IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Arcos de las Heras, Diego
20th of March 2023

Outline

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

Executive Summary

- Summary of methodology
  - Data collection and pre-processing → Data cleaning
  - Data analysis → EDA and obtaining of insights (charts included)
  - Visualization → Folium and Dash analysis
  - Predictive analysis → Classifications algorithms evaluation
- Summary of all results
  - We will determinate which if we have success landing our first stage and which features are the most important to predict it (and the cost).

Introduction

- Project background and context

  Space Y, a new aircraft company wants to start its path in this industry. But first, it needs to make a comprehensive study where to determinate if they can compete with Space X. To do this, we have to obtain all the possible information about our competitor and create and predictive model to know if we can earn money recovering the rocket's first stage (Falcon 9) or not, depending on other parameters like orbit, payload or customer. Anyway, the main objective is to reduce the cost of launches and see which features are implicated.

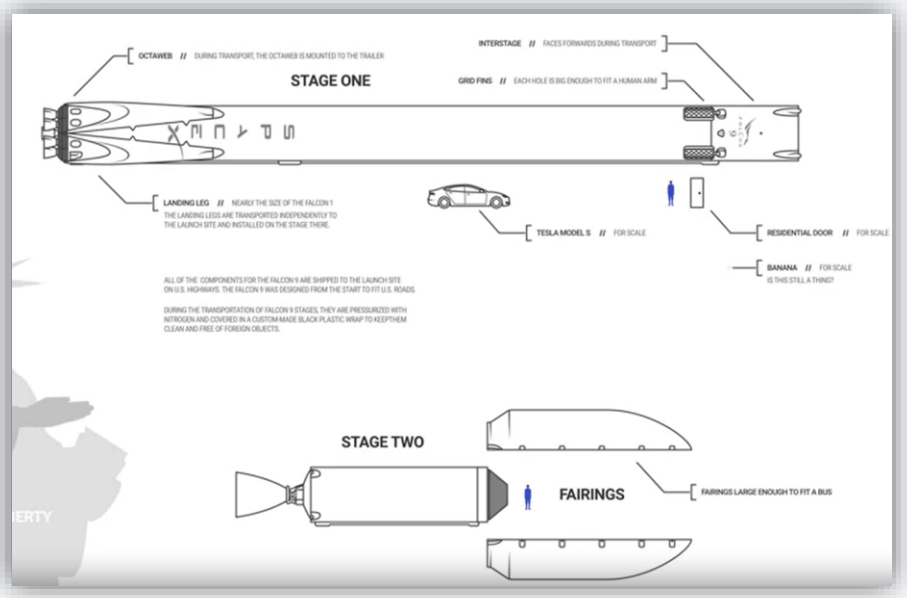
- Problems you want to find answers

  Main questions:

  ❑ How much does each launch cost?

  ❑ Could the first stage land properly?

  ❑ Is necessary to reuse the first stage?



Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - By SpaceX REST API and carrying out web scraping (BeautifulSoup package).
- Perform data wrangling
  - Transforming our JSON data format into a flat dataset, sampling data and then cleaning it (imputing nulls, changing formats, merging the REST API responses, etc.).
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - Evaluating several classification algorithms (Decision Trees, Logistic regression, etc.)

Data Collection – SpaceX API

Flowchart process

1 - Create the call functions:

- getBoosterVersion()
- getLaunchSite()
- getPayloadData()
- getCoreData()

2 - Normalize JSON data format into a data frame

3 - Filter a new dataset only with Falcon 9 aircraft type

4 - Impute nulls in variable ‘PayLoadMass’ with the mean.

Data collection – GitHub link

Data Collection - Scraping

Flowchart process

1 - Request the Falcon9 Launch Wiki page from its URL creating a BeautifulSoup object

2 - Extract all column/variable names from the HTML table header

3 - Create a data frame by parsing the launch HTML tables

Data web scraping – GitHub link

Data Wrangling

Flowchart process

1 - Carry out a brief analysis: nulls, type attributes

2 - Calculate the number of launches on each site (the same process with orbit type and mission outcome

3 - Generate a binary feature called ‘landing_class’, with 0 if we have a bad outcome and vice versa

4 - We can find a 67% of success in our launches

Data wrangling – GitHub link

EDA with Data Visualization

Flowchart process

1 - Plotting relationship charts between several attributes

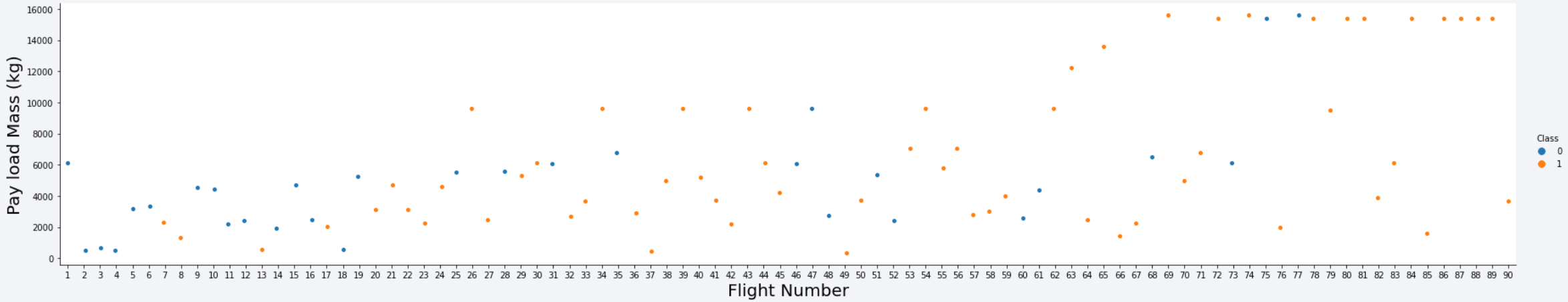
2 - Visualize the land success yearly trend

3 - Transform categorical variables into numeric by dummies and change type of the rest to ‘float’

