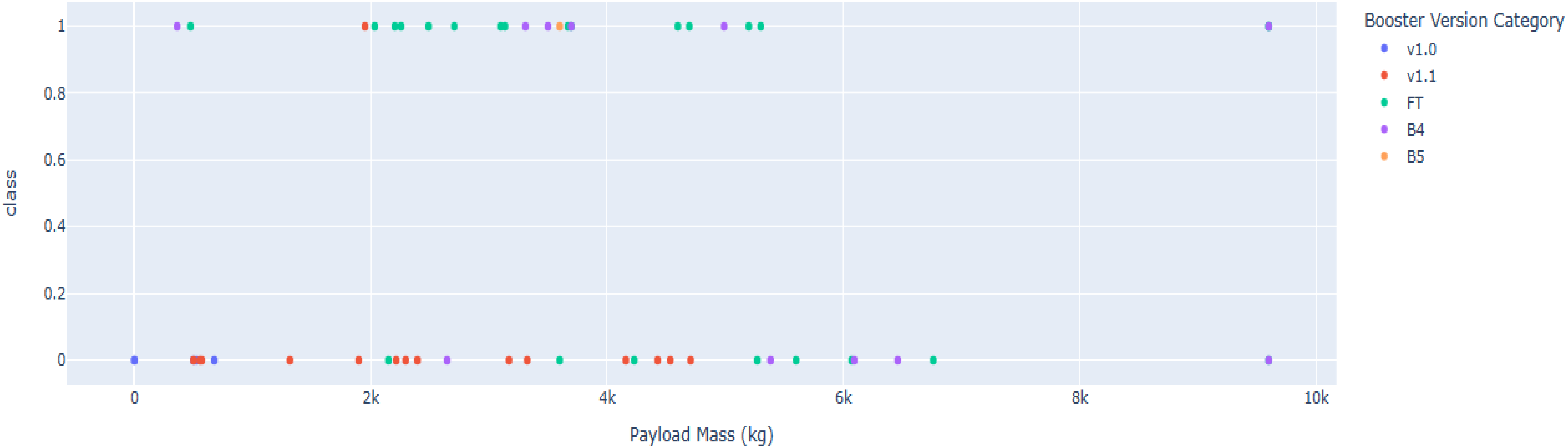



Q. Which payload range(s) has the lowest launch success rate?

Ans-Payload Range between 6000-9000 (kg) has no successful launch. Payload Range between 1000-2000 (kg) and 9000-10000 (kg) has only 1 successful launch.

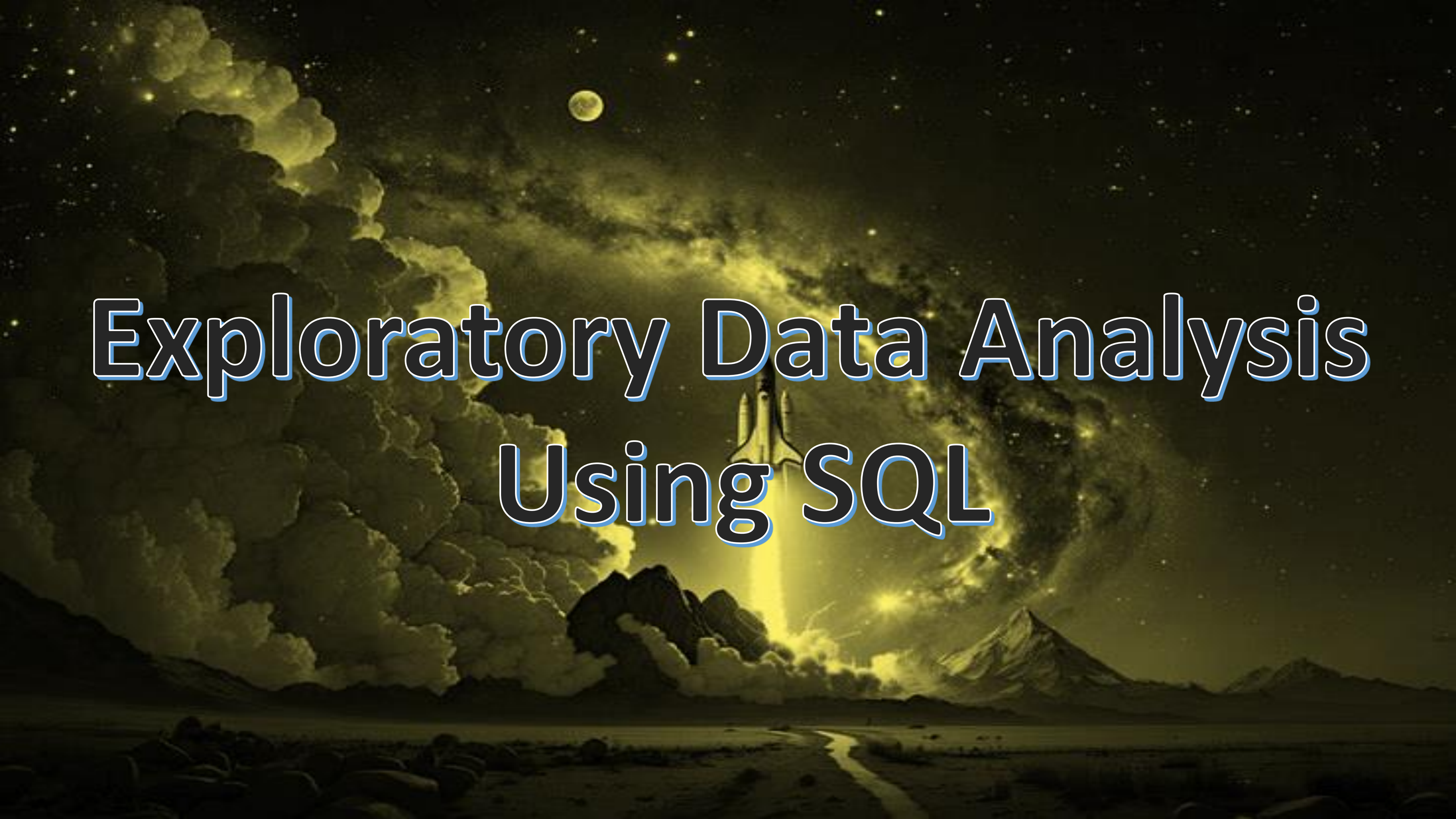
# F9 Booster version (v1.0,v1.1,FT,B4,B5, etc.) has the highest launch success rate




