Data Science Capstone project

<Yiman Li>

<2021.08.31>

Outline

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

Executive Summary

- This is the presentation slides of the Project “SpaceX Falcon9 first stage Landing Prediction”
- All the projects can be find in my GitHub

Introduction

- In this section, we will predict if the Falcon 9 first stage will land successfully.
- SpaceX advertises Falcon 9 rocket launches on its website with a cost of 62 million dollars; other providers cost upward of 165 million dollars each, much of the savings is because SpaceX can reuse the first stage. Therefore if we can determine if the first stage will land, we can determine the cost of a launch.

Methodology

- Data collection methodology:
  - Request to the SpaceX API
- Perform data wrangling
  - Implementing functions on pandas Dataframe
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - How to build, tune, evaluate classification models

Methodology

Data collection

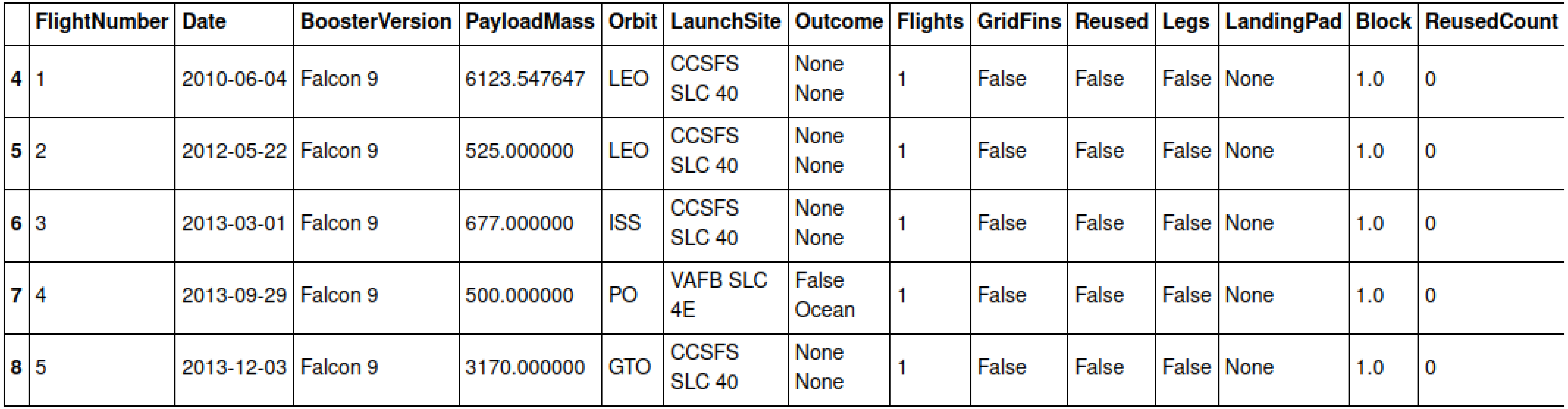
- We first define several functions, like getBoosterVersion, gerLaunchSite and use these functions to extract the information contained in the static url (API).
- The extracted information is then converted to pandas Dataframe for further cleaning and analysis.

Data collection – SpaceX API

Added a flowchart of SpaceX API calls here

Description r.t. previous page

URL



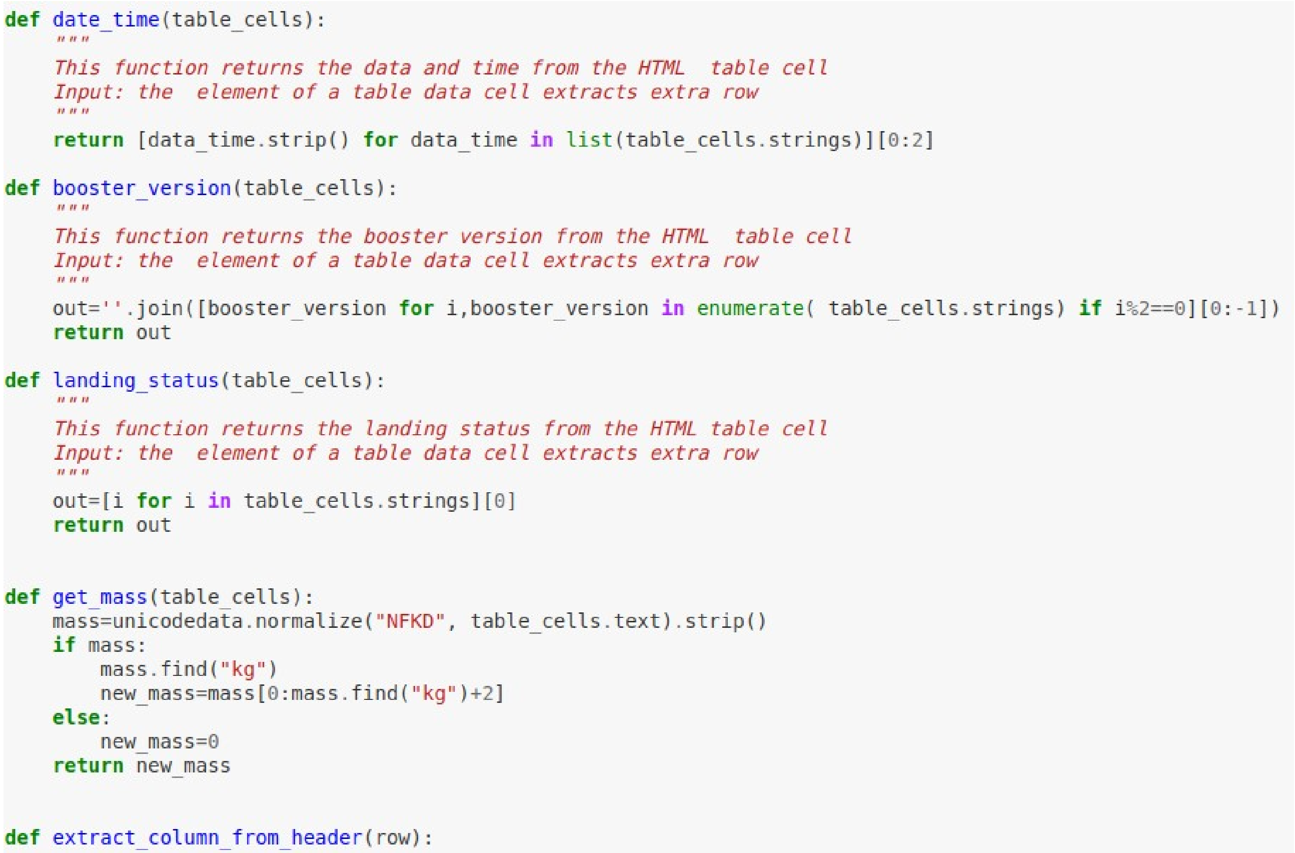
	FlightNumber	Date	BoosterVersion	PayloadMass	Orbit	LaunchSite	Outcome	Flights	GridFins	Reused	Legs	LandingPad	Block	ReusedCount
4	1	2010-06-04	Falcon 9	6123.547647	LEO	CCSFS SLC 40	None None	1	False	False	False	None	1.0	0
5	2	2012-05-22	Falcon 9	525.000000	LEO	CCSFS SLC 40	None None	1	False	False	False	None	1.0	0
6	3	2013-03-01	Falcon 9	677.000000	ISS	CCSFS SLC 40	None None	1	False	False	False	None	1.0	0
7	4	2013-09-29	Falcon 9	500.000000	PO	VAFB SLC 4E	False Ocean	1	False	False	False	None	1.0	0
8	5	2013-12-03	Falcon 9	3170.000000	GTO	CCSFS SLC 40	None None	1	False	False	False	None	1.0	0

