Winning Space Race with Data Science

Ian Christopher Roman
08/07/2023

Outline

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

Executive Summary

- Summary of methodologies
  - Data collection through SpaceX API and Web Scraping
  - Data Wrangling
  - Exploratory Data Analysis (EDA) using SQL queries and Data Visualization
  - Interactive Visual Analysis using Folium and Plotly
  - Predictive Analysis – Machine Learning (various Classification models)
- Summary of all results
  - Results of EDA and Predictive Analysis
  - Static and Interactive Visualizations

Introduction

- Project background and context
  - SpaceX advertises Falcon 9 rocket launches on its website with a cost of $62M while other providers will cost more than $165M because Falcon 9 reuses its first stage of the rocket itself. Typically, most rocket's first stage does not land and examining historical launch data, it will be possible to determine if the first stage will land successfully. This information will be valuable to alternate companies whose bidding against SpaceX for a successful rocket launch. The objective of this data science project is to perform data analysis to existing launch data and to create a machine learning pipeline in predicting whether the first stage of the rocket will land successfully.
- Problems you want to find answers
  - What are the attributes / factors affecting success of rocket landing?
    - How are these attributes / factors related to one another and how would these affect a/an successful/unsuccessful landing?
  - What are the required operational conditions to guarantee the success of a landing program?

Section 1

Methodology

Methodology

Executive Summary

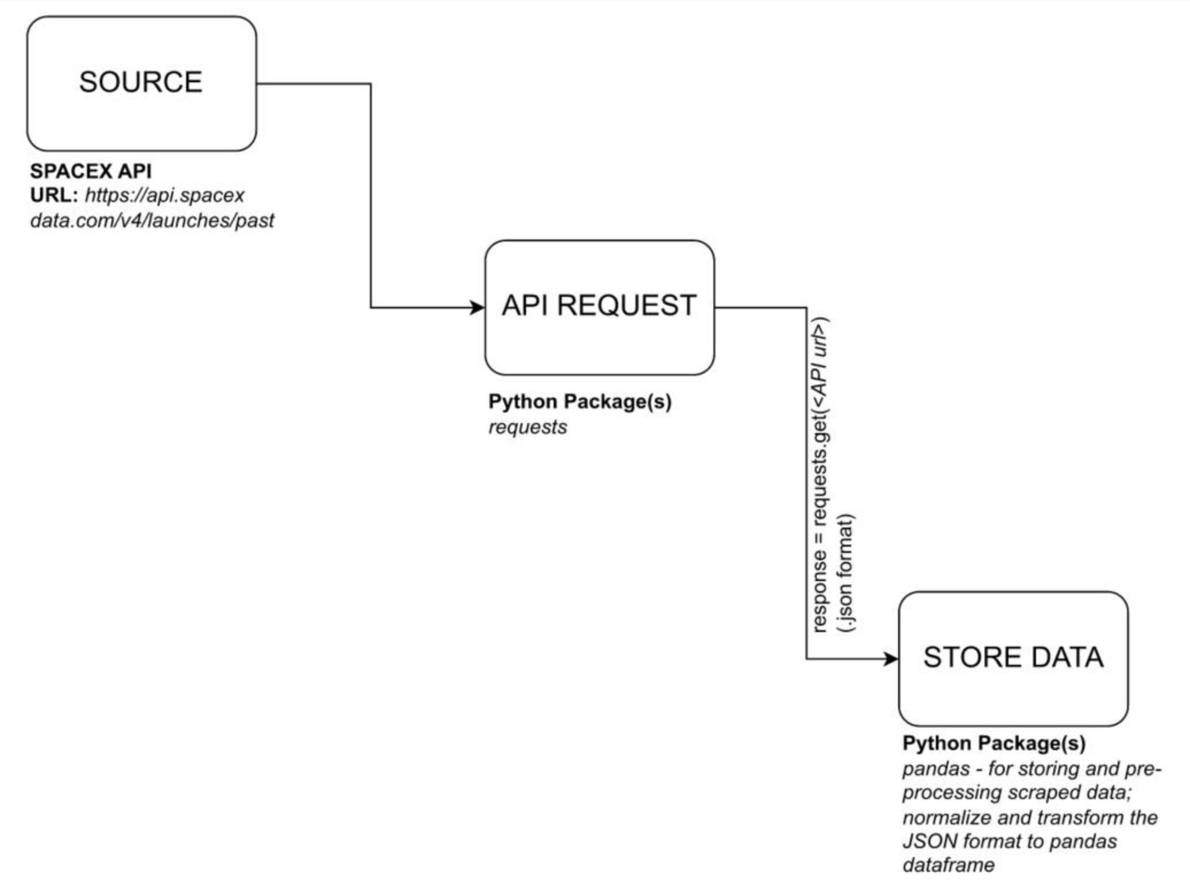
- Data collection methodology:
  - Data was collected using SpaceX API and Web Scraping other information from Wikipedia
- Perform data wrangling
  - Categorical features on dataset underwent one-hot encoding
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - Classification Model used – Logistic Regression, Support Vector Machines (SVM), Decision Tree, and K Nearest Neighbor (KNN)
  - Hyperparameters were tuned using GridSearch cross – validation (GridSearchCV)
  - Accuracy and score were determined for each classification method used as well as Confusion matrices were plotted

