SPACEX

WINNING SPACE RACE WITH DATA SCIENCE

Cihan Acilov

13/02/2022

Outline

What is covered in the presentation?

Executive Summary

METHODOLOGIES

Data collection
- WebScraping (Wiki)
- Through SpaceX API

Data wrangling

Exploratory Data Analysis
- Data **visualization**/Seaborn
- **EDA with SQL**

Interactieve Visuals
- **Map with Folium**
- **Dashboard with Plotly**

Predictive analysis (Classification methods)

RESULTS

Exploratory data analysis results

Interactive analytics demo in screenshots

Predictive analysis results

Introduction

Project background and context

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. This information can be used if an alternate company wants to bid against SpaceX for a rocket launch.

Problems we want to find answers

Can we determine/predict whether the first stage of the rocket will land successfully?

Which variables are playing a significant role in determining whether a rocket will land successfully, and what are their influences on the outcome?

Can we manipulate the significant variables to influence the outcome of rocket launches?

Section 1

METHODOLOGY

Methodology

Executive Summary

Data collection methodology:

Through SpaceX API

Web Scraping

Perform data wrangling

Data preprocessing

Dealing with missing values

EDA using visualization and SQL

Interactive visual with Folium and Plotly Dash

Predictive analysis: Classification models

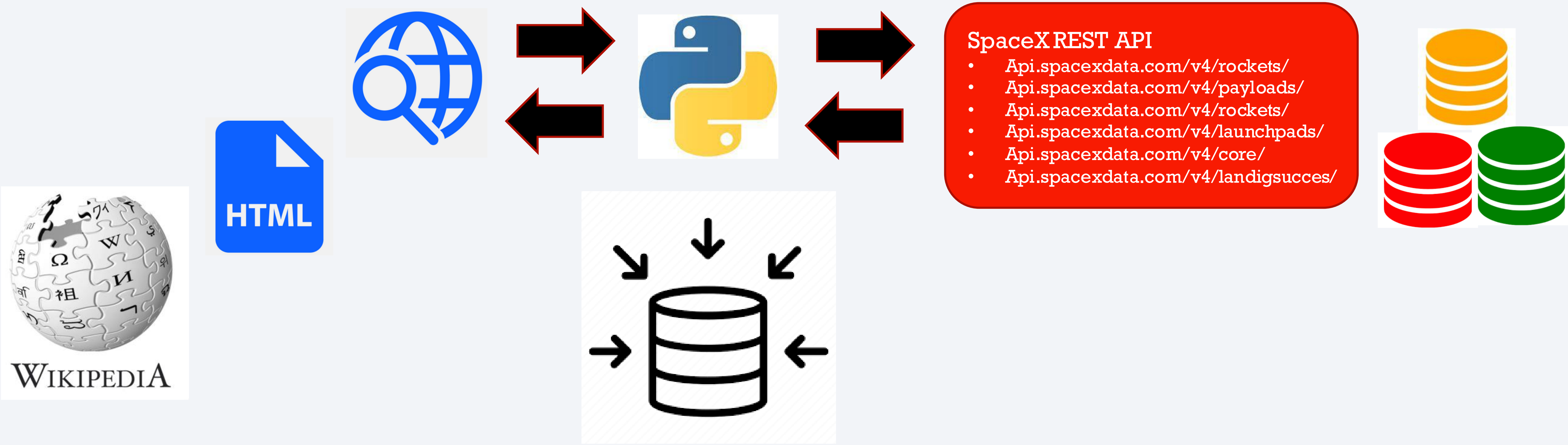
Build, tune, evaluate classification models

Data Collection

Data set is collected through:

1. SpaceX REST API: https://api.spacexdata.com/v4/

2.Wikipedia: https://en.wikipedia.org/w/index.php?title=List_of_Falcon_9_and_Falcon_Heavy_launches&oldid=1027686922



Data Collection – SpaceX API

GitHub URL of the completed SpaceX API calls notebook: GitHub

```
spacex_url="https://api.spacexdata.com/v4/launches/past"
```

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

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

```
data = pd.json_normalize(response)
```

```
# Get the head of the dataframe
data.head(1)
```

rocket	payloads	launchpad	cores	flight_number	date_utc	date
e9d0d95eda69955f709d1eb	5eb0e4b5b6c3bb0006eeb1e1	5e9e4502f5090995de566f86	{'core': '5e9e289df35918033d3b2623', 'flight': 1, 'gridfins': False, 'legs': False, 'reused': False, 'landing_attempt': False, 'landing_success': None, 'landing_type': None, 'landpad': None}	1	2006-03-24T22:30:00.000Z	2006-03-24

```
launch_dict = {'FlightNumber': list(data['flight_number']),
'Date': list(data['date']),
'BoosterVersion':BoosterVersion,
'PayloadMass':PayloadMass,
'Orbit':Orbit,
'LaunchSite':LaunchSite,
'Outcome':Outcome,
'Flights':Flights,
'GridFins':GridFins,
'Reused':Reused,
'Legs':Legs,
'LandingPad':LandingPad,
'Block':Block,
'ReusedCount':ReusedCount,
'Serial':Serial,
'Longitude': Longitude,
'Latitude': Latitude}
```

94 rows & 17 columns of data collected

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 94 entries, 0 to 93
Data columns (total 17 columns):
 #   Column          Non-Null Count  Dtype
---  ------          --------------  -----
 0   FlightNumber    94 non-null     int64
 1   Date            94 non-null     object
 2   BoosterVersion  94 non-null     object
 3   PayloadMass     88 non-null     float64
 4   Orbit           94 non-null     object
 5   LaunchSite      94 non-null     object
 6   Outcome         94 non-null     object
 7   Flights         94 non-null     int64
 8   GridFins        94 non-null     bool
 9   Reused          94 non-null     bool
 10  Legs            94 non-null     bool
 11  LandingPad      64 non-null     object
 12  Block           90 non-null     float64
 13  ReusedCount     94 non-null     int64
 14  Serial          94 non-null     object
 15  Longitude       94 non-null     float64
 16  Latitude        94 non-null     float64
dtypes: bool(3), float64(4), int64(3), object(7)
memory usage: 10.7+ KB
```

Data Collection - Scraping

GitHub URL of the completed SpaceX API calls notebook: GitHub

HTTP GET method to request the Falcon9 Launch HTML page, as an HTTP response

BeautifulSoup object from the HTML response

Locate the table with the launch **records** of SpaceX

Extract data from the tables and create a data from parsing the HTML

	Flight No.	Launch site	Payload	Payload mass	Orbit	Customer	Launch outcome	Version Booster	Booster landing	Date	Time
0	1	CCAFS	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success\n	F9 v1.0B0003.1	Failure	4 June 2010	18:45
1	2	CCAFS	Dragon	0	LEO	NASA	Success	F9 v1.0B0004.1	Failure	8 December 2010	15:43
2	3	CCAFS	Dragon	525 kg	LEO	NASA	Success	F9 v1.0B0005.1	No attempt\n	22 May 2012	07:44
3	4	CCAFS	SpaceX CRS-1	4,700 kg	LEO	NASA	Success\n	F9 v1.0B0006.1	No attempt	8 October 2012	00:35
4	5	CCAFS	SpaceX CRS-2	4,877 kg	LEO	NASA	Success\n	F9 v1.0B0007.1	No attempt\n	1 March 2013	15:10

Data Wrangling

- Missing values by Orbit is replaced by MEAN
- All variable's excluding label are normalized
- Unnacsarry columns are removed
- One Hot Encoding