Q. Which F9 Booster version (v1.0,v1.1,FT,B4,B5, etc.) has the highest launch success rate?

Ans-F9 Booster version (FT) has the highest launch success rate.

# Exploratory Data Analysis Using SQL



## Q1. Display the names of the unique launch sites in the space mission

```
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 |

## Q2. Display 5 records where launch sites begin with the string 'CCA'

```
1 %sql SELECT * FROM SPACEXTABLE WHERE Launch_Site LIKE 'CCA%' limit 5
```

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

| 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 |

## Q3. Display the total payload mass carried by boosters launched by NASA (CRS)

```
%sql SELECT SUM(PAYLOAD_MASS__KG_) AS Total_PayLoad_Mass FROM SPACEXTABLE \
WHERE Customer="NASA (CRS)"
```

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

| Total_PayLoad_Mass |
|---|
| 45596 |

## Q4. Display average payload mass carried by booster version F9 v1.1¶

```
%sql SELECT AVG(PAYLOAD_MASS__KG_) AS Avg_Payloaad_Mass FROM SPACEXTABLE \
WHERE Booster_Version="F9 v1.1"
```

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

| Avg_Payloaad_Mass |
|---|
| 2928.4 |

## Q5. **List the date when the first succesful landing outcome in ground pad was acheived.**

```
%sql SELECT MIN(Date) AS First_Succesful_ground_pad_Landing_Date FROM SPACEXTABLE\
WHERE Landing_Outcome="Success (ground pad)"
```

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

| First_Succesful_ground_pad_Landing_Date |
|---|
| 2015-12-22 |

## Q6. **List the names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000.**

```
%sql SELECT Booster_Version,PAYLOAD_MASS__KG_ FROM SPACEXTABLE\
WHERE Landing_Outcome= "Success (drone ship)" \
AND PAYLOAD_MASS__KG_ BETWEEN 4000 AND 6000
```

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

| Booster_Version | PAYLOAD_MASS__KG_ |
|---|---|
| F9 FT B1022 | 4696 |
| F9 FT B1026 | 4600 |
| F9 FT B1021.2 | 5300 |
| F9 FT B1031.2 | 5200 |

# Q7. List the total number of successful and failure mission outcomes

```
%sql SELECT\
(SELECT COUNT(Mission_Outcome) FROM SPACEXTABLE WHERE Mission_Outcome LIKE '%Success%')\
AS Total_Successful_Mission_Outcome,\
(SELECT COUNT(Mission_Outcome) FROM SPACEXTABLE WHERE Mission_Outcome LIKE '%Failure%')\
AS Total_Failure_Mission_Outcome;
```

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

| Total_Successful_Mission_Outcome | Total_Failure_Mission_Outcome |
|---|---|
| 100 | 1 |

```
%sql SELECT Mission_Outcome, COUNT(*) AS Total_Mission_Outcome \
FROM SPACEXTABLE \
GROUP BY Mission_Outcome;
```

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

| Mission_Outcome | Total_Mission_Outcome |
|---|---|
| Failure (in flight) | 1 |
| Success | 98 |
| Success | 1 |
| Success (payload status unclear) | 1 |

## Q8. **List the names of the booster_versions which have carried the maximum payload mass. Use a subquery**