Data Collection

- To gather the necessary data for analysis and modeling, the following steps were undertaken:
  - SpaceX API:
    1. GET request to SpaceX API URL
    2. Data received in JSON/.json format; transformed and normalized using `.json_normalized( )` method
    3. Data pre-processing
  - Web Scraping
    1. Web Source: [Wikipedia – List of Falcon 9 and Falcon Heavy Launches]
    2. HTTP GET to the Wikipedia page
    3. Scrape web data using *BeautifulSoup* package
    4. Data pre-processing

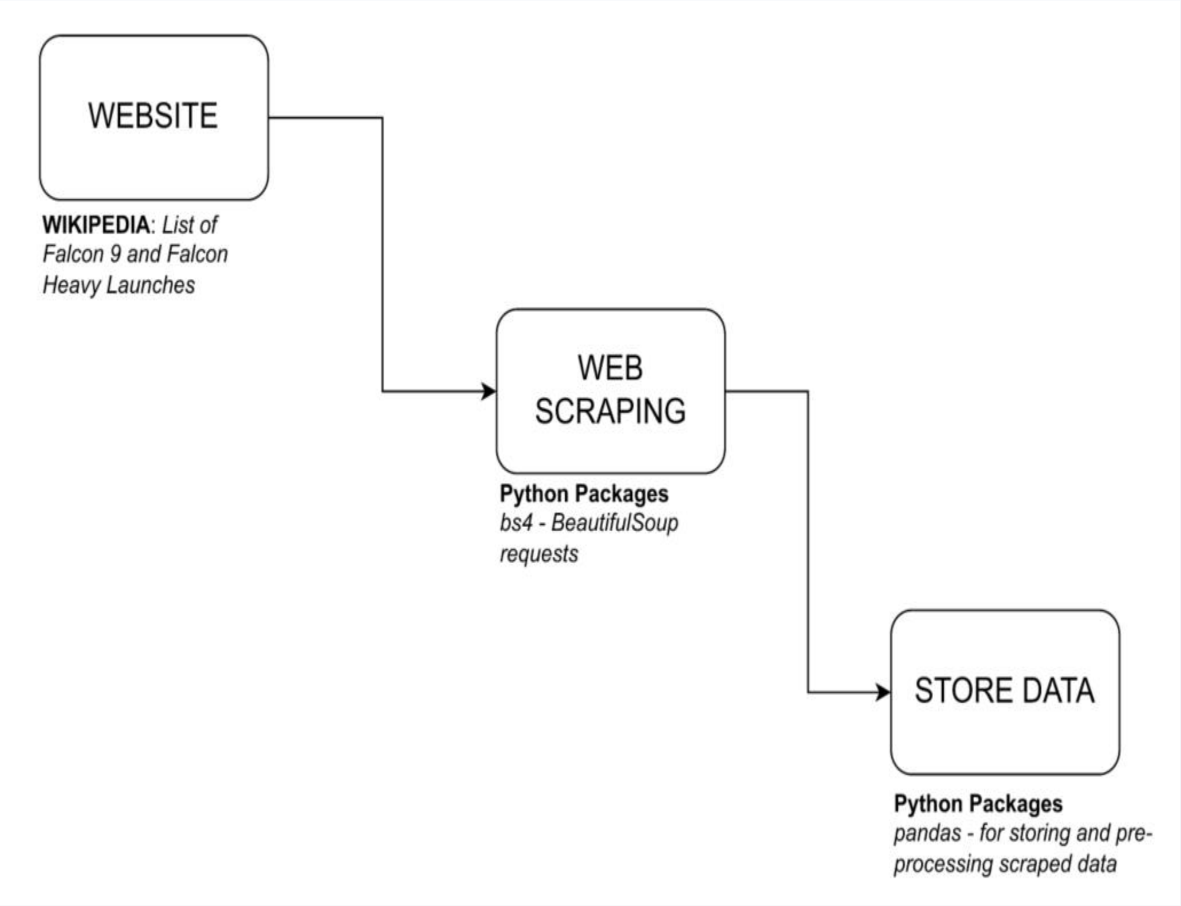
Data Collection – SpaceX API

- A GET request was sent to SpaceX API URL (e.g. `response = requests.get(spacex_url)`)
- JSON was the default data format; using `.json_normalize( )` method in *pandas*, data is converted to a *pandas* data frame ready for pre-processing
- Click HERE to go to the Jupyter Notebook.



