Data Science Capstone

BY JULIA R.

COURSERA

IBM

Table of contents

Executive Summary

In this capstone project, we aim to predict whether the Falcon 9 first stage will successfully land. SpaceX advertises its Falcon 9 rocket launches at a cost of $62 million, while other providers charge upwards of $165 million per launch. A significant portion of the cost savings comes from SpaceX's ability to reuse the first stage. By accurately predicting whether this stage will land successfully, we can estimate the overall cost of a launch. This information is crucial for any alternative company looking to compete with SpaceX in the bidding process for rocket launches.

To conduct this analysis, we will utilize the SpaceX API, which provides a wealth of historical data on landings, technical specifications of launches, and other relevant information. We will explore this data using data exploration methods (data vizualisation tools such as folium, seaborn, plotly dash…) to identify the factors influencing landing success. Visualizations tools will be created to facilitate the understanding of results and to highlight relationships between various variables.

Furthermore, we will apply predictive analysis techniques to model and forecast landing success based on the collected data.(KNN, SVM, Decision Trees…) This project will thus provide a solid foundation for helping Space Y, a new competitor in the market, to develop effective pricing strategies and position itself against SpaceX.

Introduction

The commercial space industry is undergoing a rapid transformation, with private companies leading the way in making space travel more accessible. Among these, SpaceX has distinguished itself with its Falcon 9 rocket, which has successfully reduced launch costs through the reusability of its first stage. However, the success of this landing is not guaranteed, as it depends on various factors such as payload specifications, mission parameters, and environmental conditions.

This analysis aims to address the following research questions:

1. What historical patterns can be observed regarding the success rates of Falcon 9 first-stage landings?

2. What are the key factors influencing the likelihood of a successful landing?

3. How can predictive modeling techniques be used to forecast landing success based on historical data?

Utilizing data from the SpaceX API, this study will explore these questions through data analysis and predictive modeling, ultimately providing actionable insights for new market entrants like Space Y.

EDA and interactive visual analytics methodology

We will build a dashboard to analyze launch records interactively with Plotly Dash. You will then build an interactive map to analyze the launch site proximity with Folium.

Predictive analysis methodology

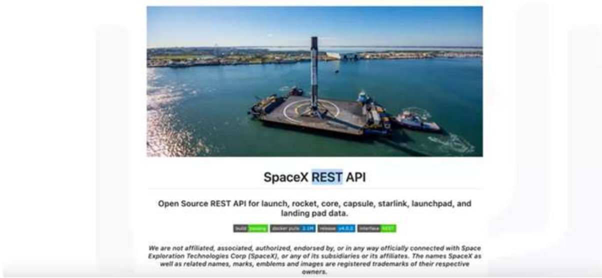
We will use machine learning to determine if the first stage of Falcon 9 will land successfully. You will split your data into training data and test data to find the best Hyperparameter for SVM, Classification Trees, and Logistic Regression. Then find the method that performs best using test data.

Data collection & wrangling

Data collection & wrangling

	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

SpaceX REST API



We created a class column that contains`1` if the booster successfully landed `0` otherwise based on the mission Outcome.

Some values were missing, we had to replace them by the mean.

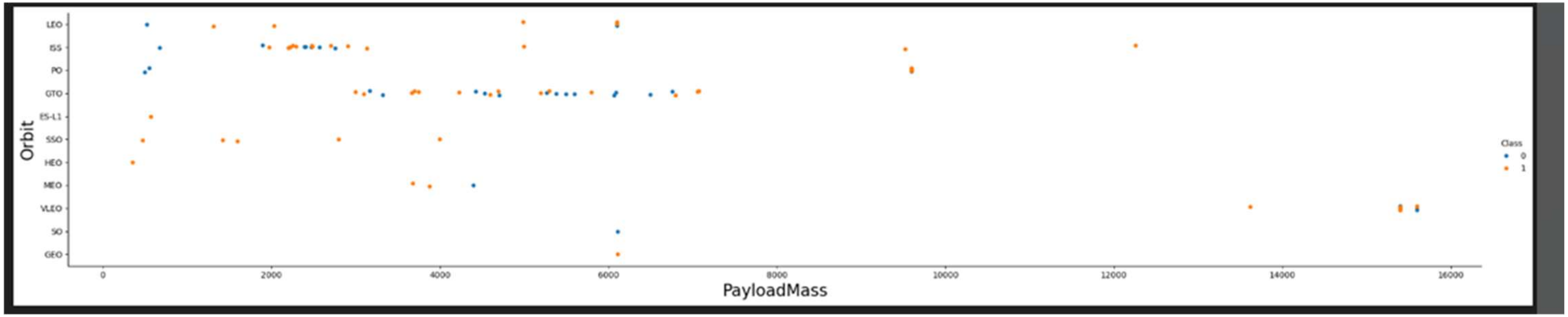
We had to convert categorical features to numerical in order to facilitate model building

Web scraping Falcon 9 Launch records

Web scraping with BeautifulSoup

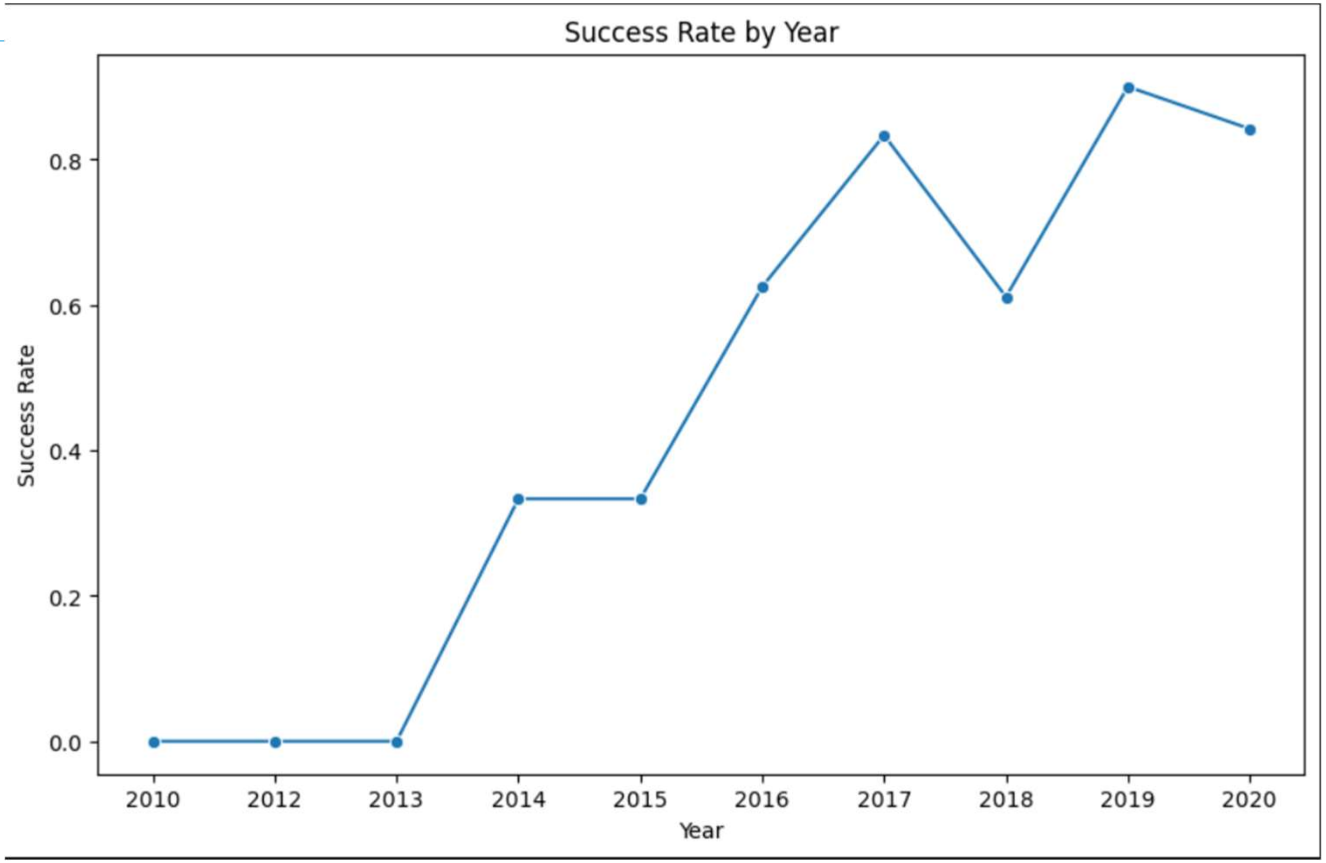
	FlightNumber	Date	BoosterVersion	PayloadMass	Orbit	LaunchSite	Outcome	Flights	GridFins	Reused	Legs	LandingPad	Block	ReusedCount	Serial	Longitude	Latitude
0	1	2006-03-24	Falcon 1	20.0	LEO	Kwajalein Atoll	None None	1	False	False	False	None	NaN	0	Merlin1A	167.743129	9.047721
1	2	2007-03-21	Falcon 1	NaN	LEO	Kwajalein Atoll	None None	1	False	False	False	None	NaN	0	Merlin2A	167.743129	9.047721
2	4	2008-09-28	Falcon 1	165.0	LEO	Kwajalein Atoll	None None	1	False	False	False	None	NaN	0	Merlin2C	167.743129	9.047721
3	5	2009-07-13	Falcon 1	200.0	LEO	Kwajalein Atoll	None None	1	False	False	False	None	NaN	0	Merlin3C	167.743129	9.047721
4	6	2010-06-04	Falcon 9	NaN	LEO	CCAFS SLC 40	None None	1	False	False	False	None	1.0	0	B0003	-80.577366	28.561857

