



IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Fabio Andrés Restrepo L.
23 – Aug - 2022



Outline



Executive
Summary



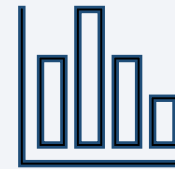
Methodology



Conclusion



Introduction



Results



Appendix

Executive Summary



Methodologies used in the analysis

- Data Collection
 - SpaceX API
 - Wikipedia Web Scraping
- Exploratory Data Analysis – EDA
 - Data Wrangling
 - Data Visualization
 - EDA with SQL
 - Interactive Visual Geospatial Analytics (Folium)
 - Dashboards (Plotly Dash)
- Machine Learning
 - Logistic Regression
 - Support Vector Machine
 - Decision Tree
 - K Nearest Neighbors

Results

- The success of the missions improved over time
- The destination orbits of the rockets help predict the outcome in some cases, but not for all of it due to the lack of data for some orbits that have just a couple of
- It took 4 years of attempts to finally achieve a successful landing (2013-2017)
- With the geospatial analysis done it isn't enough to identify a launching site for SpaceY, however we identified a couple insights, as the proximity to coastlines and railways to the launching sites, this due to the need to have access to the rockets as easy as we can to be transported from the fabrication site to the launching site
- There's a payload range where the success/failure rate are meaningful (between 1900 kg and 3700 kg)
- From the 4 models used for the prediction model the one with the best performance was the Decision Tree

Introduction



Here in SpaceY our CEO Allon Mask wants to know the viability of reusing the first stage rockets in our mission to conquer Venus.

To achieve that we are using ~~rocket~~ data science to determine the probabilities of successful landings for the first stage rockets.

Making use of public information of our competitor SpaceX and their evil CEO, Elon Musk.

We want to know the best way to estimate the cost of launching rockets to the space. How?

- By predicting successful landing of the first stage rockets.
- Finding the best place for launching the rockets.
- Identifying the relationships between all the possible variables (launching sites, rocket architecture, load characteristics) on the success of the mission.

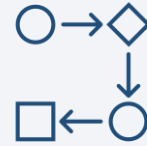


Section 1

Methodology



Methodology



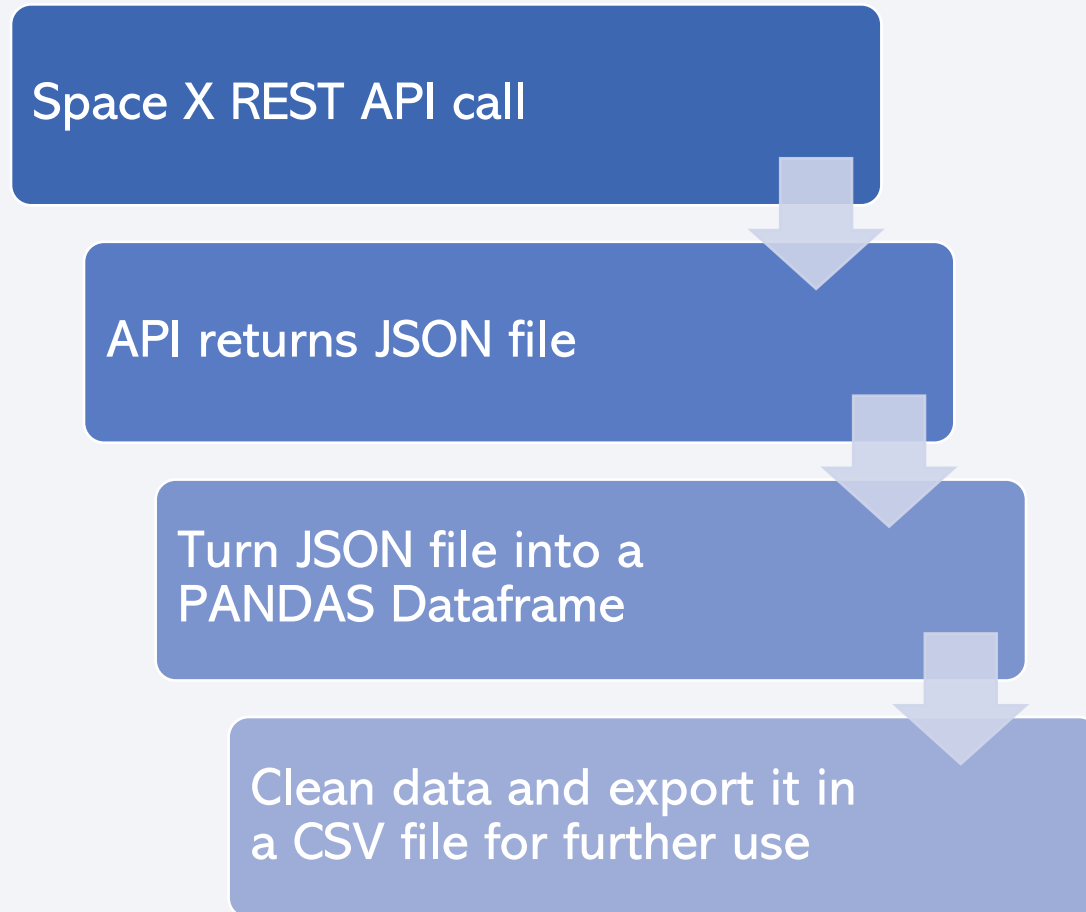
Executive Summary

- Data collection methodology:
 - From API – SpaceX API¹
 - From WebScraping - Wikipedia²
- Perform data wrangling
 - Identifying data types and missing values
 - Summarizing data
 - Creating a landing outcome label from Outcome data
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
 - Data normalized and splitted in training and testing sets
 - Evaluating the data with this models: Logistic Regression, Support Vector Machine (SVM), Decision Tree and, K Nearest Neighbors (KNN)
 - Calculating accuracies for each model for different hyperparameters and solvers

1. <https://api.spacexdata.com/v4/launches/past>

2. [https://en.wikipedia.org/wiki/List_of_Falcon/9/ and Falcon Heavy launches](https://en.wikipedia.org/wiki/List_of_Falcon/9_and_Falcon_Heavy_launches)

Data Collection – SpaceX API



The Space X REST API URL is

<https://api.spacexdata.com/v4/launches/past>

- Data requested using `requests.get()` method
- Normalized using `pd.json_normalize()` method on the `response.json()`

The information collected are rockets, cores, payloads and, launches information



Data Collection – Web Scrapping



Get HTML response from Wikipedia

Extract data from HTML using BeautifulSoup

Turn the HTML table into a PANDAS Dataframe

Clean data and export it in a CSV file for further use

The Space X REST API URL is

https://en.wikipedia.org/wiki/List_of_Falcon_9_and_Falcon_Heavy_launches

- Data requested using `requests.get()` method
- Created the BeautifulSoup object with `BeautifulSoup(response.text, 'html')` method
- Found the `<th>` element to identify the objective table

The information collected are rockets, launch sites, orbits, boosters' information

Data Collection WebScraping



Notebook

Data Wrangling

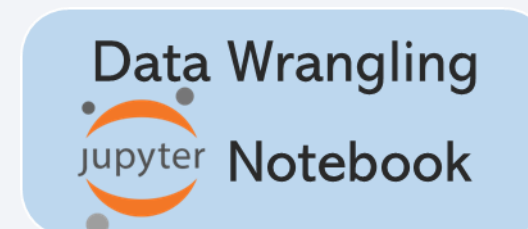


Exploratory Data
Analysis - EDA

Data
Summarization

Determine
Training Labels

- Identified all parameters data types using `.dtypes`
- Identified and calculated the % of the missing values in each attribute using `isnull()`
- Summarized 'LaunchSite', 'Orbit' and, 'Outcomes' with `.value_counts()`
- Created a landing outcome label from 'Outcome' column transforming string variables (Ocean, RTLS, ASDS being True or False) into categorical variables where 1 means True or successful landing and 0 False or a failure



EDA with Data Visualization



- To find correlation between variables I used Scatter Plots
- For relation between categoric variables I used Bar Charts
- The trends are better visualized with Line Plots



Flight Number
vs. Payload
Mass



Flight Number
vs. Launch
Site



Payload Mass
vs. Launch
Site



Success Rate
vs. Orbit



Payload vs.
Orbit Type



Orbit vs.
Payload Mass



Orbit Type vs.
Flight Number



Success Rate
vs. Year

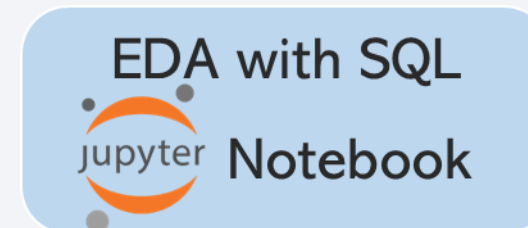


EDA with SQL



The SQL queries performed to gather and analyze data where:







- The names of the unique `launch sites` in the space mission
- 5 records where `launch sites` begin with the string `'CCA'`
- The total `payload mass` carried by boosters launched by NASA (CRS)
- Average `payload mass` carried by booster version F9 v1.1.
- The date when the first successful landing outcome in ground pad was achieved
- The names of the `boosters` which have success in drone ship and have `payload mass` between 4000 and 6000 kg
- The total number of `successful` and `failure` mission outcomes
- List of the names of the `booster versions` which have carried the maximum `payload mass`
- The `failed` landing outcomes in drone ship, their `booster versions`, and `launch site` names for the year 2015
- Rank the count of `successful` landing outcomes between the date `04-Jun-2010` and `20-Mar-2017` in descending order

