Understanding space launching with Data Science.

Marco Alegria – July 2023

Contents

Executive Summary

The analysis performed on the spacecraft launching data from SpaceX, gives us the following insights for considering new launches:

- There are 4 preferred launching sites for the space exploration missions:
    1. CCAFS LC-40
    2. VAFB SLC-4E
    3. KSC LC-39A
    4. CCAFS SLC-40

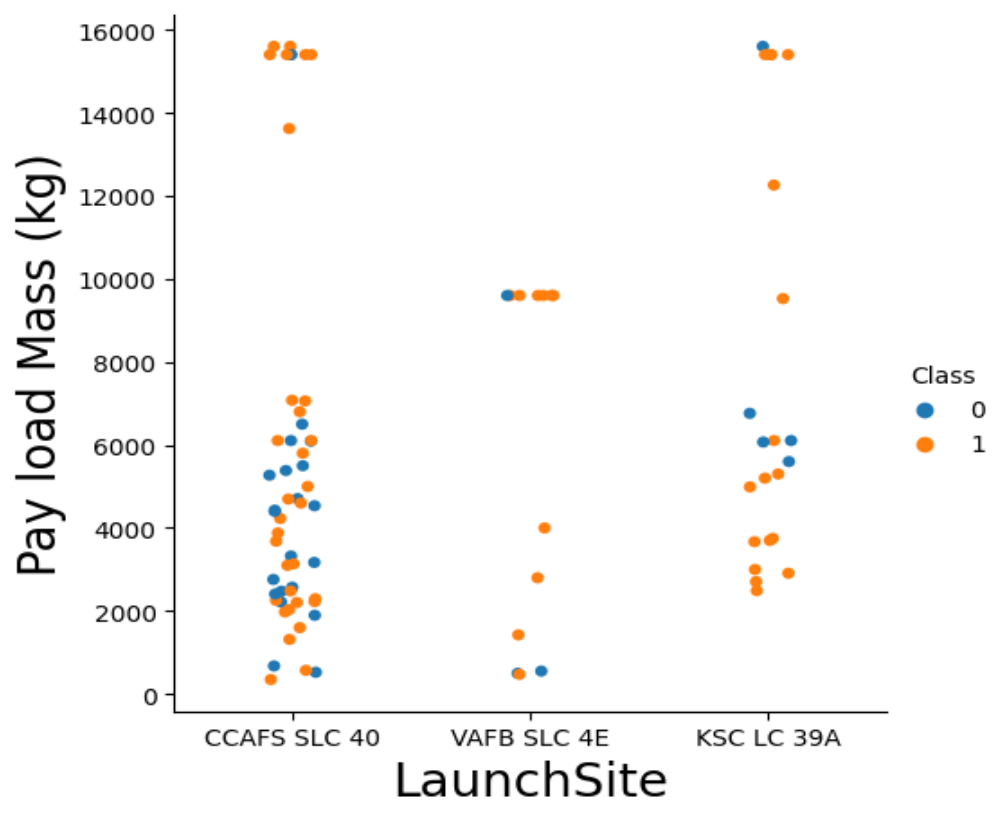
All the launching sites previously mentioned are on the nearest beach of the West and East coast.

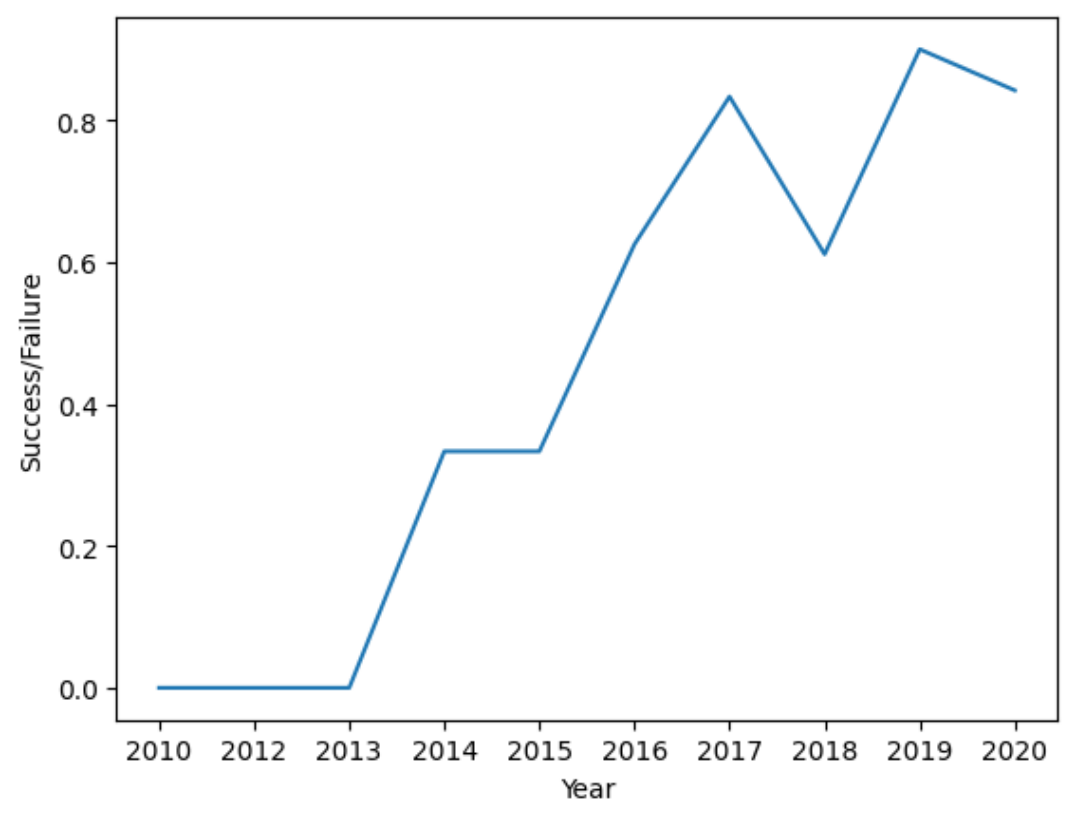
Executive Summary

The most used Launching site is Cape Canaveral Space Force station.

Executive Summary

- Thanks to the EDA we find that the most used Launching site is CCAFS SLC 40, however, KSC LC 39A shows that it has had more successful launches with less tries. This launching sites are the ones that have sent higher pay loads.
- Since the trials started on 2013, we can find the peak of the success rate was hit around 2019.



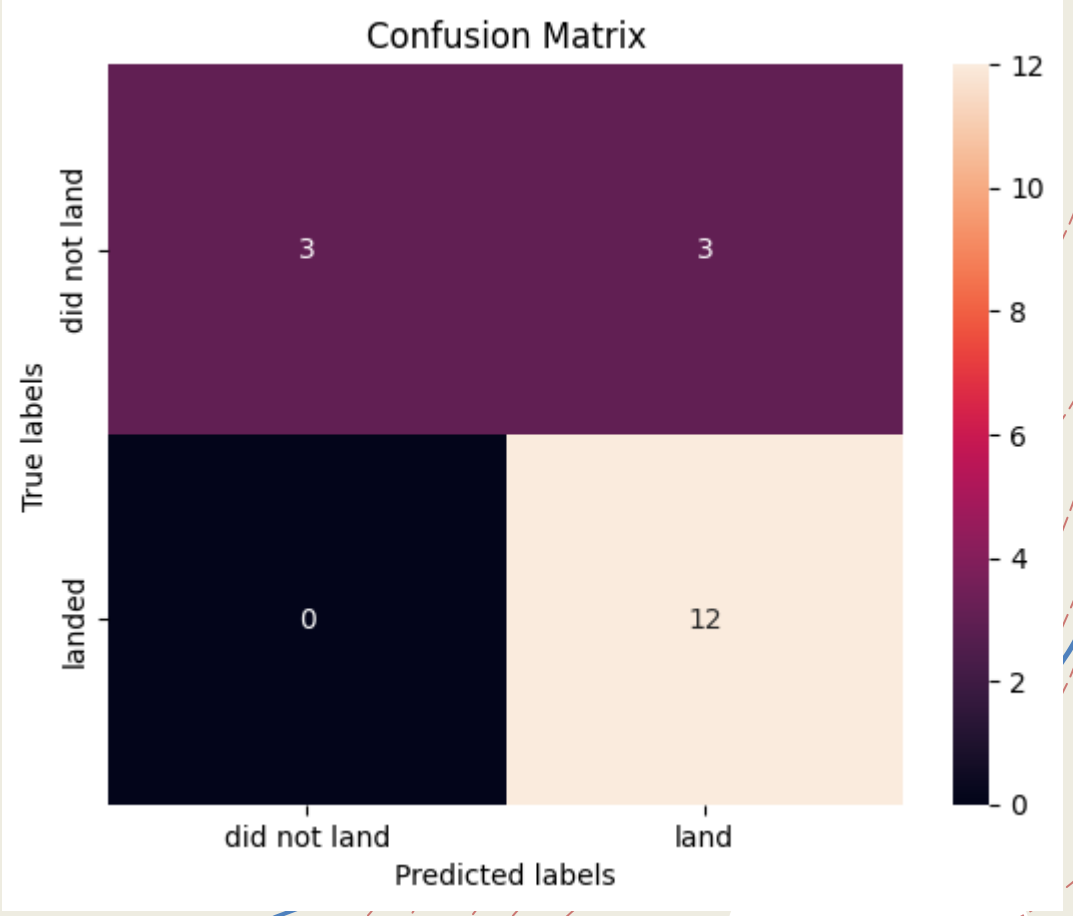


Executive Summary

With all of this information gathered we can provide a ML algorithm to predict the success of a launch, considering different parameters as:

- Launching site
- Payload mass
- Orbit used

Using different ML methods; Logreg, SVM, Decision tree and KNN. We can find that the performance of the prediction is basically the same with an 83% accuracy.

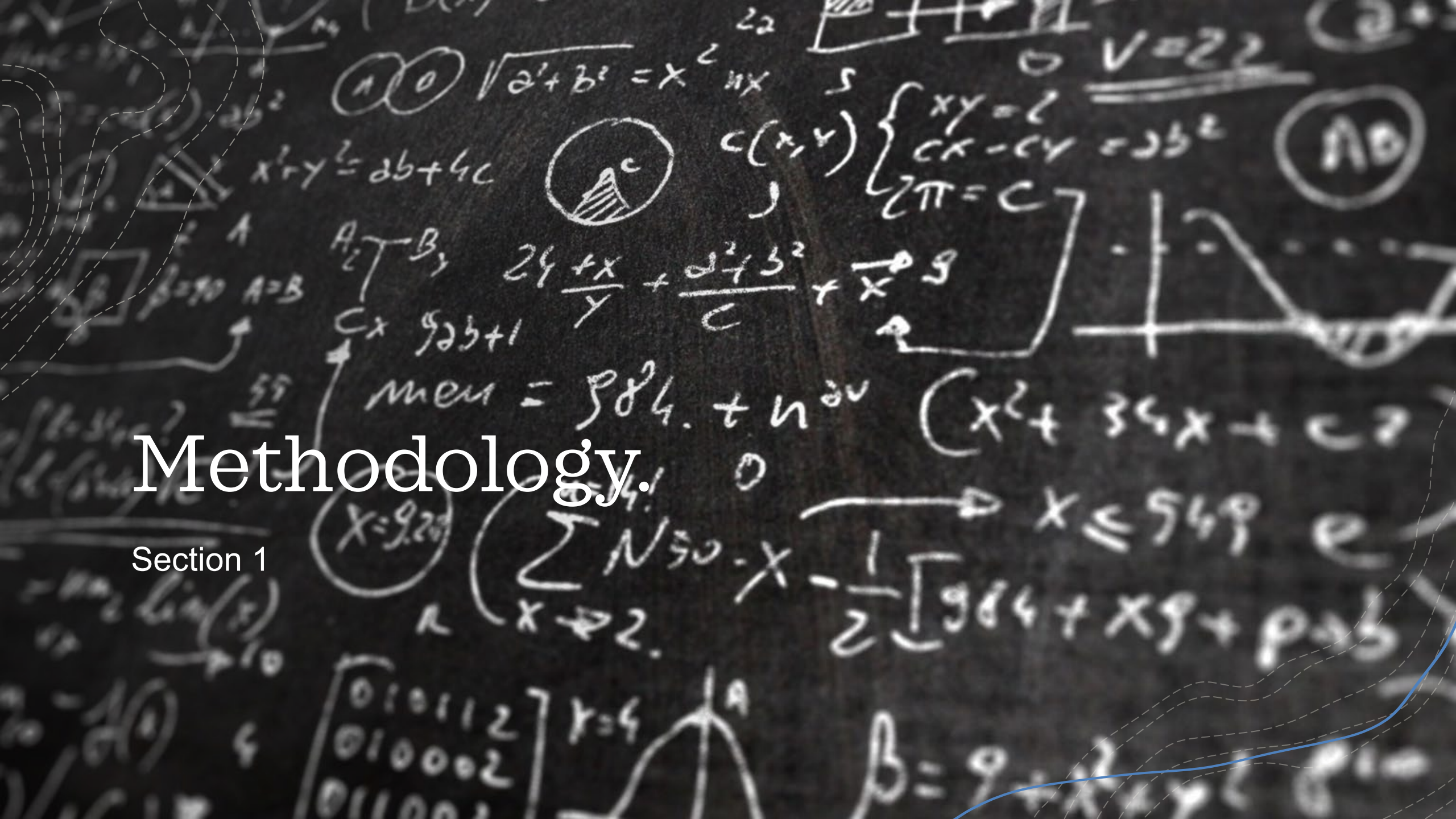


Introduction.

- Context?
- How can we find the best condition for a successful space launch?

What are we doing?

