IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

<Name>
<Date>

Outline

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

Executive Summary

- Summary of methodologies
  - **Data Collection**
  - **Data Wrangling**
  - **EDA with Data Visualization**
  - **EDA with SQL**
  - **Building an interactive map with Folium**
  - **Building a dashboard with Plotly Dash**
  - **Predictive Analytics (Classification)**

- Summary of all results
  - **EDA Results**
  - **Interactive Analytics**
  - **Predictive Analytics**

Introduction

- Project background and context

SpaceX advertises Falcon 9 rocket launches on its website, with a cost of 62 million dollars. Other providers cost more than 165 million dollars each. Most of the savings is because SpaceX can reuse the first stage.

- Problems you want to find answers

The project is to predict if the first stage of the SpaceX falcon 9 rocket will land successfully.

Section 1

Methodology

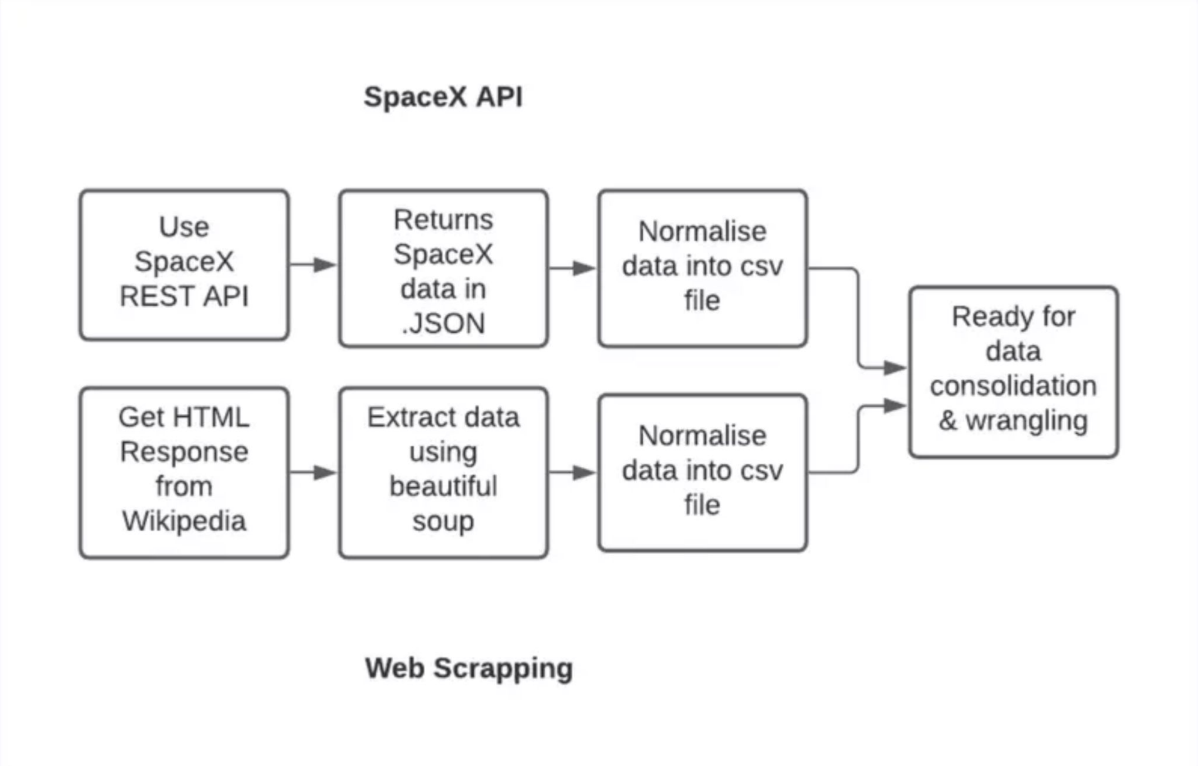
Methodology

Executive Summary

- Data collection methodology:
  - SpaceX REST API
  - Web Scraping from Wikipedia
- Perform data wrangling
  - Cleaning null values and irrelevant columns, also hot encoding data fields for machine learning
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - LR, KNN, SVM, DT models have been built and evaluated for the best classifier

Data Collection

- Collected Datasets
  - SpaceX launch data is gathered from SpaceX REST API: api.spacexdata.com/v4
  - BeautifulSoup was used for web scraping Wikipedia for Flacon 9 launch data



Data Collection – SpaceX API

- Data collection using SpaceX REST api
- [Notebook]

In [6]:
```
spacex_url="https://api.spacexdata.com/v4/launches/past"
```

In [7]:
```
response = requests.get(spacex_url)
```

```
static_json_url='https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBM-DS0321EN-SkillsNetwork/datasets/API_call_spacex_a
```

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

```
response.status_code
```

200

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

```
response = requests.get(static_json_url)
```

```
# Use json_normalize method to convert the json result into a dataframe
data = pd.json_normalize(response.json())
```

Using the dataframe `data` print the first 5 rows

```
# Get the head of the dataframe
data.head(5)
```

	static_fire_date_utc	static_fire_date_unix	tbd	net	window	rocket	success	details	crew	ships	capsules	payloa
0	2006-03-17T00:00:00.000Z	1.142554e+09	False	False	0.0	5e9d0d95eda69955f709d1eb	False	Engine failure at 33 seconds and loss of vehicle	[]	[]	[]	[5eb0e4b5b6c3bb0006eeb1e
								Successful first stage burn and				

Data Collection - Scraping

- Web scraping wikipedia using BeautifulSoup
- [Notebook]

```python
# use requests.get() method with the provided static_url
response = requests.get(static_url)
# assign the response to a object
data = response.text
```

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

```python
# Use BeautifulSoup() to create a BeautifulSoup object from a response text content
soup = BeautifulSoup(data, 'html5lib')
```

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

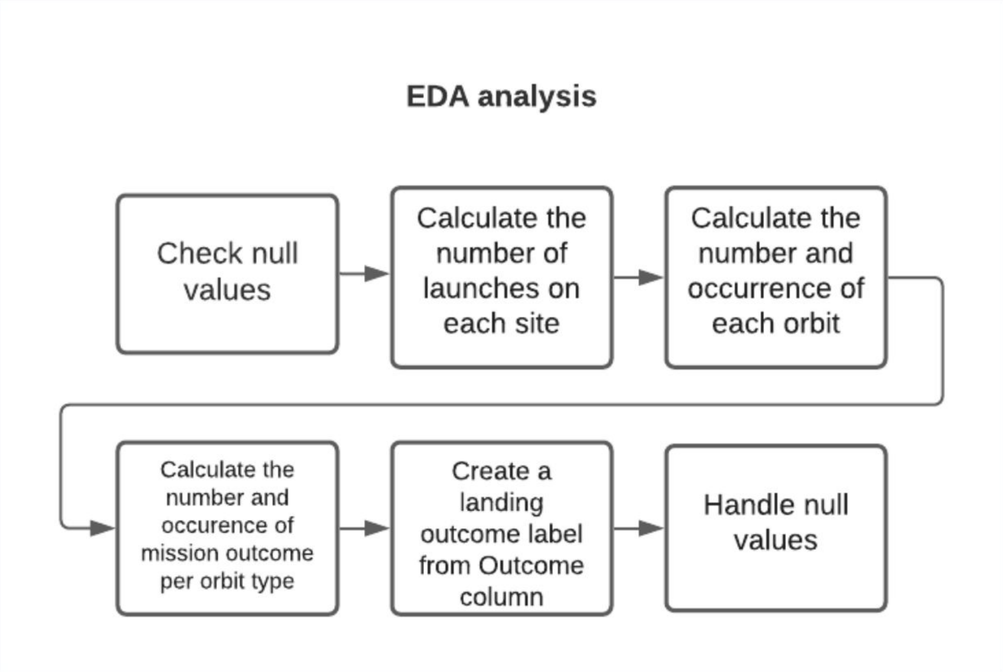
```python
# Use the find_all function in the BeautifulSoup object, with element type `table`
# Assign the result to a list called `html_tables`
html_tables = soup.find_all('table')
```