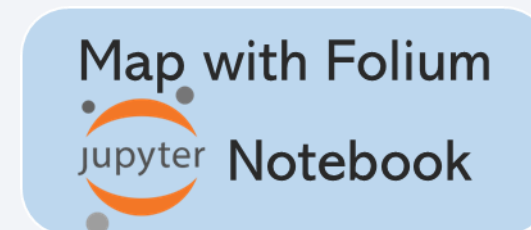


Build an Interactive Map with Folium



All objects were created to clearly understand one of the problems we have, the best launching location
With this objects on the map, we can identify the launching sites and their surroundings





- A map centered on the NASA Johnson Space Center at Houston, Texas, with a **red** circle with its name in a label using `folium.Circle`, `folium.Popup`, `folium.map.Marker` 
- A map with **reddish/orange** circles at each launching site with its name in a label using `folium.Circle`, `folium.Popup`, `folium.map.Marker`, `folium.features.DivIcon` 
- A map with a grouped cluster of success/failure indicators at each launching site, color coded (**success**, **failure**) using `folium.map.Marker`, `folium.Icon`, `marker_cluster` 
- A map with markers that indicate distance between the launching site VAFB SLC-4C in California and the nearest coastline, railway, highway and city, and a line connecting each site with the launching site in different colors using `folium.map.Marker`, `folium.features.DivIcon`, `folium.PolyLine`
  



Build a Dashboard with Plotly Dash



The Dashboard include:

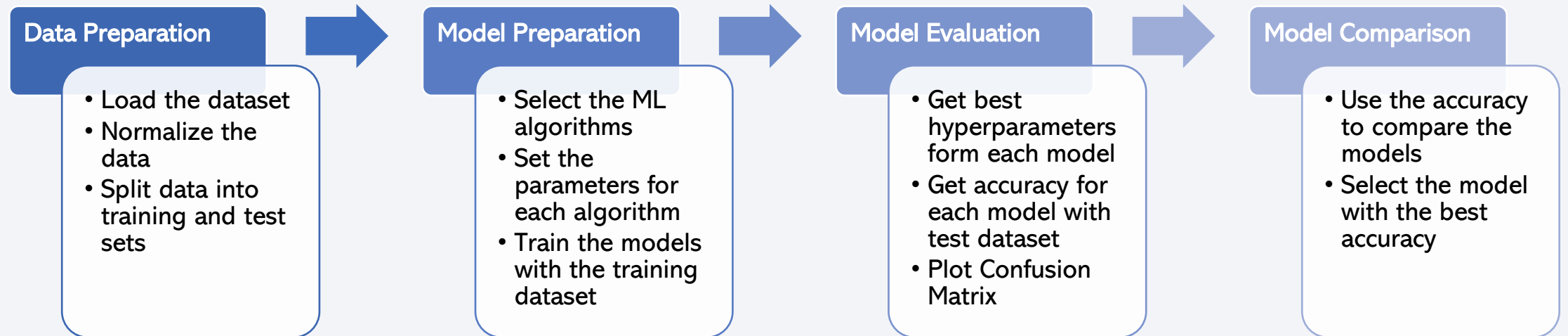
- a Dropdown menu 
- a Pie Chart 
- a Range Slider 
- a Scatter Plot 

Those elements allowed a fast and intuitive way to analyze the relations between launching sites, its successful rate and the payloads of each launch

- Dropdown menu to allow the user to choose between launching sites, made using `dash_core_components.Dropdown`
- Pie Chart to present the total success/failure rate for each or all launching sites depending on the dropdown, made using `plotly.express.pie`
- Range Slider that allows the selection of a range of payload mass, made using `dash_core_components.RangeSlider`
- Scatter Plot to show the relationship between success/failure rate and payload mass, made using `plotly.express.scatter`



Predictive Analysis (Classification)



Predictive Analysis



Notebook

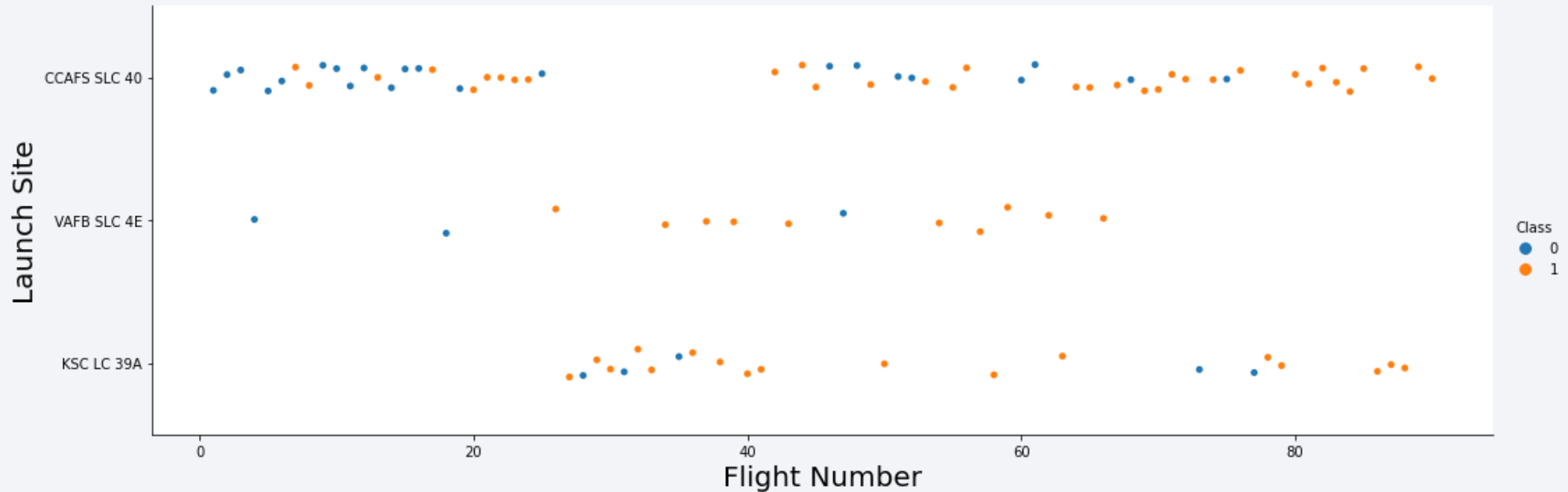
The background of the slide is an abstract composition. It features a dark blue base color. Overlaid on this are numerous diagonal streaks in shades of red and cyan. A faint, light blue grid pattern is also visible, particularly in the lower half of the image. The overall effect is dynamic and technological.

Section 2

Insights drawn from EDA

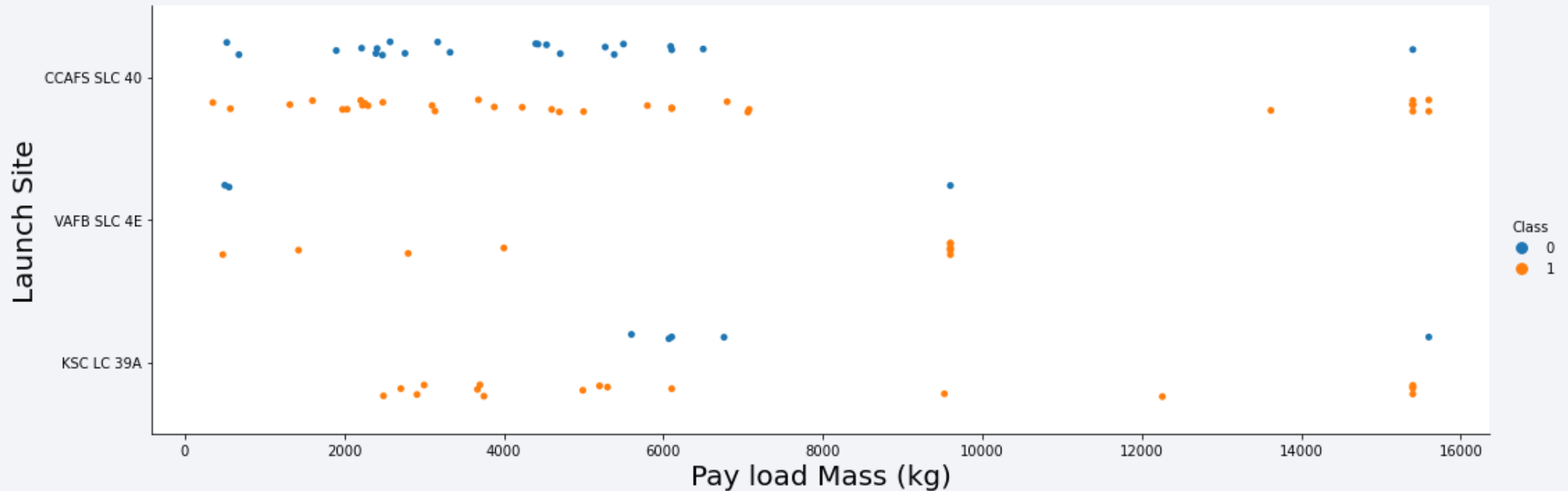


Flight Number vs. Launch Site



- The first launching site SpaceX used was CCAFS SLC 40 with low success rate at the beginning
- Launching site VAFB SLC 4E was used sporadically with a relatively good success rate
- The success rate of the KSC LC 39A launching site are kinda high along the time but it wasn't used as constant as CCAFS SLC 40
- After a while they return to use the CCAFS SLC 40 launching site, being this one the most used

