IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Ali Morovati Pasand
July 01, 2022

Outline

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

Executive Summary

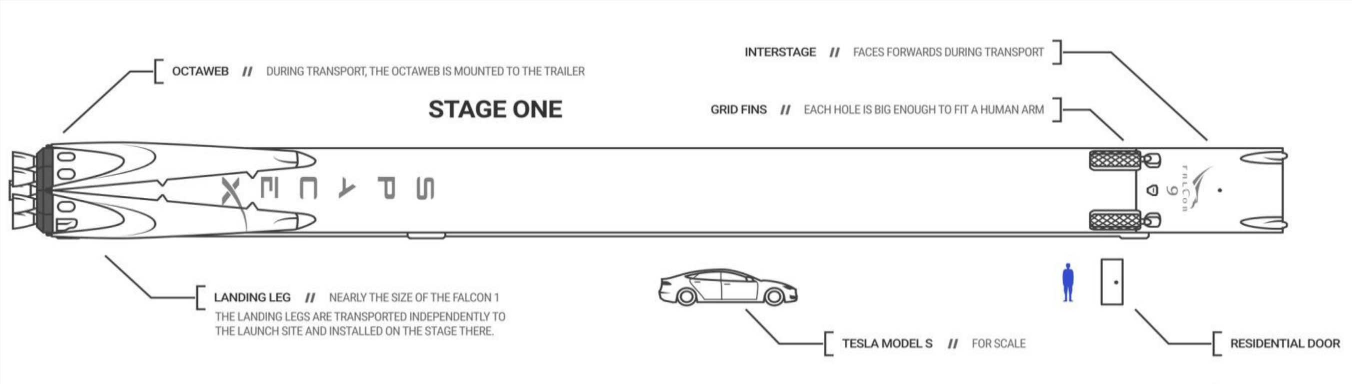
- Summary of methodologies
  - ✓ Data collection
  - ✓ Data Wrangling
  - ✓ EDA (Exploratory Data Analysis) and Data visualization
  - ✓ Predictive analysis
- Summary of all results
  - ✓ The success rate of SpaceX successful landing kept increasing since 2013
  - ✓ We could predict the outcome of new with 83% accuracy

Introduction

- Project background and context

Companies are making space travel affordable for everyone. Virgin Galactic is providing suborbital spaceflights. Rocket Lab is a small satellite provider. Blue Origin manufactures sub-orbital and orbital reusable rockets. Perhaps the most successful is SpaceX.

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



- Problems we want to find answers for

If we can determine if the first stage will land, we can determine the cost of a launch.

So we are going to predict whether the first stage lands.

Section 1

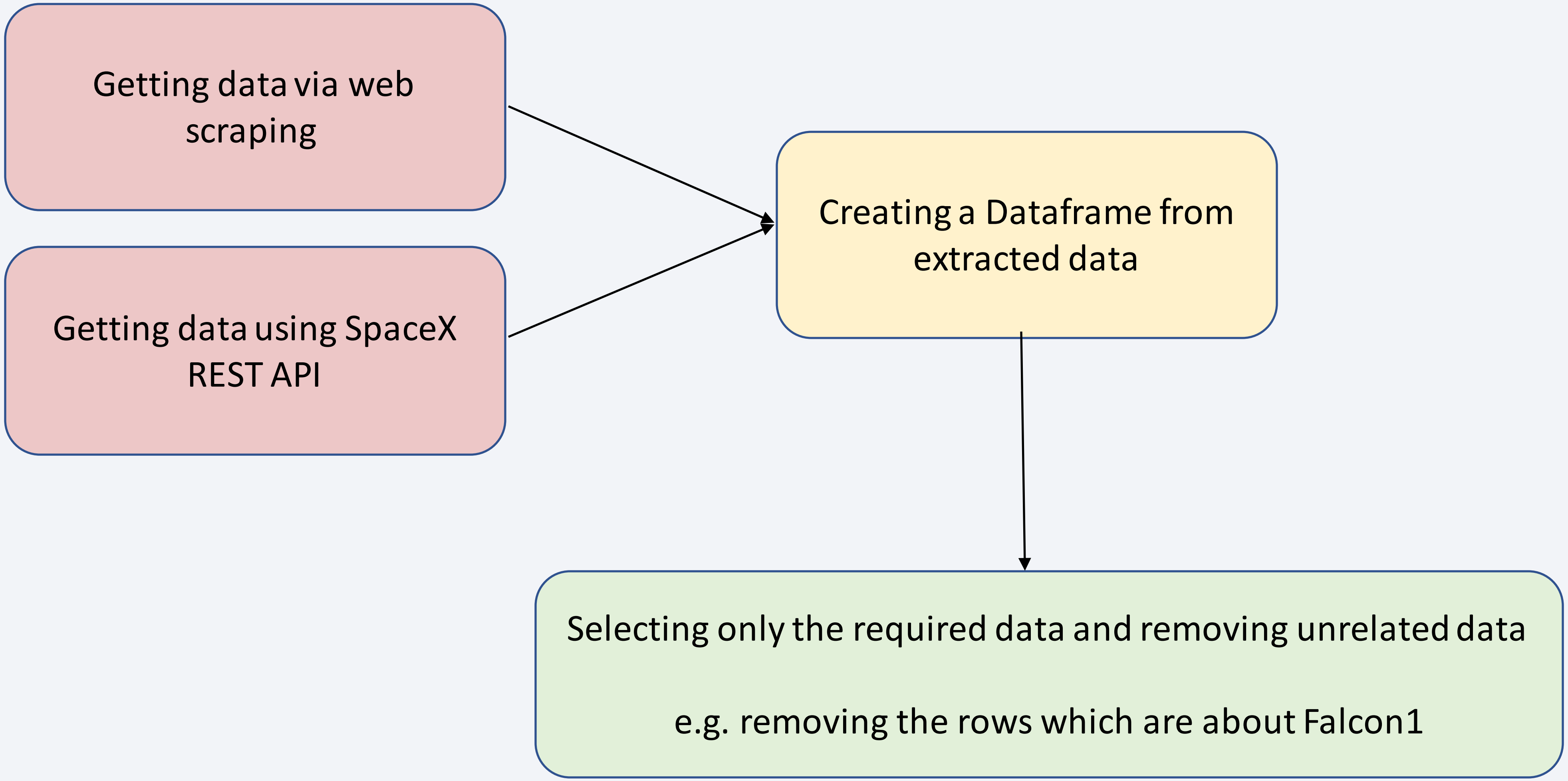
Methodology

Methodology

Executive Summary

- Data collection methodology:
  - Using SpaceX Rest API with Python
  - Web scraping related Wiki pages Python
- Perform data wrangling
  - Using Python pandas and numpy libraries for Data Wrangling
- 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

Data Collection



Data Collection – SpaceX API

1. Request and parse the SpaceX launch data using the GET request
2. Filter the dataframe to only include **Falcon 9** launches
3. relace None values in the **PayloadMass** with the mean

- GitHub URL of the completed SpaceX API calls notebook:

https://github.com/Amorovati/IBM_Data_Science_Capstone/blob/main/jupyter_labs_spacex_data_collection_api.ipynb

In [9]:
```
static_json_url='https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBM-DS0321EN-SkillsNetwork/datasets/API_call_spacex_api.json'
response = requests.get(static_json_url)
```

We should see that the request was successfull with the 200 status response code

In [10]:
```
response.status_code
```

Out[10]:
```
200
```

In [10]:

Now we decode the response content as a Json using `.json()` and turn it into a Pandas dataframe using `.json_normalize()`

In [11]:
```
# Use json_normalize meethod to convert the json result into a dataframe
data=pd.json_normalize(response.json())
```

Using the dataframe `data` print the first 5 rows

Data Collection - Scraping

Web scrap Falcon 9 launch records with BeautifulSoup:

- Extract a Falcon 9 launch records HTML table from Wikipedia
- Parse the table and convert it into a Pandas data frame

- GitHub URL of the completed web scraping notebook:

https://github.com/Amorovati/IBM_Data_Science_Capstone/blob/main/jupyter_labs_webscraping.ipynb

In [70]:
```
static_url = "https://en.wikipedia.org/w/index.php?title=List_of_Falcon_9_and_Falcon_Heavy_launches&oldid=1027686922"
```

Next, request the HTML page from the above URL and get a `response` object

TASK 1: Request the Falcon9 Launch Wiki page from its URL

First, let's perform an HTTP GET method to request the Falcon9 Launch HTML page, as an HTTP response.

