



IBM Developer  
SKILLS NETWORK

# Winning Space Race with Data Science

Bruno Andrielli  
15/10/2024



# Outline

---

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

# Executive Summary

---

- On this project the main data science methodologies were put into practice, we began extracting the data and cleansing it furthermore we understood it through exploratory data analysis and with all set and done tried applying machine learning models to predict the outcomes of rocket launches.
- We observed that different machine learning techniques often came close to the same accuracy, indicating the need to better sophisticate our models through deeper analysis of the data or gathering more information.

# Introduction

---

- In this project we analyzed the data from different spaceX launches that were diverse in their landing outcome, orbit, payload, model, versions and other relevant factors
- We seek to predict and better understand what enables the launchers to successfully land back to retain expenses.



Section 1

# Methodology

# Methodology

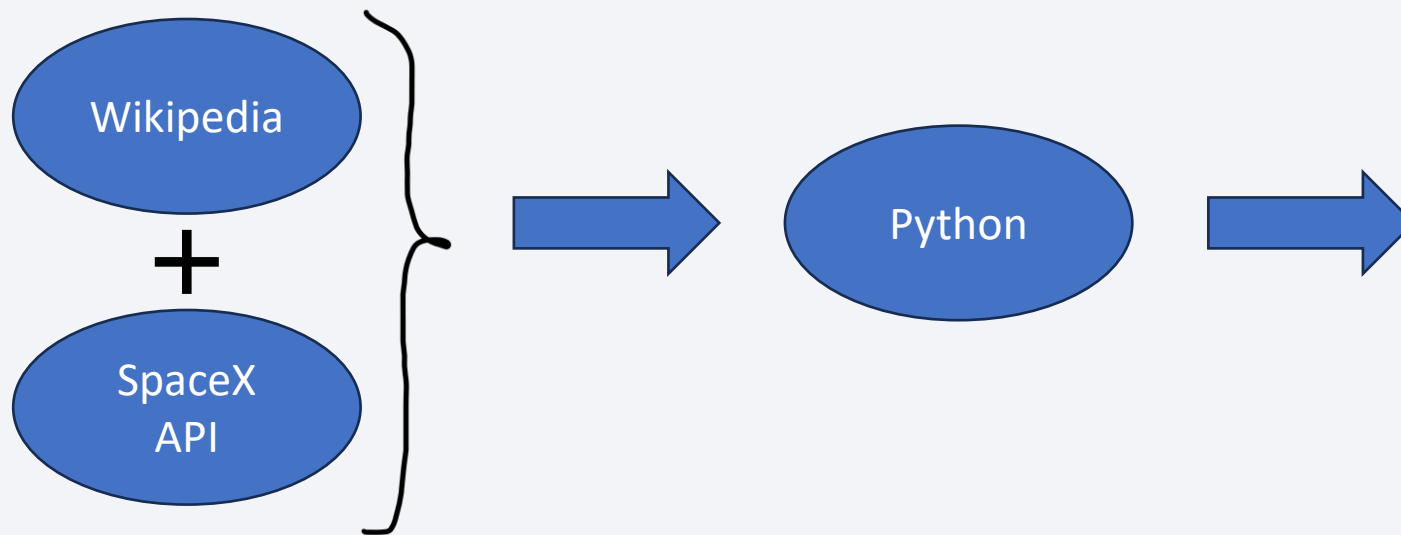
---

## Executive Summary

- Data collection methodology:
  - We collected the data through SpaceX API and web scrapping
- Perform data wrangling
  - We deleted nan values and labeled the outcomes
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - We mainly used scikit learn from python and GridSearch

# Data Collection

- We used python to web scrape some data and in addition and to make sure we didn't lose any important data we also got it from SpaceX own API



```
In [19]: df.head(8)
```

	FlightNumber	Date	BoosterVersion	PayloadMass	Orbit	LaunchSite	Outcome	Flights	GridFins	Reused	Legs	LandingPad	Block
0	1	2010-06-04	Falcon 9	6104.959412	LEO	CCAFS SLC 40	None None	1	False	False	False	NaN	
1	2	2012-05-22	Falcon 9	525.000000	LEO	CCAFS SLC 40	None None	1	False	False	False	NaN	
2	3	2013-03-01	Falcon 9	677.000000	ISS	CCAFS SLC 40	None None	1	False	False	False	NaN	
3	4	2013-09-29	Falcon 9	500.000000	PO	VAFB SLC 4E	False Ocean	1	False	False	False	NaN	
4	5	2013-12-03	Falcon 9	3170.000000	GTO	CCAFS SLC 40	None None	1	False	False	False	NaN	
5	6	2014-01-06	Falcon 9	3325.000000	GTO	CCAFS SLC 40	None None	1	False	False	False	NaN	
6	7	2014-04-18	Falcon 9	2296.000000	ISS	CCAFS SLC 40	True Ocean	1	False	False	True	NaN	
7	8	2014-07-14	Falcon 9	1316.000000	LEO	CCAFS SLC 40	True Ocean	1	False	False	True	NaN	

We can use the following line of code to determine the success rate:

```
In [20]: df['Outcome'] == 'True Ocean'
```

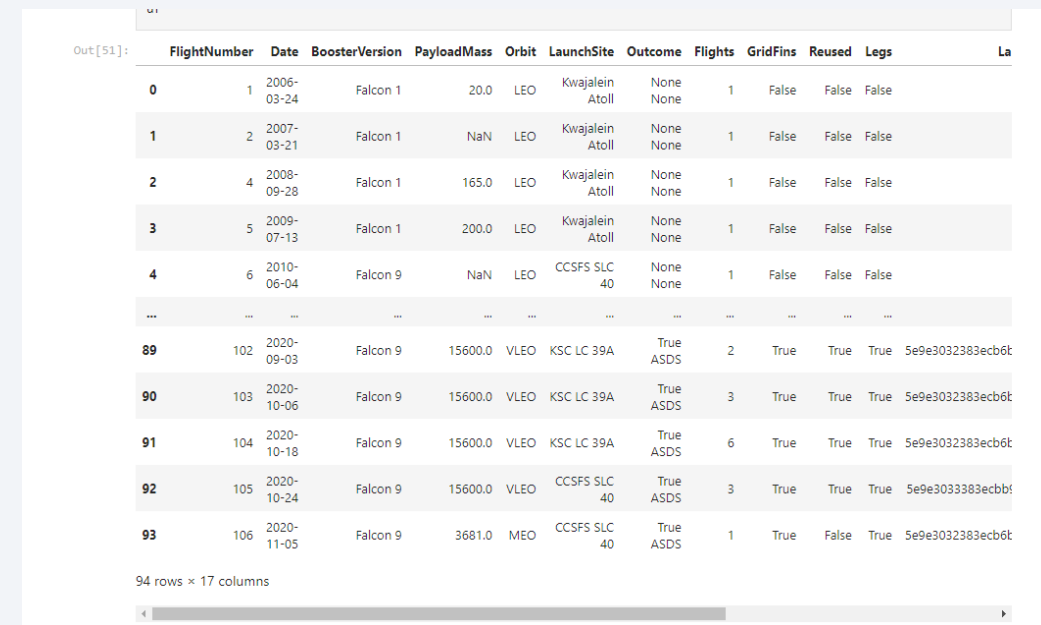
# Data Collection – SpaceX API

- <https://github.com/bruno0906/Coursera---data-science-SpaceX-project/blob/main/jupyter-labs-spacex-data-collection-api.ipynb>