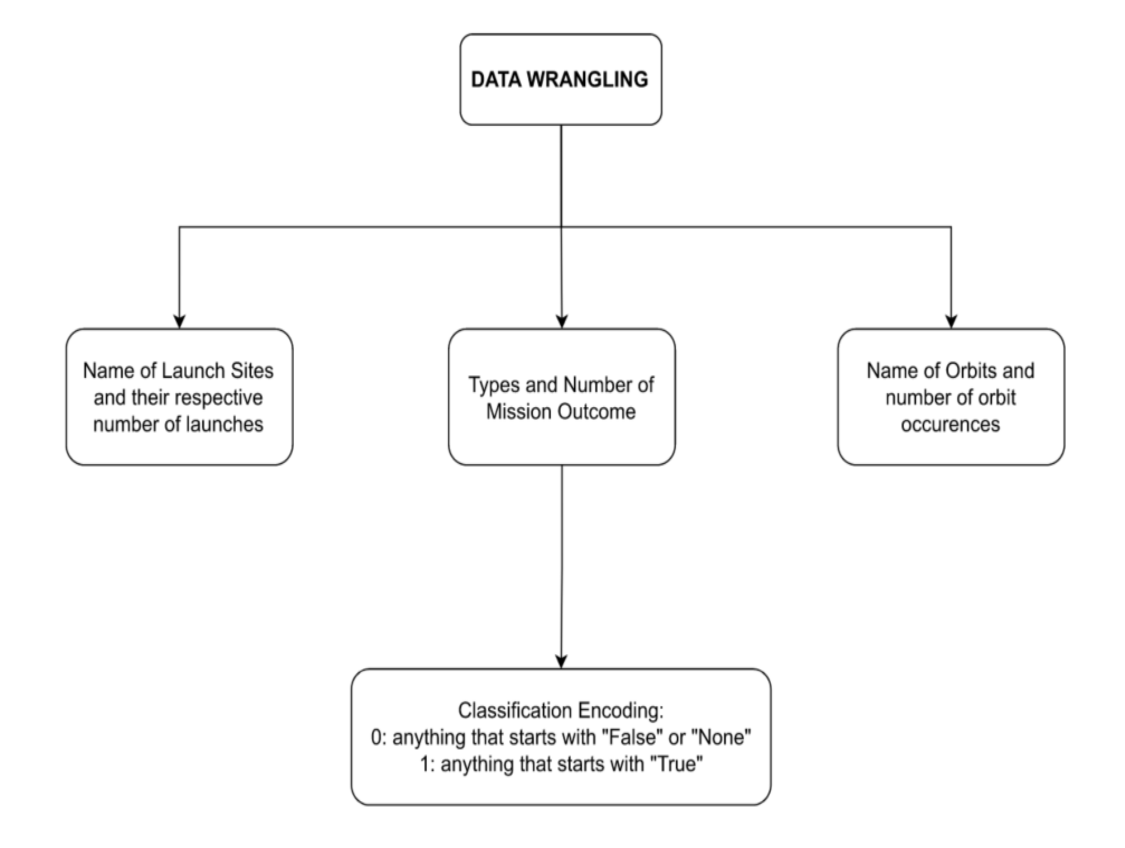
Data Collection – Web Scraping

- HTTP GET request sent to the Falcon 9 Launch Wikipedia page
- Data was parsed and stored as an HTML table using the *BeautifulSoup* package
- Click HERE to go to the Jupyter Notebook.



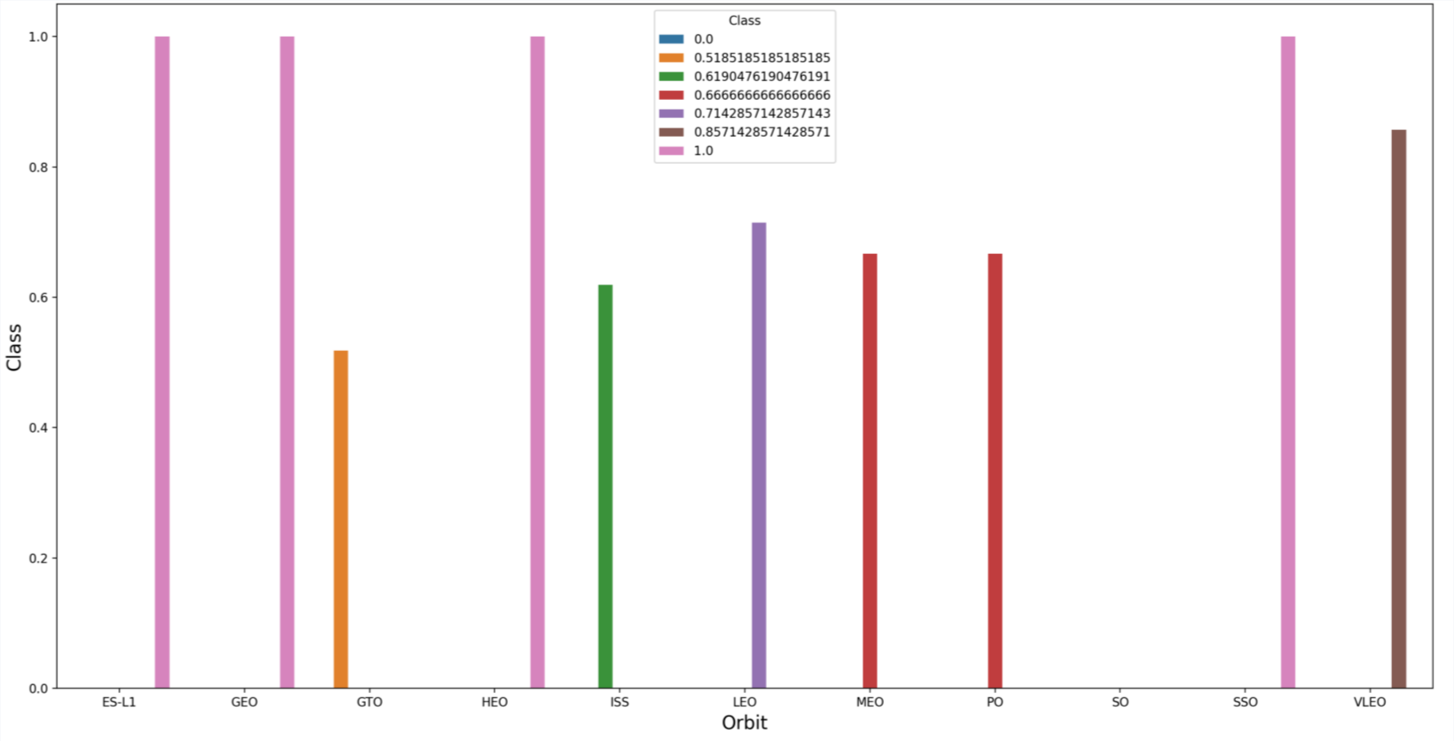
Data Wrangling

- Performed summarization on:
  - No. of launches on each Launch Sites
  - Orbit Name and occurrences
  - Type and No. of mission outcome
- Assigned model training label for the mission outcome column to a new column called `class`
- Click HERE to go to the Jupyter Notebook