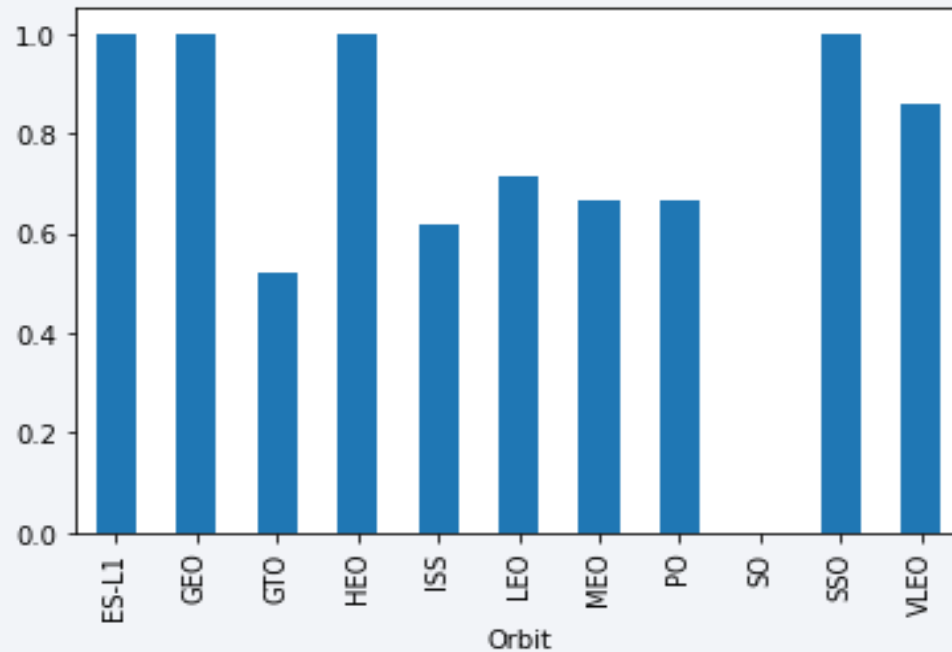
Payload vs. Launch Site



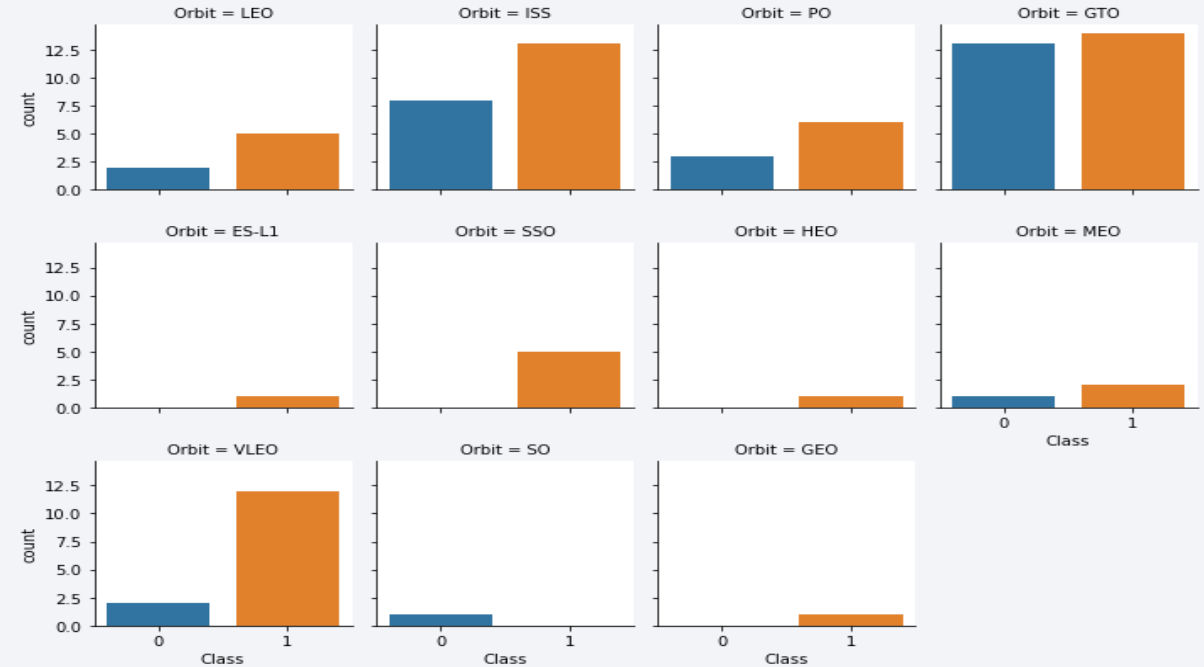
- There are no launches in VAFB-SLC launching site with payload mass above 10,000 kg
- For heavy payloads CCAFS SLC 40 launching site have a better success rate
- The KSC LC 39A launching site have a very low fail rate



Success Rate vs. Orbit Type



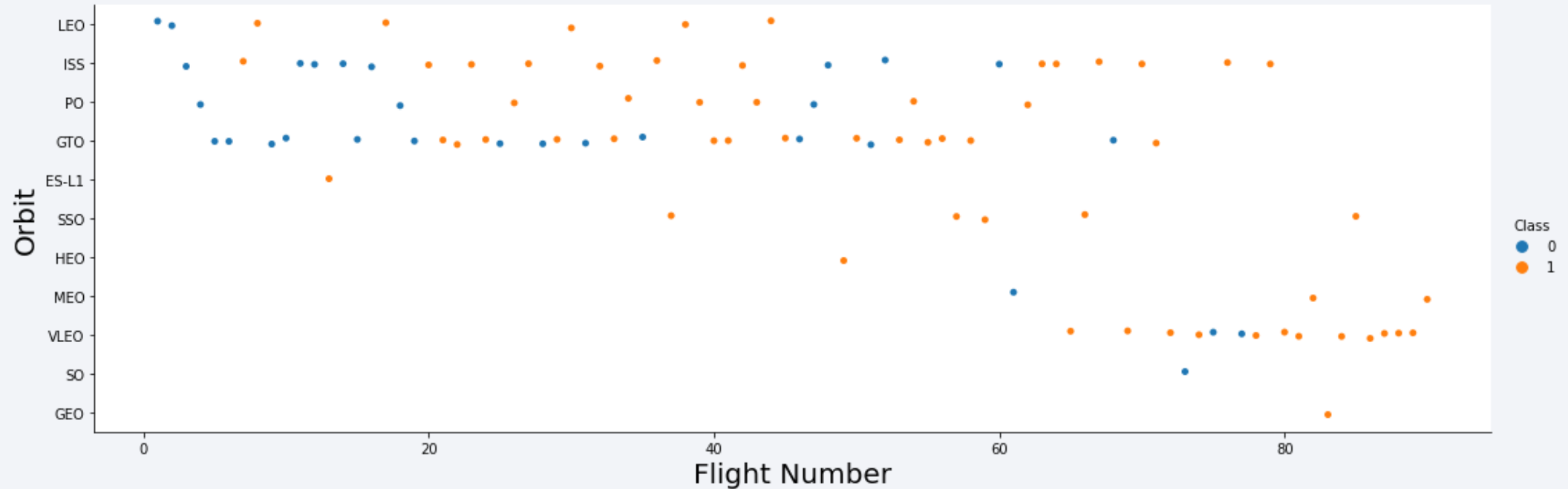
- VLEO orbit have the best ratio success/failure of all orbits
- The only launch to the SO orbit was a landing failure



- On the other hand, all the launches to the ES-L1, SSO, HEO and, GEO orbits have successful landings.



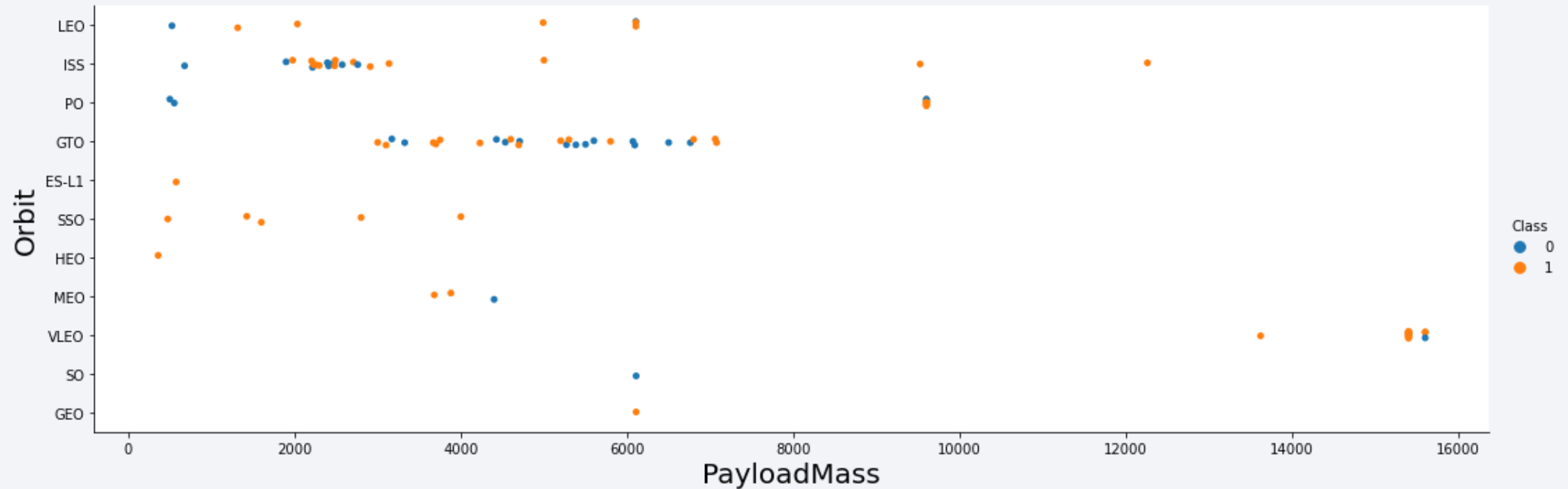
Flight Number vs. Orbit Type



- At the beginning the SSO, HEO, MEO, VLEO, SO and, GEO orbits wasn't destinations for the rockets
- The success for the launches to the LEO, MEO and, VLEO improve with the time
- The rockets with SSO orbit destination are the best in launching performance
- The launches with destination GTO orbit have the more erratic landing behavior



