Winning Space Race with Data Science

Santiago Semensi

01/26/24

Outline

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

Executive summary

Methodology

Data Collection | Data Wrangling | Exploratory Data Analysis | Interactive Visualization | Predictive Modeling

Key Results and Insights

Historical Success Rate | Trend of Improvement | Influential Factors | Predictive Model Accuracy

Introduction

Context

Problem Statement

Project Goal

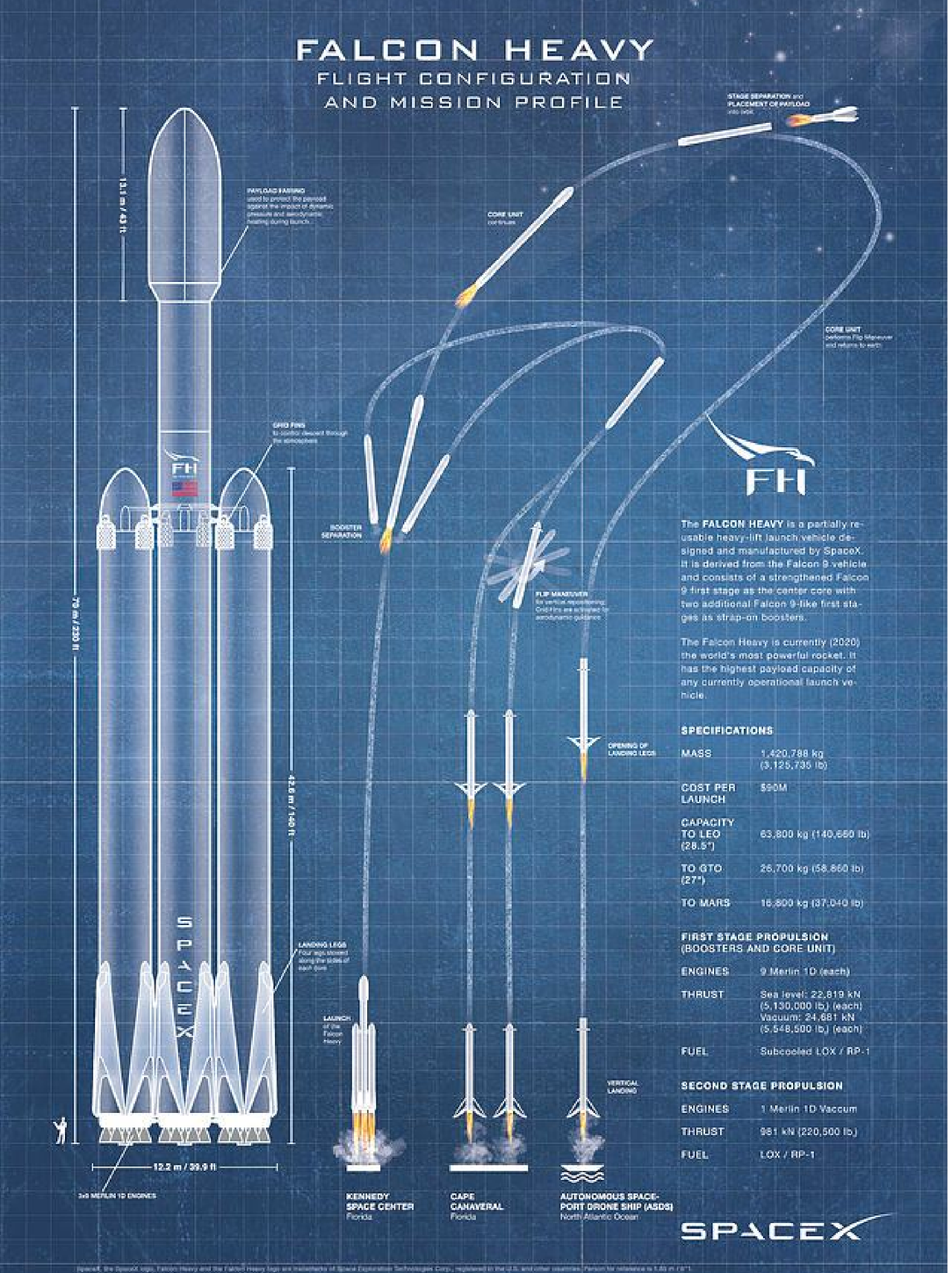
Introduction

Context

We are at the forefront of the commercial space age, a transformative era where space travel is becoming more accessible and affordable. Many companies are revolutionizing space exploration with innovative technologies. Among these pioneers, **SpaceX stands out** for its remarkable achievements, from sending spacecraft to the International Space Station and launching Starlink for global Internet access, to conducting manned space missions.

Problem Statement

Project Goal



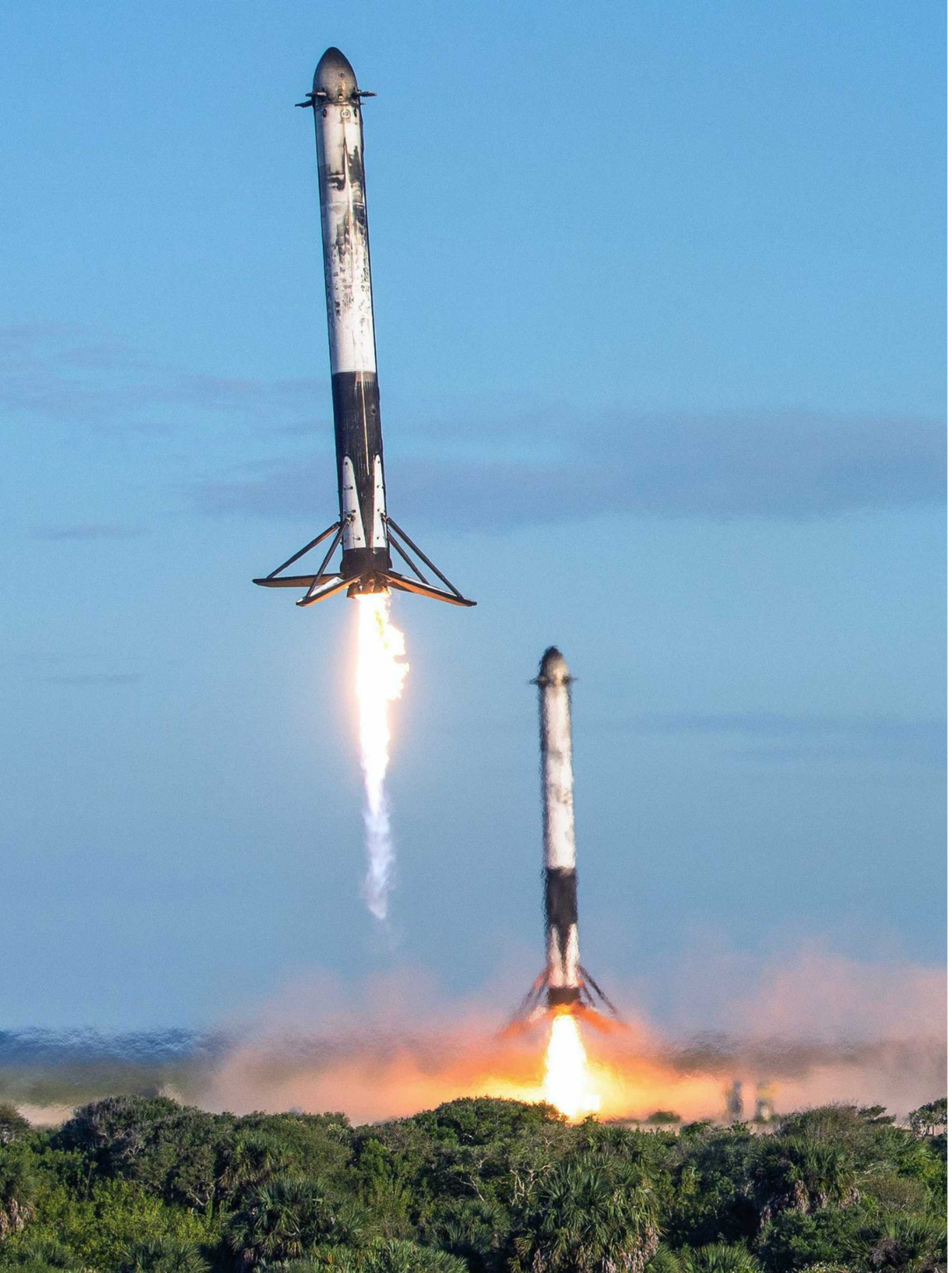
Introduction

Context

Problem Statement

A key factor in SpaceX's success is its **cost-effective** approach to space launches. The Falcon 9 rocket, advertised at $62 million per launch, dramatically undercuts the competition, which can exceed $165 million. This price advantage largely stems from the groundbreaking ability to **reuse the first stage of the rocket**, a feat that once seemed like science fiction. In this context, accurately predicting the success of Falcon 9 first stage landings becomes crucial.

Project Goal



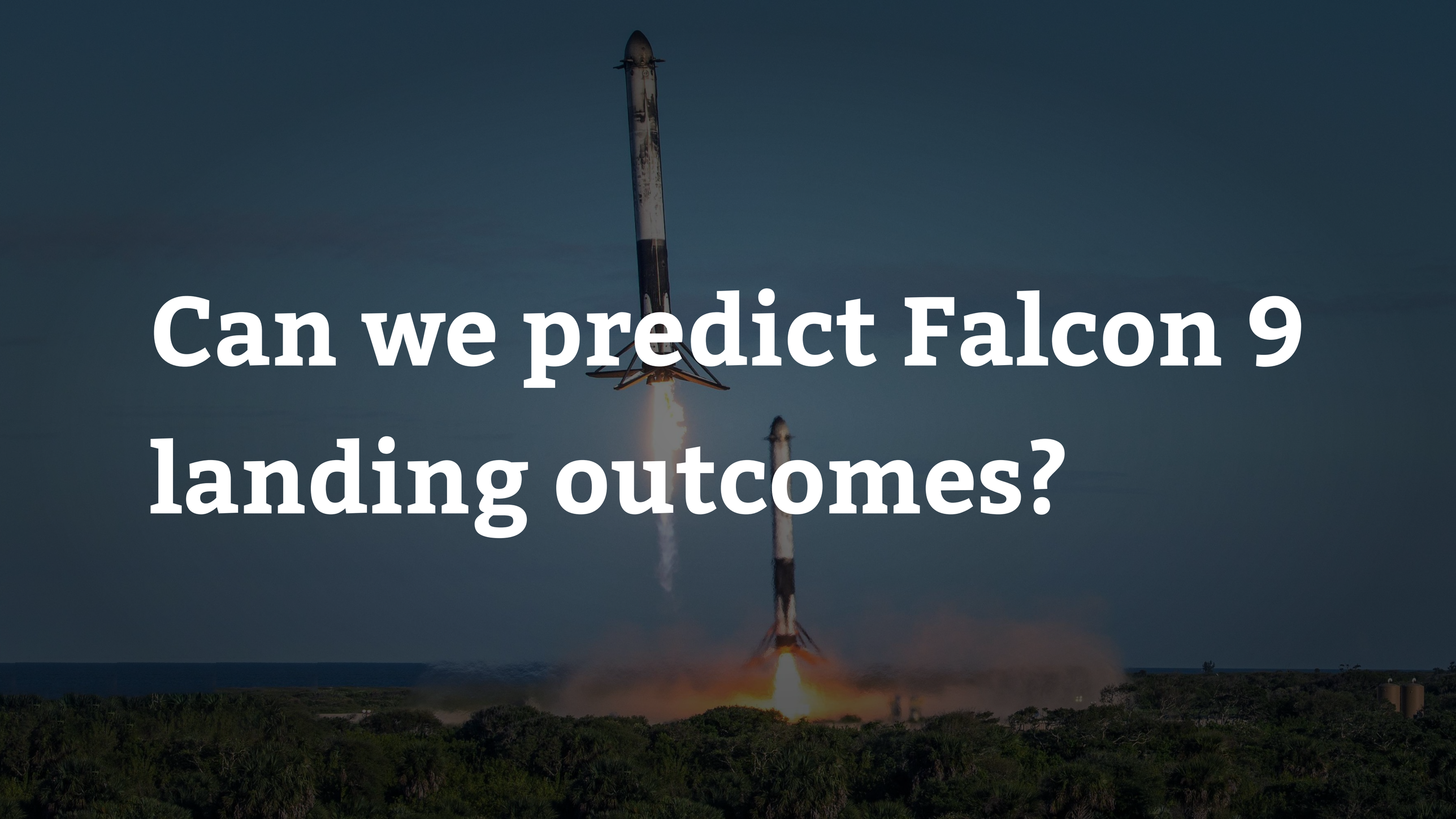
Introduction

Context

Problem Statement

Project Goal

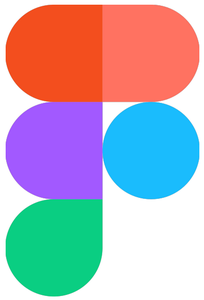
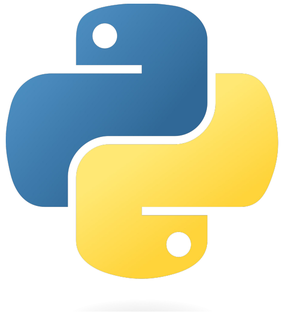
This project aims to employ data science and machine learning to predict whether SpaceX's Falcon 9 first stage will land successfully. This capability is not just a technical challenge but also holds profound financial and strategic implications in the competitive space travel industry. By accurately **forecasting landing outcomes**, we can enhance the economics of space launches and further drive innovation in this transformative era.



Can we predict Falcon 9 landing outcomes?


We’ll get to that answer later…

What tools did we use?