EDA with Data Visualization

- Plotting among the various attributes (e.g., flight number, launch site, payload mass, number of flights, orbit, launch success) allowed for the observation of relationships and trends helpful in understanding what drives the success of a launch.
- Click [HERE]() to go to the Jupyter Notebook



EDA with SQL

- Using SQLAlchemy to establish a database connection, the following SQL queries were performed:
    - Names of unique existing launch sites
    - Total payload mass (kg) launched by NASA (CRS)
    - Average payload mass (kg) carried by booster version F9 v1.1
    - Date of the first successful landing on a ground pad
    - Names of booster version which have a success landing on a drone ship with a payload mass between 4000 and 6000 kg
    - Total number of success/fail mission outcomes
    - Names of booster version which carried a maximum payload mass (required a *subquery*)
    - Fail landing outcomes on a drone ship given a date constraint
    - Summary of all landing outcome and occurrences with date constraints
- Click HERE to go to the Jupyter Notebook

Build an Interactive Map with Folium

- We designated all the launch sites and incorporated map elements like markers, circles, and lines to indicate the outcomes of launches – whether successful or unsuccessful – for each site on the Folium map.
- Assigned feature launch outcome labels for modeling: 0 for failure and 1 for success
- Calculated distances from the launch sites to key point of interests such as nearest highway, coastline, and cities
- Click HERE to go to the Jupyter Notebook

Build a Dashboard with Plotly Dash

- Built an interactive dashboard using *Plotly dash* package
- Dashboard components:
  - A **pie chart** exhibiting percentages for each launch sites with respect to the total launch.
  - A **scatter plot** exhibiting the relationship of Outcome and Payload Mass (kg) for different Booster Versions. It also has a Payload Range slider to visualize the effect of low and heavy payload weight.
- Click HERE to go to the `.py` file

Predictive Analysis (Classification)

- Performed a train/test split (i.e., 80/20 split, 80% for training, 20% for testing)
- Developed various classification models and used feature engineering and hyperparameter tuning using grid search cross – validation (GridSearchCV)
- Accuracy and confusion matrices were the evaluation metrics used
- Best performing classification model was determined based on the calculated metrics
- Click HERE to go to the Jupyter Notebook