```
1  data_falcon9.isnull().sum()
```

```
FlightNumber       0
Date               0
BoosterVersion     0
PayloadMass        5
Orbit              0
LaunchSite         0
Outcome            0
Flights            0
GridFins           0
Reused             0
Legs               0
LandingPad        26
Block              0
ReusedCount        0
Serial             0
Longitude          0
Latitude           0
dtype: int64
```

EDA with Data Visualization

Scatter Plot is used to see the relationship between different variables:

- Flight Number and Launch Site
- Flight Number and Payload Mass
- Launch Site and Payload Mass
- Orbit and Flight Number
- Orbit and Payload Mass

▪ Add the GitHub URL

Bar Plot is used to see the relationship between success rate of each orbit:

- Orbit vs Class

Line Plot is used to see the yearly trend of successful launches:

- Year vs Class

EDA with SQL

Following SQL queries are implemented:

- Names of the unique launch sites
- Total payload mass carried by boosters launched by NASA (CRS)
- Average payload mass carried by booster version
- Listing successful landing outcomes
- Names of the boosters which have success in drone ship
- Total number of successful and failure mission outcomes
- Boosters with maximum payload mass
- Failed landing outcomes, and launching sides

Interactive Map with Python *Folium*

The launch success rate may depend on the location and the proximity of launch site. To locate an OPTIMAL LOCATION, the available data is visualized in interactive maps by using Python Folium library:

