IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Federico
2022/01/28

Outline

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

Executive Summary

- In this course, data regarding SpaceX launches were analyzed, in order to predict the probability of success of a certain space mission in the future.
- Data were formerly collected via API from SpaceX website and via web scraping from Wikipedia, and then merged. Data were explored to find out the most important features and possible correlations between them, in order to set up a predictive Machine Learning (ML) algorithm. Data were then organized in a remote database and read by a sql package for python.
- A dashboard has been created to visualize both location-based data and scattered/pie-chart data to check potential correlations.
- Lastly, data were exploited to train a ML algorithm with different methodologies and for different parameters, running each algorithm on a grid search. Results were visualized on confusion matrices and a comparison has been done between the various training methodologies.

Introduction

The idea of this project come from the need to optimize the success of space launches, starting from data. To date, analysis and prediction of some events or phenomena were performed by creating a physical-based model and testing the model on available data (often few data).

Recently, with the increasing availability of data, many problems can be dealt with by deriving the model of behavior of a certain phenomenon exploiting the data and the result of some events [1] (even in physics [2]). By correctly tuning the model, it can be possible to predict whether an event will be successful or not.

In this project, we make use of data collected on many space launches from SpaceX to predict, via machine learning techniques, if the next launch will be more likely to succeed or to fail, based on the launch parameters.

1) Data-driven modeling and learning in science and engineering, *Comptes Rendus Mécanique*, **347**, 11 (2019) https://doi.org/10.1016/j.crme.2019.11.009.
2) The rise of data-driven modelling. *Nat Rev Phys* **3,** 383 (2021). https://doi.org/10.1038/s42254-021-00336-z

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - Data were collected by means of SpaceX API and web scraping from Wikipedia, to integrate lacking data from SpaceX.
- Perform data wrangling
  - Data were pre-processed with the package pandas for python. Data were grouped by orbit type, by landing outcome, and the number of launches per each launch site was derived. These operations were needed to determine the training labels for the machine learning algorithms.
- 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 were standardized, split into train and test sets and used to train several machine learning algorithms, namely Logistic Regression, Support Vector Machine (SVM), Decision Tree Classifier and K-nearest Neighbour.

Data Collection

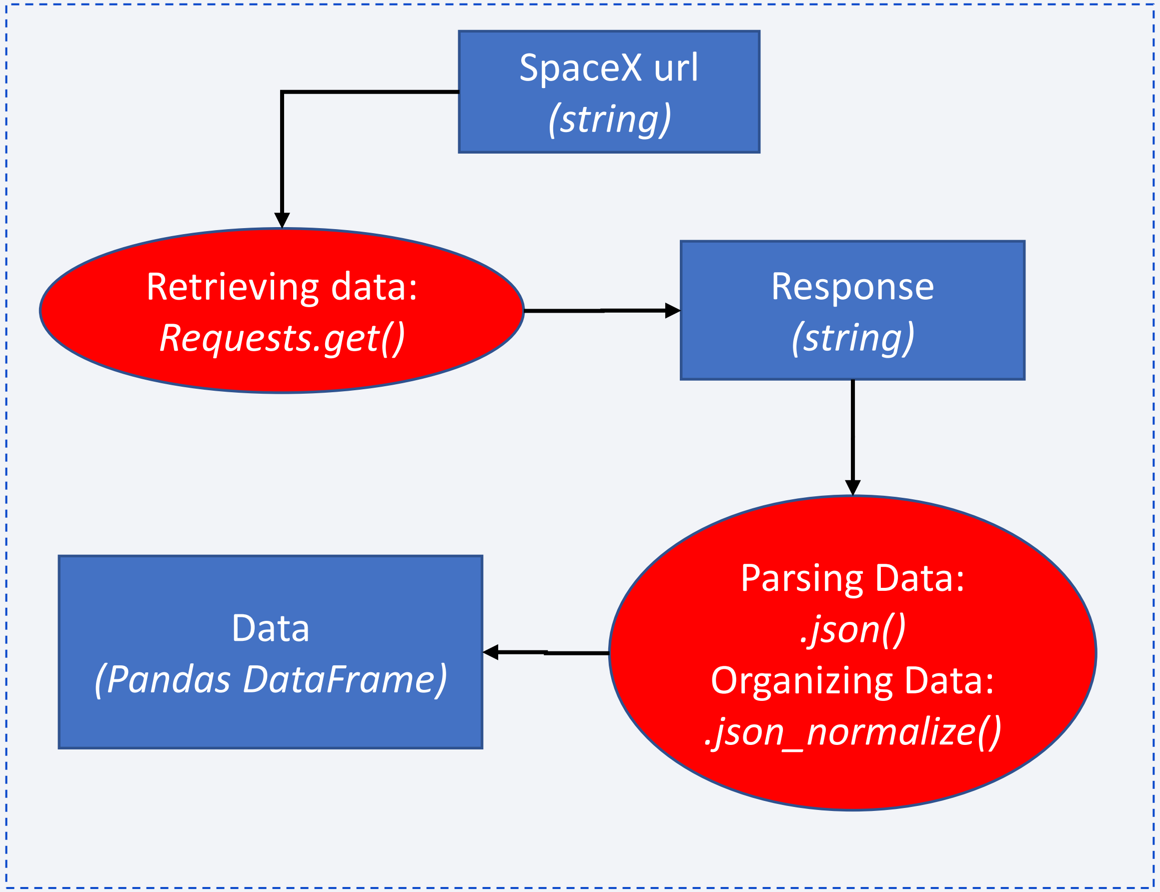
- Data were collected in three different ways:

  - From the **SpaceX API**, by using *requests* library to communicate with the API and retrieve the wanted data.
  - From a **static json url** from coursera server, by using *requests* library, and organizing the response data into a dataframe.
  - From the **Falcon9 Wikipedia page**, by using *requests* library as well.

Data Collection – SpaceX API

- The flowchart on the right represents the operations needed to retrieve and organize data in an understandable structure, as a DataFrame.

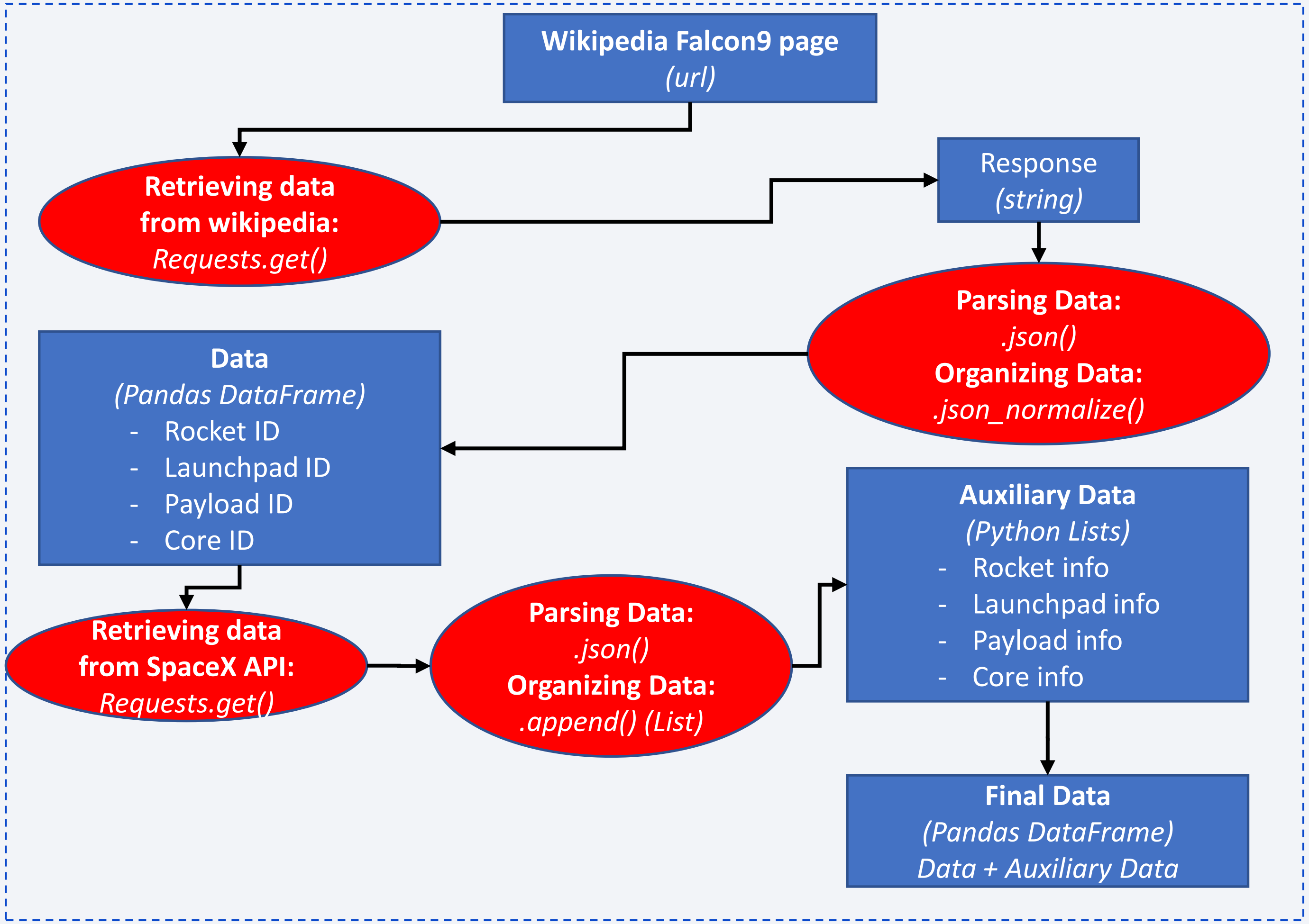
- This is the link to the github page of the related notebook: https://github.com/pignatta/Coursera_Capstone/blob/master/Data%20Collection%20API.ipynb



Data Collection - Scraping

- More in details, data were initially retrieved from the Wikipedia Falcon9 page and organized in a dataframe. Many columns only report IDs, which in turn can be passed as further requests to the SpaceX API. The new data collected are then added to the former DataFrame.