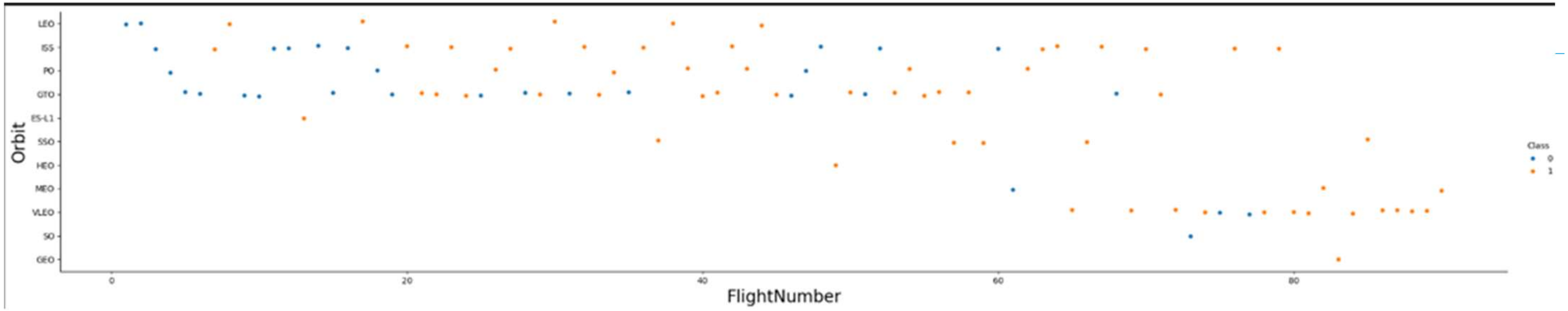
EDA



With heavy payloads, the successful landing or positive landing rate are more Polar, LEO, and ISS. However, for GTO it’s difficult to distinguish between successful and unsucessful landings as both outcomes are present

The success rate increases continually from 2010 to 2020.



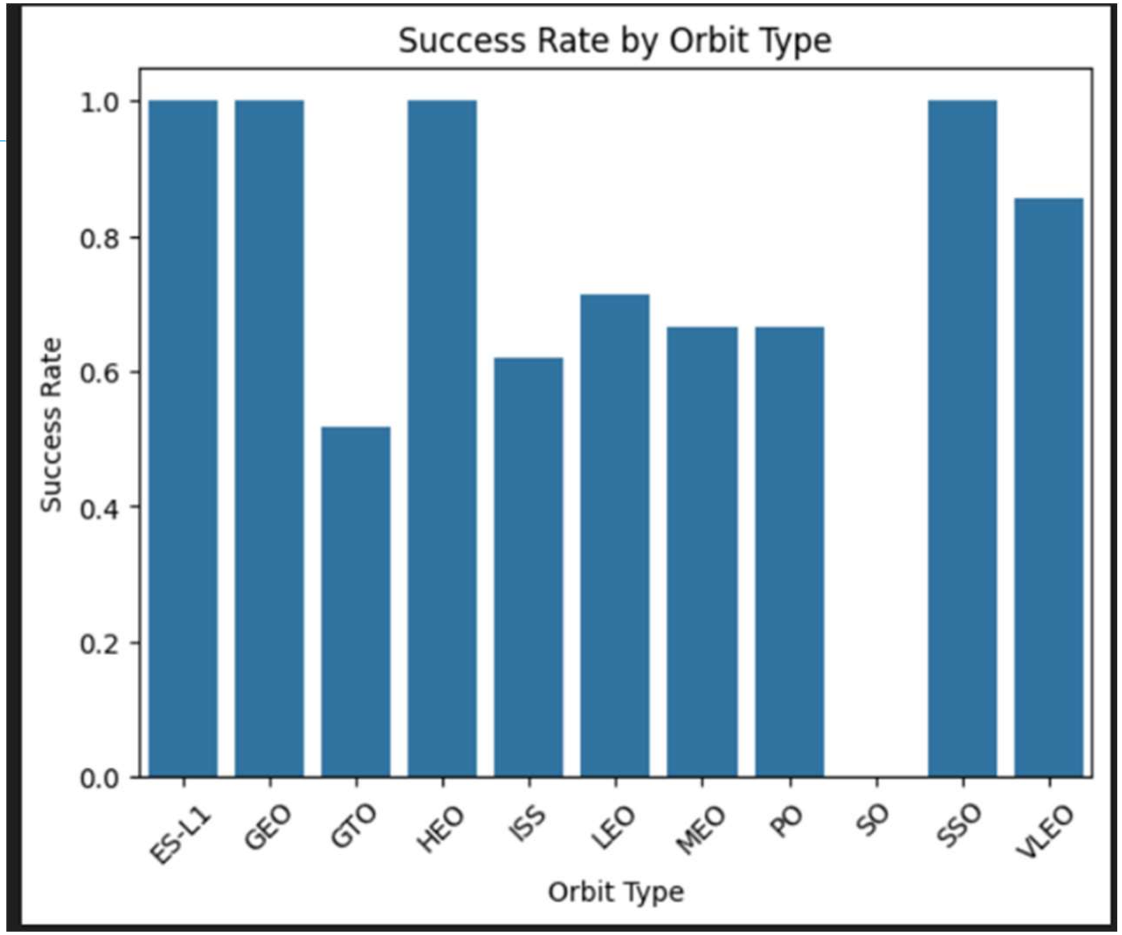


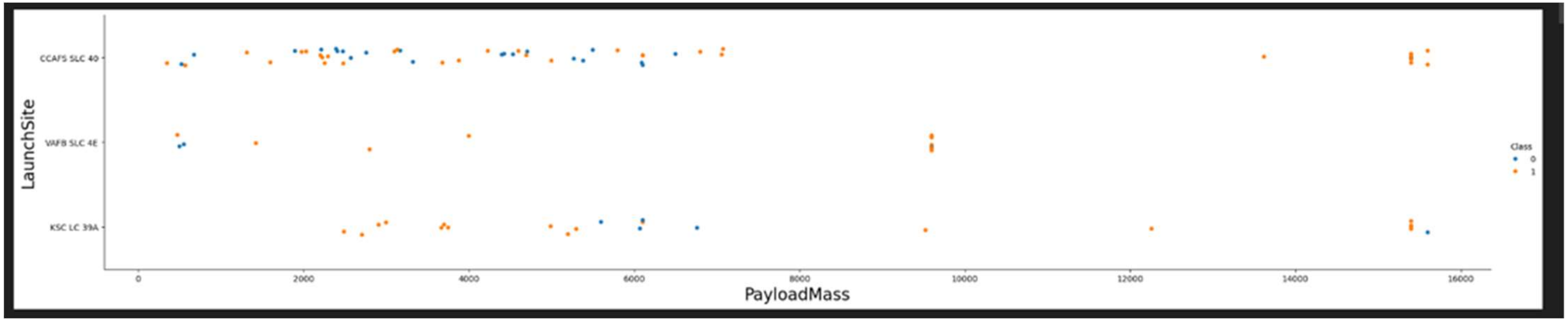
We can observe that in the LEO orbit, success seems to be related to the number of flights.

Conversely, in the GTO orbit, there appears to be no relationship between flight number and success.

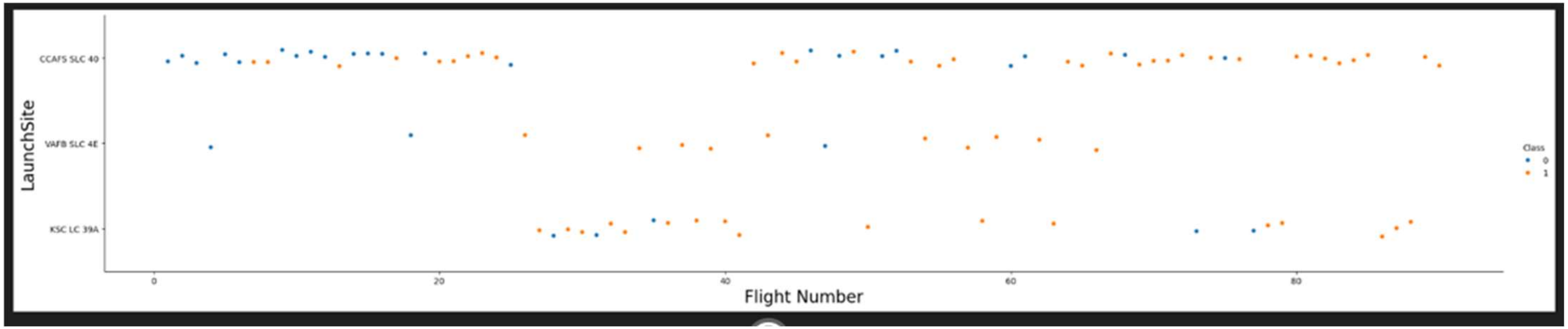
The orbits that have the highest success rate are:

SSO,HEO, ES-L1 and GEO

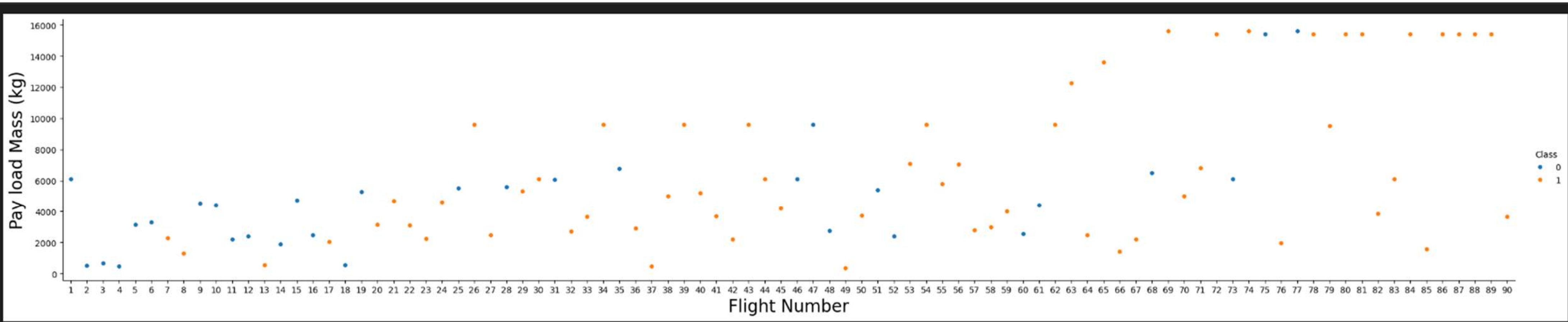




We can see that for the VAFB-SLC Launchsite, there are no rockets launched for heavypayload mass(greater than 10000)



The success rate seems higher for the CCAFS SLC 40 Launchsite.