- Launch sites are marked on the map based on LAT. & LONG.
- Launch sites are separated by Circles and Marks (names of the sites).
- **Succeeded** and **failed** launches are marked per Launch Site.
- Distance is calculated to railway, highway, coastline etc. To see whether successful launches have commonalities regarding location.

Dashboard with Python Plotly Dash

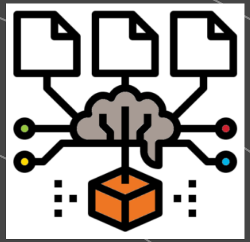
An Interactive Plotly Dash application is built for users to perform visual analytics on SpaceX launch data in real-time

- A drop-down menu is created for the selection of a specific Launch Site, when a Launch site is selected the visuals in the dashboard filtered accordingly.
- Number of launches by Launch Site are visualized with a Pei Chart, which clearly shows the performance difference of different Launch Site.
- An interactive scatter plot is built including a slider which helps user to select a range for Payload Mass. The visual clearly shows the difference in Booster Version and Payload Mass and whether they succeeded or failed.

Predictive Analysis (Classification)

Model Building

- **Class** (target variable) is extracted from Panda's Data Frame into a NumPy array.
- Relevant features are selected and **normalized** to create a common scale between predicting variables/features
- The dataset is split into **training** and **testing** set, so that the predictions can be validated through the testing test.
- Different **classification** models are used and the best performing **hyperparameters** are selected by making use of Sklearn GridSearchCV