```
%sql SELECT BOOSTER_VERSION,(PAYLOAD_MASS__KG_) AS Max_Payload_Mass  FROM SPACEXTABLE \
WHERE PAYLOAD_MASS__KG_ = (SELECT MAX(PAYLOAD_MASS__KG_) FROM SPACEXTABLE);
```

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

| Booster_Version | Max_Payload_Mass |
|---|---|
| F9 B5 B1048.4 | 15600 |
| F9 B5 B1049.4 | 15600 |
| F9 B5 B1051.3 | 15600 |
| F9 B5 B1056.4 | 15600 |
| F9 B5 B1048.5 | 15600 |
| F9 B5 B1051.4 | 15600 |
| F9 B5 B1049.5 | 15600 |
| F9 B5 B1060.2 | 15600 |
| F9 B5 B1058.3 | 15600 |
| F9 B5 B1051.6 | 15600 |
| F9 B5 B1060.3 | 15600 |
| F9 B5 B1049.7 | 15600 |

Q9. **List the records which will display the month names, failure landing_outcomes in drone ship ,booster versions, launch_site for the months in year 2015.**

```
%sql SELECT substr(Date,6,2) as month,Date,Booster_Version, Launch_Site, Landing_Outcome\
FROM SPACEXTABLE\
where Landing_Outcome= 'Failure (drone ship)' and substr(Date,0,5)='2015';
```

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

| month | Date | Booster_Version | Launch_Site | Landing_Outcome |
|---|---|---|---|---|
| 01 | 2015-01-10 | F9 v1.1 B1012 | CCAFS LC-40 | Failure (drone ship) |
| 04 | 2015-04-14 | F9 v1.1 B1015 | CCAFS LC-40 | Failure (drone ship) |

**Q10. Rank 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.**