Data EDA visualization – GitHub link

EDA with Data Visualization

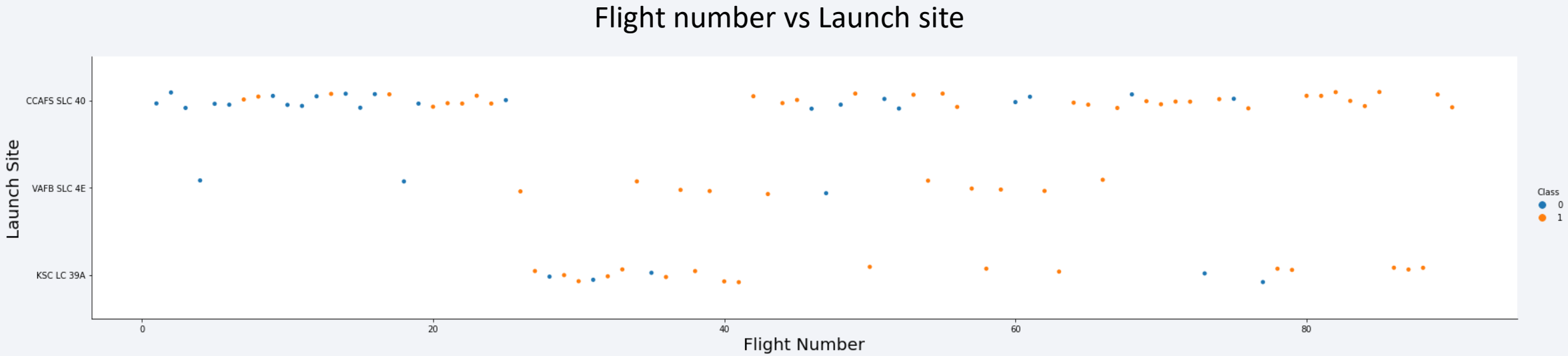
Flight number vs Pay load mass (kg)



We see that as the flight number increases, the first stage is more likely to land successfully. The payload mass is also important; it seems the more massive the payload, the less likely the first stage will return.

Data EDA visualization – GitHub link

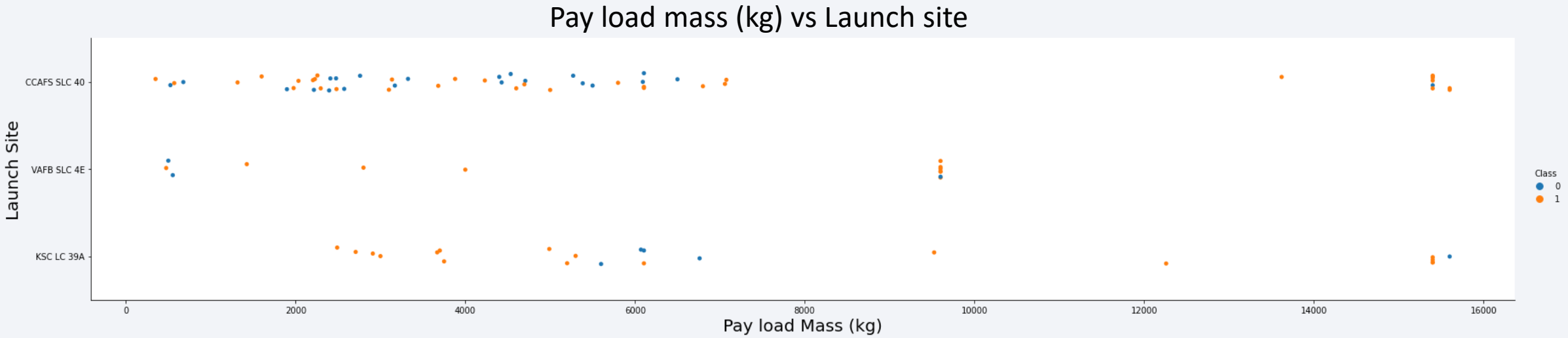
EDA with Data Visualization



We see that different launch sites have different success rates. CCAFS LC-40, has a success ratio of 60 %, while KSC LC-39A and VAFB SLC 4E has a success rate of 77%. In addition, CCAFS LC-40 was the most used place from flight number 0 to 20 and from 40 to 85, followed by KSC LC-39A and the last VAFB SLC 4E.

Data EDA visualization – GitHub link

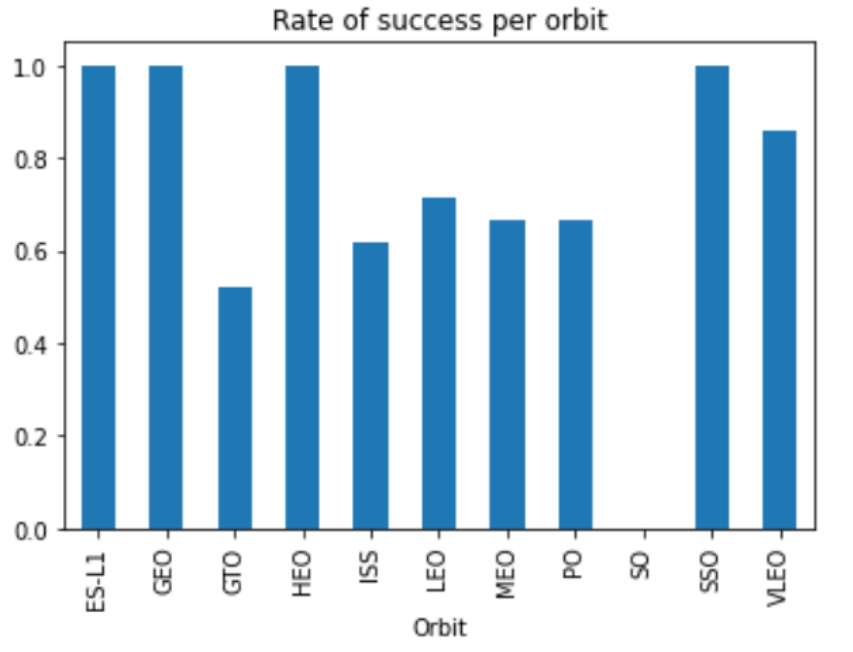
EDA with Data Visualization

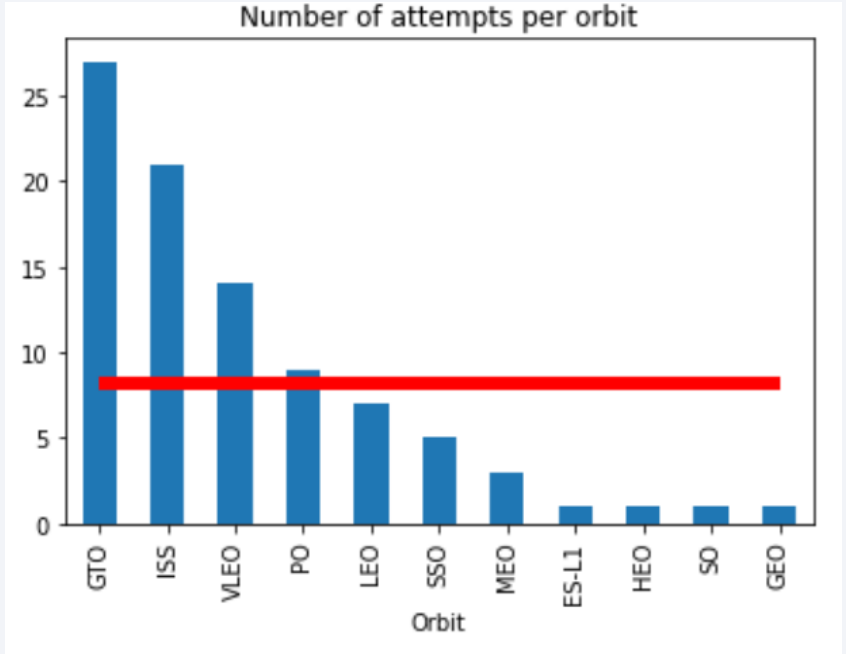


Now if we observe the chart you will find for the VAFB-SLC launch site there are no rockets launched for heavy pay load mass (greater than 10000). Also, as we have seen before, CCAFS LC-40 is the most used place.