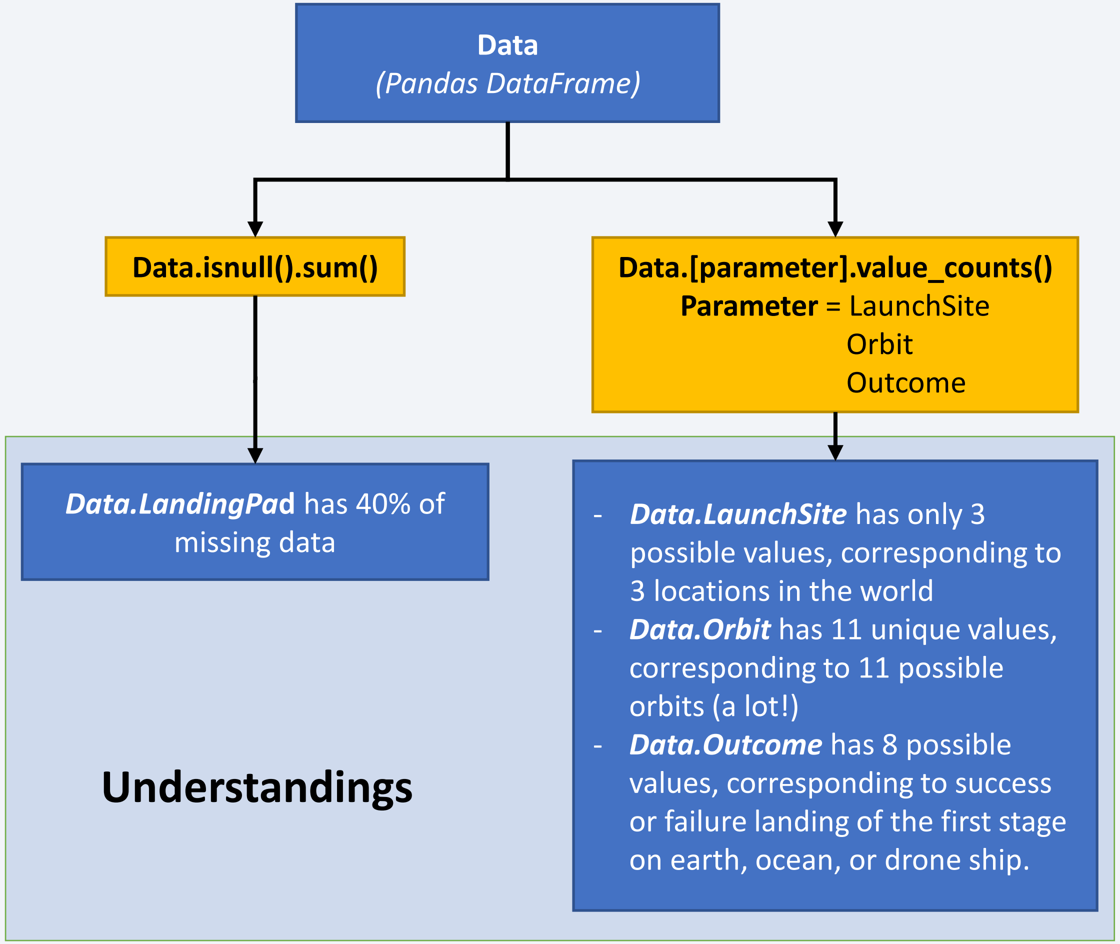
- This is the link to the github page of the related notebook: https://github.com/pignatta/Coursera_Capstone/blob/master/Data%20Collection%20API.ipynb



Data Wrangling

- Once data are organized in an ordered structure, we need to check the quality of data, like the presence of *NaN* or *Null* cells.

- First operations on data can be calculating the occurrence per each column, as the number of launches per each launch site, or the occurrences of orbits among all the launches.



Data
(Pandas DataFrame)

Data.isnull().sum()

Data.[parameter].value_counts()
Parameter = LaunchSite
Orbit
Outcome

Data.LandingPad has 40% of missing data

Understandings

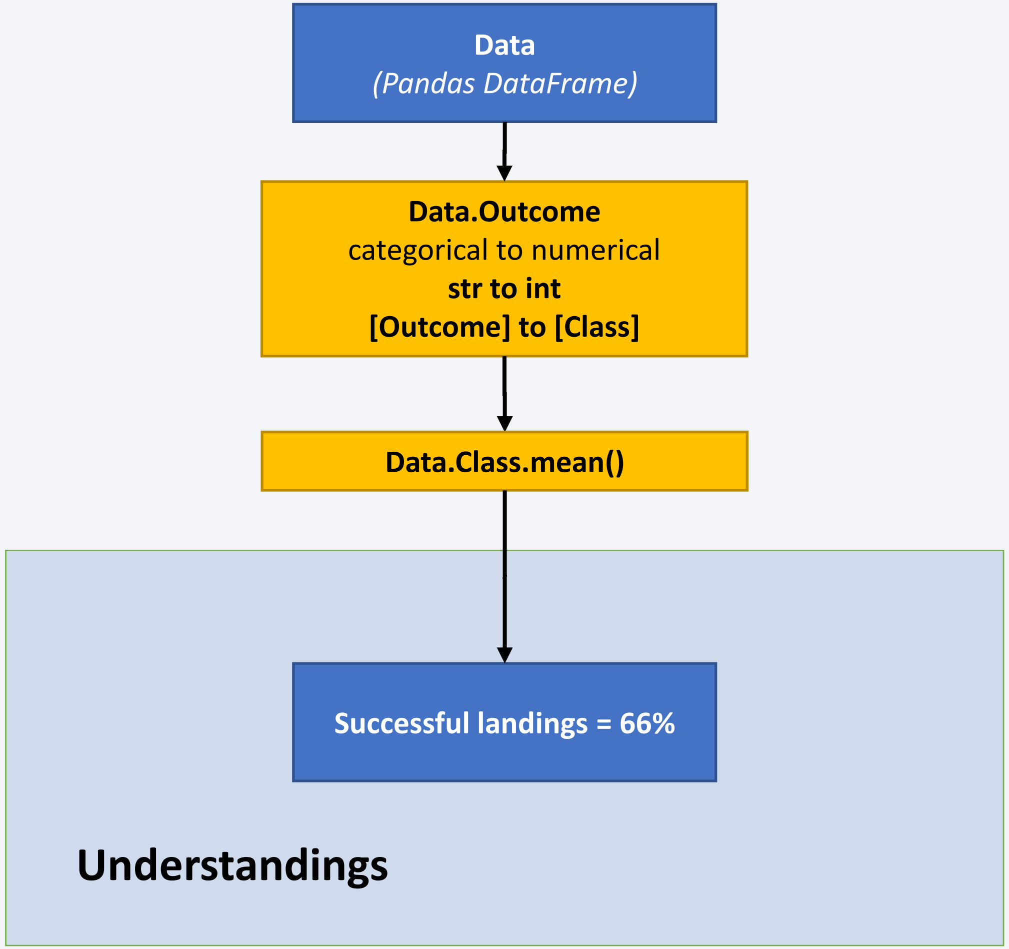
- ***Data.LaunchSite*** has only 3 possible values, corresponding to 3 locations in the world
- ***Data.Orbit*** has 11 unique values, corresponding to 11 possible orbits (a lot!)
- ***Data.Outcome*** has 8 possible values, corresponding to success or failure landing of the first stage on earth, ocean, or drone ship.

Data Wrangling

- Mission Outcomes are organized in a set of possibilities, but we want this parameter to be numerical, rather than categorical.

- Now we can also calculate the mean rate of success of first stage landed.

- This is the link to the github page of the related notebook: https://github.com/pignatta/Coursera_Capstone/blob/master/Data_Wrangling.ipynb



EDA with Data Visualization

Once data are organized as we want, several charts were plotted to gain insights.

- **FlightNumber vs PayloadMass**: It is important to understand if the latter missions are more likely to success, rather than the first missions. Moreover, we can understand if there is correlation between the payload mass and the landing success rate.
- **FlightNumber vs LaunchSite:** Here we can see if different launch sites have different landing outcome, as time goes by.
- **PayloadMass vs LaunchSite:** Here we can understand if a launch site is more suitable for certain payload mass.
- **SuccessRate vs Orbit:** It shows how likely a mission with a certain orbit will land successfully, and if there are “cursed” orbits.

EDA with Data Visualization

- **FlightNumber vs Orbit:** Here we can see if the success of an orbit is related to the flight number, i.e. as time goes by.
- **PayloadMass vs Orbit:** Here we can reveal the relationship between payload mass and the orbit type, and if success is more likely to occur in certain combinations of payload and orbit.
- **Yearly trend of success:** Last, but not least, here we show the trend of successful missions as the time goes by, to understand if SpaceX gained benefits from experience of past launches.
- This is the link to the github page of the related notebook: https://github.com/pignatta/Coursera_Capstone/blob/master/EDA%20with%20Data%20Visualization.ipynb

EDA with SQL

Here we present a list of useful SQL queries to gain insights from the Falcon9 missions database.

- **select distinct LAUNCH_SITE from SPACEXDATASET**: this query displays the name of the unique launch sites in the space mission
- **select * from SPACEXDATASET where LAUNCH_SITE like '%CCA%' limit 5:** this query displays 5 records where launch sites begin with the string ‘CCA’
- **select sum(payload_mass__kg_) from SPACEXDATASET where customer like '%NASA (CRS)%’**: this query displays the total payload mass carried by boosters launched by NASA (CRS)
- **select avg(payload_mass__kg_) from SPACEXDATASET where booster_version like '%F9 v1.1%’:** this query displays average payload mass carried by booster version F9 v1.1
- **select min(DATE) from SPACEXDATASET where landing__outcome like '%Success (ground%’**: this query lists the date when first successfully landing outcome in ground pad was achieved
- **select distinct booster_version from SPACEXDATASET where landing__outcome like '%Success (drone ship)%' and payload_mass__kg_ between 4000 and 6000:** this query lists the names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000

EDA with SQL

- **select distinct mission_outcome, count(*) from SPACEXDATASET group by mission_outcome:** this query lists the total number of successful and failure mission outcomes
- **select distinct booster_version from SPACEXDATASET where payload_mass__kg_ like (select max(payload_mass__kg_) from SPACEXDATASET):** this query lists the names of the booster_versions which have carried the maximum payload mass
- **select booster_version, launch_site from SPACEXDATASET where landing__outcome like '%Failure (drone ship)%' AND DATE like '%2015%’:** this query lists the failed landing_outcomes in drone ship, their booster versions, and launch site names for the in year 2015
- **select distinct landing__outcome, count(*) number from SPACEXDATASET where DATE between '2010-06-04' and '2017-03-20' group by landing__outcome order by count(*) desc:** this query ranks the count of landing outcomes (such as Failure (drone ship) or Success (ground pad)) between the date 2010-06-04 and 2017-03-20, in descending order
- This is the link to the github page of the related notebook: https://github.com/pignatta/Coursera_Capstone/blob/master/EDA_with_SQL.ipynb

Build an Interactive Map with Folium

In this section we explain how we build a map with Folium, using the data available in our Falcon9 missions dataframe.

- First, we created the base map with a standard layout
- Then, we used **Circle** object to mark the geographic locations of the four launch sites. It turns out that 3 of them are in Florida (very close one to each other) and the last is in California.
- Then, we created a **Marker** for all launch records. If a launch was successful \`(class=1)\`, then we use a green marker and if a launch was failed, we use a red marker \`(class=0)\`.
- To group all the Markers for the same launch site, we created a **MarkerCluster** object for each launch site, then plotted the results on the map.
- Then, we used the object **PolyLine** to draw a line between one of the launch site and the nearest railway.
- This is the link to the github page of the related notebook: https://github.com/pignatta/Coursera_Capstone/blob/master/Interactive%20Visual%20Analytics%20with%20Folium%20lab.ipynb