Payload vs. Orbit Type

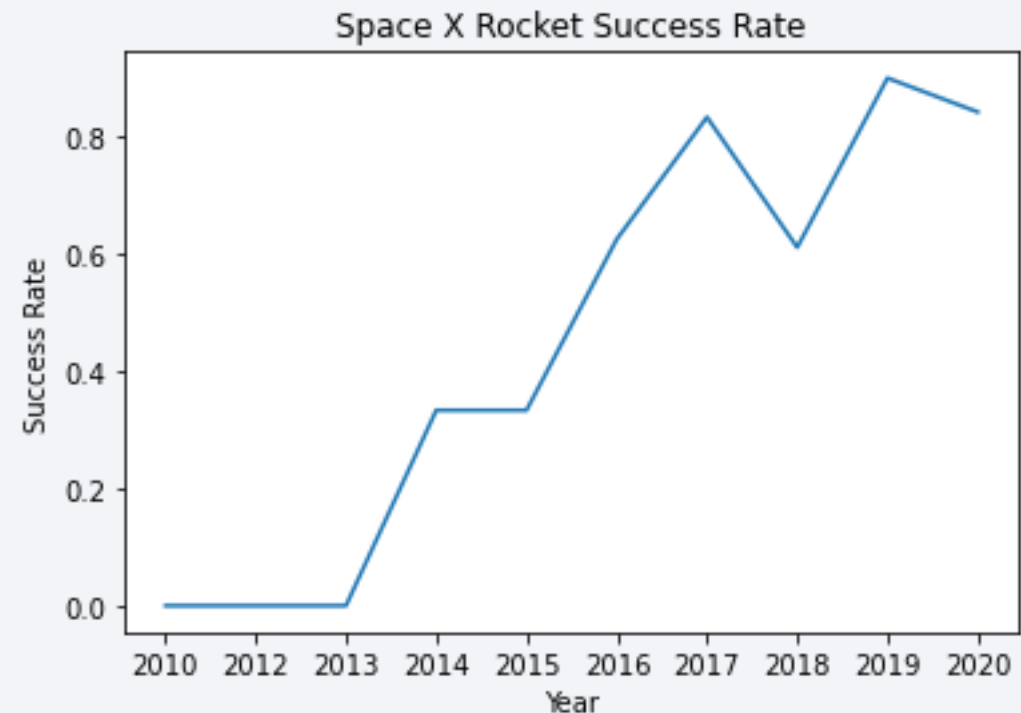


- The rockets with LEO, ISS and SSO destination improve the landing success rate when they was carrying heavier payload mass
- The launches to ES-L1, HEO, MEO, VLEO, SO, GEO and, PO have non recognizable pattern respecting the payload mass
- The GTO orbit destination are the one with the unidentifiable rate of success/failure along the payload mass range.



Launch Success Yearly Trend

- The first three years was full of failure landings
- This trend changed in 2014 when the success rate start improving and keep growing until 2017
- 2018 cope with a reduction of almost 20% in the successful landings
- In the last 2 years of data the successful rate stayed around 85%





All Launch Site Names

There're five launching sites

SQL Query

```
SELECT DISTINCT LAUNCH_SITE  
FROM SPACEXTBL;
```

Used `DISTINCT` to remove
LAUNCH_SITE duplicates

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



Launch Site Names Begin with 'CCA'

SQL Query

```
SELECT * FROM SPACEXTBL WHERE  
LAUNCH_SITE LIKE 'CCA%' LIMIT  
5;
```

Used `WHERE ... LIKE` clause to filter the launch sites that have `'CCA%'`
And `LIMIT 5` to just show the first 5 rows of the query

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

SQL Query

```
SELECT SUM(PAYLOAD_MASS__KG_)
AS Total_Payload FROM
SPACEXTBL WHERE CUSTOMER =
'NASA (CRS) ' ;
```

Used SUM to totalize the payload mass

The AS to give a name to the result

And WHERE ... = to define the launches
belonging to the NASA

Total_Payload
45596



Average Payload Mass by F9 v1.1

SQL Query

```
SELECT AVG (PAYLOAD_MASS__KG_)
AS "AVG_Payload_by_F9_v1.1"
FROM SPACEXTBL WHERE
BOOSTER_VERSION = 'F9 v1.1';
```

Used `AVG` to average the payload mass

The `AS` to give a name to the result

And `WHERE ... =` to define the launches
were made with a specific booster

AVG_Payload_by_F9_v1.1
2928.4



First Successful Ground Landing Date

SQL Query

```
SELECT MIN (DATE) AS  
First_Ground_Pad_Success FROM  
SPACEXTBL WHERE "Landing  
_Outcome" = 'Success (ground  
pad) ' ;
```

Used MIN to find the first value in DATE
The AS to give a name to the result
And WHERE ... = to define the launches
were made with a specific booster

First_Ground_Pad_Success
01-05-2017



Successful Drone Ship Landing with Payload between 4000 and 6000

SQL Query

```
SELECT DISTINCT BOOSTER_VERSION  
FROM SPACEXTBL WHERE "Landing  
_Outcome" = 'Success (drone  
ship)' \  
AND PAYLOAD_MASS__KG_ BETWEEN  
4000 AND 6000;
```

Used `DISTINCT` to remove
BOOSTER_VERSION duplicates
`WHERE ... = ... AND` to define the
conditions of the query
And `BETWEEN # AND #` to define the
range of the second condition

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



Total Number of Successful and Failure Mission Outcomes

SQL Query

```
SELECT (SELECT  
COUNT(MISSION_OUTCOME) FROM  
SPACEXTBL WHERE MISSION_OUTCOME  
LIKE '%Success%') AS Success,  
(SELECT COUNT(MISSION_OUTCOME)  
FROM SPACEXTBL WHERE  
MISSION_OUTCOME LIKE  
'%Failure%') AS Failure;
```

Is a nested query

Used `SELECT COUNT` to count the number
of values that meet the condition

`WHERE ... LIKE` to define the condition
using similarity in the values

The `AS` to give a name to the result

Success	Failure
100	1



Boosters Carried Maximum Payload

SQL Query

```
SELECT DISTINCT BOOSTER_VERSION  
FROM SPACEXTBL WHERE  
PAYLOAD_MASS__KG_ = (SELECT  
MAX(PAYLOAD_MASS__KG_) FROM  
SPACEXTBL) ORDER BY  
BOOSTER_VERSION;
```

Is a nested query

Used **DISTINCT** to remove
BOOSTER_VERSION duplicates

Used **MAX** to find the maximum value in
Payload Mass column

And **ORDER BY** to sort the result values

Booster_Version

F9 B5 B1048.4

F9 B5 B1048.5

F9 B5 B1049.4

F9 B5 B1049.5

F9 B5 B1049.7

F9 B5 B1051.3

F9 B5 B1051.4

F9 B5 B1051.6

F9 B5 B1056.4

F9 B5 B1058.3

F9 B5 B1060.2

F9 B5 B1060.3



2015 Launch Records

SQL Query

```
SELECT substr(DATE, 4, 2) AS  
Month, BOOSTER_VERSION,  
LAUNCH_SITE FROM SPACEXTBL WHERE  
"LANDING_OUTCOME" = 'Failure  
(drone ship)' AND substr(DATE,  
7, 4) = '2015';
```

Used `AS` to give a name to the result

`WHERE ... = ... AND` to define the
conditions of the query

And `substr(DATE, #, #)`

To define months and year, as SQLite
doesn't support date names

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

SQL Query

```
SELECT "LANDING _OUTCOME",  
COUNT (*) AS QTY FROM SPACEXTBL  
WHERE DATE >= '04-06-2010' AND  
DATE <= '20-03-2017' AND "LANDING  
_OUTCOME" LIKE '%Success%' GROUP  
BY "LANDING _OUTCOME" ORDER BY  
COUNT ("LANDING _OUTCOME") DESC;
```