Methodology

Data Collection

Data Wrangling

Exploratory Data Analysis

Interactive Visualization

Predictive Modeling

Compared two different methods

SpaceX API

Source is more reliable

Some data needs to be decoded

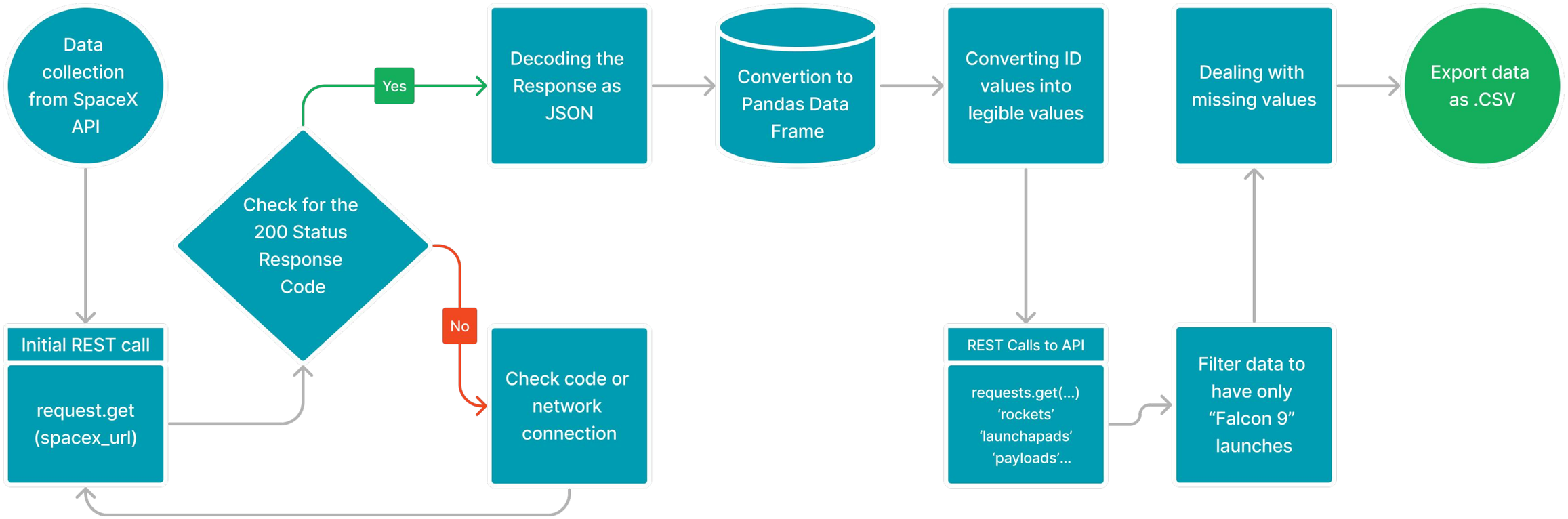
Web Scraping - Wikipedia

Data is stored in a table

Might contain noise

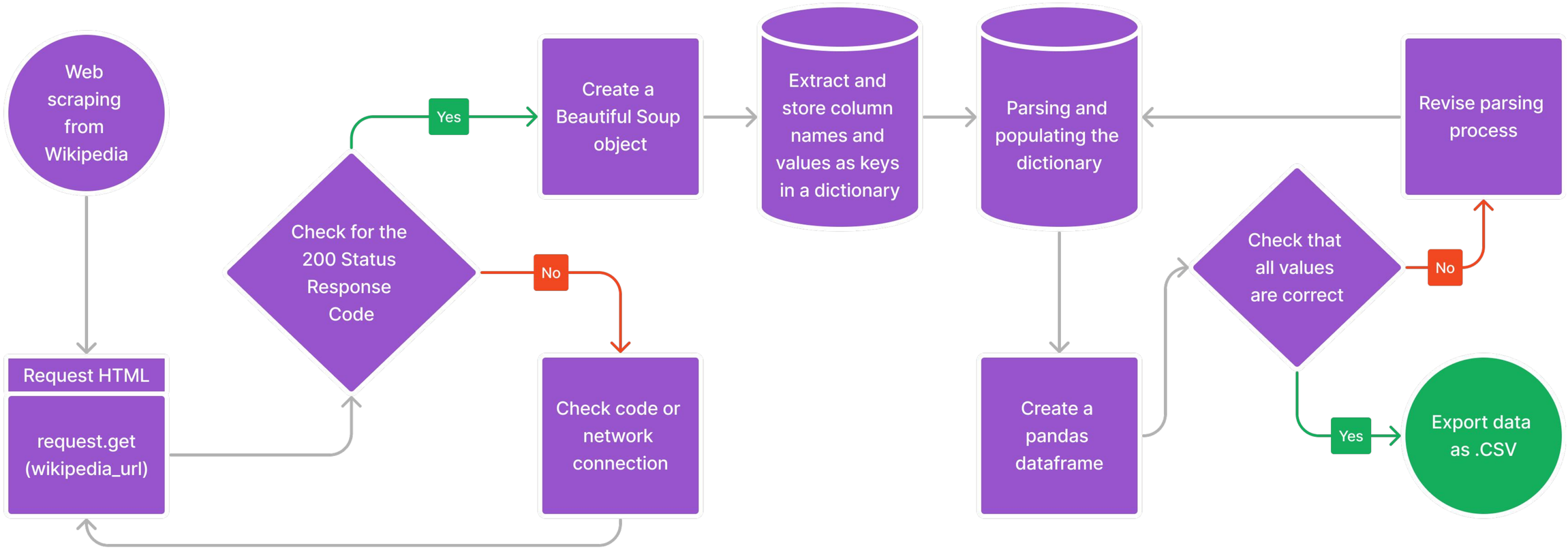
Need to extract and parse

SpaceX API



To see the complete notebook please click on this link.

Web Scraping



To see the complete notebook please click on this link.

Conclusion

SpaceX API seems to be the best method of **data collection**, despite the need to decode and transform ID’s into legible data, it comes straight from the company and that makes it more reliable.

Methodology

Data Collection

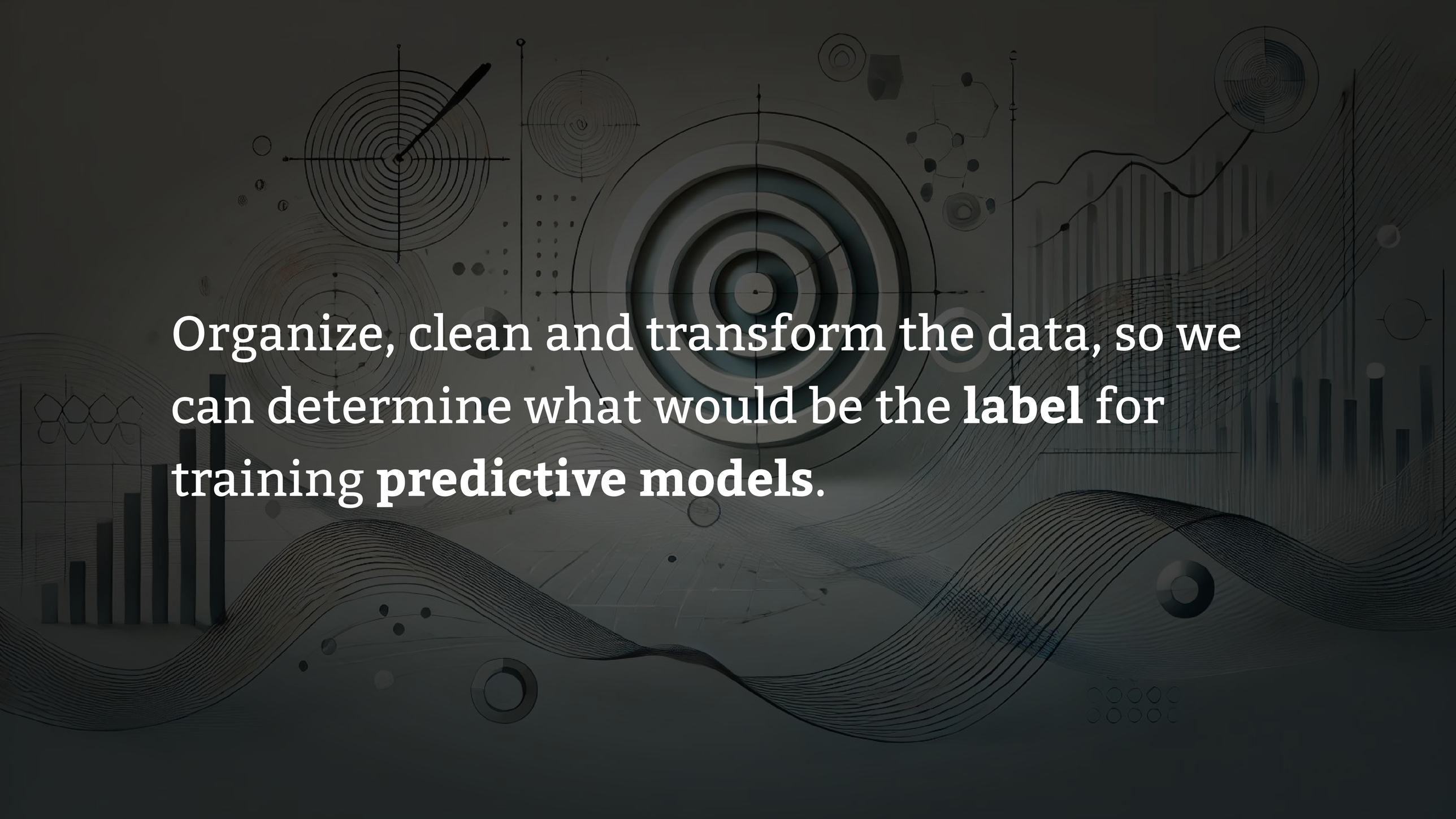
Data Wrangling

Exploratory Data Analysis

Interactive Visualization

Predictive Modeling


What is our goal in this step?


Organize, clean and transform the data, so we can determine what would be the **label** for training **predictive models**.

Taking a look at the dataset

```
import pandas as pd
df=pd.read_csv("https://cf-courses-data.s3.us.cloud...)
df.head(6)
```

	FlightNumber	Date	BoosterVersion	PayloadMass	Orbit	LaunchSite	Outcome	Flights	GridFins	Reused	Legs	LandingPad	Block	ReusedCount	Serial	Longitude	Latitude
0	1	2010-06-04	Falcon 9	6104.959412	LEO	CCAFS SLC 40	None None	1	False	False	False	NaN	1.0	0	B0003	-80.577366	28.561857
1	2	2012-05-22	Falcon 9	525.000000	LEO	CCAFS SLC 40	None None	1	False	False	False	NaN	1.0	0	B0005	-80.577366	28.561857
2	3	2013-03-01	Falcon 9	677.000000	ISS	CCAFS SLC 40	None None	1	False	False	False	NaN	1.0	0	B0007	-80.577366	28.561857
3	4	2013-09-29	Falcon 9	500.000000	PO	VAFB SLC 4E	False Ocean	1	False	False	False	NaN	1.0	0	B1003	-120.610829	34.632093
4	5	2013-12-03	Falcon 9	3170.000000	GTO	CCAFS SLC 40	None None	1	False	False	False	NaN	1.0	0	B1004	-80.577366	28.561857
5	6	2014-01-06	Falcon 9	3325.000000	GTO	CCAFS SLC 40	None None	1	False	False	False	NaN	1.0	0	B1005	-80.577366	28.561857

Read dataset and see top 6 rows

```
df.isnull().sum()/len(df)*100
```

```
FlightNumber        0.000000
Date                0.000000
BoosterVersion      0.000000
PayloadMass         0.000000
Orbit               0.000000
LaunchSite          0.000000
Outcome             0.000000
Flights             0.000000
GridFins            0.000000
Reused              0.000000
Legs                0.000000
LandingPad         28.888889
Block               0.000000
ReusedCount         0.000000
Serial              0.000000
Longitude           0.000000
Latitude            0.000000
dtype: float64
```

28% didn't use a landing pad

Check for missing values. Everything seems OK!

```
landing_outcomes=df['Outcome'].value_counts()
landing_outcomes
```

```
True ASDS      41
None None      19
True RTLS      14
False ASDS      6
True Ocean      5
False Ocean     2
None ASDS       2
False RTLS      1
Name: Outcome, dtype: int64
```

Determine the number of landing outcomes.

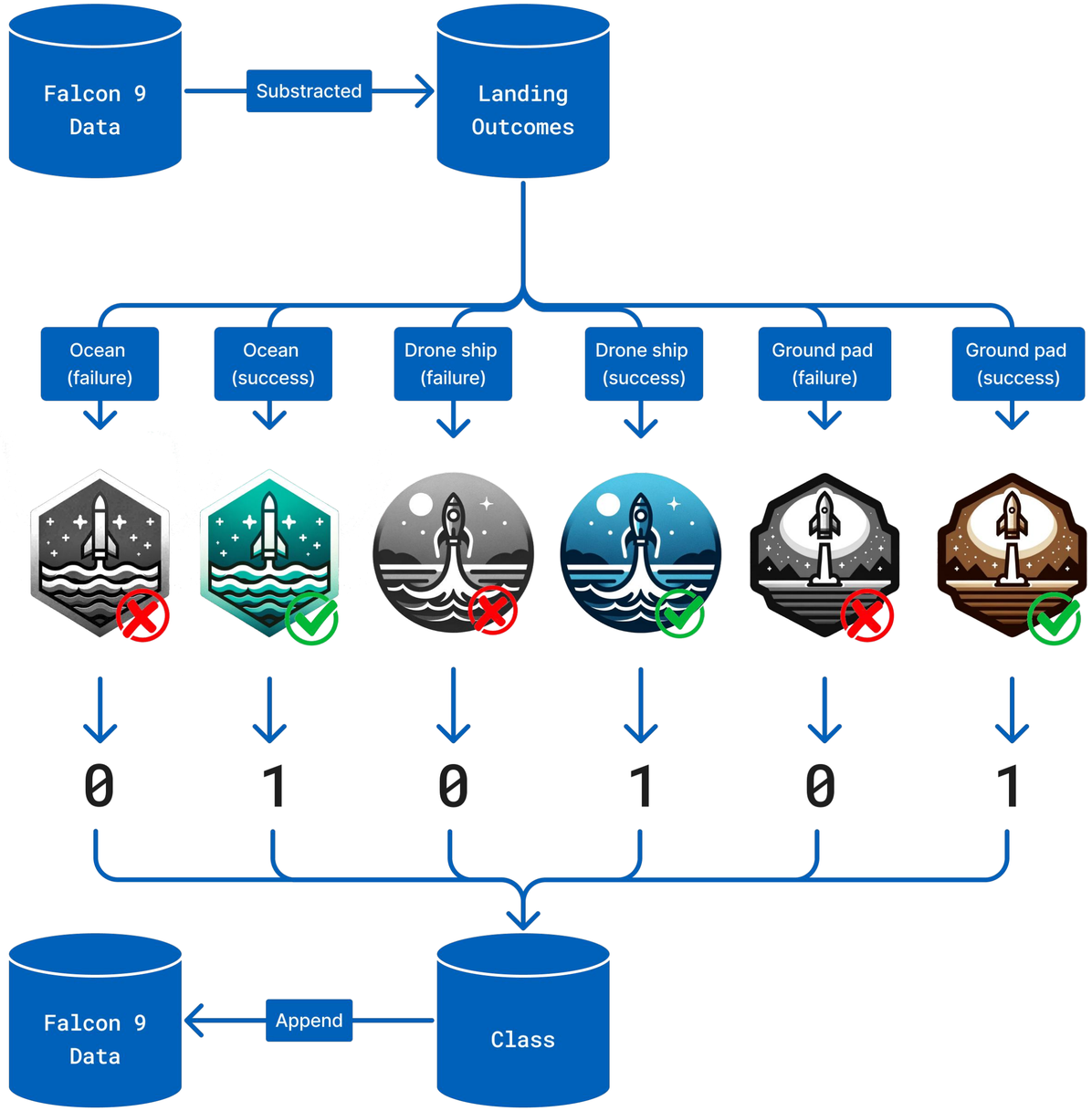
```
landing_outcomes=df['Outcome'].value_counts()
landing_outcomes
```

```
True ASDS      41
None None      19
True RTLS      14
False ASDS      6
True Ocean      5
False Ocean     2
None ASDS       2
False RTLS      1
Name: Outcome, dtype: int64
```

All of the **True** values mean the mission outcome was successful, and all of the **False** or **None** values mean the outcomes were unsuccessful. 'Ocean' targets a specific region of the ocean for the landing, 'RTLS' aims for a ground pad, and 'ASDS' a drone ship.

We performed **binary encoding** on the landing outcomes, creating a separate column "**Class**". This will be the target variable.

Click to see Notebook.



We can see the Python code for the binary encoding process and later export to .csv

```
bad_outcomes = {'False ASDS',
                'False Ocean',
                'False RTLS',
                'None ASDS',
                'None None'}

df['Class'] = df['Outcome'].apply(lambda x: 0 if x in bad_outcomes else 1)

df.to_csv("dataset_part_2.csv", index=False)
```

Methodology

Data Collection

Data Wrangling

Exploratory Data Analysis

Interactive Visualization

Predictive Modeling

Different **relationships between variables** were explored, giving us insights about the data that helped us build a predictive model.

Let’s start asking some questions

When was the first successful landing, and what is the success rate over time?

	FlightNumber	Date	BoosterVersion	PayloadMass	Orbit	LaunchSite	Outcome	Flights	GridFins	Reused	Legs	LandingPad	Block	ReusedCount	Serial	Longitude	Latitude	Class
0	1	2010-06-04	Falcon 9	6104.959412	LEO	CCAFS SLC 40	None None	1	False	False	False	NaN	1.0	0	B0003	-80.577366	28.561857	0
1	2	2012-05-22	Falcon 9	525.000000	LEO	CCAFS SLC 40	None None	1	False	False	False	NaN	1.0	0	B0005	-80.577366	28.561857	0
2	3	2013-03-01	Falcon 9	677.000000	ISS	CCAFS SLC 40	None None	1	False	False	False	NaN	1.0	0	B0007	-80.577366	28.561857	0
3	4	2013-09-29	Falcon 9	500.000000	PO	VAFB SLC 4E	False Ocean	1	False	False	False	NaN	1.0	0	B1003	-120.610829	34.632093	0
4	5	2013-12-03	Falcon 9	3170.000000	GTO	CCAFS SLC 40	None None	1	False	False	False	NaN	1.0	0	B1004	-80.577366	28.561857	0

How does the payload capacity increase over time?