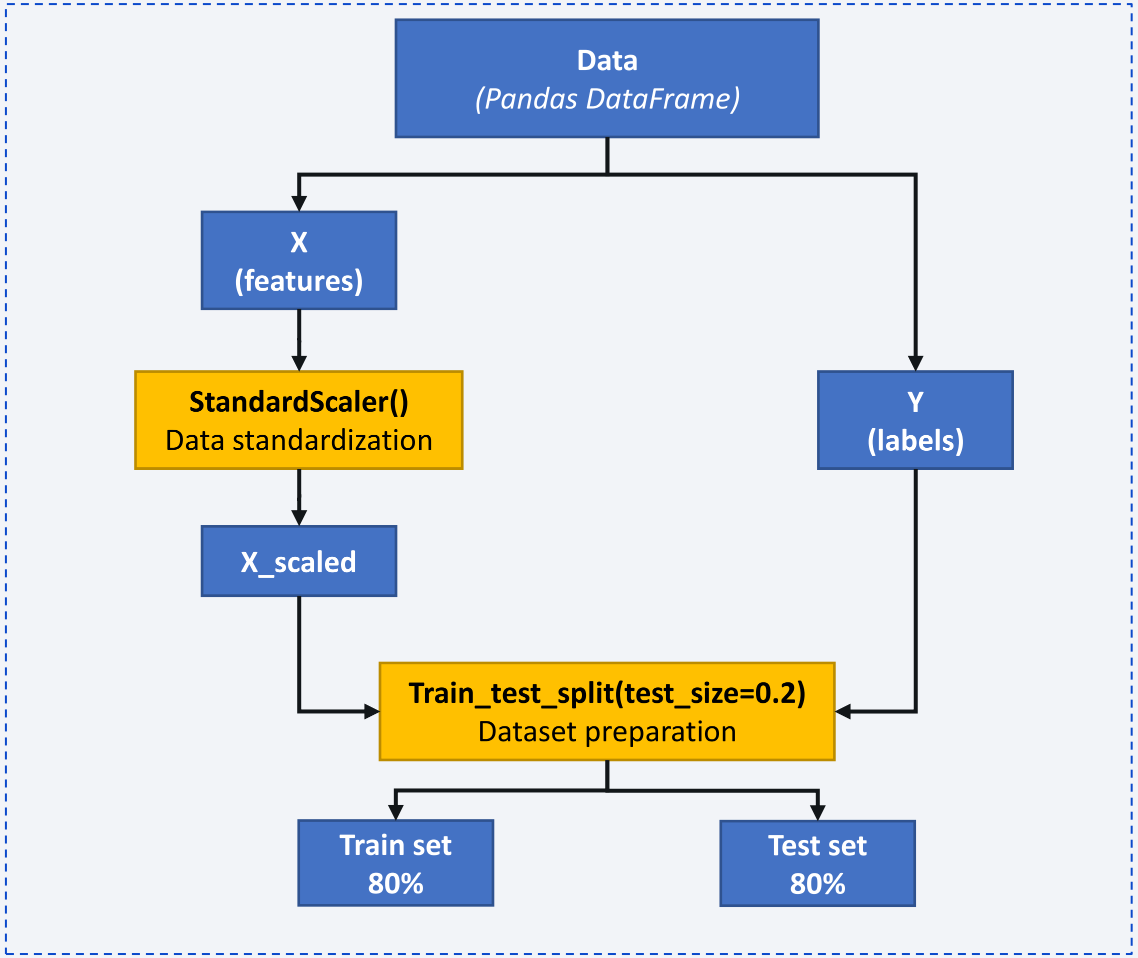
Build a Dashboard with Plotly Dash

- First, we added a Launch Site **Drop-down** input component, to select all the missions for the corresponding launch site.
- Then, we added a callback function to render the **success-pie-chart** based on selected site dropdown.
- Now, we added a **Range Slider** to select the Payload Mass. In this way it was possible to choose a minimum and maximum payload to restrict the selection of the missions and visualize the selection on a scatter chart.
- In addition to the pie-chart, we added a callback function to render the **success-payload-scatter-chart**, corresponding to a scatter plot PayloadMass vs Class, for all launch sites. The number of plotted missions depends on the parameters of the said range slider.
- This is the link to the github page of the related code: https://github.com/pignatta/Coursera_Capstone/blob/master/spacex_dash_app.py
- This is the link to the data to make the app run: https://github.com/pignatta/Coursera_Capstone/blob/master/spacex_launch_dash.csv

Predictive Analysis (Classification)

Here, data from Falcon9 launches are fed into several kind of prediction models, and trained within a grid search, to understand which perform better.

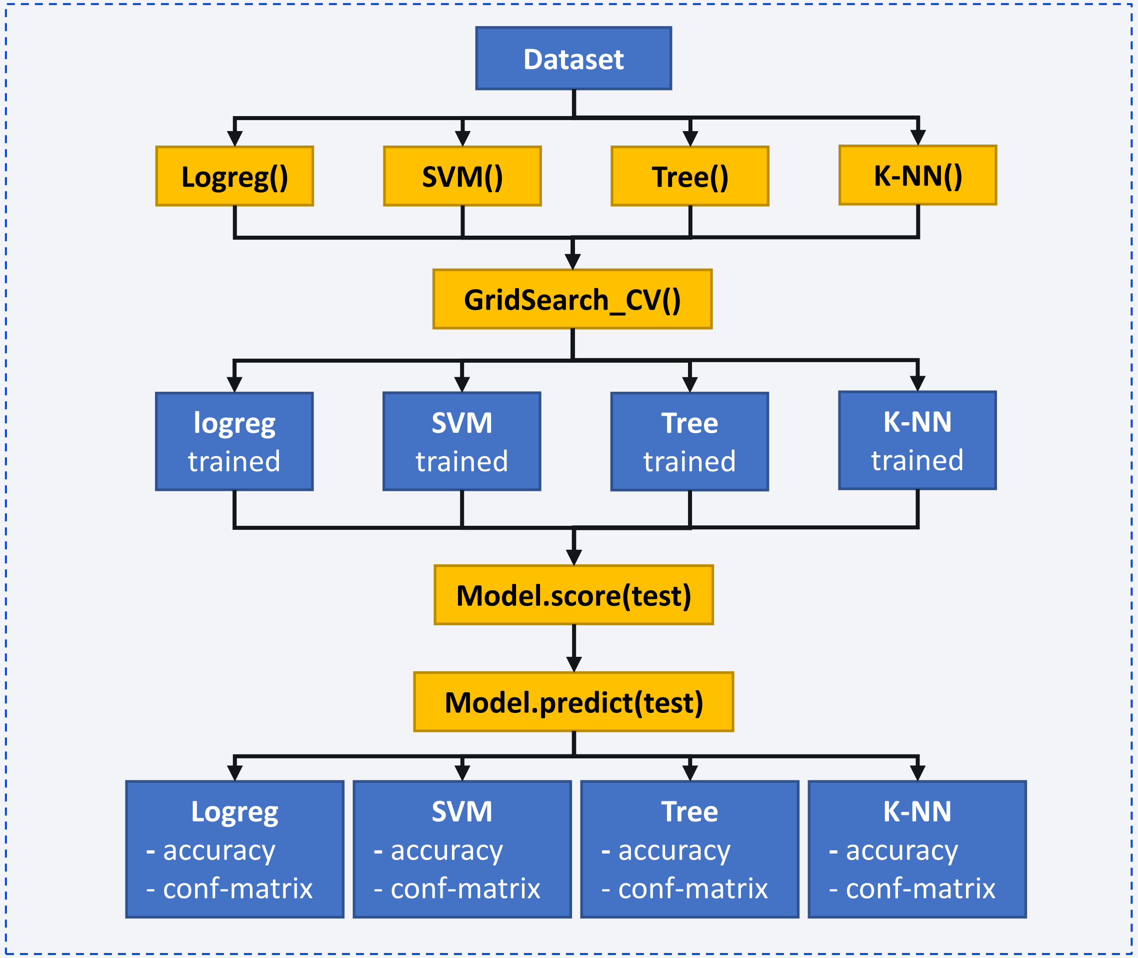
- **Dataset preparation**
  - Features (X) correspond to all data from the original dataframe, without the landing outcome. Categorical data are transformed into dummy variables of (0,1) value, leading to a total of 83 features.
  - The tags or labels (Y) are the landing outcome, corresponding to 0 (fail) and 1 (success).
  - The X were standardized with the function **StandardScaler** from sklearn.
  - Then, the X and Y were randomly split into a train set and a test set, in the proportion 80%-20%.



Predictive Analysis (Classification)

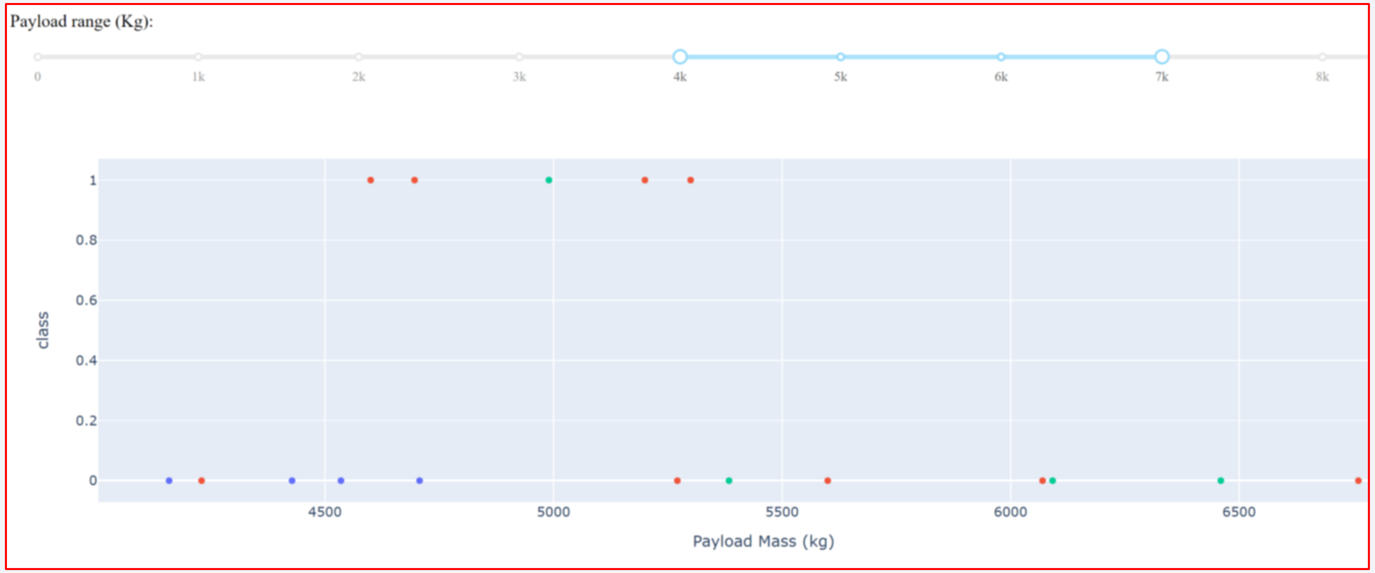
- **Models training**
  - four different models were trained with a grid search, to find the best combination of hyperparameters: **Logistic regression**, **Support Vector Machine**, **Decision Tree Classifier** and **k-Nearest Neighbours**.
  - After training, **accuracy** and **confusion matrices** for the best try for each model were calculated and plotted
  - It seems all the models perform quite well on the test set, but the decision tree has performed better (accuracy = 0.9444). The confusion matrices are comparable.

This is the link to the github page of the related code:
https://github.com/pignatta/Coursera_Capstone/blob/master/Machine%20Learning%20Prediction.ipynb



Results

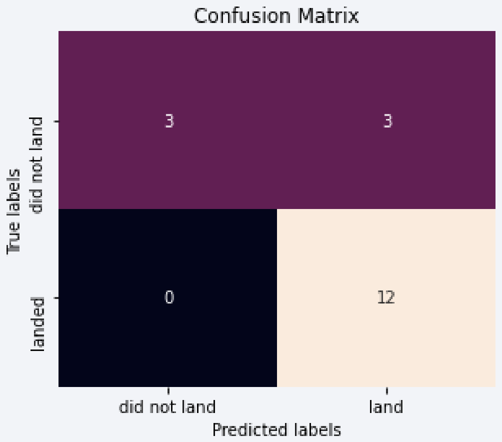
- SpaceX Falcon9 missions are more likely to success when the destination orbit are ES-L1, GEO, HEO and SSO, while SO orbit is not so lucky, even if only one mission was done on that orbit.
- Some orbits are more suitable for some payload mass, indeed heavy payloads have a negative influence on GTO orbits and positive on VLEO and Polar LEO (ISS) orbits.
- In the last 10 years SpaceX has done a lot of work to improve the success rate of their missions. Until 2013 no mission landed successfully, while in 2020 only 2 mission over 10 had a bad epilogue.
- A powerful instrument to see data interactively is plotly Dash, allowing to change parameters in real time, and to isolate and underline the info we want.



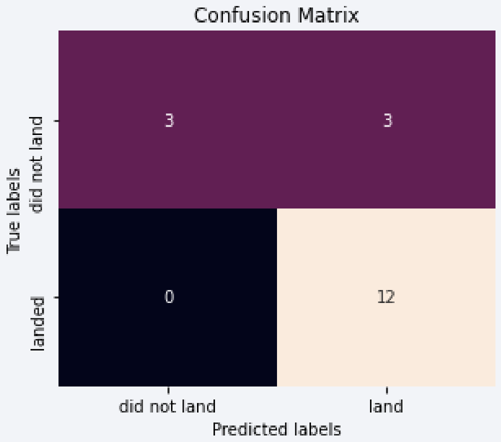
Results

- Here are the **confusion matrices** of the four trained models on the **test set**.