Data EDA visualization – GitHub link

EDA with Data Visualization



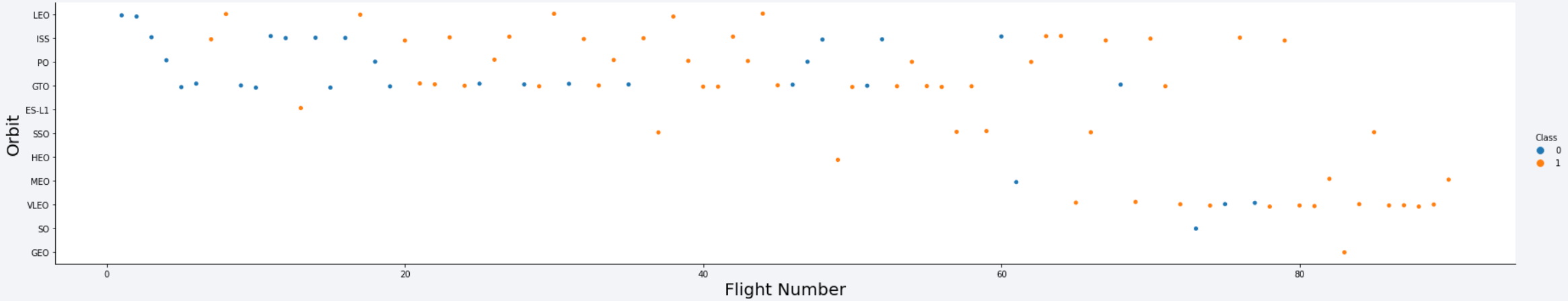


We could determinate that ES-L1, GEO, HEO and SSO orbits have a 100% of success, meanwhile GTO and ISS have the poorest rate. Although if we observe the number of launches per orbit, we could see that the majority do not pass the mean. For this reason ES-L1, GEO, HEA and SSO have this high ratio.

Data EDA visualization – GitHub link

EDA with Data Visualization

Flight number vs Orbit

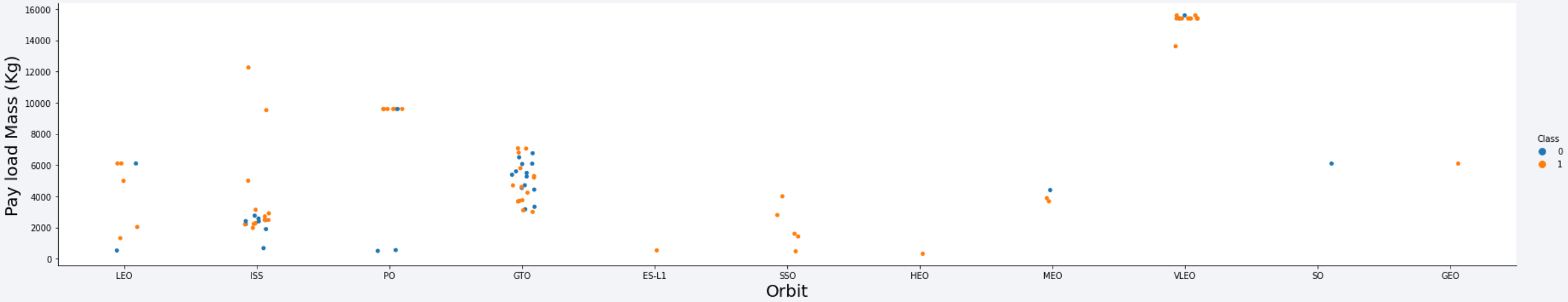


You should see that in the LEO orbit the success appears related to the number of flights; on the other hand, there seems to be no relationship between flight number when in GTO orbit. In addition orbit VLEO is only used for the last flights.

Data EDA visualization – GitHub link

EDA with Data Visualization

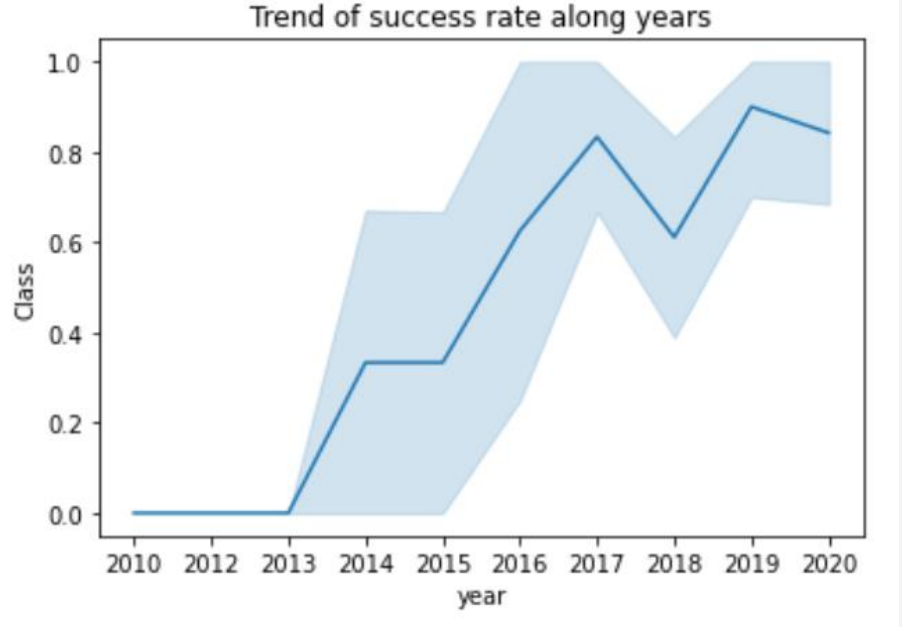
Pay Load Mass (kg) vs Orbit



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

Data EDA visualization – GitHub link

EDA with Data Visualization



In this graph, we can see that till 2013 the success ratio did not start to grow up. In 2017 had a small drop but in 2018 kept this growing trend.