	FlightNumber	Date	BoosterVersion	PayloadMass	Orbit	LaunchSite	Outcome	Flights	GridFins	Reused	Legs	LandingPad	Block	ReusedCount	Serial	Longitude	Latitude	Class
0	1	2010-06-04	Falcon 9	6104.959412	LEO	CCAFS SLC 40	None None	1	False	False	False	NaN	1.0	0	B0003	-80.577366	28.561857	0
1	2	2012-05-22	Falcon 9	525.000000	LEO	CCAFS SLC 40	None None	1	False	False	False	NaN	1.0	0	B0005	-80.577366	28.561857	0
2	3	2013-03-01	Falcon 9	677.000000	ISS	CCAFS SLC 40	None None	1	False	False	False	NaN	1.0	0	B0007	-80.577366	28.561857	0
3	4	2013-09-29	Falcon 9	500.000000	PO	VAFB SLC 4E	False Ocean	1	False	False	False	NaN	1.0	0	B1003	-120.610829	34.632093	0
4	5	2013-12-03	Falcon 9	3170.000000	GTO	CCAFS SLC 40	None None	1	False	False	False	NaN	1.0	0	B1004	-80.577366	28.561857	0

What is the relation between payload and the orbit choice?

	FlightNumber	Date	BoosterVersion	PayloadMass	Orbit	LaunchSite	Outcome	Flights	GridFins	Reused	Legs	LandingPad	Block	ReusedCount	Serial	Longitude	Latitude	Class
0	1	2010-06-04	Falcon 9	6104.959412	LEO	CCAFS SLC 40	None None	1	False	False	False	NaN	1.0	0	B0003	-80.577366	28.561857	0
1	2	2012-05-22	Falcon 9	525.000000	LEO	CCAFS SLC 40	None None	1	False	False	False	NaN	1.0	0	B0005	-80.577366	28.561857	0
2	3	2013-03-01	Falcon 9	677.000000	ISS	CCAFS SLC 40	None None	1	False	False	False	NaN	1.0	0	B0007	-80.577366	28.561857	0
3	4	2013-09-29	Falcon 9	500.000000	PO	VAFB SLC 4E	False Ocean	1	False	False	False	NaN	1.0	0	B1003	-120.610829	34.632093	0
4	5	2013-12-03	Falcon 9	3170.000000	GTO	CCAFS SLC 40	None None	1	False	False	False	NaN	1.0	0	B1004	-80.577366	28.561857	0

What's the location with the most launches, and what's the success rate of each one?

	FlightNumber	Date	BoosterVersion	PayloadMass	Orbit	LaunchSite	Outcome	Flights	GridFins	Reused	Legs	LandingPad	Block	ReusedCount	Serial	Longitude	Latitude	Class
0	1	2010-06-04	Falcon 9	6104.959412	LEO	CCAFS SLC 40	None None	1	False	False	False	NaN	1.0	0	B0003	-80.577366	28.561857	0
1	2	2012-05-22	Falcon 9	525.000000	LEO	CCAFS SLC 40	None None	1	False	False	False	NaN	1.0	0	B0005	-80.577366	28.561857	0
2	3	2013-03-01	Falcon 9	677.000000	ISS	CCAFS SLC 40	None None	1	False	False	False	NaN	1.0	0	B0007	-80.577366	28.561857	0
3	4	2013-09-29	Falcon 9	500.000000	PO	VAFB SLC 4E	False Ocean	1	False	False	False	NaN	1.0	0	B1003	-120.610829	34.632093	0
4	5	2013-12-03	Falcon 9	3170.000000	GTO	CCAFS SLC 40	None None	1	False	False	False	NaN	1.0	0	B1004	-80.577366	28.561857	0

What orbits were used and what's their success rate?

	FlightNumber	Date	BoosterVersion	PayloadMass	Orbit	LaunchSite	Outcome	Flights	GridFins	Reused	Legs	LandingPad	Block	ReusedCount	Serial	Longitude	Latitude	Class
0	1	2010-06-04	Falcon 9	6104.959412	LEO	CCAFS SLC 40	None None	1	False	False	False	NaN	1.0	0	B0003	-80.577366	28.561857	0
1	2	2012-05-22	Falcon 9	525.000000	LEO	CCAFS SLC 40	None None	1	False	False	False	NaN	1.0	0	B0005	-80.577366	28.561857	0
2	3	2013-03-01	Falcon 9	677.000000	ISS	CCAFS SLC 40	None None	1	False	False	False	NaN	1.0	0	B0007	-80.577366	28.561857	0
3	4	2013-09-29	Falcon 9	500.000000	PO	VAFB SLC 4E	False Ocean	1	False	False	False	NaN	1.0	0	B1003	-120.610829	34.632093	0
4	5	2013-12-03	Falcon 9	3170.000000	GTO	CCAFS SLC 40	None None	1	False	False	False	NaN	1.0	0	B1004	-80.577366	28.561857	0

What orbits were used and what's their success rate?

	FlightNumber	Date	BoosterVersion	PayloadMass	Orbit	LaunchSite	Outcome	Flights	GridFins	Reused	Legs	LandingPad	Block	ReusedCount	Serial	Longitude	Latitude	Class
0	1	2010-06-04	Falcon 9	6104.959412	LEO	CCAFS SLC 40	None None	1	False	False	False	NaN	1.0	0	B0003	-80.577366	28.561857	0
1	2	2012-05-22	Falcon 9	525.000000	LEO	CCAFS SLC 40	None None	1	False	False	False	NaN	1.0	0	B0005	-80.577366	28.561857	0
2	3	2013-03-01	Falcon 9	677.000000	ISS	CCAFS SLC 40	None None	1	False	False	False	NaN	1.0	0	B0007	-80.577366	28.561857	0
3	4	2013-09-29	Falcon 9	500.000000	PO	VAFB SLC 4E	False Ocean	1	False	False	False	NaN	1.0	0	B1003	-120.610829	34.632093	0
4	5	2013-12-03	Falcon 9	3170.000000	GTO	CCAFS SLC 40	None None	1	False	False	False	NaN	1.0	0	B1004	-80.577366	28.561857	0

Looking for some answers with SQL

*First we'll use some **SQL** queries to start answering this questions. For this we'll install the necessary libraries on the **Jupyter Notebook** and establish a connection to the database file using **SQLite**.*

```
import csv, sqlite3
con = sqlite3.connect("my_data1.db")
cur = con.cursor()

import pandas as pd
df=pd.read_csv("https://cf-courses-data.s3.us.cloud...)
df.to_sql("SPACEXTBL",con,if_exists='replace',index=False,method="multi")
```

When was the first successful landing, and what is the success rate over time?

To see the complete code please click on this link.

```
%%sql
SELECT date AS first_successful_landing_date, Landing_Outcome
FROM SPACEXTABLE
WHERE date = (SELECT MIN(date)
FROM SPACEXTABLE WHERE Landing_Outcome LIKE 'Success%'
OR Landing_Outcome LIKE 'Controlled%');
```

When was the first successful landing, and what is the success rate over time?

First_Successful_Landing	Landing_Outcome
2014-04-18	Controlled (ocean)

How does the payload capacity increase over time?

```
%%sql
SELECT strftime('%Y', Date) AS year,
SUM(PAYLOAD_MASS__KG_) AS total_payload_mass
FROM SPACEXTABLE
GROUP BY year;
```

How does the payload capacity increase over time?

year	total_payload_mass
2010	0
2012	1025
2013	4347
2014	18116
2015	17715
2016	27213
2017	95978
2018	96957
2019	80761
2020	277855

What's the location with the most launches, and what's the success rate of each one?

```
%%sql
SELECT Launch_Site,
    COUNT(*) AS Total_Launches,
    ROUND(
        100.0 * SUM(CASE WHEN Landing_Outcome LIKE 'Success%' OR
Landing_Outcome LIKE 'Controlled%' THEN 1 ELSE 0 END) / COUNT(*), 2
    ) AS Success_Rate
FROM SPACEXTABLE
GROUP BY Launch_Site
ORDER BY Total_Launches DESC;
```

To see the complete notebook please click on this link.

What's the location with the most launches, and what's the success rate of each one?

Launch_Site	Total_Launches	Success_Rate
CCAFS SLC-40	60	58.33
KSC LC-39A	25	80.0
VAFB SLC-4E	16	68.75

What orbits were used and what's their total payload and success rate?

```
%%sql
SELECT Orbit,
    SUM(PAYLOAD_MASS__KG_) AS Total_Payload_Mass,
    ROUND(
        100.0 * SUM(CASE WHEN Landing_Outcome LIKE 'Success%' OR
Landing_Outcome LIKE 'Controlled%' THEN 1 ELSE 0 END) / COUNT(*), 2
    ) AS Success_Rate
FROM SPACEXTABLE
GROUP BY Orbit
ORDER BY Total_Payload_Mass DESC;
```

What orbits were used and what's their total payload and success rate?

Orbit	Total_Payload_Mass	Success_Rate
LEO	276587	84.0
GTO	150038	53.33
LEO (ISS)	85823	57.69
Polar LEO	64560	62.5
SSO	16955	83.33
MEO	13022	66.67
Sub-orbital	12050	0.0
HEO	932	100.0

Visualizing the data with Seaborn

*Now we'll keep answering those questions plotting some **charts**. For that we'll need to import the required libraries like **Pandas, Numpy, Matplotlib** and **Seaborn***

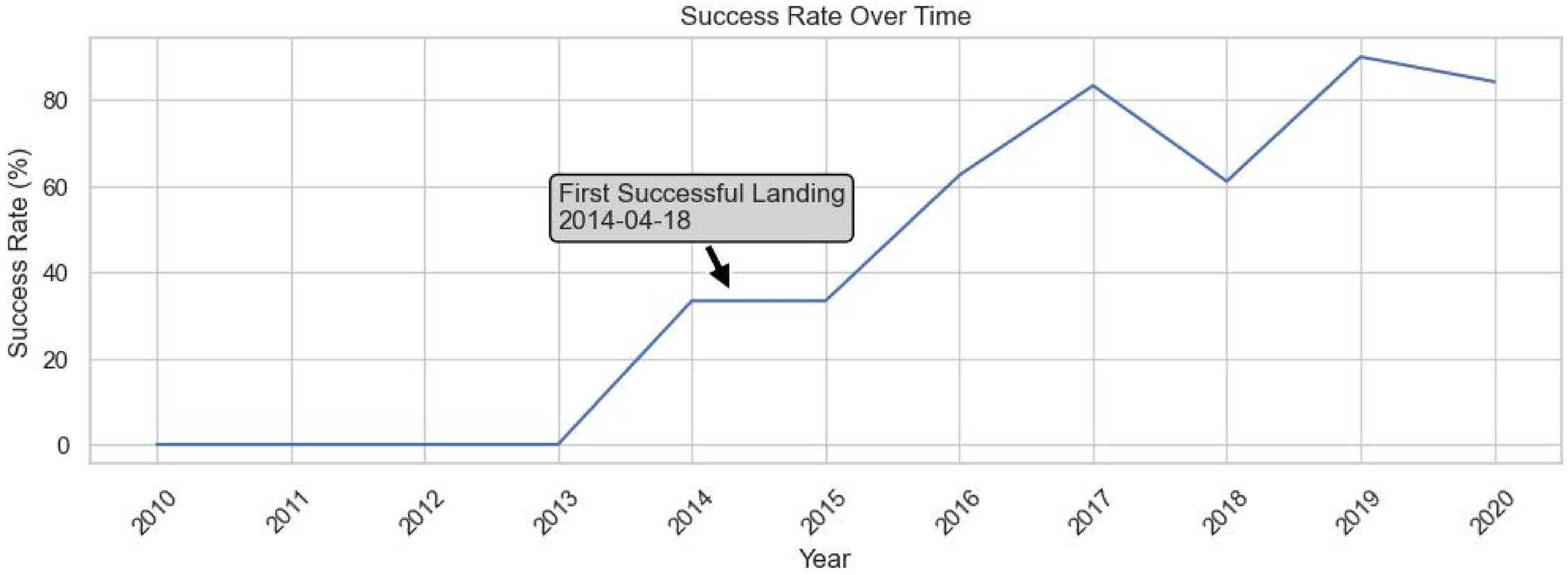
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from js import fetch
import io

resp = await fetch("https://cf-courses-data.s3.us.cloud...)
dataset_part_2_csv = io.BytesIO((await resp.arrayBuffer()).to_py())
df=pd.read_csv(dataset_part_2_csv)
```

When was the first successful landing, and what is the success rate over time?

To see the complete code please click on this link.

```
df['Year'] = pd.to_datetime(df['Date']).dt.year
result = df.groupby('Year')['Class'].mean().reset_index()
result['SuccessRate'] = result['Class'] * 100

sns.set(style="whitegrid")
sns.lineplot(x="Year", y="SuccessRate", data=result)
plt.xlabel("Year")
plt.ylabel("Success Rate (%)")
plt.title("Success Rate Over Time")
plt.show()
```

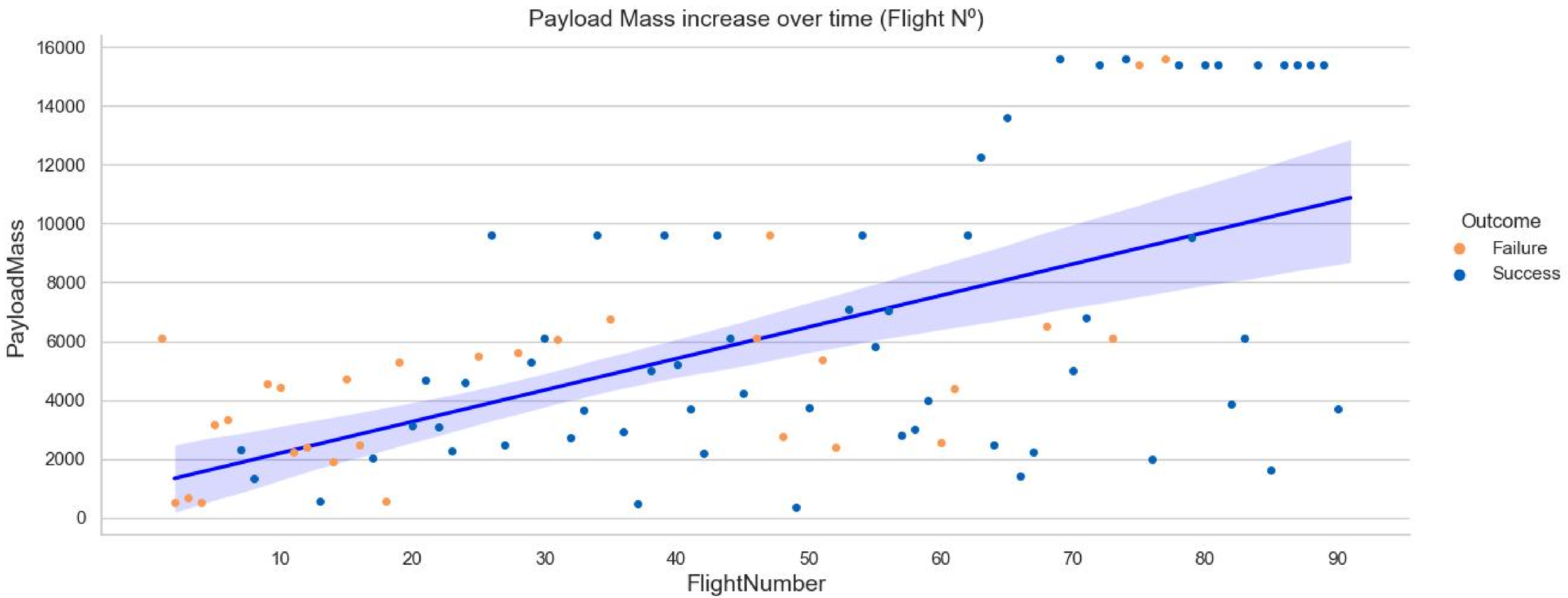
When was the first successful landing, and what is the success rate over time?



How does the payload capacity increase over time?