- The objective of the following analysis, is to use the data collected by Space X launches, to be able to model an algorithm that can identify the success of a rocket launch.
- All the Data analysis is performed on Jupyter Notebooks which can be found on my Git: Coursera_IBM_DataScience (github.com)
- The methodology used on this whole analysis will be explained as follows:



Methodology.

Section 1

Data collection.

- Step-by-step methodology:

1. Space X launch data collection from API and web scraping from Wikipedia.
2. The data wrangling is performed with the Pandas and NumPy libraries from python.

+ The data scraped was added to a Pandas Data Frame which was saved as a CSV file.

	Flight No.	Launch site	Payload	Payload mass	Orbit	Customer	Launch outcome	Version Booster	Booster landing	Date	Time
0	1	CCAFS	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success\n	F9 v1.0B0003.1	Failure	4 June 2010	18:45
1	2	CCAFS	Dragon	0	LEO	NASA (COTS)\nNRO	Success	F9 v1.0B0004.1	Failure	8 December 2010	15:43
2	3	CCAFS	Dragon	525 kg	LEO	NASA (COTS)	Success	F9 v1.0B0005.1	No attempt\n	22 May 2012	07:44
3	4	CCAFS	SpaceX CRS-1	4,700 kg	LEO	NASA (CRS)	Success\n	F9 v1.0B0006.1	No attempt	8 October 2012	00:35
4	5	CCAFS	SpaceX CRS-2	4,877 kg	LEO	NASA (CRS)	Success\n	F9 v1.0B0007.1	No attempt\n	1 March 2013	15:10
...	...	...	...	...	...	...	...	...	...	...	...
116	117	CCSFS	Starlink	15,600 kg	LEO	SpaceX	Success\n	F9 B5B1051.10	Success	9 May 2021	06:42
117	118	KSC	Starlink	~14,000 kg	LEO	SpaceX Capella Space and Tech	Success\n	F9 B5B1058.8	Success	15 May 2021	22:56

Data collection – SpaceX API

+API connection

- The API data collection uses *“Requests”* module from python to extract the data from the JSON format.
- API url: https://api.spacexdata.com/v4/launches/past"

Requests module

Use request module to connect to API.

Pandas

Convert JSON response to structured data frame.

Data Wrangling

Find NA values, replace for MEAN in column

Save to CSV

Data collection – Web Scraping

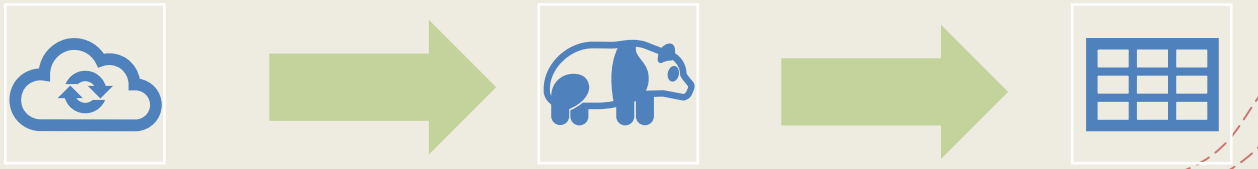
+Wikipedia SpaceX scraping

- Extract the information found on the Wikipedia page of the Falcon 9 & Falcon Heavy from SpaceX launches.
- Wiki url: [List of Falcon 9 and Falcon Heavy launches - Wikipedia]

2020 [edit]

In late 2019, Gwynne Shotwell stated that SpaceX hoped for as many as 24 launches for Starlink satellites in 2020,[490] in addition to 14 or 15 non-Starlink launches. At 26 launches, 13 of which for Starlink satellites, Falcon 9 had its most prolific year, and Falcon rockets were second most prolific rocket family of 2020, only behind China's Long March rocket family.[491]

[hide] Flight No.	Date and time (UTC)	Version, Booster[b]	Launch site	Payload[c]	Payload mass	Orbit	Customer	Launch outcome	Booster landing
78	7 January 2020, 02:19:21[492]	F9 B5 △ B1049.4	CCAFS, SLC-40	Starlink 2 v1.0 (60 satellites)	15,600 kg (34,400 lb)[D]	LEO	SpaceX	Success	Success (drone ship)
	Third large batch and second operational flight of Starlink constellation. One of the 60 satellites included a test coating to make the satellite less reflective, and thus less likely to interfere with ground-based astronomical observations.[493]								
79	19 January 2020, 15:30[494]	F9 B5 △ B1046.4	KSC, LC-39A	Crew Dragon in-flight abort test[495] (Dragon C205.1)	12,050 kg (26,570 lb)	Sub-orbital[496]	NASA (CTS)[497]	Success	No attempt
	An atmospheric test of the Dragon 2 abort system after Max Q. The capsule fired its SuperDraco engines, reached an apogee of 40 km (25 mi), deployed parachutes after reentry, and splashed down in the ocean 31 km (19 mi) downrange from the launch site. The test was previously slated to be accomplished with the Crew Dragon Demo-1 capsule,[498] but that test article exploded during a ground test of SuperDraco engines on 20 April 2019.[4][5] The abort test used the capsule originally intended for the first crewed flight.[499] As expected, the booster was destroyed by aerodynamic forces after the capsule aborted.[500] First flight of a Falcon 9 with only one functional stage — the second stage had a mass simulator in place of its engine.								
80	29 January 2020, 14:07[501]	F9 B5 △ B1051.3	CCAFS, SLC-40	Starlink 3 v1.0 (60 satellites)	15,600 kg (34,400 lb)[D]	LEO	SpaceX	Success	Success (drone ship)
	Third operational and fourth large batch of Starlink satellites, deployed in a circular 290 km (180 mi) orbit. One of the fairing halves was caught, while the other was fished out of the ocean.[502]								
81	17 February 2020, 15:05[503]	F9 B5 △ B1056.4	CCAFS, SLC-40	Starlink 4 v1.0 (60 satellites)	15,600 kg (34,400 lb)[D]	LEO	SpaceX	Success	Failure (drone ship)
	Fourth operational and fifth large batch of Starlink satellites. Used a new flight profile which deployed into a 212 km × 386 km (132 mi × 240 mi) elliptical orbit instead of launching into a circular orbit and firing the second stage engine twice. The first stage booster failed to land on the drone ship[504] due to incorrect wind data.[505] This was the first time a flight proven booster failed to land.								
82	7 March 2020, 04:50[506]	F9 B5 △ B1059.2	CCAFS, SLC-40	SpaceX CRS-20 (Dragon C112.3 △)	1,977 kg (4,359 lb)[507]	LEO (ISS)	NASA (CRS)	Success	Success (ground pad)
	Last launch of phase 1 of the CRS contract. Carries *Bartolomeo*, an ESA platform for hosting external payloads onto ISS.[508] Originally scheduled to launch on 2 March 2020, the launch date was pushed back due to a second stage engine failure. SpaceX decided to swap out the second stage instead of replacing the faulty part.[509] It was SpaceX's 50th successful landing of a first stage booster, the third flight of the Dragon C112 and the last launch of the cargo Dragon spacecraft.								
83	18 March 2020, 12:16[510]	F9 B5 △ B1048.5	KSC, LC-39A	Starlink 5 v1.0 (60 satellites)	15,600 kg (34,400 lb)[D]	LEO	SpaceX	Success	Failure (drone ship)
	Fifth operational launch of Starlink satellites. It was the first time a first stage booster flew for a fifth time and the second time the fairings were reused (Starlink flight in May 2019).[511] Towards the end of the first stage burn, the booster suffered premature shut down of an engine, the first of a Merlin 1D variant and first since the CRS-1 mission in October 2012. However, the payload still reached the targeted orbit.[512] This was the second Starlink launch booster landing failure in a row, later revealed to be caused by residual cleaning fluid trapped inside a sensor.[513]								
84	22 April 2020, 19:30[514]	F9 B5 △ B1051.4	KSC, LC-39A	Starlink 6 v1.0 (60 satellites)	15,600 kg (34,400 lb)[D]	LEO	SpaceX	Success	Success (drone ship)



Requests & Beautiful soup module

Use request and BeautifulSoup html parser to extract the tables on the wiki page.

Pandas

Create a Data frame from the soup previously gathered.

Data Wrangling

Assign each row on the soup to the stablished format.

Save to CSV

Data Collection

SpaceX API

+ Coursera_IBM_DataScience/01_IBMCapstone_Spacex-data-collection-api.ipynb at main · MarcoAlegria94/Coursera_IBM_DataScience (github.com)

Now let's start requesting rocket launch data from SpaceX API with the following URL:

```
spacex_url="https://api.spacexdata.com/v4/launches/past"
```

```
response = requests.get(spacex_url)
```

Wikipedia web scraping

+ Coursera_IBM_DataScience/02_IBMCapstone_Spacex-data-webScraping.ipynb at main · MarcoAlegria94/Coursera_IBM_DataScience (github.com)

First, let's perform an HTTP GET method to request the Falcon9 Launch HTML page, as an HTTP response.

```
# use requests.get() method with the provided static_url
# assign the response to a object
response = requests.get(static_url)
#print(response.text)
```

Create a `BeautifulSoup` object from the HTML `response`

```
# Use BeautifulSoup() to create a BeautifulSoup object from a response text content
soup = BeautifulSoup(response.content, "html.parser")
```

Print the page title to verify if the `BeautifulSoup` object was created properly

```
# Use soup.title attribute
soup.title
```

```
<title>List of Falcon 9 and Falcon Heavy launches - Wikipedia</title>
```

Data wrangling

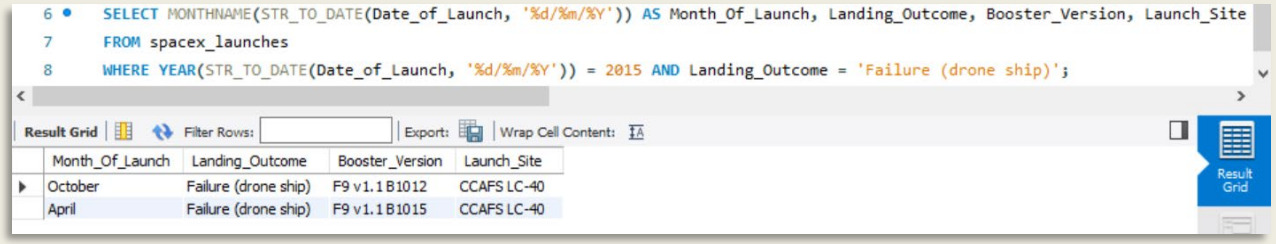
- The data wrangling in all of this journey is crucial in order to assign the correct structure and labeling for the later EDA, DataViz and Predictive modeling. Thus, creating the correct structure for the data in CSV format is what Data Wrangling is.

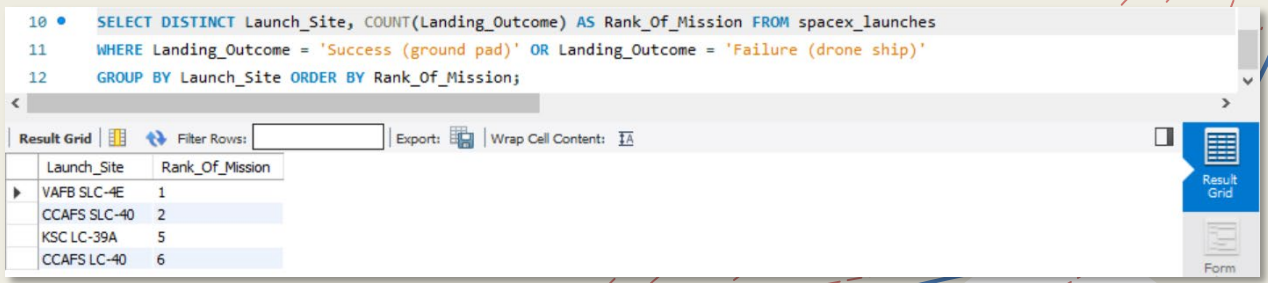
- The steps for the correct structure are the following:
  - Identify percentage of missing values for each attribute.
  - Identify the different Launching sites.
  - Finally, identify the outcomes of the missions and creating a column that classifies the success or fail of the mission.

EDA with SQL

- For the EDA it was required to setup a local SQL server using MySQL and the mysqlconnector for integrating Python to be able to send queries into the local server.

- The previously made CSV from the Data Wrangling was imported into the MySQL server, from there the EDA was made on the Python environment.