Next, let's drill down to each site visualize its detailed launch records.

EDA sql

34]:

```
%sql select distinct Launch_Site from SPACEXTBL;
```

* sqlite:///my_data1.db
Done.

34]:

Launch_Site
CCAFS LC-40
VAFB SLC-4E
KSC LC-39A
CCAFS SLC-40

Task 1

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

```
[ ]: %sql select distinct Landing_outcome from SPACEXTBL where Date is not null
```

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

Landing_Outcome
Failure (parachute)
No attempt
Uncontrolled (ocean)
Controlled (ocean)
Failure (drone ship)
Precluded (drone ship)
Success (ground pad)
Success (drone ship)
Success
Failure
No attempt

Task 4

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

```
[94]: %sql select AVG(PAYLOAD_MASS__KG_) from SPACEXTBL WHERE booster_version ='F9 v1.1'
```

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

[94]:

AVG(PAYLOAD_MASS__KG_)
2928.4

Task 5

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

Hint:Use min function

[95]:
```
%sql select Date FROM SPACEXTBL where Landing_Outcome='Success (ground pad)' order by Date DESC LIMIT 1;
```

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

[95]:

Date
2018-01-08

Task 6

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

[96]:
```
%sql select Booster_Version, Landing_Outcome, PAYLOAD_MASS__KG_ FROM SPACEXTBL where Landing_Outcome ='Success (drone ship)' and PAYLOAD_MASS__KG_ BETWEEN 4000 AND 6000;
```

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

[96]:

Booster_Version	Landing_Outcome	PAYLOAD_MASS__KG_
F9 FT B1022	Success (drone ship)	4696
F9 FT B1026	Success (drone ship)	4600
F9 FT B1021.2	Success (drone ship)	5300
F9 FT B1031.2	Success (drone ship)	5200

Task 7

List the total number of successful and failure mission outcomes

[97]:
```
%sql SELECT SUM(CASE WHEN Landing_Outcome LIKE '%Success%' THEN 1 ELSE 0 END) AS Success_Count, SUM(CASE WHEN Landing_Outcome LIKE '%Failure%' THEN 1 ELSE 0 END) AS Failure_Count FROM SPACEXTBL;
```

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

[97]:

Success_Count	Failure_Count
61	10

Task 8

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

```
%sql Select Booster_version, PAYLOAD_MASS__KG_ FROM SPACEXTBL where PAYLOAD_MASS__KG_=(Select MAX(PAYLOAD_MASS__KG_) FROM SPACEXTBL)
```

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

Booster_Version	PAYLOAD_MASS__KG_
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

Task 9

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.

Note: SQLLite does not support monthnames. So you need to use substr(Date, 6,2) as month to get the months and substr(Date,0,5)='2015' for year.

```
%sql SELECT substr(Date, 6, 2) AS Month, substr(Date, 1, 4) AS Year, Landing_Outcome, Booster_Version, Launch_Site FROM SPACEXTBL WHERE substr(Date, 1, 4) = '2015';
```

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

Month	Year	Landing_Outcome	Booster_Version	Launch_Site
01	2015	Failure (drone ship)	F9 v1.1 B1012	CCAFS LC-40
02	2015	Controlled (ocean)	F9 v1.1 B1013	CCAFS LC-40
03	2015	No attempt	F9 v1.1 B1014	CCAFS LC-40
04	2015	Failure (drone ship)	F9 v1.1 B1015	CCAFS LC-40
04	2015	No attempt	F9 v1.1 B1016	CCAFS LC-40
06	2015	Precluded (drone ship)	F9 v1.1 B1018	CCAFS LC-40
12	2015	Success (ground pad)	F9 FT B1019	CCAFS LC-40