Data EDA visualization – GitHub link

EDA with SQL

Data EDA SQL – GitHub link

First of all, we connected to database, and then:

- Display the names of the unique launch sites in the space mission
- Display 5 records where launch sites begin with the string 'CCA‘
- Display the total payload mass carried by boosters launched by NASA (CRS)
- Display average payload mass carried by booster version F9 v1.1
- List the date when the first successful landing outcome in ground pad was achieved
- List the names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000
- List the total number of successful and failure mission outcomes
- List the names of the booster_versions which have carried the maximum payload mass. Use a subquery
- List the records which will display the month names, failure landing_outcomes in drone ship ,booster versions, launch_site for the months in year 2015
- Rank the count of successful landing_outcomes between the date 04-06-2010 and 20-03-2017 in descending order

All Launch Site Names

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

```
%%sql

select distinct(Launch_Site) from SPACEXTBL
```

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

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

Launch Site Names Begin with 'CCA'

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

```
%%sql

select * from SPACEXTBL
where Launch_Site like 'CCA%'
limit 5
```

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

Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Orbit	Customer	Mission_Outcome	Landing _Outcome
04-06-2010	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success	Failure (parachute)
08-12-2010	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)
22-05-2012	07:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success	No attempt
08-10-2012	00:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500	LEO (ISS)	NASA (CRS)	Success	No attempt
01-03-2013	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677	LEO (ISS)	NASA (CRS)	Success	No attempt

Total Payload Mass

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

```
%%sql

select customer, sum(PAYLOAD_MASS__KG_) as Total_payload_mass from SPACEXTBL
where Customer like '%NASA (CRS)%'
```

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

Customer	Total_payload_mass
NASA (CRS)	48213

Average Payload Mass by F9 v1.1

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

```
%%sql

select Booster_Version, avg(PAYLOAD_MASS__KG_) as Avg_payload_mass from SPACEXTBL
where Booster_Version like '%F9 v1.1%'
```

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

Booster_Version	Avg_payload_mass
F9 v1.1 B1003	2534.6666666666665

First Successful Ground Landing Date

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

Hint:Use min function

```
%%sql

select Date, "Landing _Outcome" from SPACEXTBL
where "Landing _Outcome" like '%pad%'
order by Date Desc
Limit 1
```

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

Date	Landing _Outcome
22-12-2015	Success (ground pad)

Successful Drone Ship Landing with Payload between 4000 and 6000

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

```
%%sql

select Booster_Version, "Landing _Outcome" from SPACEXTBL
where "Landing _Outcome" like '%Success (drone ship)%' and (PAYLOAD_MASS__KG_ between 4000 and 6000)
```

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

Booster_Version	Landing _Outcome
F9 FT B1022	Success (drone ship)
F9 FT B1026	Success (drone ship)
F9 FT B1021.2	Success (drone ship)
F9 FT B1031.2	Success (drone ship)

Total Number of Successful and Failure Mission Outcomes

List the total number of successful and failure mission outcomes

```
%%sql

select Mission_Outcome, count(Mission_Outcome) as all_missions from SPACEXTBL
group by Mission_Outcome
```

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

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

Boosters Carried Maximum Payload

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

2015 Launch Records

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, 4, 2) as month to get the months and substr(Date,7,4)='2015' for year.

```
%%sql

select substr(Date,4,2) as month, substr(Date,7,4) as year,Booster_Version, Launch_Site, "Landing _Outcome" from SPACEXTBL
where substr(Date,7,4)='2015' and "Landing _Outcome" like '%Failure (drone ship)%'
```

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

month	year	Booster_Version	Launch_Site	Landing _Outcome
01	2015	F9 v1.1 B1012	CCAFS LC-40	Failure (drone ship)
04	2015	F9 v1.1 B1015	CCAFS LC-40	Failure (drone ship)

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

```
%%sql

select "Landing _Outcome", Date from SPACEXTBL
where "Landing _Outcome" like '%ucc%' and (substr(Date,7,4) >= '2010' and substr(Date,7,4) <= '2017')
order by substr(Date,7,4) desc
```