In [71]:
```
# use requests.get() method with the provided static_url
# assign the response to a object
response=requests.get(static_url)
```

In [72]:
```
html=response.content
```

Create a `BeautifulSoup` object from the HTML `response`

In [73]:
```
# Use BeautifulSoup() to create a BeautifulSoup object from a response text content
soup=BeautifulSoup(html,'html.parser')
```

In [73]:

Print the page title to verify if the `BeautifulSoup` object was created properly

In [74]:
```
# Use soup.title attribute
soup.title
```

Out[74]:
```
<title>List of Falcon 9 and Falcon Heavy launches - Wikipedia</title>
```

Data Wrangling

- We are going to get a better insight from our Data and finally convert the outcomes into Training Labels with 1 means the booster successfully landed and 0 means it was unsuccessful.

Calculate the number of launches on each site → Calculate the number and occurrence of each orbit → Calculate the number and occurrence of mission outcome per orbit type → **Create a landing outcome label from Outcome column**

•GitHub URL of the completed Data Wrangling notebook:

https://github.com/Amorovati/IBM_Data_Science_Capstone/blob/main/labs_jupyter_spacex_Data_wrangling.ipynb

EDA with Data Visualization

- We have plotted some charts as below to have a preliminary insights about how each important variable (feature) would affect the success rate. Then, we will select the features.

The charts are:

1- Relationship between Flight Number and Launch Site,

2- Relationship between Payload and Launch Site

3- success rate of each orbit type

4- Relationship between Flight Number and Orbit type

5- Relationship between Payload and Orbit type

6- launch success yearly trend

•GitHub URL of the completed EDA with data visualization notebook:

https://github.com/Amorovati/IBM_Data_Science_Capstone/blob/main/jupyter_labs_eda_dataviz.ipynb

EDA with SQL

- Performed SQL queries

- `select distinct (LAUNCH_SITE) from SPACEXTBL`
- `select * from SPACEXTBL where LAUNCH_SITE LIKE 'CCA%' limit 5`
- `select SUM(payload_mass__kg_) from SPACEXTBL where customer='NASA (CRS)'`
- `select AVG(payload_mass__kg_) from SPACEXTBL where booster_version='F9 v1.1';`
- `select min(DATE) from SPACEXTBL where landing__outcome='Success (ground pad)'`
- `select booster_version, payload_mass__kg_ from SPACEXTBL where landing__outcome='Success (drone ship)' and payload_mass__kg_ between 4000 and 6000;`
- `select mission_outcome, count(mission_outcome) as total_number from SPACEXTBL group by (mission_outcome)`
- `select booster_version, payload_mass__kg_ from SPACEXTBL where payload_mass__kg_=(select max(payload_mass__kg_) from SPACEXTBL)`
- `select substr(DATE,6,2) as month,landing__outcome, booster_version, launch_site from SPACEXTBL where substr(DATE,1,4)='2015' and landing__outcome='Failure (drone ship)'`
- `select count(landing__outcome) as Number_of_Successful_Landing from SPACEXTBL where landing__outcome in ('Success','Success (drone ship)','Success (ground pad)') and DATE between '2010-06-04' and '2017-03-20' group by landing__outcome order by Number_of_Successful_Landing DESC`

- GitHub URL of the completed EDA with SQL notebook:

https://github.com/Amorovati/IBM_Data_Science_Capstone/blob/main/jupyter_labs_eda_sql_coursera_sqllite.ipynb

Build an Interactive Map with Folium

- We have added a circle for each launch site
- We have marked the success/failed launches for each site on the map using MarkerCluster

Note that a launch only happens in one of the four launch sites, which means many launch records wil l have the exact same coordinate. Marker clusters can be a good way to simplify a map containing man y markers having the same coordinate.

- We have Calculated the distances between a launch site to its proximities and drawn a line between them

- GitHub URL of the completed interactive map with Folium map

https://github.com/Amorovati/IBM_Data_Science_Capstone/blob/main/lab_jupyter_launch_site_location.ipynb

Build a Dashboard with Plotly Dash

- We have added a Launch Site Drop-down Input Component
- We have added a callback function to render `success-pie-chart` based on selected site dropdown
- We have added a Range Slider to Select Payload
- We have added a callback function to render the `success-payload-scatter-chart` scatter plot

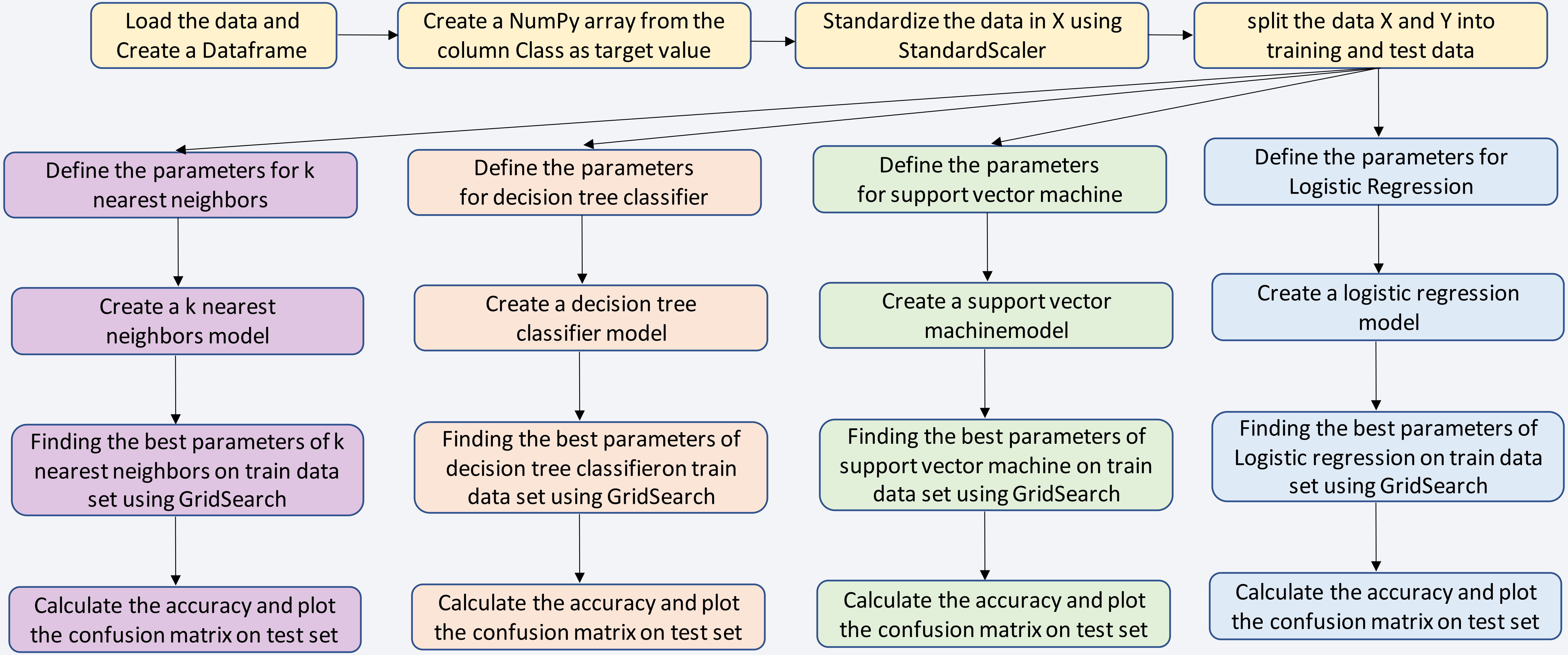
With these interactive visualization, one should be able to obtain some insights to answer the following five questions:

1. Which site has the largest successful launches?
2. Which site has the highest launch success rate?
3. Which payload range(s) has the highest launch success rate?
4. Which payload range(s) has the lowest launch success rate?
5. Which F9 Booster version (`v1.0`, `v1.1`, FT, B4, B5, etc.) has the highest

- GitHub URL of the completed Plotly Dash lab:

https://github.com/Amorovati/IBM_Data_Science_Capstone/blob/main/spacex_dash_app.py

Predictive Analysis (Classification)



•GitHub URL of the completed Plotly Dash lab:
https://github.com/Amorovati/IBM_Data_Science_Capstone/blob/main/SpaceX_Machine_Learning_Prediction_Part_5.ipynb

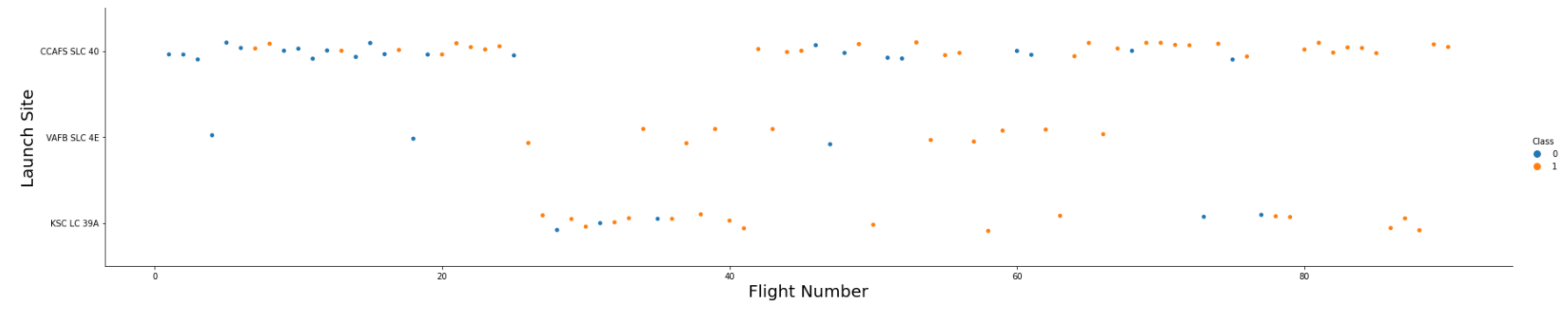
Results

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

Section 2

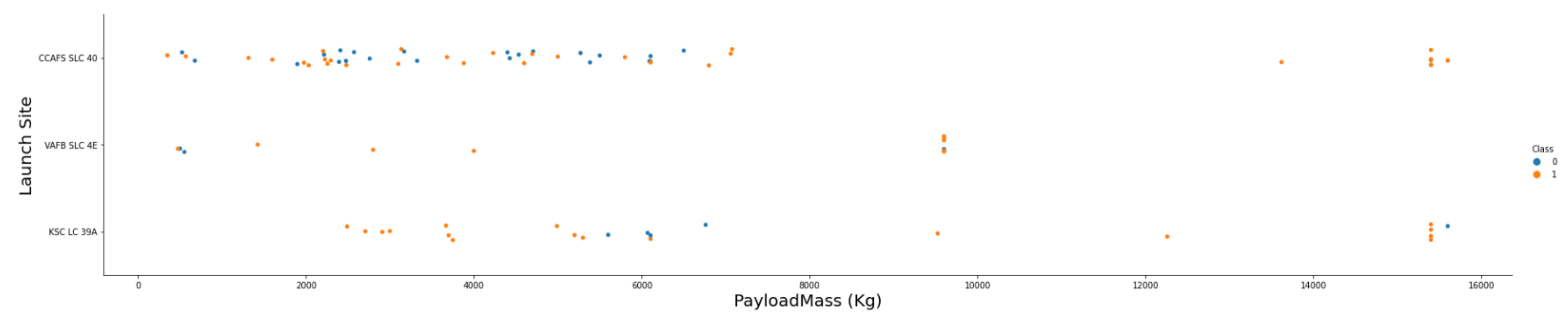
Insights drawn from EDA

Flight Number vs. Launch Site



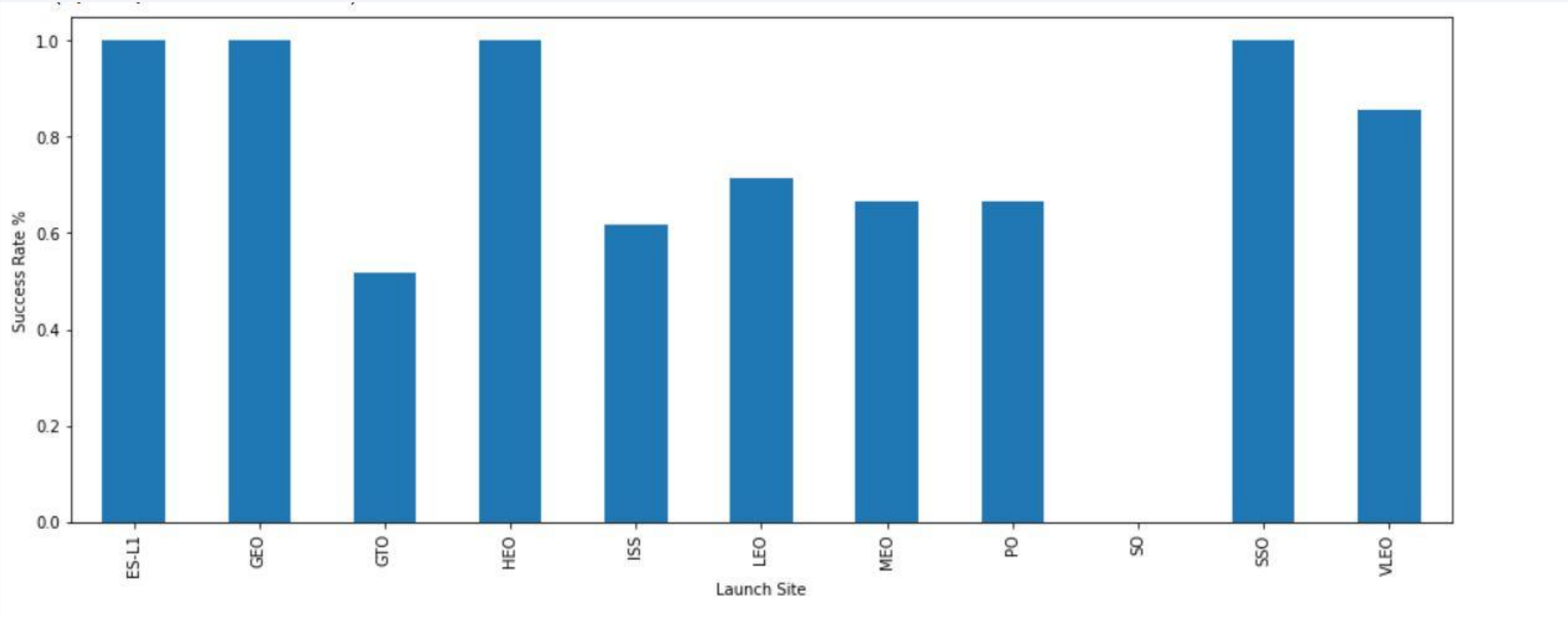
We see that different launch sites have different success rates. CCAFS LC-40, has a success rate of 60 %, while KSC LC-39A and VAFB SLC 4E has a success rate of 77%. It is also clear that by increasing the flight number, we have more successful landing.