Starting from the third table is our target table contains the actual launch records.

```python
# Let's print the third table and check its content
first_launch_table = html_tables[2]
print(first_launch_table)
```

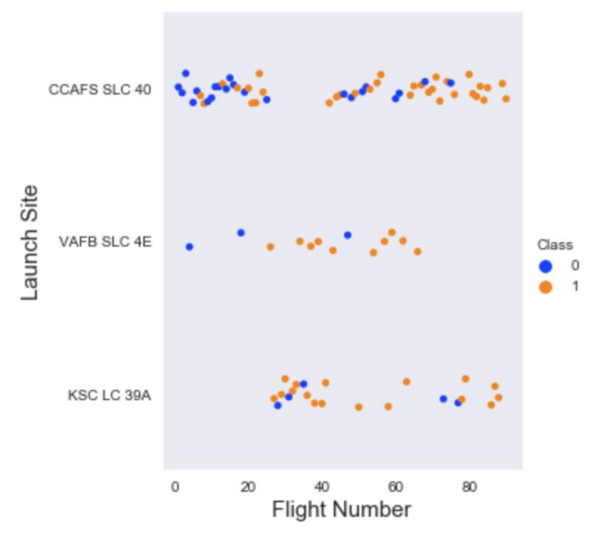
Flight No.	Date and time (UTC)	Version, Booster [b]	Launch site	Payload[c]	Payload mass	Orbit	Customer	Launch outcome	Booster landing
1	4 June 2010, 18:45	F9 v1.0[7] B0003.1[8]	CCAFS, SLC-40	Dragon Spacecraft Qualification Unit		LEO	SpaceX	Success	Failure[9][10] (parachute)
	First flight of Falcon 9 v1.0.[11] Used a boilerplate version of Dragon capsule which was not designed to separate from the second stage.(more details below) Attempted to recover the first stage by parachuting it into the ocean, but it burned up on reentry, before the parachutes even deployed.[12]								
2	8 December 2010, 15:43[13]	F9 v1.0[7] B0004.1[8]	CCAFS, SLC-40	Dragon demo flight C1 (Dragon C101)		LEO (ISS)	• NASA (COTS) • NRO	Success[9]	Failure[9][14] (parachute)

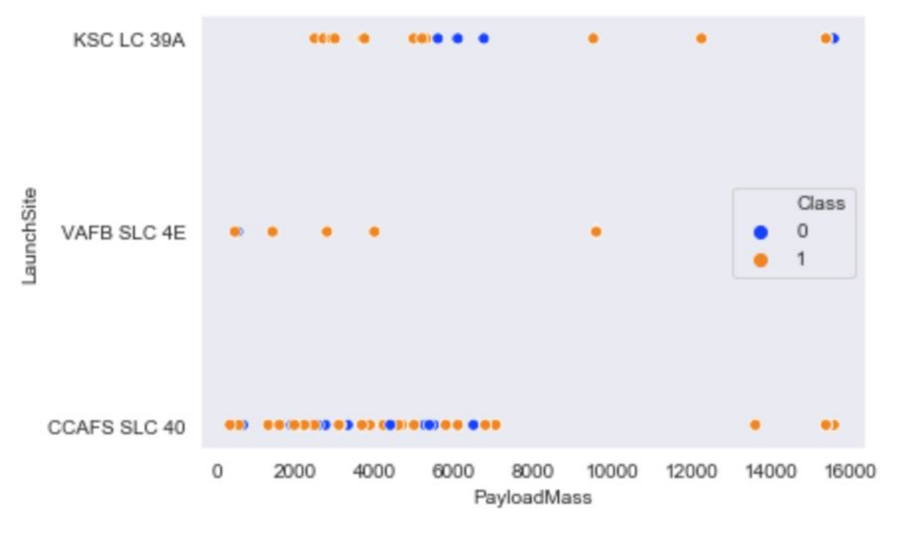
Data Wrangling

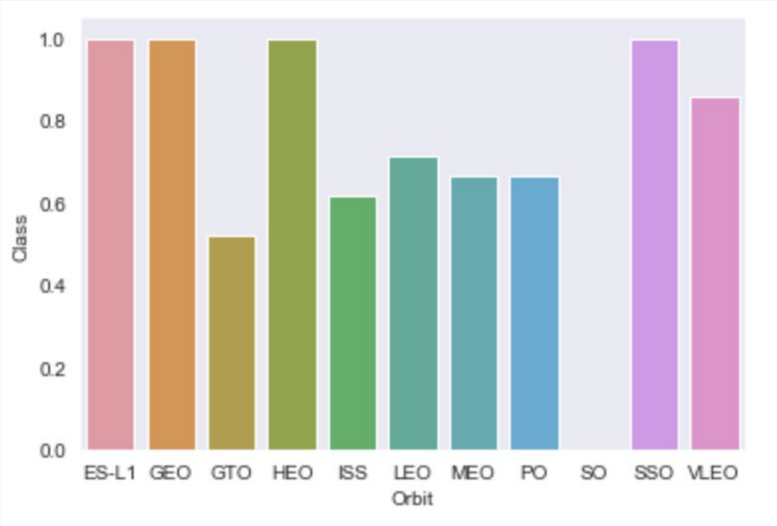


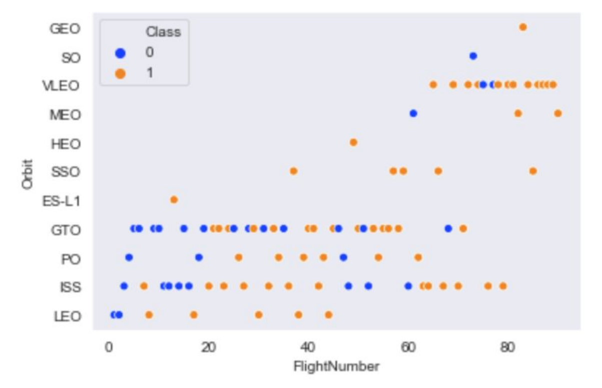
Notebook

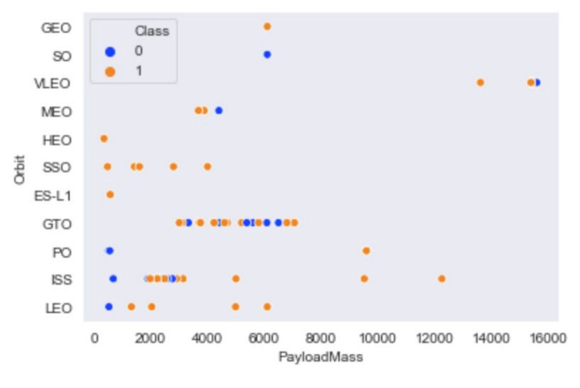
EDA with Data Visualization

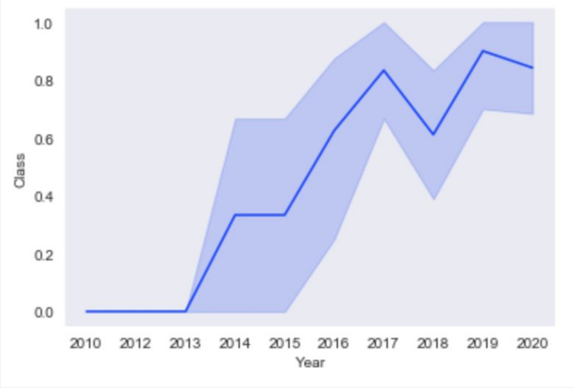












These visuals helped in determining the contribution of different factors on the success. Here is the notebook.