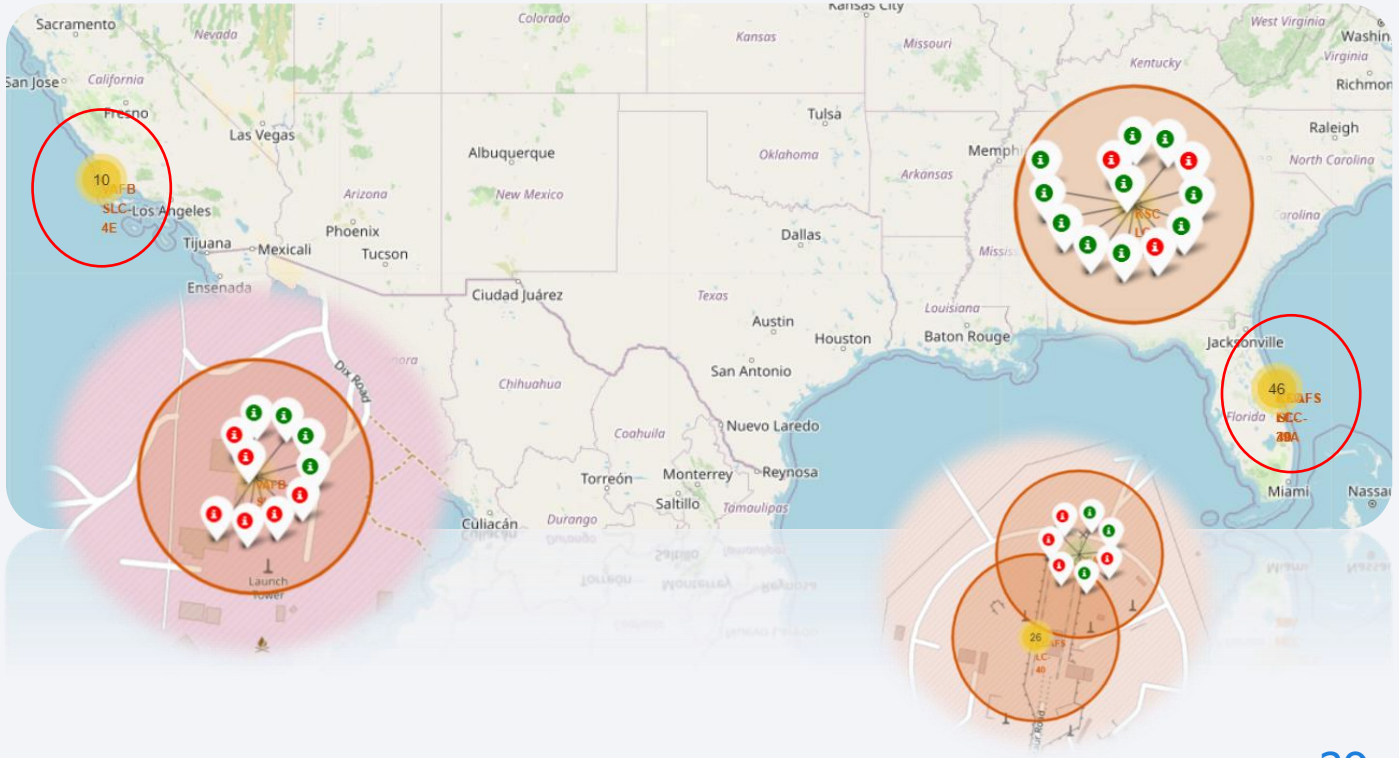
```
 * sqlite:///my_data1.db
Done.
```

Landing _Outcome	Date
Success (drone ship)	14-01-2017
Success (ground pad)	19-02-2017
Success (drone ship)	30-03-2017
Success (ground pad)	01-05-2017
Success (ground pad)	03-06-2017
Success (drone ship)	23-06-2017
Success (drone ship)	25-06-2017
Success (ground pad)	14-08-2017
Success (drone ship)	24-08-2017
Success (ground pad)	07-09-2017
Success (drone ship)	09-10-2017
Success (drone ship)	11-10-2017
Success (drone ship)	30-10-2017
Success (ground pad)	15-12-2017
Success (drone ship)	08-04-2016
Success (drone ship)	06-05-2016
Success (drone ship)	27-05-2016
Success (ground pad)	18-07-2016
Success (drone ship)	14-08-2016
Success (ground pad)	22-12-2015

Build an Interactive Map with Folium

We have marked the principal locations where the rockets were launched, with different colors depending on the success (green or red) and clustering by its proximity.

	Launch Site	Lat	Long
0	CCAFS LC-40	28.562302	-80.577356
1	CCAFS SLC-40	28.563197	-80.576820
2	KSC LC-39A	28.573255	-80.646895
3	VAFB SLC-4E	34.632834	-120.610745

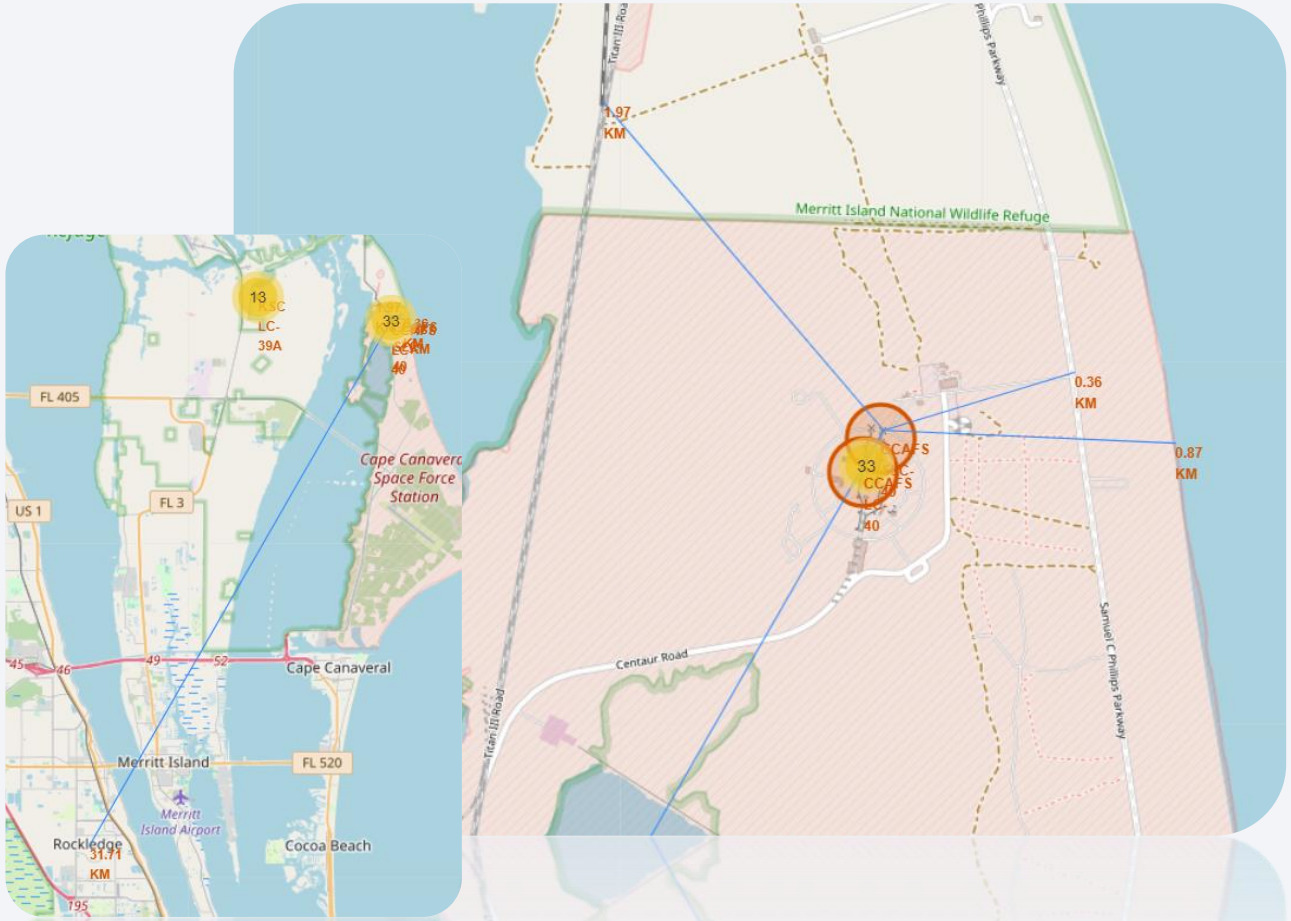


[Data launch site Folium – GitHub link]

Build an Interactive Map with Folium

In the second part, we plotted the distance between one launch point and:

- Nearest highway → 0.36 km
- Rockledge city → 31.71km
- Nearest railway → 1.97 km
- Nearest coast distance → 0.87 km



[Data launch site Folium – GitHub link]