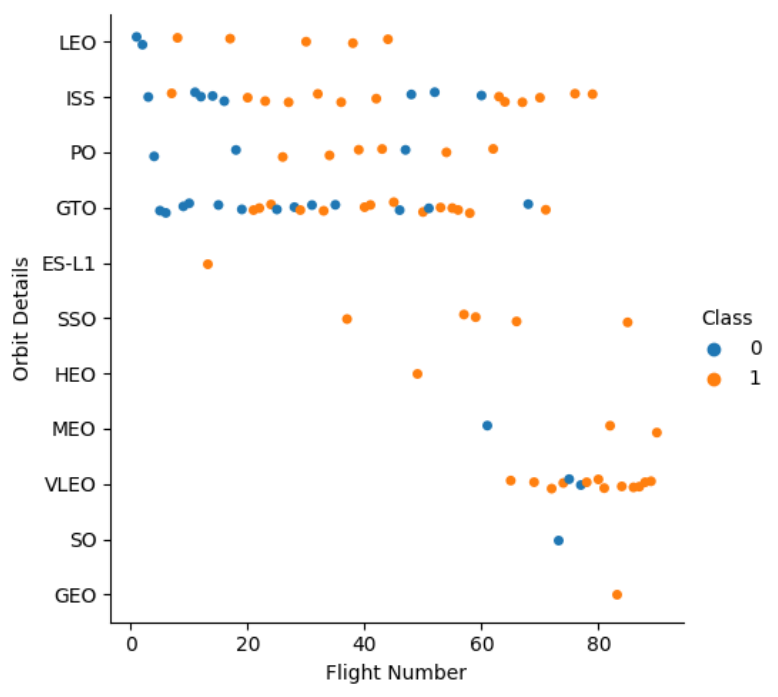


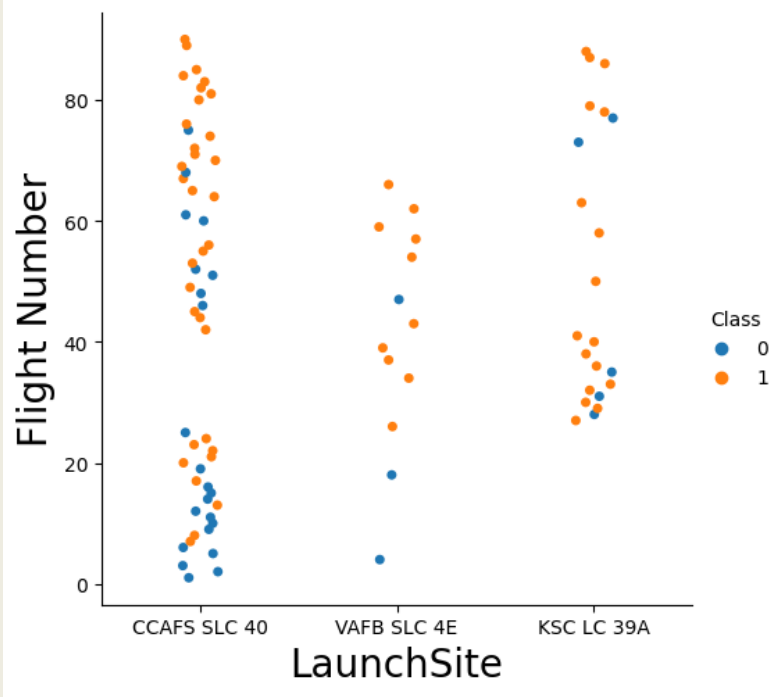


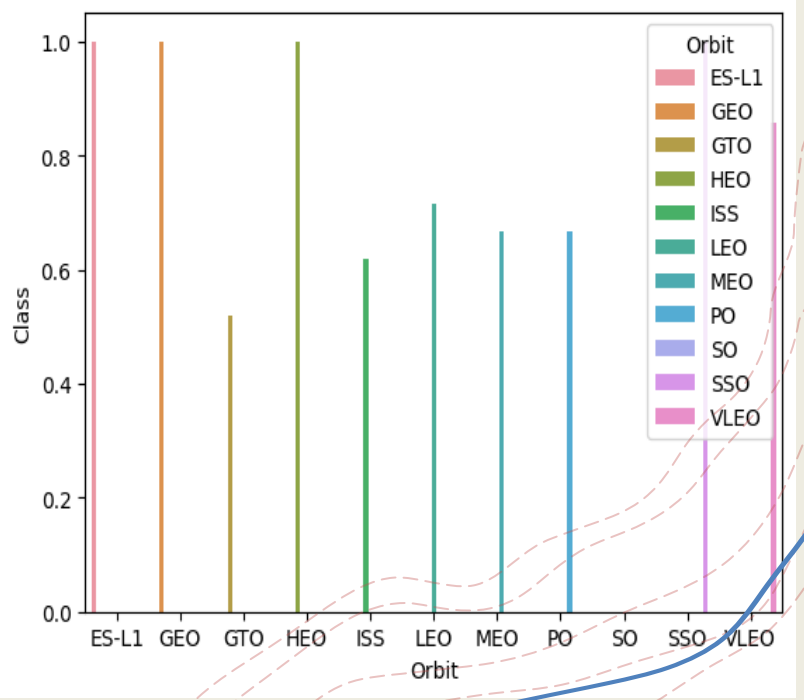
EDA with Data Visualization

- Getting into the fun part of Data Analysis, EDA with Matplotlib. Here we start to find patterns and correlations between the parameters gathered in our data base. The data visualization gives us a better insight into the data, and we can find useful information for our predictive model.
- Different information can be inferred by the visual representation.

- From the data visualized we can find that:
  - As expected, the success rate from the flights increased by the number of trials that were made. In the beginning the first 20 flights show a low success rate, however, on the latest trials we can see that must of the flights were successful, even when aiming for higher orbit zones.
  - The most used launching site is CCAFS however the must successful launching site can be considered KSC as it has more successful launches in comparison to the failed.
  - The most successful orbits are ES-L1, GEO and HEO.





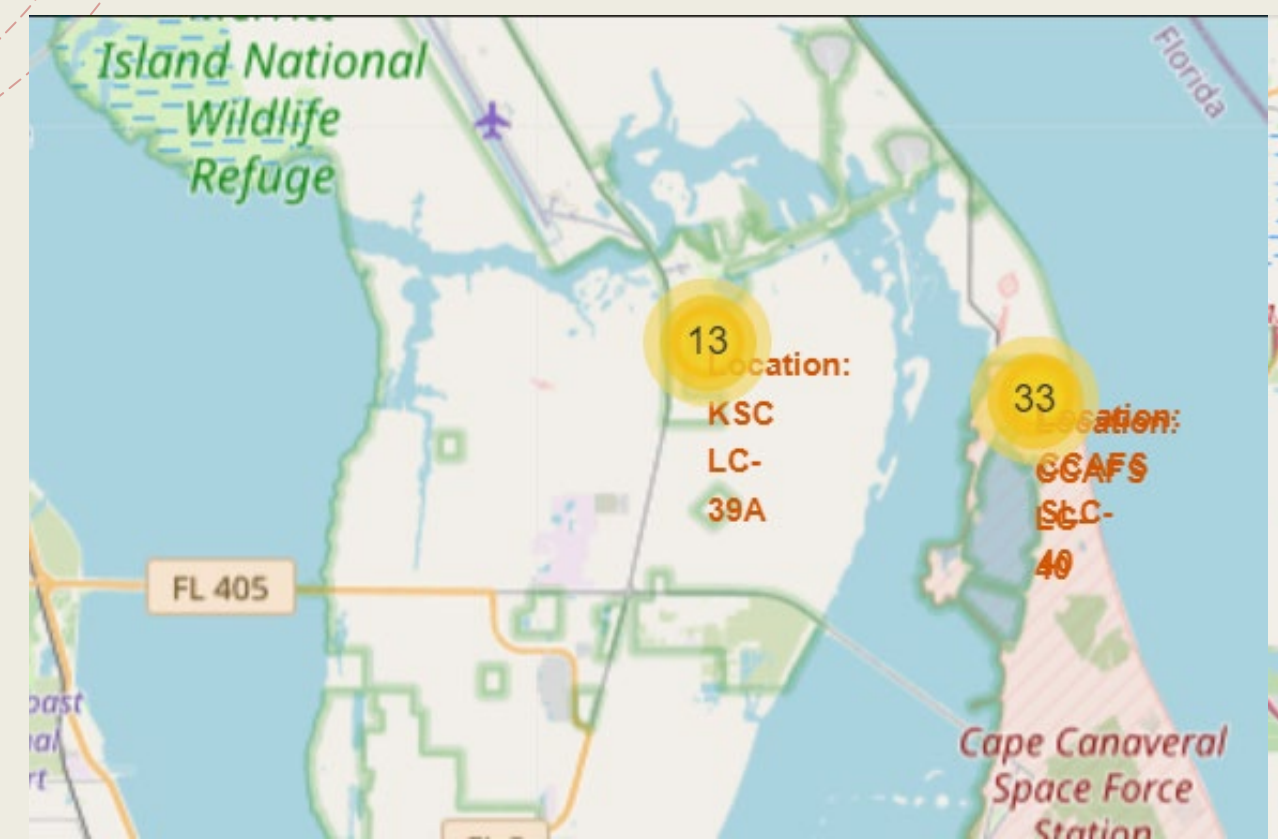


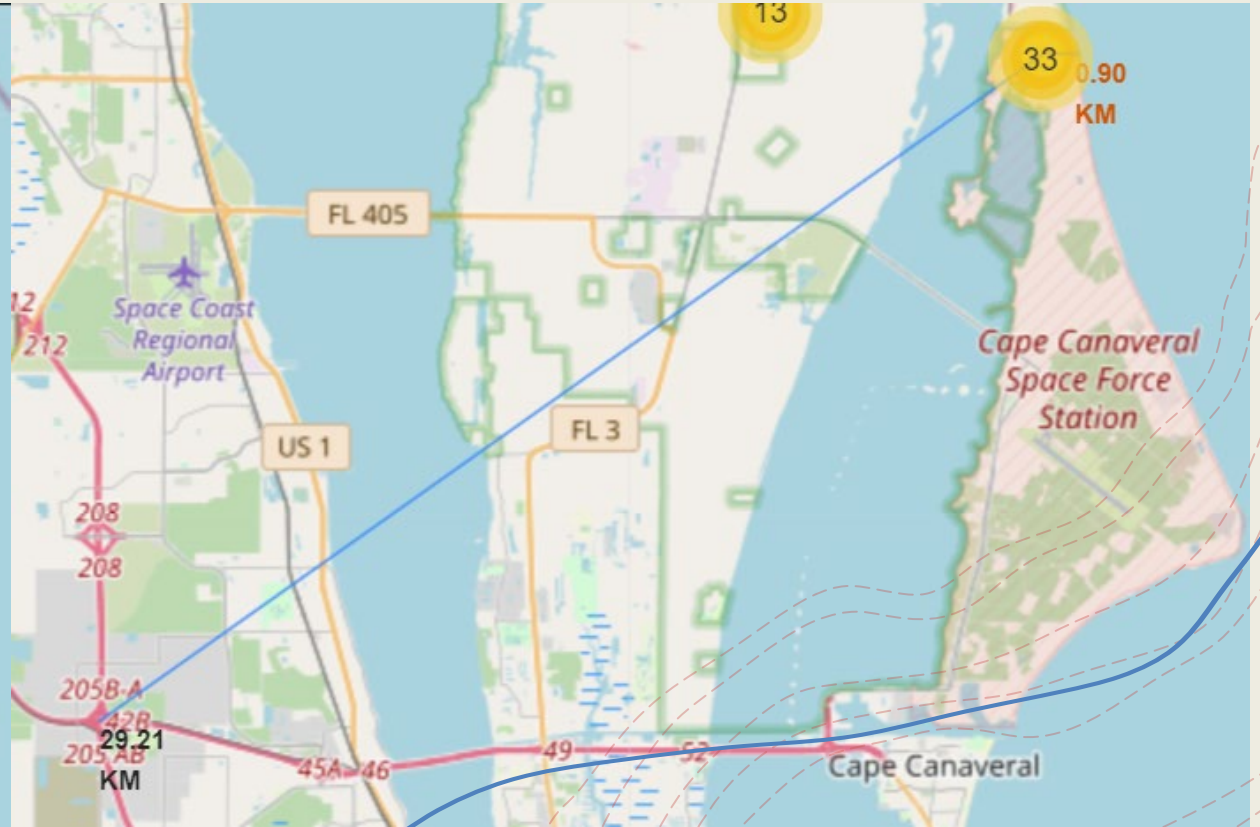
Let's do things more interactive...

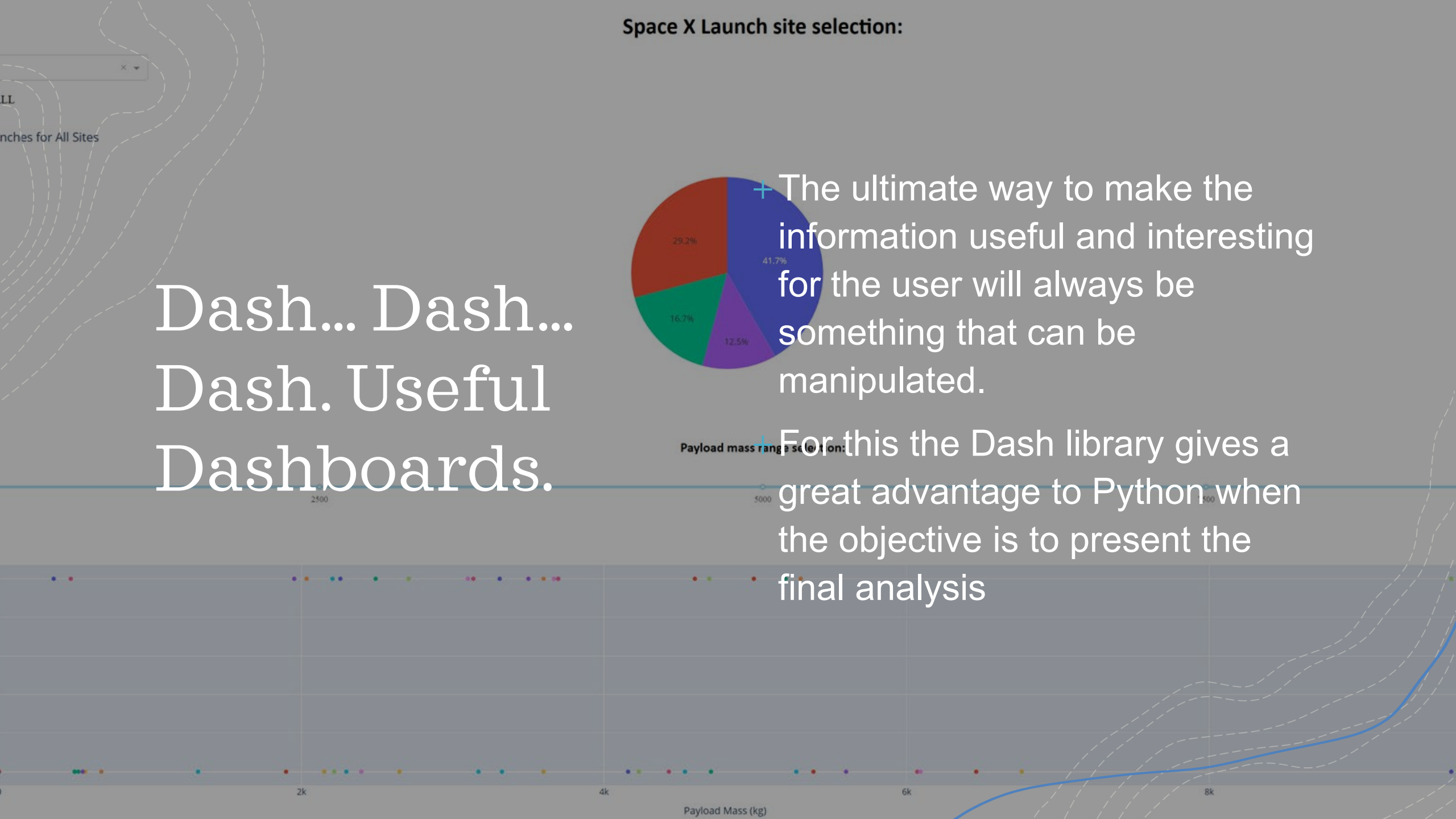
- Thanks to the python versatility we can use the Folium interactive maps to make interesting insights for the data
- Beginning with pop-up labels for the must used launching sites, gathering the information from the latitude and longitude of each one, we can present them interactively in a map for the user.