```
%sql  SELECT Landing_Outcome, COUNT(*) AS Total_Successful_Landing_Outcome FROM SPACEXTABLE\
WHERE Landing_Outcome LIKE '%Success%' AND DATE BETWEEN '2010-06-04' AND '2017-03-20'\
GROUP BY Landing_Outcome\
ORDER BY Total_Successful_Landing_Outcome DESC
```

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

| Landing_Outcome | Total_Successful_Landing_Outcome |
|---|---|
| Success (drone ship) | 5 |
| Success (ground pad) | 3 |

# Predictive Analysis

- A machine learning pipeline is employed to predict whether the first stage will successfully land based on the provided data.
- Classification algorithms, including Logistic Regression, Support Vector Machine (SVM), Decision Tree, and K Nearest Neighbour, are utilized for this prediction.
- The GridSearchCV method with a cross-validation value of 10 is applied to identify the best parameters for each algorithm.
- The accuracy on the validation data is assessed using the data attributes for all algorithms.
- Among the algorithms, the Decision Tree algorithm exhibits slightly superior performance in terms of accuracy on the validation data.

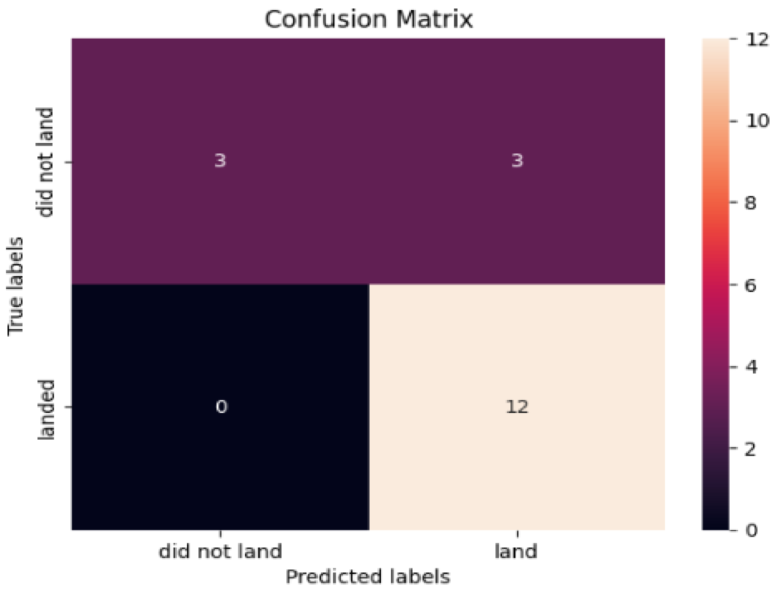
| ML Algorithm | Accuracy Score Test (%) | Accuracy (%) | Best Parameter |
|---|---|---|---|
| Decision Tree | 83.333333 | 87.678571 | {'criterion': 'entropy', 'max_depth': 18, 'max_features': 'sqrt', 'min_samples_leaf': 4, 'min_samples_split': 10, 'splitter': 'random'} |

| ML Algorithm | Accuracy Score Test (%) | Accuracy (%) | Best Parameter |
|---|---|---|---|
| Support Vector Machine | 83.333333 | 84.821429 | {'C': 1.0, 'gamma': 0.03162277660168379, 'kern... |
| Logistic Regression | 83.333333 | 84.642857 | {'C': 0.01, 'penalty': 'l2', 'solver': 'lbfgs'} |
| K Nearest Neighbour | 83.333333 | 84.821429 | {'algorithm': 'auto', 'n_neighbors': 10, 'p': 1} |
| Decision Tree | 83.333333 | 87.678571 | {'criterion': 'entropy', 'max_depth': 18, 'max... |

The Decision Tree algorithm stands out as the best predictor among the various algorithms, demonstrating superior accuracy on the validation data.

| ML Algorithm | Accuracy Score Test (%) | Accuracy (%) | Best Parameter |
|---|---|---|---|
| Decision Tree | 83.333333 | 87.678571 | {'criterion': 'entropy', 'max_depth': 18, 'max_features': 'sqrt', 'min_samples_leaf': 4, 'min_samples_split': 10, 'splitter': 'random'} |

# DECISION TREE CLASSIFICATION CONFUSION MATRIX




| | |
|---|---|
| True Negative (TN) = 3 | False Positive (FP) = 3 |
| False Negative (FN) =0 | True Positive (TP) =12 |

| | | Predicted | |
|---|---|---|---|
| | | Negative (N) - | Positive (P) + |
| Actual | Negative - | True Negative (TN) | False Positive (FP) Type I Error |
| | Positive + | False Negative (FN) Type II Error | True Positive (TP) |

# CONFUSION MATRIX OUTPUT

In a confusion matrix, a false positive (FP) occurs when the model predicts the positive class, but the actual class is negative. In other words, the model incorrectly identifies instances as positive when they are, in fact, negative. This situation is also known as a Type I error.

# Conclusions

- As the flight number increases from the launch sites, the success rate also improves. Initially, the success rate is poor.
- The higher the payload, the higher the success rate.
- Most payloads weighing more than 12000 kg are launched from CCAFS SLC 40 launch site.
- VAFB-SLC launch site there are no rockets launched for heavy payload mass(greater than 10000).
- ES-L1, GEO, HEO, and SSO have a 100% success rate.
- SO has a 0% success rate.
- GTO, ISS, LEO, MEO, and PO have success rates between 50% and 80%.
- Space launch sites are situated near the equator to leverage the Earth's higher rotational velocity, reducing the energy required for launches. This efficiency allows rockets to carry heavier payloads and reach various orbital inclinations. Proximity to the equator also offers safety benefits by directing launch debris over open ocean areas.
- Space launch sites near coasts provide clear, unpopulated paths over the ocean, minimizing risks in case of launch failures. Coastal locations aid logistical efficiency by allowing transport of rockets and components by sea.
- The Decision Tree algorithm stands out as the best predictor among the various algorithms, demonstrating superior accuracy on the validation data.

# Insights

- Launch Site: Choose equatorial sites for efficient launches and heavier payloads. Opt for coastal locations to ensure safety and streamline logistics.
- Payload Optimization: Focus on higher payloads for better success rates. CCAFS SLC 40 is suitable for payloads >12000 kg; avoid VAFB-SLC for payloads >10000 kg.
- Orbital Success: Target high success rate orbits like ES-L1, GEO, HEO, and SSO (100% success). Avoid SO orbit with 0% success.
- Algorithmic Prediction: Use Decision Tree algorithm for accurate predictive modeling, considering factors like launch site, payload weight, and orbital destination.
- Risk Mitigation: Manage risks for orbits like GTO, ISS, LEO, MEO, and PO (50-80% success) with robust strategies, including redundancy, thorough testing, and continuous improvement.
- Logistics Efficiency: Optimize logistics with coastal launch sites, facilitating sea transport for rockets and components, enhancing operational efficiency, and minimizing risks in populated areas.

# Project Files Links

| CONTENT | FILE LINKS |
|---|---|
| Project Repository | GitHub Repository Link |
| Data Collection API | Data collection from API file link |
| Data Collection Web Scrapping | Web Scrapping file link |
| Data Wrangling | Data Wrangling File Link |
| Exploratory Data Analysis – SQL | SQL FILE LINK |
| Exploratory Data Analysis – Visualization | EDA Visualization File Link |
| Visual Analytics with Folium | Interactive Map with Folium File Link |
| Interactive Space X Dashboard - Dash | Interactive Dashboard with plotly dash file link |
| Predictive Analysis – Machine Learning | Predictive analysis (Machine Learning) File Link |