Data collection – Web scraping

Add a flowchart of web scraping here

We first define several functions to extract information from the website (especially rely on the package BeautifulSoup), then parse the HTML file to extract details, like table and head inforamtion

URL



```
def date_time(table_cells):
    """
    This function returns the data and time from the HTML  table cell
    Input: the  element of a table data cell extracts extra row
    """
    return [data_time.strip() for data_time in list(table_cells.strings)][0:2]

def booster_version(table_cells):
    """
    This function returns the booster version from the HTML  table cell
    Input: the  element of a table data cell extracts extra row
    """
    out=''.join([booster_version for i,booster_version in enumerate( table_cells.strings) if i%2==0][0:-1])
    return out

def landing_status(table_cells):
    """
    This function returns the landing status from the HTML table cell
    Input: the  element of a table data cell extracts extra row
    """
    out=[i for i in table_cells.strings][0]
    return out


def get_mass(table_cells):
    mass=unicodedata.normalize("NFKD", table_cells.text).strip()
    if mass:
        mass.find("kg")
        new_mass=mass[0:mass.find("kg")+2]
    else:
        new_mass=0
    return new_mass


def extract_column_from_header(row):
```

Data wrangling

- Based on the previous notebook, we can extract information and convert it to pandas Dataframe.
- Here we main manipulate with pandas function to convert between feature, like one-hot coding, or give certain values based on given conditions.
- URL

EDA with data visualization

- Here we mostly use scatter plot to show the distribution and line plot to show the trend with time.

- URL

EDA with SQL

- Here we mainly create a bounding between the SQL and python notebook, then implementing SQL queries in notebook.

- URL

Build an interactive map with Folium

- According to the instructions, here we have added the marker, icon, circle to the maps to visualize the Launch Sites.

- URL

Build a Dashboard with Plotly Dash

- Summarize what plots/graphs and interactions you have added to a dashboard

- Explain why you added those plots and interactions

- Add the GitHub URL of your completed Plotly Dash lab, as an external reference and peer-review purpose

Predictive analysis (Classification)

- We built the LR, kNN, DT and SVM models, find the best parameters using grid search, and evaluate by using confusion matrix and computing confusion matrix

- URL

Results

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

EDA with Visualization

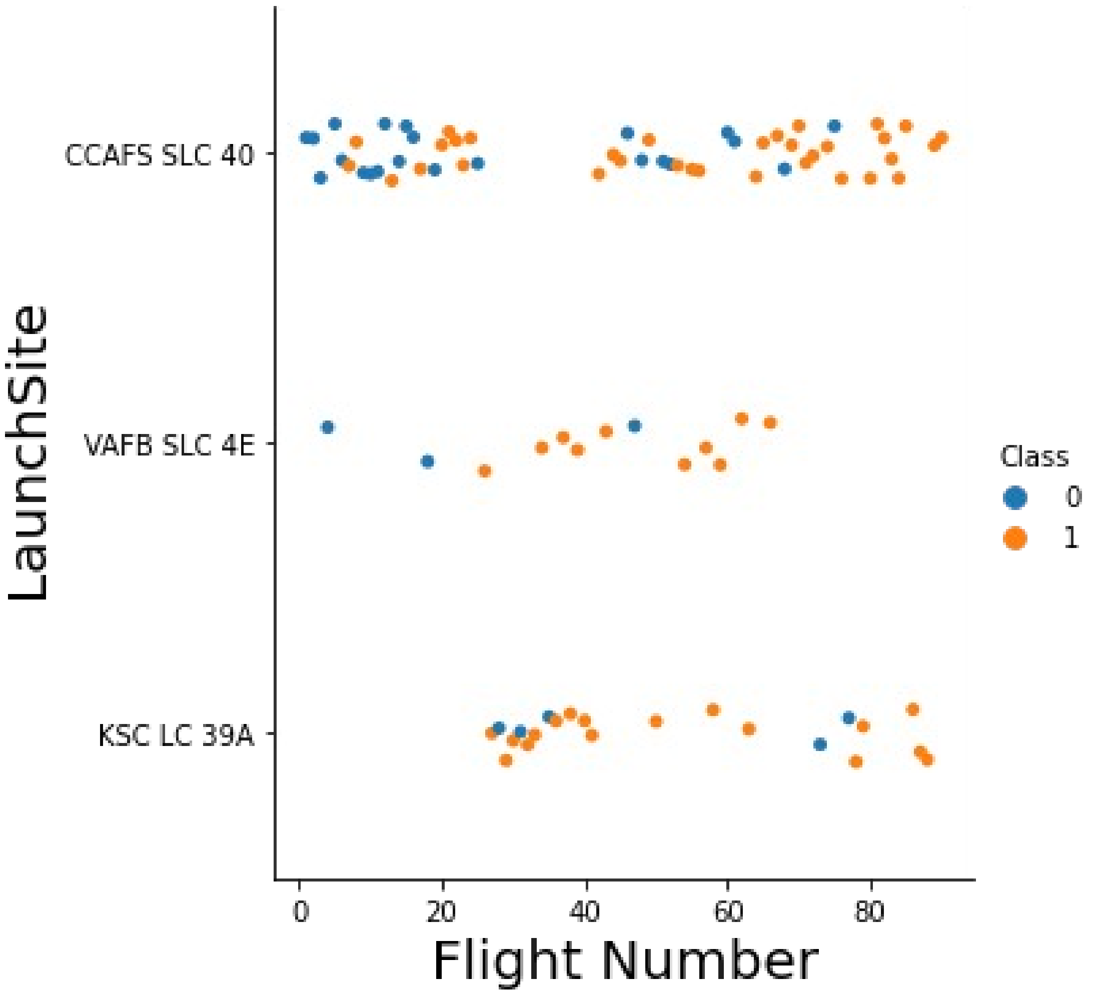
URL

Flight Number vs. Launch Site

Show a scatter plot of Flight Number vs. Launch Site

Most Launch with class 0 locate in “CCAFS SLC 40”;