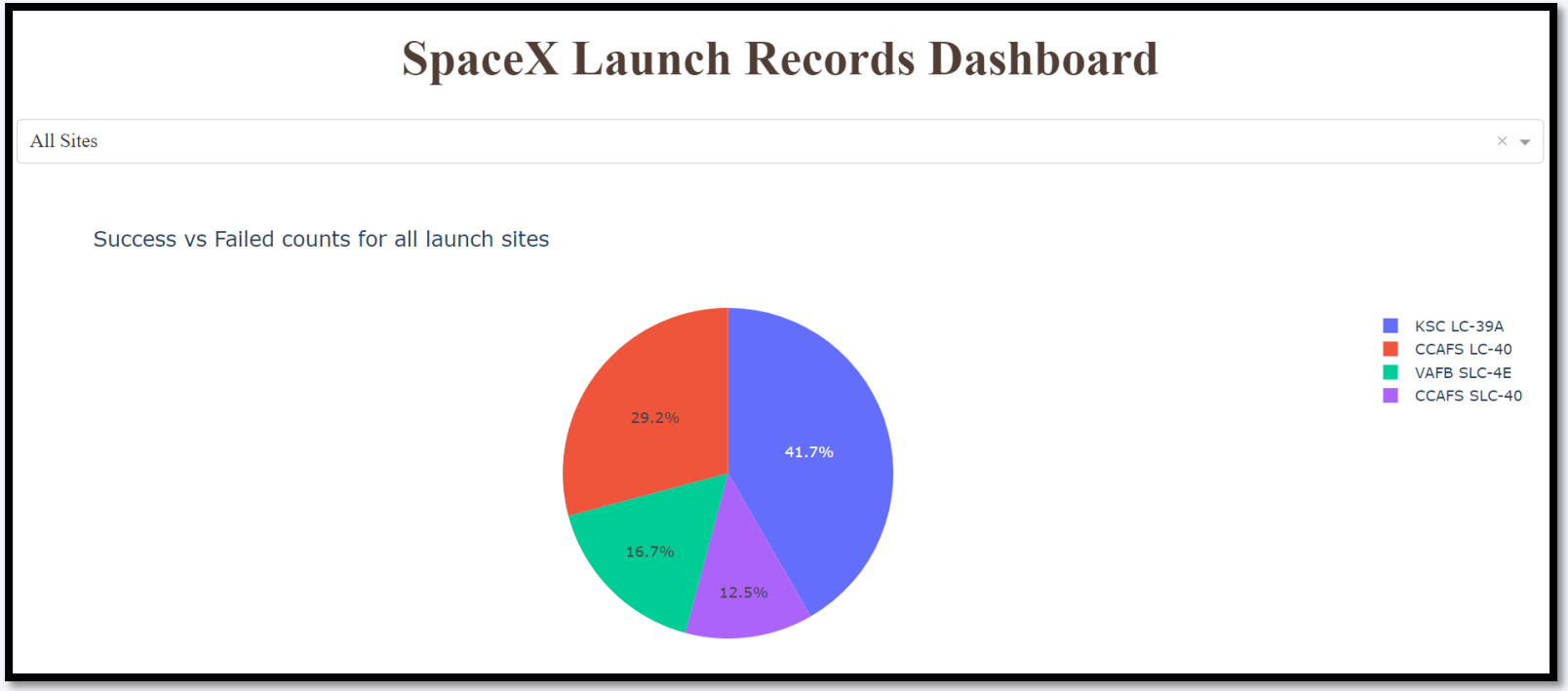
Build a Dashboard with Plotly Dash

We deployed a Dash layout with:

- A pie chart where we can select in a drop-down menu a launch location and visualize its ratio of success. In addition, we can observe the global percentage rate of success of each location too.
- In the second chart, we can delimitate the payload range and see in a scatter plot chart the number of success and non-success cases, classified by ‘booster version category’ variable.

Dashboard code – GitHub link

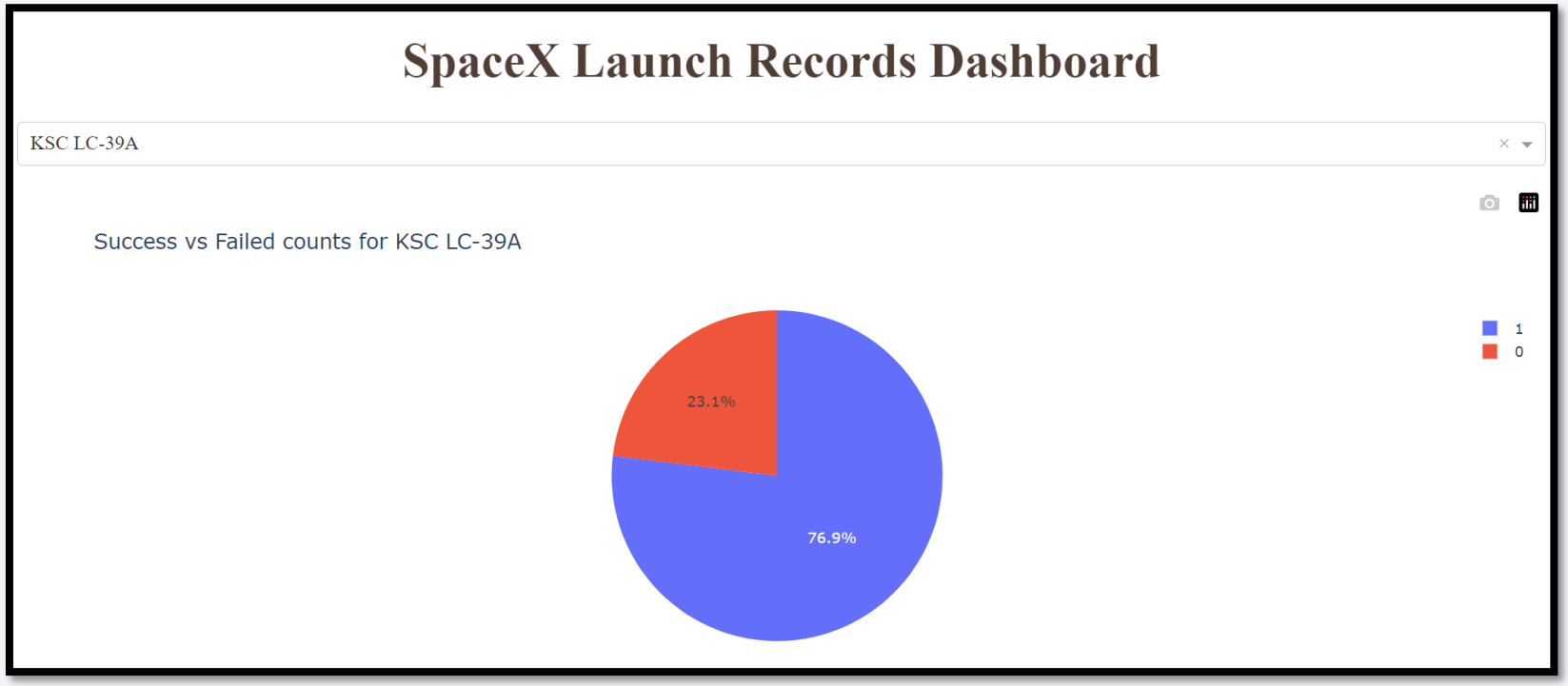
Success vs failed counts for all launch sites



In this graph, we can see that KSC LC-39A location has the most success ratio, following by CCAFS LC-40. This is the global percentage calculated of each place with respect to the total.