Payload vs. Launch Site



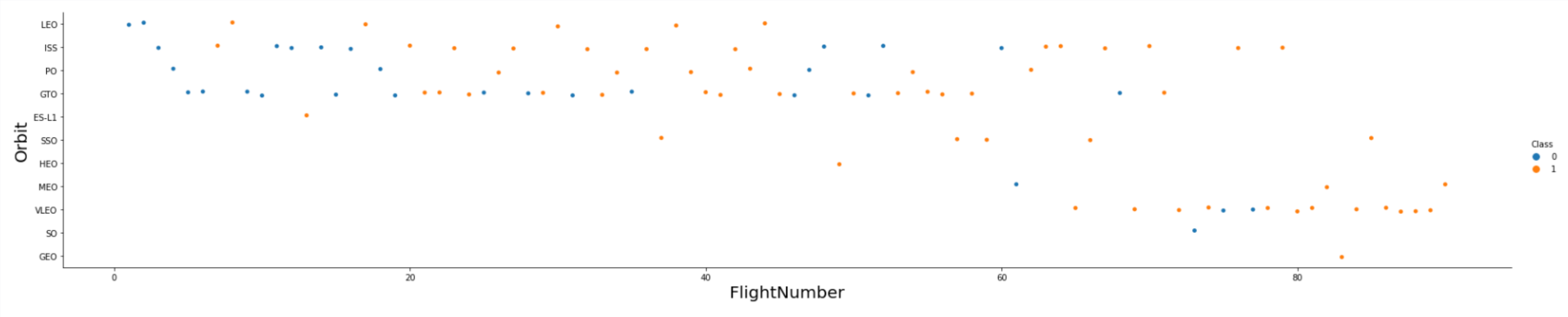
- Now if we observe Payload Vs. Launch Site scatter point chart you will find for the VAFB-SLC launch site there are no rockets launched for heavy payload mass(greater than 10000)
- We can see that with so heavy payload (greater than 10000), the chance of successful landing is very high.
- For the KSC LC 39A Launch Site with payload less than 5500 Kg we have a very good chance of successful landing

Success Rate vs. Orbit Type



We can see the ES-L1, GEO, HEO and SSO orbit types have the 100% success rate and SO orbit type has 0% success rate. It i
clear that VLEO orbit type has also good success rate (greater than 80%)

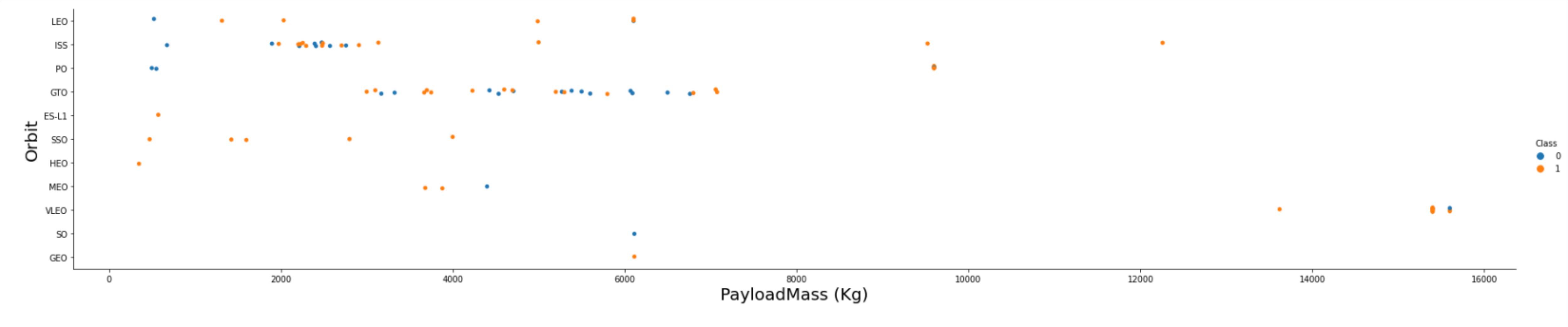
Flight Number vs. Orbit Type



We can see that in the LEO and VLEO orbit the Success appears related to the number of flights (the more the flight number is, the more success appears); on the other hand, there seems to be no relationship between flight number when in GTO orbit and ISS orbit.

Although we see the ES-L1, GEO, HEO and SSO orbits have the 100% success, there are few number of test in these orbits.

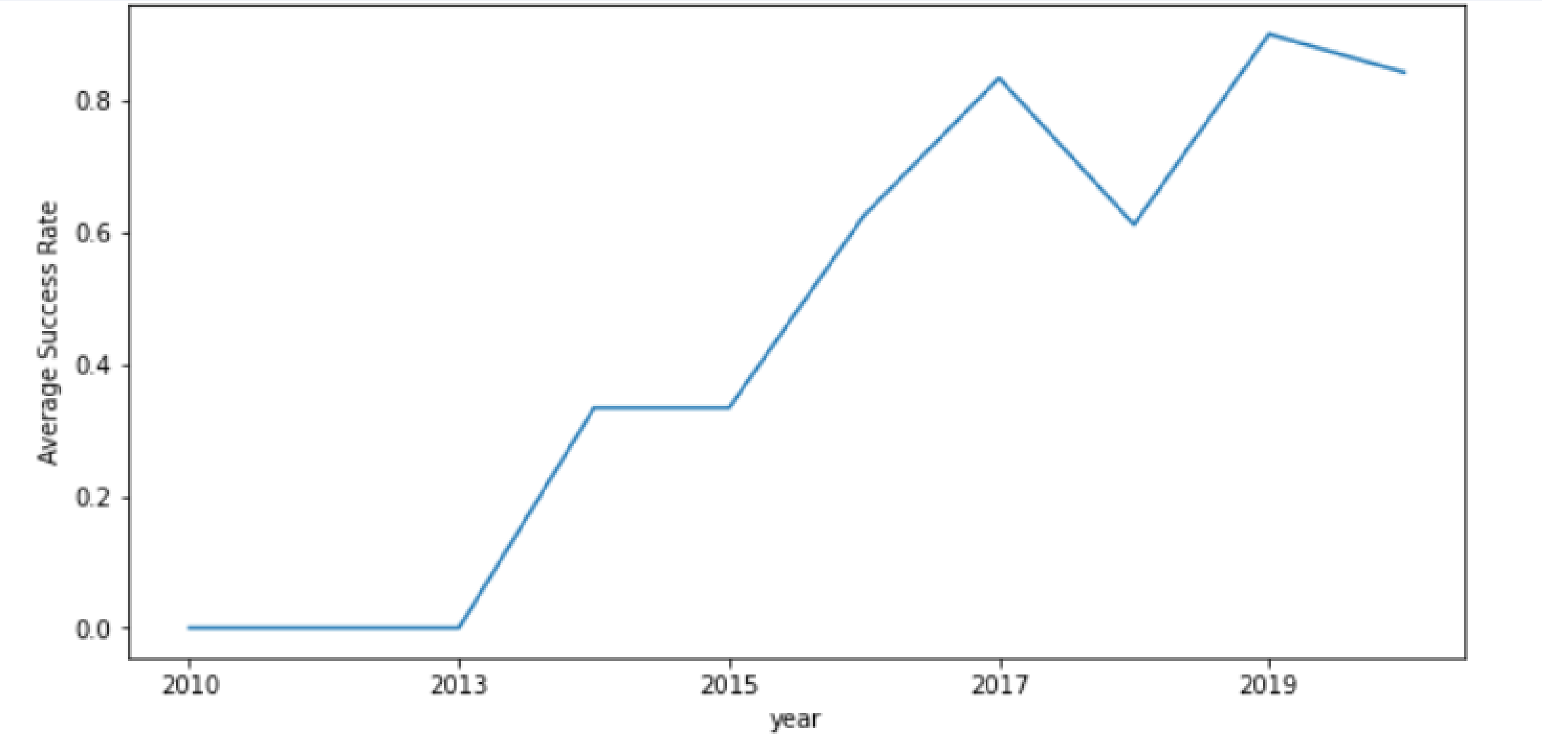
Payload vs. Orbit Type



With heavy payloads the successful landing or positive landing rate are more for Polar,LEO and ISS.

However for GTO we cannot distinguish this well as both positive landing rate and negative landing(unsuccessful mission) are both there.