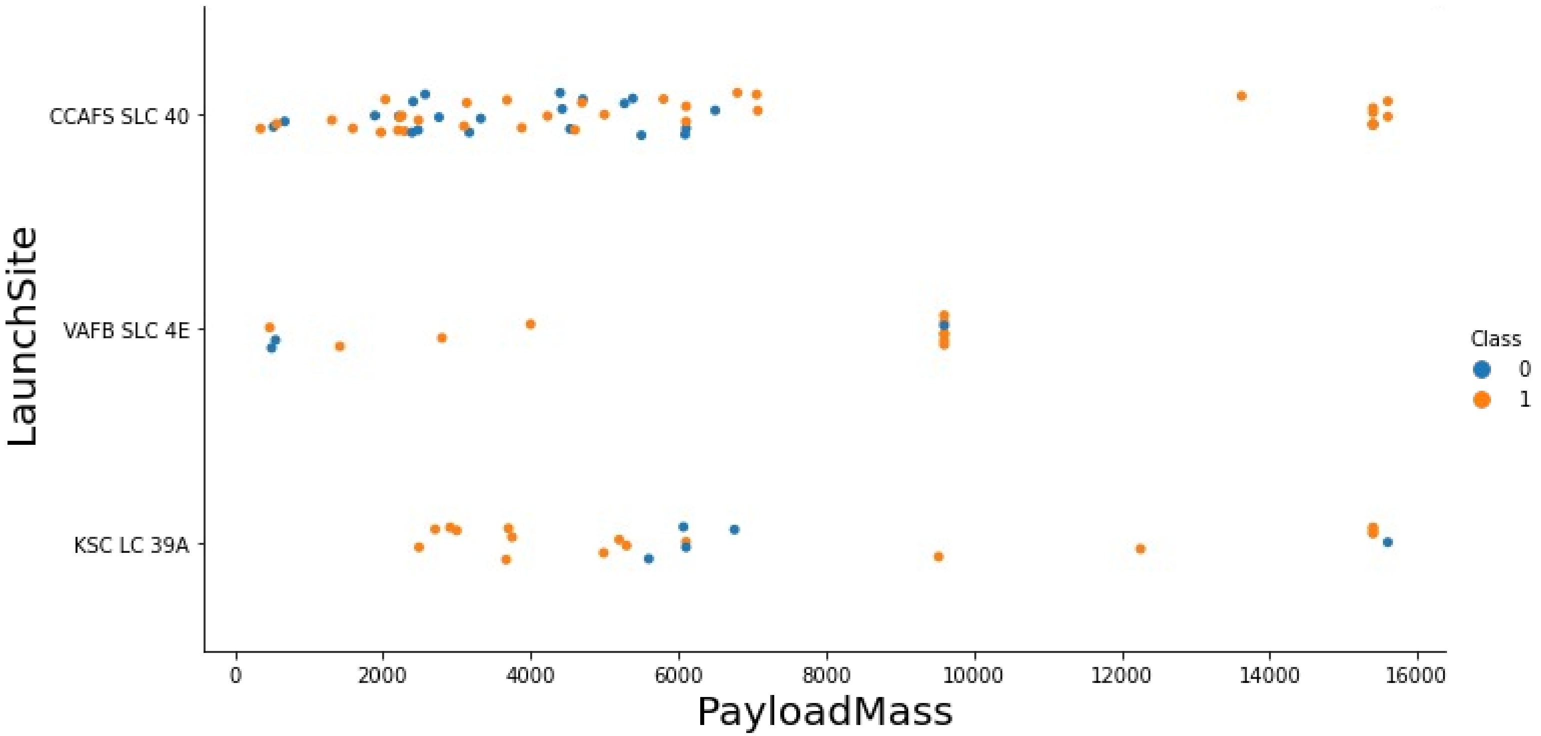
Launch with class 1 are in 3 sites all evenly visible.



Payload vs. Launch Site

Show a scatter plot of Payload vs. Launch Site

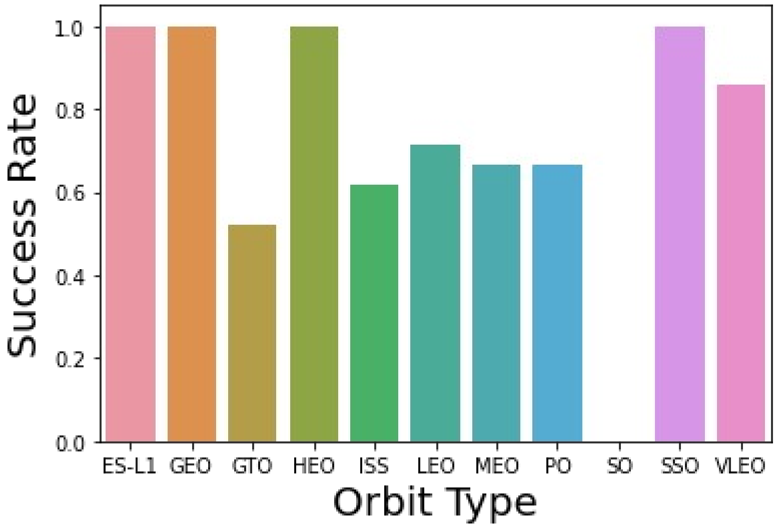
Most Launch with class 0 has payload lower than 8000Kg;



Success rate vs. Orbit type

Show a barchart for the success rate of each orbit type

ES-L1, GEO, HEO, SSO seem always have very high success rate, while SO always fails.

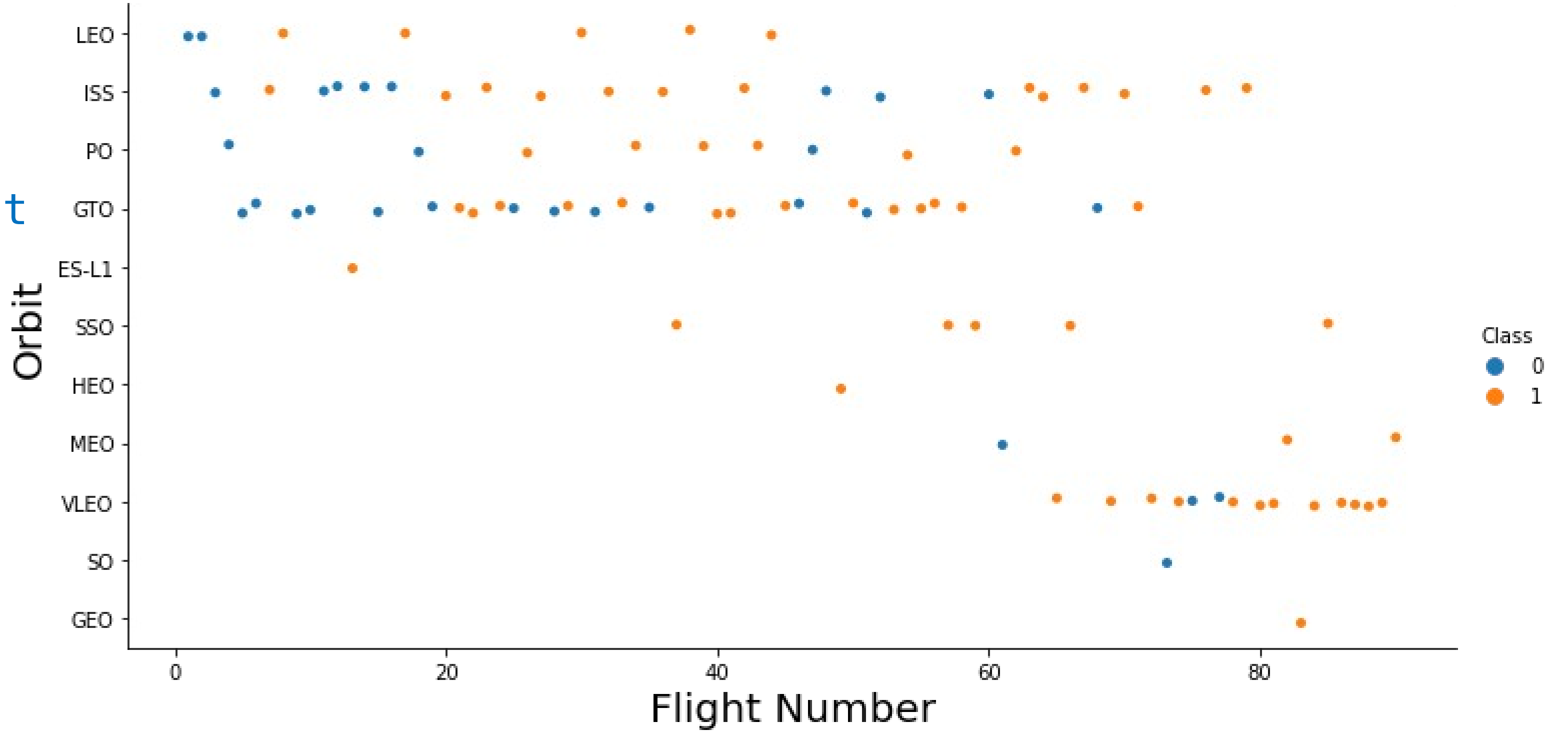


Flight Number vs. Orbit type

Show a scatter point of Flight number vs. Orbit type

The Launch with class 0 is mostly in orbit LEO, ISS and GTO;