Dashboard code – GitHub link

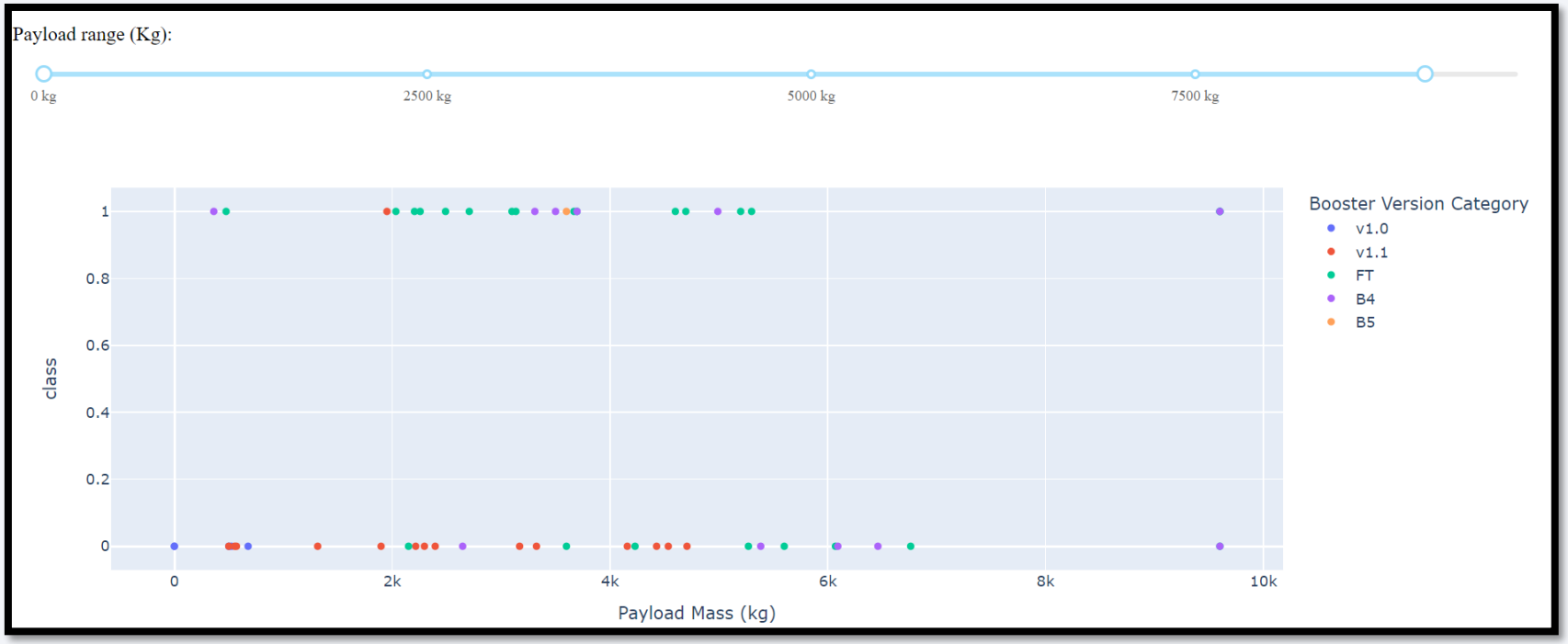
Success vs failed ratio for KSC LC-39A



Here we can see the success ratio of KSC LC-39A location with a 77%.

[Dashboard code – GitHub link]

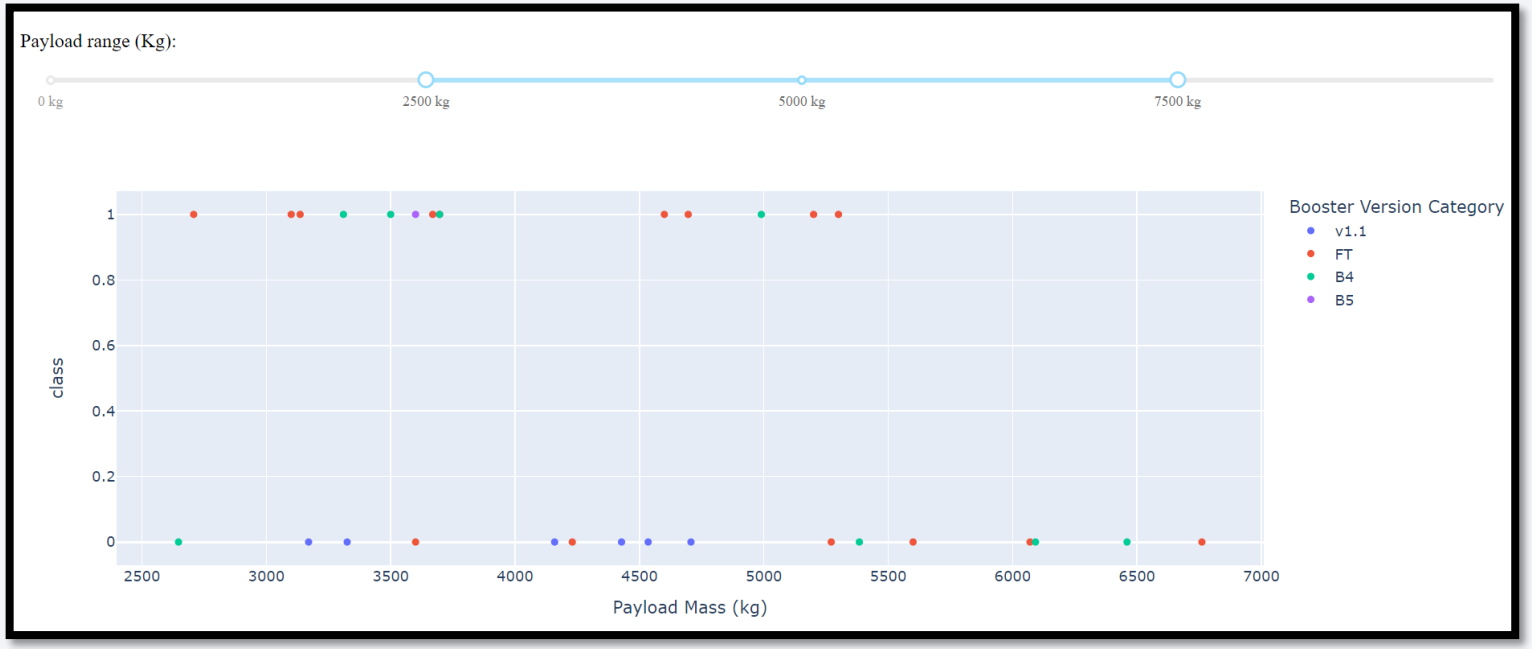
Relation between PLM and success per BVC (0-9000kg)



This chart shows the relationship between success/non-success class and PLM on a determinate range of kgs. We can see that FT booster version has the best success ratio, in contrast v1.1 the worst.