Launch Success Yearly Trend



We can observe that the sucess rate since 2013 kept increasing till 2020

All Launch Site Names

- Names of the unique launch sites
- We use distinct function to specify unique values (here launch site)

```
[ ]  %sql select distinct (LAUNCH_SITE) from SPACEXTBL

 * ibm_db_sa://tst71274:***@9938aec0-8105-433e-8bf9
Done.
```

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

Launch Site Names Begin with 'CCA'

- Find 5 records where launch sites begin with `CCA`
- We use select with like 'CCA%' function. Please note that the % sign must be placed after the CCA.

```
[ ]  %sql select * from SPACEXTBL where LAUNCH_SITE LIKE 'CCA%' limit 5;

 * ibm_db_sa://tst71274:***@9938aec0-8105-433e-8bf9-0fbb7e483086.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud:32459/bludb
Done.
```

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

Total Payload Mass

- Calculate the total payload carried by boosters from NASA
- We use SUM function and where clause

```
[ ]  %sql select SUM(payload_mass__kg_) from SPACEXTBL where customer='NASA (CRS)';

 * ibm_db_sa://tst71274:***@9938aec0-8105-433e-8bf9-0fbb7e483086.c1ogj3sd0tgtu0lqd
Done.
  1
45596
```

Average Payload Mass by F9 v1.1

- Calculate the average payload mass carried by booster version F9 v1.1
- We use AVG(average) function and where clause

```
[ ]  %sql select AVG(payload_mass__kg_) from SPACEXTBL where booster_version='F9 v1.1';

 * ibm_db_sa://tst71274:***@9938aec0-8105-433e-8bf9-0fbb7e483086.c1ogj3sd0tgtu0lqde0
Done.
  1
 2928
```

First Successful Ground Landing Date

- Find the dates of the first successful landing outcome on ground pad
- We use min function and where clause

```
[ ]  %sql select min(DATE) from SPACEXTBL where landing__outcome='Success (ground pad)';

 * ibm_db_sa://tst71274:***@9938aec0-8105-433e-8bf9-0fbb7e483086.c1ogj3sd0tgtu0lqde00.data
Done.
      1
2015-12-22
```

Successful Drone Ship Landing with Payload between 4000 and 6000

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

- We use select and where clause with function between

```
[ ] %sql select booster_version, payload_mass__kg_ from SPACEXTBL where landing__outcome='Success (drone ship)' and payload_mass__kg_ between 4000 and 6000;

 * ibm_db_sa://tst71274:***@9938aec0-8105-433e-8bf9-0fbb7e483086.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud:32459/bludb
Done.
```

booster_version	payload_mass__kg_
F9 FT B1022	4696
F9 FT B1026	4600
F9 FT B1021.2	5300
F9 FT B1031.2	5200

Total Number of Successful and Failure Mission Outcomes

- Calculate the total number of successful and failure mission outcomes
- We use count function and group by clause

```
[ ] %sql select mission_outcome, count(mission_outcome) as total_number from SPACEXTBL group by (mission_outcome)
```

```
 * ibm_db_sa://tst71274:***@9938aec0-8105-433e-8bf9-0fbb7e483086.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud:32459/bludb
Done.
```

mission_outcome	total_number
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
- We use a subquery with max function and where clause

```
[ ]  %sql select booster_version, payload_mass__kg_ from SPACEXTBL where payload_mass__kg_=(select max(payload_mass__kg_) from SPACEXTBL)
```

```
 * ibm_db_sa://tst71274:***@9938aec0-8105-433e-8bf9-0fbb7e483086.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud:32459/bludb
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 failed landing_outcomes in drone ship, their booster versions, and launch site names for in year 2015
- We use substr function to get the month and year from the Date and where clause

```
[ ] %sql select substr(DATE,6,2) as month,landing__outcome, booster_version, launch_site from SPACEXTBL where substr(DATE,1,4)='2015' and landing__outcome='Failure (drone ship)'
```

```
 * ibm_db_sa://tst71274:***@9938aec0-8105-433e-8bf9-0fbb7e483086.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud:32459/bludb
Done.
```

MONTH	landing__outcome	booster_version	launch_site
01	Failure (drone ship)	F9 v1.1 B1012	CCAFS LC-40
04	Failure (drone ship)	F9 v1.1 B1015	CCAFS LC-40

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

- Rank 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
- We use count function and where clause, group by clause, order by clause and DESC (for descending order)

```
%%sql
select landing__outcome, count(landing__outcome) as Number_of_Landing from SPACEXTBL
 where landing__outcome in ('Success','Success (drone ship)','Success (ground pad)','Failure (drone ship)','Failure (parachute)')
 and DATE between '2010-06-04' and '2017-03-20'
 group by landing__outcome order by Number_of_Landing DESC
```