Model Evaluation

- Accuracy measurement
- Tweaking hyper parameters.
- Confusion Matrix
- Feature engineering

Model Selection

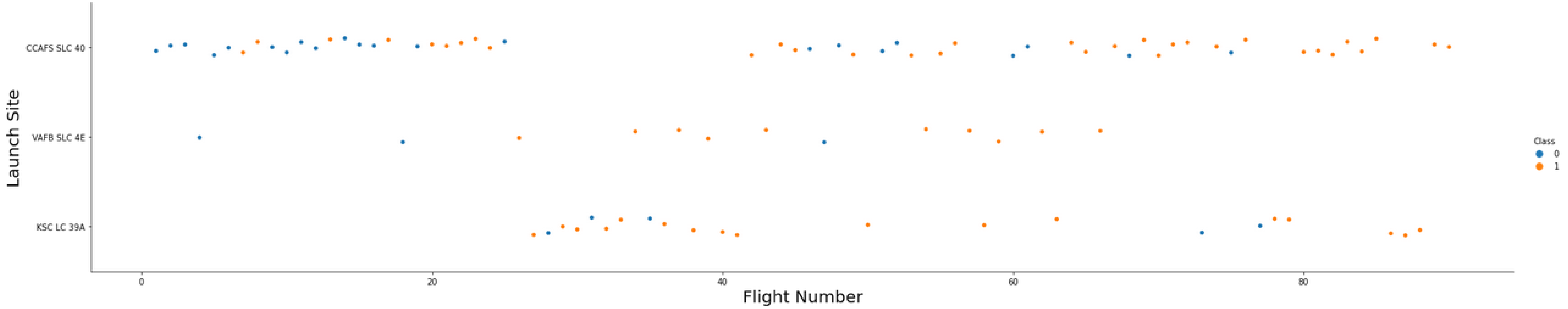
Multiple classification models are trained. **Best performing model** will be selected based on the **performance** of the model on an unknown data.

Results

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

Section 2

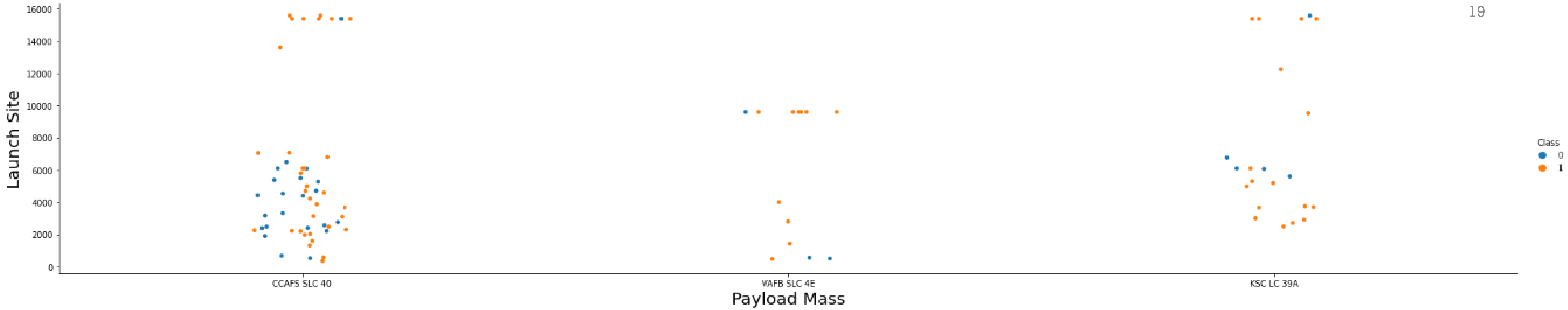
Insight drawn from EDA



As the Flight Number increases there are more successful landings by all Launch Sites.

The launch site VAFB-SLC has no Flights after approximately Fight Number 60.

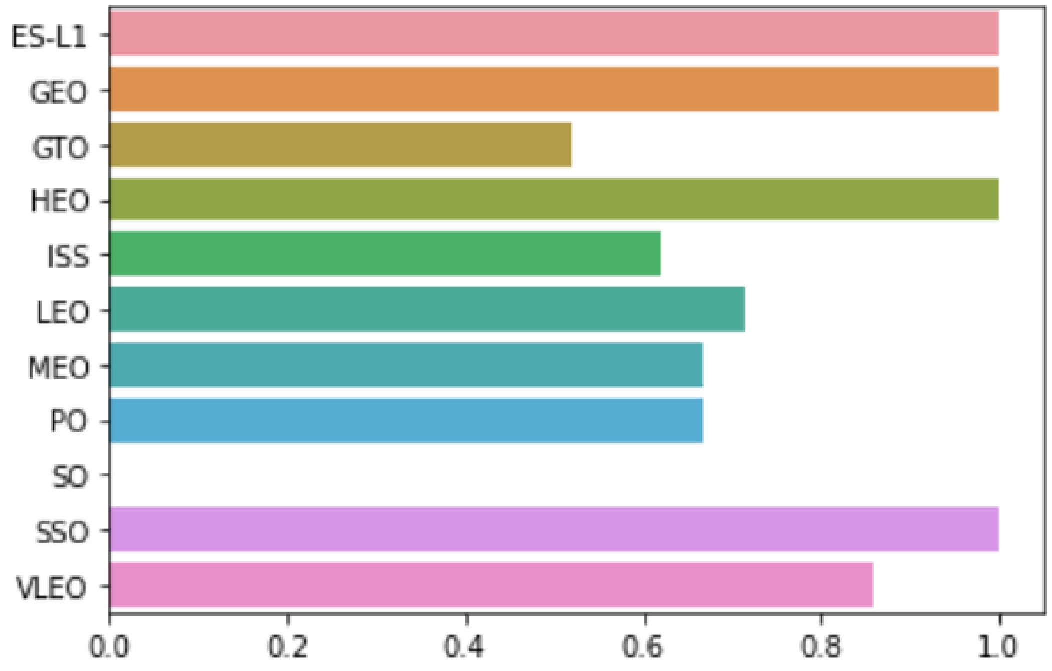
Flight Number vs. Launch Site



The launch site VAFB-SLC has no rockets launched for heavy payload mass(greater than 10000)

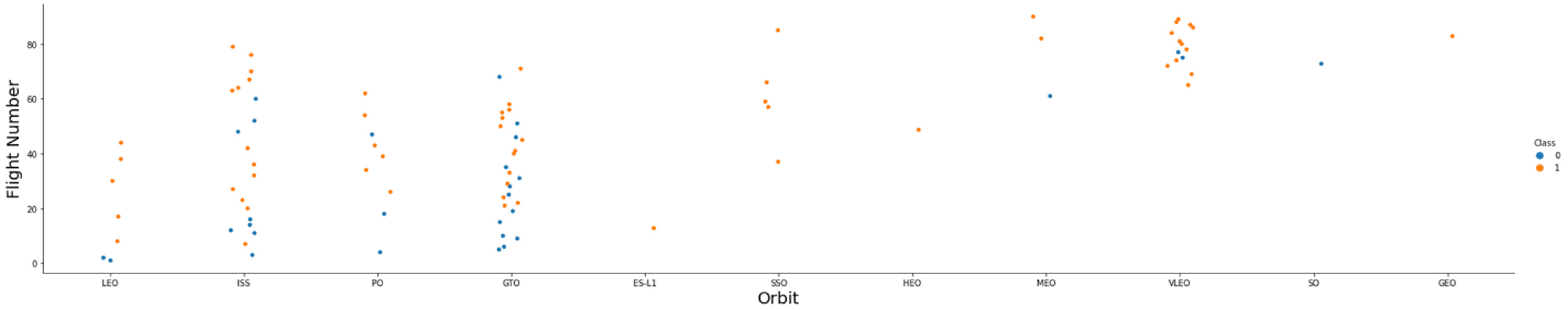
Launch Site vs. Payload Mass

Success Rate vs. Orbit Type



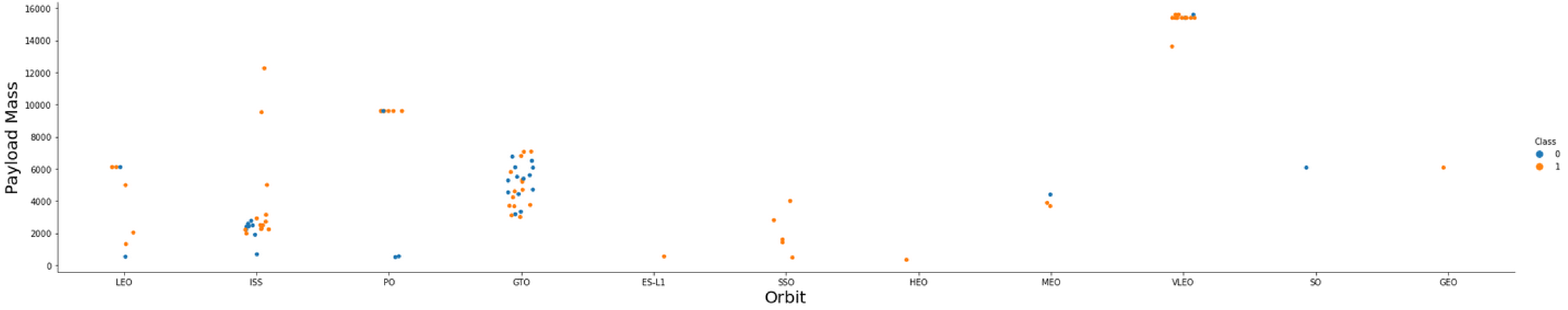
The Bar Chart is showing different orbits and the success rate between 0 (failure) and 1 (successful) for rocket launches.

Successful landings are frequent during the launches to ES-L1,GEO,HEO and SSO.



LEO orbit the Success appears related to the number of flights; on the other hand, there seems to be no relationship between flight number when in GTO orbit

Flight Number vs. Orbit Type

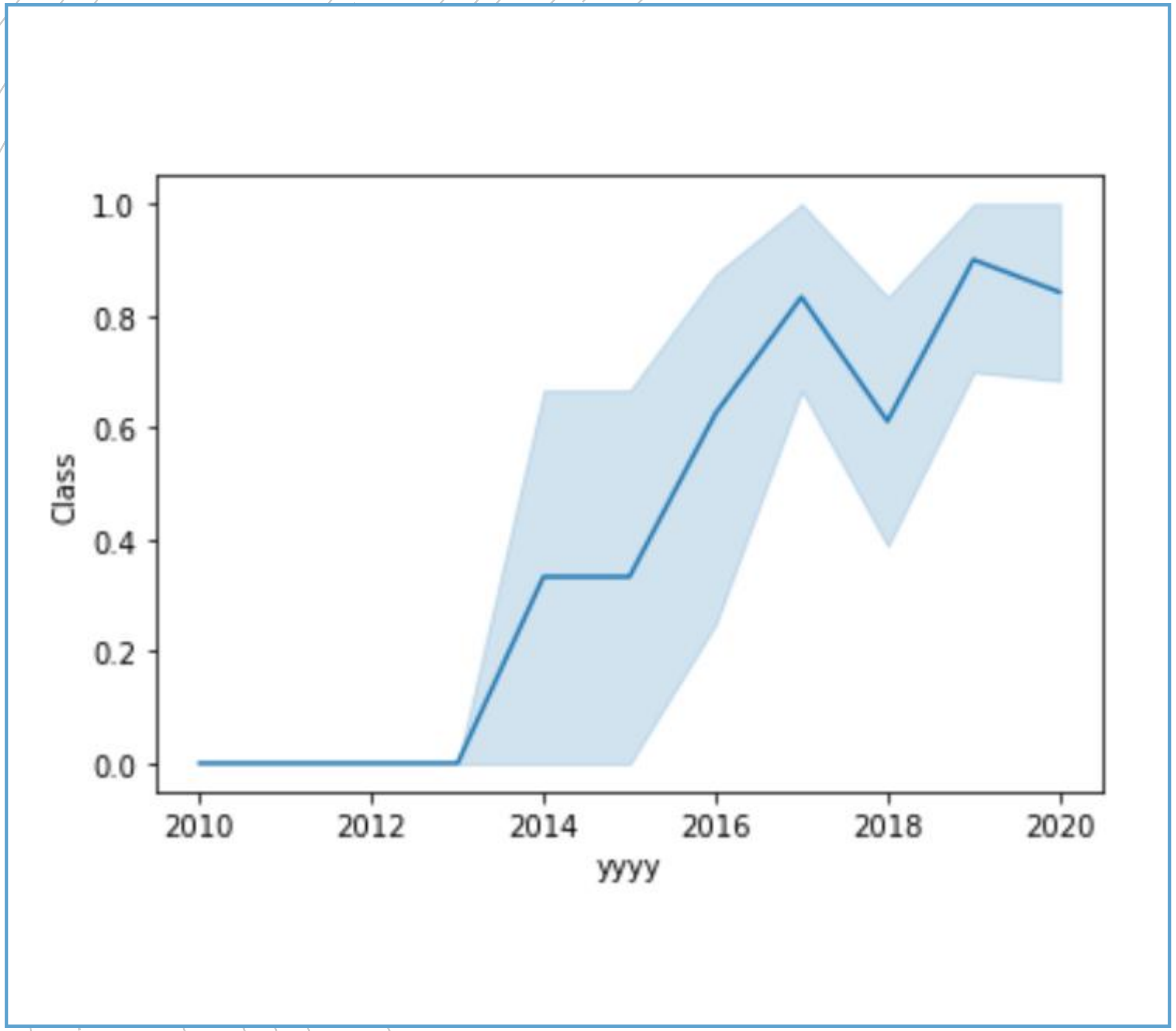


With heavy payloads the successful landing or positive landing rate are more for Polar, LEO and ISS.