```
g = sns.catplot(y="PayloadMass", x="FlightNumber", hue="Class",
data=df_mapped)

g.set_axis_labels("Flight Number", "Payload Mass (kg)", fontsize=14)

plt.title("Payload Mass increase over time (Flight Nº)", fontsize=14)

sns.regplot(y="PayloadMass", x="FlightNumber", data=df_mapped,
scatter=False, color="blue")

plt.show()
```

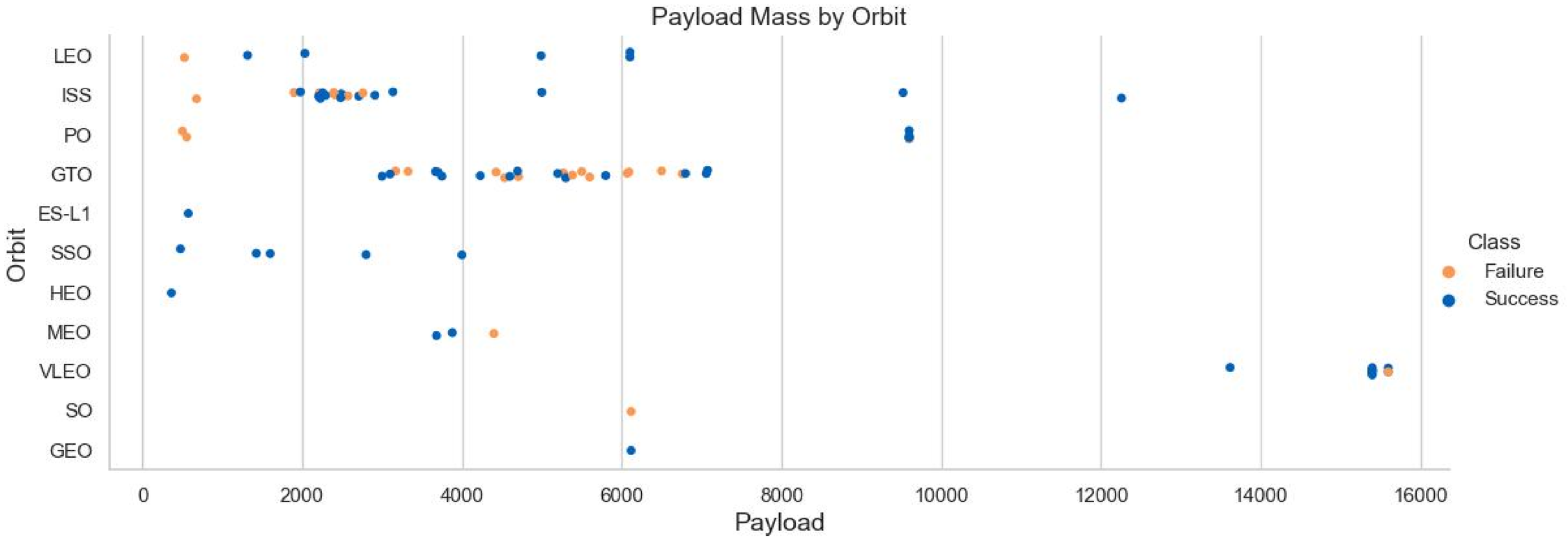
How does the payload capacity increase over time?



What is the relation between payload and orbit choice?

```
g = sns.catplot(y="Orbit", x="PayloadMass", hue="Class",
data=df_mapped, aspect = 3, palette=custom_palette)

g.figure.set_size_inches(12, 4)

g.set_axis_labels("Payload", "Orbit", fontsize=14)

plt.title("Payload Mass by Orbit", fontsize=14)

plt.show()
```

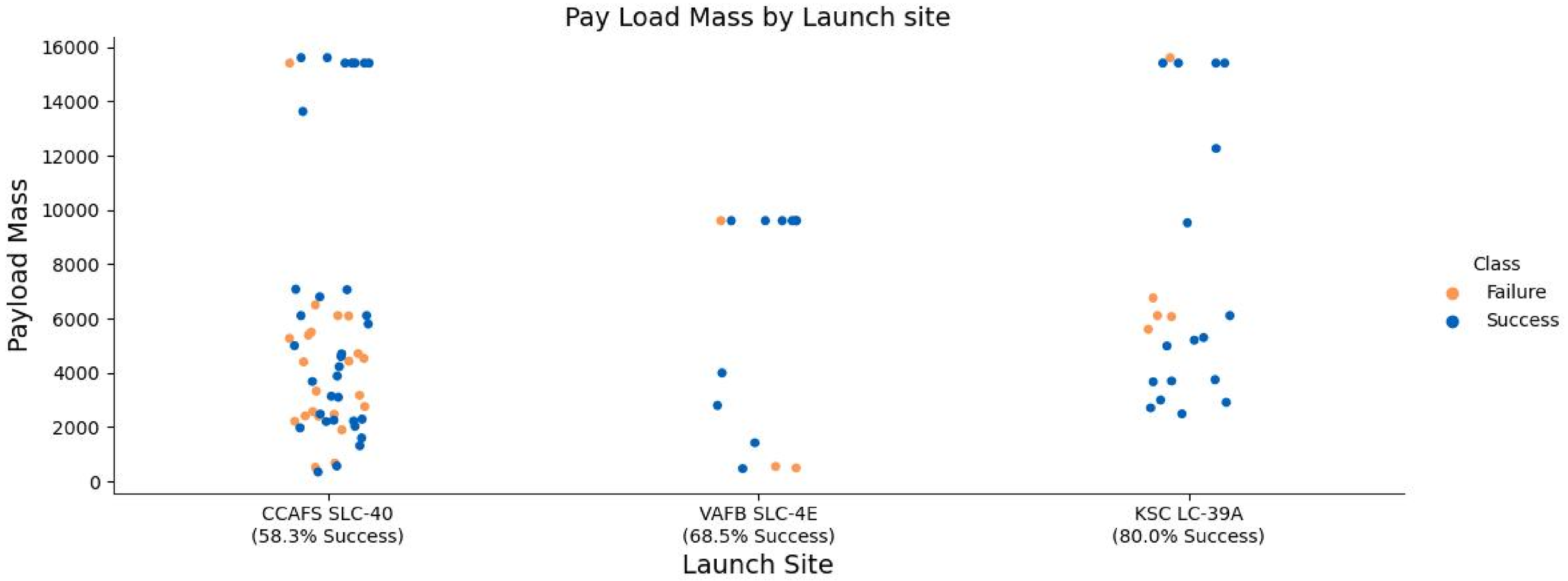
What is the relation between payload and orbit choice?



What's the location with the most launches, and what's the success rate of each one? Let's also see the relation with the Payload Mass ...

```
g = sns.catplot(y="PayloadMass", x="LaunchSite", hue="Class",
data=df_mapped, aspect=2, palette=custom_palette)

new_labels = ['CCAFS SLC-40\n(58.3% Success)', 'VAFB SLC-4E\n(68.5%
Success)', 'KSC LC-39A\n(80.0% Success)']
g.set_xticklabels(new_labels)

g.set_axis_labels("Payload Mass", "Launch Site", fontsize=14)
plt.title("Payload Mass by Launch Site", fontsize=14)
plt.show()
```

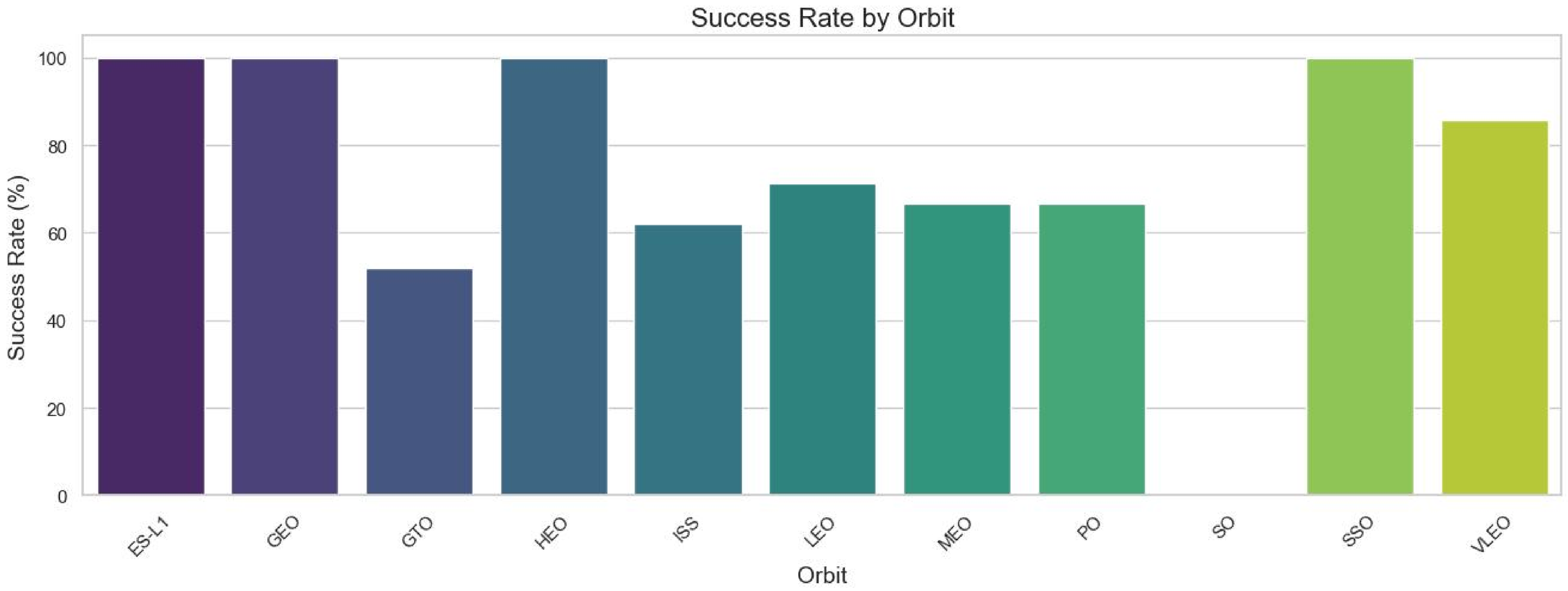
What's the location with the most launches, and what's the success rate of each one? Let's also see the relation with the Payload Mass …



What orbits were used and what's their success rate?

```
result = df.groupby('Orbit')['Class'].mean() * 100
new_df = result.reset_index()
sns.barplot(x="Orbit", y="Class", data=new_df, palette="viridis")