EDA with SQL

- Display the names of the unique launch sites in the space mission
- Display 5 records where launch sites begin with the string 'CCA'
- Display the total payload mass carried by boosters launched by NASA (CRS)
- Display average payload mass carried by booster version F9 v1.1
- List the date when the first successful landing outcome in ground pad was achieved.
- List the names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000
- List the total number of successful and failure mission outcomes
- List the names of the booster_versions which have carried the maximum payload mass. Use a subquery
- List the failed landing_outcomes in drone ship, their booster versions, and launch site names for in year 2015
- 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

Notebook

Build an Interactive Map with Folium

- Map markers have been added to the map with aim to finding an optimal location for building a launch site
- The reason the markers were added is to show where the launches were made.

[Notebook]

Build a Dashboard with Plotly Dash

- Pie charts to check success rate by sites and scatter plot for success and payload weight
- This gives us the opportunity to identify which sites and most successful and filter by site and payload weight.

[Python script]()

Predictive Analysis (Classification)

- Model development -> Model Evaluation -> Find the successful model

Notebook

Results

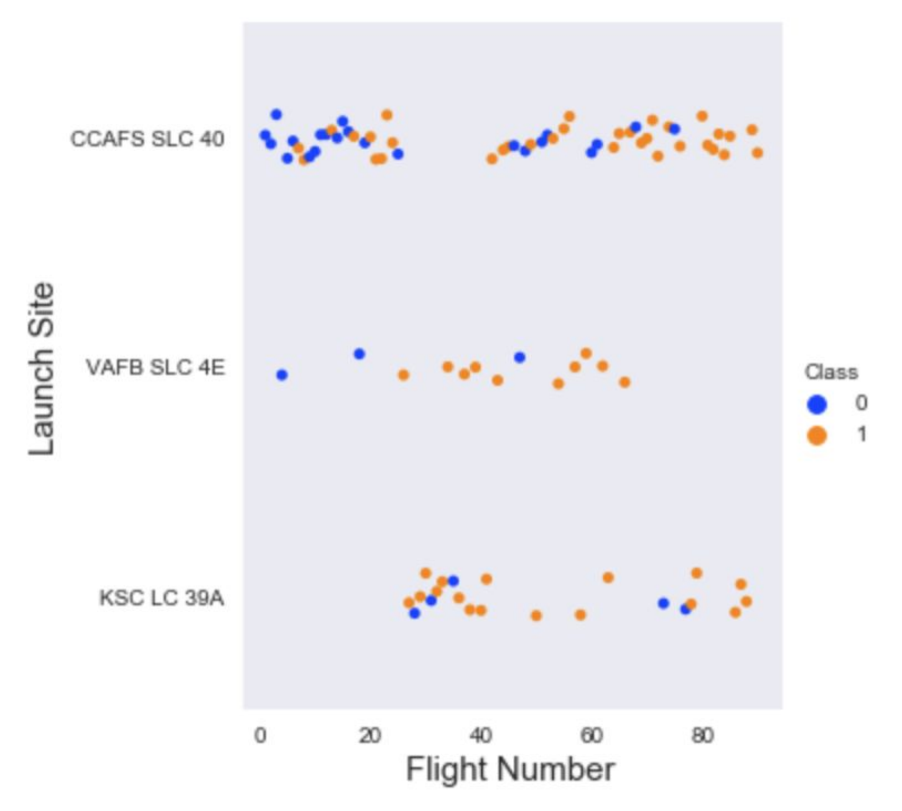
- The SVM, KNN, and Logistic Regression models are the best in terms of prediction accuracy for this dataset.
- Low weighted payloads perform better than the heavier payloads.
- The success rates for SpaceX launches is directly proportional time in years they will eventually perfect the launches.
- KSC LC 39A had the most successful launches from all the sites.
- Orbit GEO,HEO,SSO,ES L1 has the best Success Rate.

Section 2

Insights drawn from EDA

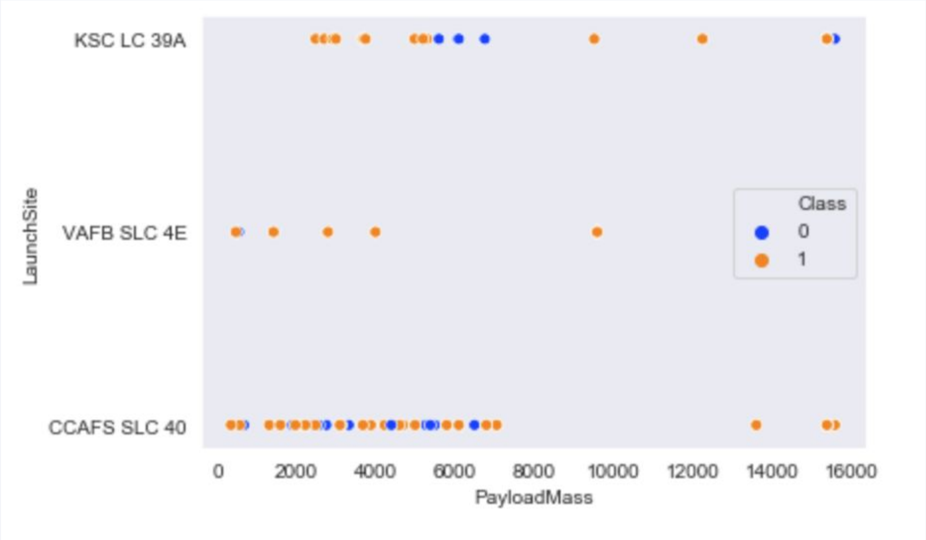
Flight Number vs. Launch Site

- Launches from the site of CCAFS SLC 40 are significantly higher than launches form other sites.



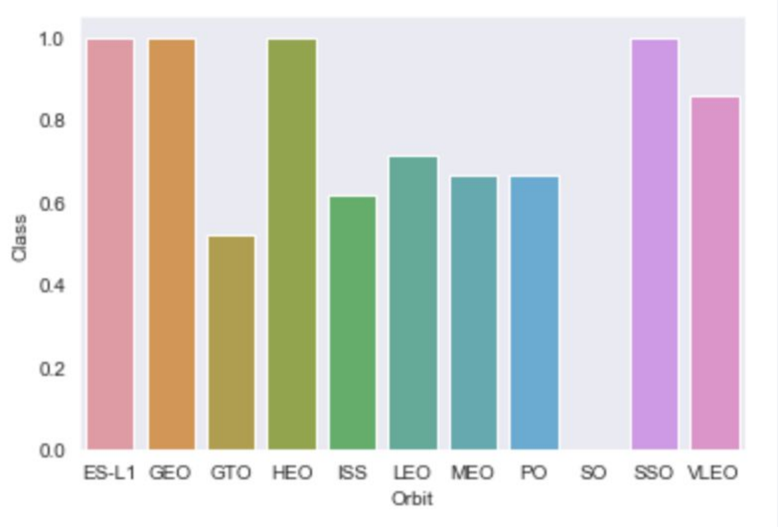
Payload vs. Launch Site

- The majority of Pay Loads with lower Mass have been launched from CCAFS SLC 40.



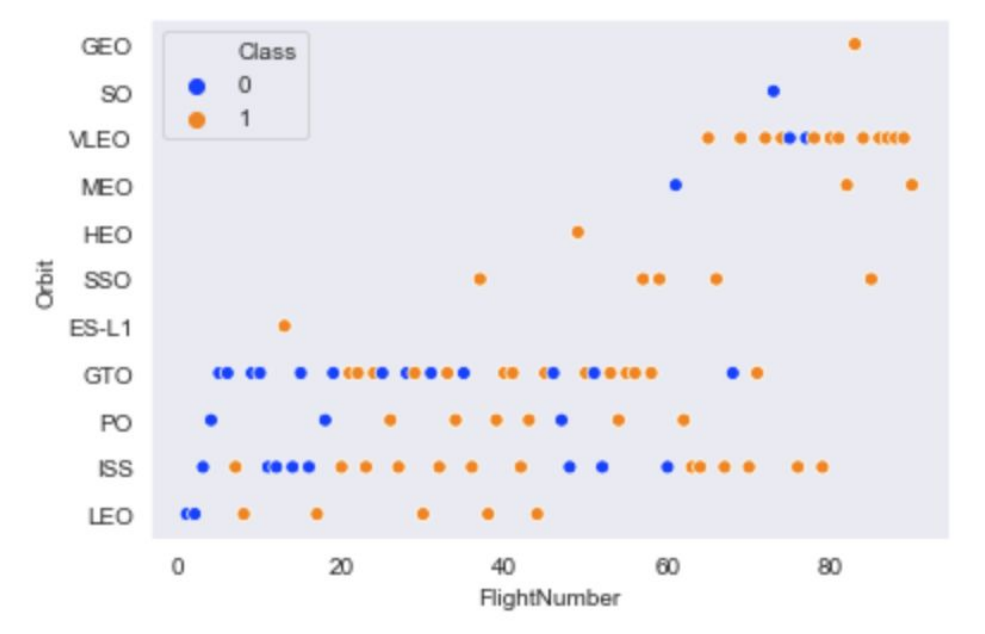
Success Rate vs. Orbit Type

- The orbit types of ES-L1, GEO, HEO, SSO are among the highest success rate.