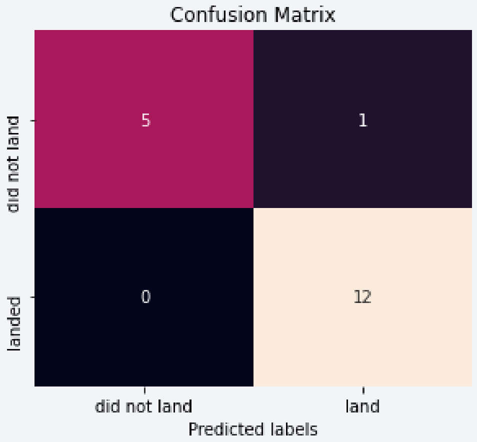
Logistic Regression



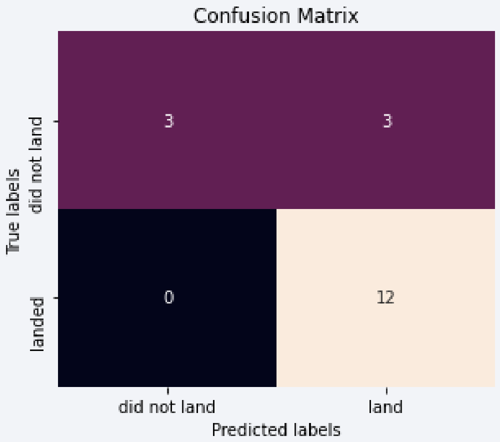
Support Vector Machine



Decision Tree



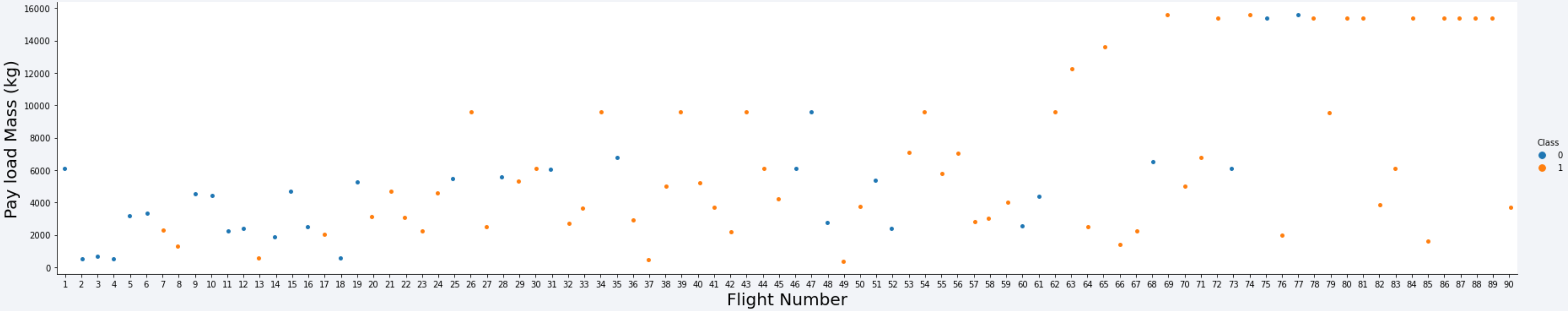
K-Nearest Neighbours



Section 2

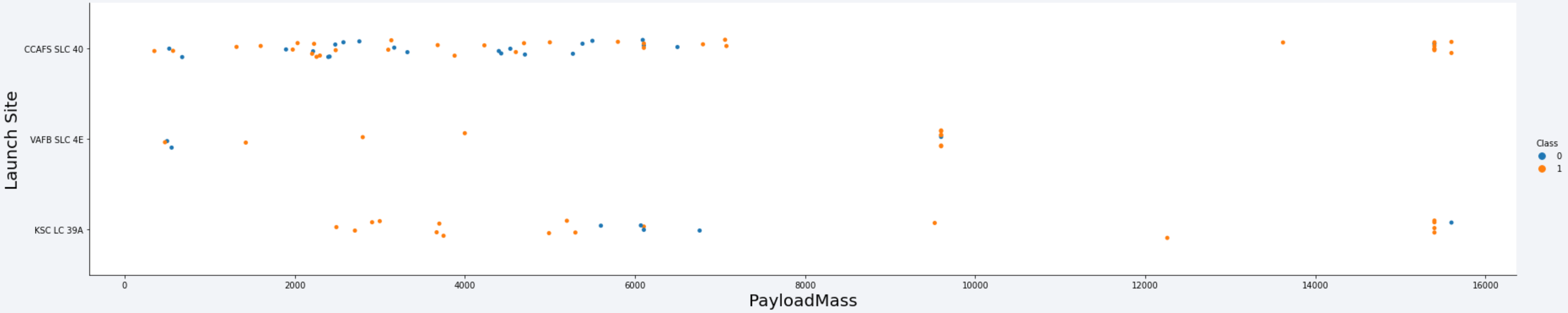
Insights drawn from EDA

Flight Number vs. Launch Site



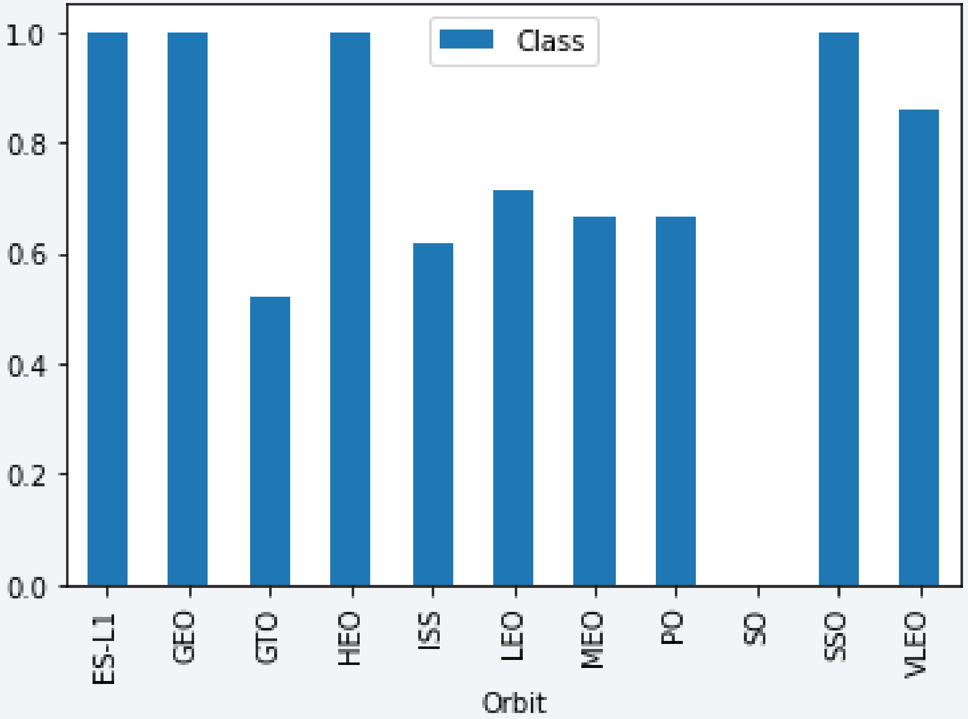
We see that as the flight number increases, the first stage is more likely to land successfully. The payload mass is also important; it seems the more massive the payload, the less likely the first stage will return.

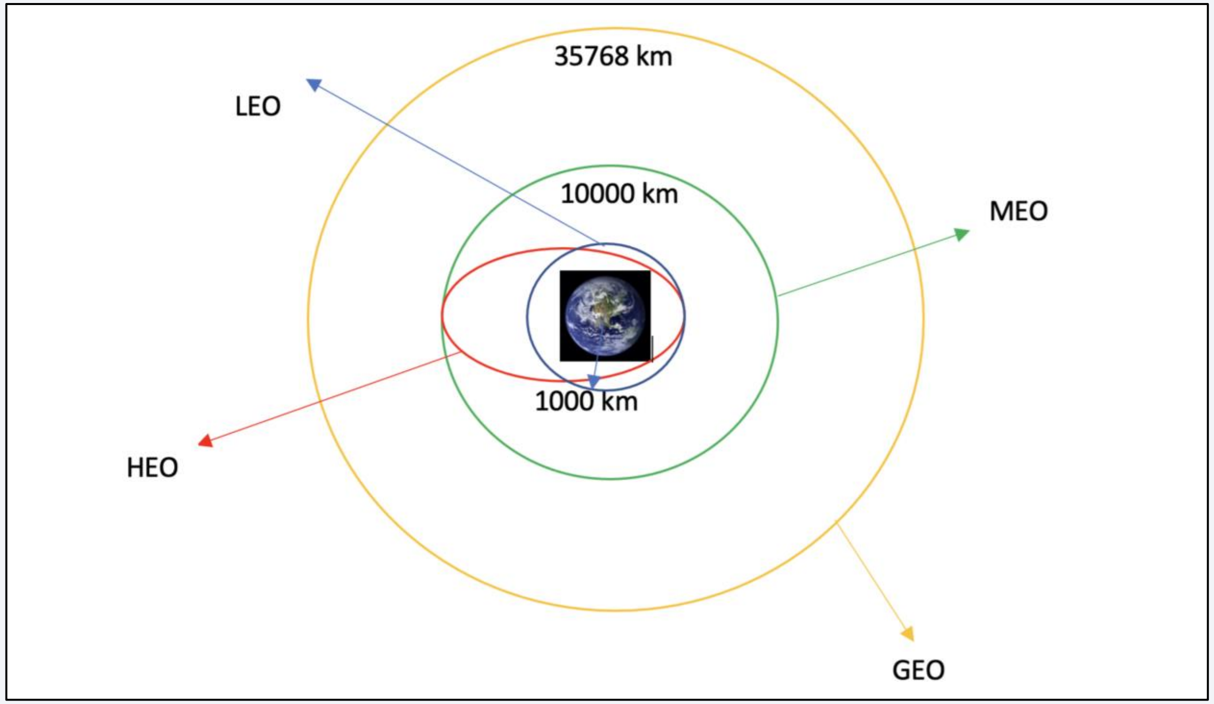
Payload vs. Launch Site



We see that some launch sited are more suitable for light payloads, while others are more suitable for heavier payloads. CCAFS SLC 40 (Cape Canaveral) seems to have problems with light and medium payloads. VAFB SLC 4E (California) cannot handle heavy payloads and has a success rate of under 50%.

Success Rate vs. Orbit Type

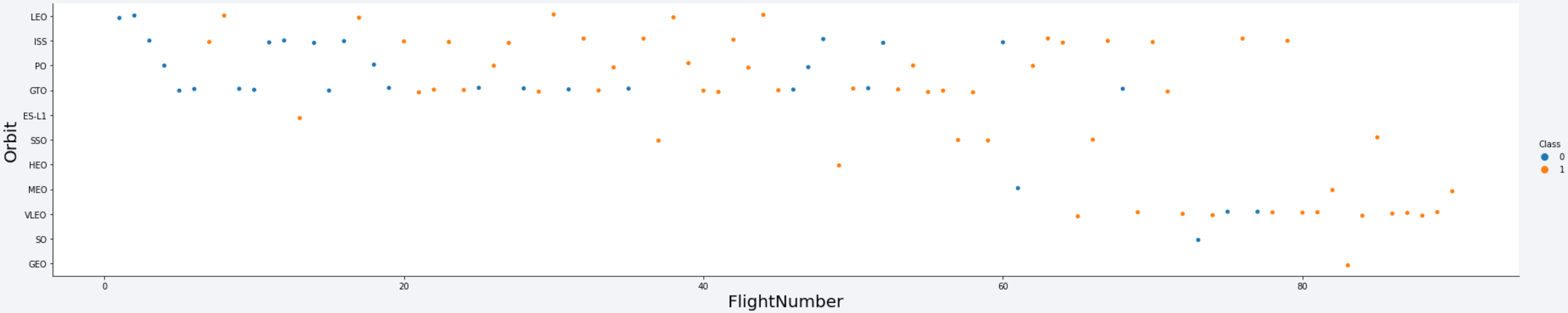




Here are reported the success rate of each orbit. It seems that SO (Sun Synchronous Orbit) is not a good orbit for Falcon9, even if only one attempt has been done for this orbit. Maybe this orbit can be grouped with SSO.