1 – REQUEST response = requests.get(spacex\_url)

2 – PARSE data = pd.json\_normalize(response.json())

3 – Apply different methods to get the data in the format we need it to be in a table



	FlightNumber	Date	BoosterVersion	PayloadMass	Orbit	LaunchSite	Outcome	Flights	GridFins	Reused	Legs	LaunchComplex
0	1	2006-03-24	Falcon 1	20.0	LEO	Kwajalein Atoll	None	1	False	False	False	
1	2	2007-03-21	Falcon 1	NaN	LEO	Kwajalein Atoll	None	1	False	False	False	
2	4	2008-09-28	Falcon 1	165.0	LEO	Kwajalein Atoll	None	1	False	False	False	
3	5	2009-07-13	Falcon 1	200.0	LEO	Kwajalein Atoll	None	1	False	False	False	
4	6	2010-06-04	Falcon 9	NaN	LEO	CCSFS SLC 40	None	1	False	False	False	
...	...	...	...	...	...	...	...	...	...	...	...	
89	102	2020-09-03	Falcon 9	15600.0	VLEO	KSC LC 39A	True ASDS	2	True	True	True	5e9e3032383ecb6t
90	103	2020-10-06	Falcon 9	15600.0	VLEO	KSC LC 39A	True ASDS	3	True	True	True	5e9e3032383ecb6t
91	104	2020-10-18	Falcon 9	15600.0	VLEO	KSC LC 39A	True ASDS	6	True	True	True	5e9e3032383ecb6t
92	105	2020-10-24	Falcon 9	15600.0	VLEO	CCSFS SLC 40	True ASDS	3	True	True	True	5e9e3033383ecb6t
93	106	2020-11-05	Falcon 9	3681.0	MEO	CCSFS SLC 40	True ASDS	1	True	False	True	5e9e3032383ecb6t

94 rows × 17 columns



# Data Collection - Scraping

---

- <https://github.com/bruno0906/Coursera---data-science-SpaceX-project/blob/main/jupyter-labs-webscraping.ipynb>

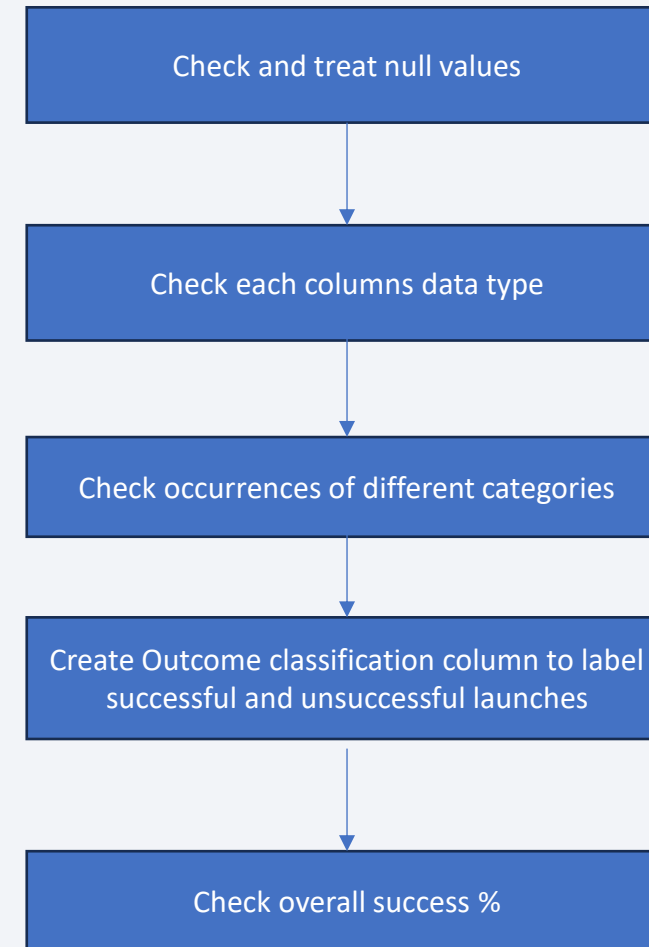
- 1 – Request Data
- 2 – Transform it into a BeautifulSoup
- 3 – Filter Soup to get the table we need
- 4 – Iterate the table to get column names and data

Place your flowchart of web scraping here

# Data Wrangling

---

- <https://github.com/bruno0906/Course-ra---data-science-SpaceX-project/blob/main/abs-jupyter-spacex-Data%20wrangling.ipynb>



# EDA with Data Visualization

---

- For EDA we mainly used scatterplots to verify the impact/relationship of different values with each other but used once a bar chart. Also, checked the success rate by a yearly trend with a lineplot. To ended it we created a dataset with dummy variables for categorical variables we would want to use on our machine learning models
- <https://github.com/bruno0906/Coursera---data-science-SpaceX-project/blob/main/edadataviz.ipynb>

# EDA with SQL

---

- Drop
- Select
- From
- Where
- Limit
- Distinct
- Sum
- AVG
- Between
- Like
- Group BY
- Order BY
- [https://github.com/bruno0906/Coursera---data-science-SpaceX-project/blob/main/jupyter-labs-eda-sql-coursera\\_sqlite.ipynb](https://github.com/bruno0906/Coursera---data-science-SpaceX-project/blob/main/jupyter-labs-eda-sql-coursera_sqlite.ipynb)

# Build an Interactive Map with Folium

---

- Created markers as circles for launch sites and colored it based on success or failure landing. Also created some lines and markers on near by railroad, coast, highway and city name to check its distance to a launch site, as it can be a useful information on how to locate yourself and how to get to the launchsite
- [https://github.com/bruno0906/Coursera---data-science-SpaceX-project/blob/main/lab\\_jupyter\\_launch\\_site\\_location.ipynb](https://github.com/bruno0906/Coursera---data-science-SpaceX-project/blob/main/lab_jupyter_launch_site_location.ipynb)