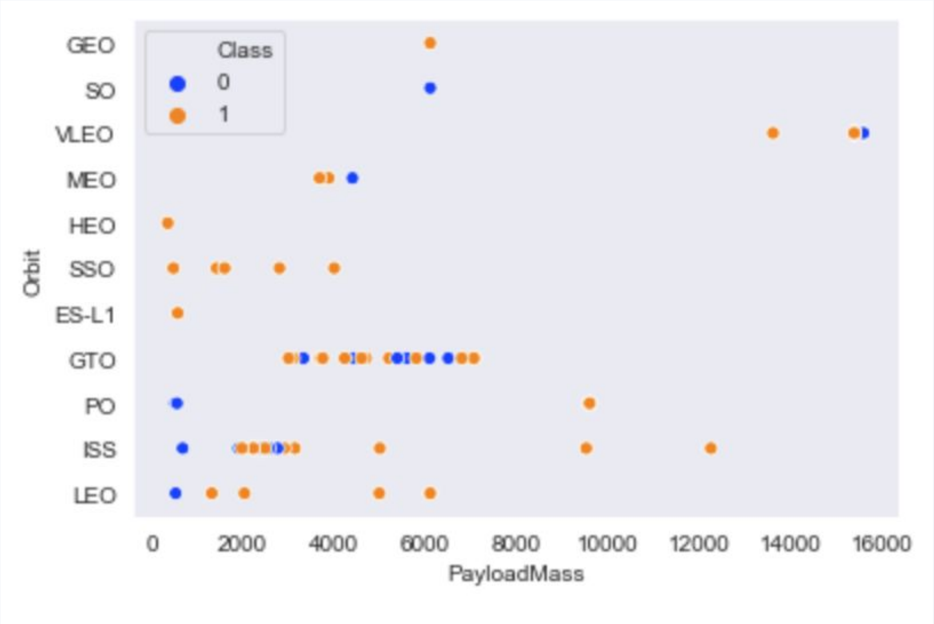
Flight Number vs. Orbit Type

- A trend can be observed of shifting to VLEO launches in recent years.



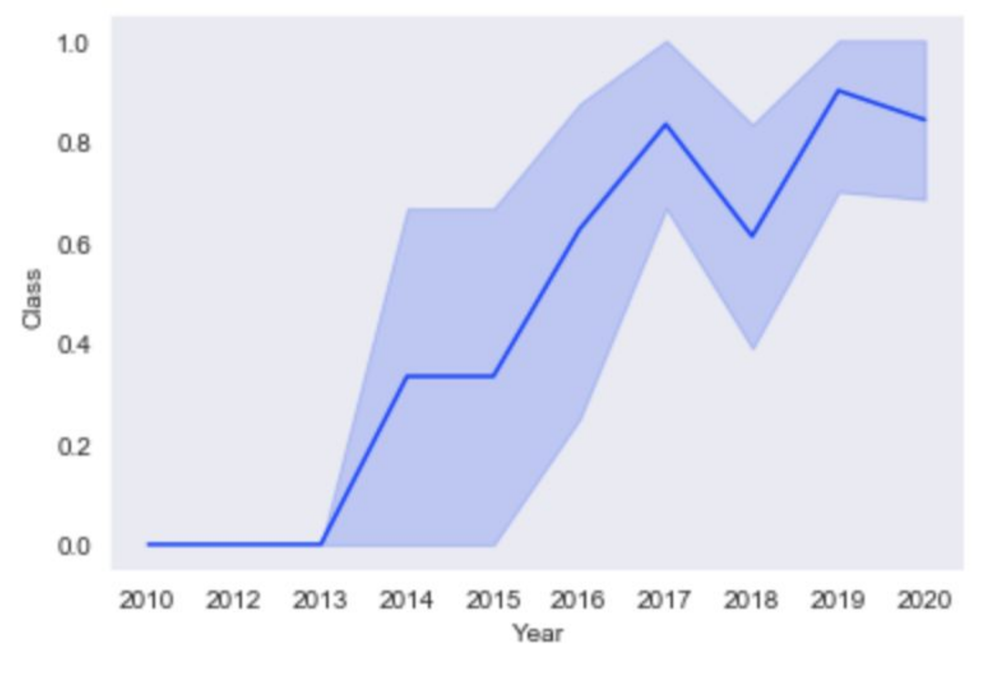
Payload vs. Orbit Type

- There are strong correlation between ISS and Payload at the range around 2000, as well as between GTO and the range of 4000-8000.



Launch Success Yearly Trend

- Launch success rate has increased significantly since 2013 and has stabilized since 2019, potentially due to advance in technology and lessons learned.



All Launch Site Names

- %sql select distinct(LAUNCH_SITE) from SPACEXTBL

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

Launch Site Names Begin with 'CCA'

- %sql SELECT * FROM SPACEXTBL 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

Total Payload Mass

- %sql SELECT SUM(PAYLOAD_MASS__KG_) AS TOTAL_PAYLOAD_MASS FROM SPACEXTBL \

  WHERE CUSTOMER = 'NASA (CRS)';

total_payload_mass
45596

Average Payload Mass by F9 v1.1

- %sql SELECT AVG(PAYLOAD_MASS__KG_) AS AVERAGE_PAYLOAD_MASS FROM SPACEXTBL \

  WHERE BOOSTER_VERSION = 'F9 v1.1';

average_payload_mass
2928

First Successful Ground Landing Date

- %sql SELECT MIN(DATE) AS FIRST_SUCCESSFUL_GROUND_LANDING FROM SPACEXTBL \

  WHERE LANDING__OUTCOME = 'Success (ground pad)';

first_successful_ground_landing
2015-12-22

Successful Drone Ship Landing with Payload between 4000 and 6000

- %sql SELECT BOOSTER_VERSION FROM SPACEXTBL \
  WHERE (LANDING__OUTCOME = 'Success (drone ship)') AND
  (PAYLOAD_MASS__KG_ BETWEEN 4000 AND 6000);

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 SELECT MISSION_OUTCOME, COUNT(MISSION_OUTCOME) AS TOTAL_NUMBER FROM SPACEXTBL GROUP BY MISSION_OUTCOME;

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

Boosters Carried Maximum Payload

- %sql SELECT MISSION_OUTCOME, COUNT(MISSION_OUTCOME) AS TOTAL_NUMBER FROM SPACEXTBL GROUP BY MISSION_OUTCOME;

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

2015 Failed Launch Records

- %sql SELECT BOOSTER_VERSION, LAUNCH_SITE FROM SPACEXTBL\
  WHERE (LANDING__OUTCOME = 'Failure (drone ship)') AND (EXTRACT(YEAR FROM DATE) = '2015');