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

Payload Mass vs. Orbit Type

Launch Success Yearly Trend



Since 2013 the success rate is increasing.

EDA SQL

All Launch Site Names

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

The dataset contains entry for four DISTINCT launch sites.

EDA SQL

Launch Site Names Begin with 'CCA'

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

The dataset is filtered using SQL commands to retrieve first 5 entries where the launch site name begins with "CCA"

EDA SQL Total Payload Mass

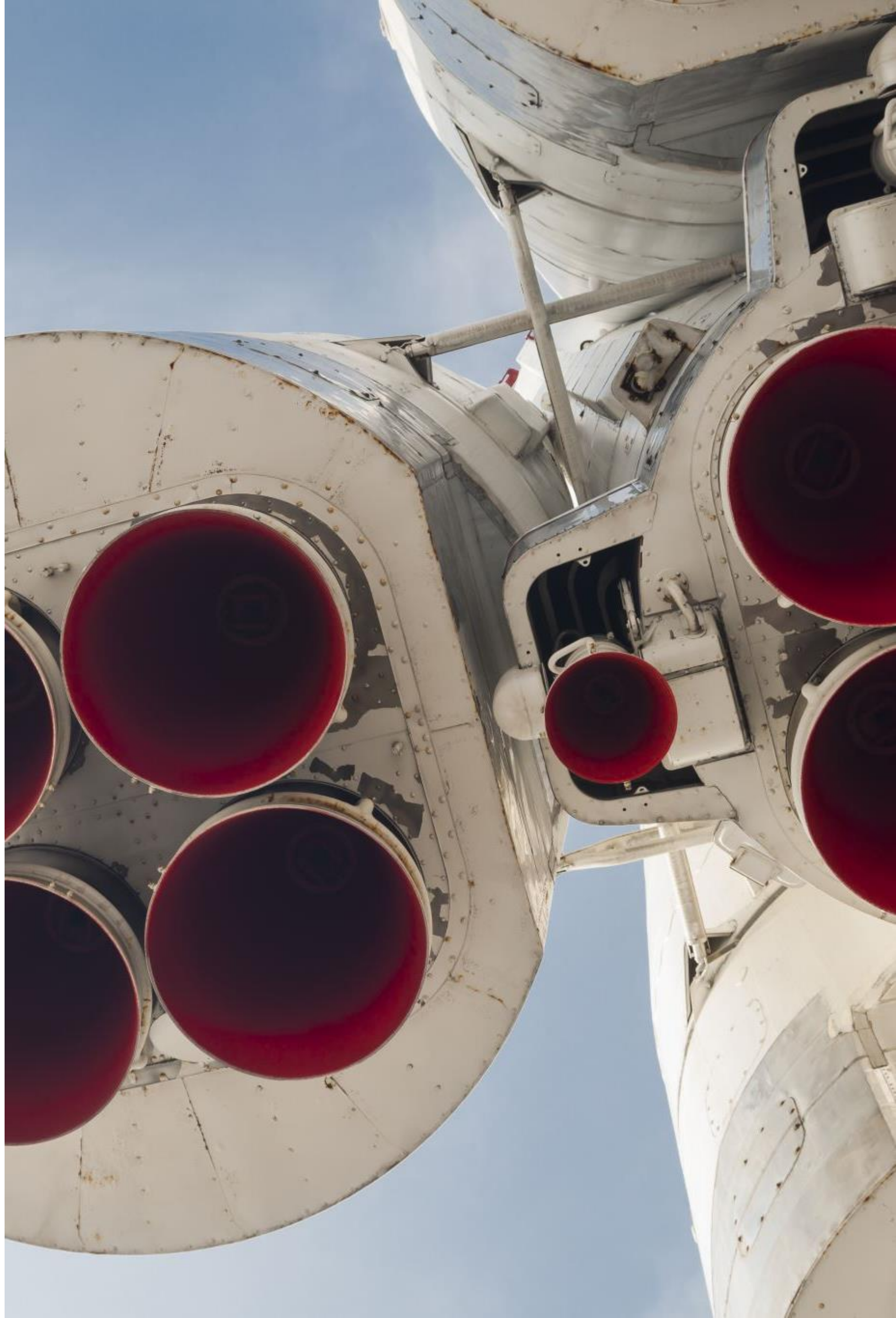
1
45596

Total payload mass carried by boosters launched by NASA (CRS)

EDA SQL
Average Payload Mass by F9 v1.1

1
2928

F9 v1.1 rocket launchers have an average payload mass of 2928 kg.



EDA SQL First Successful Ground Landing Date

landing__outcome
Controlled (ocean)
Failure
Failure (drone ship)
Failure (parachute)
No attempt
Precluded (drone ship)
Success
Success (drone ship)
Success (ground pad)
Uncontrolled (ocean)

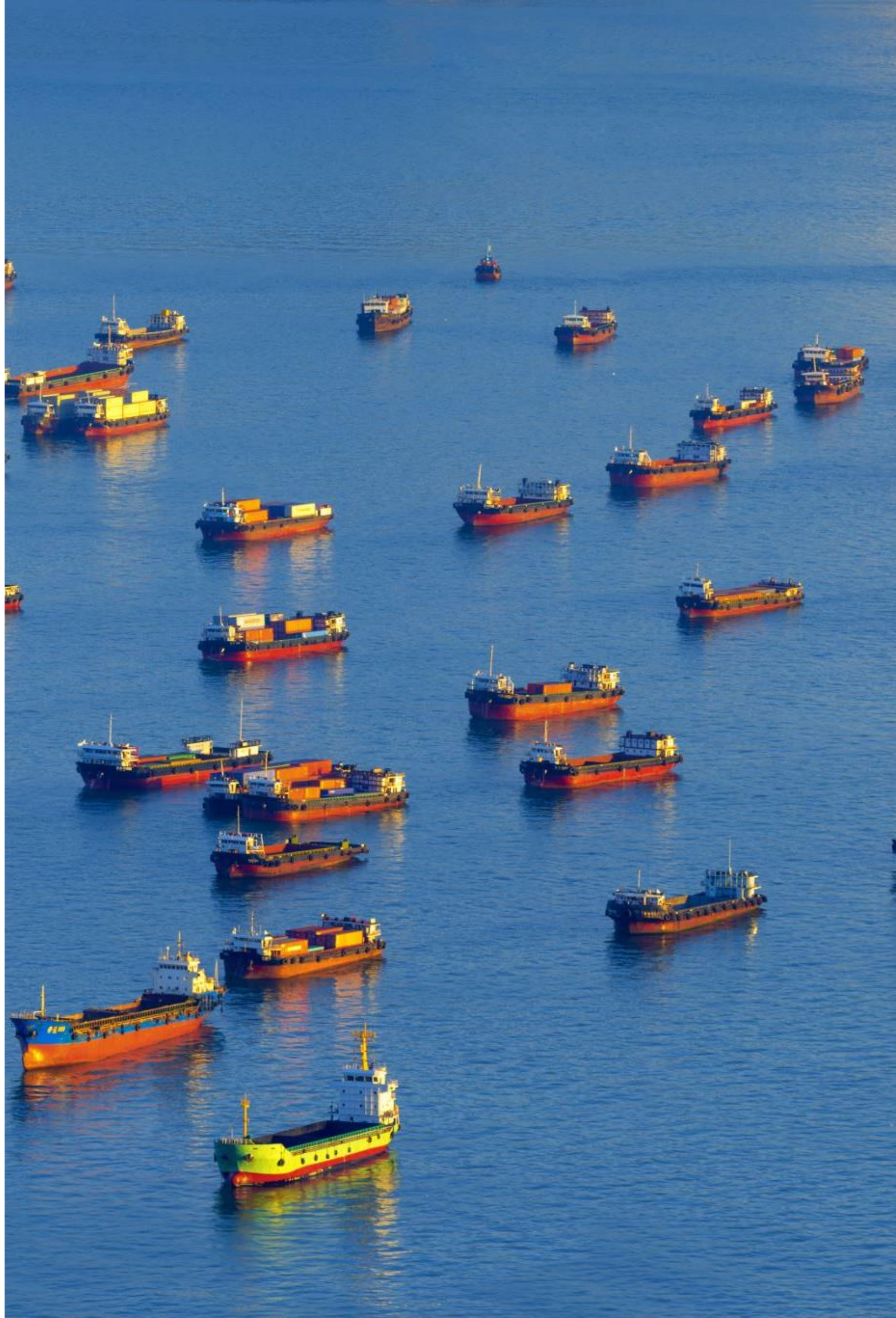
1
2015-12-22

There are 10 different landing outcomes, the first successful Ground Landing date was on 22 December 2015

Successful Drone Ship Landing with Payload between 4000 and 6000

The booster version F9 FT B10XXX have frequent successful landing outcome with heavy payloads.

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



Total Number of Successful and Failure Mission Outcomes

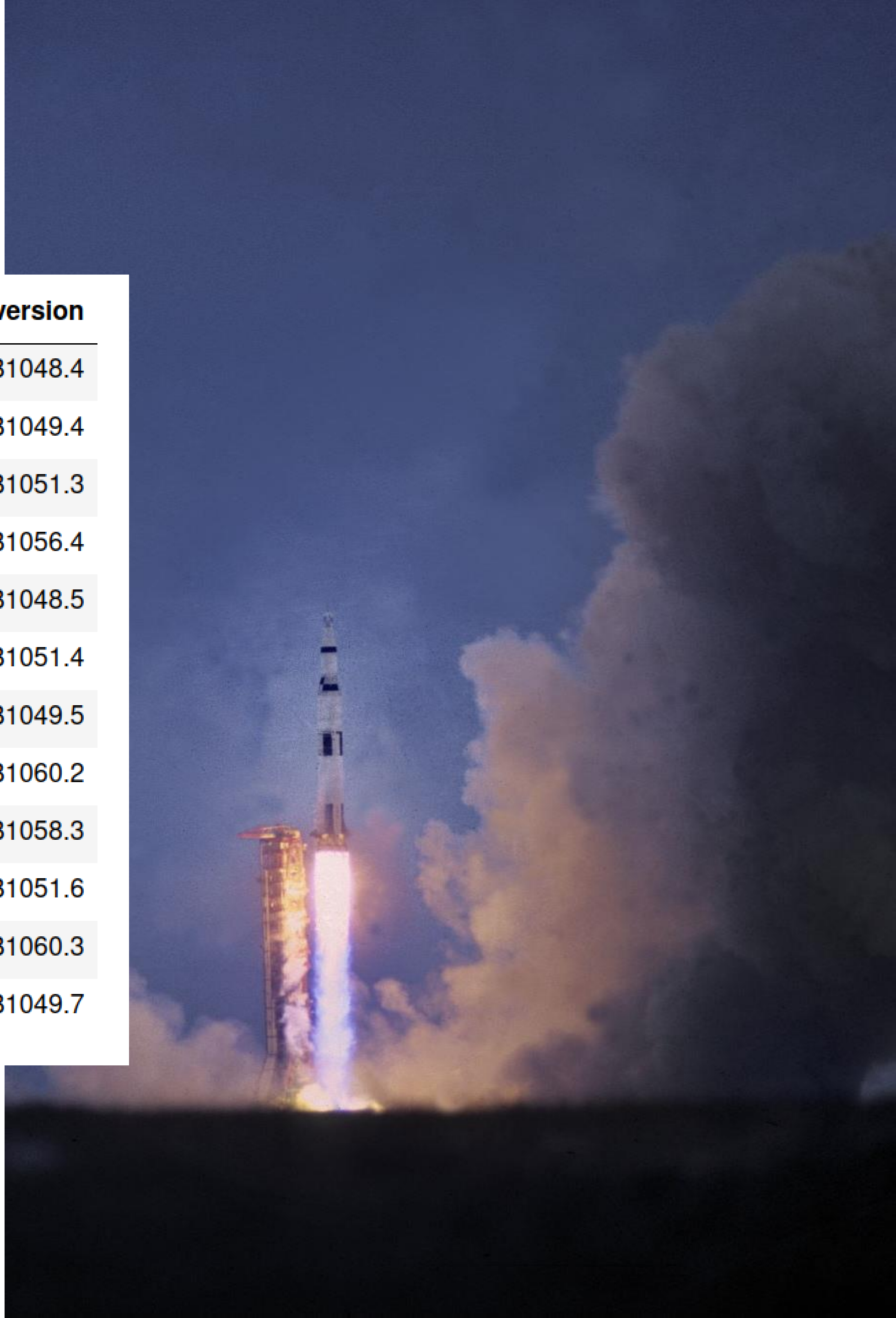
mission_outcome	freq
Failure (in flight)	1