Task 10

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 Count(*) FROM SPACEXTBL where Date BETWEEN '2010-06-04' and '2017-03-20'and Landing_Outcome in ('Failure (drone ship)','Success (ground pad)') group by Landing_Outcome Order by Count(*) DESC;
```

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

Count(*)
5
3

Task 2

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

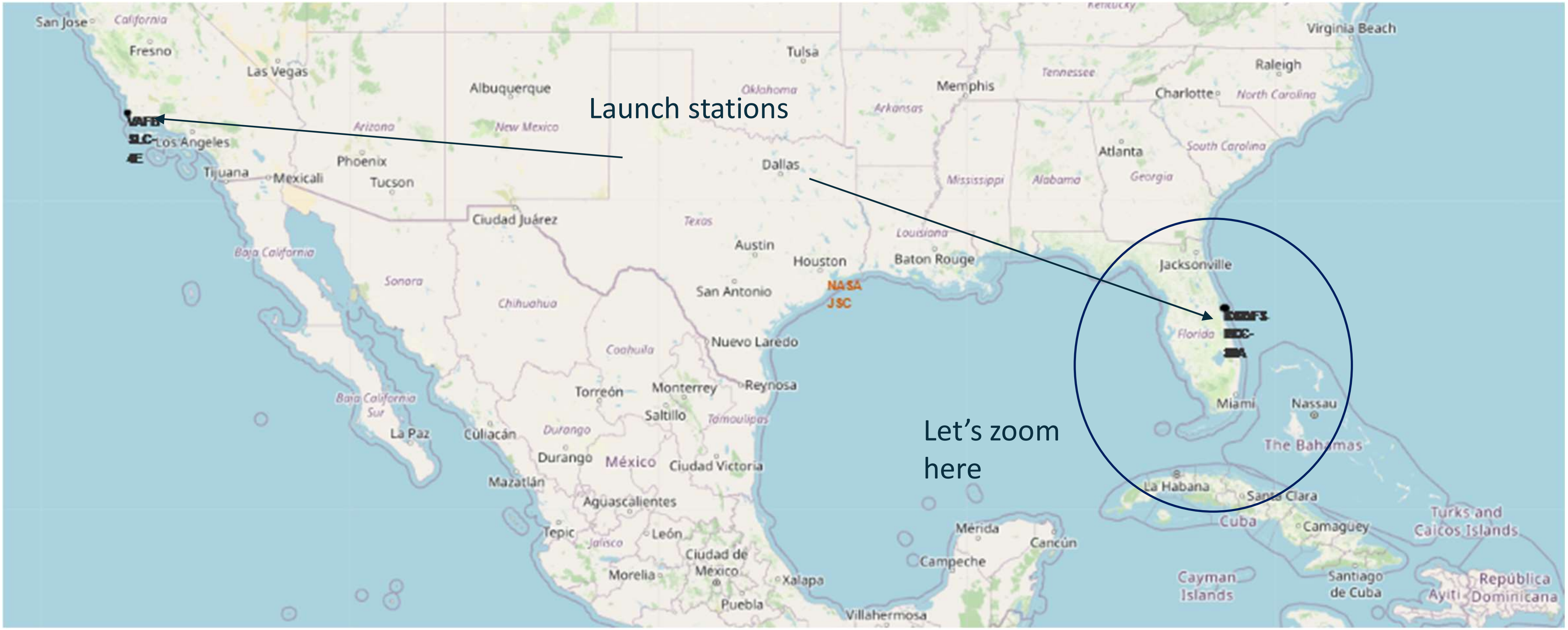
92]:
```
%sql select *  from SPACEXTBL where Launch_site like 'CCA%'  AND Date is not null
```

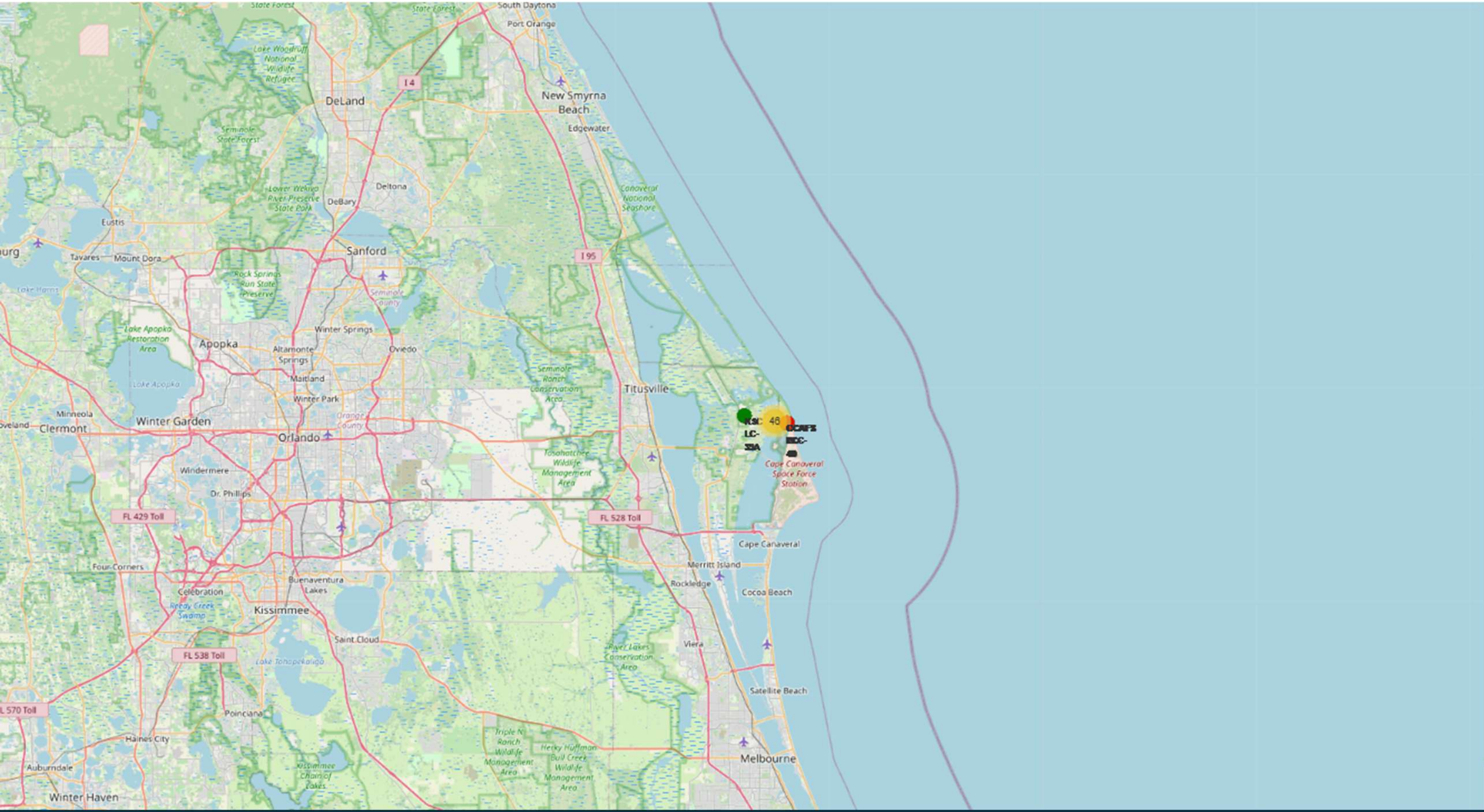
```
 * sqlite:///my_data1.db
Done.
```

92]:

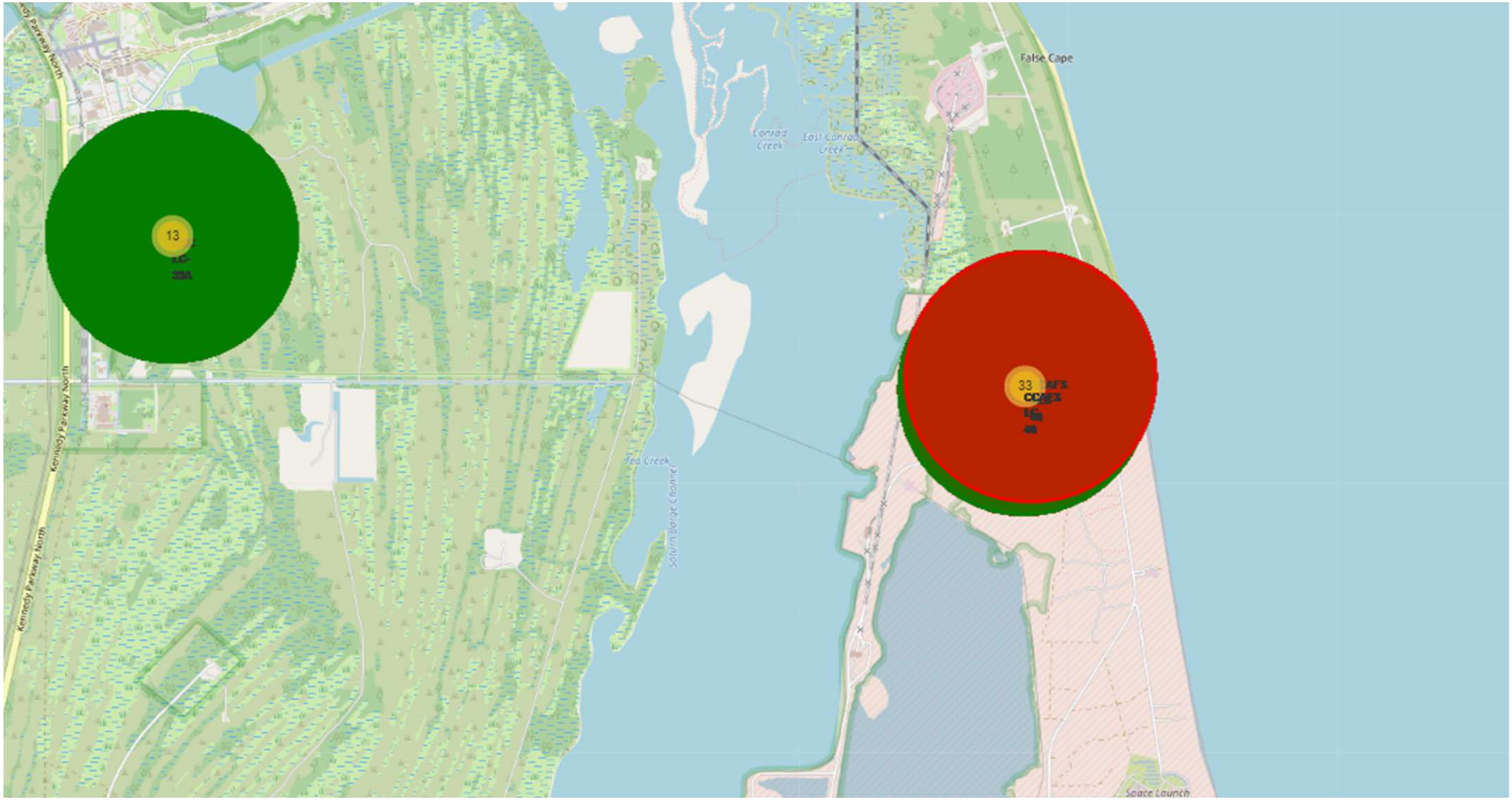
Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Orbit	Customer	Mission_Outcome	Landing_Outcome
2010-06-04	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success	Failure (parachute)
2010-12-08	15:43:00	F9 v1.0 B0004	CCAFS LC-40	Dragon demo flight 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
2013-12-03	22:41:00	F9 v1.1	CCAFS LC-40	SES-8	3170	GTO	SES	Success	No attempt
2014-01-06	22:06:00	F9 v1.1	CCAFS LC-40	Thaicom 6	3325	GTO	Thaicom	Success	No attempt
2014-04-18	19:25:00	F9 v1.1	CCAFS LC-40	SpaceX CRS-3	2296	LEO (ISS)	NASA (CRS)	Success	Controlled (ocean)
2014-07-14	15:15:00	F9 v1.1	CCAFS LC-40	OG2 Mission 1 6 Orbcomm-OG2 satellites	1316	LEO	Orbcomm	Success	Controlled (ocean)