Data insight from folium map

- From there we can start making the maps more useful, understanding the number of missions made on each launching site.
- Each launching site represents different advantages thanks to its location in the map, for example, we can find the distance from the nearest city.



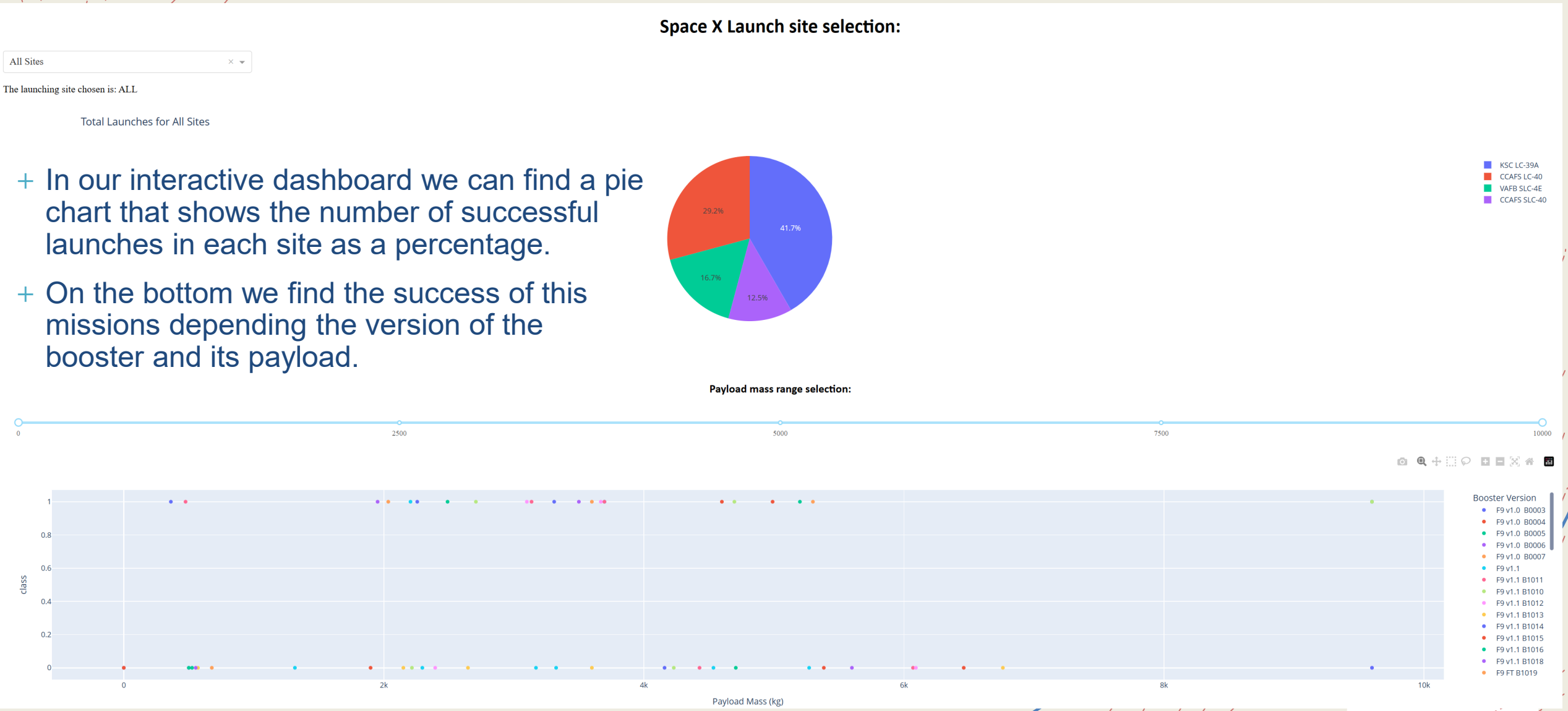




Dash… Dash… Dash. Useful Dashboards.

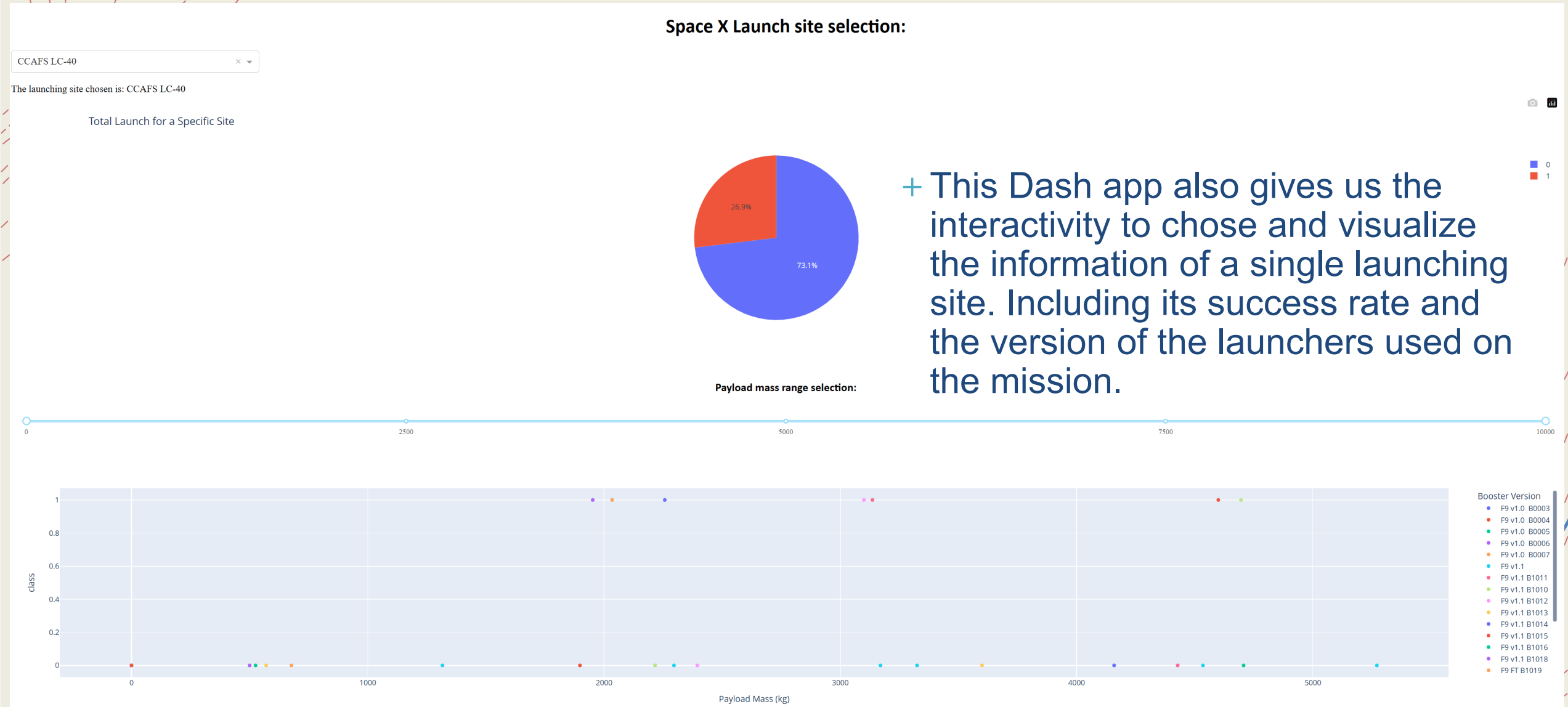
- The ultimate way to make the information useful and interesting for the user will always be something that can be manipulated.
- For this the Dash library gives a great advantage to Python when the objective is to present the final analysis

Complete data visualized



- In our interactive dashboard we can find a pie chart that shows the number of successful launches in each site as a percentage.
- On the bottom we find the success of this missions depending the version of the booster and its payload.

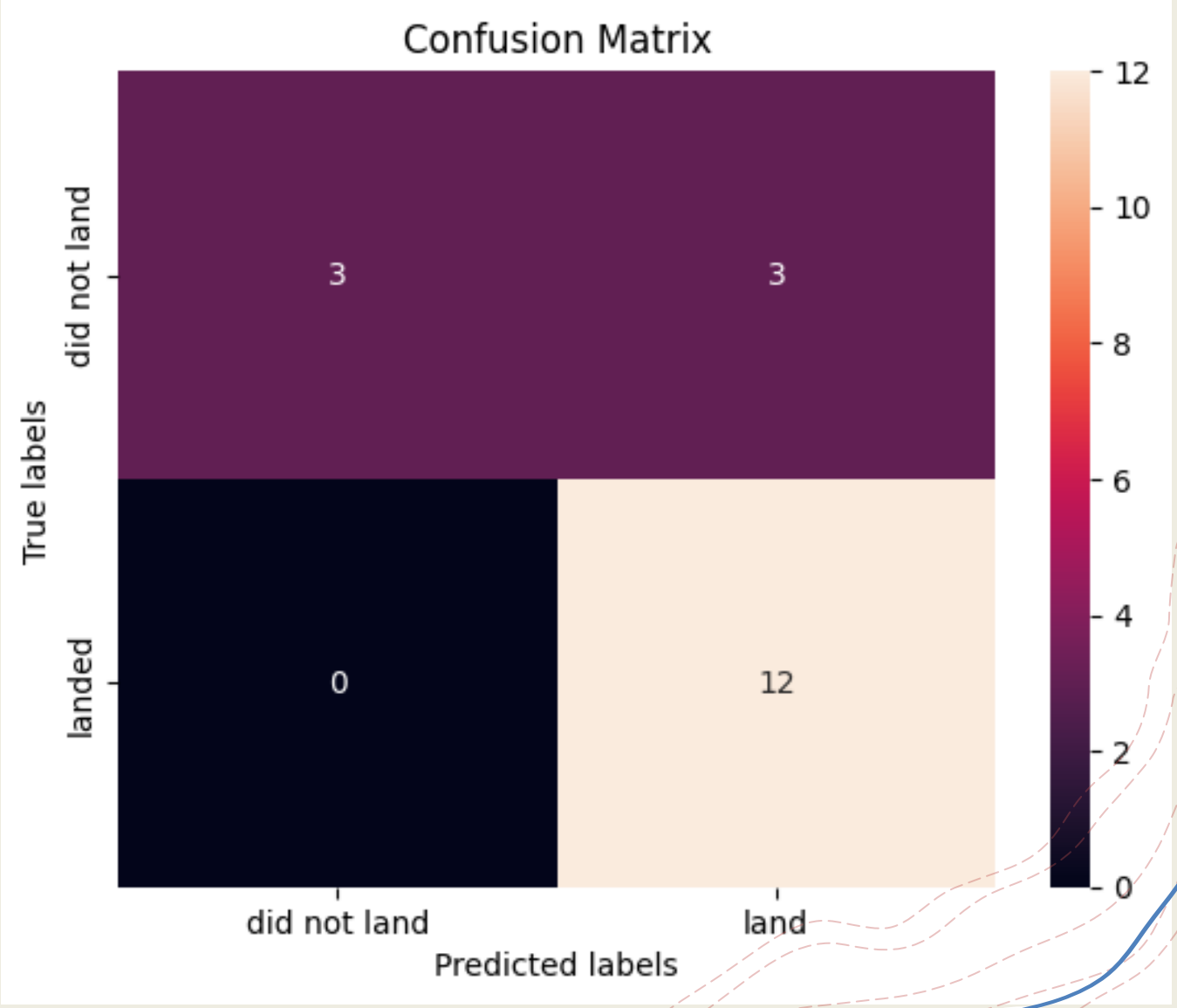
Specific data visualized



+ This Dash app also gives us the interactivity to chose and visualize the information of a single launching site. Including its success rate and the version of the launchers used on the mission.