Success	99
Success (payload status unclear)	1

The total number of successful landing is 100 (1 of which has unknow payload status). 1 mission was ended with a failure during landing.

Boosters Carried Maximum Payload

By using a SQL Sub Query, we were able to display Booster version that carried Maximum amount of Payload

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



2015 Launch Records

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

Dataset is queried to retrieve entries of 2015 that were failed to land. We can see that in 2015 the Launch Site CCAFS LC-40 is responsible of for 2 launches.

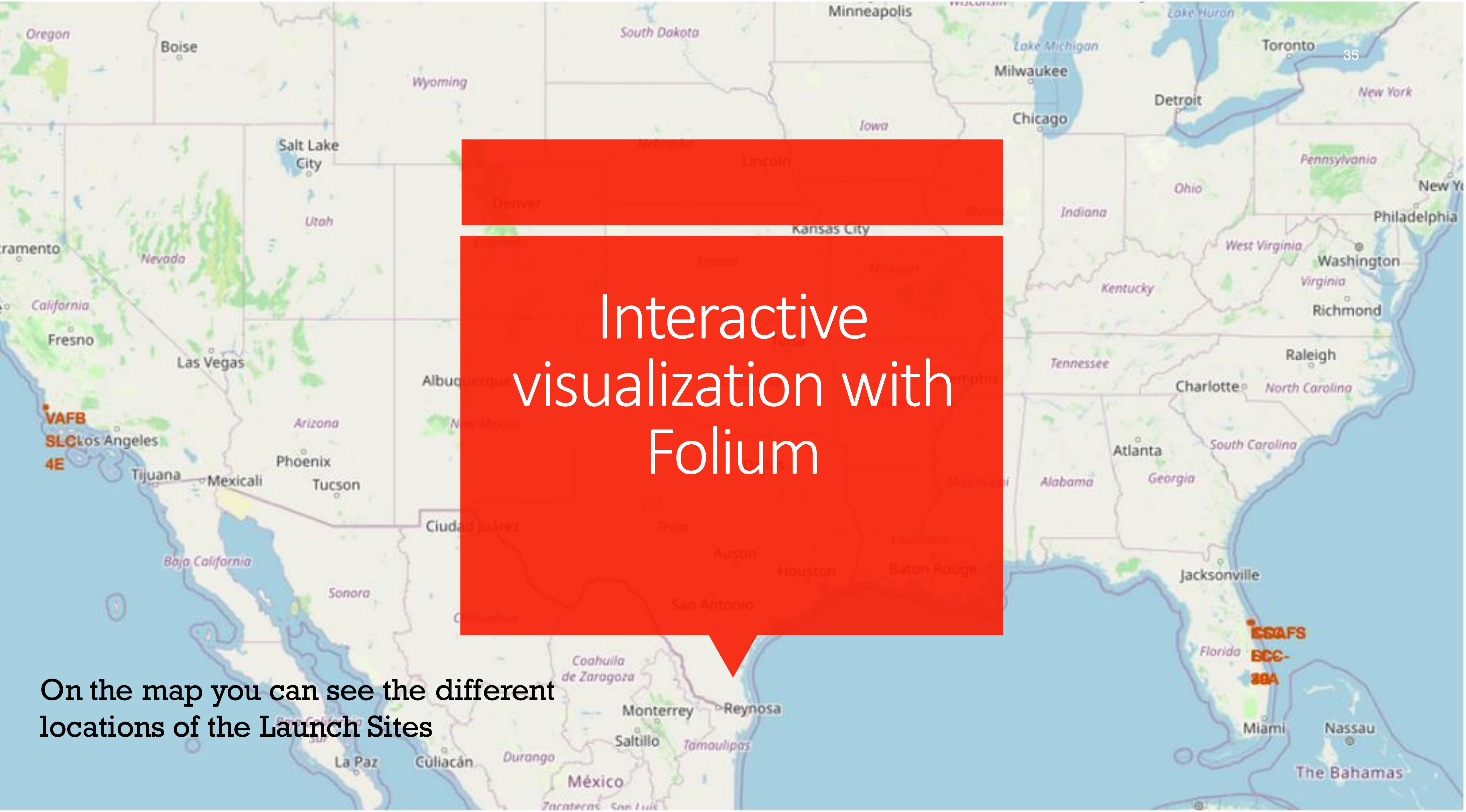
Rank Landing Outcomes Between 2010-06-04 and 2017-03-20

landing__outcome	DATE
No attempt	2017-03-16
Success (ground pad)	2017-02-19
Success (drone ship)	2017-01-14
Success (drone ship)	2016-08-14
Success (ground pad)	2016-07-18
Failure (drone ship)	2016-06-15
Success (drone ship)	2016-05-27
Success (drone ship)	2016-05-06
Success (drone ship)	2016-04-08
Failure (drone ship)	2016-03-04
Failure (drone ship)	2016-01-17
Success (ground pad)	2015-12-22
Precluded (drone ship)	2015-06-28
No attempt	2015-04-27
Failure (drone ship)	2015-04-14
No attempt	2015-03-02
Controlled (ocean)	2015-02-11
Failure (drone ship)	2015-01-10
Uncontrolled (ocean)	2014-09-21
No attempt	2014-09-07
No attempt	2014-08-05
Controlled (ocean)	2014-07-14
Controlled (ocean)	2014-04-18
No attempt	2014-01-06
No attempt	2013-12-03
Uncontrolled (ocean)	2013-09-29
No attempt	2013-03-01
No attempt	2012-10-08