2014-08-05	8:00:00	F9 v1.1	CCAFS LC-40	AsiaSat 8	4535	GTO	AsiaSat	Success	No attempt
2014-09-07	5:00:00	F9 v1.1 B1011	CCAFS LC-40	AsiaSat 6	4428	GTO	AsiaSat	Success	No attempt

Interactive map

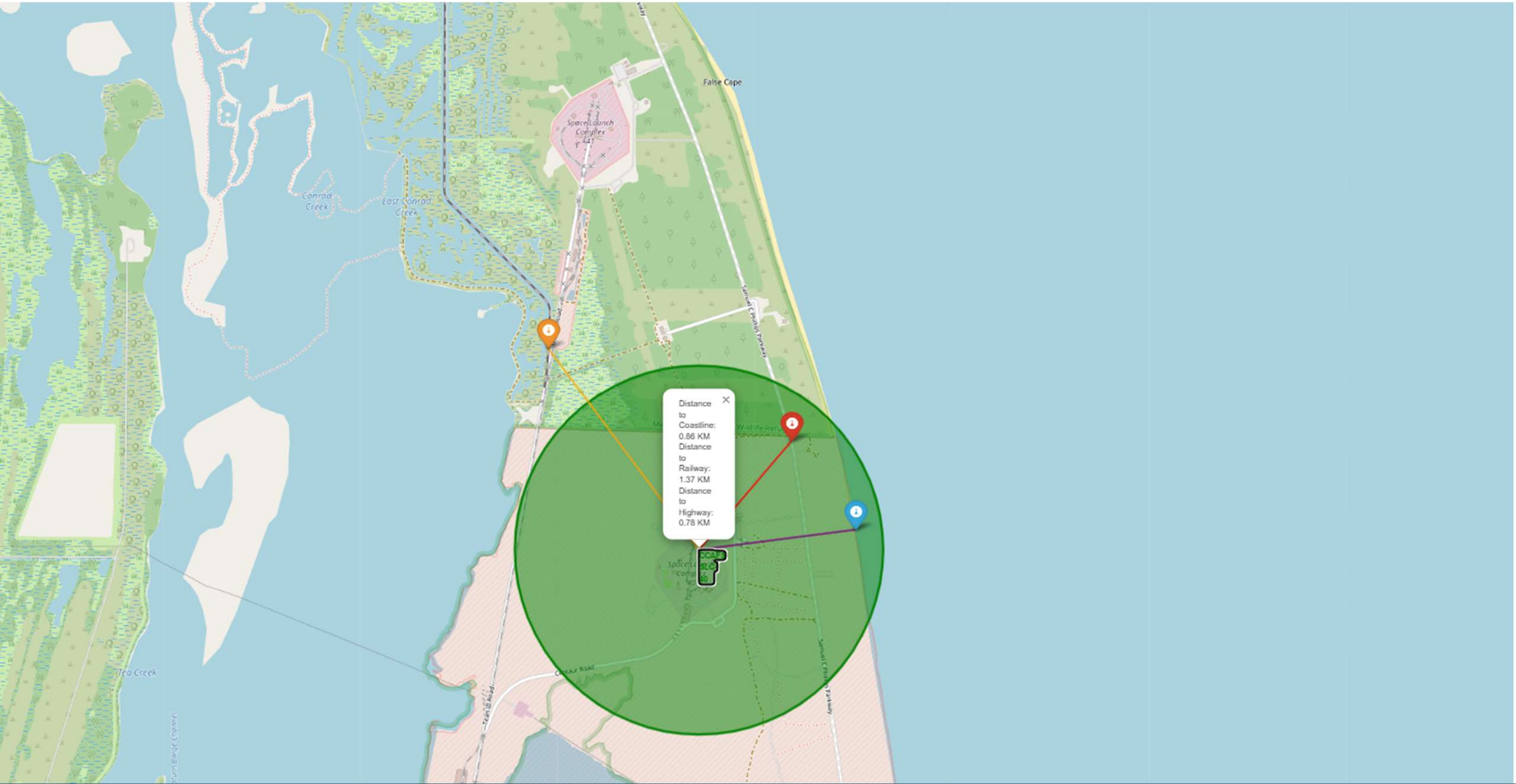

Launch stations
Let's zoom here
NASA
JSC
San Jose
California
Fresno
Las Vegas
Albuquerque
Tulsa
Oklahoma
Memphis
Tennessee
Virginia Beach
Raleigh
Charlotte
North Carolina
Arkansas
Arizona
New Mexico
Los Angeles
Tijuana
Mexicali
Phoenix
Tucson
Dallas
Atlanta
South Carolina
Georgia
Mississippi
Alabama
Ciudad Juárez
Texas
Louisiana
Austin
Houston
Baton Rouge
Jacksonville
Baja California
Sonora
Chihuahua
San Antonio
Florida
Coahuila
Nuevo Laredo
Torreón
Monterrey
Reynosa
Saltillo
Tamaulipas
Miami
Nassau
Baja California Sur
La Paz
Culiacán
Durango
México
Ciudad Victoria
The Bahamas
Mazatlán
Aguascalientes
La Habana
Santa Clara
Turks and Caicos Islands
Tepic
Jalisco
León
Mérida
Cancún
Cuba
Camaguey
Ciudad de México
Campeche
Morelia
Xalapa
Cayman Islands
Santiago de Cuba
Ayiti
República Dominicana
Puebla
Villahermosa



South Daytona
Port Orange
State Forest
Lake Woodruff National Wildlife Refuge
I 4
DeLand
New Smyrna Beach
Edgewater
Seminole State Forest
Deltona
DeBary
Lower Wekiva River Preserve State Park
Canaveral National Seashore
Eustis
Tavares
Mount Dora
Sanford
I 95