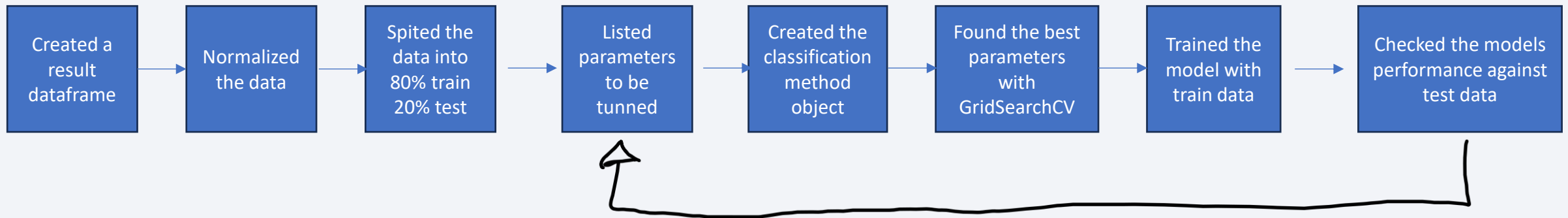
# Build a Dashboard with Plotly Dash

---

- Created a dashboard where you can filter the launch site and payload in KG. They affect a % of success landing pie chart and a scatterplot of payload mass by the classification if the landing was successful or not and a hue of launches booster version. With all that anyone can extract valuable insights of the best combinations for a successful landing.
- [https://github.com/bruno0906/Coursera---data-science-SpaceX-project/blob/main/spacex\\_dash\\_app.py](https://github.com/bruno0906/Coursera---data-science-SpaceX-project/blob/main/spacex_dash_app.py)

# Predictive Analysis (Classification)

---



Repeated the process for each classification method like:

k-nearest neighbor

Tree-decision classifier

Logistical regression

Svm

- [https://github.com/bruno0906/Coursera---data-science-SpaceX-project/blob/main/SpaceX Machine%20Learning%20Prediction Part 5.ipynb](https://github.com/bruno0906/Coursera---data-science-SpaceX-project/blob/main/SpaceX%20Machine%20Learning%20Prediction%20Part%205.ipynb)

# Results

---

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



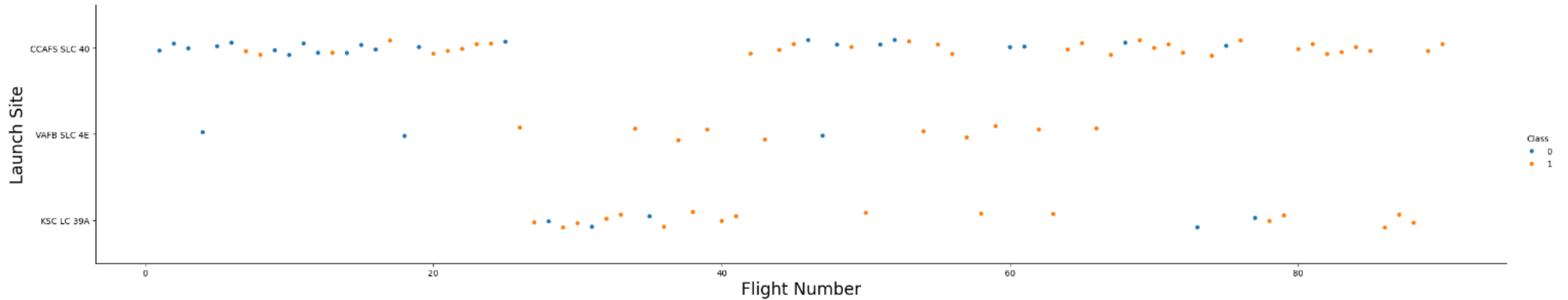
The background of the slide is an abstract composition. It features a dark blue base color. Overlaid on this are numerous diagonal streaks in shades of red and cyan. A faint, light blue grid pattern is also visible, particularly in the lower half of the image. The overall effect is dynamic and technological.

Section 2

# Insights drawn from EDA



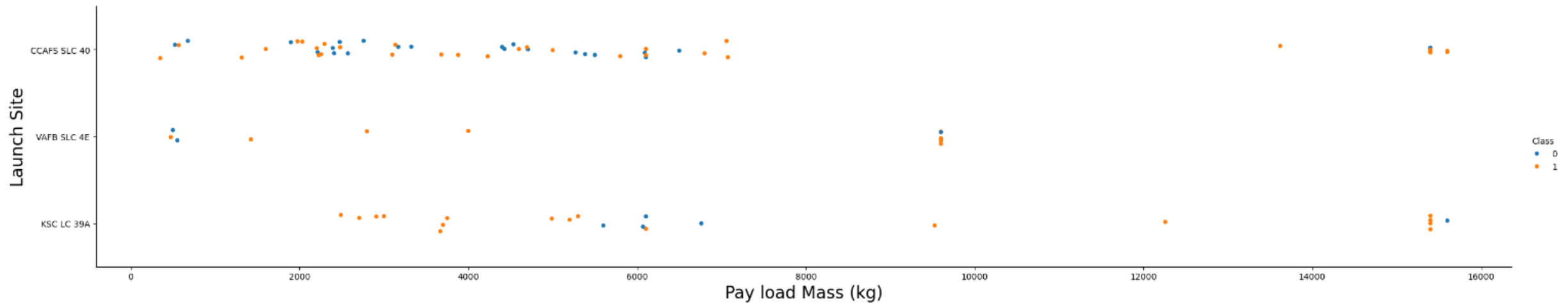
# Flight Number vs. Launch Site



- From the chart we can analyze that for all launch sites the higher the flight number the better it begun to perform. Also, we can see that for launch site SLC 4E we don't have any data past flight number 70