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

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

- %sql SELECT LANDING__OUTCOME, COUNT(LANDING__OUTCOME) AS TOTAL_NUMBER FROM SPACEXTBL \

  WHERE DATE BETWEEN '2010-06-04' AND '2017-03-20' \

  GROUP BY LANDING__OUTCOME \

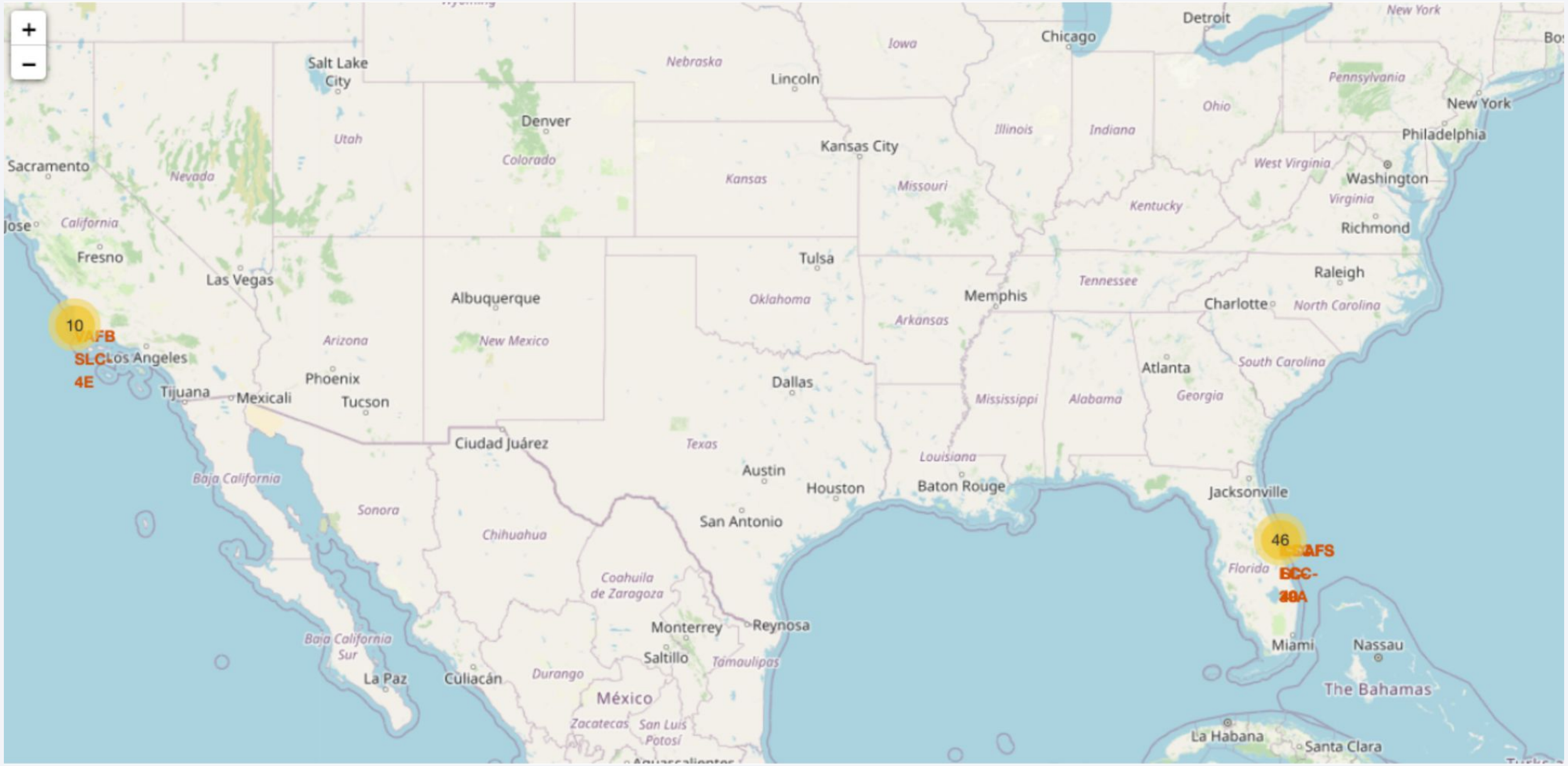
  ORDER BY TOTAL_NUMBER DESC;