Rock Springs Run State Preserve
Seminole County
Lake Harris
Lake Apopka Restoration Area
Apopka
Winter Springs
Altamonte Springs
Oviedo
Maitland
Winter Park
Lake Apopka
Seminole Ranch Conservation Area
Titusville
Minneola
Winter Garden
Clermont
Orlando
Orange County
46
Tosohatchee Wildlife Management Area
Cape Canaveral Space Force Station
Windermere
Dr. Phillips
FL 429 Toll
FL 528 Toll
Cape Canaveral
Merritt Island
Rockledge
Four Corners
Buenaventura Lakes
Celebration
Reedy Creek Swamp
Kissimmee
Cocoa Beach
Saint Cloud
FL 538 Toll
Viera
River Lakes Conservation Area
Lake Tohopekaliga
Satellite Beach
FL 570 Toll
Poinciana
Haines City
Triple N Ranch Wildlife Management Area
Herky Huffman Bull Creek Wildlife Management Area
Melbourne
Auburndale
Kissimmee Chain of Lakes
Winter Haven



False Cape
Conrad Creek
East Conrad Creek
13
LC-39A
33
CCAFS
Kennedy Parkway North
Space Launch

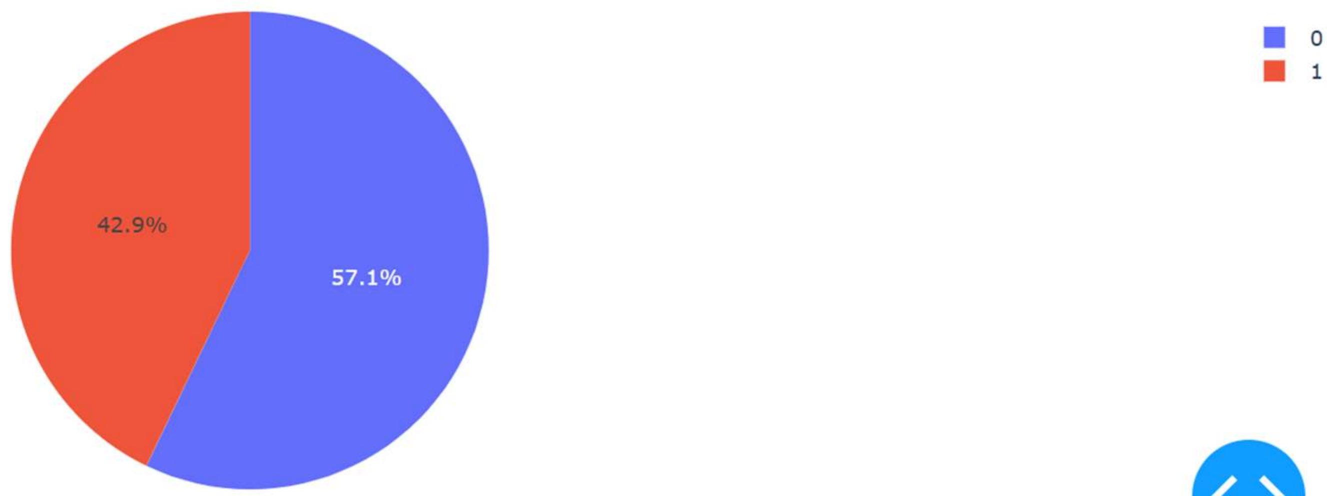

False Cape
Space Launch Complex 41
Conrad Creek
East Conrad Creek
Distance to Coastline: 0.86 KM
Distance to Railway: 1.37 KM
Distance to Highway: 0.78 KM
Samuel C Phillips Parkway
Tea Creek
Contour Road

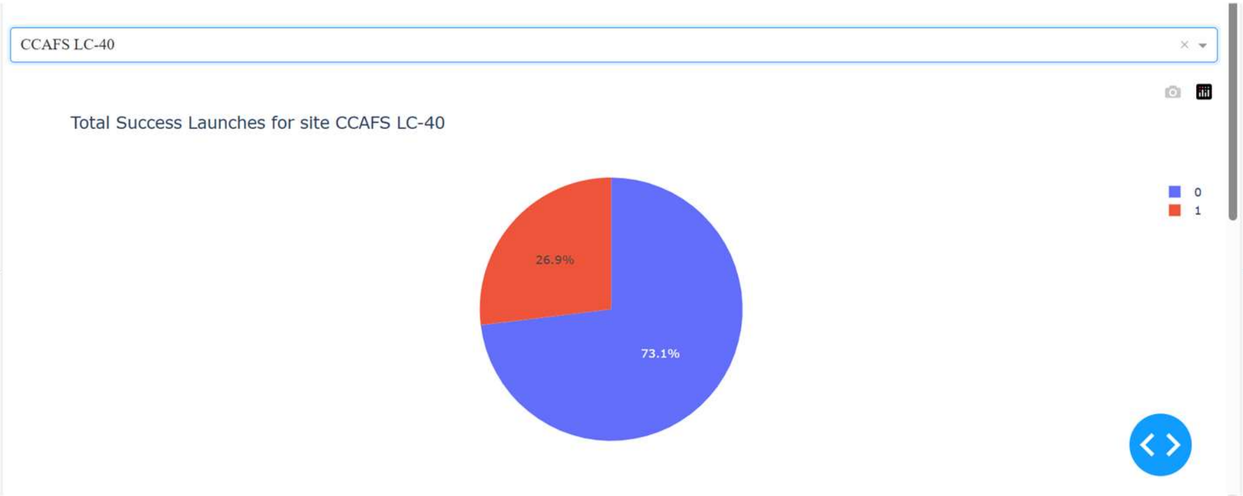
Plotly

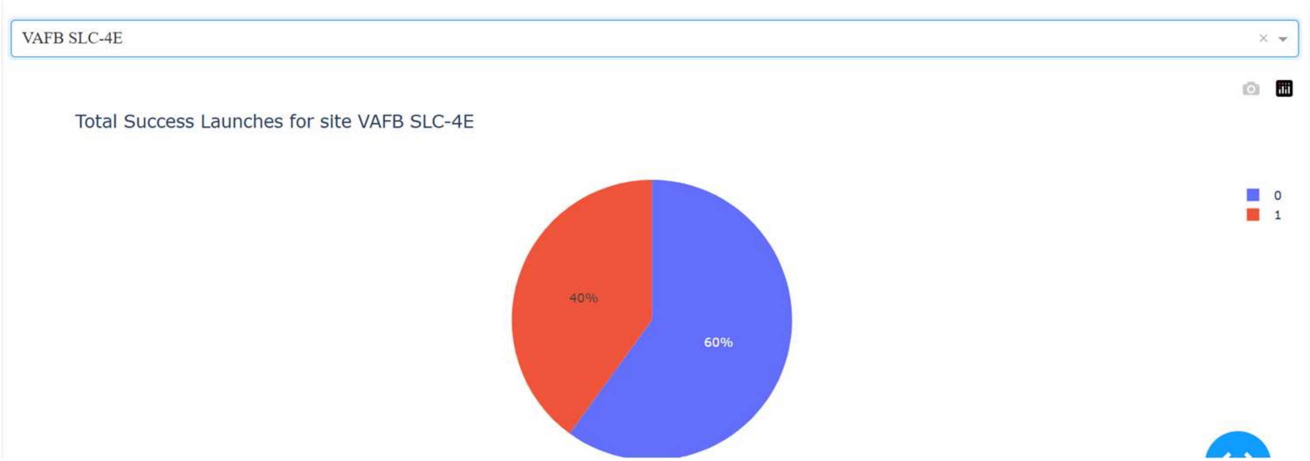
SpaceX Launch Records Dashboard

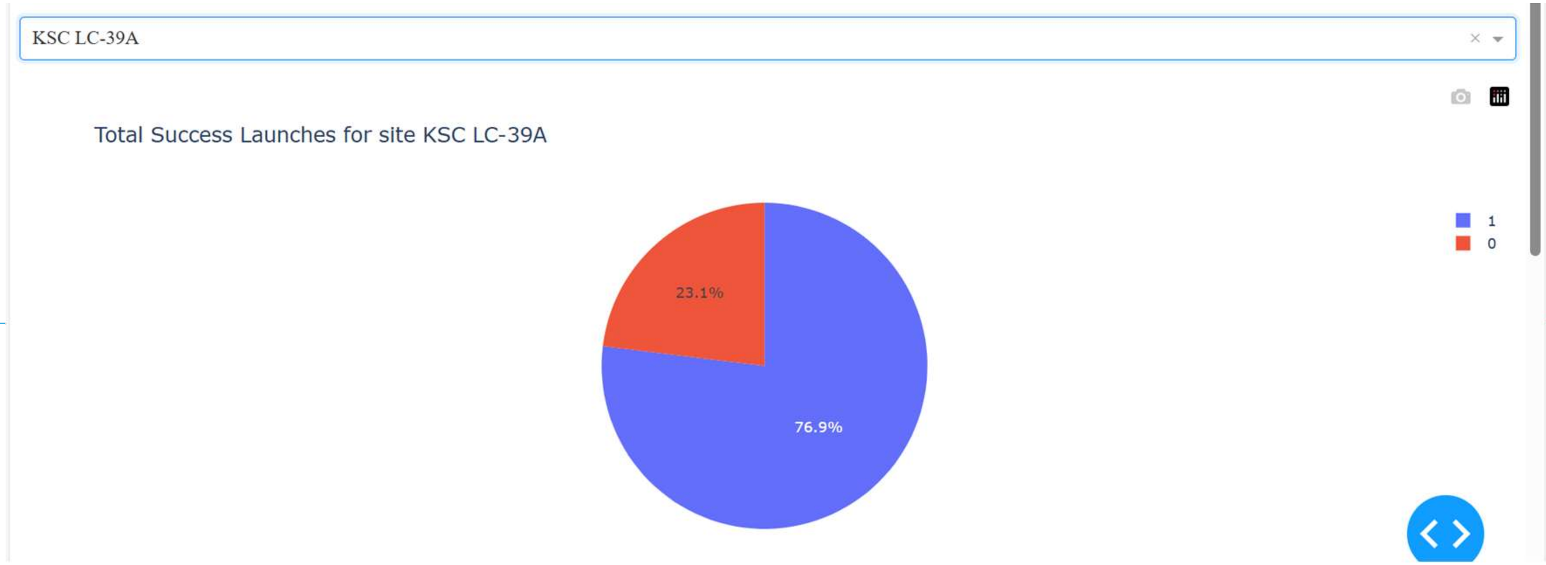
All Sites

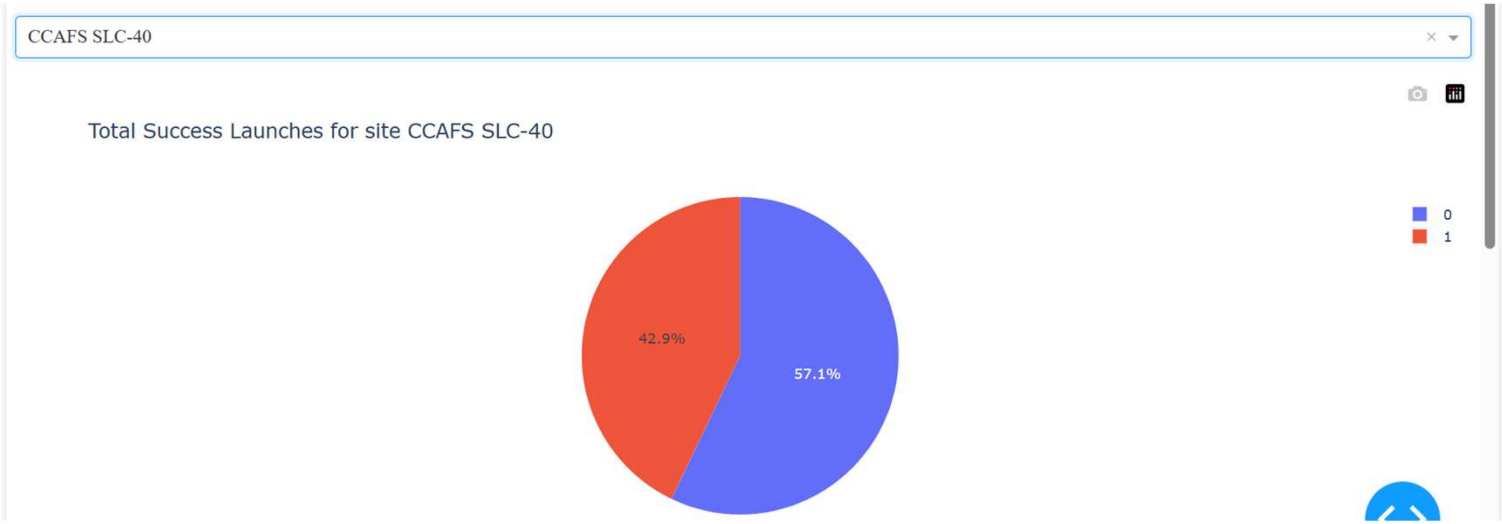