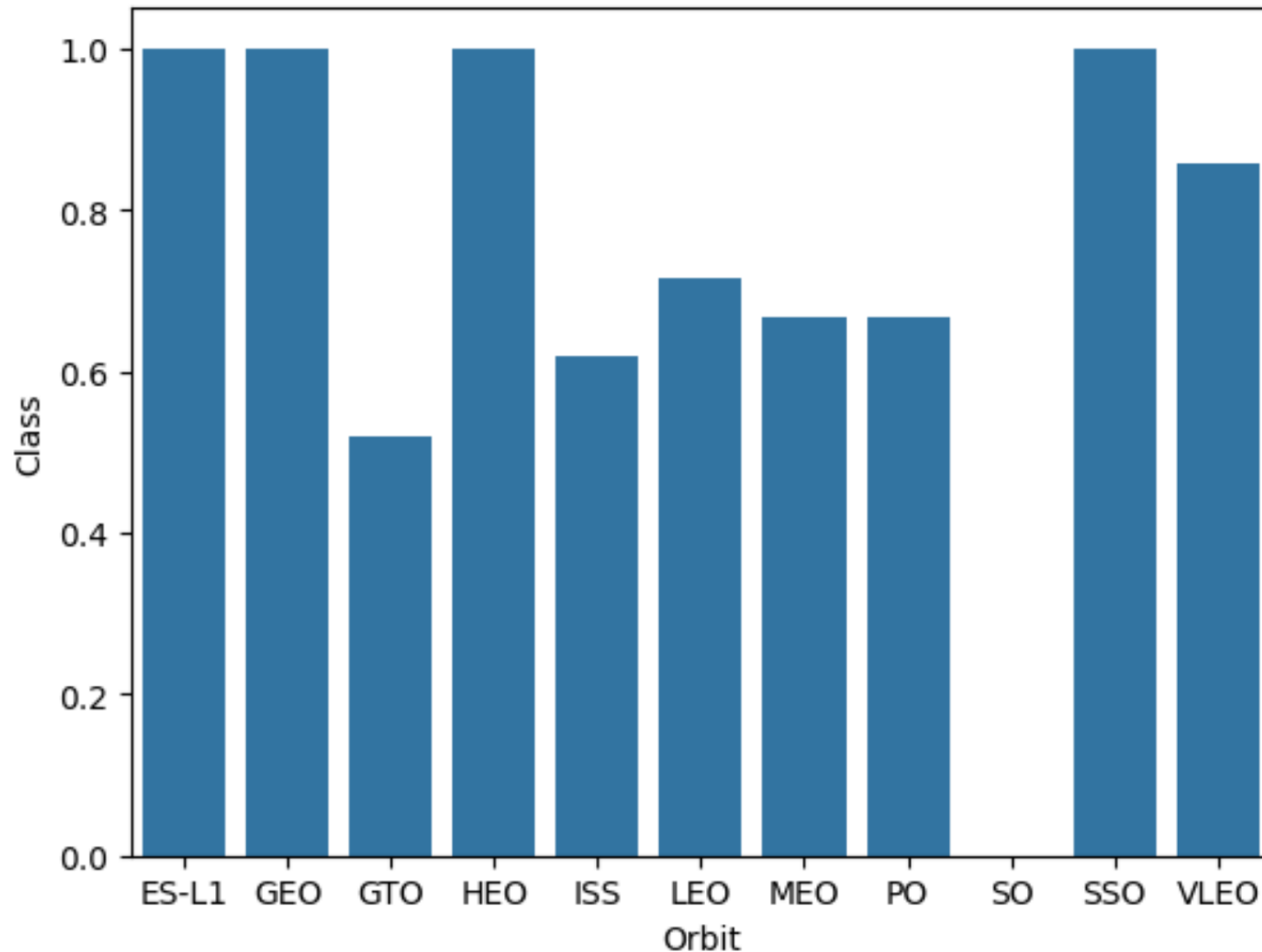


# Payload vs. Launch Site



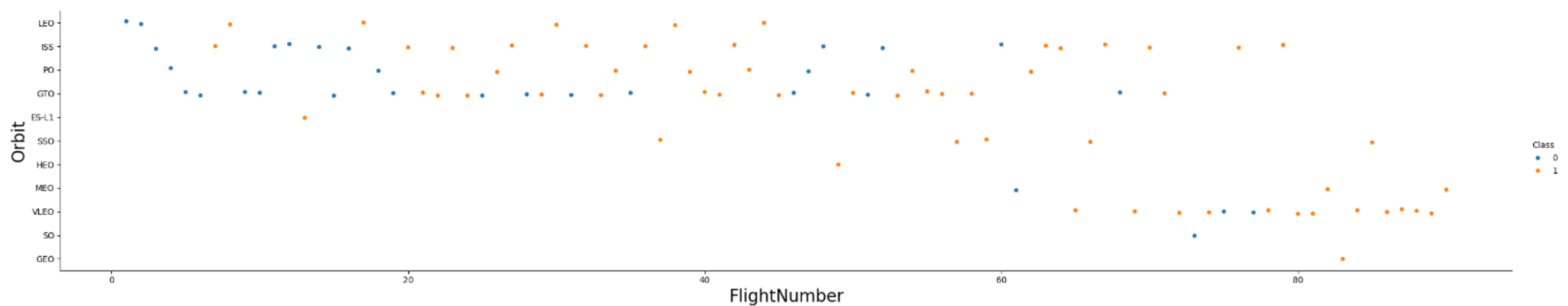
- For launch site SLC 40 and SLC 4E we can see they perform much better with high pay load mass, the same for LC 39A, which in fact, has a sweet spot around 15k where it only has successes. For it we can also see that low pay loads, beneath 5000 are also successful.

# Success Rate vs. Orbit Type



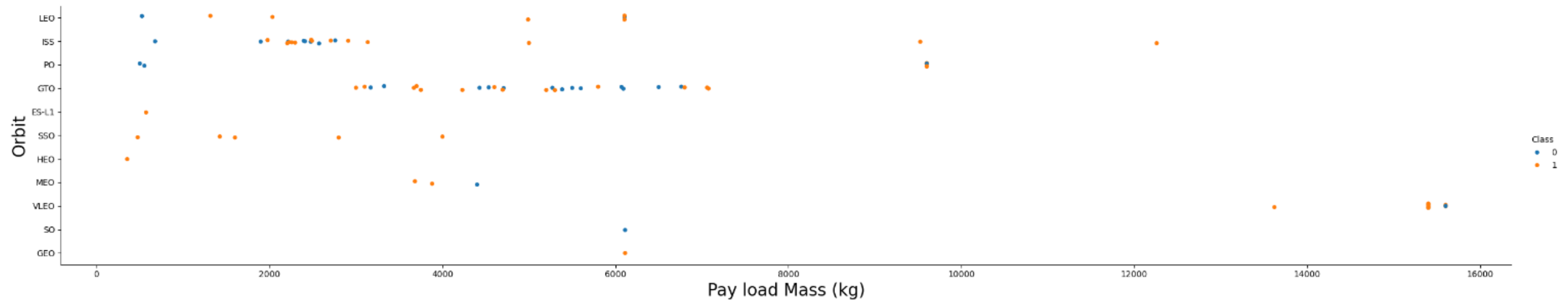
- By the graph we can attest that the best orbits are ES-L1, GEO, HO and SSO with the worst performing being SO.