plt.xlabel("Orbit", fontsize=14)
plt.ylabel("Success Rate (%)", fontsize=14)
plt.title("Success Rate by Orbit", fontsize=16)
plt.xticks(rotation=45)
plt.show()
```

What orbits were used and what's their success rate?



Methodology

Data Collection

Data Wrangling

Exploratory Data Analysis

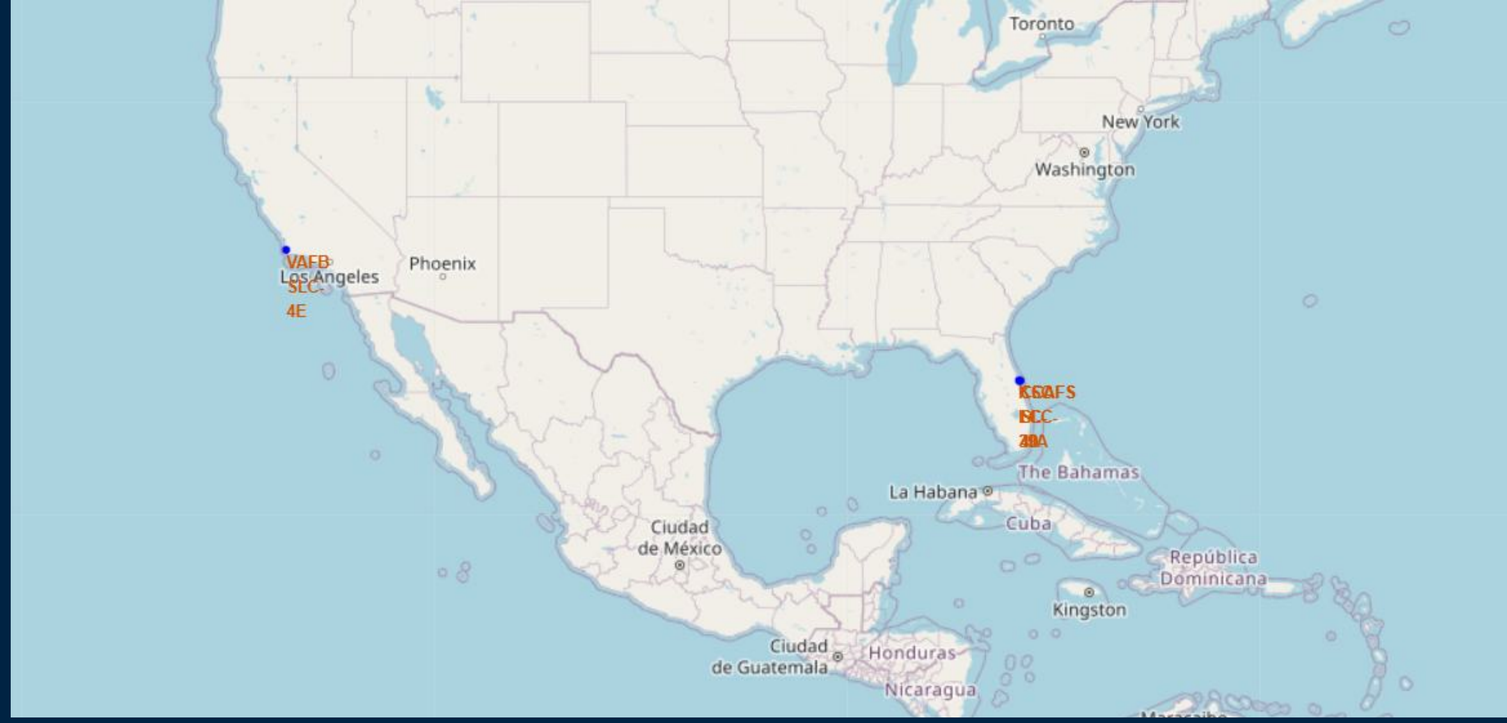
Interactive Visualization

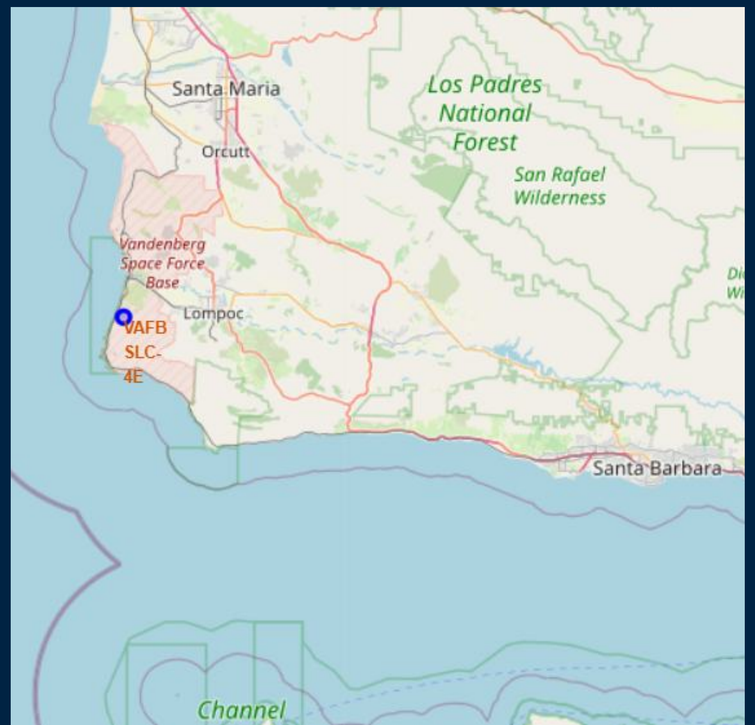
Predictive Modeling

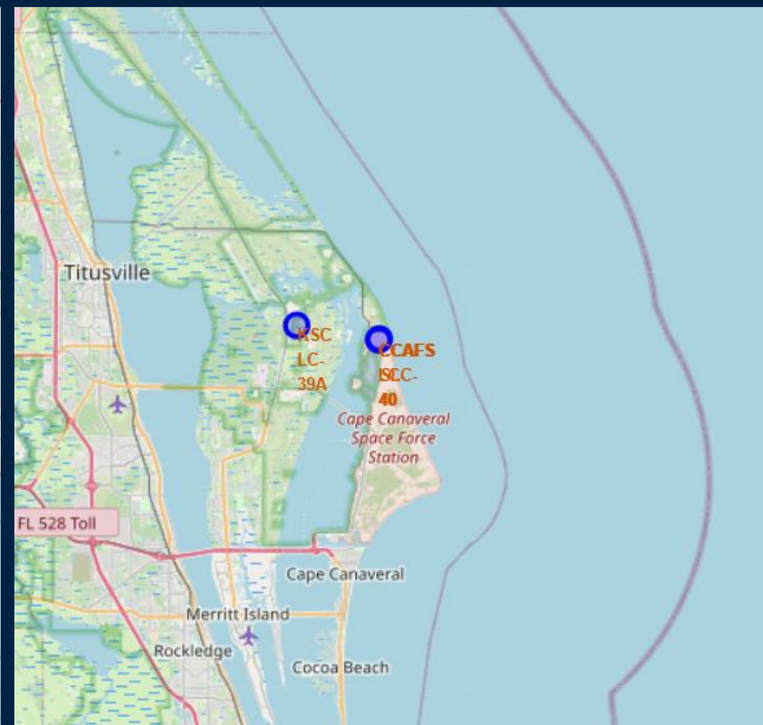
Launch Sites Proximities Analysis

We used a Folium map to show the different launch locations and their distribution across the United States.
This gives us a better dimension of the distance they have between each other.

Click to see Notebook.

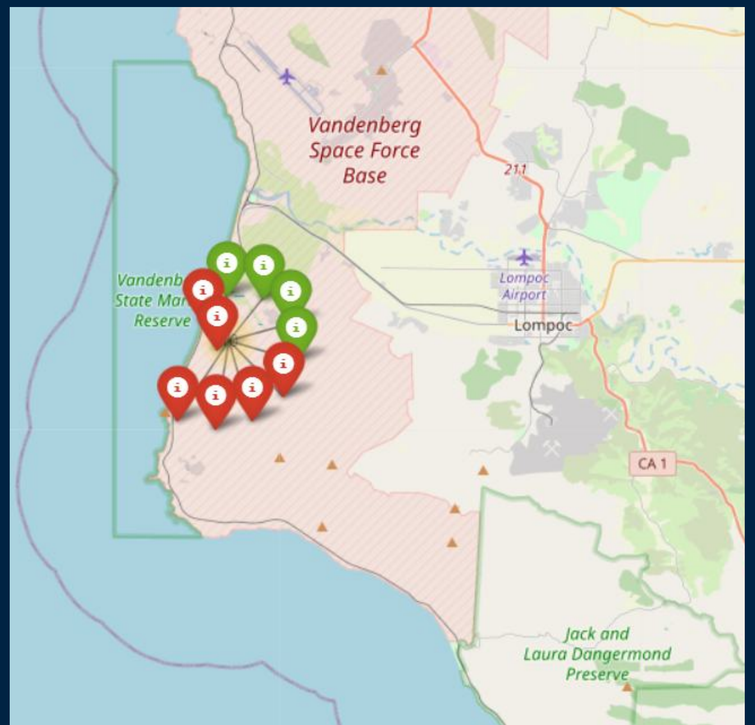


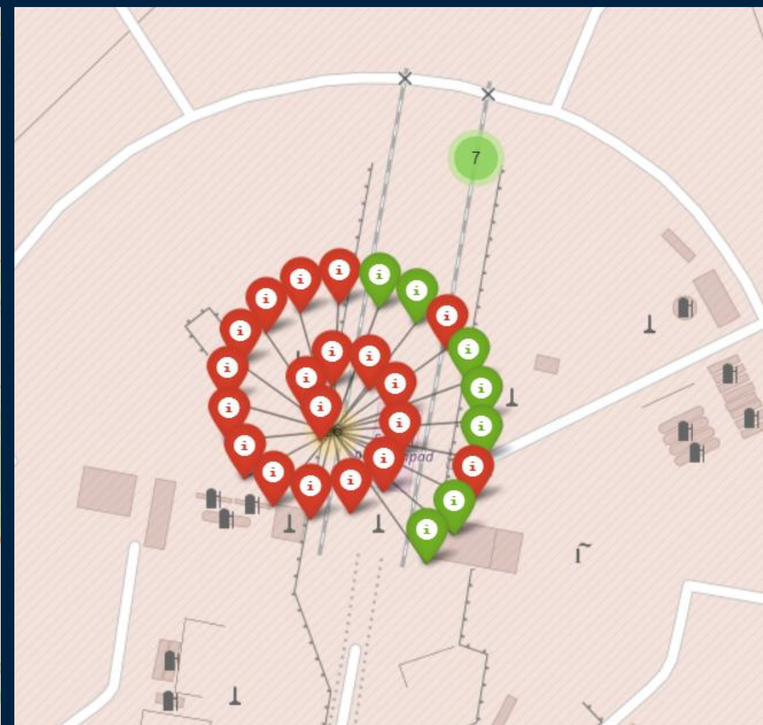




With Marker clusters we are able to show how many launches were made in each site and by clicking on them, we get the landing result for each site.

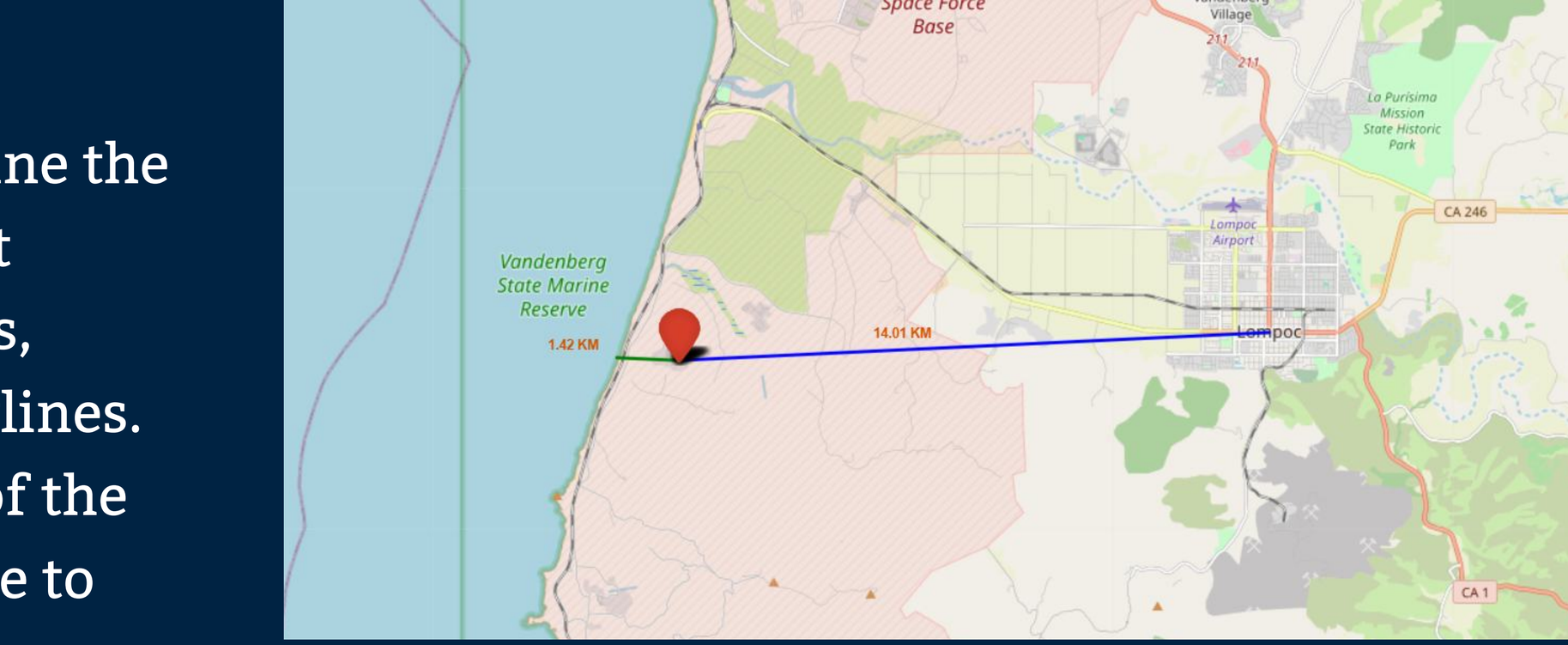
Click to see Notebook.

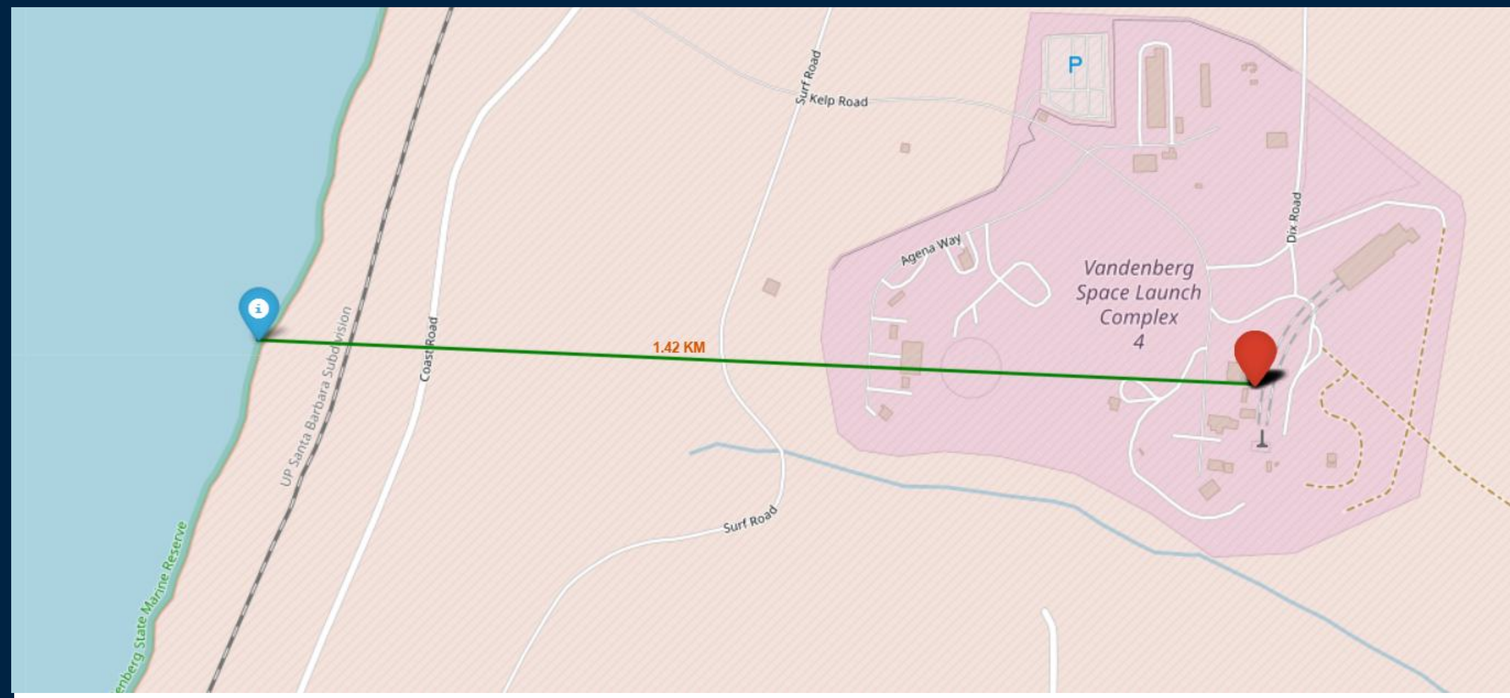




Also we can determine the distance to different landmarks like cities, airports or the coastlines. We can see that all of the launch sites are close to the ocean for various reasons, including safety, environmental and logistic considerations.