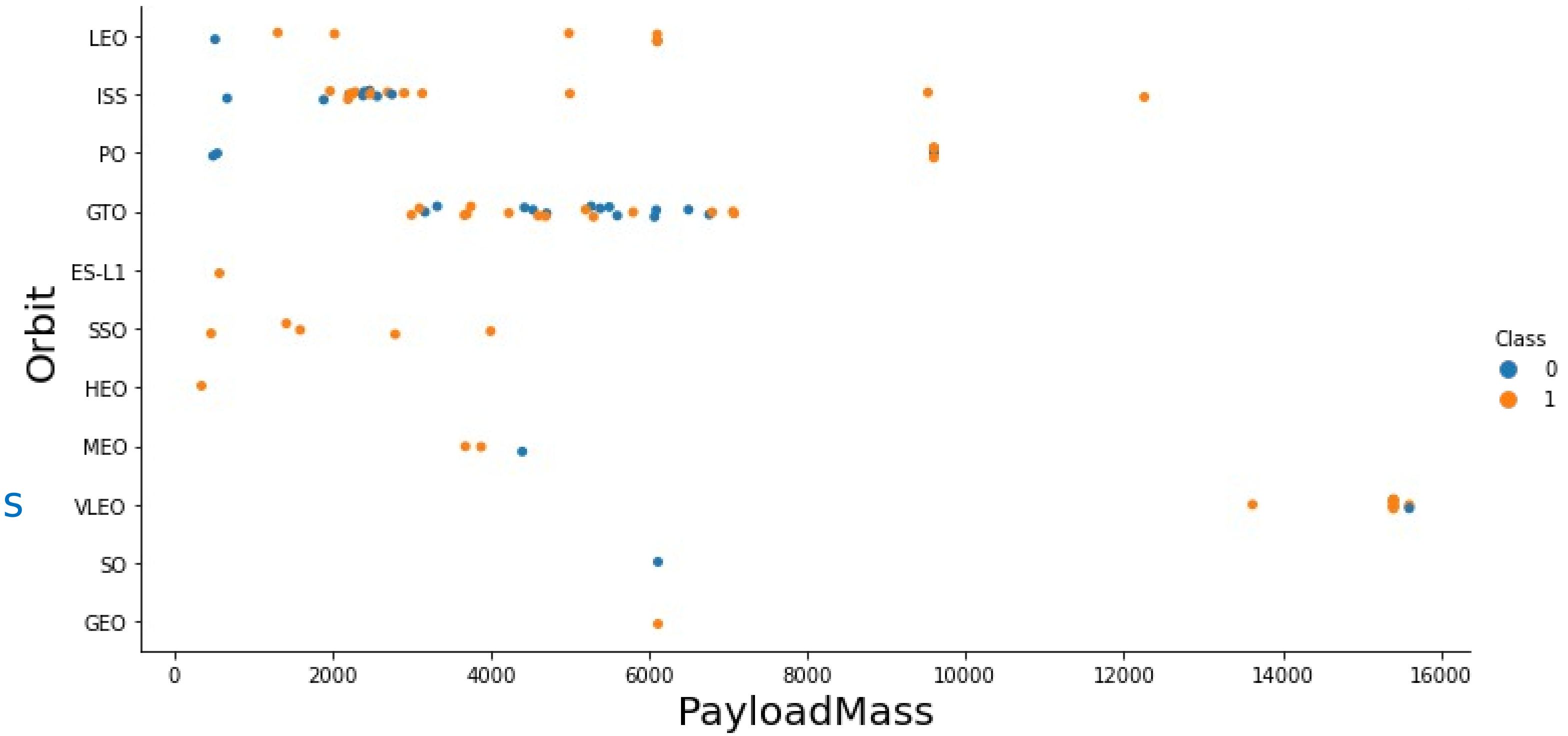
The data here is very chaos, there seems no clear relation here.



Payload vs. Orbit type

Show a scatter point of payload vs. orbit type

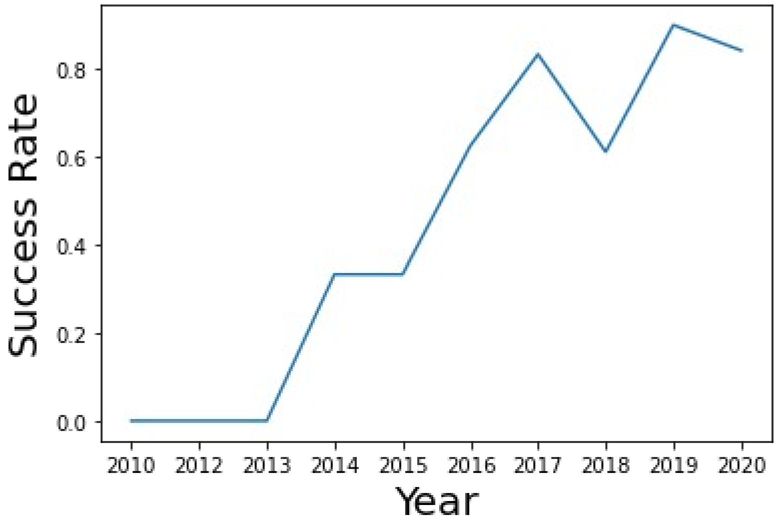
The Launch with class 0 is mostly in orbit LEO, ISS and GTO and under the payload mass of 8000kg;



Launch success yearly trend

Show a line chart of yearly average success rate

In total the success rate is gradually increasing with time, only that in 2018, there showed a very obvious decrease.