Dashboard code – GitHub link

Relation between PLM and success per BVC (2500-7500kg)



If we change the range of kgs, we can appreciate that v1.0 booster version has disappeared for this pay load mass gap. FT keep the best success ratio and v1.1 the worst.

Dashboard code – GitHub link

Predictive Analysis (Classification)

Flowchart process

1 - Scaling numerical variables and split data into training and test sets

2 - Tune of parameters of all selected algorithms and evaluate the accuracy in training and test set (also exposed the results of confusion matrix)

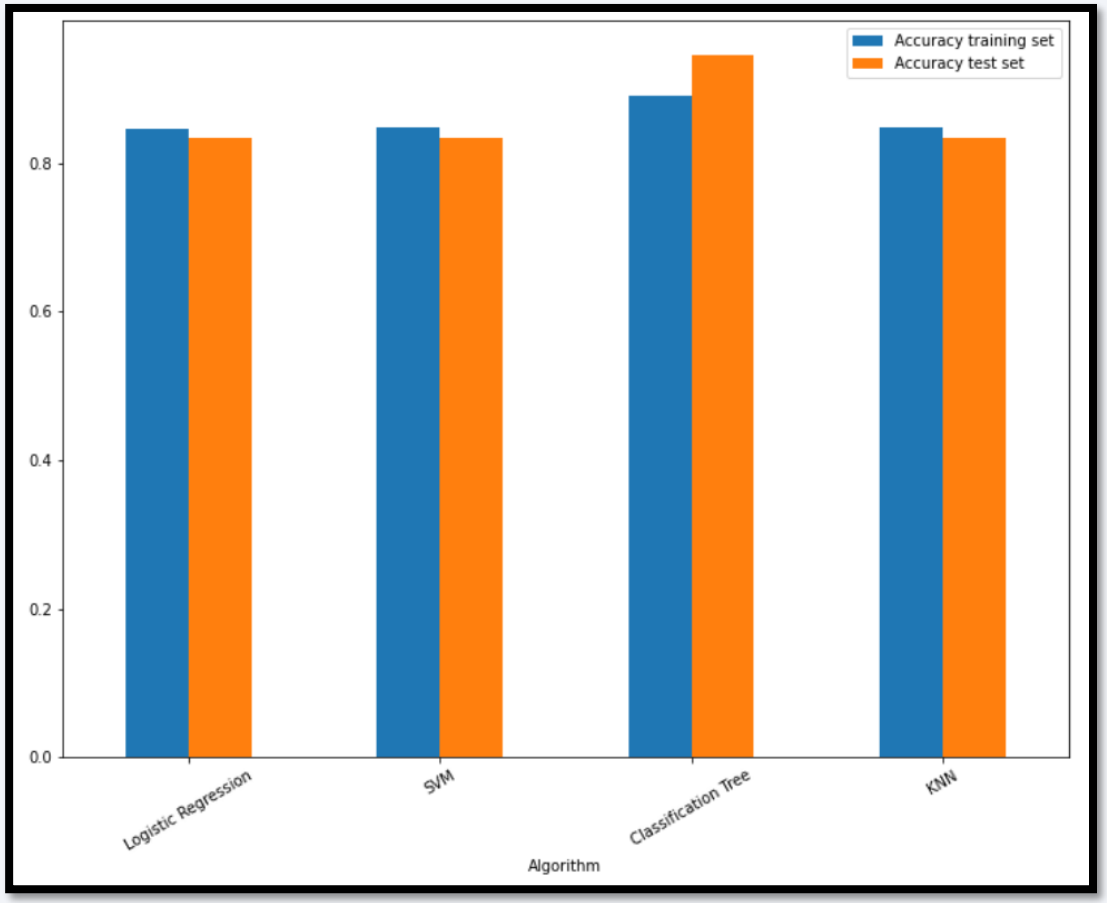
3 - Selection of the best model

ML selection model – GitHub link

Classification Accuracy

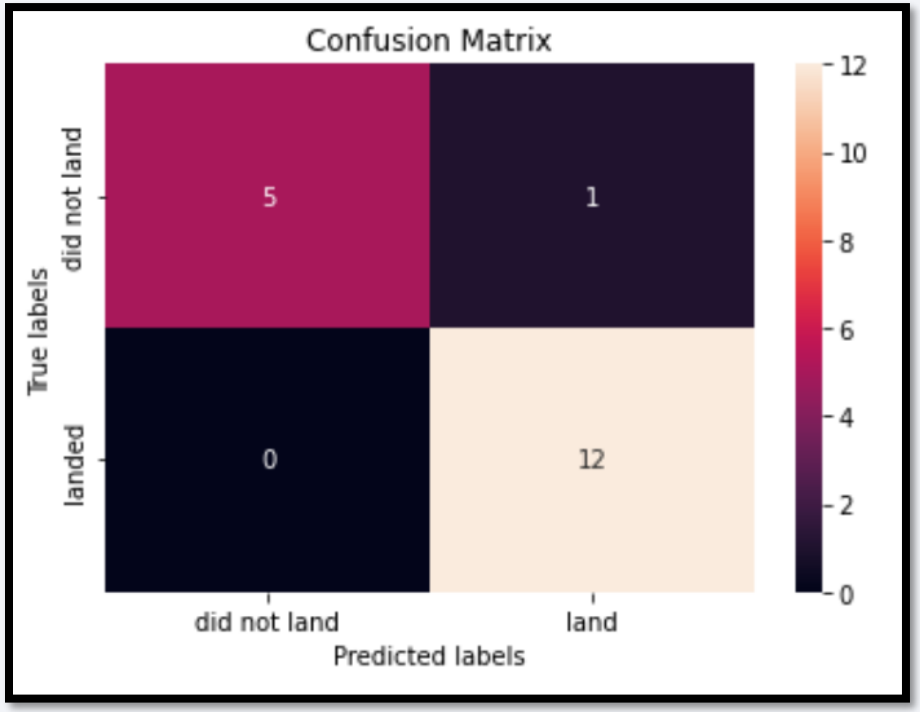
	Algorithm	Accuracy training set	Accuracy test set
0	Logistic Regression	0.846429	0.833333
1	SVM	0.848214	0.833333
2	Classification Tree	0.891071	0.944444
3	KNN	0.848214	0.833333

As we can see, classification tree algorithm has the best accuracy punctuation.



ML selection model – GitHub link

Confusion Matrix



In addition, we can appreciate in its confusion matrix that just one case was wrong classified as ‘land’ although it is actually a ‘did not land case’ (one False Positive case - FP)

ML selection model – GitHub link

Conclusions

- The location with most success ratio is KSC LC-39A location with a 77%.
- The orbit with the best success rate by number of launches (above average) and the highest pay load mass value is VLEO.
- CCAFS SLC 40 is the most used location with 55 launches.
- Till 2013 we do not have any success in our outcome.
- The best algorithm to predict if we have success in our landing is classification tree.

Thank you!