Used COUNT (*) to count the values that meet the condition

AS to give a name to the result

WHERE ... = ... AND ... LIKE to define the conditions of the query

GROUP BY for grouping the COUNT (*) clause

And ORDER BY ... DESC to order in descending order the result

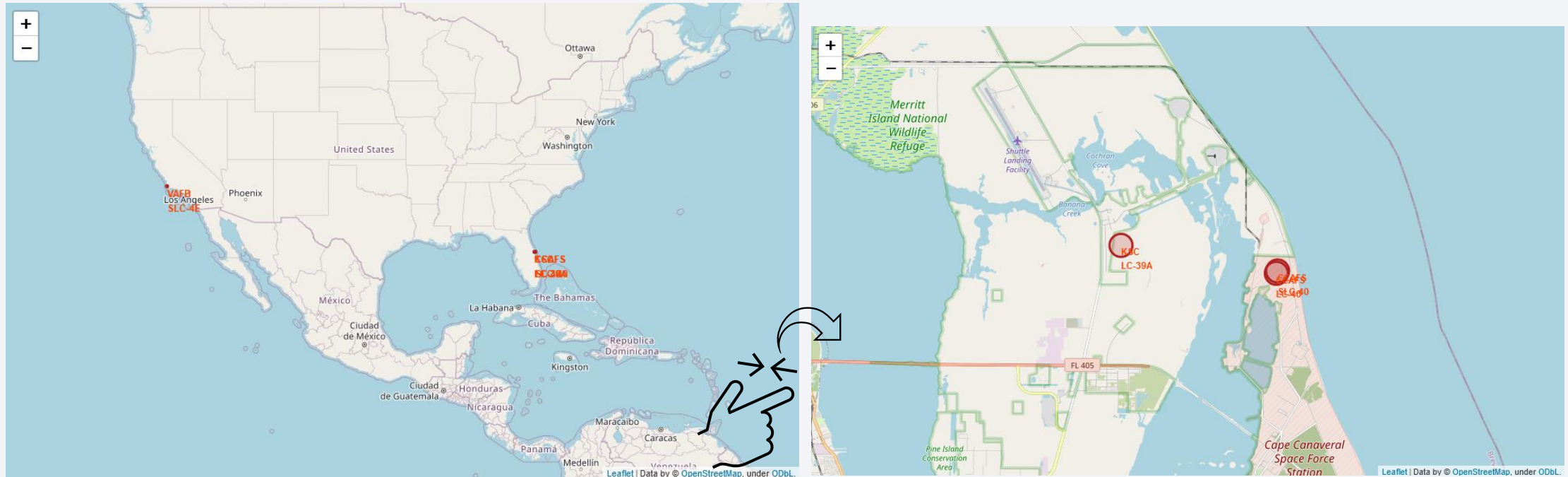
Landing_Outcome	QTY
Success	20
Success (drone ship)	8
Success (ground pad)	6

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

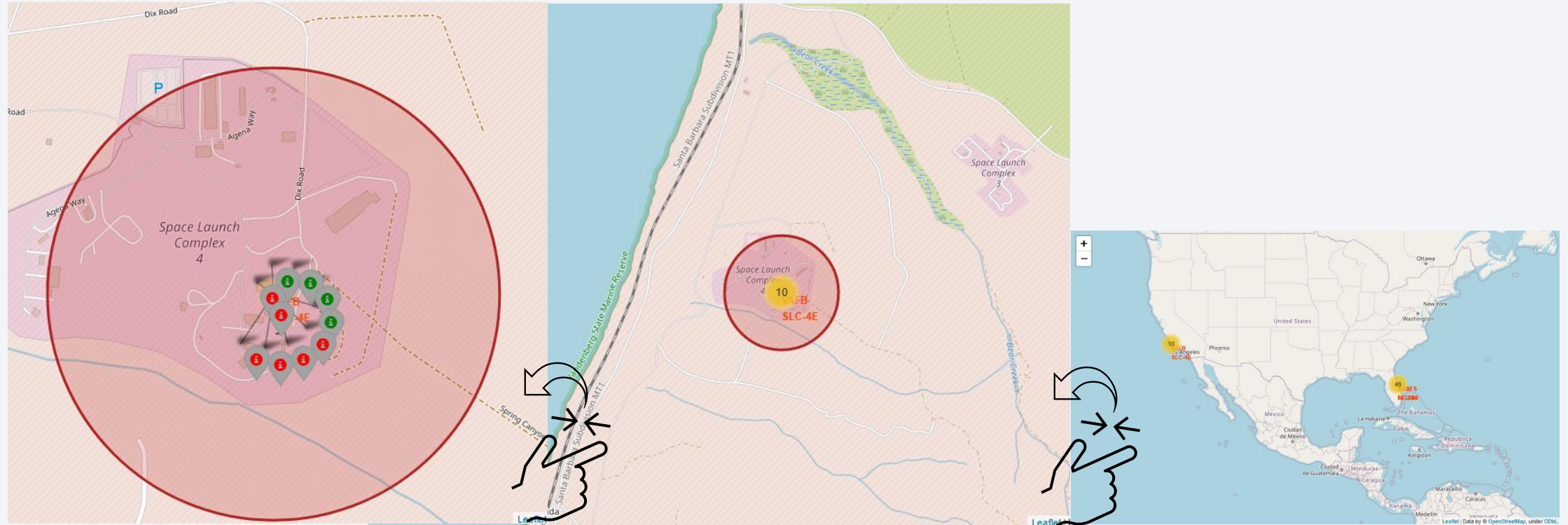
Launch Sites Proximities Analysis

Geospatial Analysis – Launching Sites



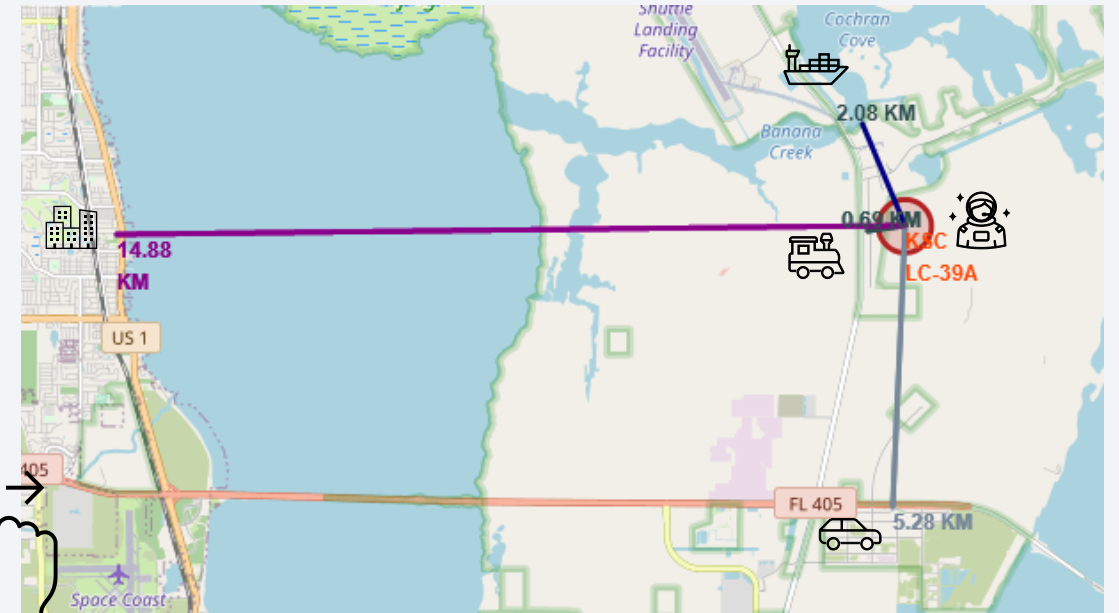
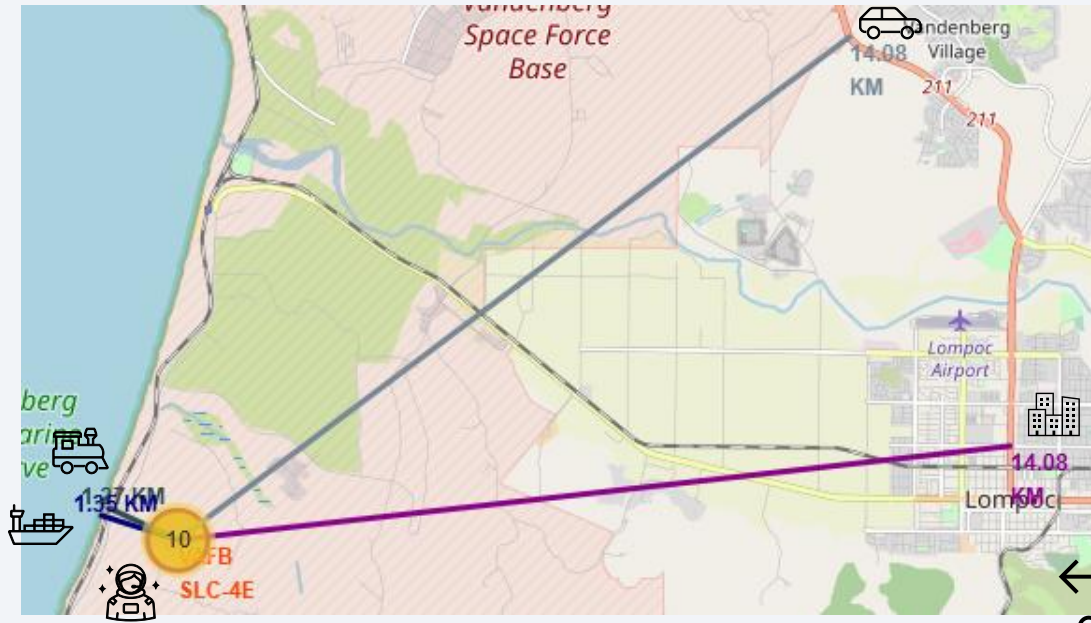
- The map on the left show the launching sites of the SpaceX program
- All the launching sites are by the coastline
- On the right side the Florida launching sites are more recognizable after zooming in

Missions' Outcomes Clusters



- In the center map we can find the VAFB SLC-4E launching site in California
- When clicked it displays color coded markers that indicates if the launch had **successful** or **failure** landings, as show in the left map