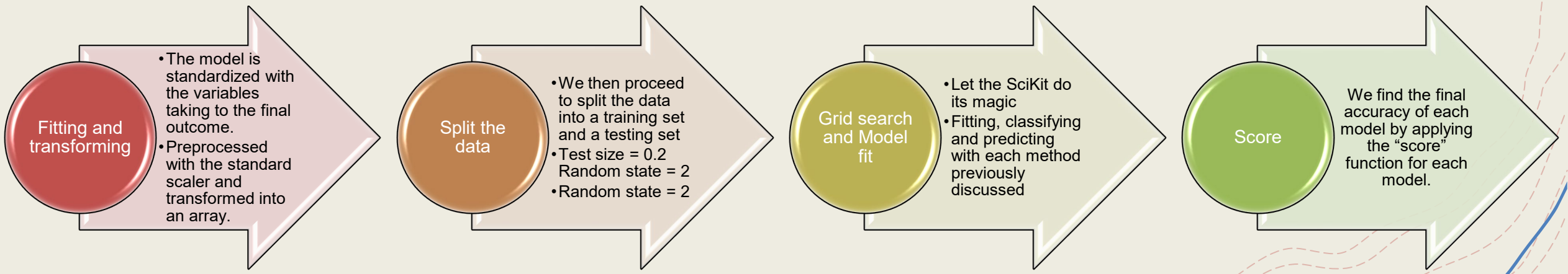
Reaching the end of the journey.

- The final objective of all of this process of Data science was to create a predictive model that can give the success of the mission.
- We begin with the idea of using the most used predictive classification models like:
  - Logistic Regression (LogReg)
  - Support Vector Machine (SVM)
  - Decision tree
  - K nearest neighbors (KNN)



Training the model

- Thanks to the Sci-Kit library for python, the process of training a model with different methods, can be done more efficiently.
- Basically, all the models consider the same steps:



Results.

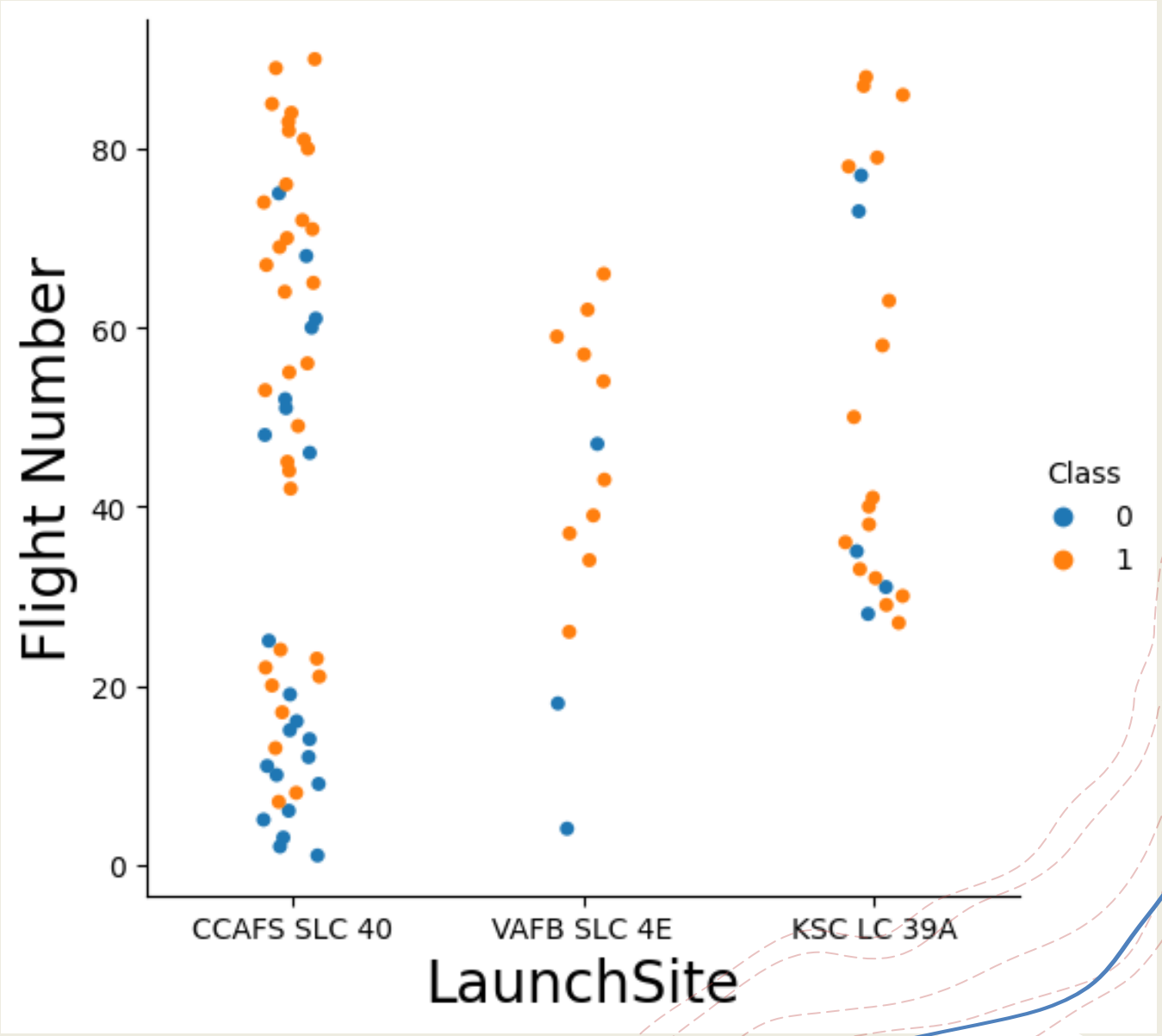
- The best option for the next launch would be better to be done from the Cape Canaveral station, aiming to the geosynchronous orbit GTO at 35,786 kilometers.
- For predicting the best outcome, the Decision Tree method can give the best result, since the accuracy of the model is up to 86%
- Finally, we can assume that if we stick to the common launching site and orbit aimed, the mission can become successful.

Insights drawn from EDA

Section 2

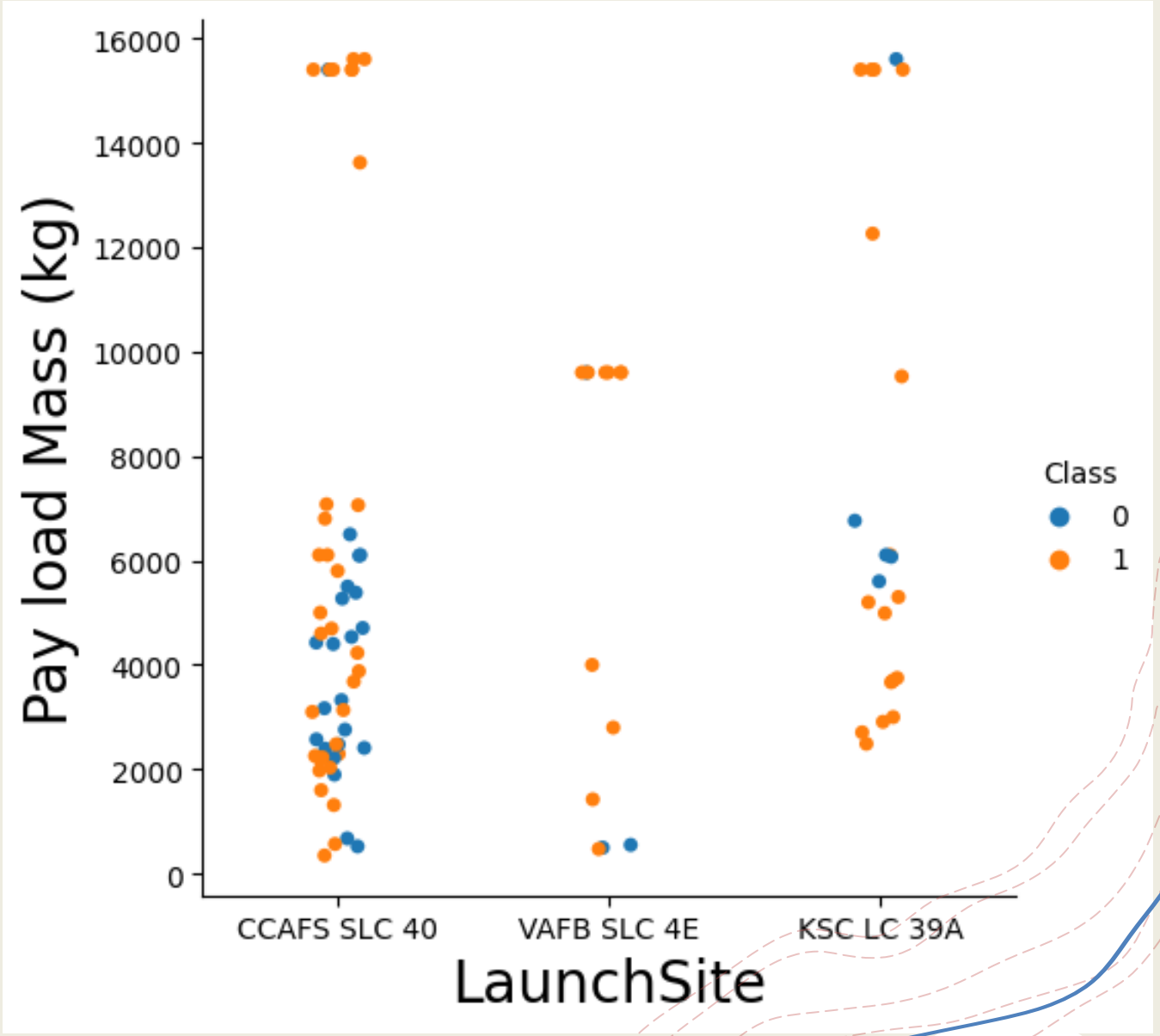
Flight Number vs. Launch Site

- Most of the launches have been made from CCAFS.
- In the beginning of the missions most were unsuccessful, however, as the number of tries increased, the success rate became bigger
- KSC has the most successful launch history.



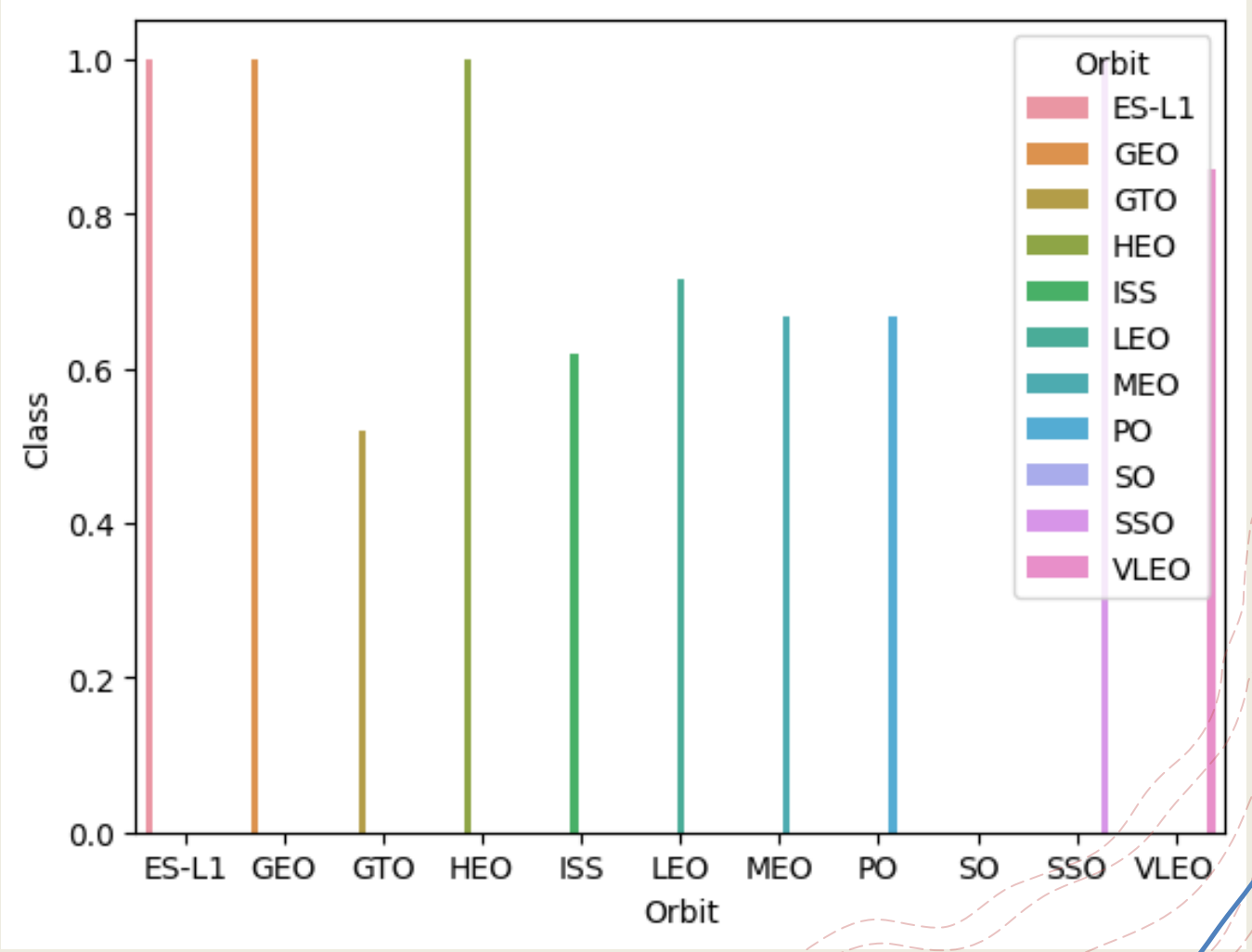
Payload vs. Launch Site

- The more mass the missions are loaded with, indicate that the success rate becomes higher.
- Around 10 to 14 tons of payload mass show a 100% success rate.
- Above this threshold, the mission should be more cautiously evaluated.