Total Success Launches for All Sites



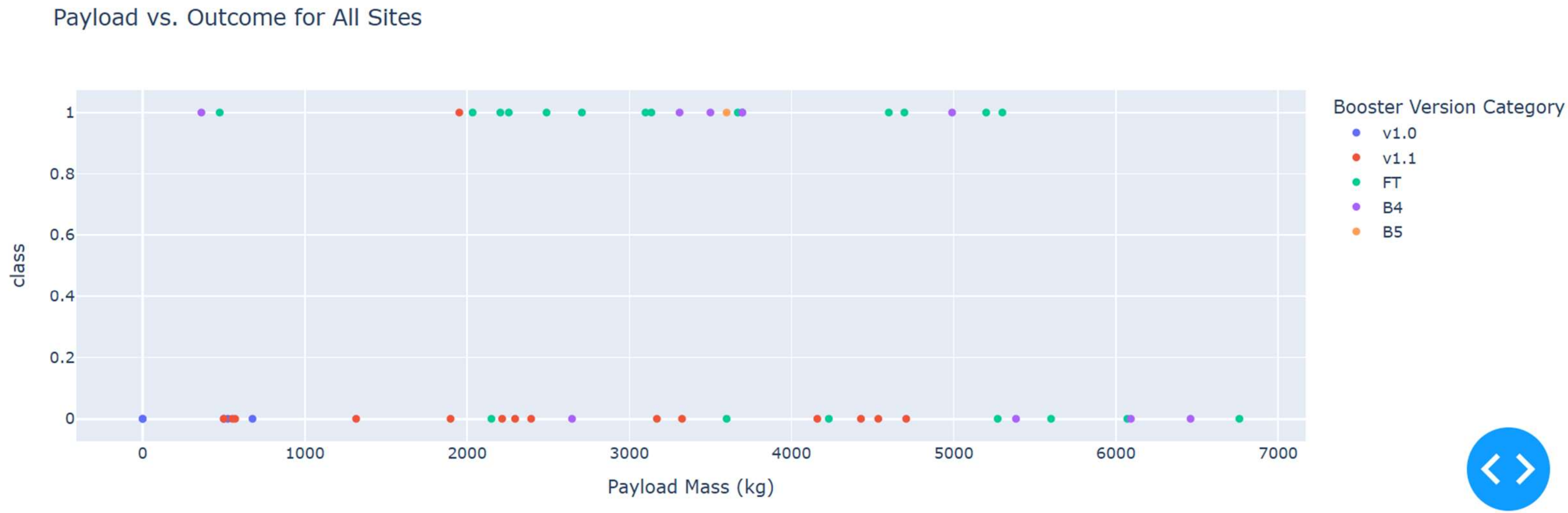

CCAFS LC-40
Total Success Launches for site CCAFS LC-40
0
1
26.9%
73.1%

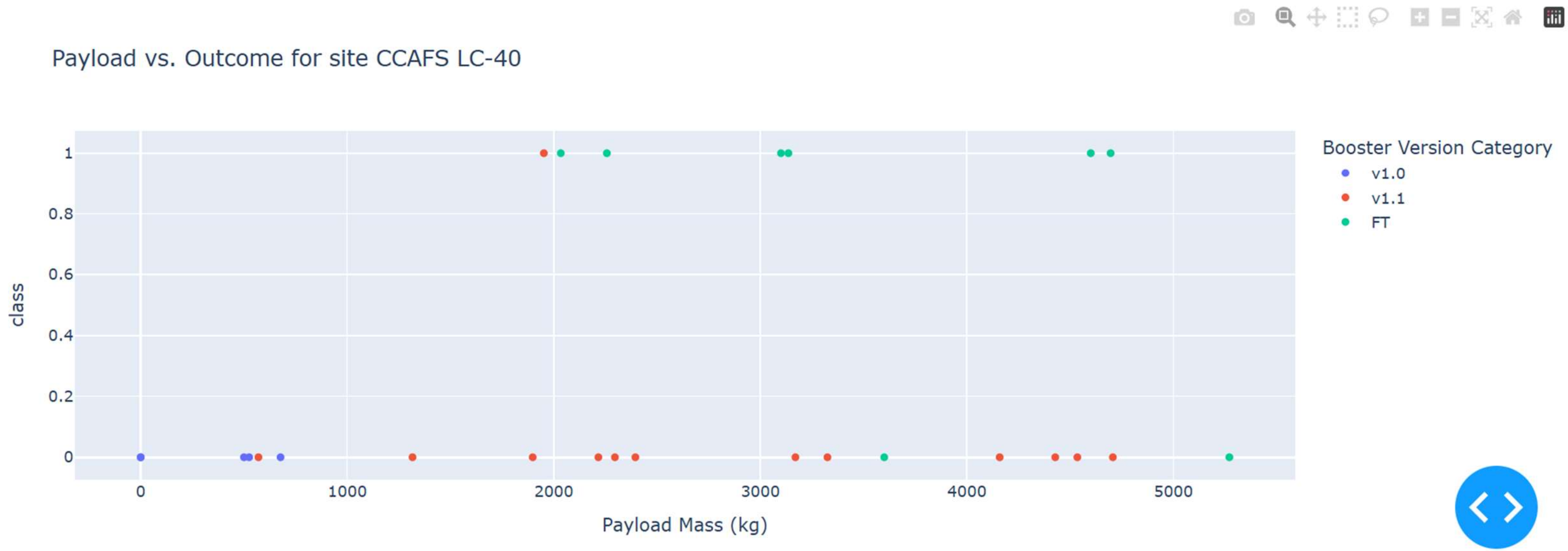

VAFB SLC-4E
Total Success Launches for site VAFB SLC-4E
0
1
40%
60%


KSC LC-39A
Total Success Launches for site KSC LC-39A
1
0
23.1%
76.9%

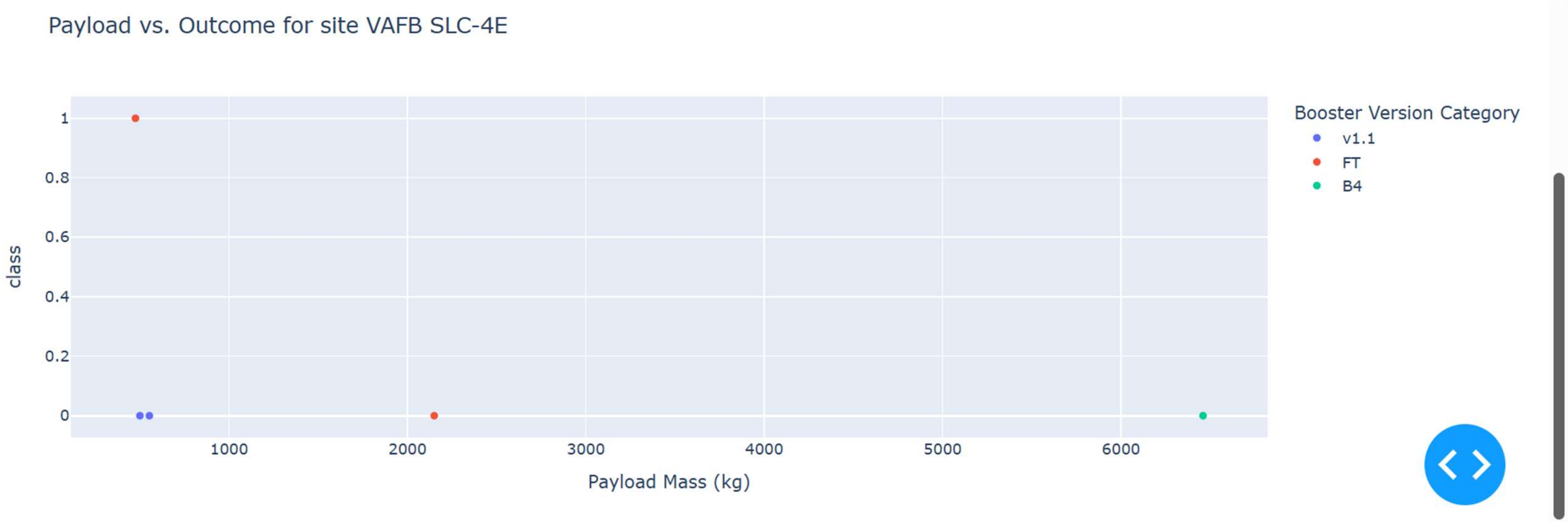

CCAFS SLC-40
Total Success Launches for site CCAFS SLC-40
0
1
42.9%
57.1%

Payload range (Kg):

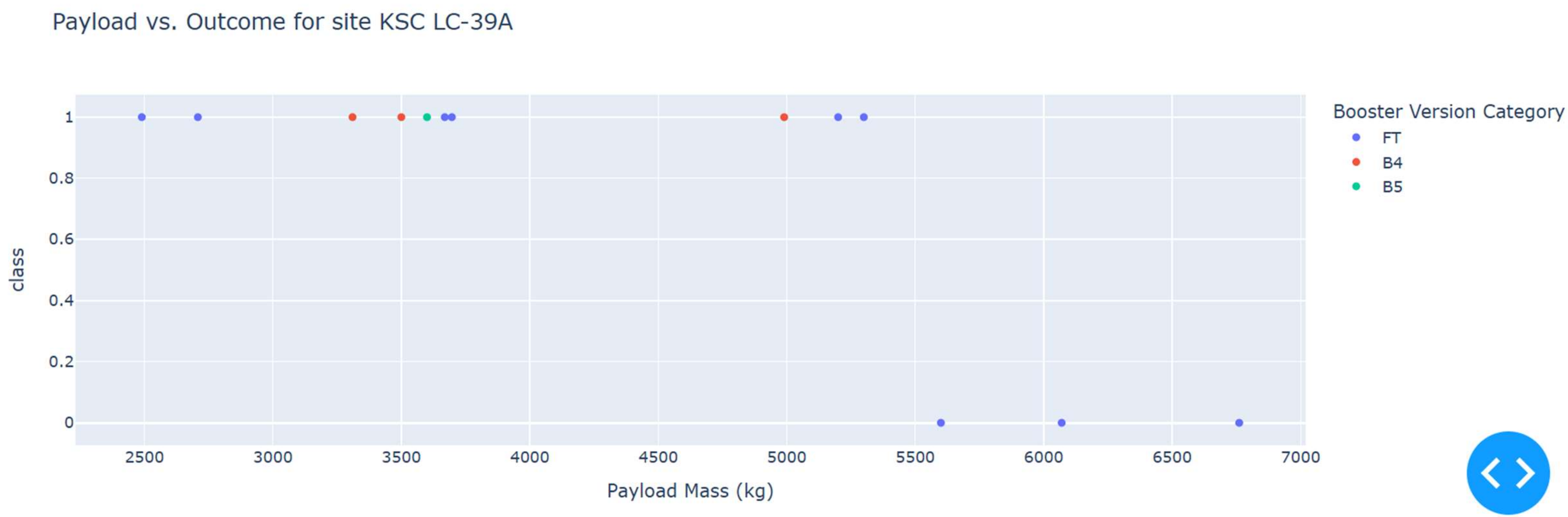


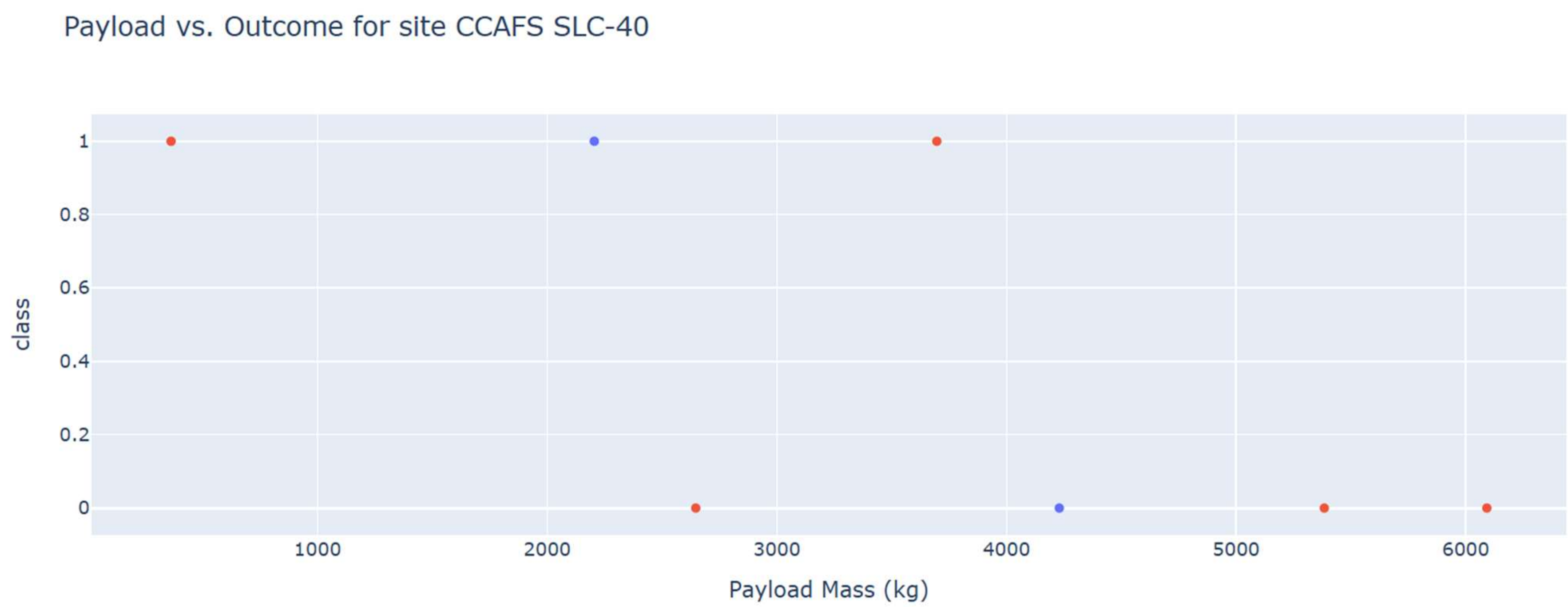


Payload vs. Outcome for site CCAFS LC-40
class
1
0.8
0.6
0.4
0.2
0
0
1000
2000
3000
4000
5000
Payload Mass (kg)
Booster Version Category
v1.0
v1.1
FT



Payload vs. Outcome for site VAFB SLC-4E
class
1
0.8
0.6
0.4
0.2
0
1000
2000
3000
4000
5000
6000
Payload Mass (kg)
Booster Version Category
v1.1
FT
B4


Payload vs. Outcome for site KSC LC-39A
class
1
0.8
0.6
0.4
0.2
0
2500
3000
3500
4000
4500
5000
5500
6000
6500
7000
Payload Mass (kg)
Booster Version Category
FT
B4
B5



Payload vs. Outcome for site CCAFS SLC-40
class
1
0.8
0.6
0.4
0.2
0
1000
2000
3000
4000
5000
6000
Payload Mass (kg)
Booster Version Category
FT
B4

Predictive analysis results

Logistic regression

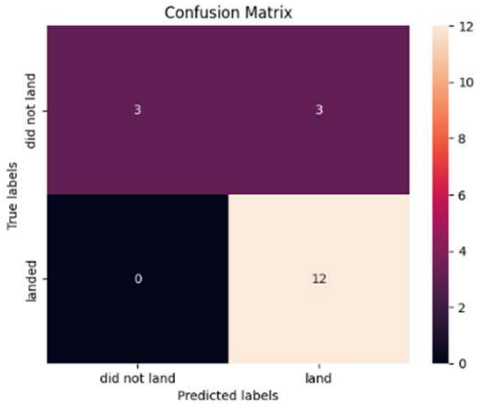
Calculate the accuracy on the test data using the method `score` :