Proximities Analysis – Safety and Logistics



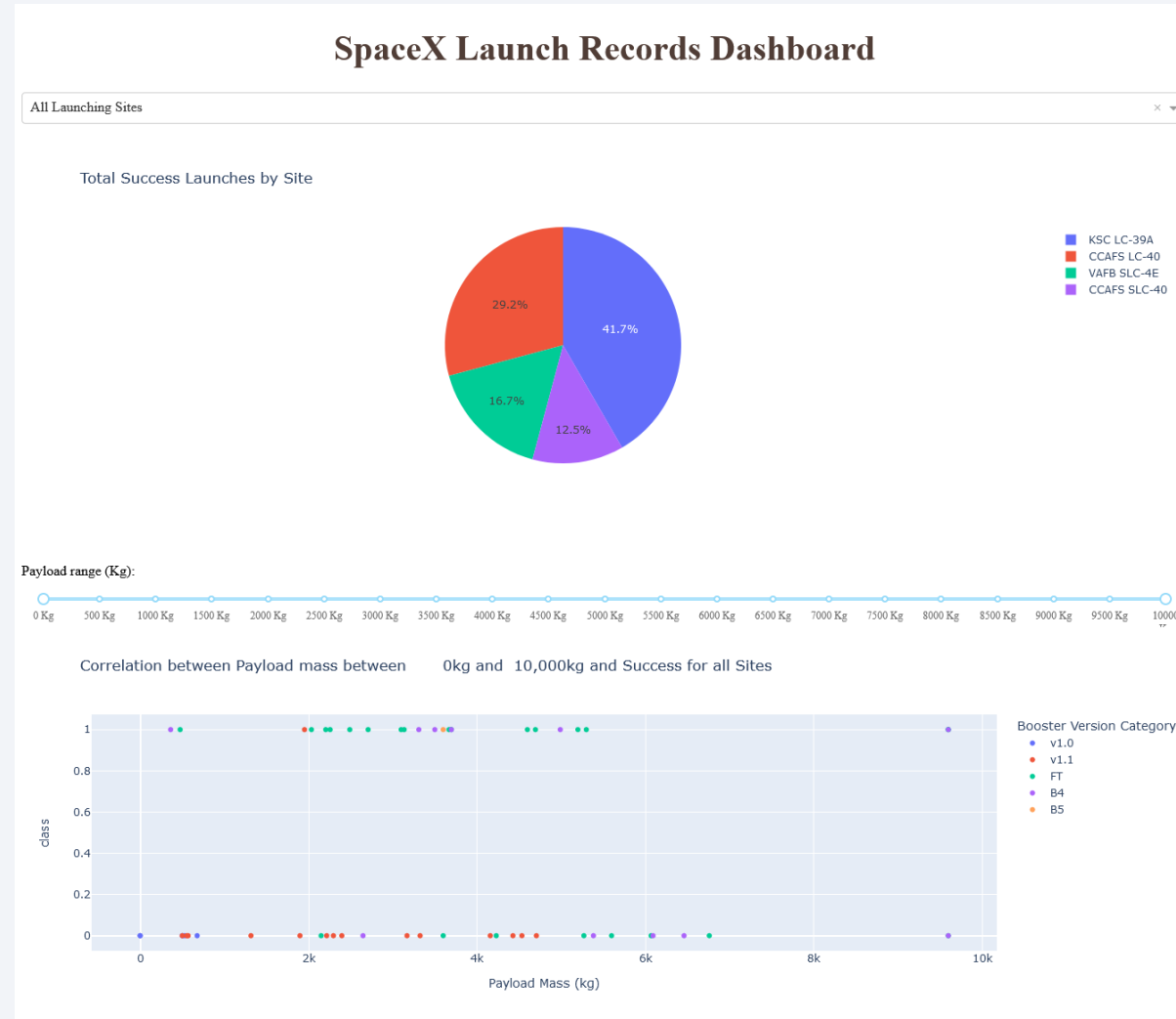
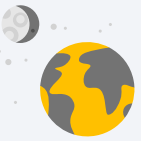
- In the left map we can see VAFB SLC-4E launching site in California and in the right side KSC LC-39A in Florida
- For logistical reasons it is necessary that the launch sites are close to the coastlines or railways to facilitate the transportation of the equipment from the manufacturing site or from the landing site.
- In the other hand the nearest cities (Lompoc in the left map and Titusville in the right map) are more than 14km away, I guess it's for security reasons.
- The same applies to the highways that are in this case more than 2km away from the launching sites.



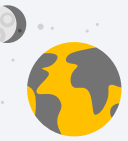
Section 4

Build a Dashboard with Plotly Dash

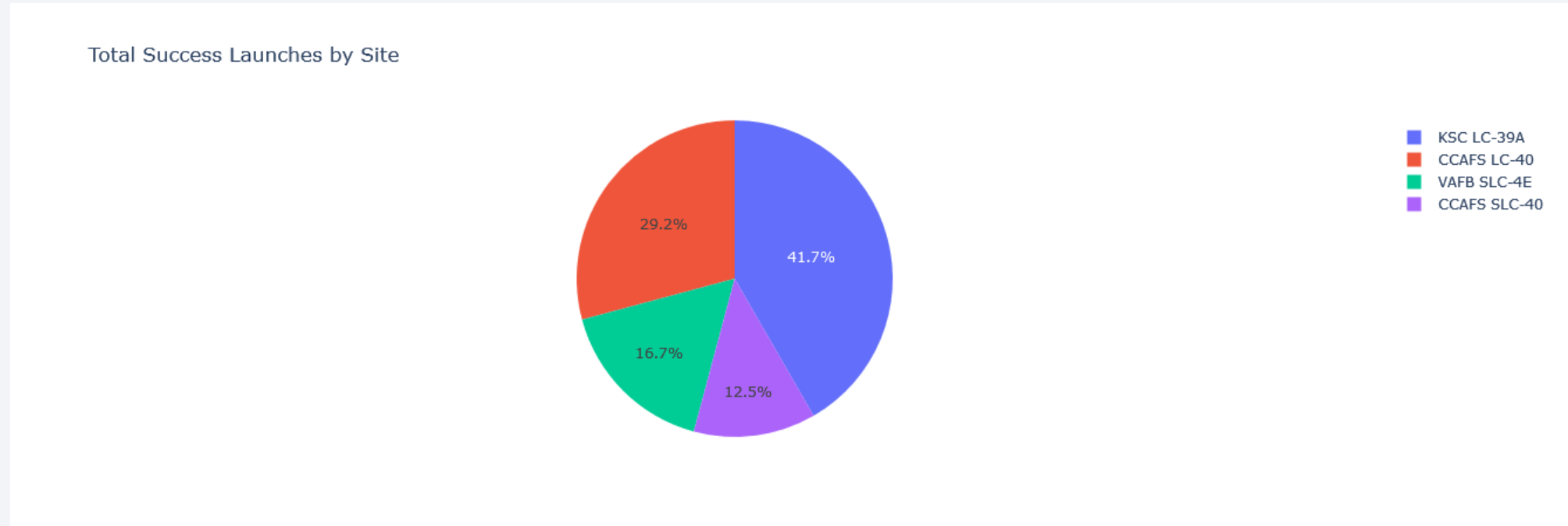
Plotly Dash Dashboard



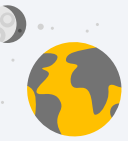
- The dropdown menu and the slider make the dashboard interactive
- It shows a pie chart and a scatter plot
- With this dashboard we can tweak here and there and identify insights like the ones show in next slides



Success Landing Rate by Launching

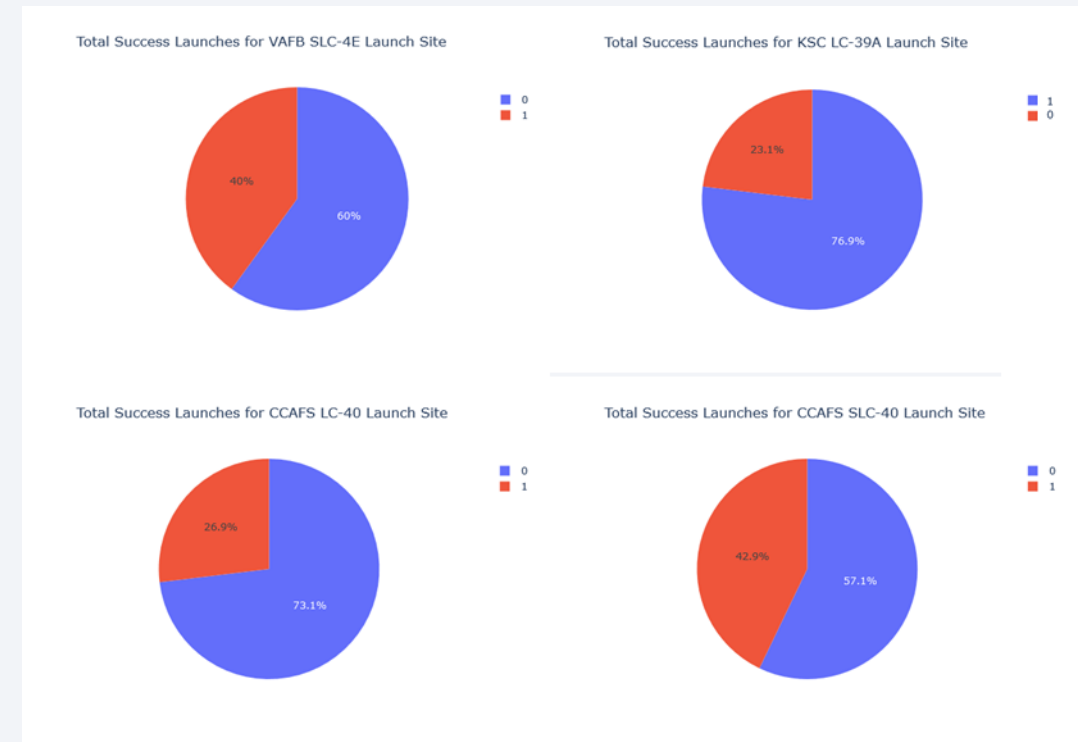


- For the four (4) launching sites the one with the best performance in the missions of the rockets launched there are the KSC LC-39A in Florida
- Considering that three (3) of four (4) launching sites are in Florida and one (1) in California it is not possible to affirm that the location of the launching site are a key factor to the possibility of a successful mission.