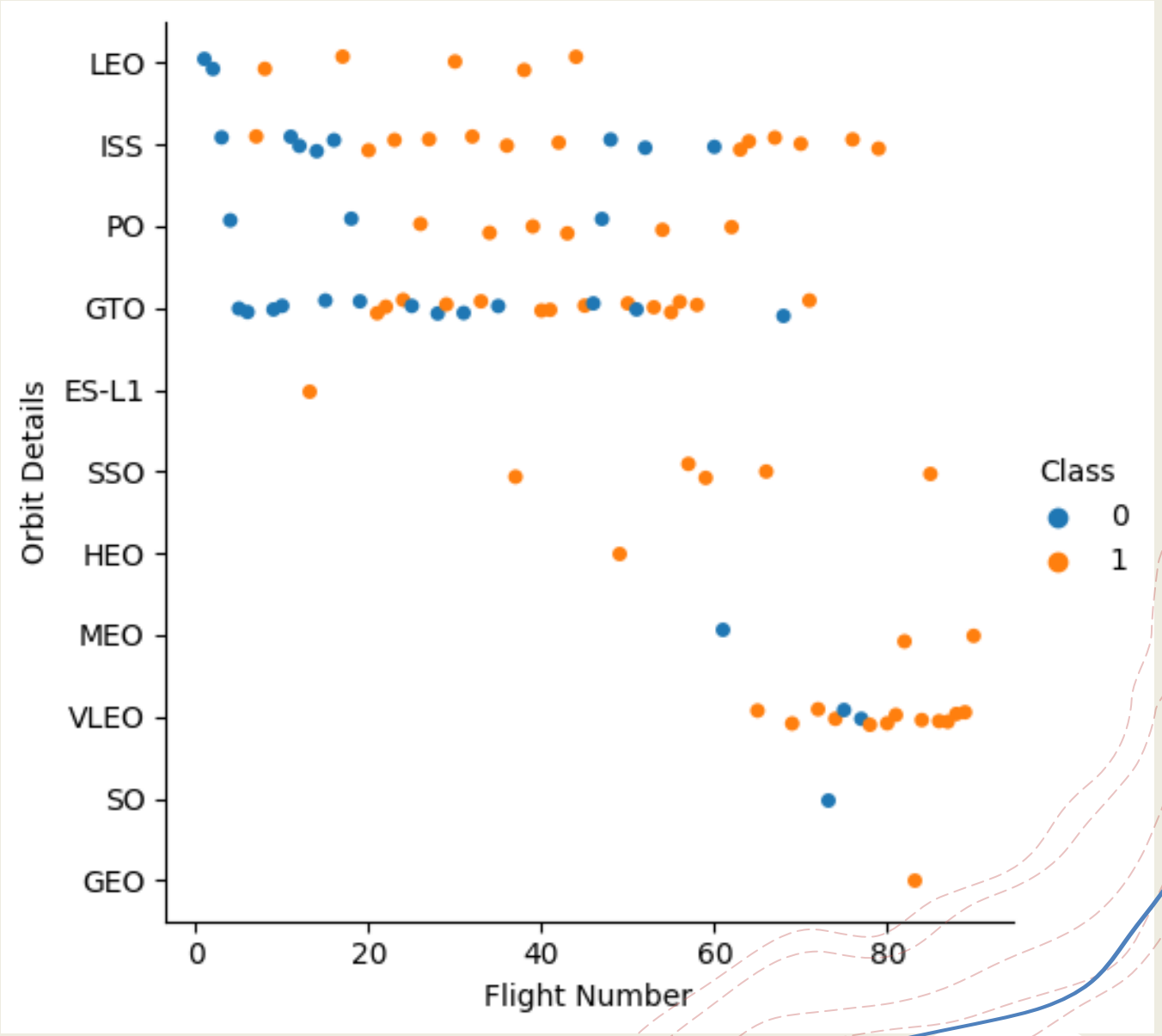
Success Rate vs. Orbit Type

- The most successful orbit missions are:
  - ES-L1
  - GEO
  - HEO
- SO has a 0% success rate.



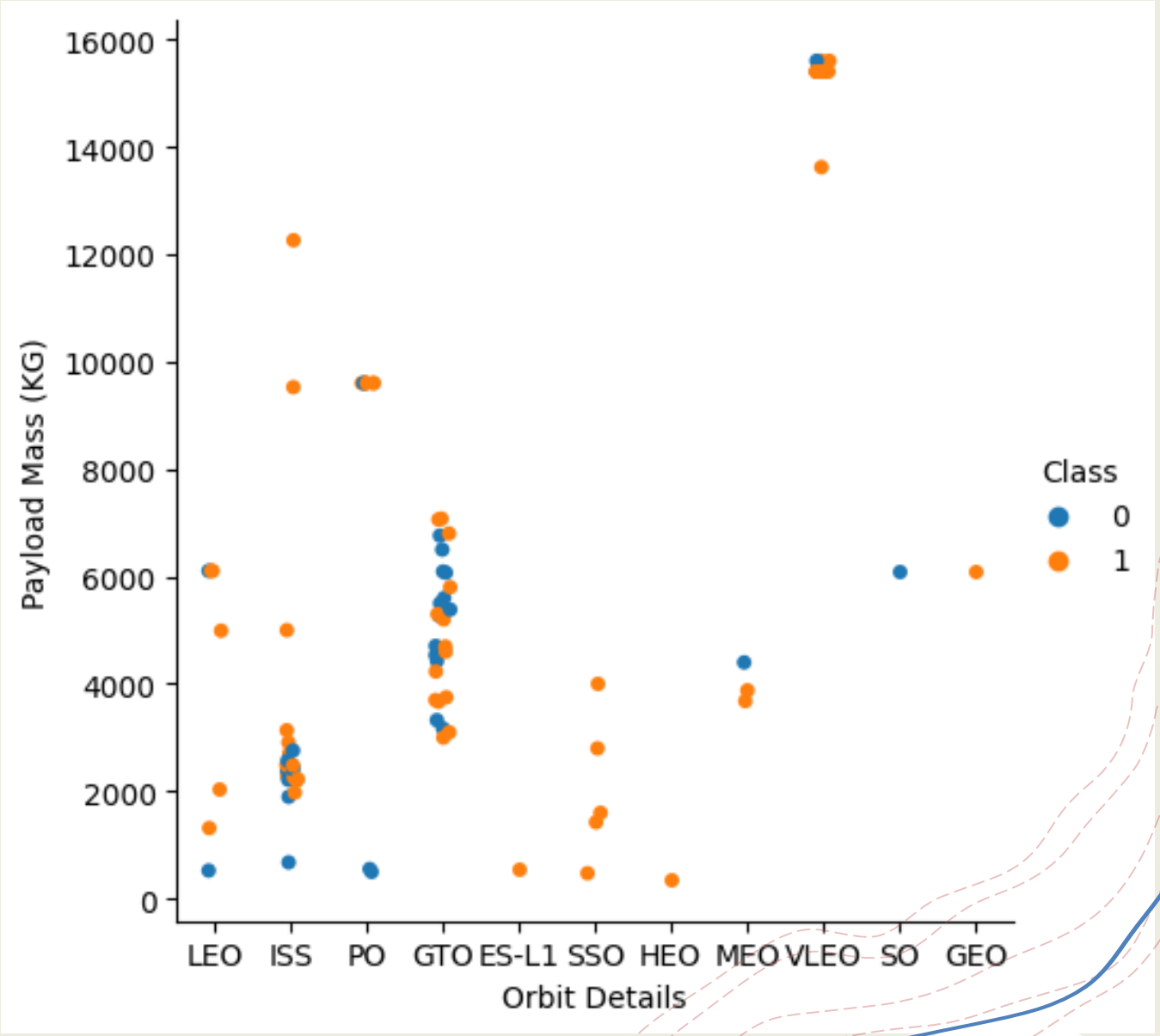
Flight Number vs. Orbit Type

- The low earth orbit, gives a high percent of success rate.
- The ES-L1, SSO and HEO had always have a successful mission rate.
- There has been only one mission to the SO which was unsuccesful.



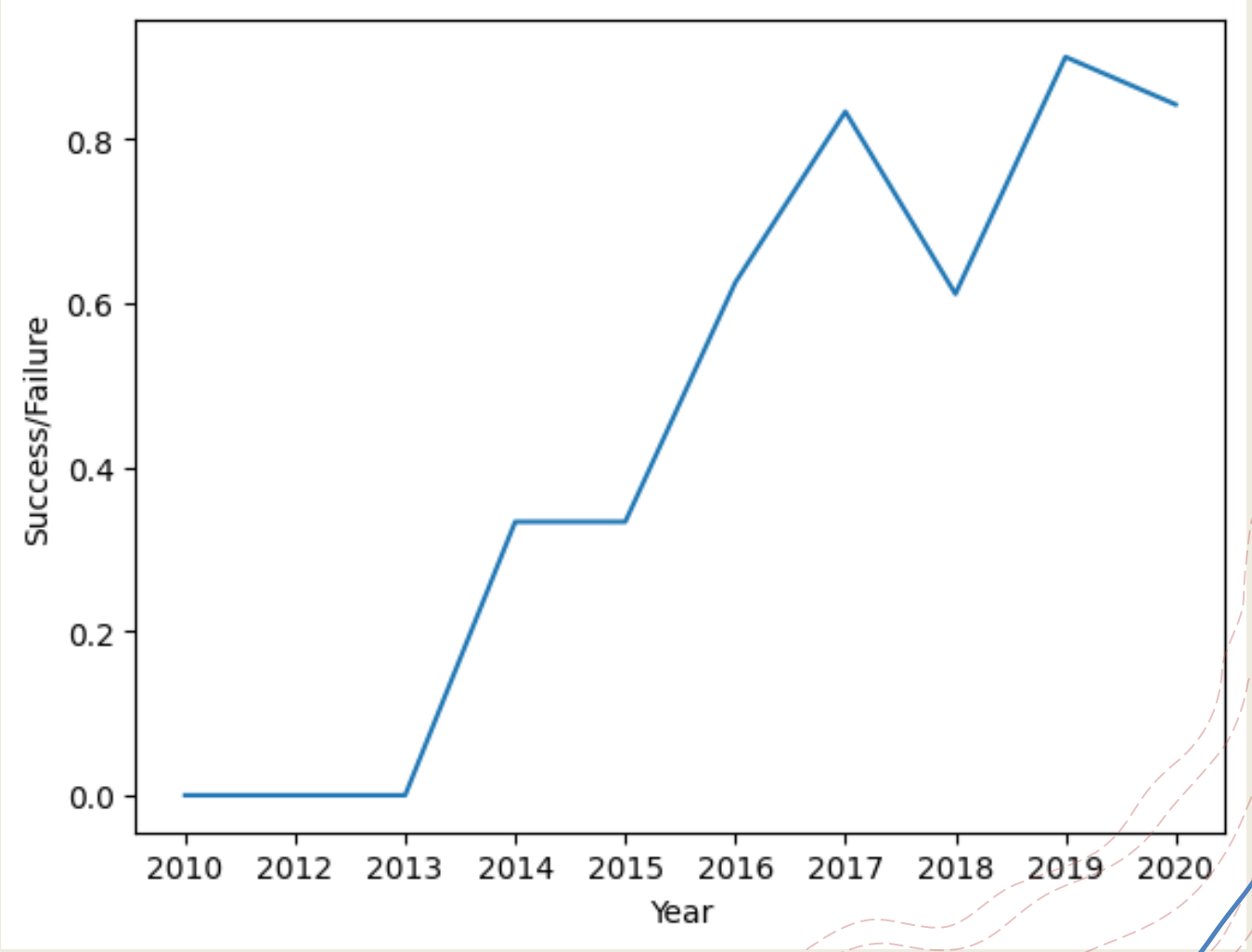
Payload vs. Orbit Type

- The highest payload is always sent to the VLEO.
- The ISS missions usually carry a low payload mass.



Launch Success Yearly Trend

- The peak of the success rate has been in 2019.