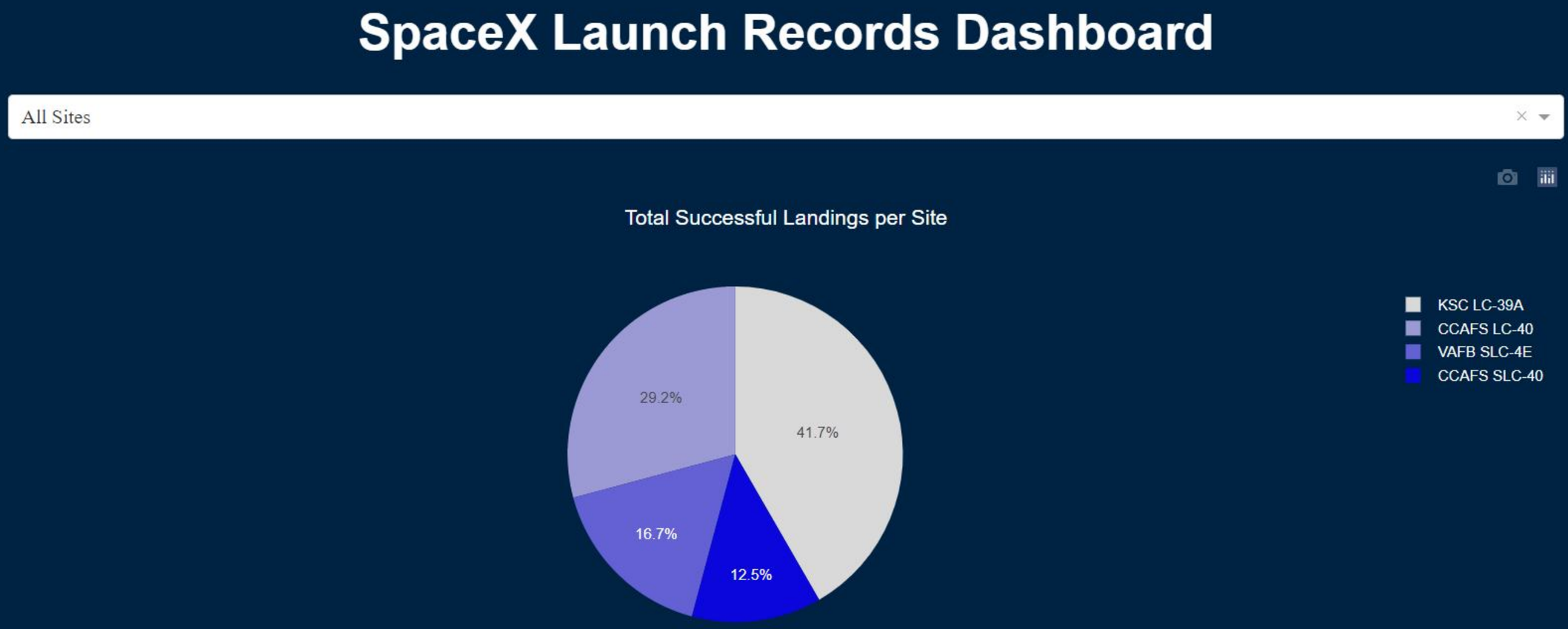
Click to see Notebook.



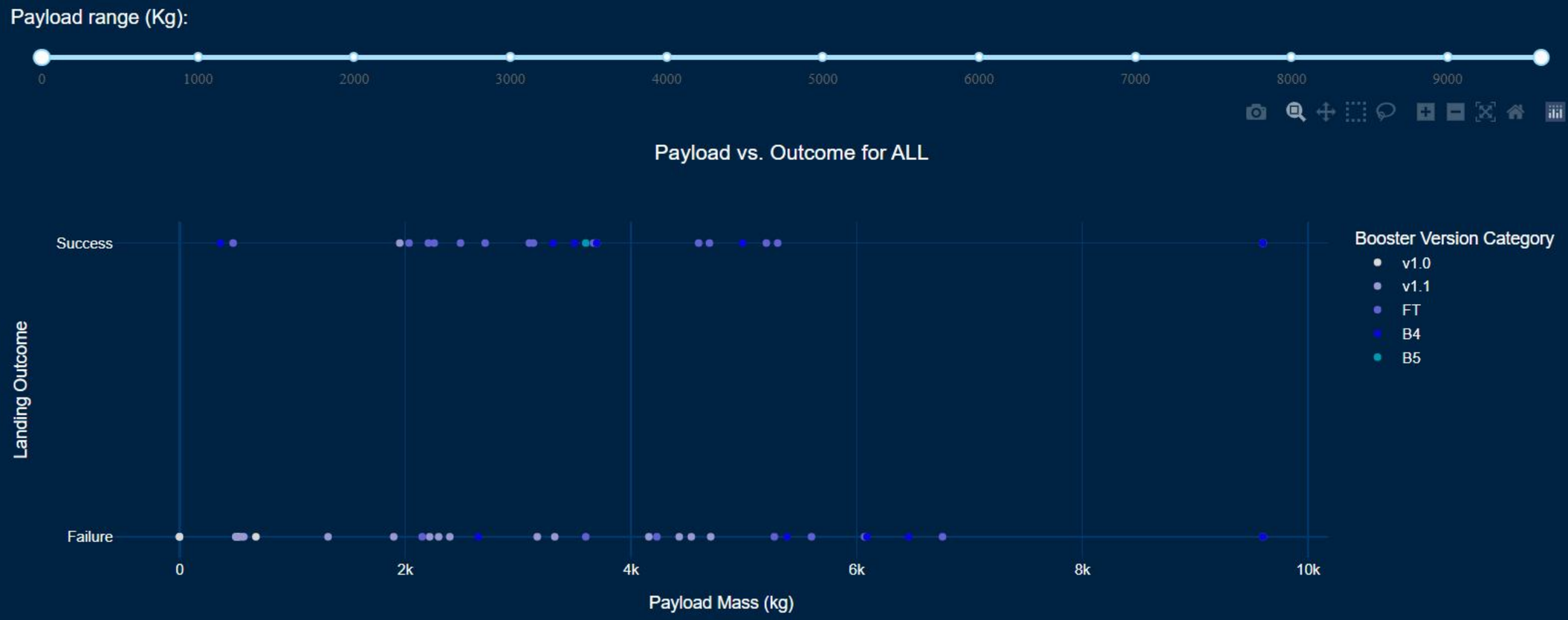


Dashboard with Plotly Dash

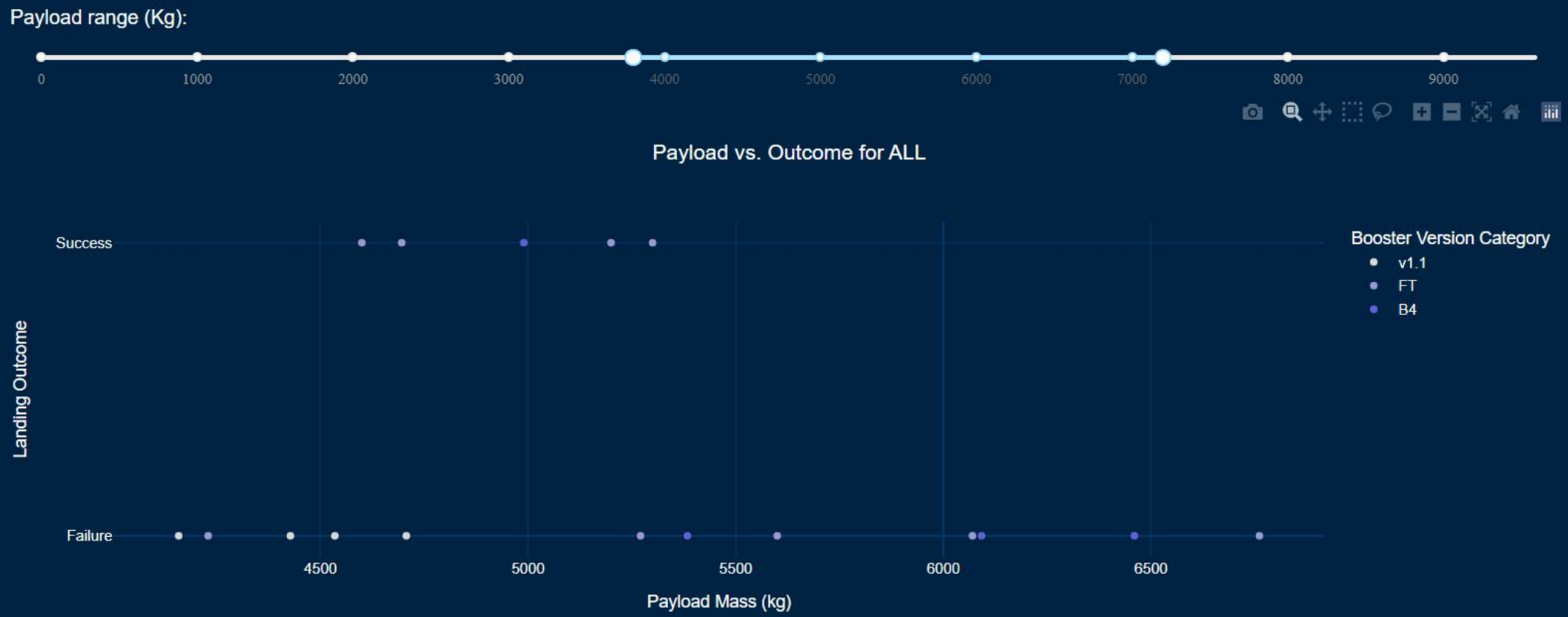
We developed a Plotly Dash app that allows us to interact with the data.
With an interactive pie chart we can choose to see what launch site was the most successful...



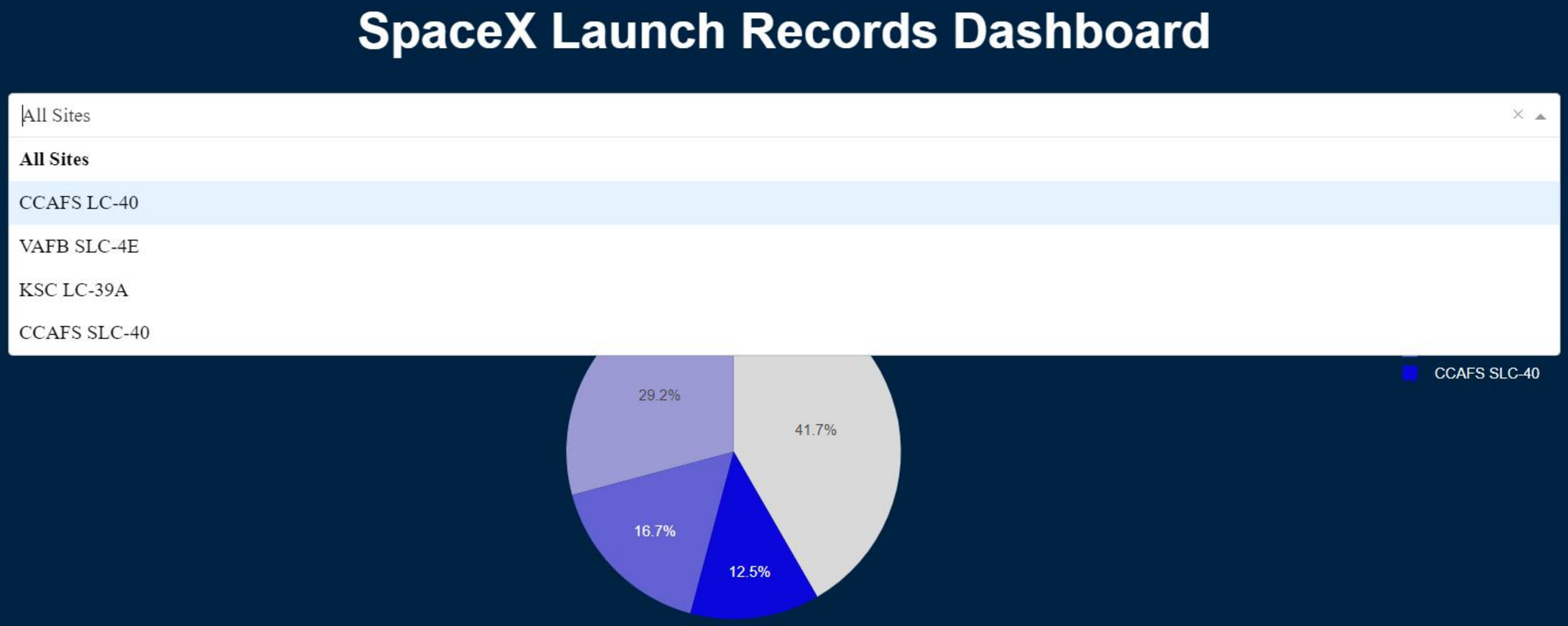
A scatter plot let us see what is the relation between Payload and Outcome, also showing what Booster version was used.



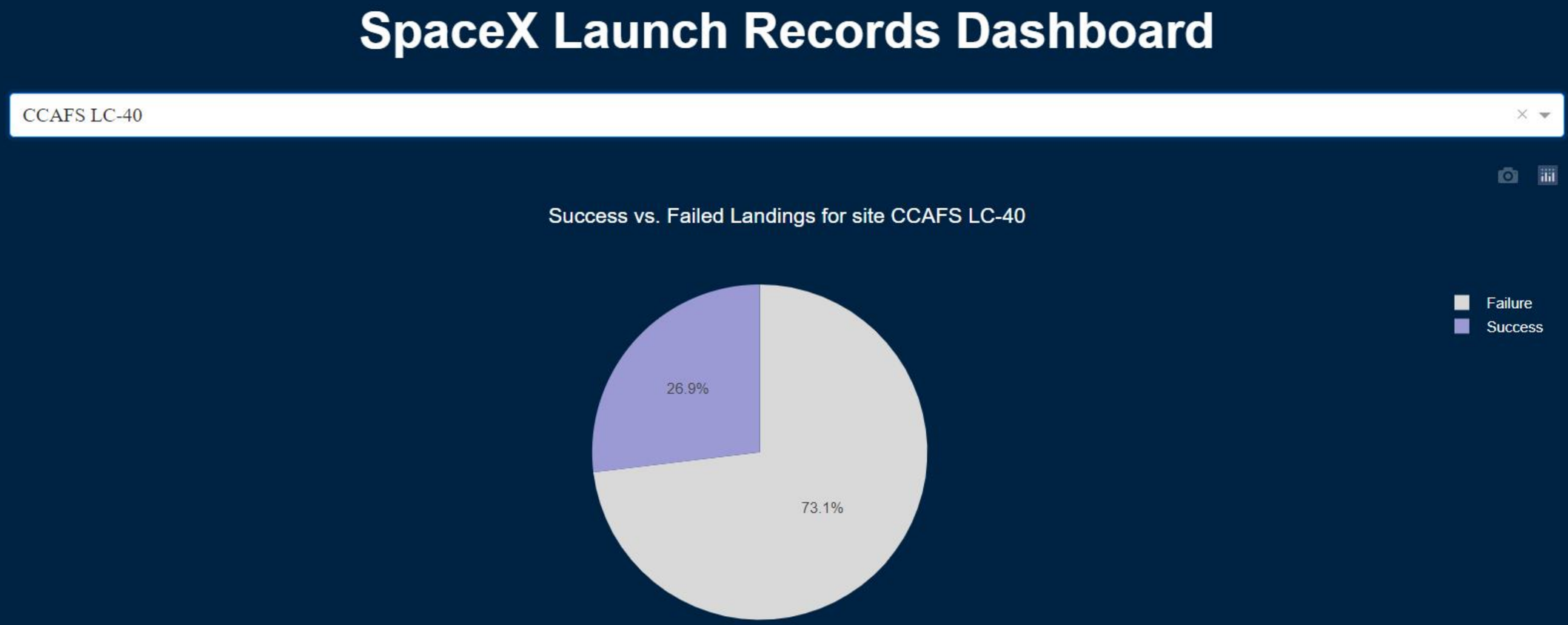
The addition of a slider allows the possibility to explore a certain range of data.



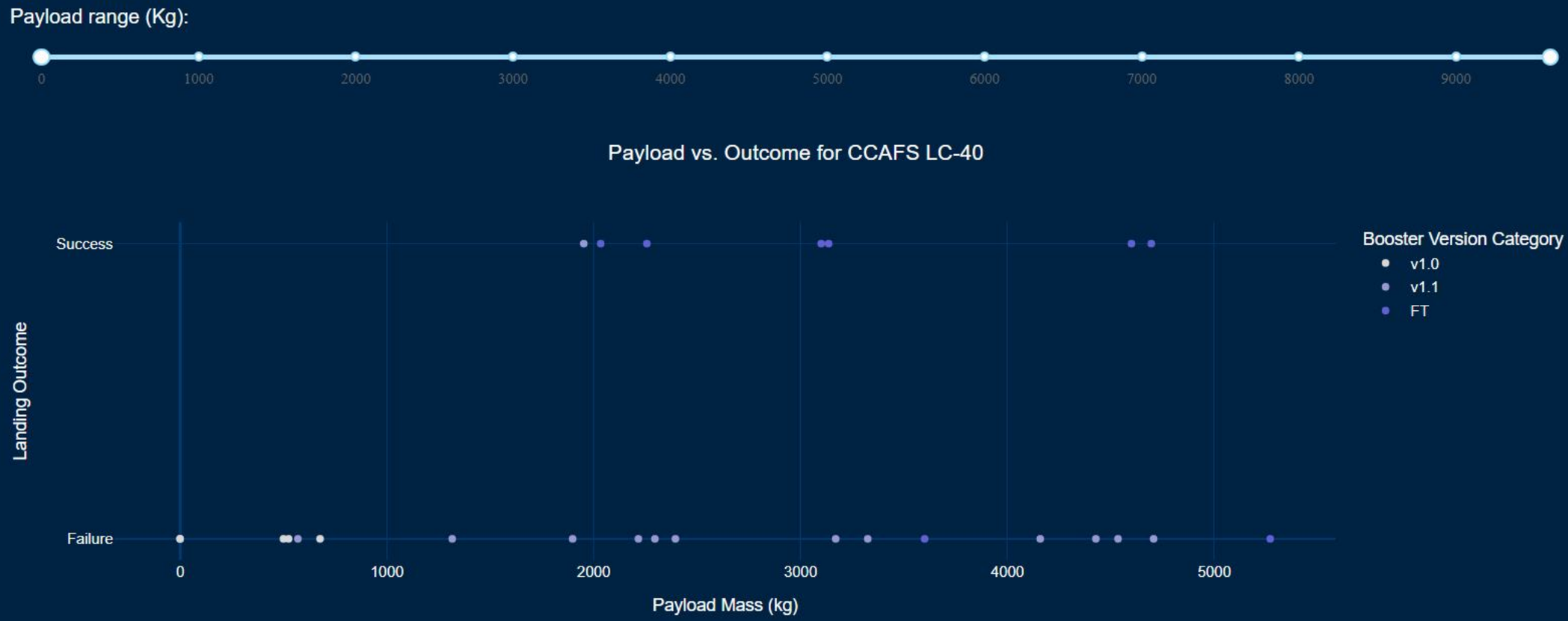
If we come back to the dropdown menu, we can choose a specific launch site to see the success and failure percentages.



If we come back to the dropdown menu, we can choose a specific launch site to see the success and failure percentages.



The scatter plot will also be affected by the dropdown menu, showing data for the chosen launch site.



Methodology

Data Collection

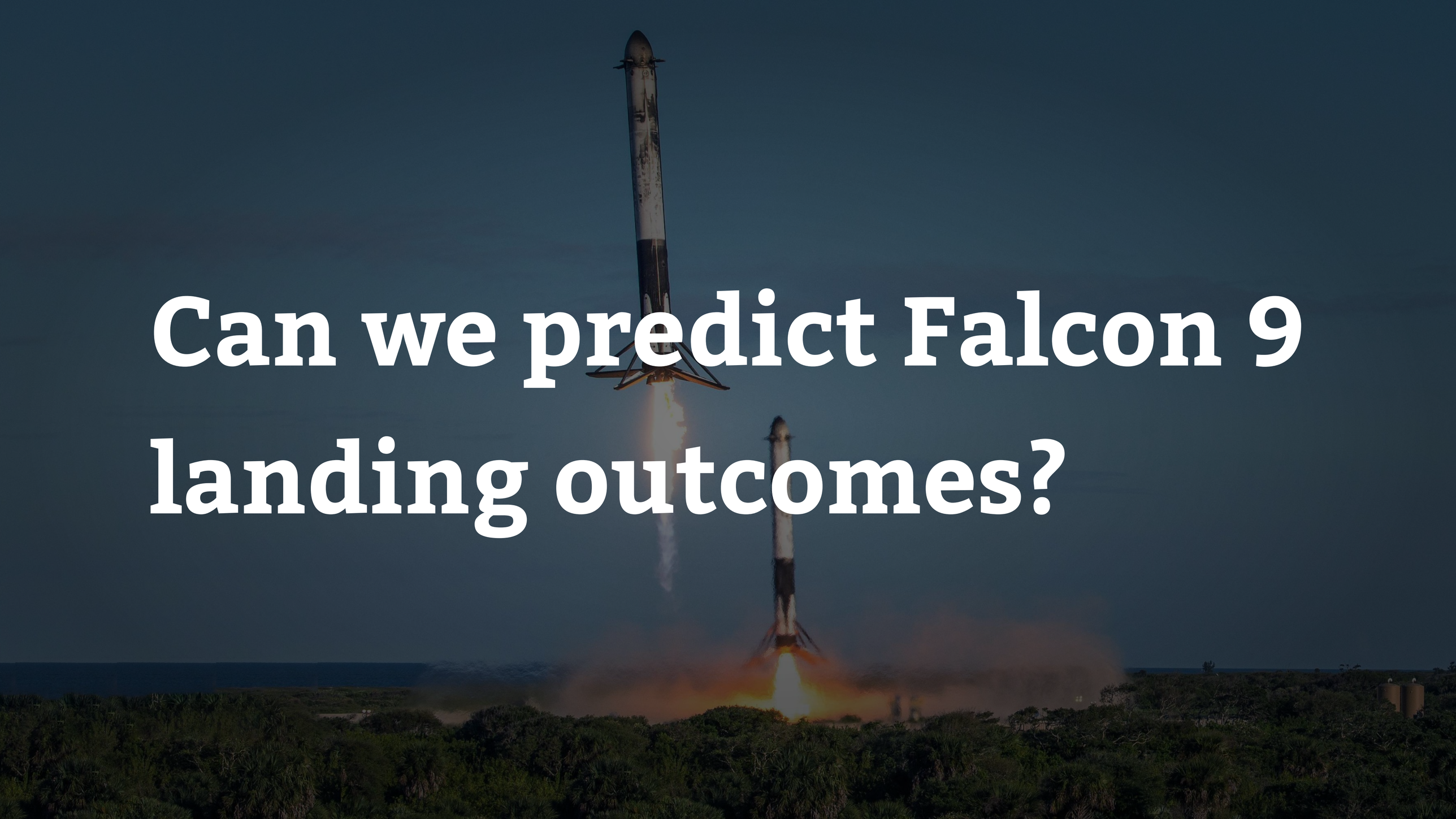
Data Wrangling

Exploratory Data Analysis

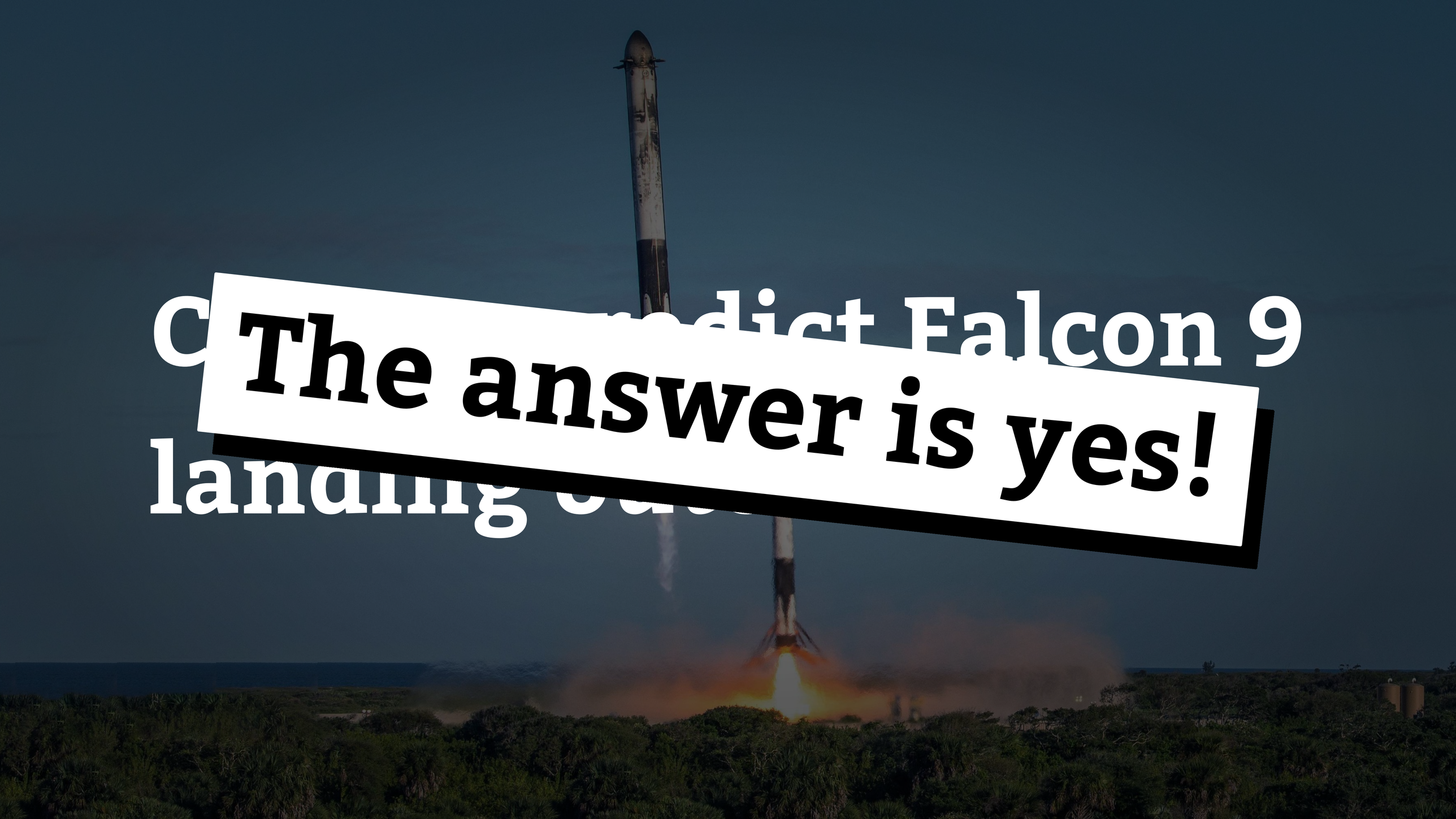
Interactive Visualization

Predictive Modeling

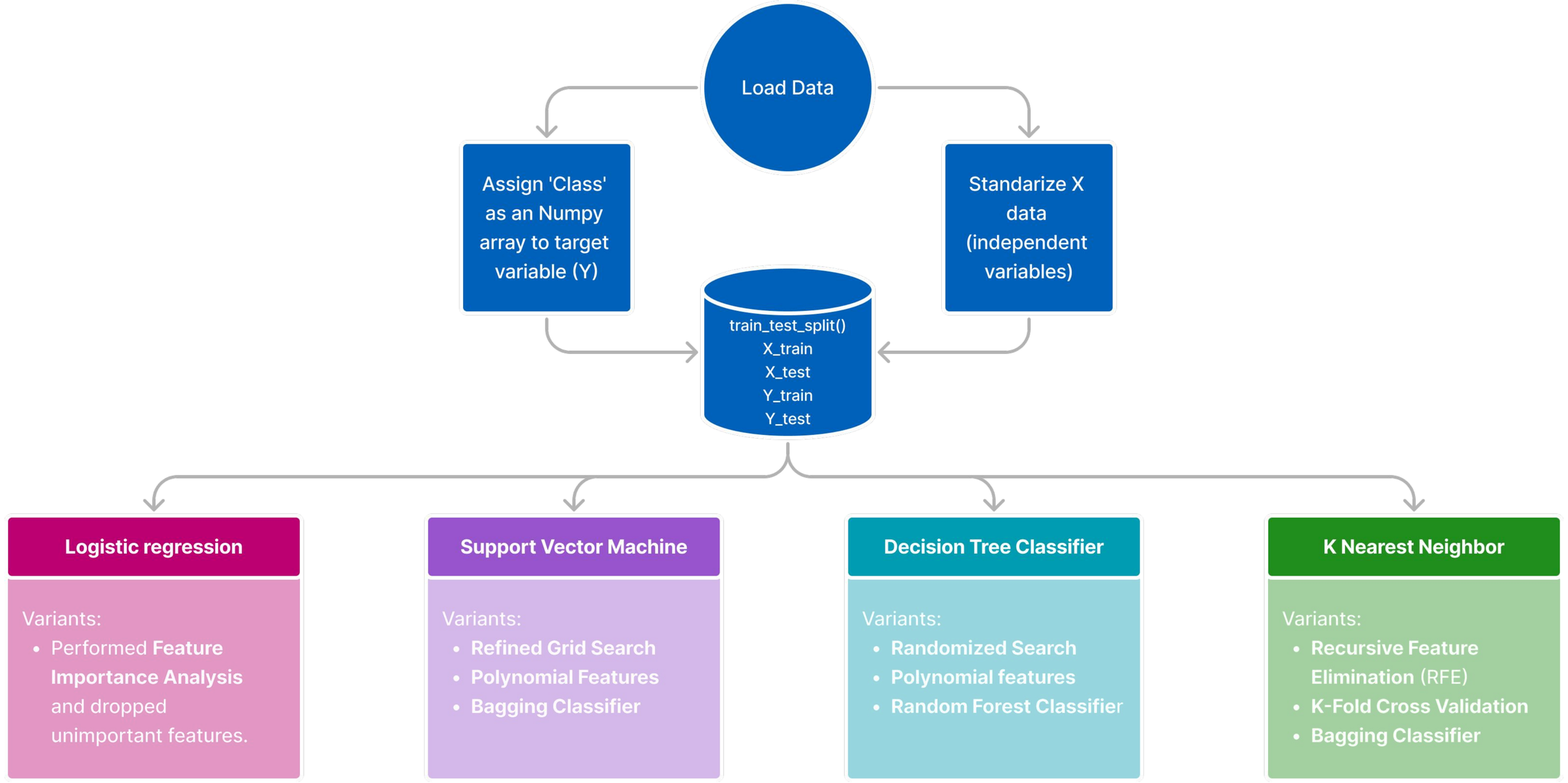