# Flight Number vs. Orbit Type



- From the graph we can see that VLEO orbit performs really well after flight number 80. And that SSO has been successful multiple times with different flight numbers.
- And that GTO ISS and PO orbit had some successes and failures over flight numbers

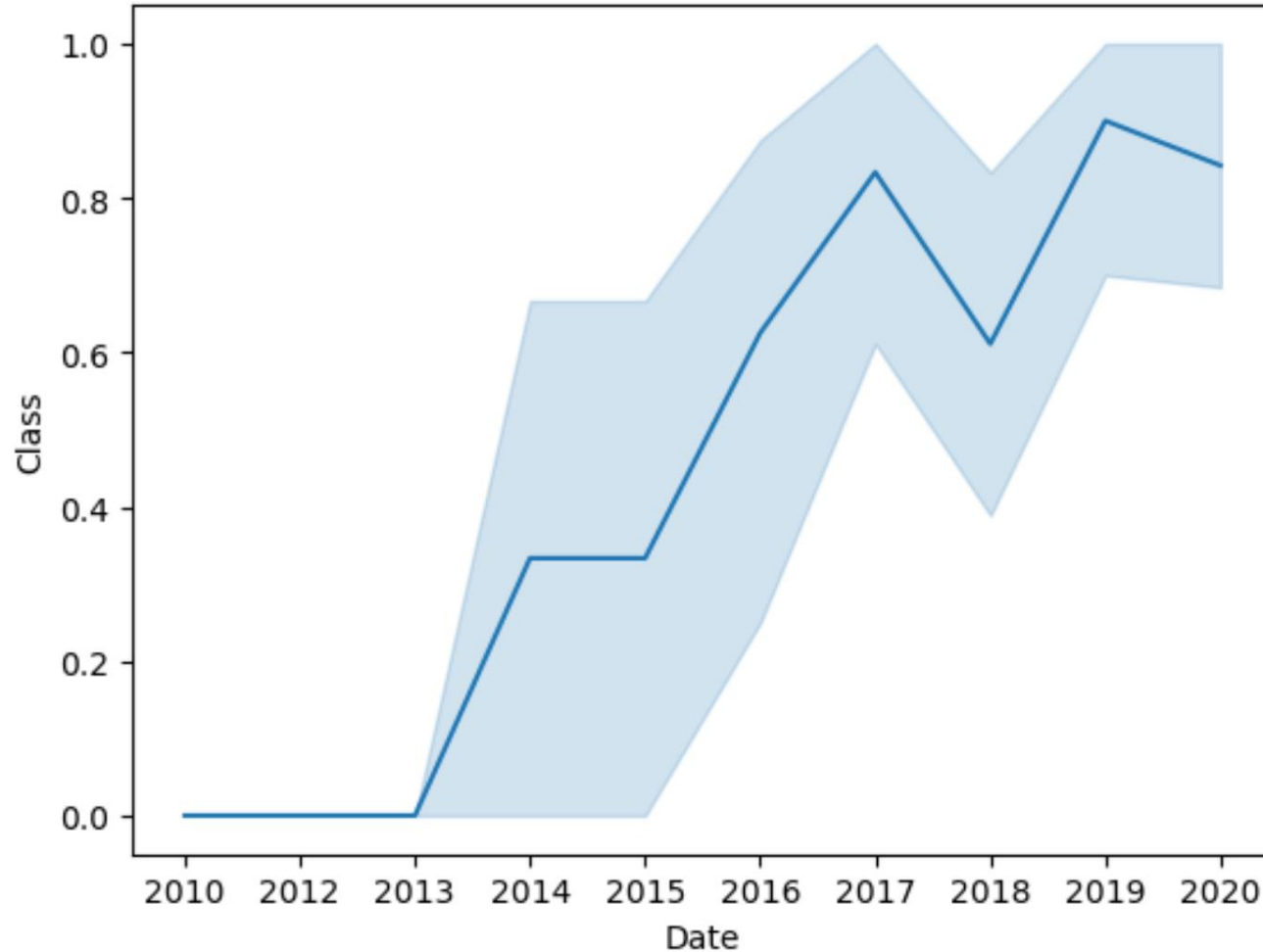
# Payload vs. Orbit Type



- LEO, VLEO, ISS and PO have performed better with high pay load mass.
- SSO performs well with different pay load masses but doesn't gave any value above the 4K KG

# Launch Success Yearly Trend

---



- We can see that with the passage of the year the success rate increased with an odd year in 2018. Which makes sense since our technology keeps improving as well as our knowledge



# All Launch Site Names

---

- A distinct call on the launch site names gave us the result we needed, since distinct show values that differs from the others

## **Launch\_Site**

---

CCAFS LC-40

VAFB SLC-4E

KSC LC-39A

CCAFS SLC-40

# Launch Site Names Begin with 'CCA'

- I queried the column launch site and checked if it started with the characters cca

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

# Total Payload Mass

---

- I filtered the customer to be nasa so we only sum it payloads masses

```
%sql SELECT SUM(PAYLOAD_MASS__KG_) FROM SPACEXTABLE WHERE CUSTOMER == 'NASA (CRS)';
```

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

```
Done.
```

<b>SUM(PAYLOAD_MASS__KG_)</b>
-------------------------------

45596
-------

# Average Payload Mass by F9 v1.1

---

- I filtered the booster version to be f9 v1.1 and use the AVG function on payload mass

```
: AVG(PAYLOAD_MASS_KG_)  
2928.4
```

# First Successful Ground Landing Date

---

- I used the MIN function to get the earliest date as well a filter on ground landing to be successful

```
[45]: %sql SELECT MIN(DATE) FROM SPACEXTABLE WHERE Landing_Outcome = 'Success (ground pad)';
```

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

```
Done.
```

```
[45]: MIN(DATE)
```

```
2015-12-22
```



## Successful Drone Ship Landing with Payload between 4000 and 6000

---

- I filtered the boosters which have successfully landed on drone ship and had payload mass greater than 4000 but less than 6000

JCSAT-14

JCSAT-16

SES-10

SES-11 / EchoStar 105

# Total Number of Successful and Failure Mission Outcomes

---

- I calculated the number of values in mission\_outcome column

Mission_Outcome	COUNT(*)
Failure (in flight)	1
Success	98
Success	1
Success (payload status unclear)	1

# Boosters Carried Maximum Payload

---

- Using Max function, I filtered the data where the max value matched the pay load mass value of that row

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

# 2015 Launch Records

---

- I filtered the landing\_outcomes that were failed in drone ship. Displaying their booster versions, launch site names and month they were launched for the year of 2015

month	Booster_Version	Launch_Site
01	F9 v1.1 B1012	CCAFS LC-40
04	F9 v1.1 B1015	CCAFS LC-40