```
# Calculate the accuracy on the test data
logreg_test_accuracy = logreg_cv.score(X_test, Y_test)
print("Accuracy on test data: ", logreg_test_accuracy)
```

```
Accuracy on test data:  0.8333333333333334
```

Lets look at the confusion matrix:

```
yhat=logreg_cv.predict(X_test)
plot_confusion_matrix(Y_test,yhat)
```



Examining the confusion matrix, we see that logistic regression can distinguish between the different classes. We see that the problem is false positives.

Overview:

True Postive - 12 (True label is landed, Predicted label is also landed)

False Postive - 3 (True label is not landed, Predicted label is landed)

SVM

TASK 7

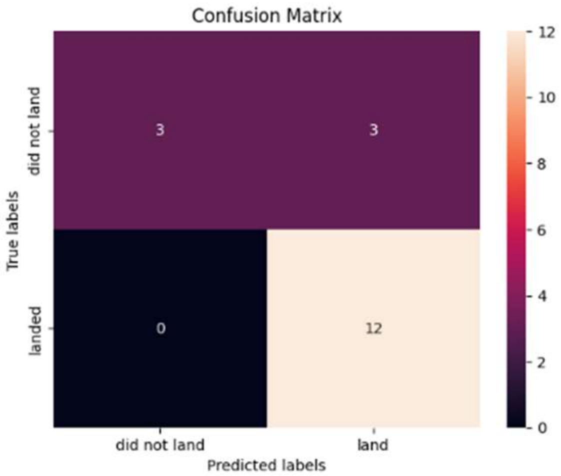
Calculate the accuracy on the test data using the method `score` :

```
[18]: # Calculate the accuracy on the test data
      svm_test_accuracy = svm_cv.score(X_test, Y_test)
      print("Accuracy on test data: ", svm_test_accuracy)
```

```
Accuracy on test data:  0.8333333333333334
```

We can plot the confusion matrix

```
[19]: yhat=svm_cv.predict(X_test)
      plot_confusion_matrix(Y_test,yhat)
```



Decision Tree

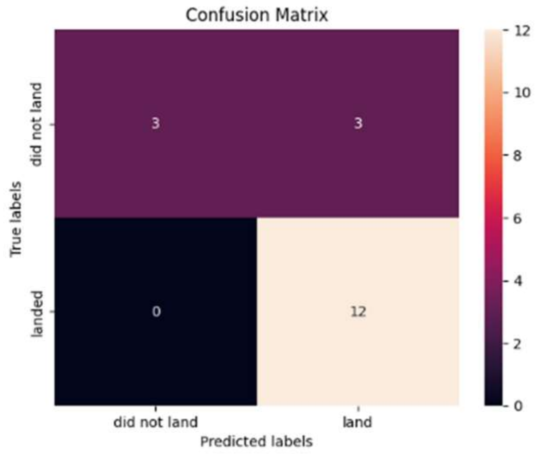
Calculate the accuracy of tree_cv on the test data using the method `score` :

```
[21]: # Calculate the accuracy on the test data
tree_test_accuracy = tree_cv.score(X_test, Y_test)
print("Accuracy on test data: ", tree_test_accuracy)
```

```
Accuracy on test data:  0.8333333333333334
```

We can plot the confusion matrix

```
[22]: yhat = tree_cv.predict(X_test)
plot_confusion_matrix(Y_test,yhat)
```



KNN

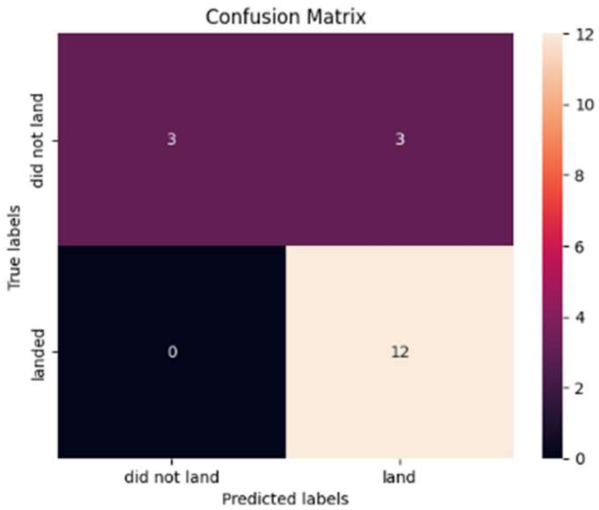
Calculate the accuracy of knn_cv on the test data using the method `score`:

```
[24]: # Calculate the accuracy on the test data
knn_test_accuracy = knn_cv.score(X_test, Y_test)
print("Accuracy on test data: ", knn_test_accuracy)
```

```
Accuracy on test data:  0.8333333333333334
```

We can plot the confusion matrix

```
[25]: yhat = knn_cv.predict(X_test)
plot_confusion_matrix(Y_test,yhat)
```



```
[27]: # Summary of results
results = {
    "Logistic Regression": logreg_test_accuracy,
    "Support Vector Machine": svm_test_accuracy,
    "Decision Tree": tree_test_accuracy,
    "K-Nearest Neighbors": knn_test_accuracy,
}

# Print out the results
for model, accuracy in results.items():
    print(f"{model}: Accuracy = {accuracy:.4f}")
```

```
Logistic Regression: Accuracy = 0.8333
Support Vector Machine: Accuracy = 0.8333
Decision Tree: Accuracy = 0.8333
K-Nearest Neighbors: Accuracy = 0.8333
```

Conclusion

The analysis of historical data from the SpaceX API revealed a steady improvement in Falcon 9 first-stage landing success rates over time. Early missions saw frequent failures due to the complexity of the technology and landing process. However, with each iteration, SpaceX has refined its techniques, leading to a significantly higher success rate in recent years. This trend indicates a learning curve in the development of reusable rockets, with a correlation between mission experience and landing success. The analysis showed that missions targeting low Earth orbits with lighter payloads have a significantly higher landing success rate.

Several factors were found to influence the likelihood of a successful landing. The weight of the payload, the mission's target orbit during launch all played crucial roles. Additionally, certain mission types, such as those requiring high orbits, reduced the chances of a successful landing

Using machine learning techniques, we developed a predictive model that accurately forecasts the success of a first-stage landing based on historical data. The model, which incorporated several factors like payload mass, orbits, etc.. and mission profile, achieved high accuracy in predicting outcomes. This predictive capability can be invaluable for companies like Space Y, allowing them to estimate the probability of a successful landing and better plan for the costs associated with rocket reusability.

Innovative insights

- New predictive factors: The integration of weather data into the forecasting models has identified significant correlations between certain atmospheric conditions (such as wind speed, atmospheric pressure, and temperature) and the chances of successful landings. These factors, often overlooked in traditional analyses, can greatly affect rocket stability during the descent and landing phases.