In this final stage, we go back to the question we asked at the beginning of this project and try to answer it...



Can we predict Falcon 9 landing outcomes?


C dict Falcon 9
landing
The answer is yes!

We tried different predictive models…


Load Data
Assign 'Class' as an Numpy array to target variable (Y)
Standarize X data (independent variables)
train_test_split()
X_train
X_test
Y_train
Y_test
Logistic regression
Variants:
Performed Feature Importance Analysis and dropped unimportant features.
Support Vector Machine
Variants:
Refined Grid Search
Polynomial Features
Bagging Classifier
Decision Tree Classifier
Variants:
Randomized Search
Polynomial features
Random Forest Classifier
K Nearest Neighbor
Variants:
Recursive Feature Elimination (RFE)
K-Fold Cross Validation
Bagging Classifier

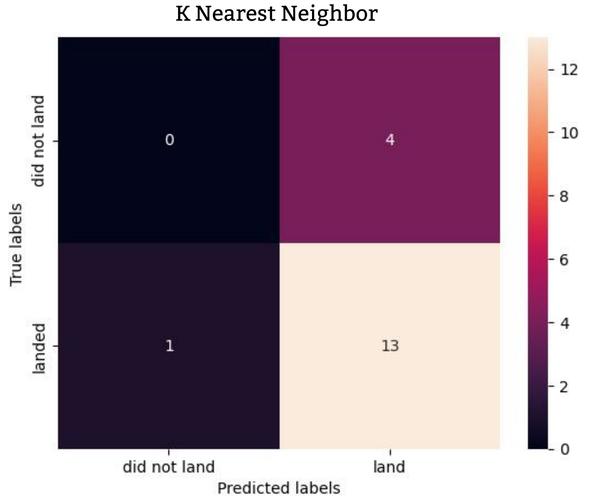
Started with 4 different models, evaluated them, and tried to improve their performance through different methods.

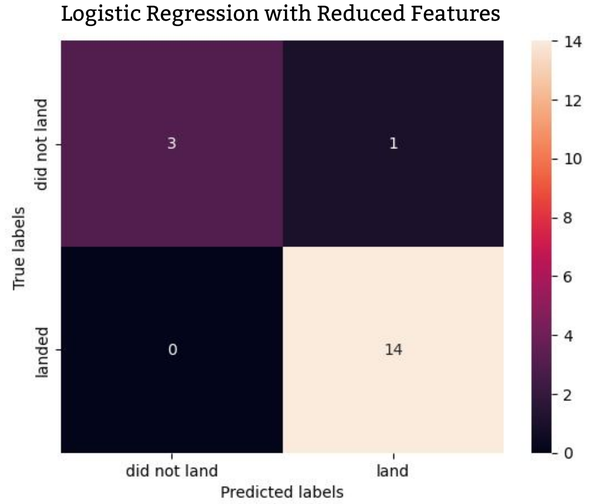
We managed to improve the **Test Accuracy** in some of the models. In other cases the Train Accuracy looked very promising, but when we tried this models on the Test data, it was clear that this was due to overfitting.

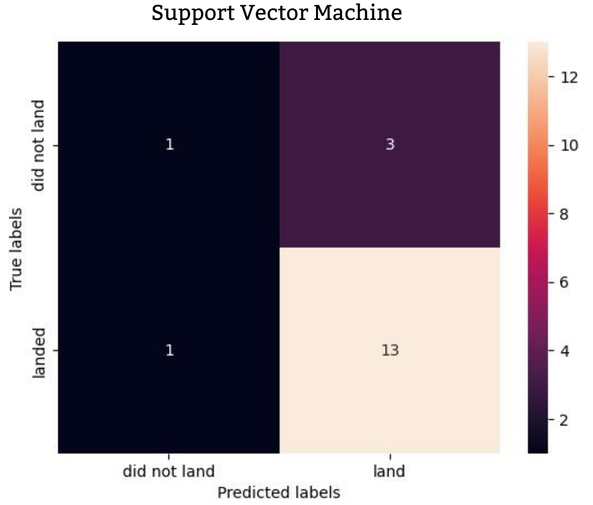
Click to see Notebook.

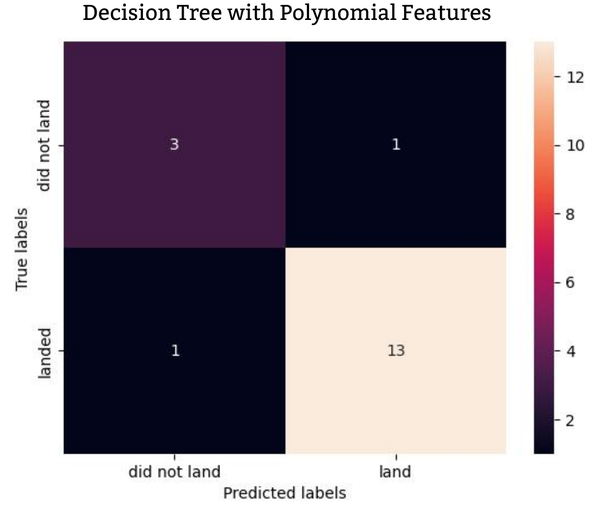