All Launch Site Names

```
SELECT DISTINCT Launch_site FROM spacex_launches;
```

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

Launch Site Names Begin with 'CCA'

```
SELECT DISTINCT Launch_site FROM spacex_launches
WHERE Launch_Site LIKE 'CCA%';
```

Launch_site
CCAFS LC-40
CCAFS SLC-40

Total Payload Mass

```
SELECT SUM(Payload_mass_kg) AS 'Total Payload from NASA'
FROM spacex_launches WHERE Customer LIKE '%NASA%';
```

	Total Payload from NASA
▶	107010

Average Payload Mass by F9 v1.1

```
SELECT AVG(Payload_mass_kg) AS 'Average Payload by F9 v1.1'
FROM spacex_launches WHERE Booster_Version LIKE '%v1.1%';
```

	Average Payload by F9 v1.1
▶	2534.6667

First Successful Ground Landing Date

```
SELECT Date_of_Launch, Landing_Outcome
FROM spacex_launches WHERE Mission_Outcome = 'Success' AND Landing
ORDER BY STR_TO_DATE(Date_of_Launch, '%d/%m/%Y');
```

	Date_of_Launch	Landing_Outcome
▶	22/12/2015	Success (ground pad)
	18/07/2016	Success (ground pad)
	05/01/2017	Success (ground pad)

Successful Drone Ship Landing with Payload between 4000 and 6000

```
SELECT Booster_Version FROM spacex_launches
WHERE Landing_Outcome = 'Success (drone ship)'
AND Payload_mass_kg BETWEEN 4000 AND 6000;
```

Booster_Version
F9 FT B1022
F9 FT B1026
F9 FT B1021.2
F9 FT B1031.2

Total Number of Successful and Failure Mission Outcomes

```
SELECT Mission_Outcome, COUNT(Mission_Outcome) AS 'Total'
FROM spacex_launches GROUP BY Mission_Outcome;
```

Mission_Outcome	Total
Success	98
Failure (in flight)	1
Success (payload status undear)	1
Success	1

Boosters Carried Maximum Payload

```
SELECT Booster_Version FROM spacex_launches
WHERE (SELECT MAX(Payload_mass_kg) FROM spacex_launches);
```

Booster_Version
F9 v1.0 B0003
F9 v1.0 B0004
F9 v1.0 B0005
F9 v1.0 B0006
F9 v1.0 B0007
F9 v1.1 B1003

2015 Launch Records

```
SELECT MONTHNAME(STR_TO_DATE(Date_of_Launch, '%d/%m/%Y'))
AS Month_Of_Launch, Landing_Outcome, Booster_Version, Launch_Site
FROM spacex_launches
WHERE YEAR(STR_TO_DATE(Date_of_Launch, '%d/%m/%Y')) = 2015
AND Landing_Outcome = 'Failure (drone ship)';
```

Month_Of_Launch	Landing_Outcome	Booster_Version	Launch_Site
October	Failure (drone ship)	F9 v1.1 B1012	CCAFS LC-40
April	Failure (drone ship)	F9 v1.1 B1015	CCAFS LC-40

Rank Landing Outcomes Between 2010-06-04 and 2017-03-20

```
SELECT DISTINCT Launch_Site, COUNT(Landing_Outcome)
AS Rank_Of_Mission FROM spacex_launches
WHERE Landing_Outcome = 'Success (ground pad)'
OR Landing_Outcome = 'Failure (drone ship)'
GROUP BY Launch_Site ORDER BY Rank_Of_Mission;
```

	Launch_Site	Rank_Of_Mission
▶	VAFB SLC-4E	1
	CCAFS SLC-40	2
	KSC LC-39A	5
	CCAFS LC-40	6

Launch sites proximities analysis

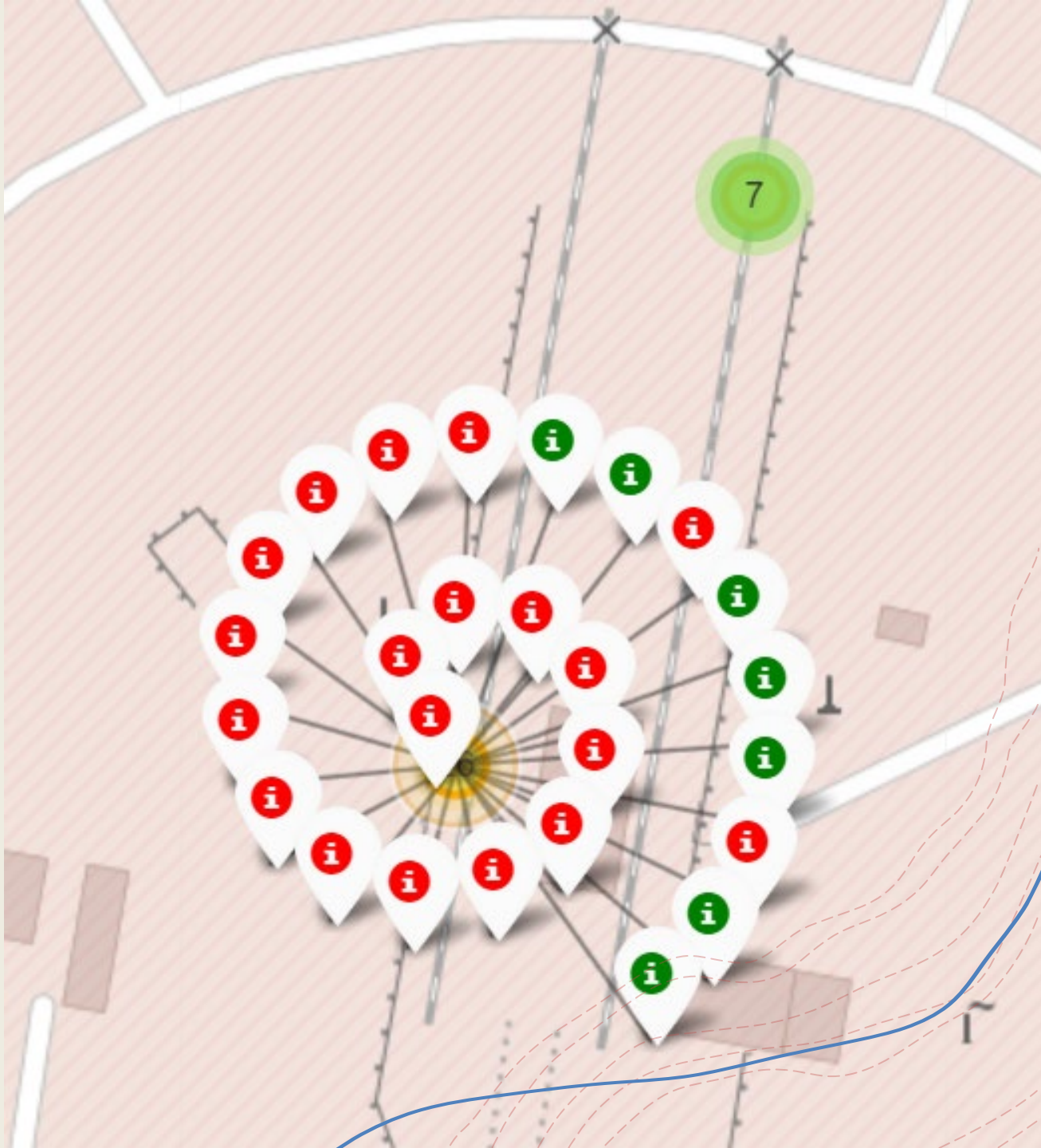
Section 3

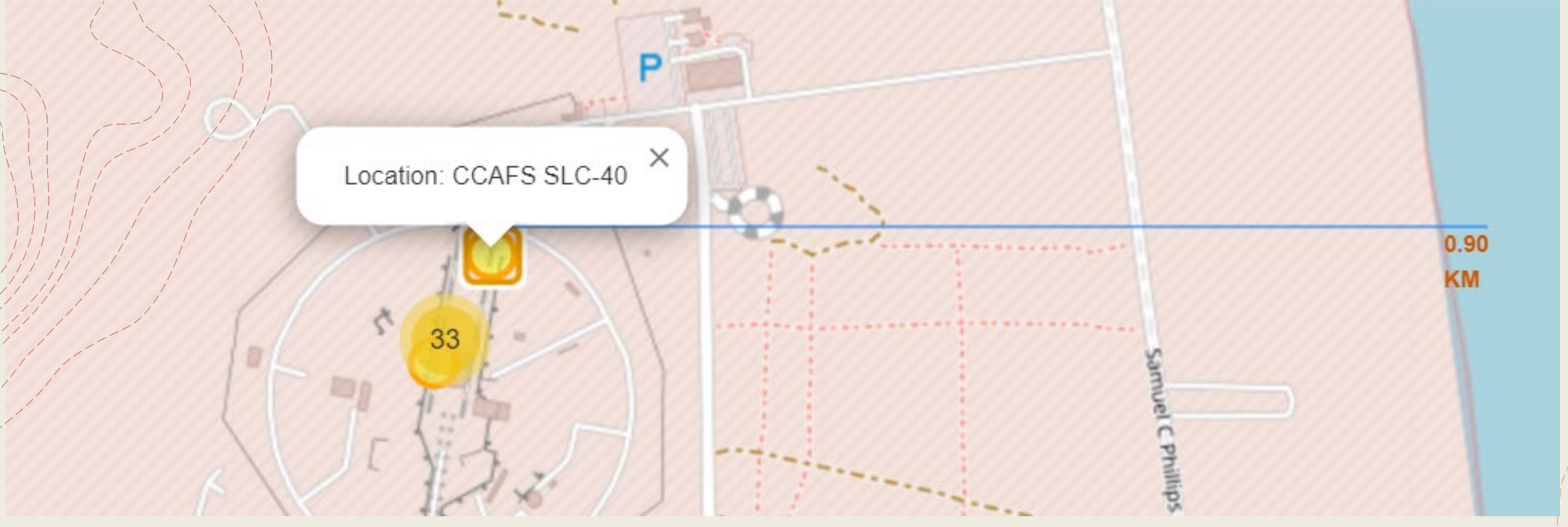
Where are my launching sites?

- Here we can find that the launching sites are primarily on the east coast, we can find some in the west coast as well.
- Must of the missions are sent from the east coast.

Am I successful enough?

- Depending on the site of launch we can find the success of the missions according to the color code.
- Green = Success
- Red = Failure





Where am I?

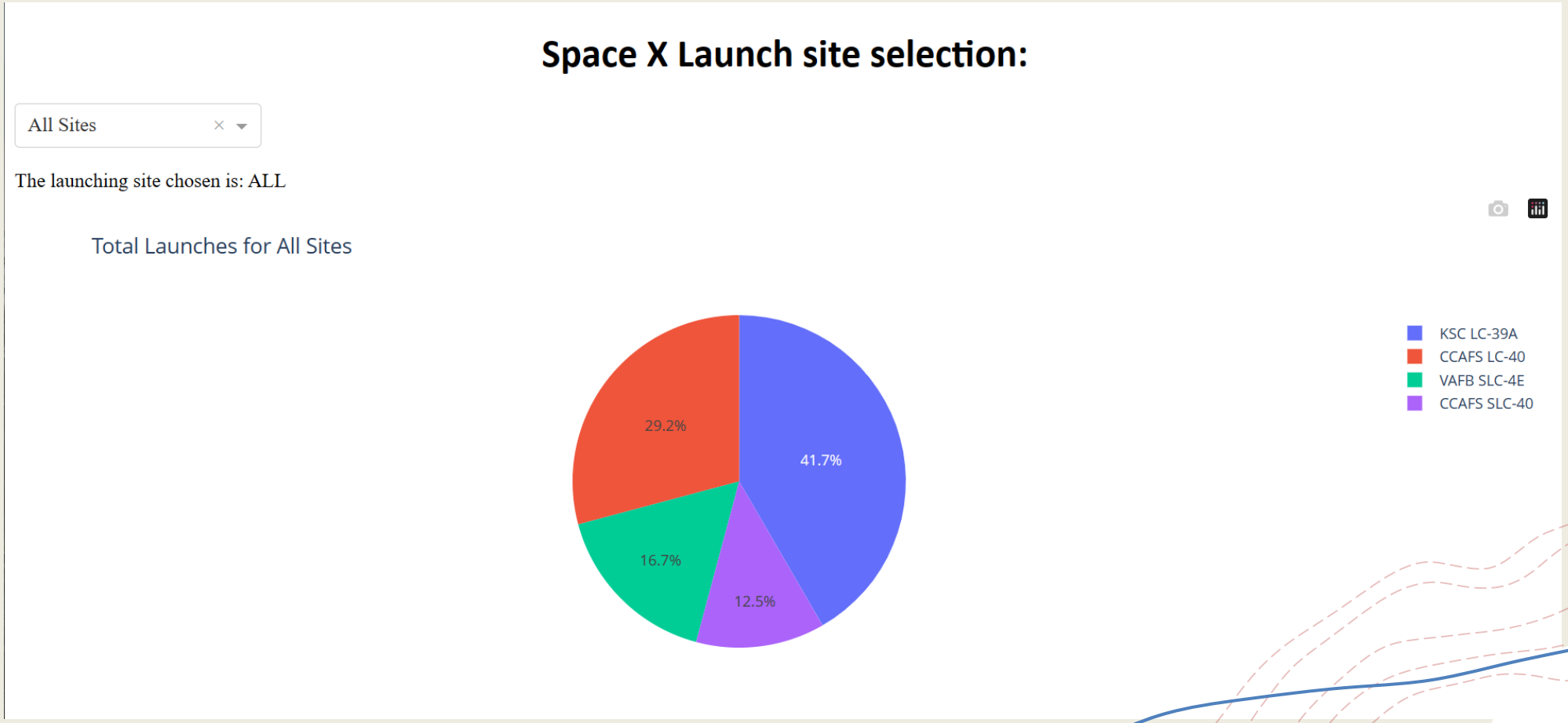
+ Opening each pop-up, we can find that each launching site is located near a coast. And we can see the distance from each point.