landing__outcome	total_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

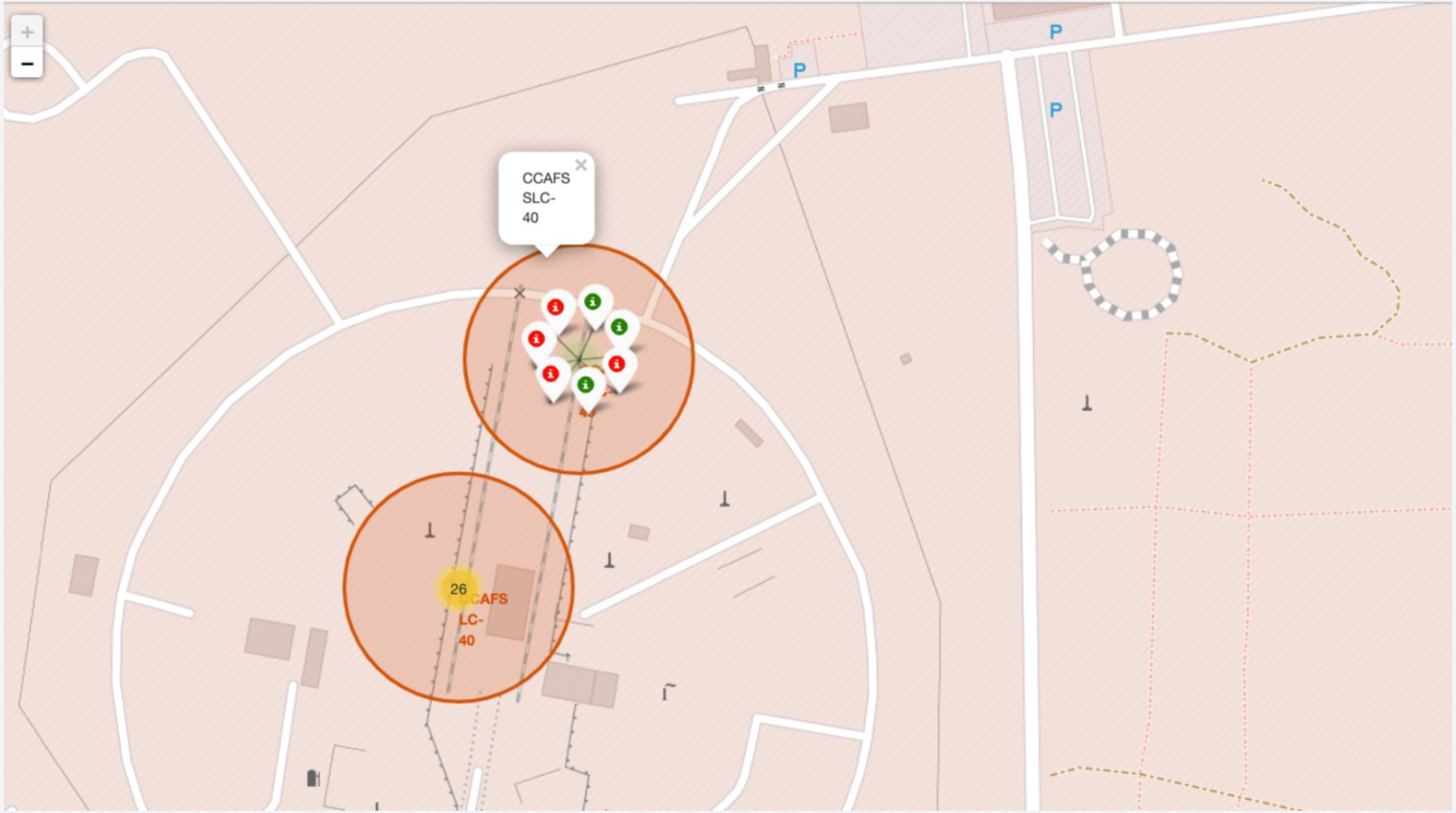
Section 3

Launch Sites Proximities Analysis

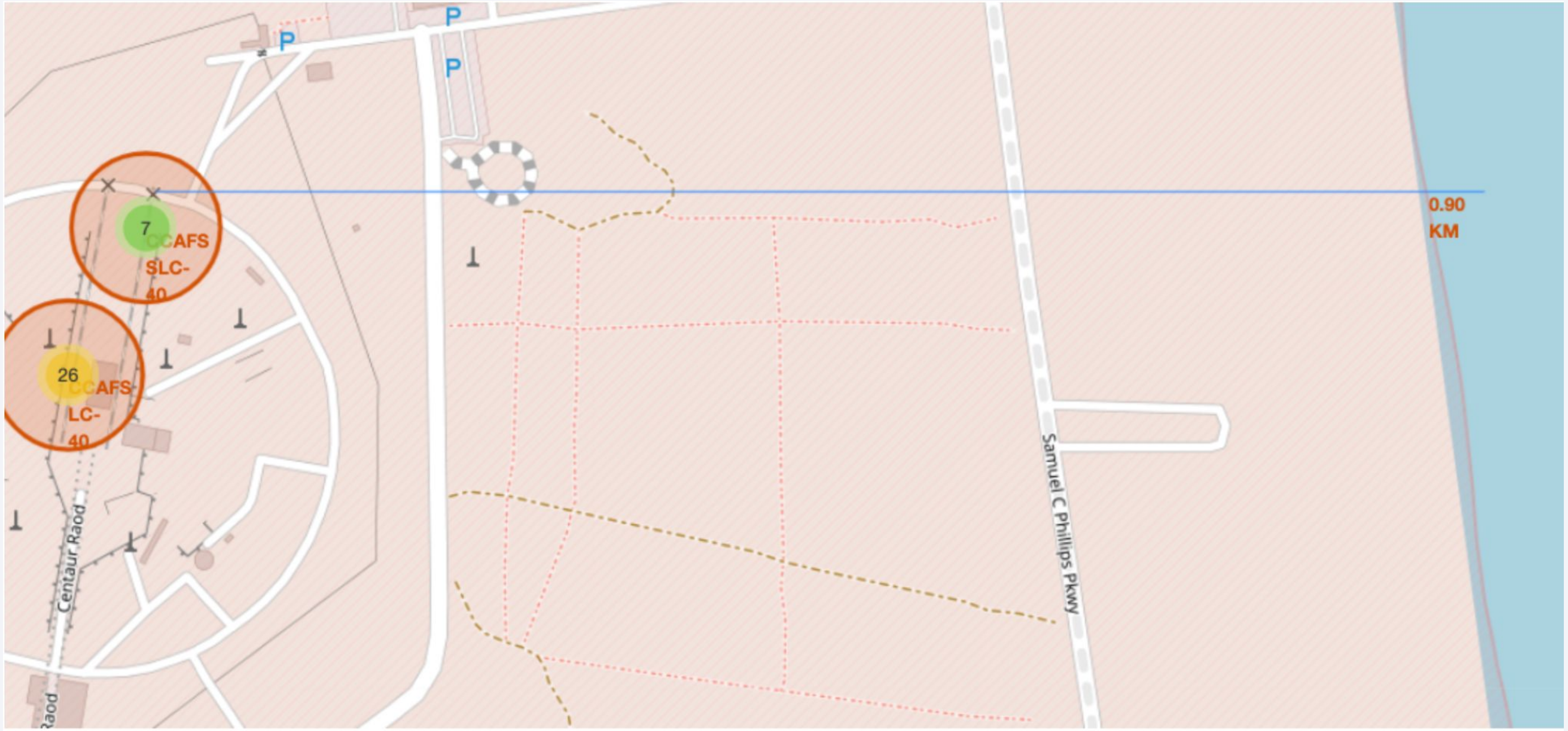
All launch sites marked on a map



Success/Failed launches marked on the map



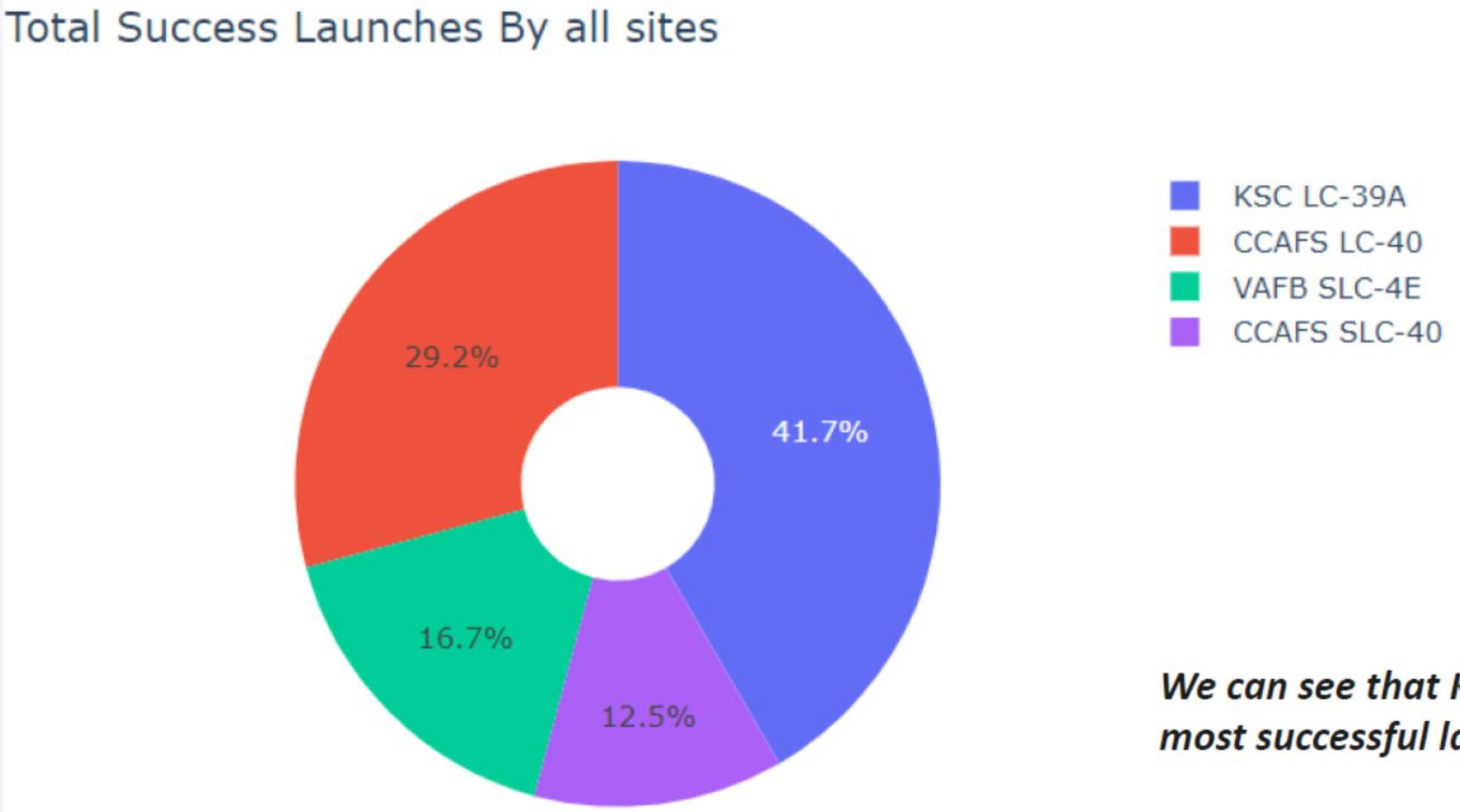
Distances between a launch site to its proximities