```
 * ibm_db_sa://tst71274:***@9938aec0-8105-433e-8bf9-0fbb7e483086.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud:32459/bludb
Done.
```

landing__outcome	number_of_landing
Failure (drone ship)	5
Success (drone ship)	5
Success (ground pad)	3
Failure (parachute)	2

Section 3

Launch Sites Proximities Analysis

Launch Site Locations on the Map

- In the following map you can see the locations of the four launch sites. Three of them (CCAFS LC-40, CCAFS SLC-40, KSC LC-39A) are very close to each other at the right side of the map and the other one (VAFB SLC-4E), you can find at the left side.

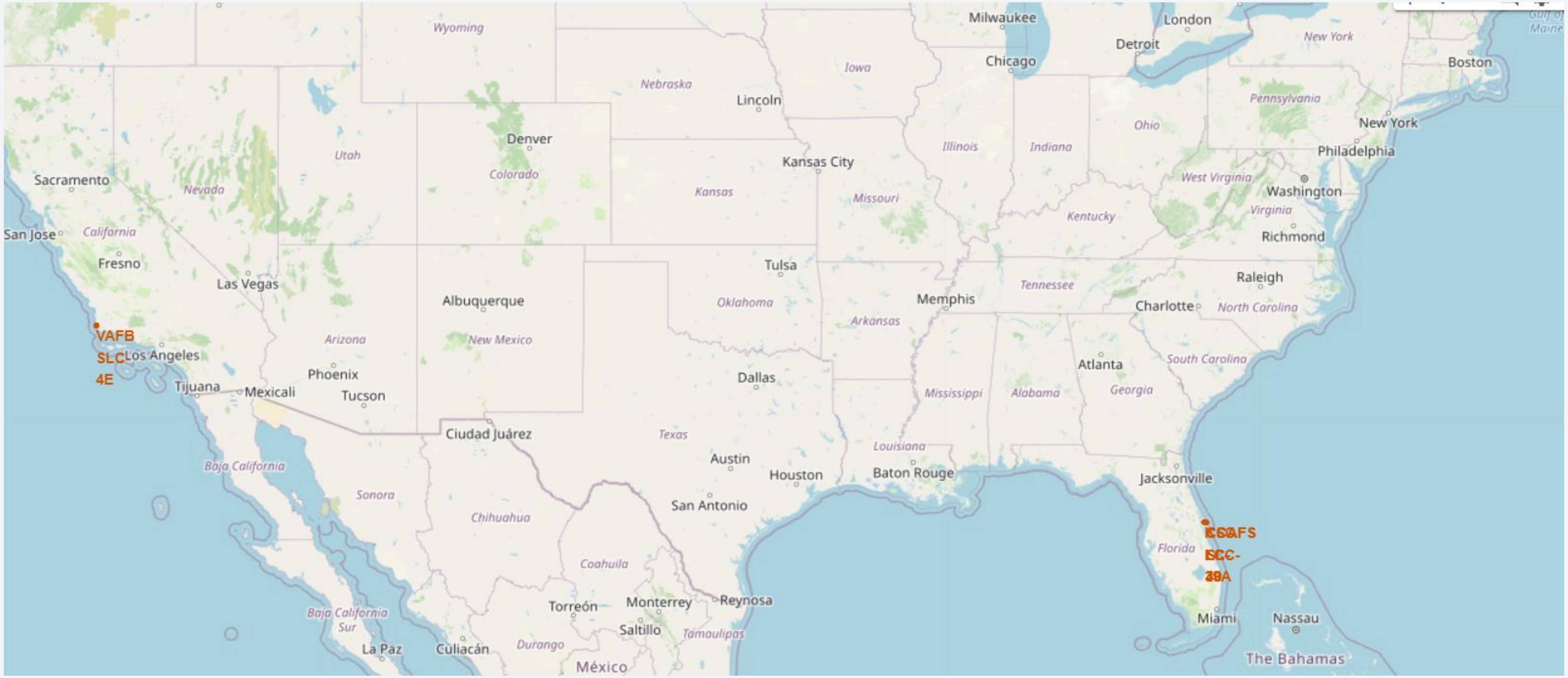
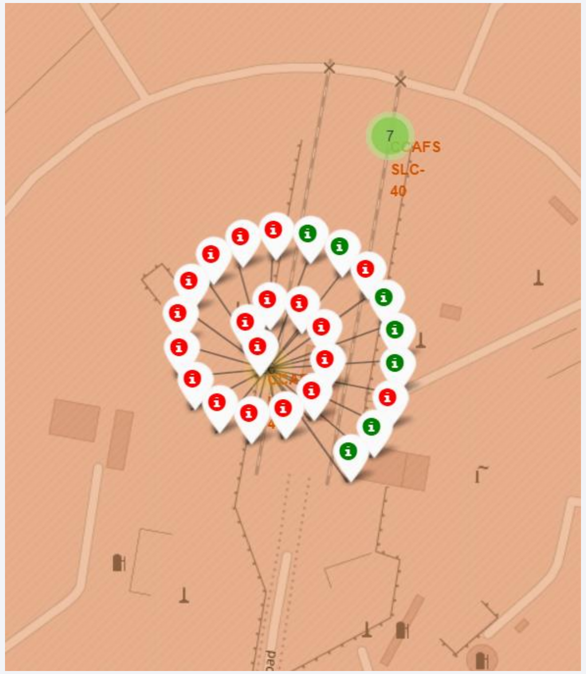
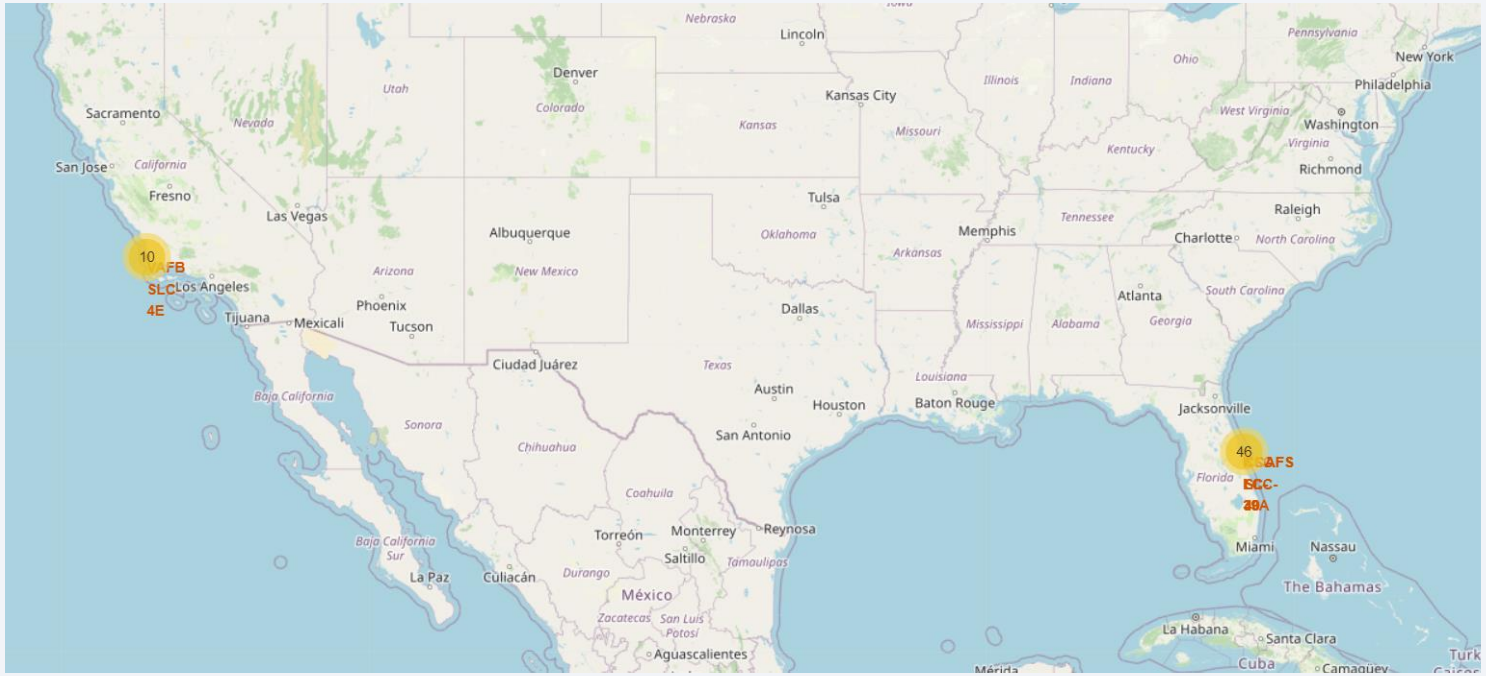


Illustration of different launch site landing records

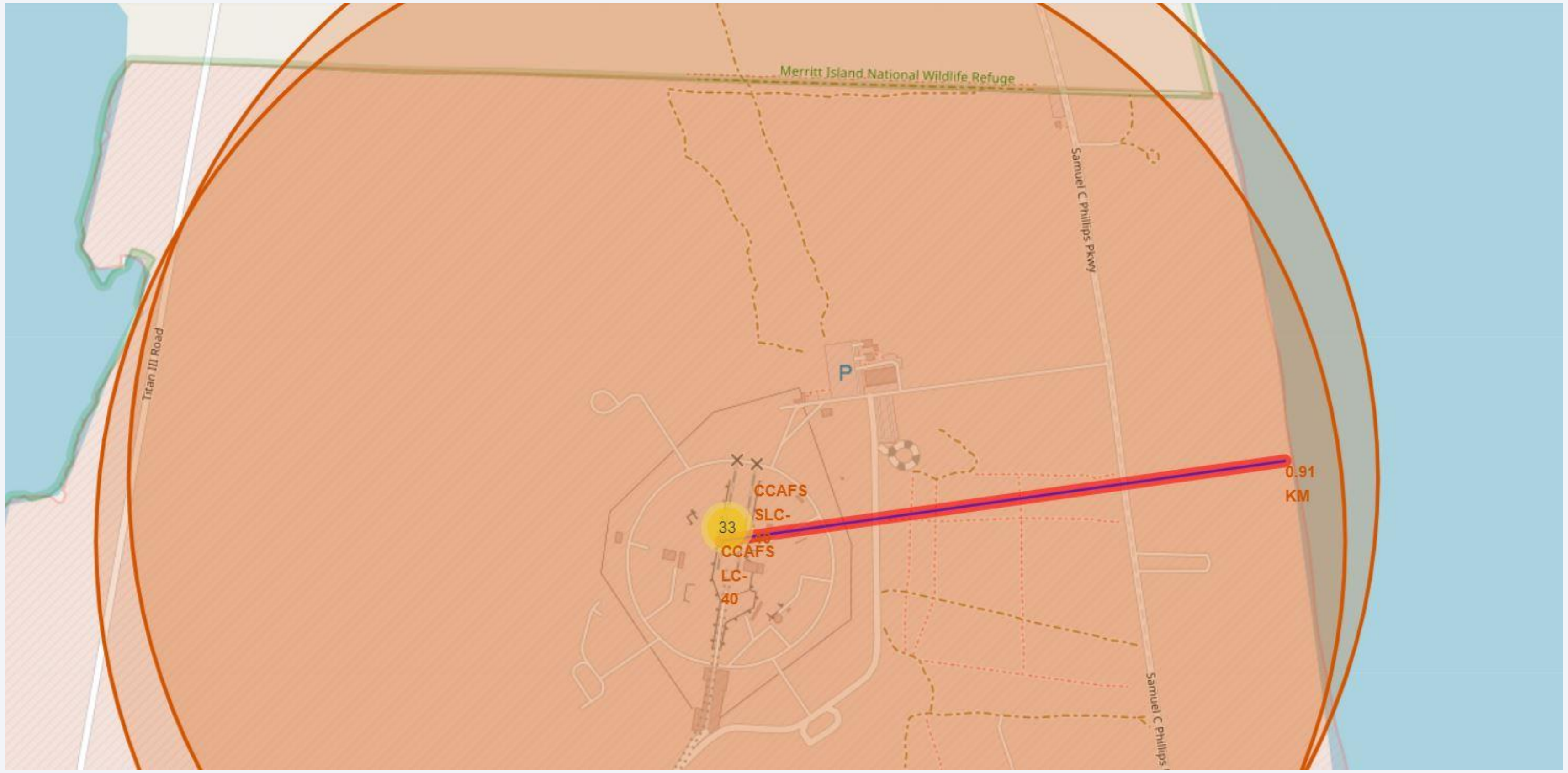
- You can see the different launch site landing records on the map. By clicking and zooming on the popup icon, you can see the landing results (green: successful and red: unsuccessful)





Distance of a Launch Site to its Proximity

- Here we have calculated the distance of the launch site CCAFS LC-40 to the coastline, which is 910 m, and drawn a line between them