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

---

- I ranked 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

```
Done.  
Out[19]:
```

<b>Landing_Outcome</b>	<b>COUNT(*)</b>
No attempt	10
Success (drone ship)	5
Failure (drone ship)	5
Success (ground pad)	3
Controlled (ocean)	3
Uncontrolled (ocean)	2
Failure (parachute)	2
Precluded (drone ship)	1

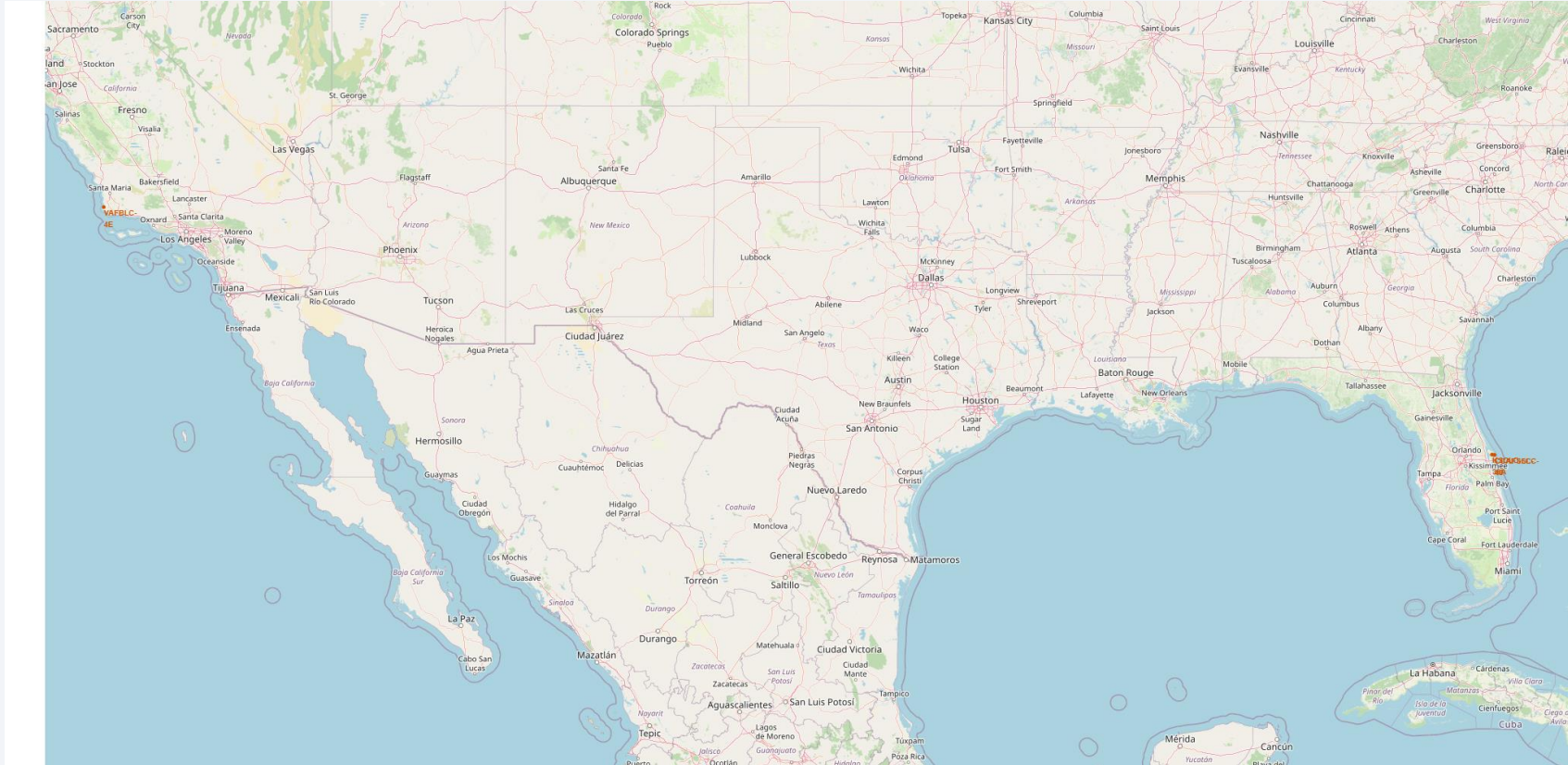
A satellite view of Earth from space, showing the curvature of the planet and city lights at night. The background is a deep blue gradient.