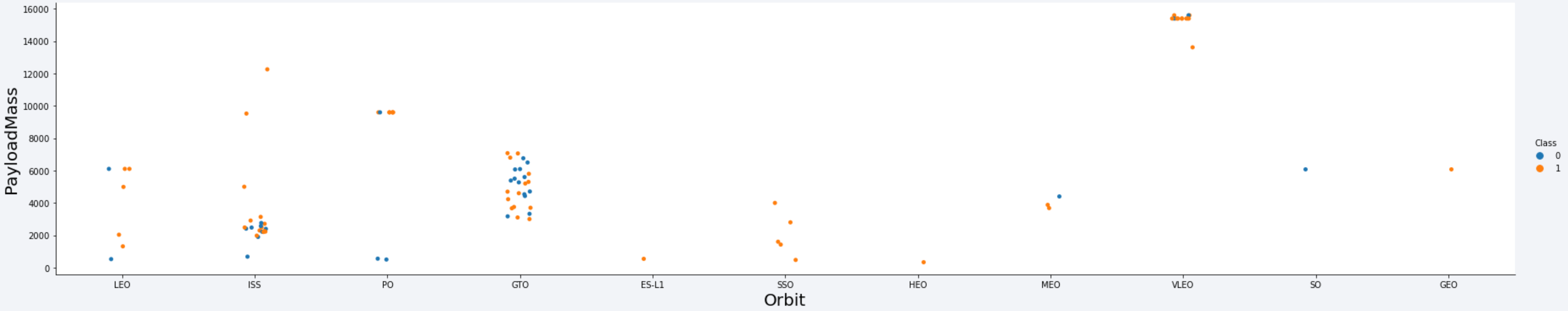
ES-L1 (lagrangian point L1), CEO, HEO and SSO reported a 100% success rate.

Flight Number vs. Orbit Type



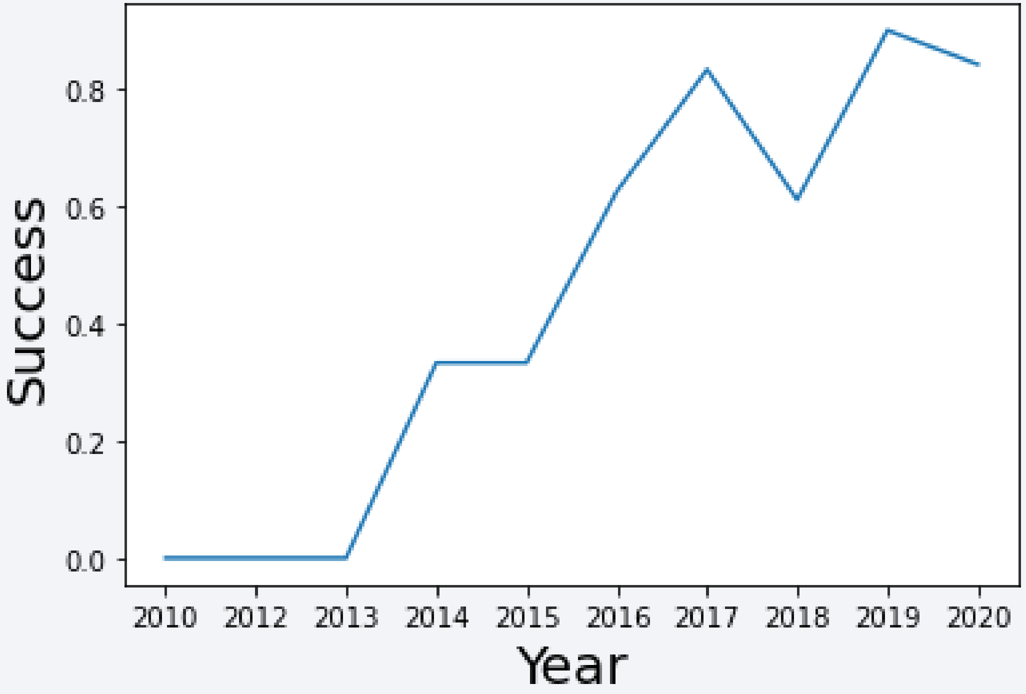
It seems that in the LEO orbit the Success appears related to the number of flights; on the other hand, there seems to be slightly related between flight number when in GTO orbit, and not related with ISS orbit.

Payload vs. Orbit Type



Medium payloads seems to have a bad influence on GTO orbits, with about 50% of failure.
VLEO orbit is suited only for heavy payloads and the rate of success is quite high.
Low payloads (~2000 Kg) are not good for ISS orbit, while this is not true for heavier payloads.

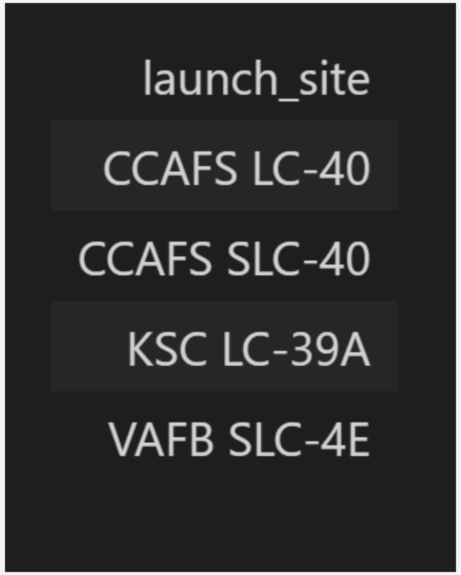
Launch Success Yearly Trend



This is one the most summarizing and most eloquent plots: it demonstrate that SpaceX has done a good job in improving their methods and learned from the experience.

All Launch Site Names

```
select distinct LAUNCH_SITE from SPACEXDATASET;
```

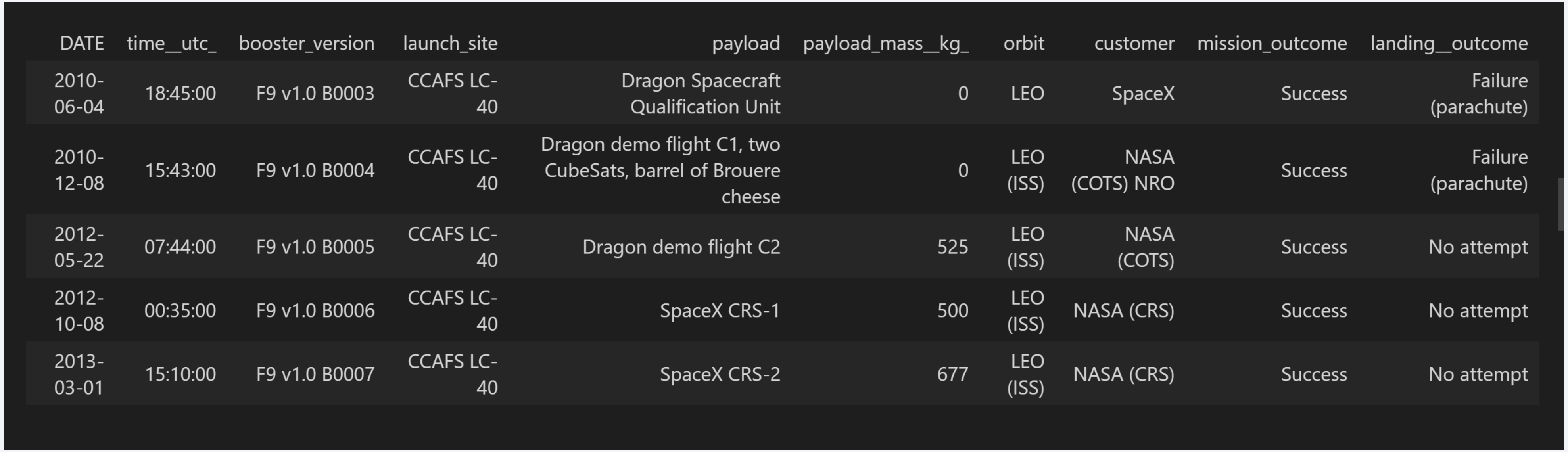


launch_site
CCAFS LC-40
CCAFS SLC-40
KSC LC-39A
VAFB SLC-4E

This command queries for the unique elements of the launch site column.

Launch Site Names Begin with 'CCA'

```
select * from SPACEXDATASET where LAUNCH_SITE like '%CCA%' limit 5
```

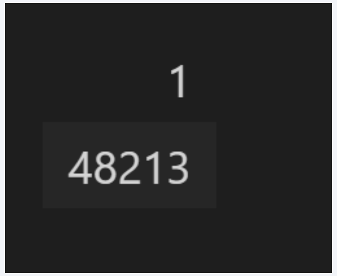


DATE	time__utc_	booster_version	launch_site	payload	payload_mass__kg_	orbit	customer	mission_outcome	landing__outcome
2010-06-04	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success	Failure (parachute)
2010-12-08	15:43:00	F9 v1.0 B0004	CCAFS LC-40	Dragon demo flight C1, two CubeSats, barrel of Brouere cheese	0	LEO (ISS)	NASA (COTS) NRO	Success	Failure (parachute)
2012-05-22	07:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success	No attempt
2012-10-08	00:35:00	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

This query search the first 5 records where the launch site columns contains the string ‘CCA’

Total Payload Mass

```
select sum(payload_mass__kg_) from SPACEXDATASET where customer like '%NASA (CRS)%'
```

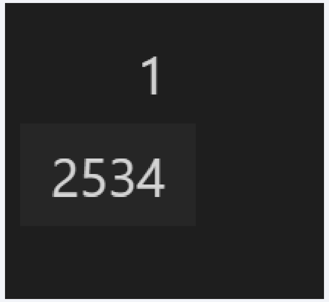


1
48213

This command displays the total payload mass carried by boosters launched by NASA (CRS).

Average Payload Mass by F9 v1.1

```
select avg(payload_mass__kg_) from SPACEXDATASET where booster_version like '%F9 v1.1%'
```

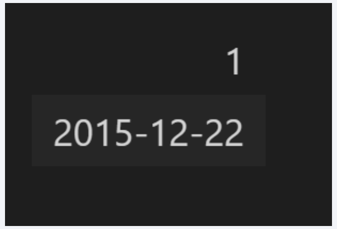


1
2534

This command displays the mean (over the whole dataset) payload mass carried by booster version F9 v1.1

First Successful Ground Landing Date

```
select min(DATE) from SPACEXDATASET where landing__outcome like '%Success (ground%'
```

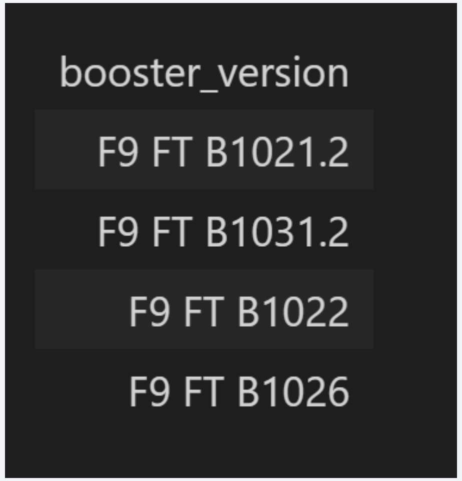


```
          1
2015-12-22
```

This command displays the date of the first successful landing outcome in ground pad achieved.