The **Confusion Matrix** helps us identify specific areas where the models are performing well or poorly.
Here we plotted the variants of each model with the best Test Accuracy.

Click to see Notebook.





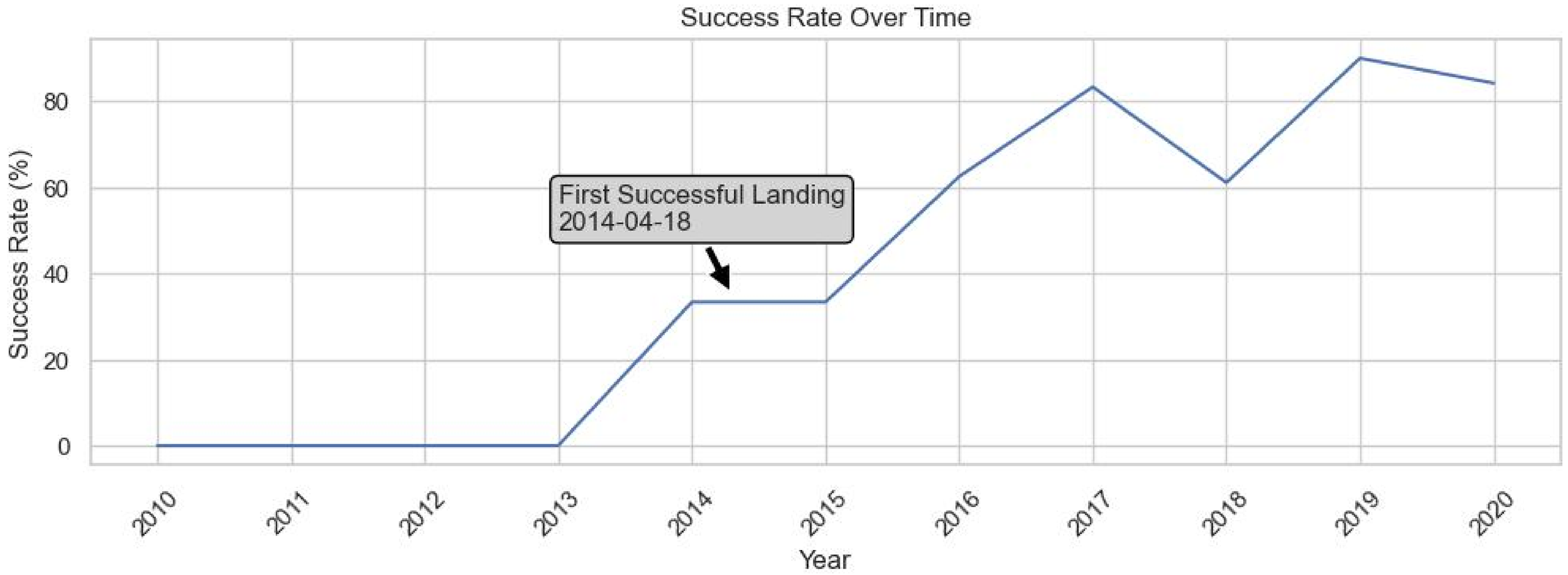




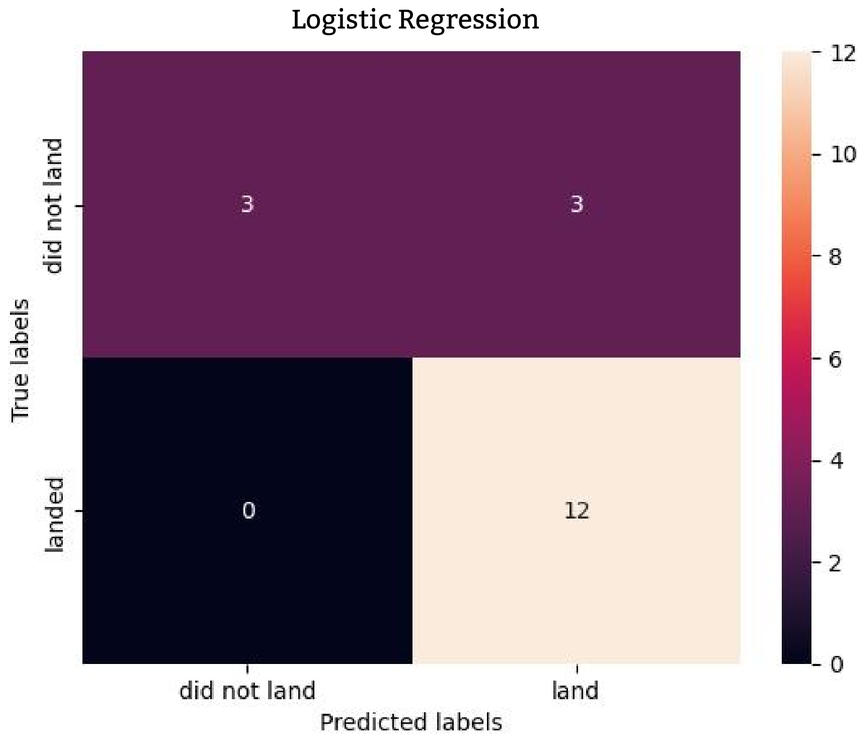
Key Results and Insights

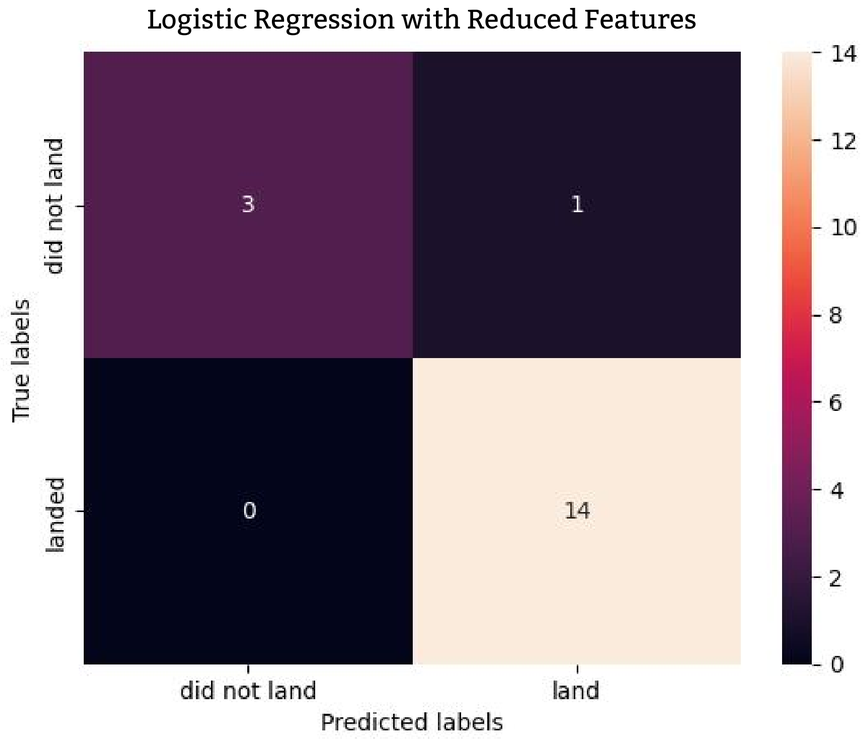
- Historical Success Rate
- Trend of Improvement
- Influential Factors
- Predictive Model Accuracy

*Thanks to the EDA process we were able to know that the **landing success rate** started to increase relatively recently, accomplishing the first successful landing on 2014.*

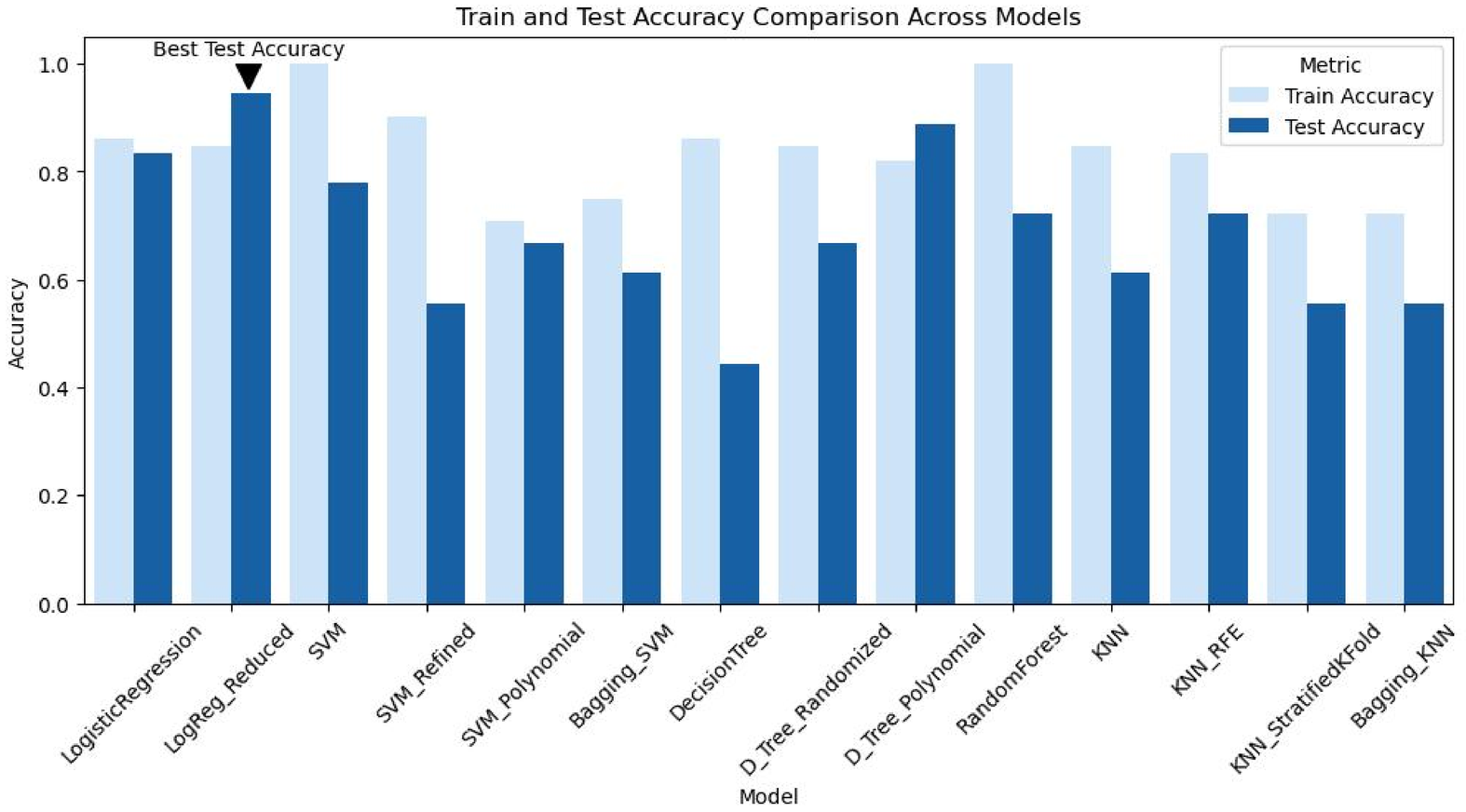


*On the original Logistic Regression model I realized that the major problem was **false positives.** By dropping the features with a coefficient of 0.0 I managed to improve its accuracy.*

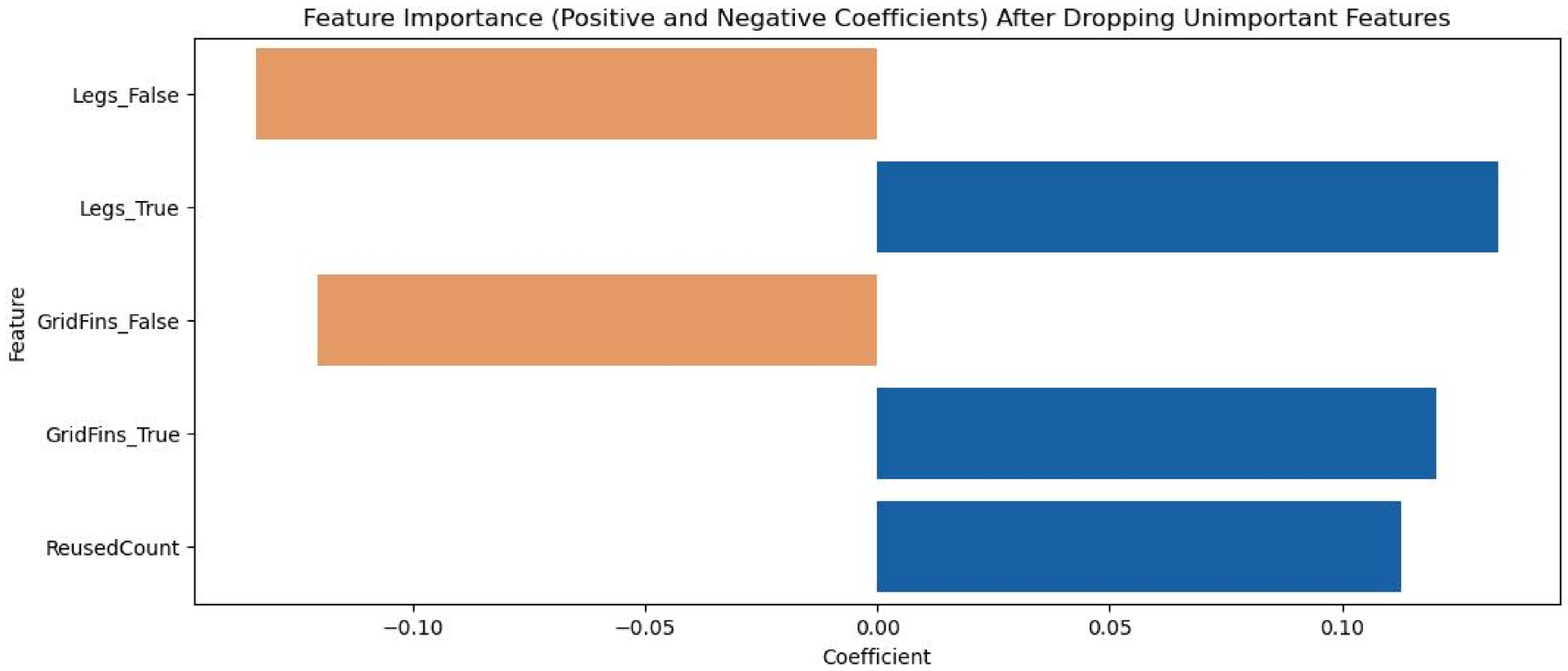




*The best performance on the Test data was achieved by a **Logistic Regression (0.94% Accuracy)** model after performing a **Feature Importance Analysis** and dropping unimportant features.*



*We also discovered that **the most important features** for predicting the target variable were the use of **legs**, **grid fins** and also the number of times the rocket was **reused**.*




Thank you!