Success Landing Rate by Launching

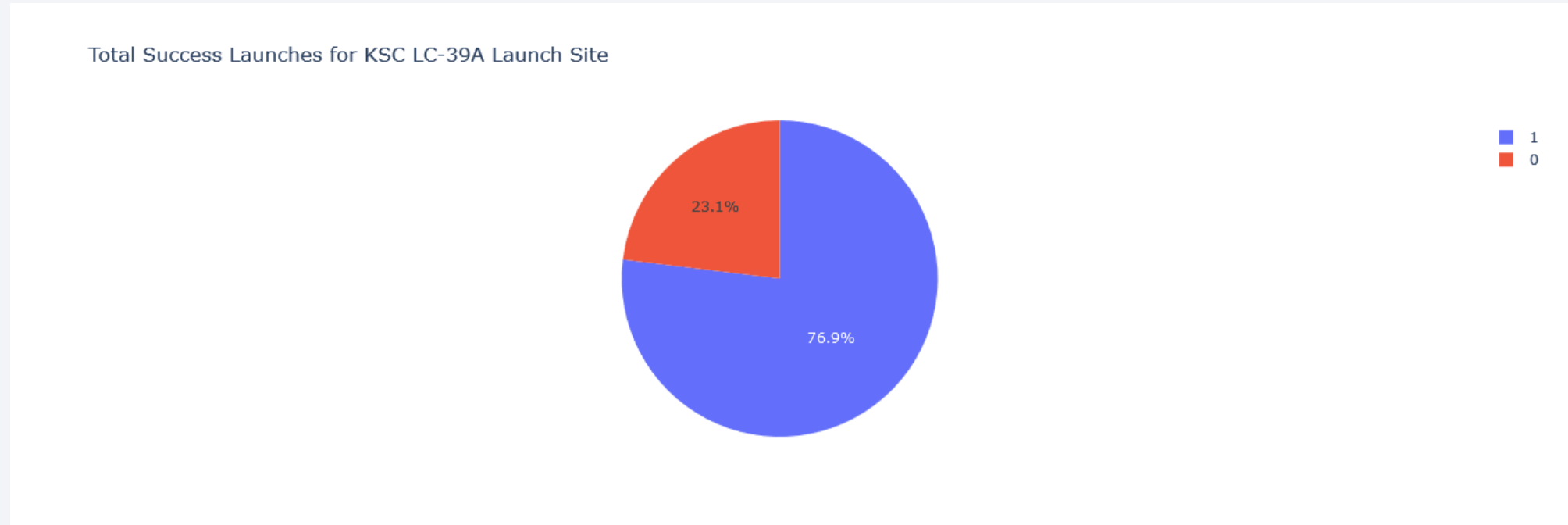
- One by one, CCAFS LC-40 launching site are the site with the lowest success rate
- Whereas that VAFB SLC-4E and CCAFS SLC-40 have a similar rate
- All these three sites have less than a half of it launches as a success



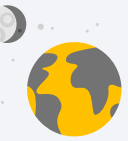
Note: Look closely each legend, in the case of the KSC LC-39A site the colors are inverted, why? Idk, Dash show these way



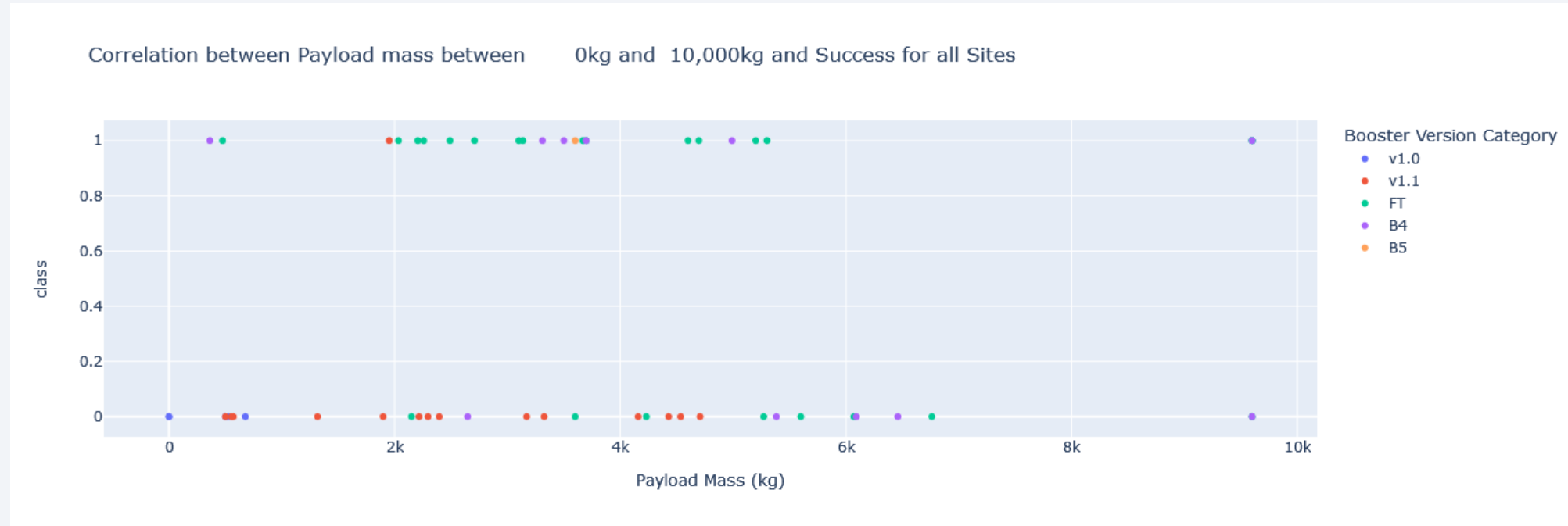
Success/Failure Ratio for KSC LC-39A



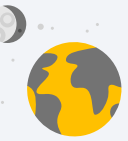
- As mentioned in the previous slide, KSC LC-39A are the site with the best performance among all the launching sites
- This launching site have a success/failure rate of above three out of four ($\frac{3}{4}$) successful mission of the total launches



Payload Launch Success Rates



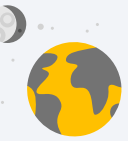
- Here we can see that the FT booster are the one with the best performance
- Likewise, the v1.1 booster are the version with the most failure missions



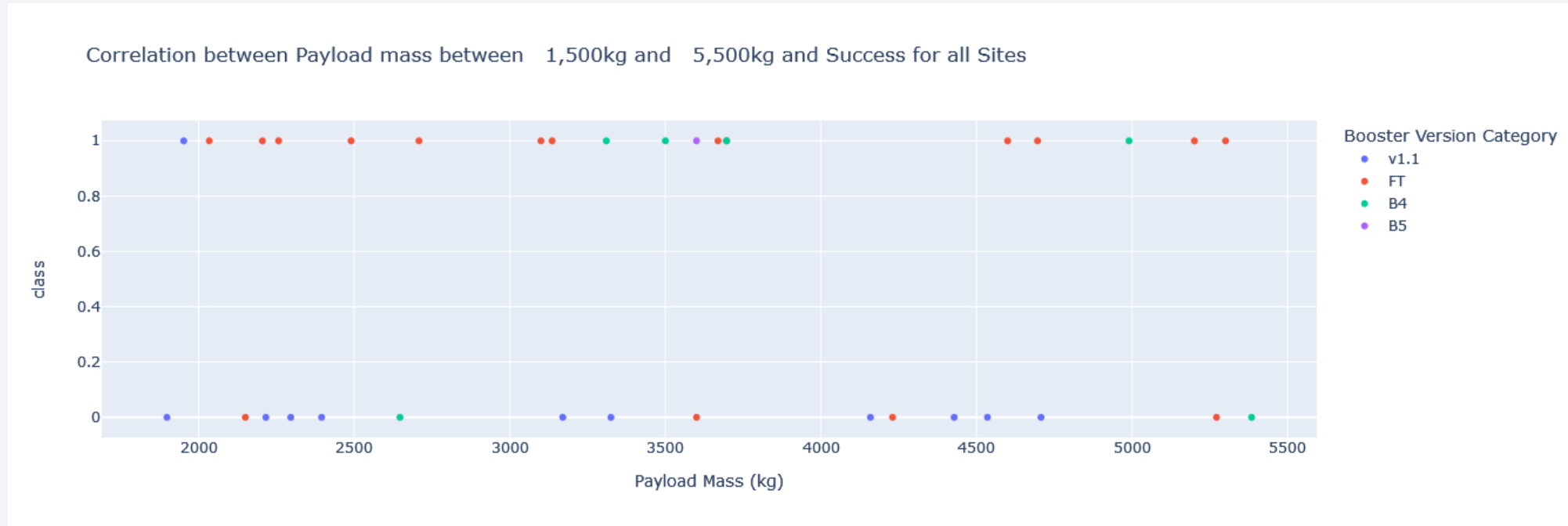
Payload Launch Success Rates



- The payload mass range with the lowest success rate are between 500 kg (about 1102.31 lb) and 750 kg (about 1653.46 lb), where we can find that 5 launches were a failure in a small payload range