Section 4

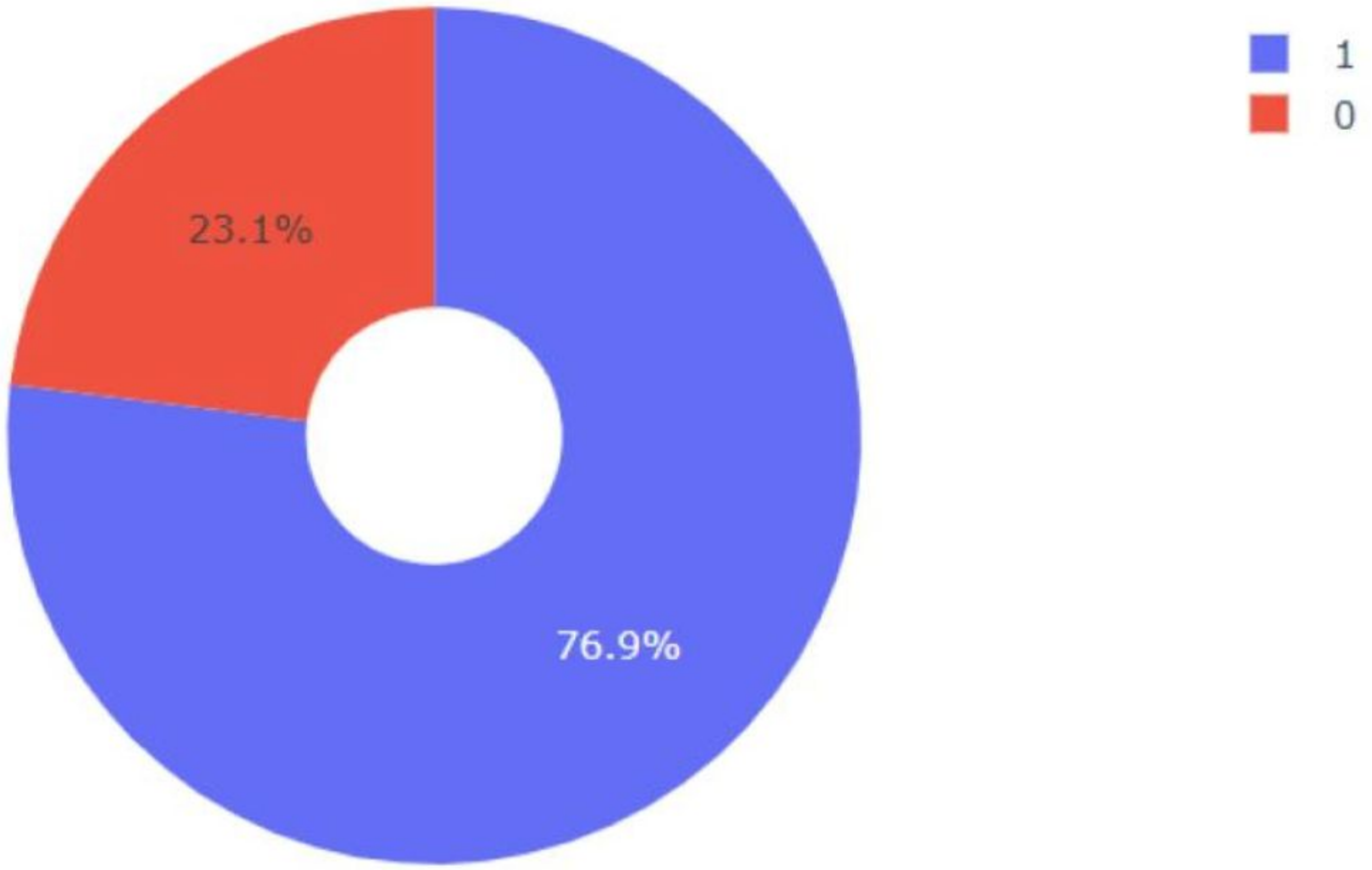
Build a Dashboard with Plotly Dash

Total success launches by all sites



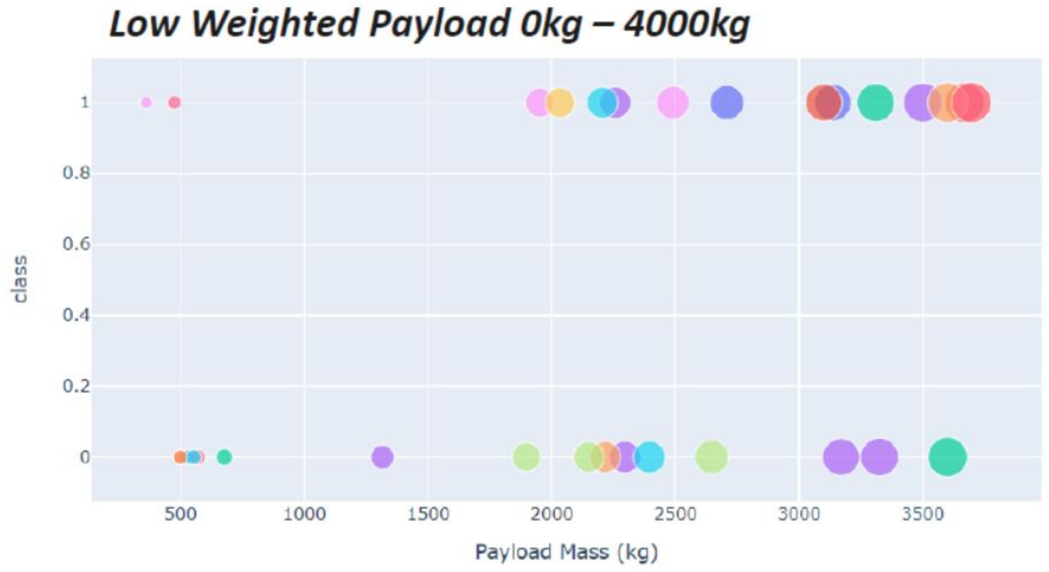
We can see that KSC LC-39A had the most successful launches from all the sites