Successful Drone Ship Landing with Payload between 4000 and 6000

```
select distinct booster_version from SPACEXDATASET where landing__outcome like
'%Success (drone ship)%' and payload_mass__kg_ between 4000 and 6000
```

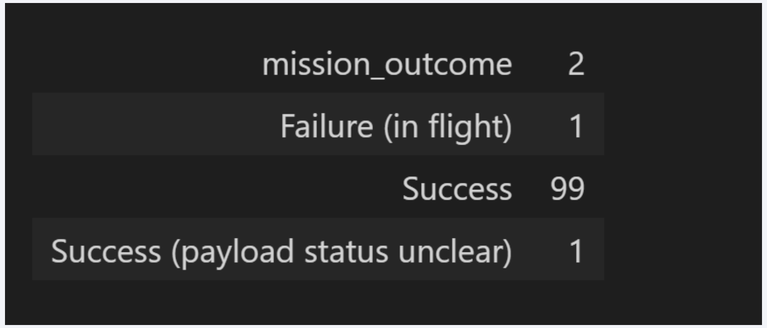


booster_version
F9 FT B1021.2
F9 FT B1031.2
F9 FT B1022
F9 FT B1026

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

Total Number of Successful and Failure Mission Outcomes

```
select distinct mission_outcome, count(*) from SPACEXDATASET group by mission_outcome
```

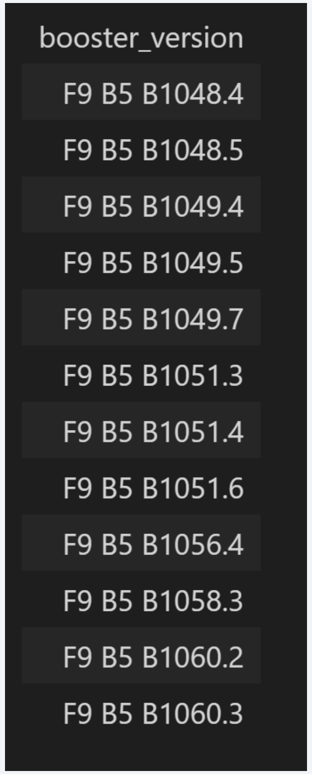


mission_outcome	2
Failure (in flight)	1
Success	99
Success (payload status unclear)	1

This command lists the total number of successful and failure mission outcomes.

Boosters Carried Maximum Payload

```
select distinct booster_version from SPACEXDATASET where payload_mass__kg_ like
(select max(payload_mass__kg_) from SPACEXDATASET)
```

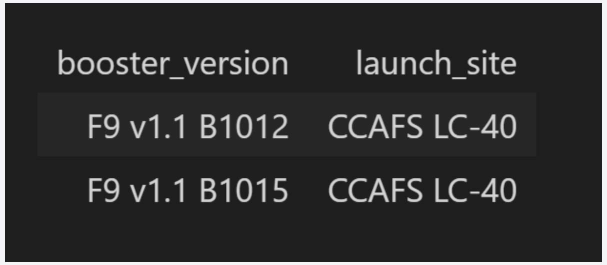


booster_version
F9 B5 B1048.4
F9 B5 B1048.5
F9 B5 B1049.4
F9 B5 B1049.5
F9 B5 B1049.7
F9 B5 B1051.3
F9 B5 B1051.4
F9 B5 B1051.6
F9 B5 B1056.4
F9 B5 B1058.3
F9 B5 B1060.2
F9 B5 B1060.3

This command lists the names of the booster_versions which have carried the maximum payload mass.

2015 Failed Launch Records

```
select booster_version, launch_site from SPACEXDATASET where landing__outcome like
'%Failure (drone ship)%' AND DATE like '%2015%'
```

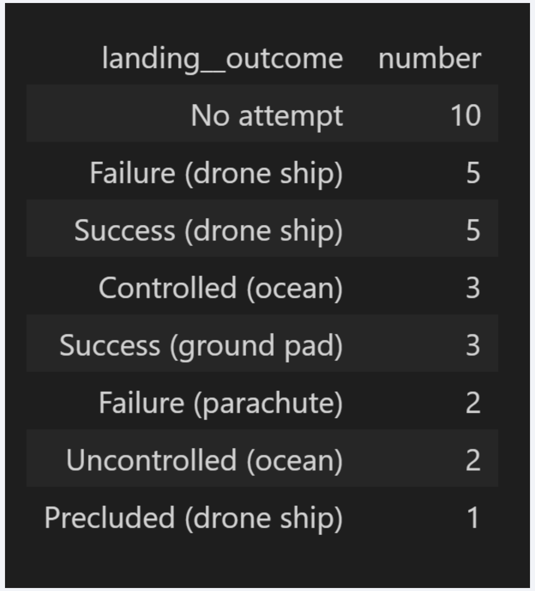


booster_version	launch_site
F9 v1.1 B1012	CCAFS LC-40
F9 v1.1 B1015	CCAFS LC-40

This command lists the failed landing_outcomes in drone ship, their booster versions, and launch site names for the in year 2015

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

```
select distinct landing__outcome, count(*) number from SPACEXDATASET where DATE between
'2010-06-04' and '2017-03-20' group by landing__outcome order by count(*) desc
```



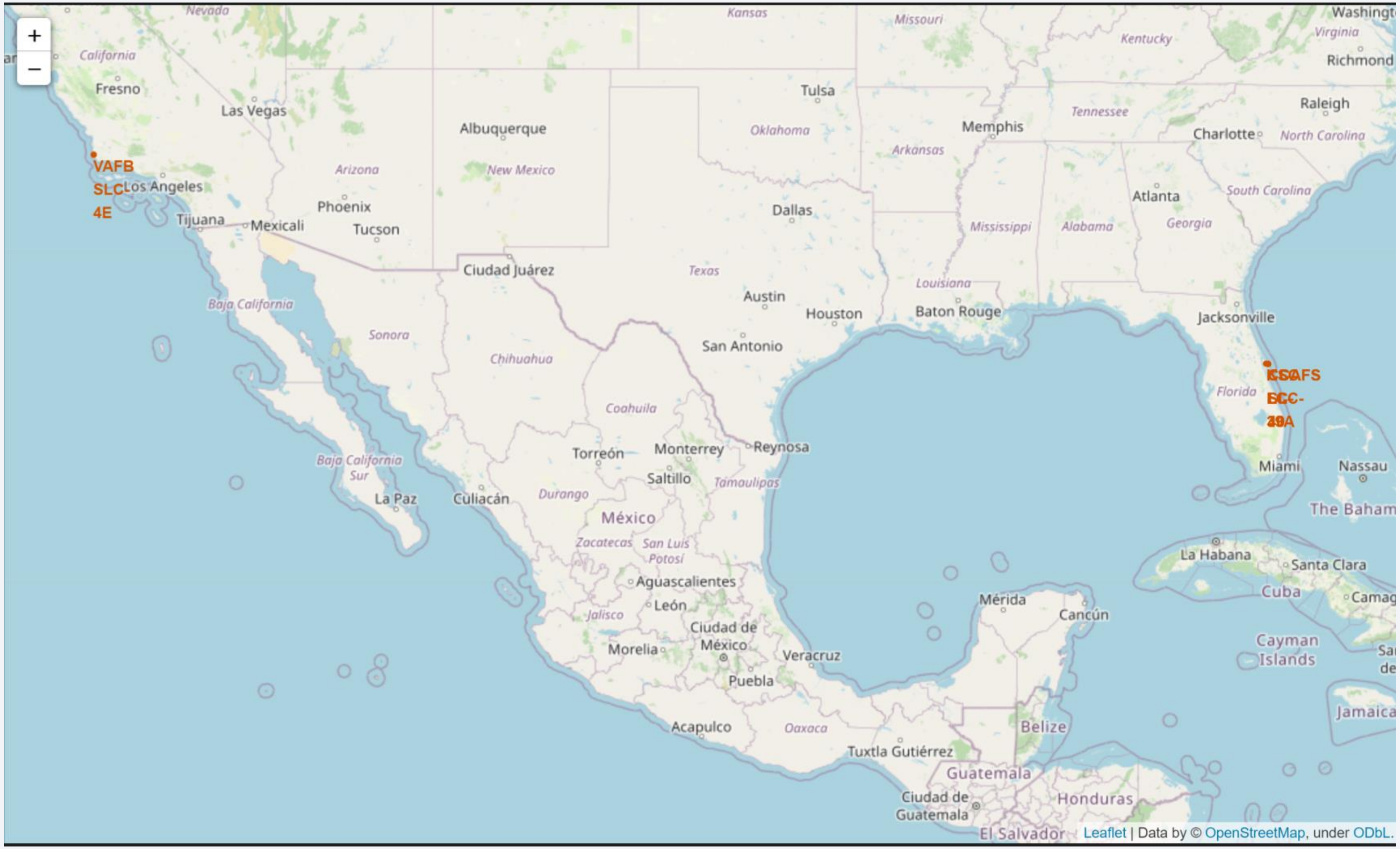
landing__outcome	number
No attempt	10
Failure (drone ship)	5
Success (drone ship)	5
Controlled (ocean)	3
Success (ground pad)	3
Failure (parachute)	2
Uncontrolled (ocean)	2
Precluded (drone ship)	1

This command ranks the count of landing outcomes (such as Failure (drone ship) or Success (ground pad)) between the date 2010-06-04 and 2017-03-20, in descending order

Section 4

Launch Sites Proximities Analysis

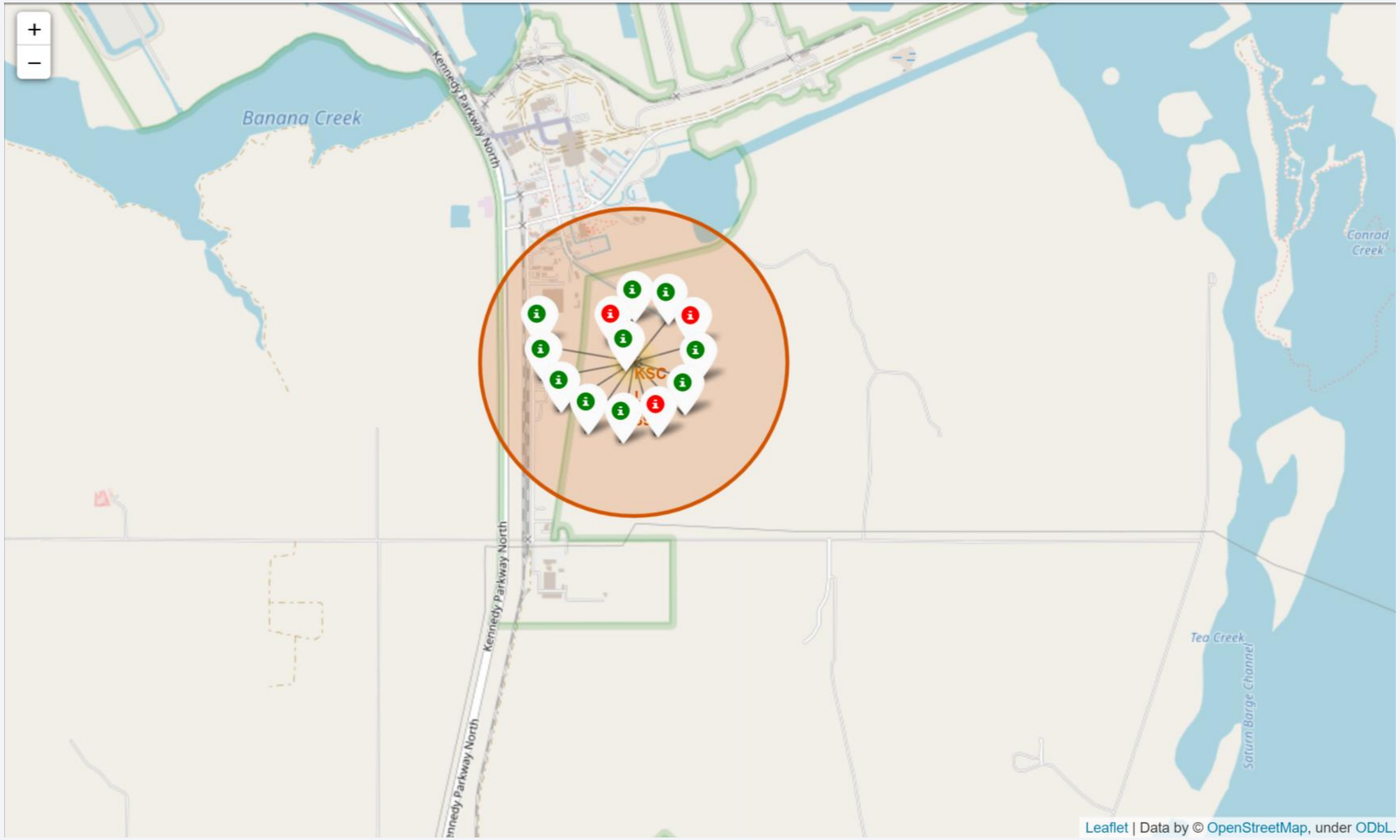
Folium Map of all the Launch Sites



This map reports the locations of all the launch sites for Falcon9 booster from 2010 to 2020.