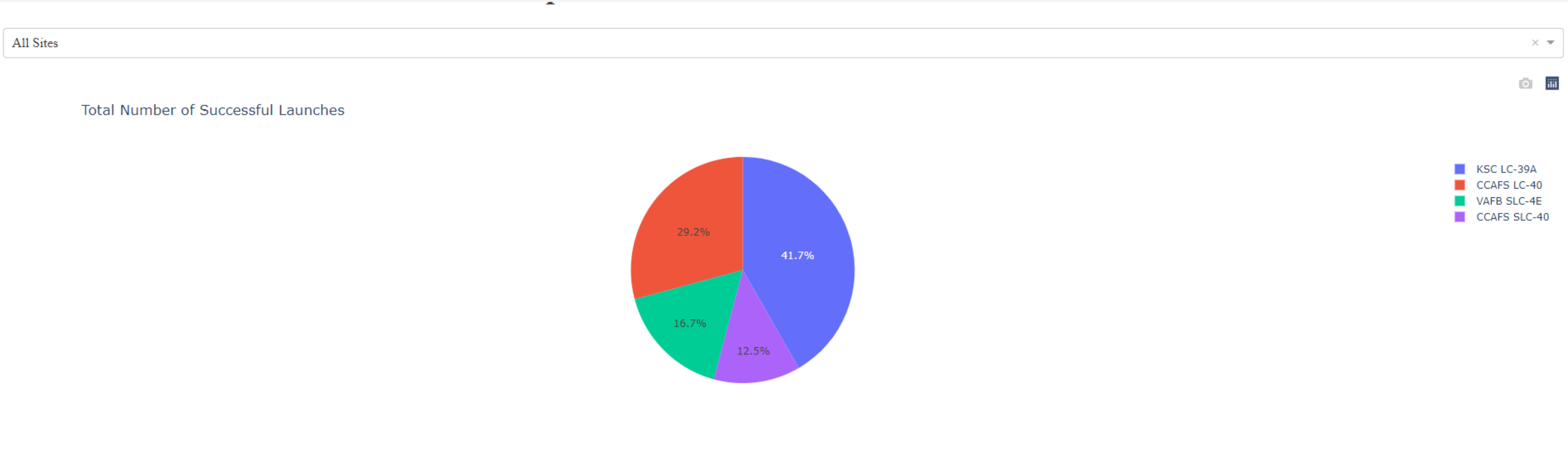


Section 4

Build a Dashboard with Plotly Dash

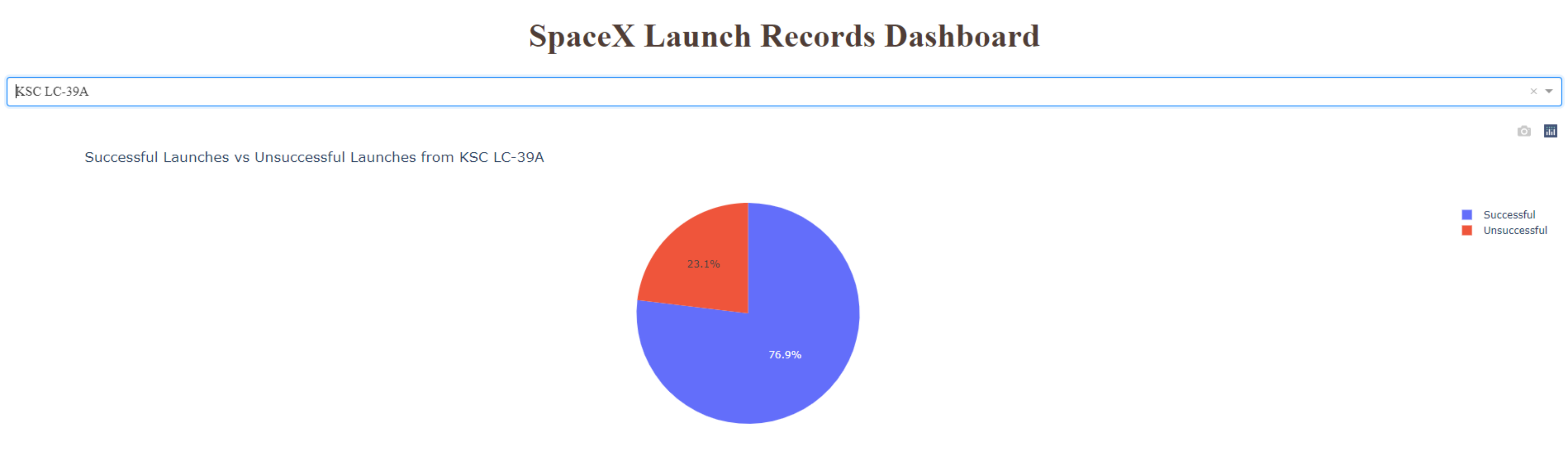
Dashboard- Success at Different Launch Sites

- You can select All Sites to see the successful launches of each site (percent in the pie-chart means the total number of successful launches of distinct site to the total number of successful launches) or select a launch site to compare the successful with unsuccessful launches
- In the piechart below, you see the launch site with the best success rate is KSC LC-39A



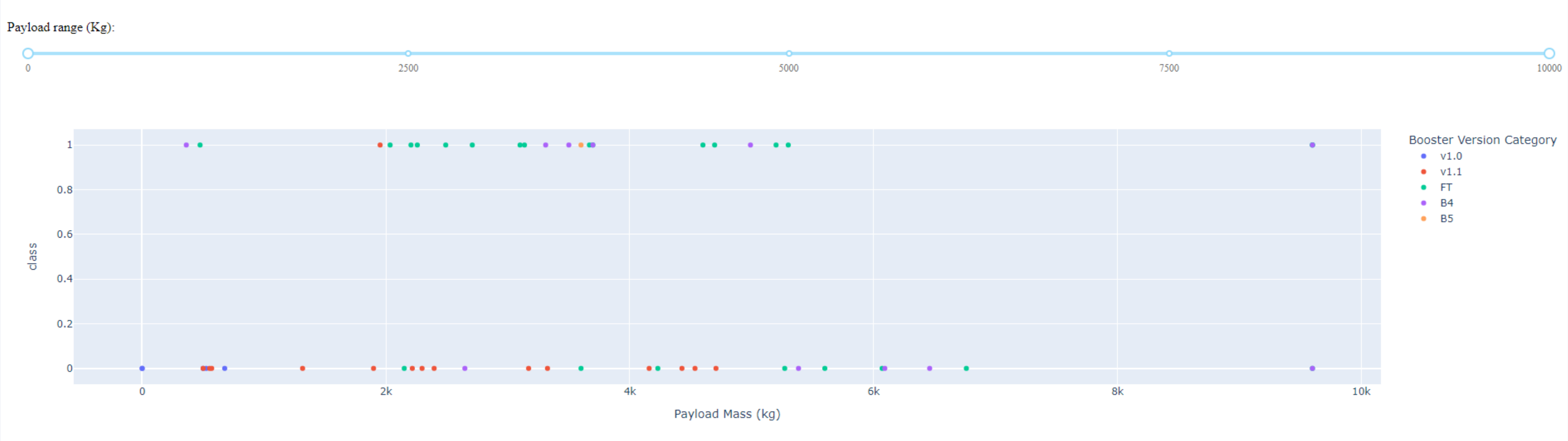
Dashboard- Launch Site with best Success Rate

- You can observe the best launch site according to success rate has 76.9% success rate



Dashboard- Payload vs Launch Outcome

- You can see that the booster versions FT and B4 have the most successful outcomes for the different Payloads and booster versions v1 and v1.1 have the lowest successful outcomes for the different Payloads.



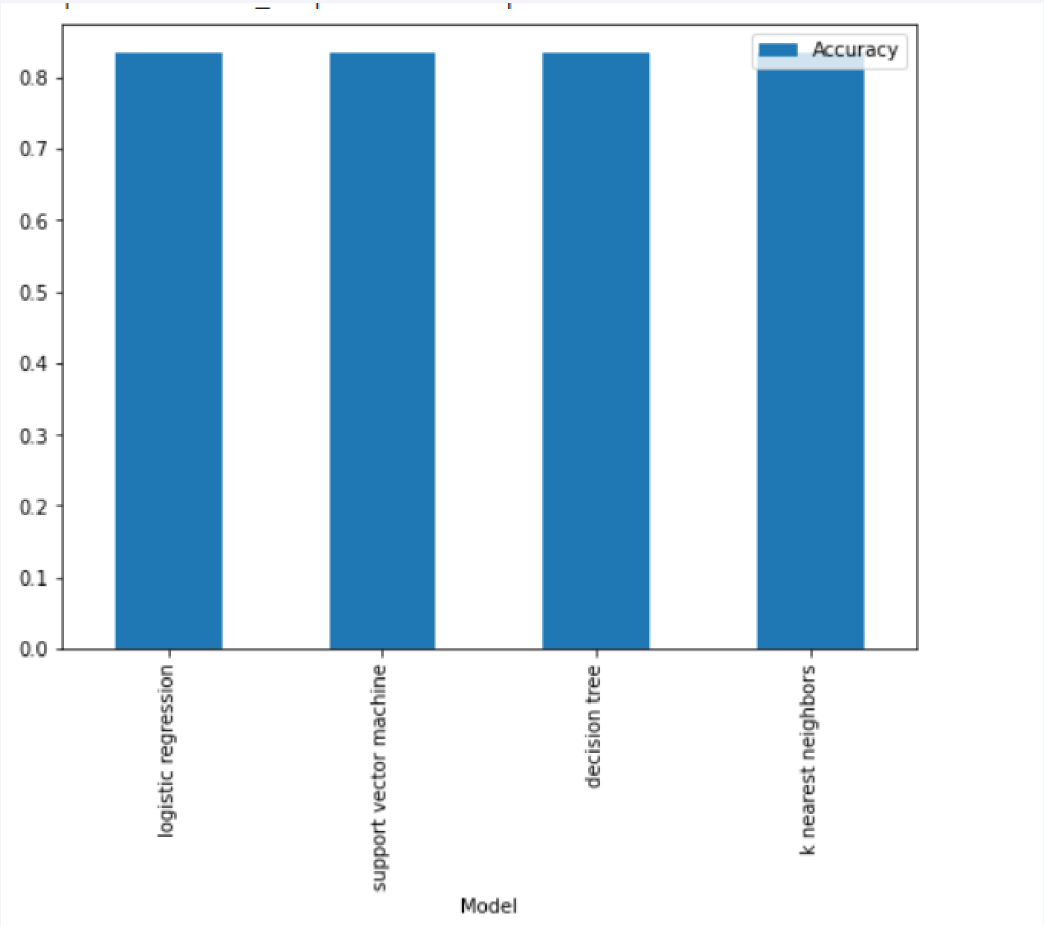
Section 5

Predictive Analysis (Classification)

Classification Accuracy

We have used four different model for prediction as follows:

- logistic regression
- support vector machine
- decision tree
- k nearest neighbors



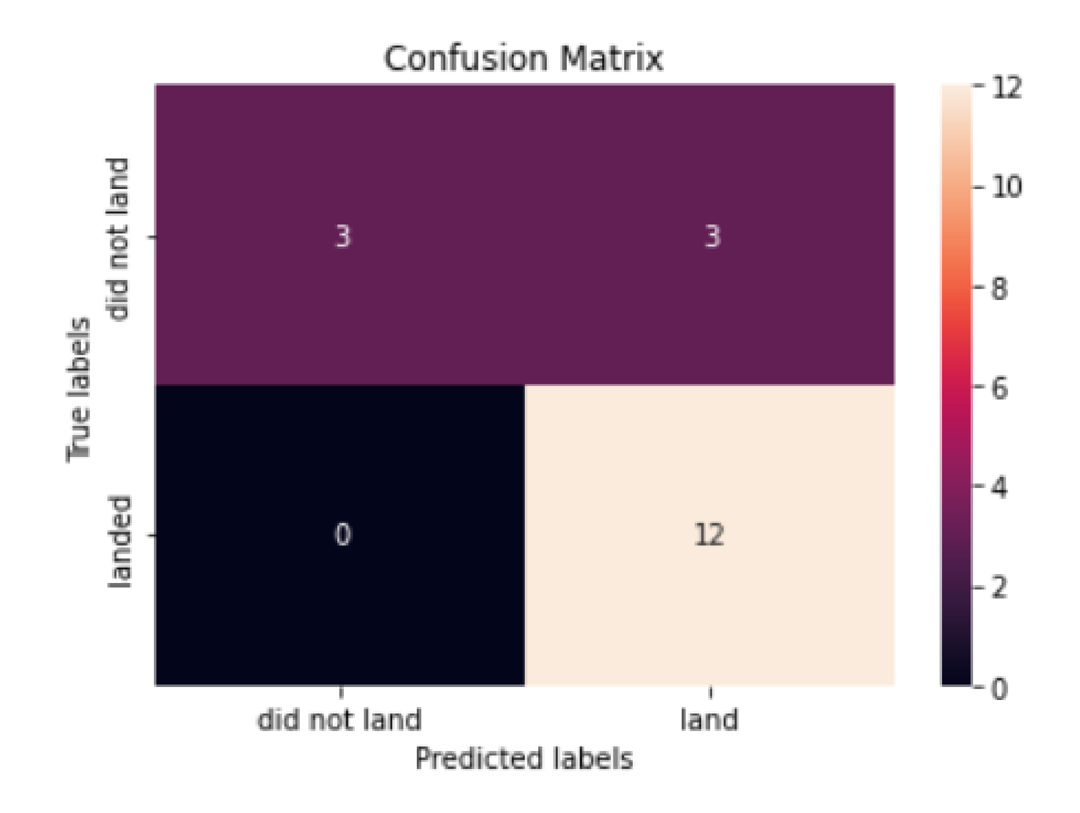
As you can observe in bar chart, all models have the same accuracy equal to 83%.

Confusion Matrix

- Since all models have the same accuracy and all confusion matrixes are the same, we illustrate just one confusion matrix as follows.

Confusion matrix shows that in our dataset there are 12 successful landing and 6 unsuccessful landing. Our model has predicted:

- 12 successful landing correctly (true positive)
- 3 unsuccessful landing incorrectly (true negative)
- 3 unsuccessful landing incorrectly as successful landing (false positive)



Conclusions

- Sucess rate kept increasing since 2013 till 2020
- Generally speaking, the more the payload is, the more the chance of successful landing is
- KSC LC-39A launch site has the best success rate
- We can now predict the outcome (successful or unsuccessful landing) with four different models with the accuracy of 83%

Appendix

- Through the Github link below, you can reach the all relevant files and notebooks of the project

https://github.com/Amorovati/IBM_Data_Science_Capstone

Thank you!