Section 3

# Launch Sites Proximities Analysis



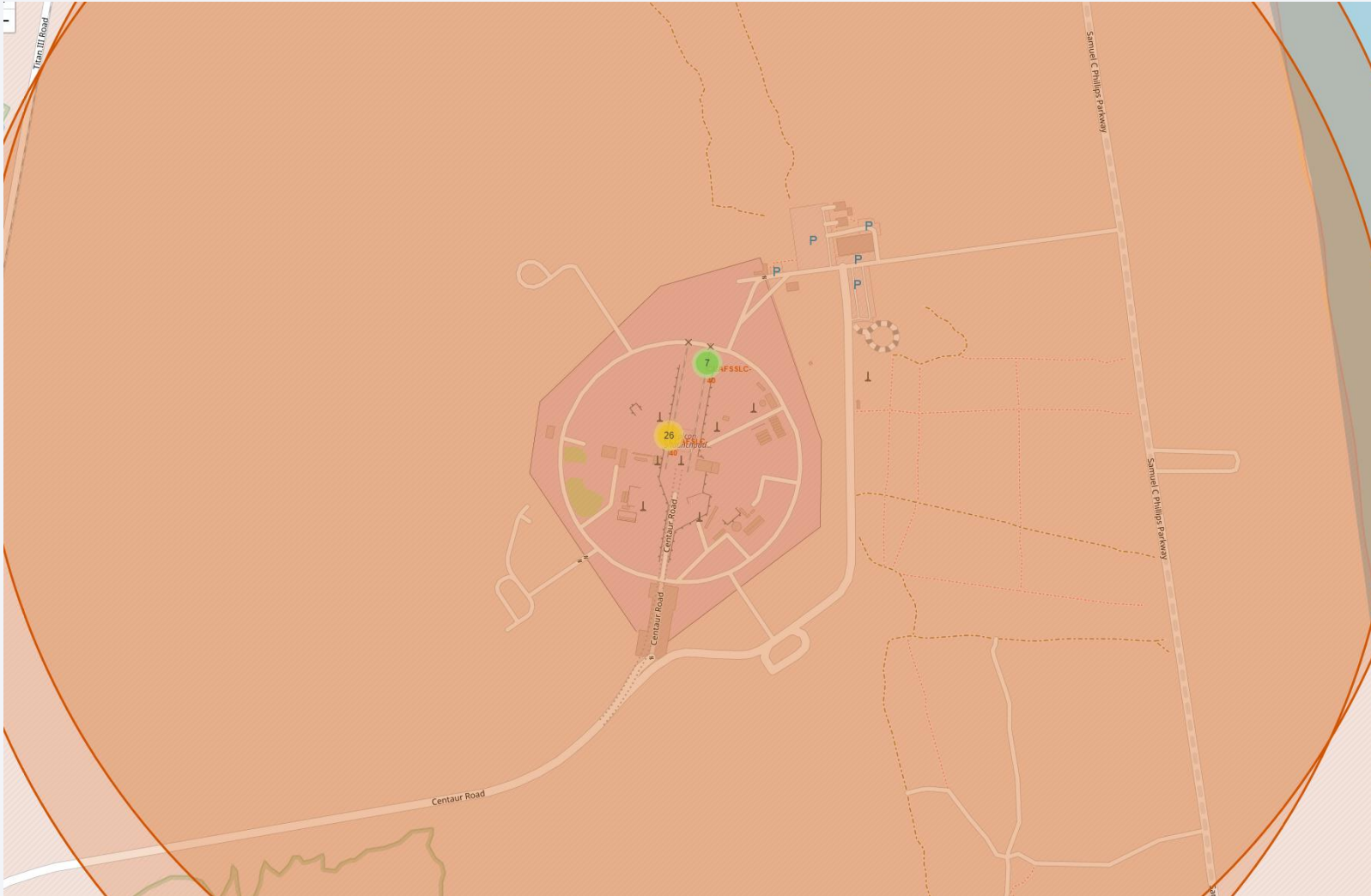
# launch Site Locations



- We can see that 3 launch sites are near each other close to Miami, and one is further away in the other side close to Miami

# Launch Site and success

---



- Here we can compare the performance of two different sites that are close to each other, by seeing its color.

# Site proximity to interest points

- We can see that two launch sites have close proximity to a railroad, a coast and a highway.
- And they are approximately 19.64 Km direct distance from cape canaveral