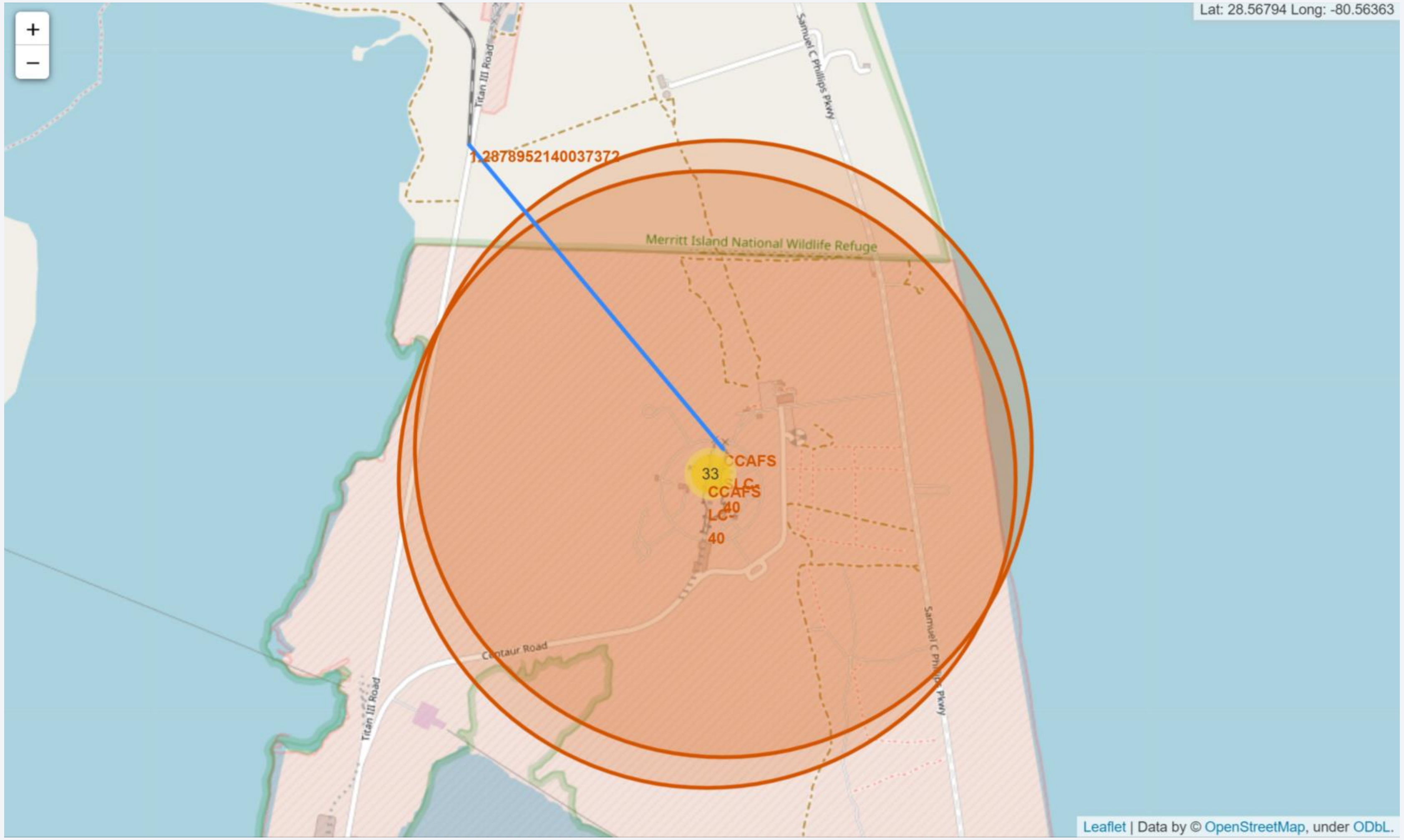
There are 3 launch sited in Florida, near Cape Canaveral, and 1 in California.

Map of the outcome for a specific Launch Site

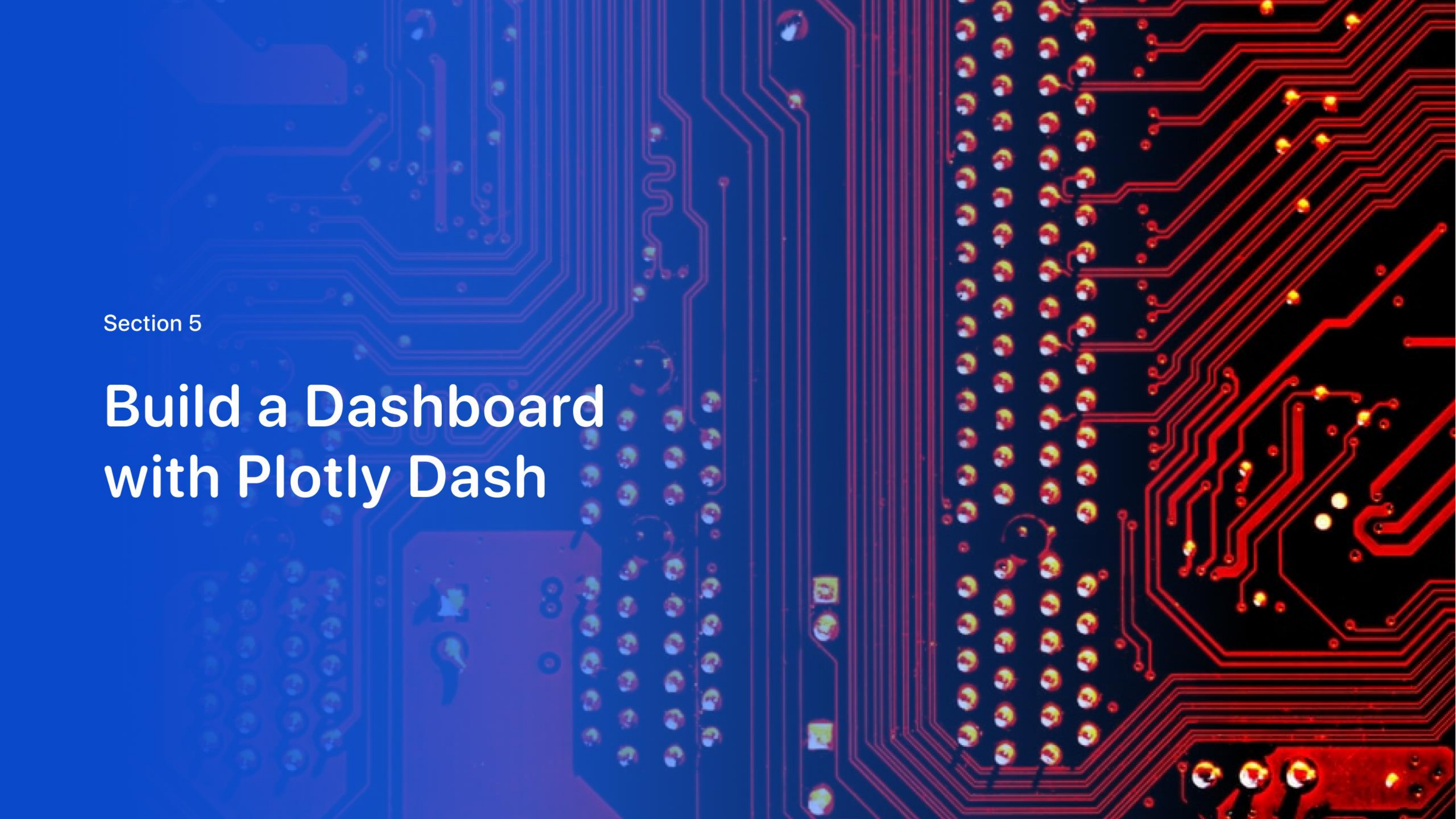


This map reports each launch performed by this site, and the relative outcome. Green circle means success, while red circle means failure. The missions started on this platform seem to have done quite good. The site is KSC LC-39A, in Florida.

Folium Map to calculate distances



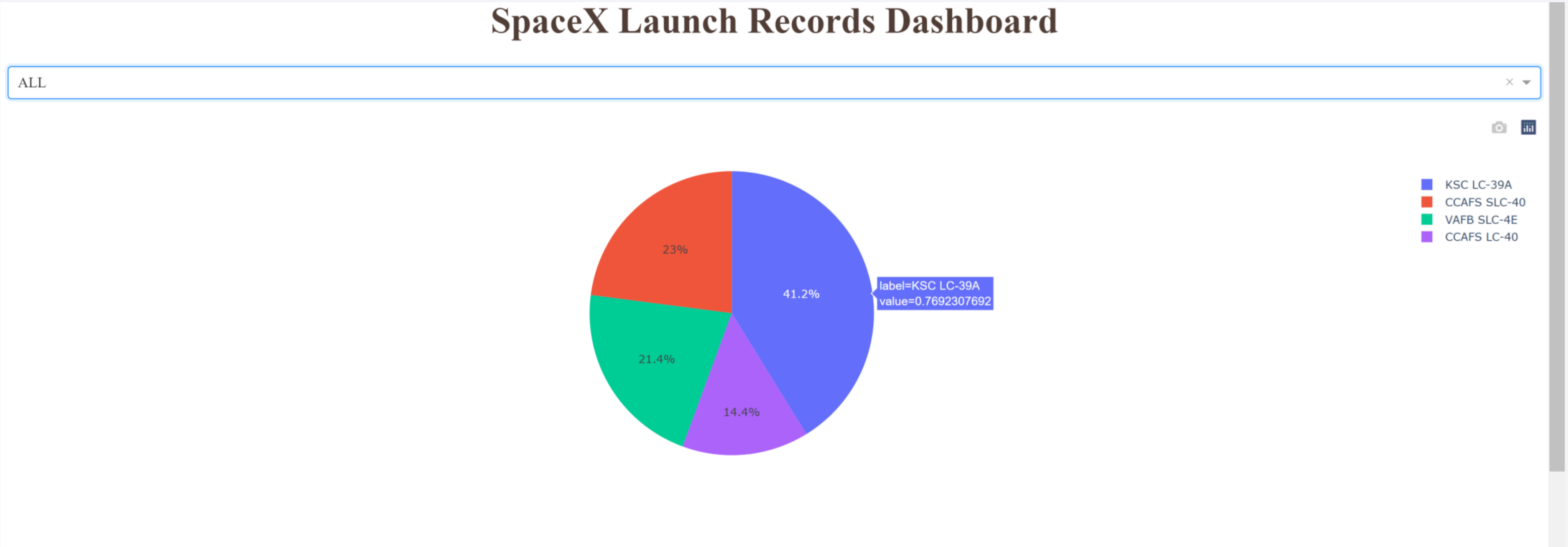
This map reports the distance between the point on one of the end of the blue line (a railway) and the launch site CCAFS SLC-40. The railway is about 1.29 Km far from the site.