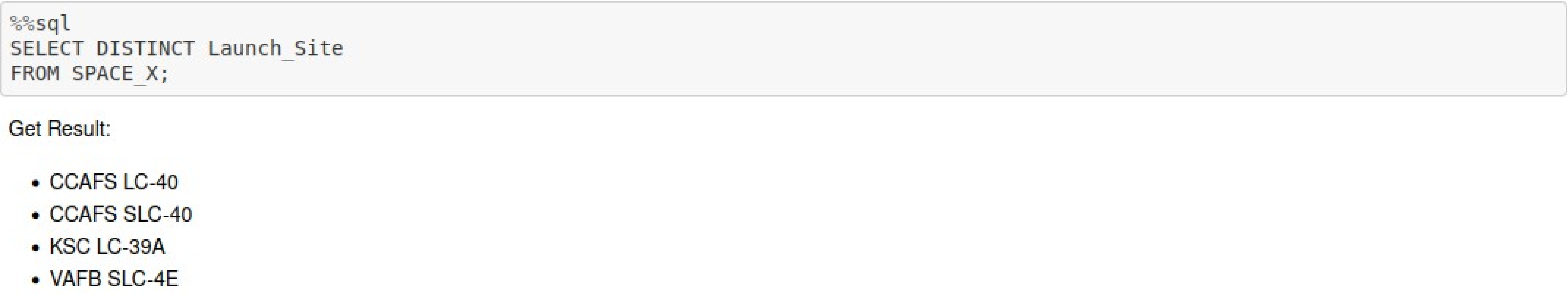


EDA with SQL

URL

All launch site names

- Find the names of the unique launch sites

- 

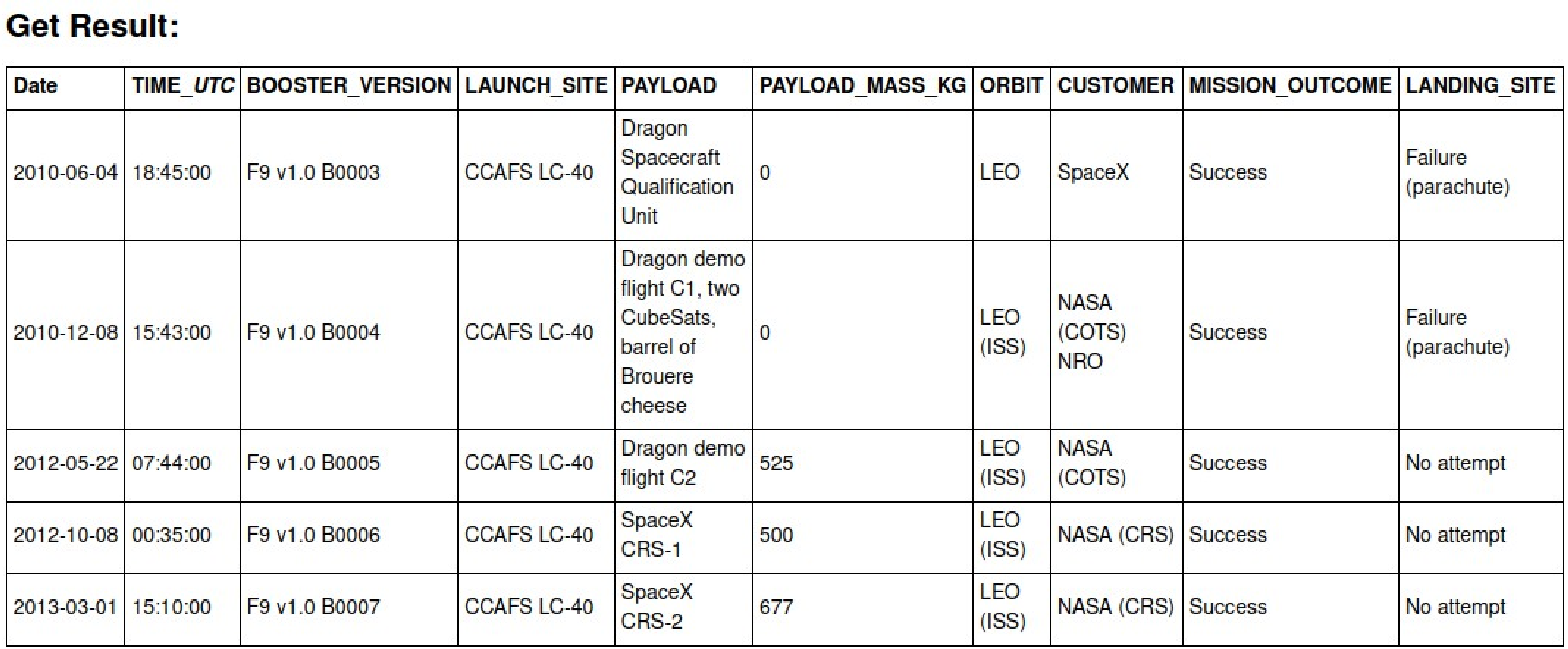
```
%%sql
SELECT DISTINCT Launch_Site
FROM SPACE_X;
```

Get Result:

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

Launch site names begin with `CCA`

- Find all launch sites begin with `CCA`

- 

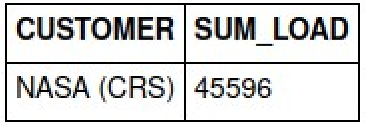
Get Result:

Date	TIME_*UTC*	BOOSTER_VERSION	LAUNCH_SITE	PAYLOAD	PAYLOAD_MASS_KG	ORBIT	CUSTOMER	MISSION_OUTCOME	LANDING_SITE
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

Total payload mass

- Calculate the total payload carried by boosters from NASA

- **Get Result:**



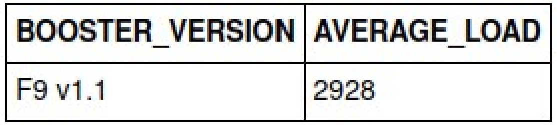
CUSTOMER	SUM_LOAD
NASA (CRS)	45596

Average payload mass by F9 v1.1

- Calculate the average payload mass carried by booster version F9 v1.1

-

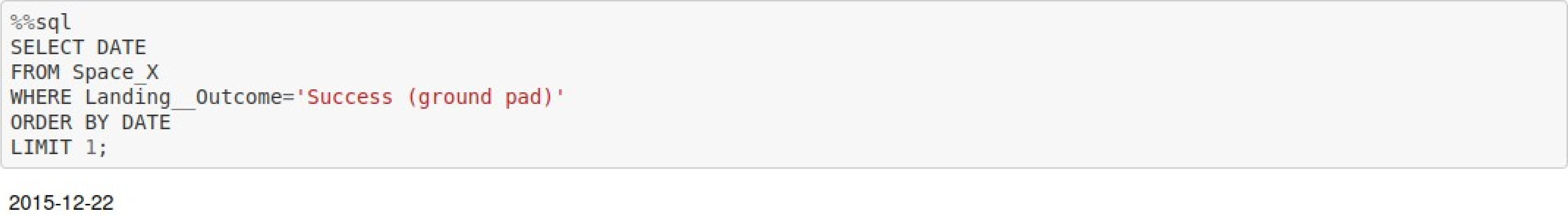
  Get Result:



BOOSTER_VERSION	AVERAGE_LOAD
F9 v1.1	2928

First successful ground landing date

- Find the date when the first successful landing outcome in ground pad

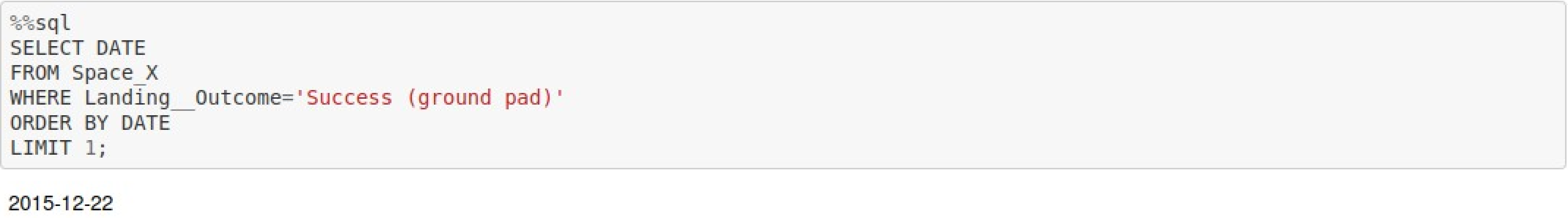
- 

```
%%sql
SELECT DATE
FROM Space_X
WHERE Landing__Outcome='Success (ground pad)'
ORDER BY DATE
LIMIT 1;
```

```
2015-12-22
```

Successful drone ship landing with payload between 4000 and 6000

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

- 

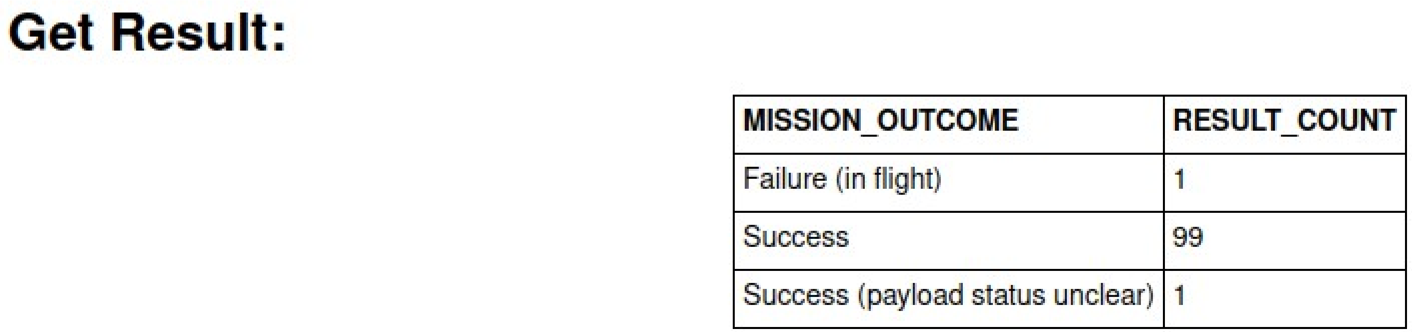
```
%%sql
SELECT DATE
FROM Space_X
WHERE Landing__Outcome='Success (ground pad)'
ORDER BY DATE
LIMIT 1;
```

```
2015-12-22
```