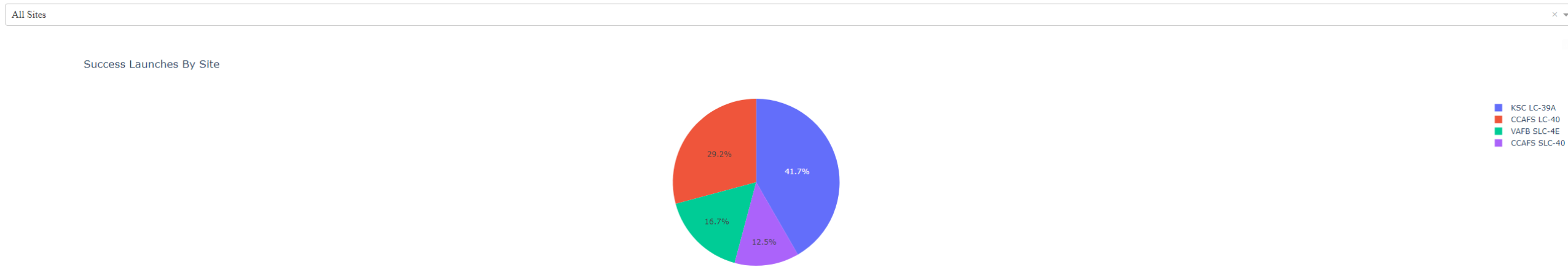
Section 4

# Build a Dashboard with Plotly Dash

# Success by Site

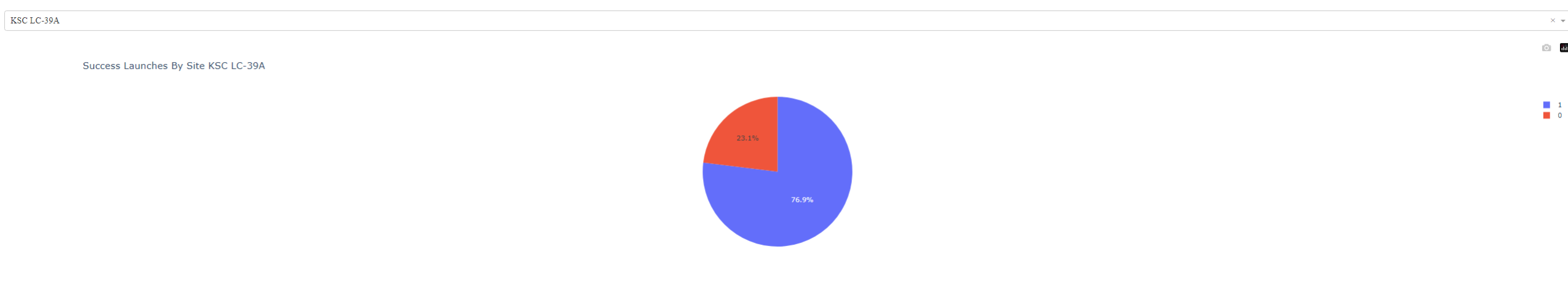
---

## SpaceX Launch Records Dashboard



- We can see the best Site is KSC LC-39A

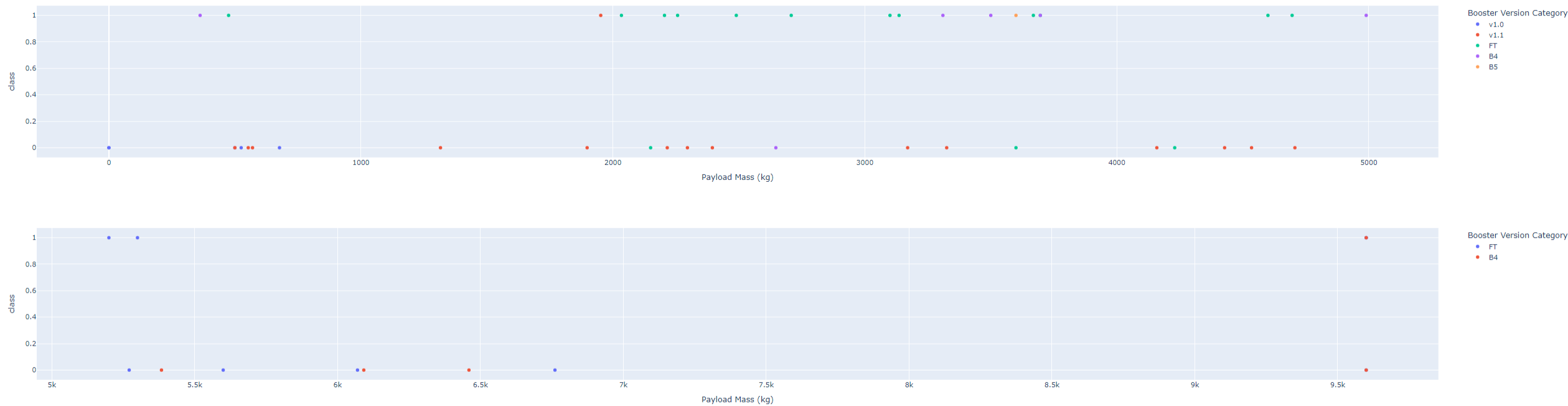
# KSC LC-39 A Success ratio



- We can see this launch site has around a 77% success ratio

# <Dashboard Screenshot 3>

Success based on Payload mass - all sites



- We can see that for payload mass over 5k the FT booster version performs better and for extreme values B4 is a safer option.
- For below 5k Kg the FT and B4 performed better making them the best version overall





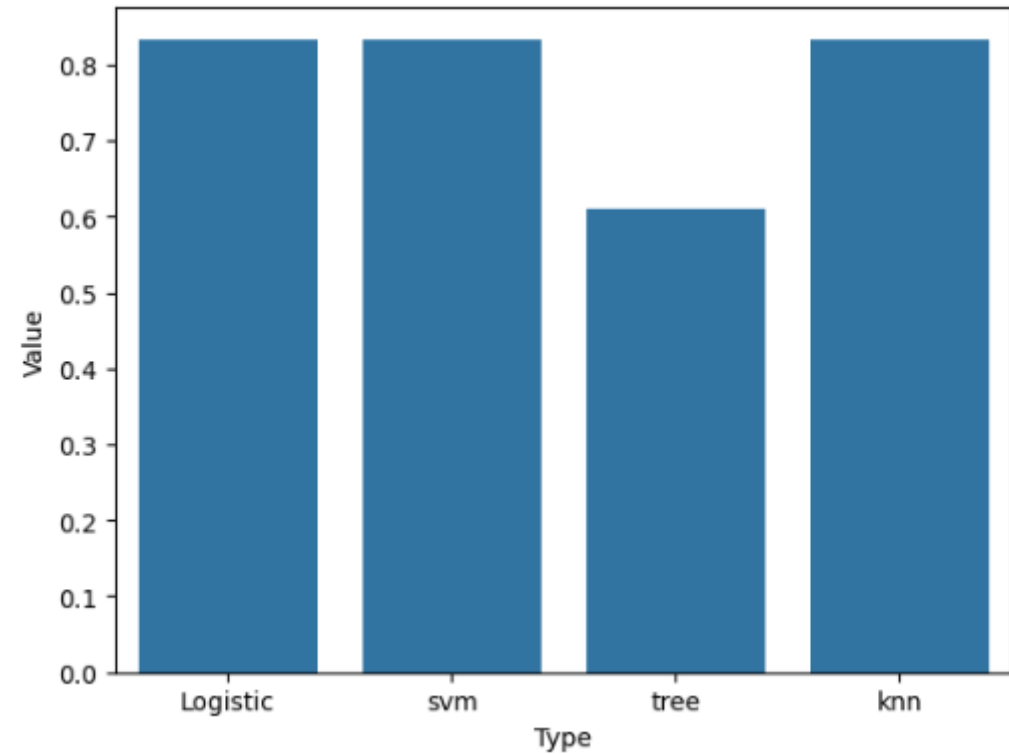
Section 5

# Predictive Analysis (Classification)

# Classification Accuracy

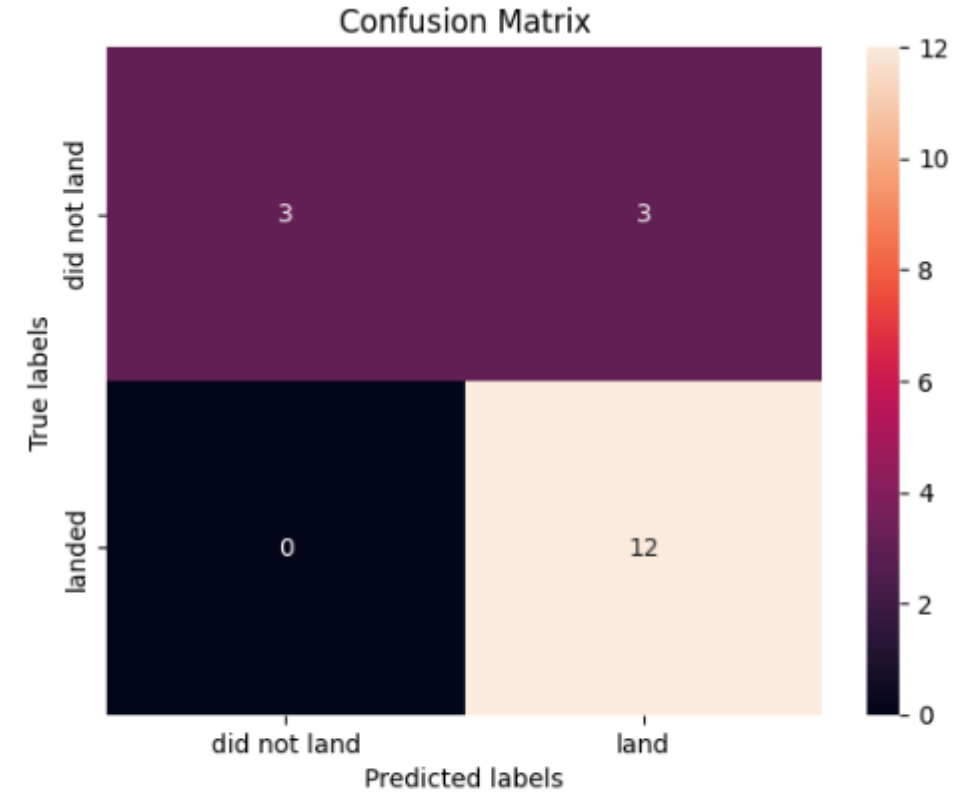
---

- Logistic, SVM and KNN performed the best but the same.



# Confusion Matrix

- We can see that the model struggled predicting launches that didn't land, having 3 false positives



TASK 12

# Conclusions

---

- From everything displayed so far, we can attest with 83.33% certainty if a rocket will land or not.
- We need to deep and improve our models maybe sorting specific orbits and launch site combinations since we saw it has a lot of impact on success rates

# Appendix

---

- <https://github.com/bruno0906/Coursera---data-science-SpaceX-project>

Thank you!