Section 5

Build a Dashboard with Plotly Dash

Pie Chart of success for all Launch Sites

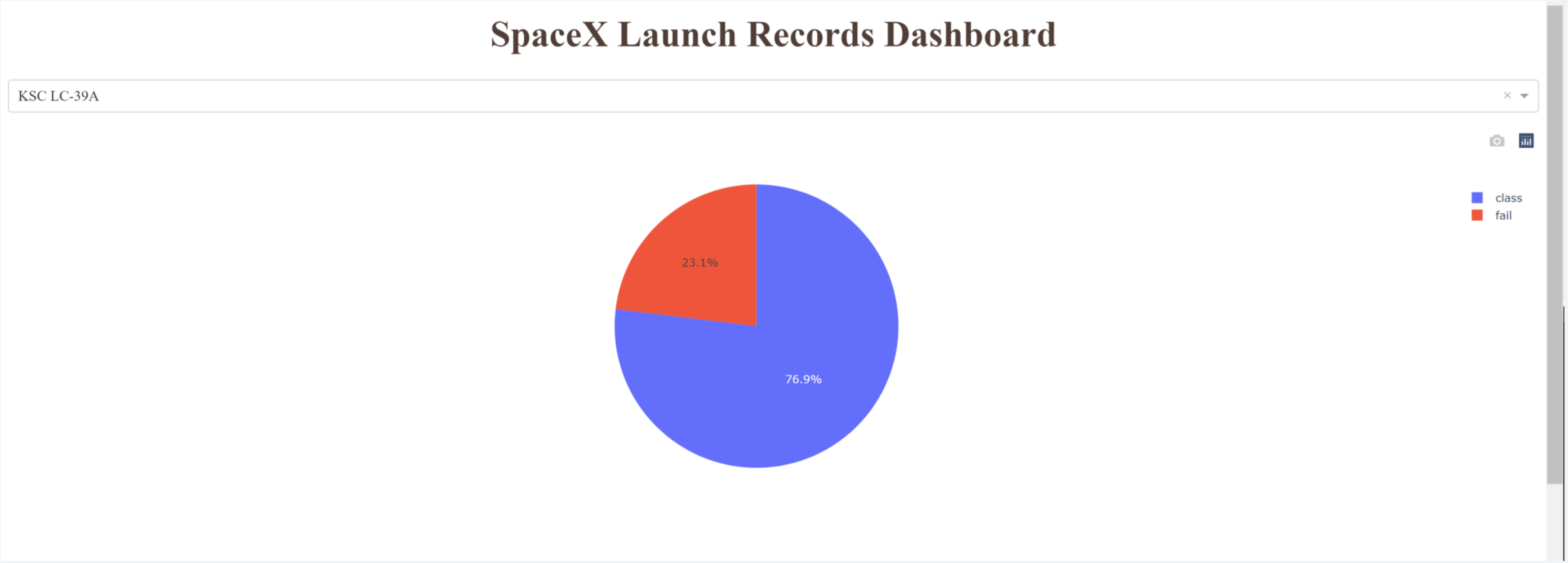


SpaceX Launch Records Dashboard

ALL

This pie chart reports the contribution of each launch site to the successful missions. Anyway, it does not tell us nothing about the success rate per each launch site, for which it is necessary to check any single launch site. Hovering with mouse over each pie sector tells us the percent of success of that launch site.

Pie chart of the most successful Launch Site

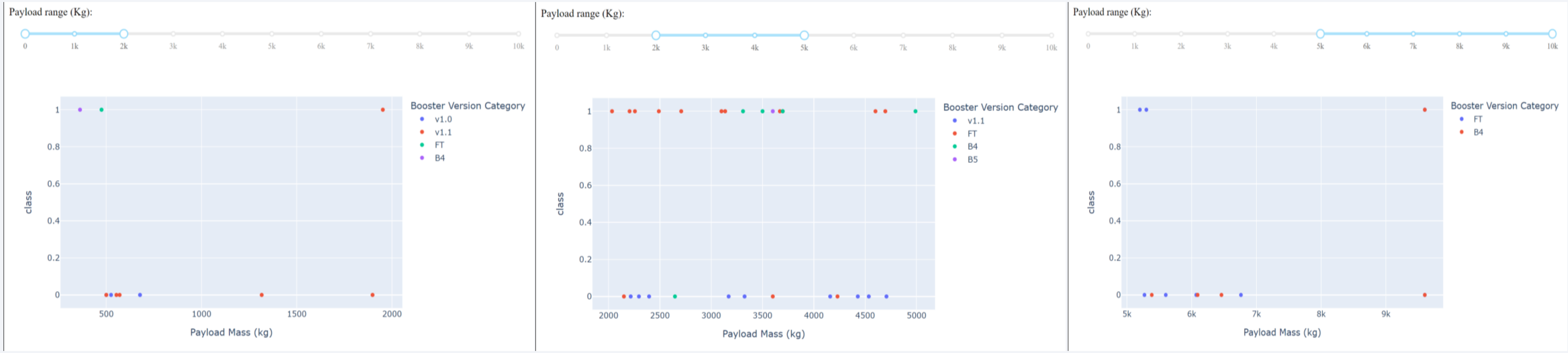


SpaceX Launch Records Dashboard

KSC LC-39A

This pie chart reports the success and failure rate for the most successful launch site, KSC LC-39A.

Scatter Plot of outcomes for different payloads

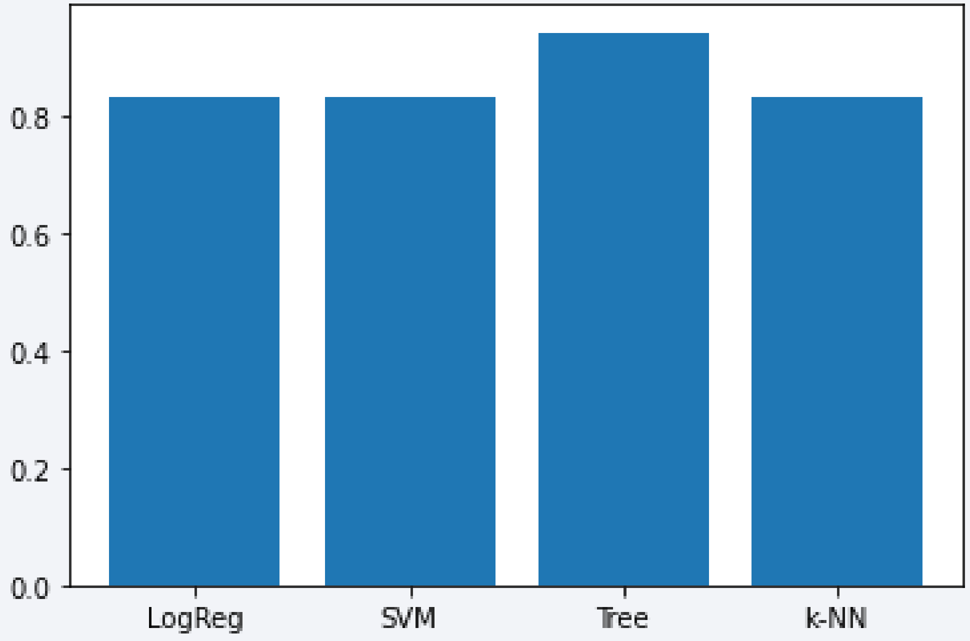


These 3 scatter plots report the landing outcome divided by the Booster version category, for 3 different payload ranges: light (0-2000 Kg), medium (2000-5000 Kg) and heavy (>5000 Kg).
It emerges that not all the boosters can handle heavy payloads (this is quite obvious). For light payloads the outcome is not very good, as most of the times the booster cannot land successfully. For medium payloads the situation seems to be better, specially for the FT booster. For heavy payloads the statistics return bad, with the most of mission ending with a failure, for both boosters.

Section 6

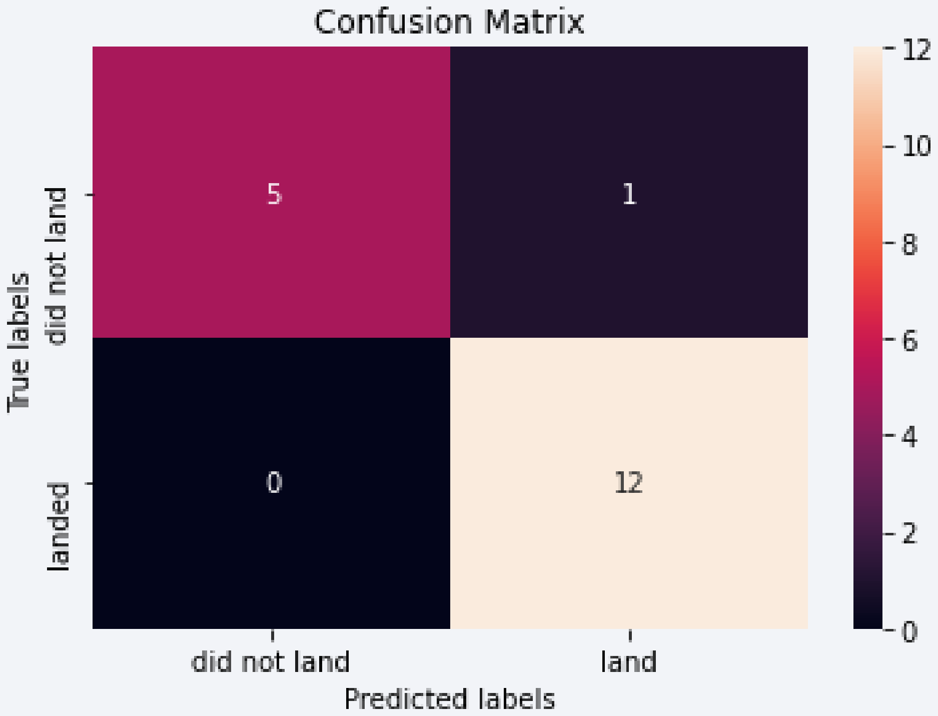
Predictive Analysis (Classification)

Classification Accuracy



The bar plot represents the classification accuracy for the 4 models on the test set. It seems that the Decision Tree performs better that the other models, for this particular problem.

Confusion Matrix



This is the confusion Matrix for the Decision Tree Classifier for the test set. The more diagonal the matrix is (i.e. the elements on the diagonal are larger that those on other tiles) the better the model performs on the selected dataset. In particular, there are no false negatives and only one false positive, giving high **precision** and **recall** values.

Conclusions

- SpaceX Falcon9 missions are more likely to success when the destination orbit are ES-L1, GEO, HEO and SSO, while SO orbit is not so lucky, even if only one mission was done on that orbit.

- Some orbits are more suitable for some payload mass, indeed heavy payloads have a negative influence on GTO orbits and positive on VLEO and Polar LEO (ISS) orbits.

- In the last 10 years SpaceX has done a lot of work to improve the success rate of their missions. Until 2013 no mission landed successfully, while in 2020 only 2 mission over 10 had a bad epilogue.

- With the Decision Tree classifier we are now able to predict (if no other parameters act from now on) with high accuracy if a certain launch, with a certain booster, from a certain site, with a certain payload will be successful or not, within a small error.

- As, probably, the parameters of the future missions will change, it is important to keep the model trained with continuously new data from last launches.

Thank you!