Total number of successful and failure mission outcomes

- Calculate the total number of successful and failure mission outcomes

-

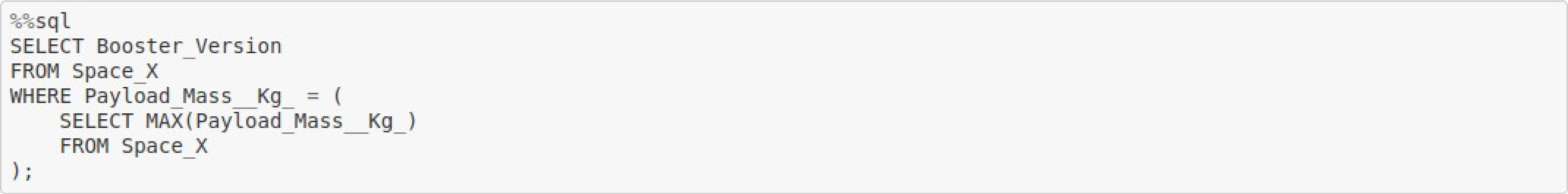


Get Result:

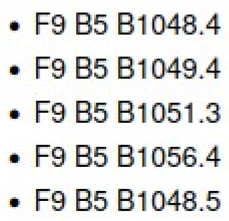
MISSION_OUTCOME	RESULT_COUNT
Failure (in flight)	1
Success	99
Success (payload status unclear)	1

Boosters carried maximum payload

- List the names of the booster which have carried the maximum payload mass



```
%%sql
SELECT Booster_Version
FROM Space_X
WHERE Payload_Mass__Kg_ = (
    SELECT MAX(Payload_Mass__Kg_)
    FROM Space_X
);
```

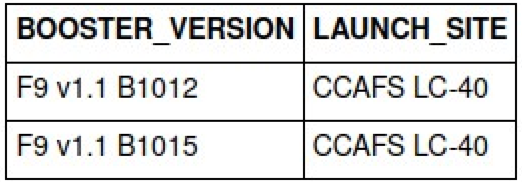


- F9 B5 B1048.4
- F9 B5 B1049.4
- F9 B5 B1051.3
- F9 B5 B1056.4
- F9 B5 B1048.5

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

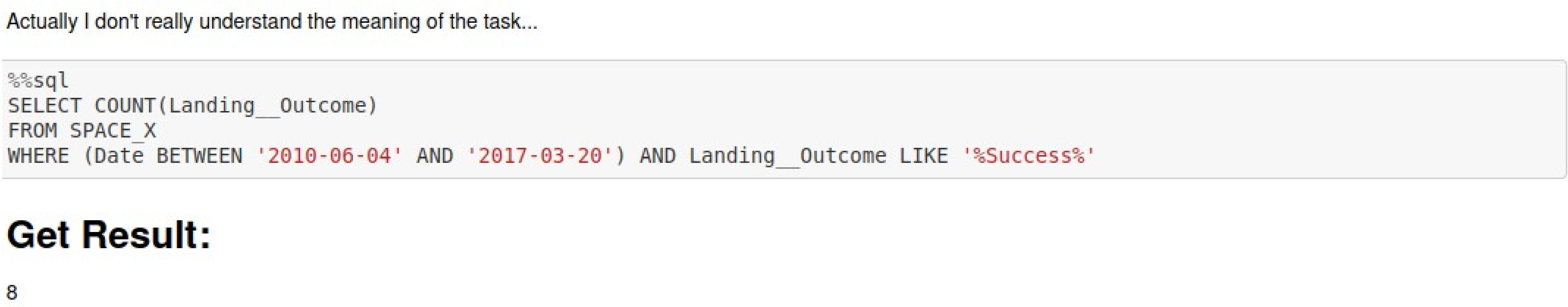
Get Result:



BOOSTER_VERSION	LAUNCH_SITE
F9 v1.1 B1012	CCAFS LC-40
F9 v1.1 B1015	CCAFS LC-40

Rank success count between 2010-06-04 and 2017-03-20

- Rank the count of successful landing_outcomes between the date 2010-06-04 and 2017-03-20 in descending order.
-



Actually I don't really understand the meaning of the task...

```
%%sql
SELECT COUNT(Landing__Outcome)
FROM SPACE_X
WHERE (Date BETWEEN '2010-06-04' AND '2017-03-20') AND Landing__Outcome LIKE '%Success%'
```

Get Result:

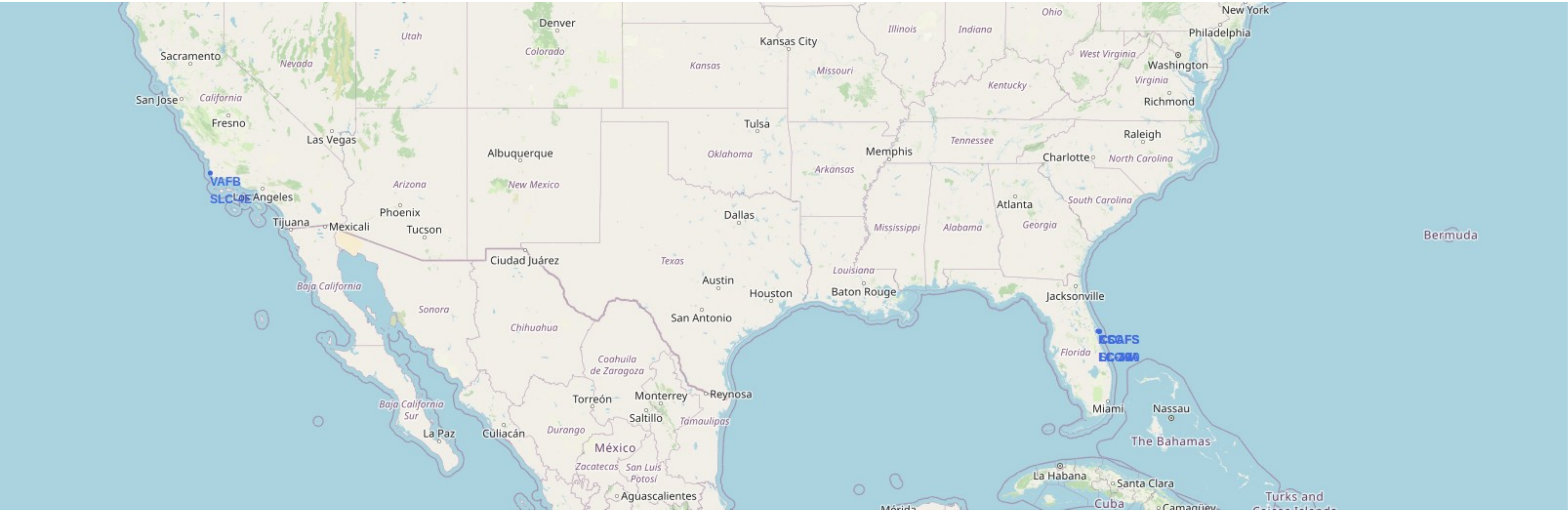
```
8
```

Interactive map with Folium

URL

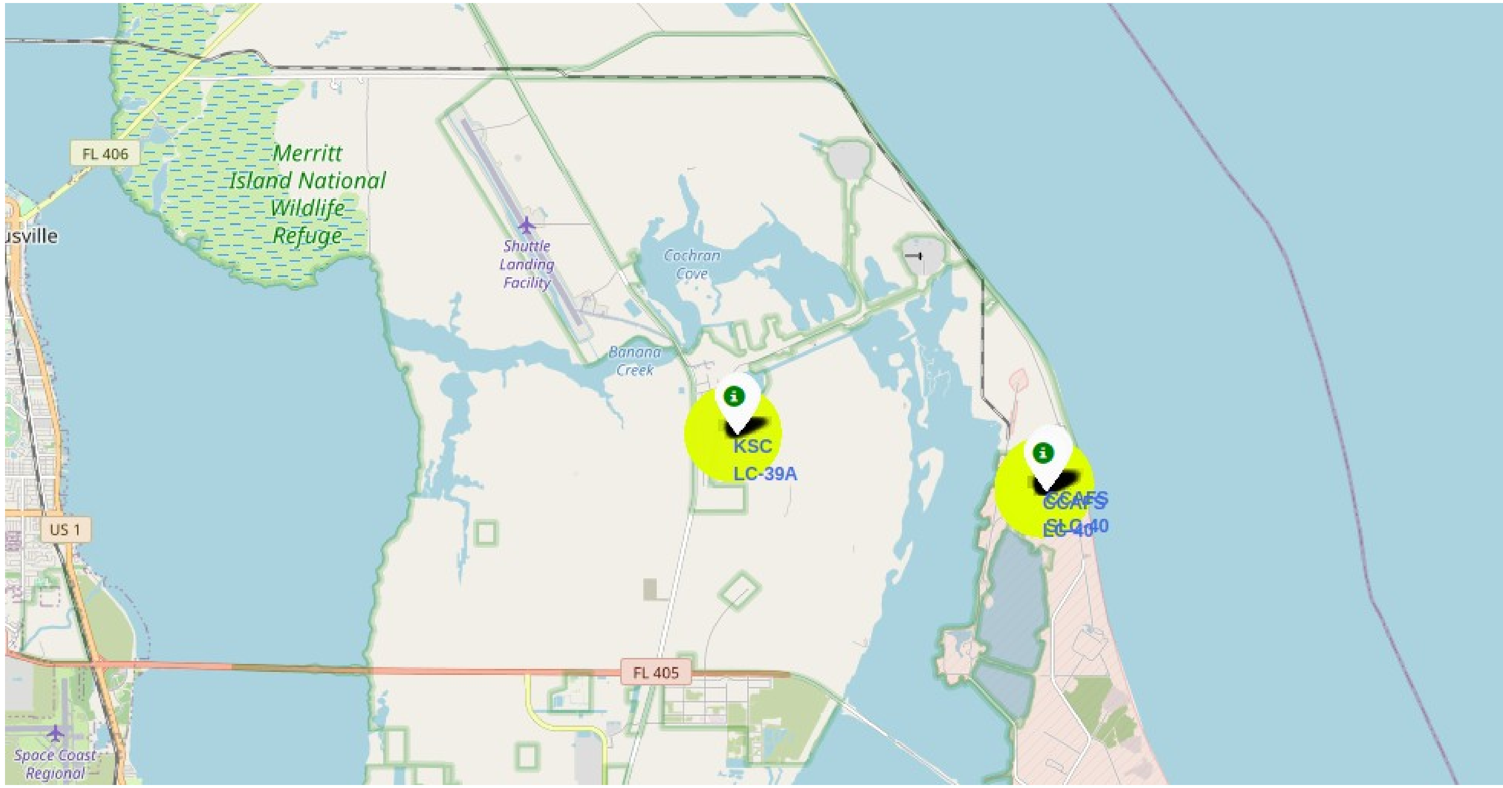
Launch Sites with name

- We can see that all these sites are located on the coast.



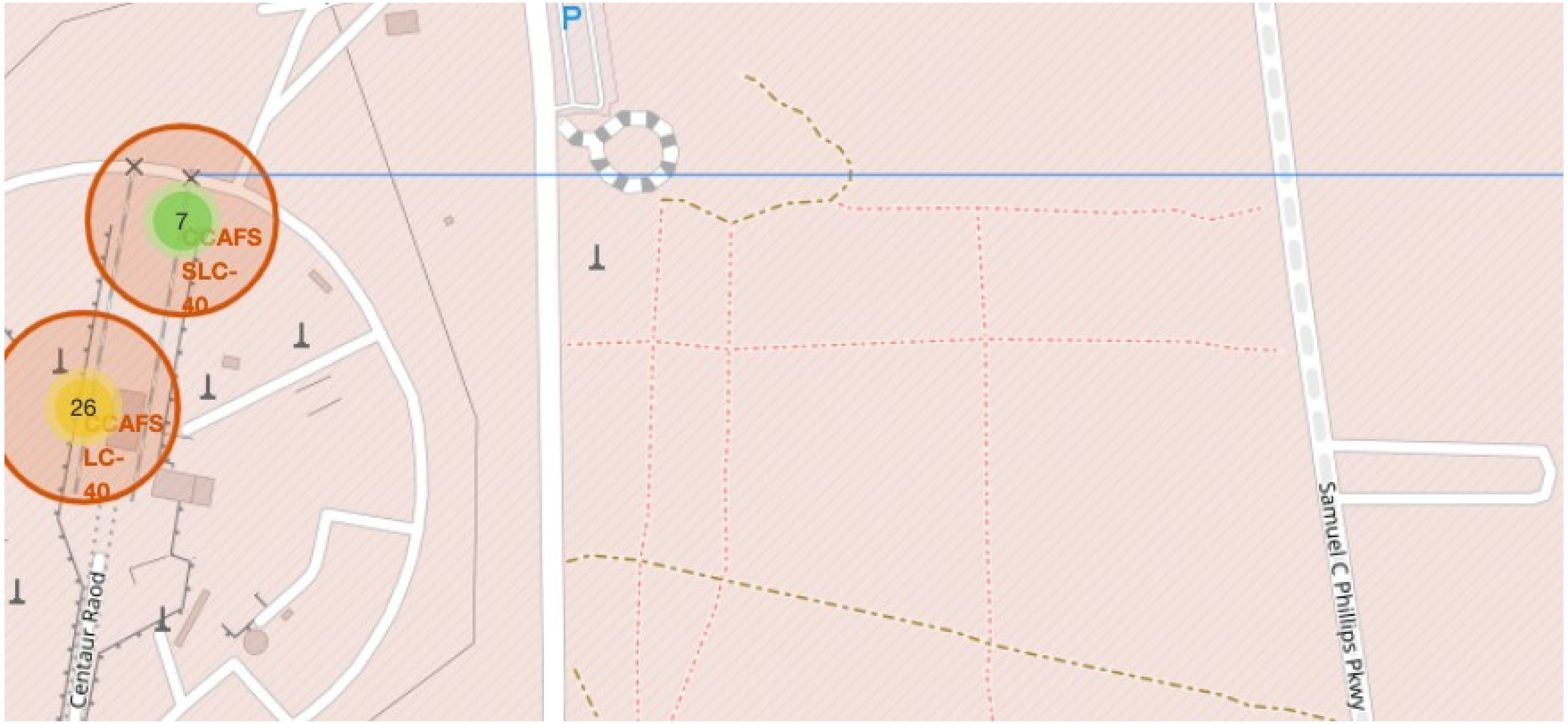
Launch Site with icon

- From the previous file, we can only see 3 sites, but actually there are multiple small sites in the 3 places.