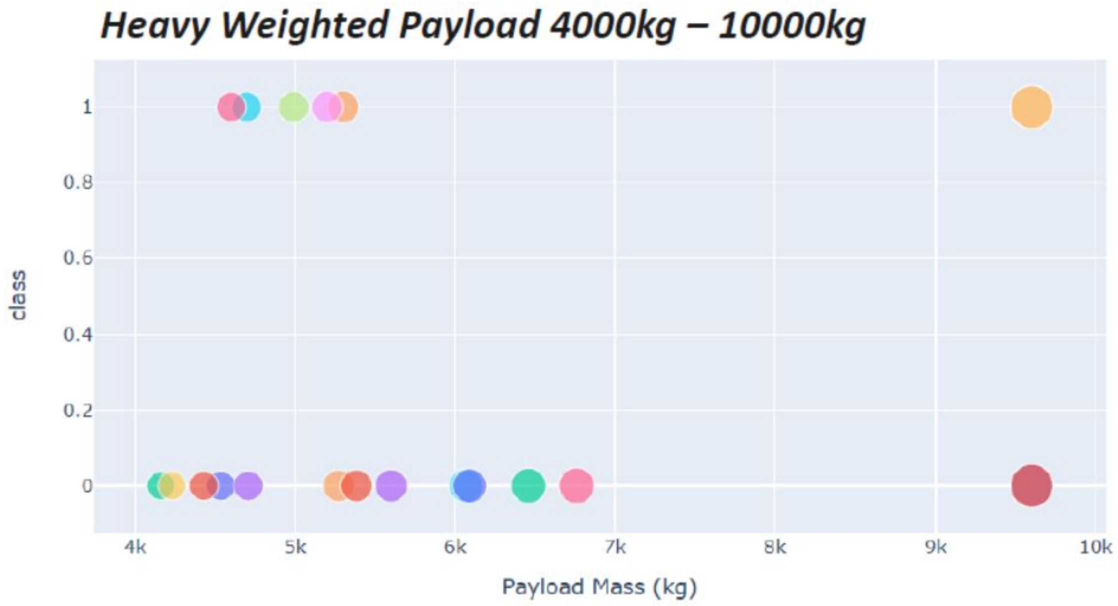
Success rate for KSC LC-39A



KSC LC-39A achieved a 76.9% success rate while getting a 23.1% failure rate

Payload vs Launch Outcome



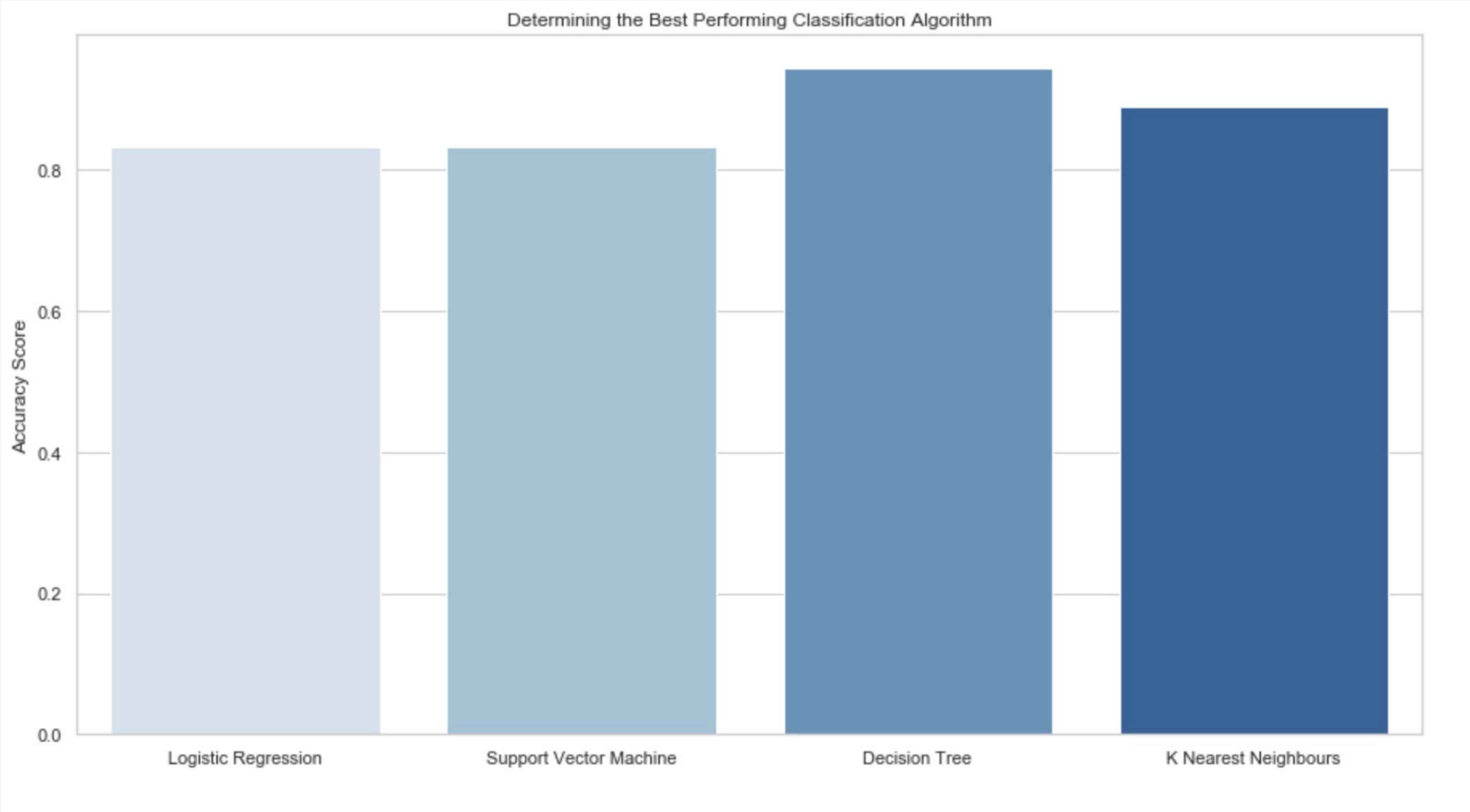


We can see the success rates for low weighted payloads is higher than the heavy weighted payloads

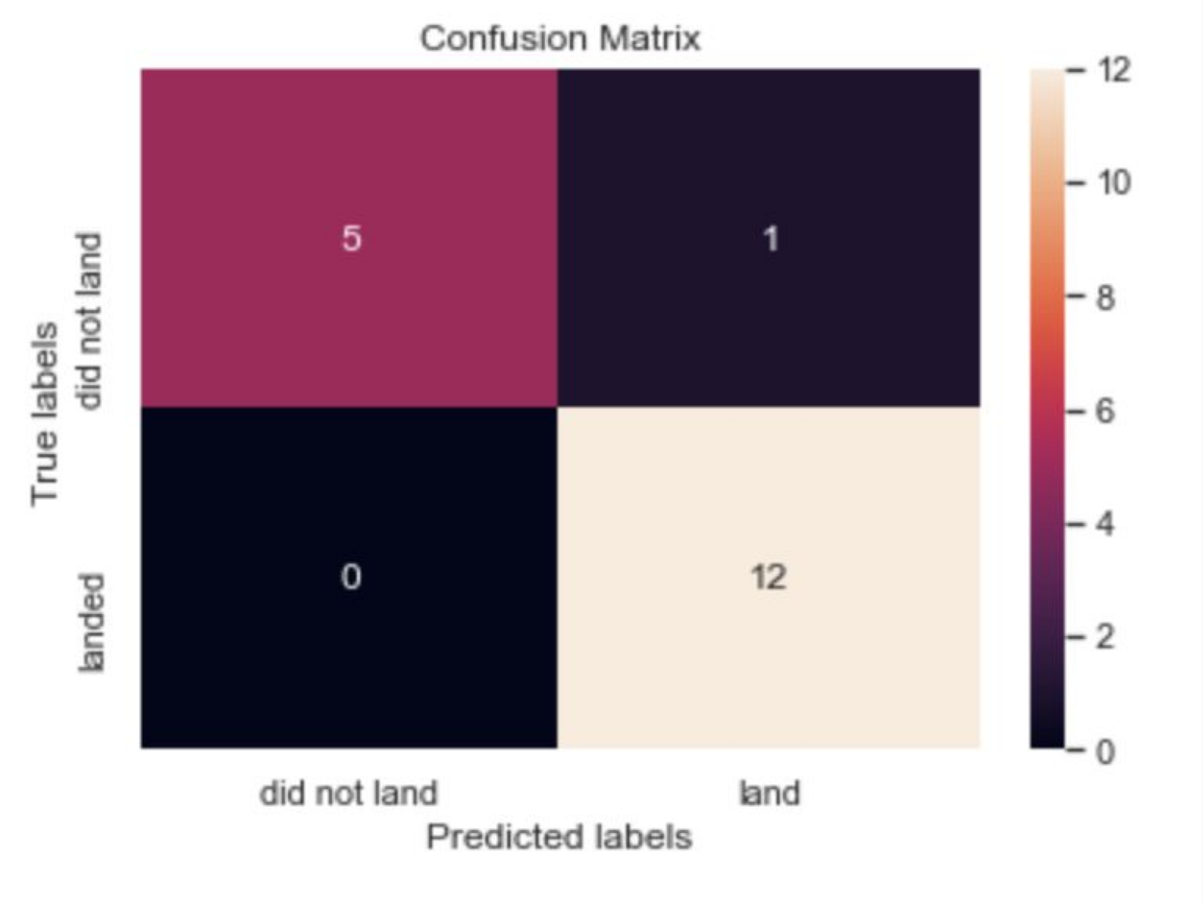
Section 5

Predictive Analysis (Classification)

Classification Accuracy



Confusion Matrix



- Decision tree confusion matrix using the accuracy method score of 94.44

Conclusions

- The decision tree model is the best in terms of prediction accuracy for this dataset.
- Low weighted payloads perform better than the heavier payloads.
- The success rates for SpaceX launches is directly proportional time in years they will eventually perfect the launches.
- KSC LC 39A had the most successful launches from all the sites.
- Orbit GEO,HEO,SSO,ES L1 has the best Success Rate.

Appendix

- Github repo with all the notebooks and csv files

Thank you!