No attempt	2012-05-22
Failure (parachute)	2010-12-08
Failure (parachute)	2010-06-04

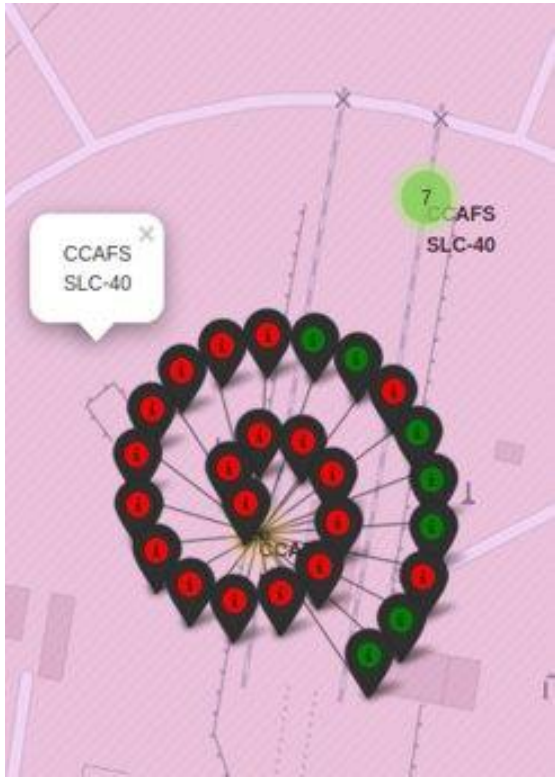
Section 3

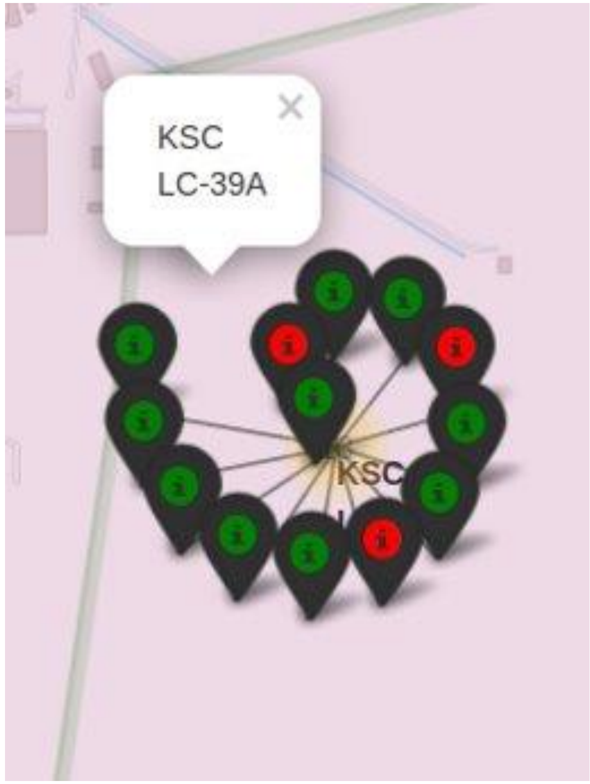
Launch Sites Proximities Analysis

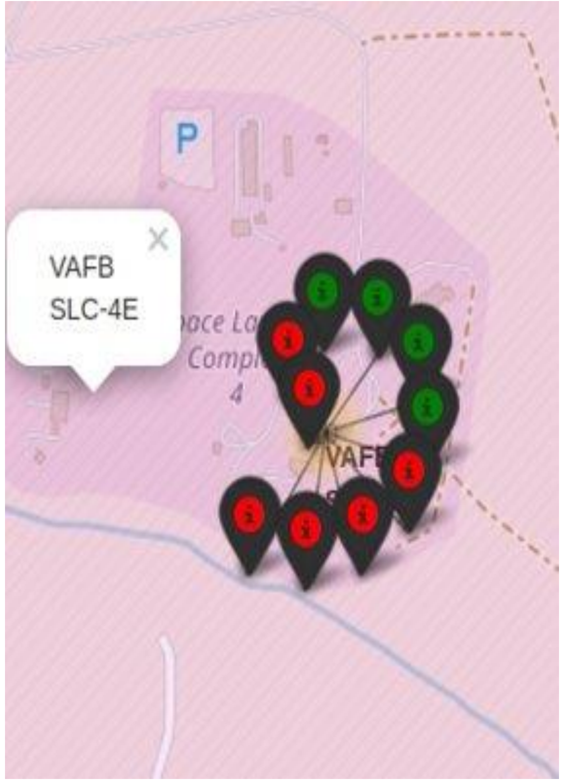
Interactive visualization with Folium



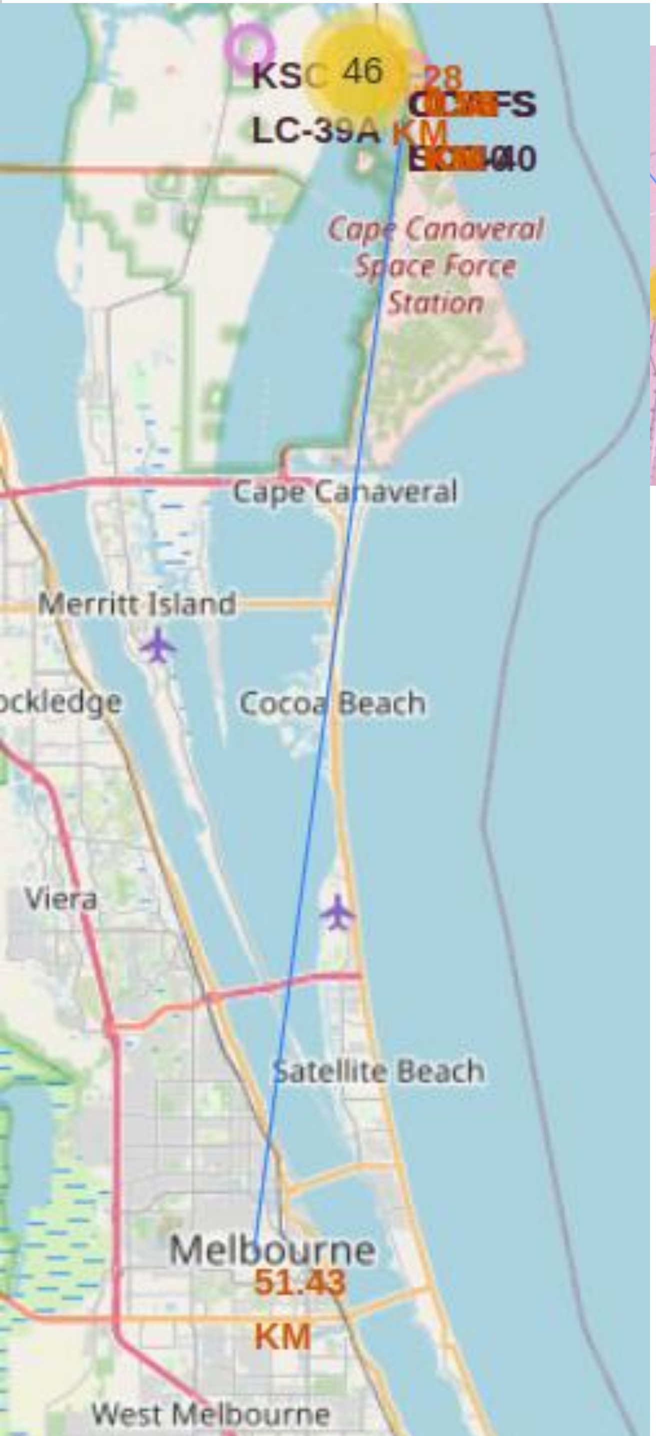
On the map you can see the different locations of the Launch Sites

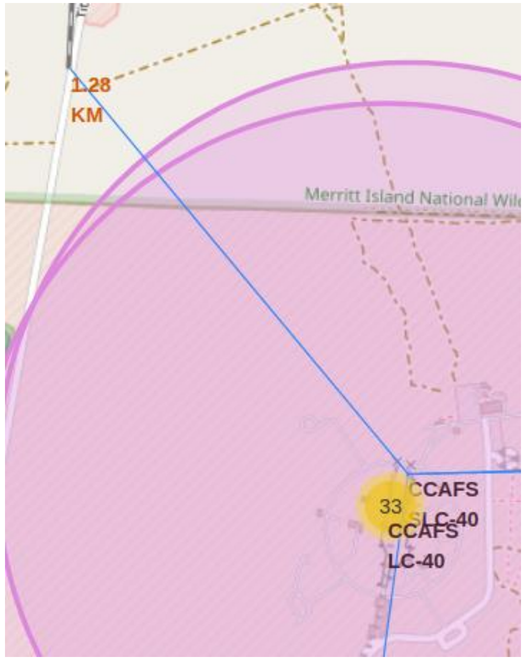






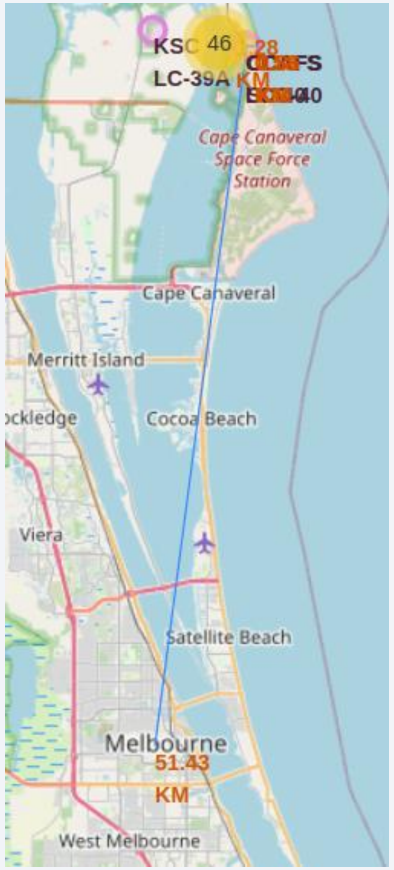
Green Markers are showing the successful launches and the red ones are showing the failed launches

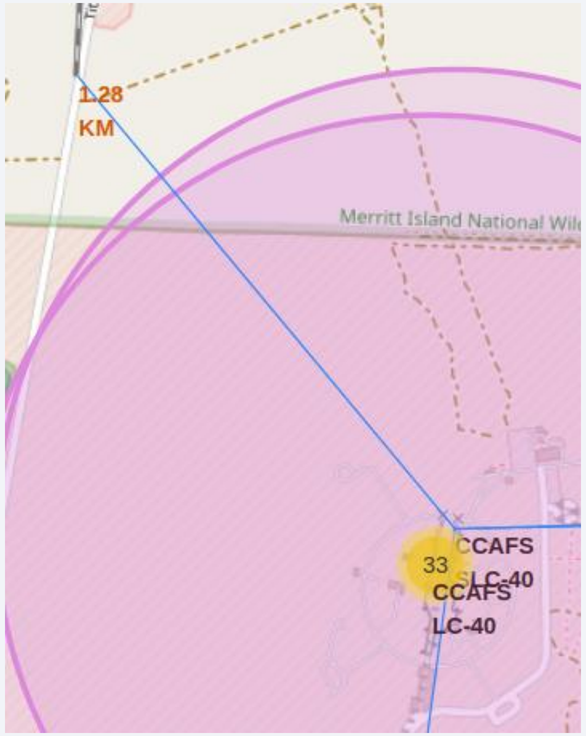




<Folium Map Screenshot 3>

<Folium Map Screenshot 3>



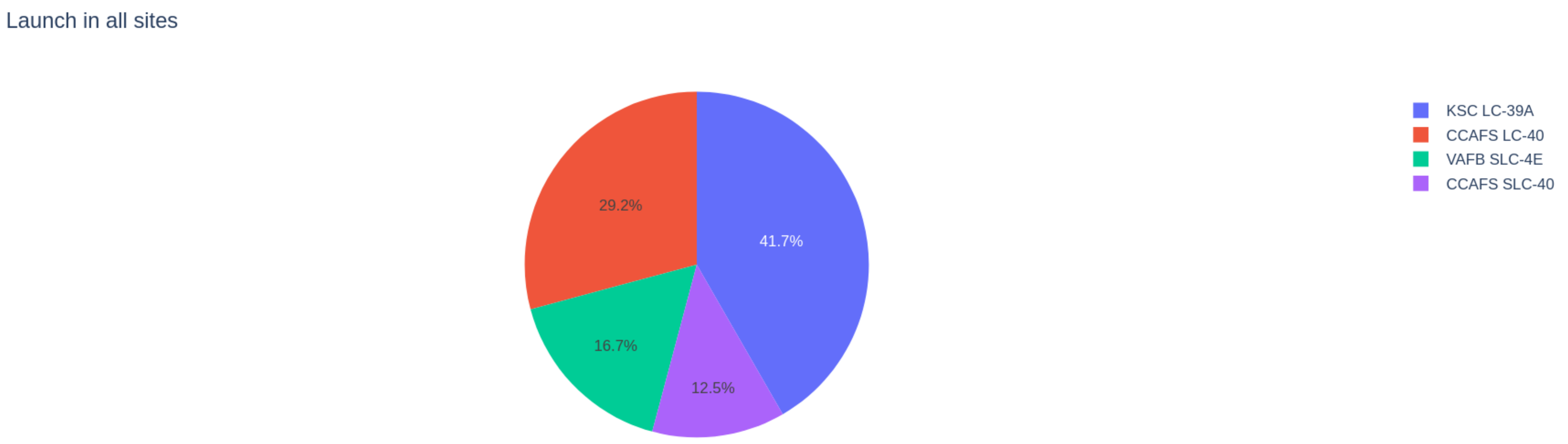


Section 4

Launch Sites Dashboard with Plotly Dash

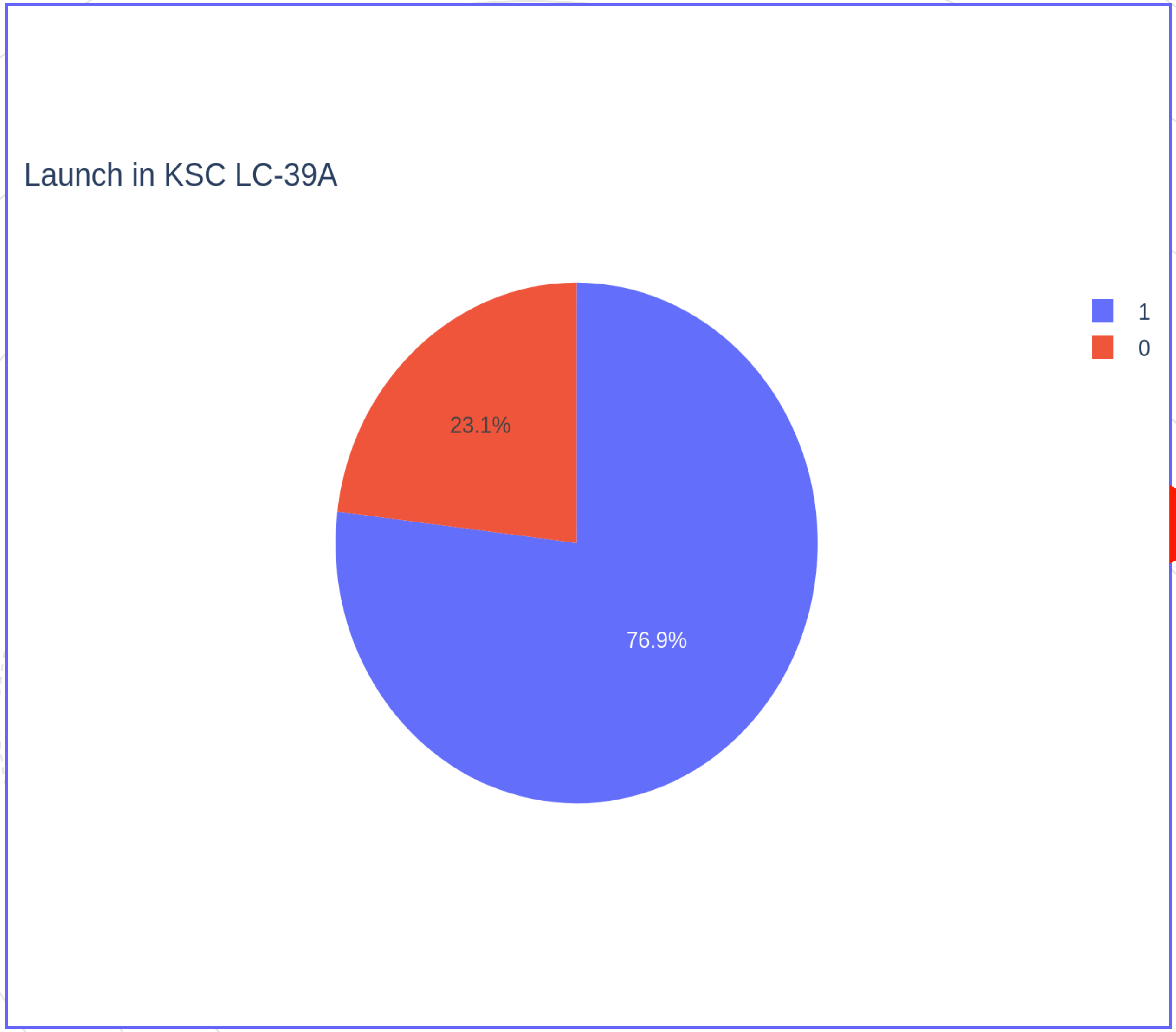
SpaceX Dashboard - Launch Records

Launch Sites - Count of Launch Success



The highest number of successful launches took place in the Launch Site KSC LC – 39A (

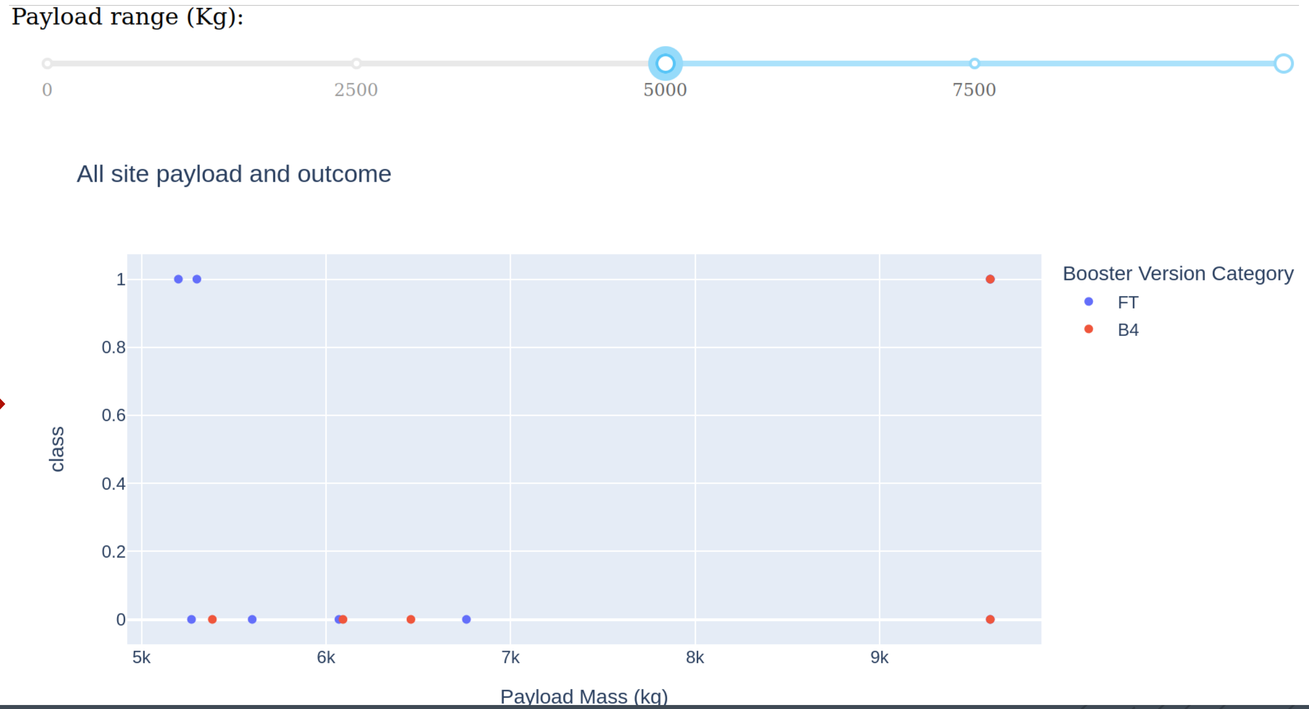
Launch Site with highest number of successful outcome



Approximately 77% of the launches from KSC LC-39A has landed successfully.

<Dashboard Screenshot 3>

While all launch sites selected many successful launches had payload mass between 2000 KG – 5000 KG.
The Booster version FT had the highest number of good outcome.



When the payload mass is set to 5000 KG and above, there are only 3 successful outcome, two of them having the booster version FT.

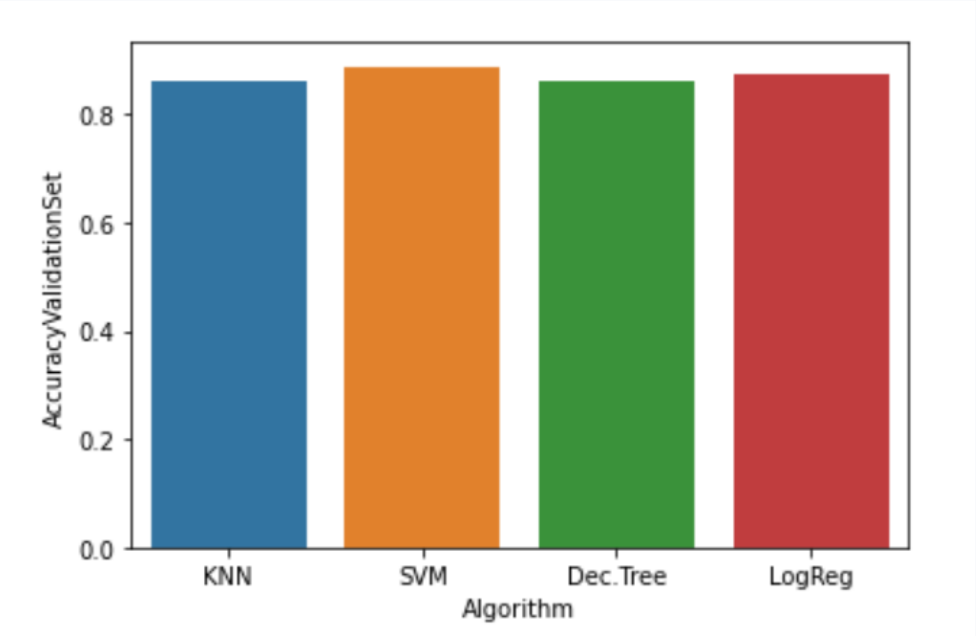