Launch Site with distance

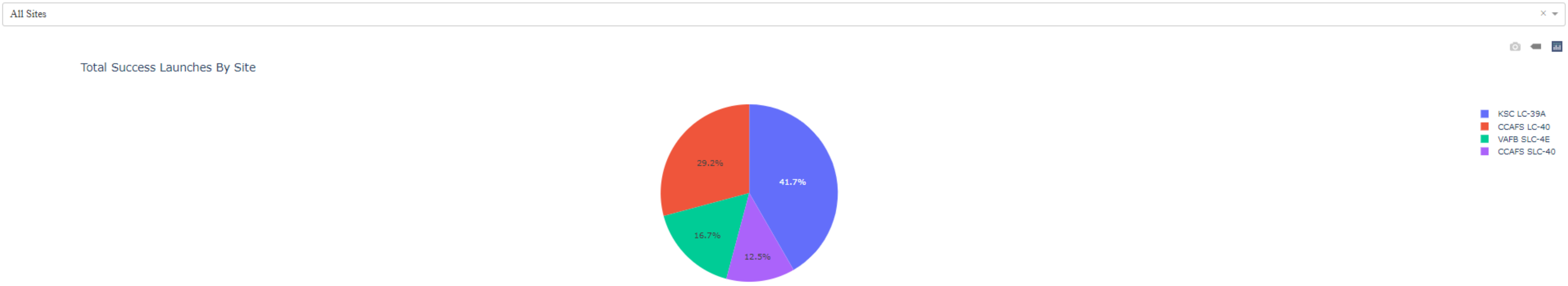
- Now we can use the function to compute the distance an annotate in the map.



Build a Dashboard with Plotly Dash

Success Ration – Launch Site

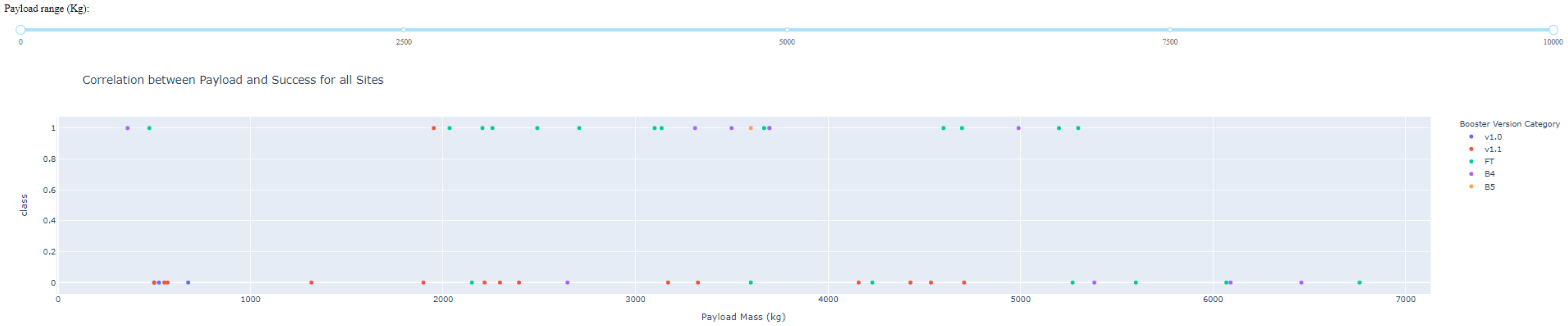
- The most successful sites seems to be KSC and CCAFS



All Sites

PayLoad – Launch Outcome

- It seems that Payload in the middle has the higher success rate, e.g. between 2000-5000 kg



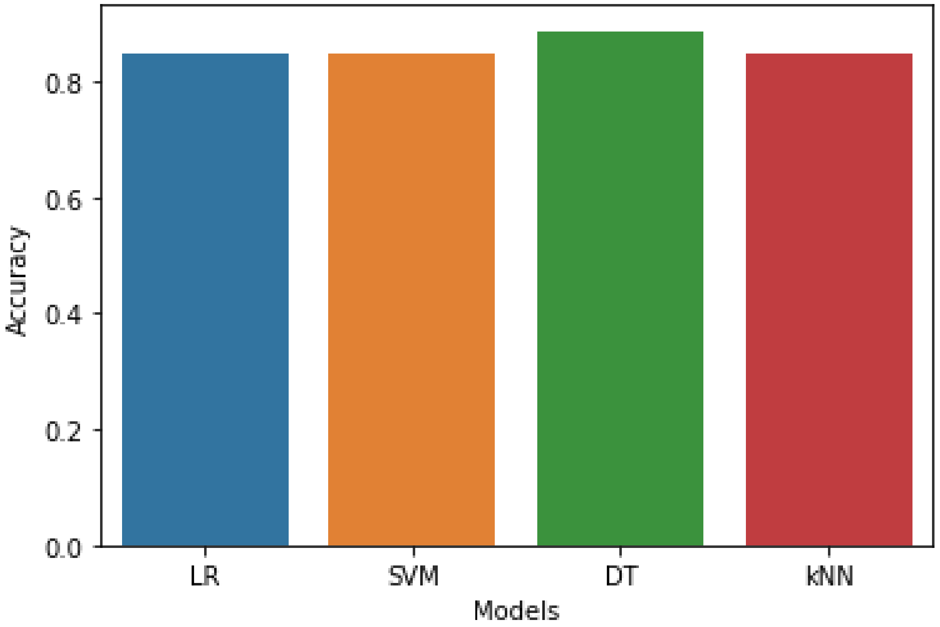
Predictive analysis (Classification)

URL

Classification Accuracy

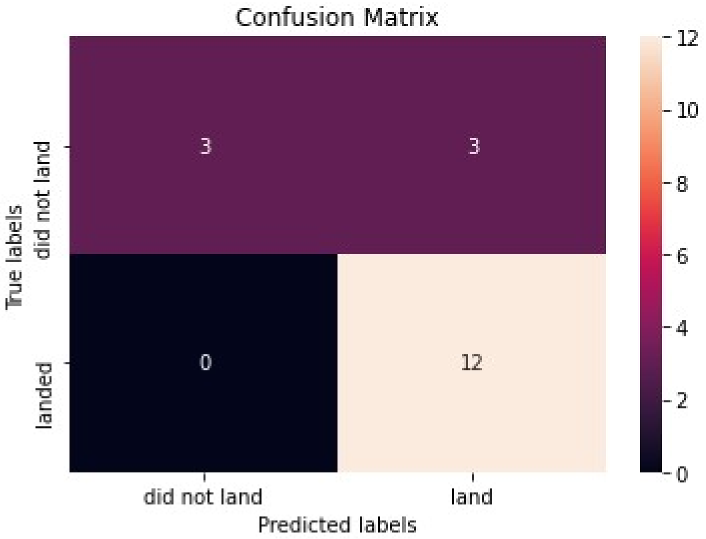
Visualize all the built model accuracy for all built models, in a barchart

All models have the same score, DT model has the highest accuracy (0.8875)



Confusion Matrix

Since there are only 18 test samples, almost all the model show the same quality, with the best confusion matrix as right.



CONCLUSION

- We have extrace information from API and convert it to the pandas Dataframe.
- Use Folium, we can visualize launch site on the maps.
- Using the dashboard, we can achieve a good interactivity.
- In the ML part, we see that all models are very strong, in this case, DT seems the most suitable for out analysis.

APPENDIX

- Pandas Reference
- Seaborn Reference
- Sklearn Reference