Payload Launch Success Rates



- The payload mass range with the highest success rate are between 1900 kg (about 4188.78 lb) and 3700 kg (about 8157.09 lb), where almost the 60% of the successful launches are

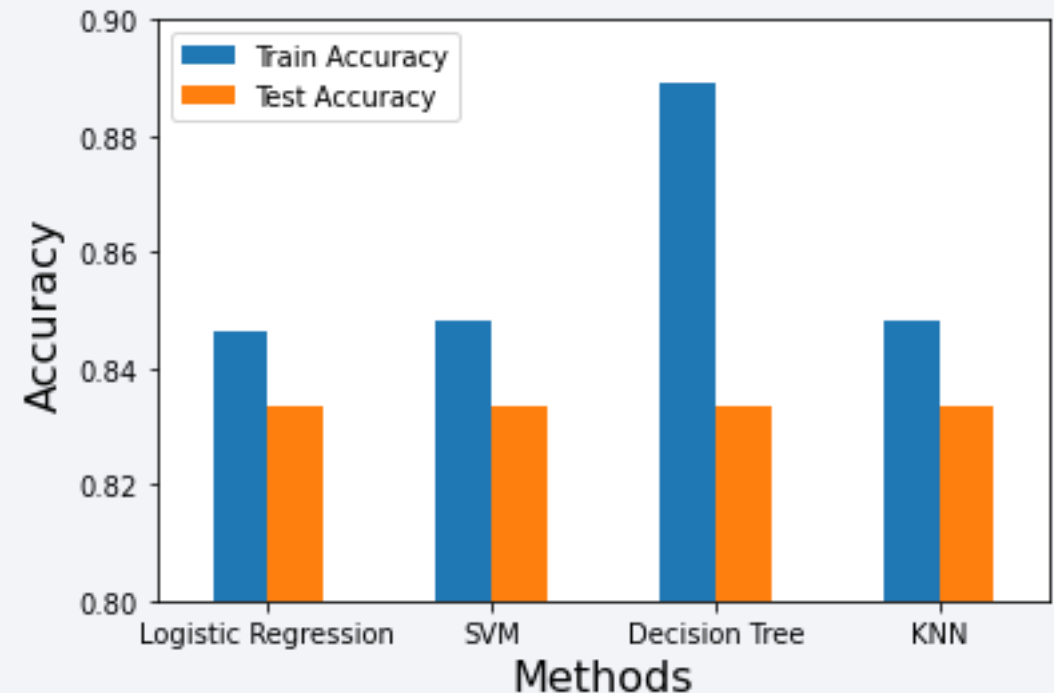
Section 5

Predictive Analysis (Classification)

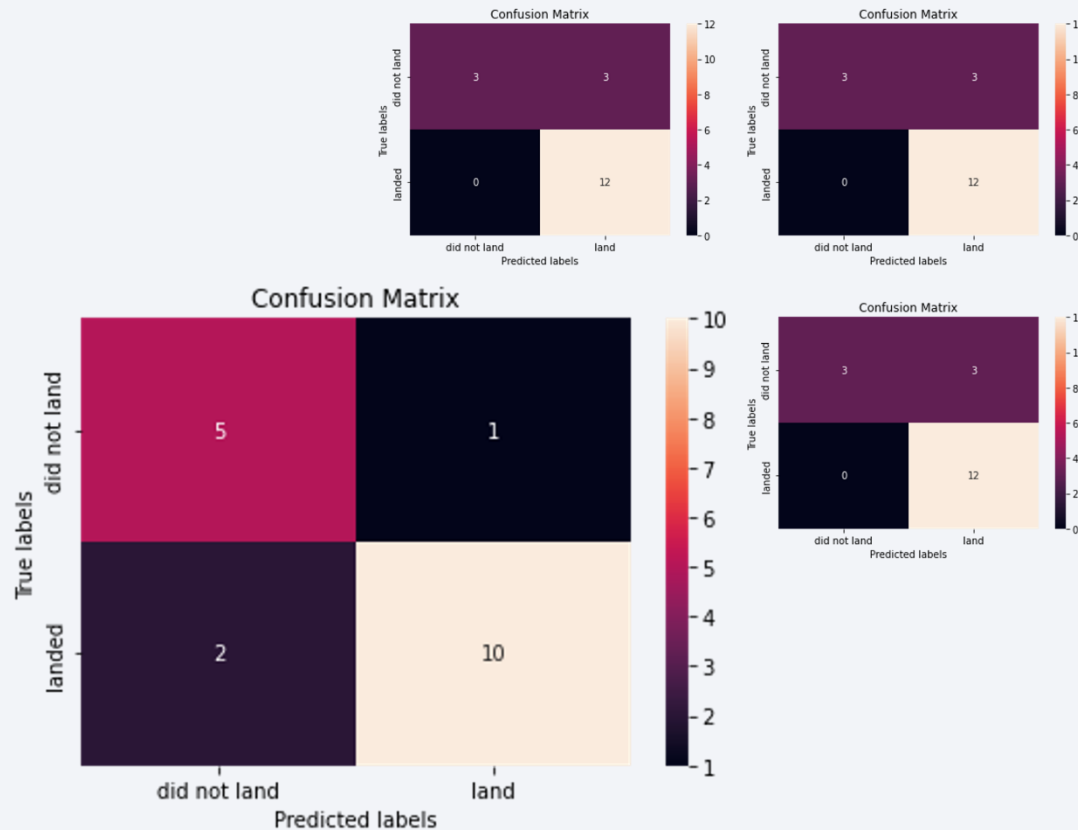
Classification Accuracy

- The accuracies for the models in the both set are show in the table below
- All the methods performs the same in the **test set**
- The best accuracy in the **train test** of the models analyzed are the **Decision Tree** with 88%

	Train Accuracy	Test Accuracy
Logistic Regression	0.846429	0.833333
SVM	0.848214	0.833333
Decision Tree	0.889286	0.833333
KNN	0.848214	0.833333



Confusion Matrix



- For Logistic Regression, Support Vector Machine (SVM) and, K Nearest Neighbors (KNN) methods the Confusion Matrix (the 3 smalls) are the same showing that the **true positives** affect to a greater level the accuracy of these models.
- On the other hand, the Confusion Matrix for the Decision Tree (bottom left) have better values for the **false positives** and the **false negatives**; likewise, the **true positives** and the **true negatives** values are OK

Conclusions

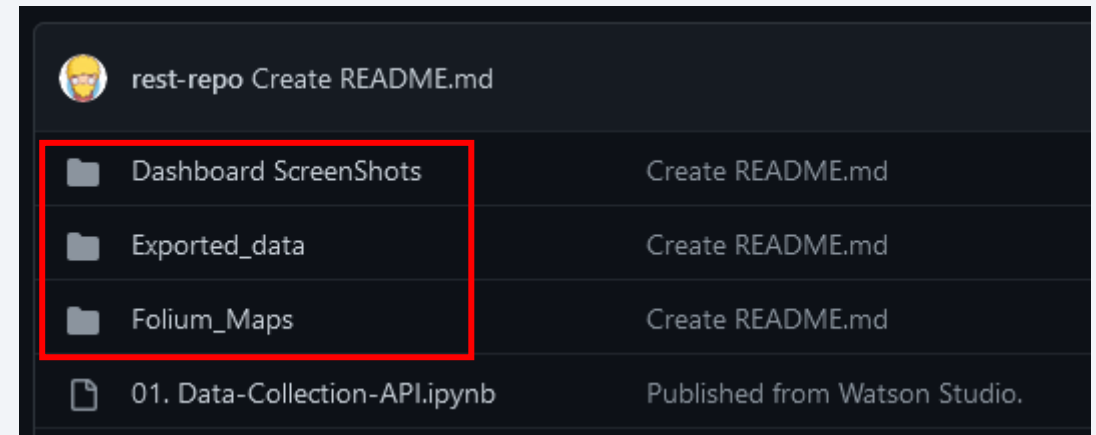


- We can forecast the success of a launching/landing mission considering a wide array of parameters as all this analysis shows
- The success of the missions improved over time
- The payload mass aren't a decisive and clear parameter using visualization to predict the outcome of a mission as analyzed in this exercise
- The destination orbits of the rockets help predict the outcome in some cases, but not for all of it due to the lack of data for some orbits that have just a couple of mission
- The correlation between orbits and payload need to be further analyzed, because the visual one wasn't decisive
- SpaceX has an interesting learning curve proved in the yearly trend
- It took 4 years of attempts to finally achieve a successful landing (2013-2017)
- With the geospatial analysis done it isn't enough to identify a launching site for SpaceY, however we identified a couple insights, as the proximity to coastlines and railways to the launching sites, this due to the need to have access to the rockets as easy as we can to be transported from the fabrication site to the launching site
- There's a payload range where the success/failure rate are meaningful (between 1900 kg and 3700 kg)
- From the 4 models used for the prediction model the one with the best performance was the Decision Three
- The analysis carried out in this exercise wasn't enough to identify the weight of each parameter in the predicting model (maybe a Bayesian model could be more accurate)

Appendix



- In the GitHub repo you can find all the CSV exported data of each step
- Screenshots from the Dashboard as well can be found in the repo
- Considering is possible you can't see the maps from the geospatial analysis because of the trustworthy or not trustworthy notebook in the repo can be found screenshots for the maps



Thank you!