Section 5

Predictive Analysis (Classification)

Classification Accuracy

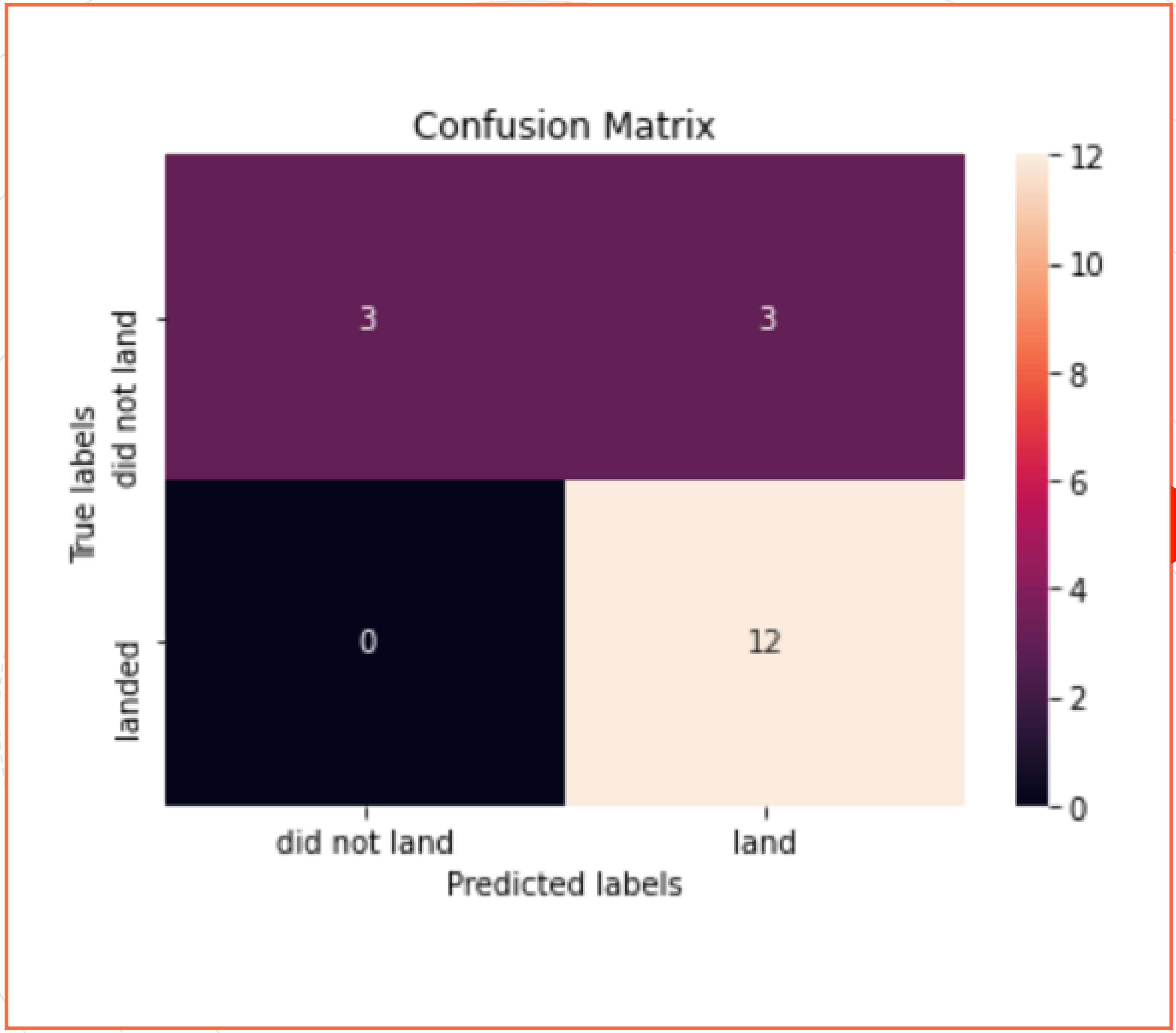
Different classification models are implemented on the prepared, normalized and split dataset.

	Algorithm	AccuracyTrainingSet
1	SVM	0.889000
3	LogReg	0.875000
2	Dec.Tree	0.861111
0	KNN	0.860000



Even though the models are performing close to each other, SVM (Support Vector Machine) is the leading classifier, with an accuracy score of 89%.

Confusion Matrix



The model is confident during the prediction of successful landings. (True Positive)

However False Negative and False Positive are equally predicted. So the model is unsure during a prediction of failed landings.

Conclusions

- Yearly trend shows an increasing success rate in launches, successful outcome is increasing by year: meaning that SpaceX is gaining significant experience.
- Payload Mass between 0 KG-5000 KG has higher chance of landing successfully.
- Launch Site KSC LC-39A has the highest number of successful launches
- Orbits ES-L1, GEO, HEO, SSO, VLEO had the most success rate.
- Support Vector Machine model is fitting the dataset better than other models, as a result slightly higher accuracy

Appendix

1. GitHub Repo: https://github.com/GlobalCJ/Pro-Data-Sci-Cert-Coursera
2. Final Rapport: https://github.com/GlobalCJ/Pro-Data-Sci-Cert-Coursera/blob/main/finalrapport.pdf
3. Dataset used SpaceX API: https://api.spacexdata.com/v4/
4. Dataset used Wikipedia: https://en.wikipedia.org/w/index.php?title=List_of_Falcon_9_and_

Thank you!