Dashboard with Plotly Dash

Section 4

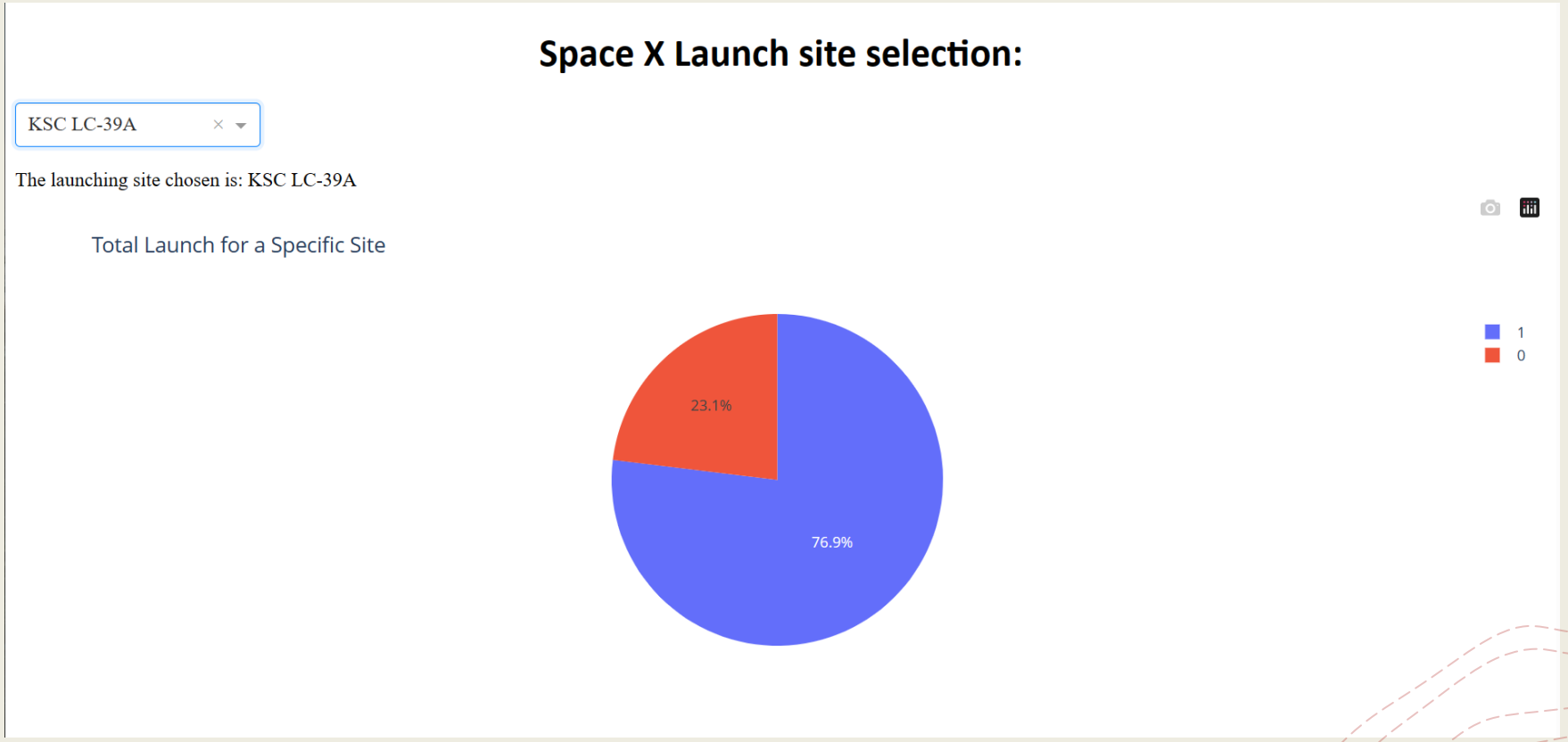
The whole picture:

- The must successful launching site is the KSC LC-39A.
- Followed by the CCAFS SLC-40



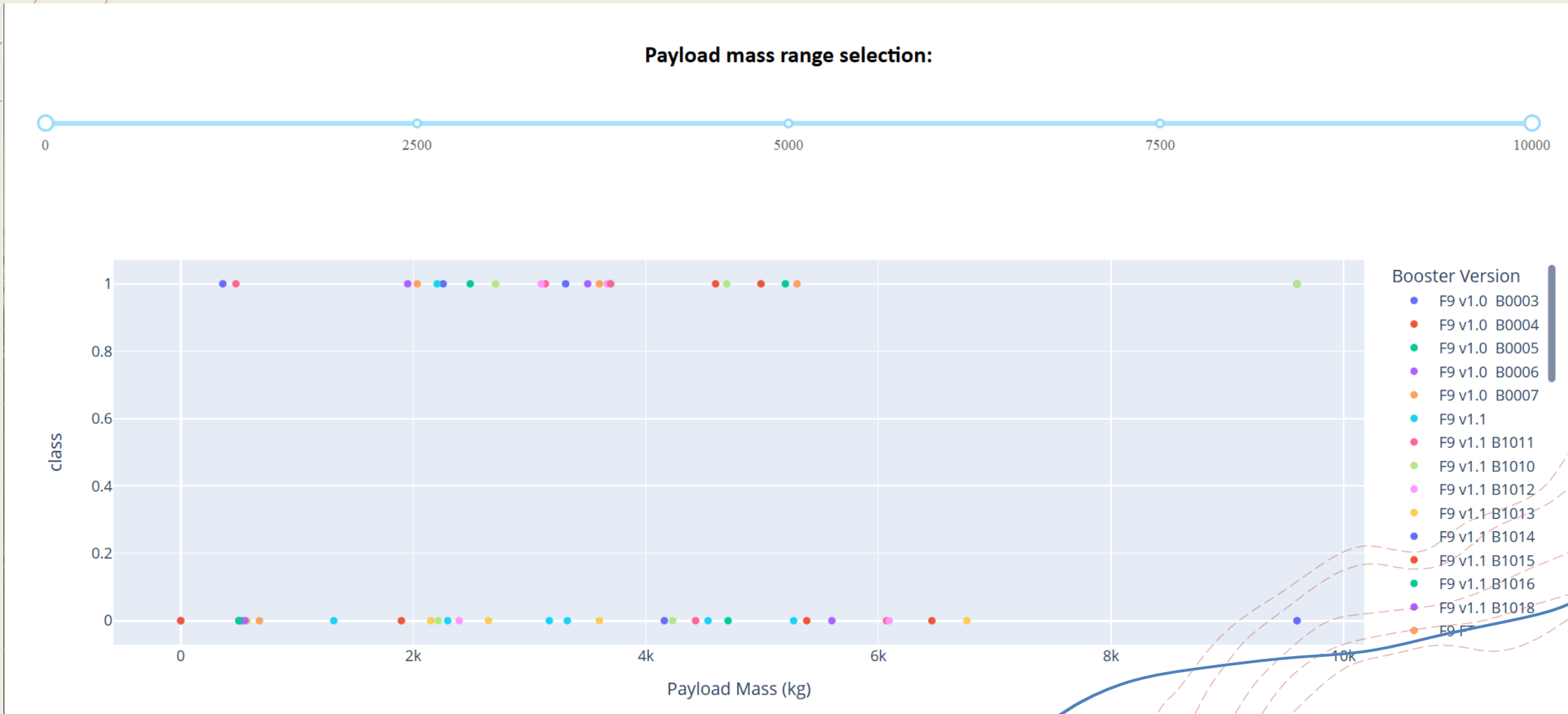
KSC LC-39A the must successful

We can see that this launching site has a 76.9% success rate on its missions.



The payload matters?

+ It seems that the average payload will give a higher success rate.
+ Around 2000 to 5000 Kg

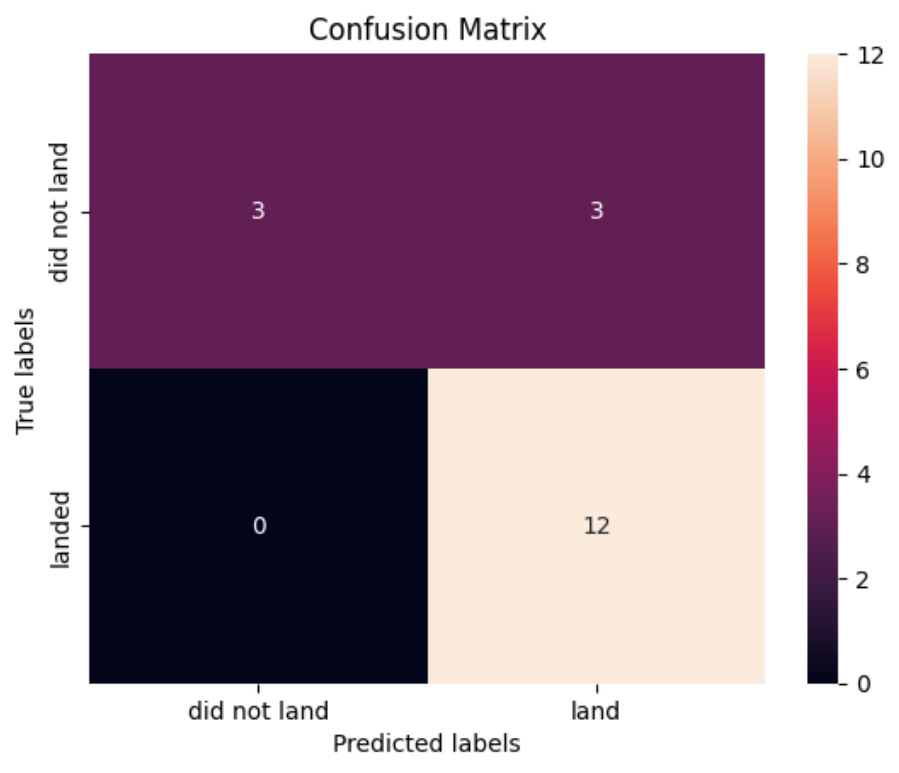


Predictive Analysis (Classification)

Section 5

The final verdict.

Confusion matrix.



Accuracy results.

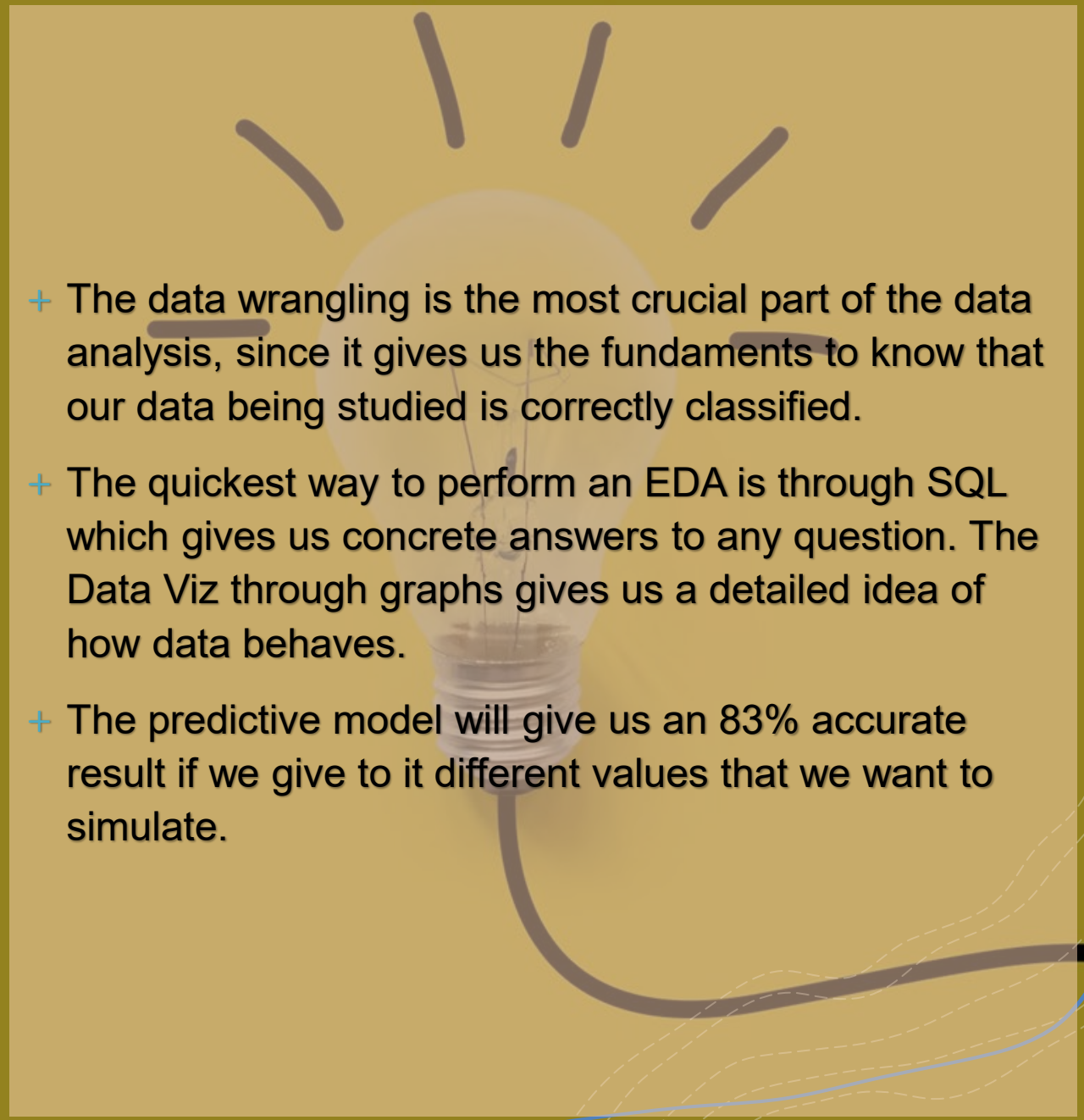
Find the method performs best:

```
print('Accuracy for Logistics Regression method:', logreg_cv.score(X_test, Y_test))
print( 'Accuracy for Support Vector Machine method:', svm_cv.score(X_test, Y_test))
print('Accuracy for Decision tree method:', tree_cv.score(X_test, Y_test))
print('Accuracy for K nearsdt neighbors method:', knn_cv.score(X_test, Y_test))
```

✓ 0.0s

```
Accuracy for Logistics Regression method: 0.8333333333333334
Accuracy for Support Vector Machine method: 0.8333333333333334
Accuracy for Decision tree method: 0.8333333333333334
Accuracy for K nearsdt neighbors method: 0.8333333333333334
```

What have we learned?



- The data wrangling is the most crucial part of the data analysis, since it gives us the fundaments to know that our data being studied is correctly classified.
- The quickest way to perform an EDA is through SQL which gives us concrete answers to any question. The Data Viz through graphs gives us a detailed idea of how data behaves.
- The predictive model will give us an 83% accurate result if we give to it different values that we want to simulate.

Appendix

- All of the Jupyter Notebooks used on the complete journey are found on mi repository on the following Github:
  - Coursera_IBM_DataScience (github.com)
- The data sets were uploaded to a local server on MySQL, however, the CSV files can be uploaded to any desired SQL server for its manipulation.