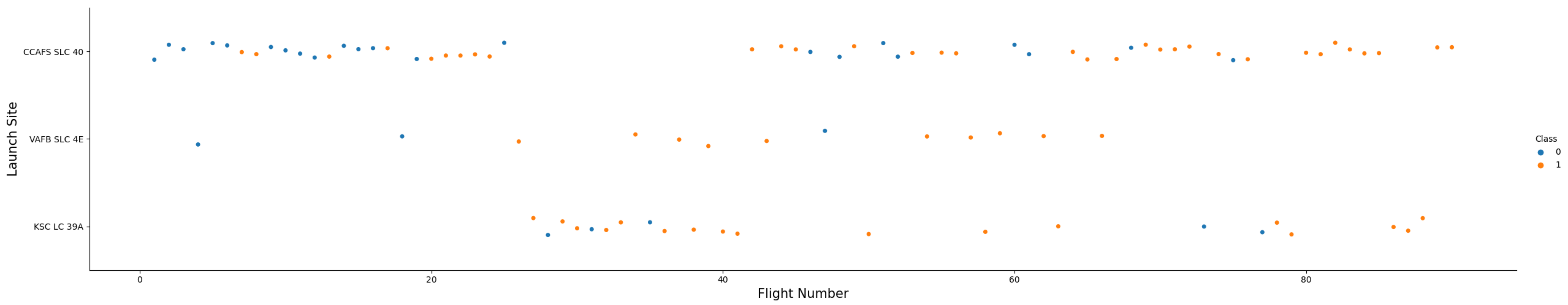
Results

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

Section 2

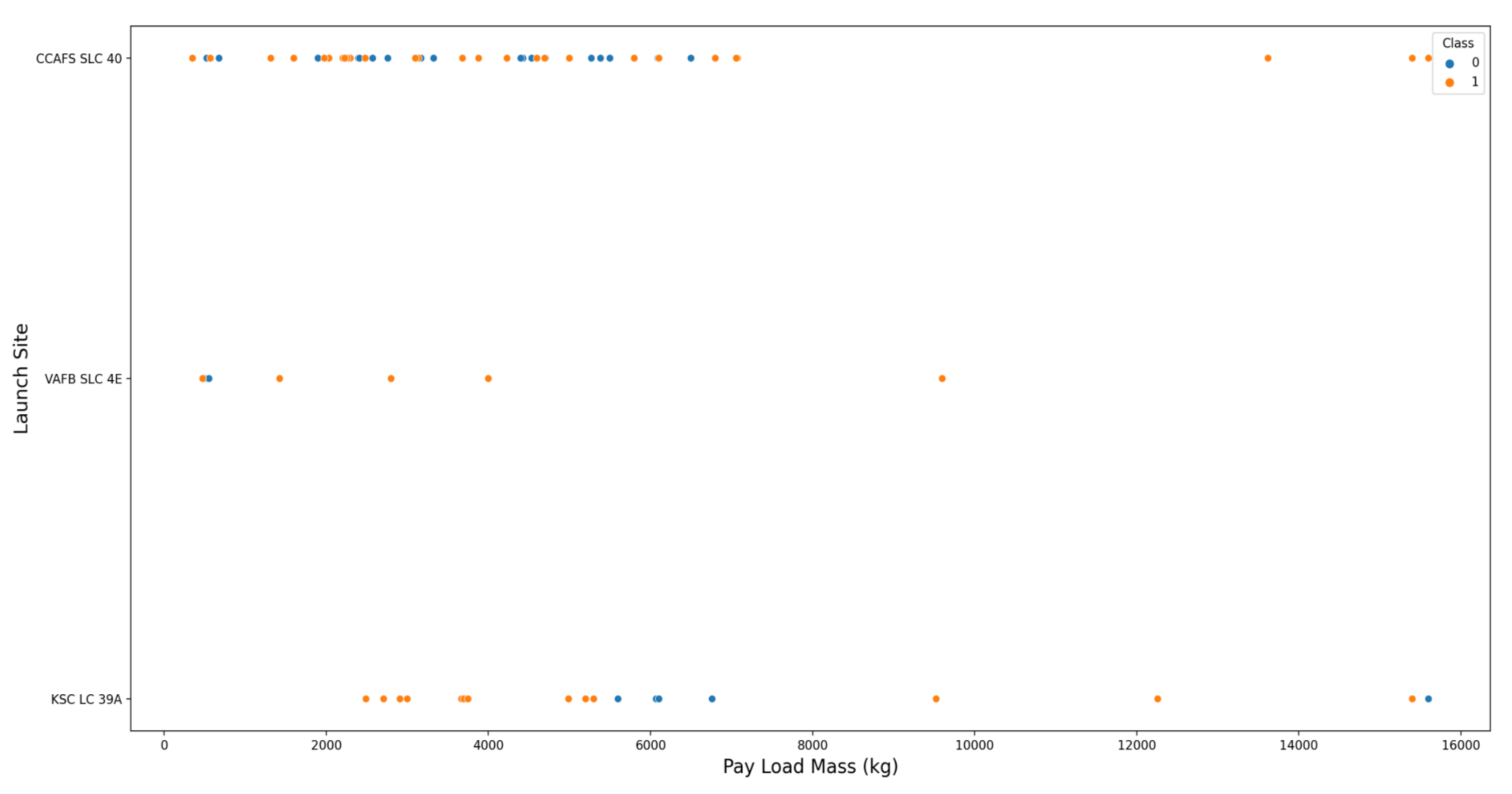
Insights drawn from EDA

Flight Number vs. Launch Site



- Key observation: Greater number of flights at each launch site yields greater rocket landing success

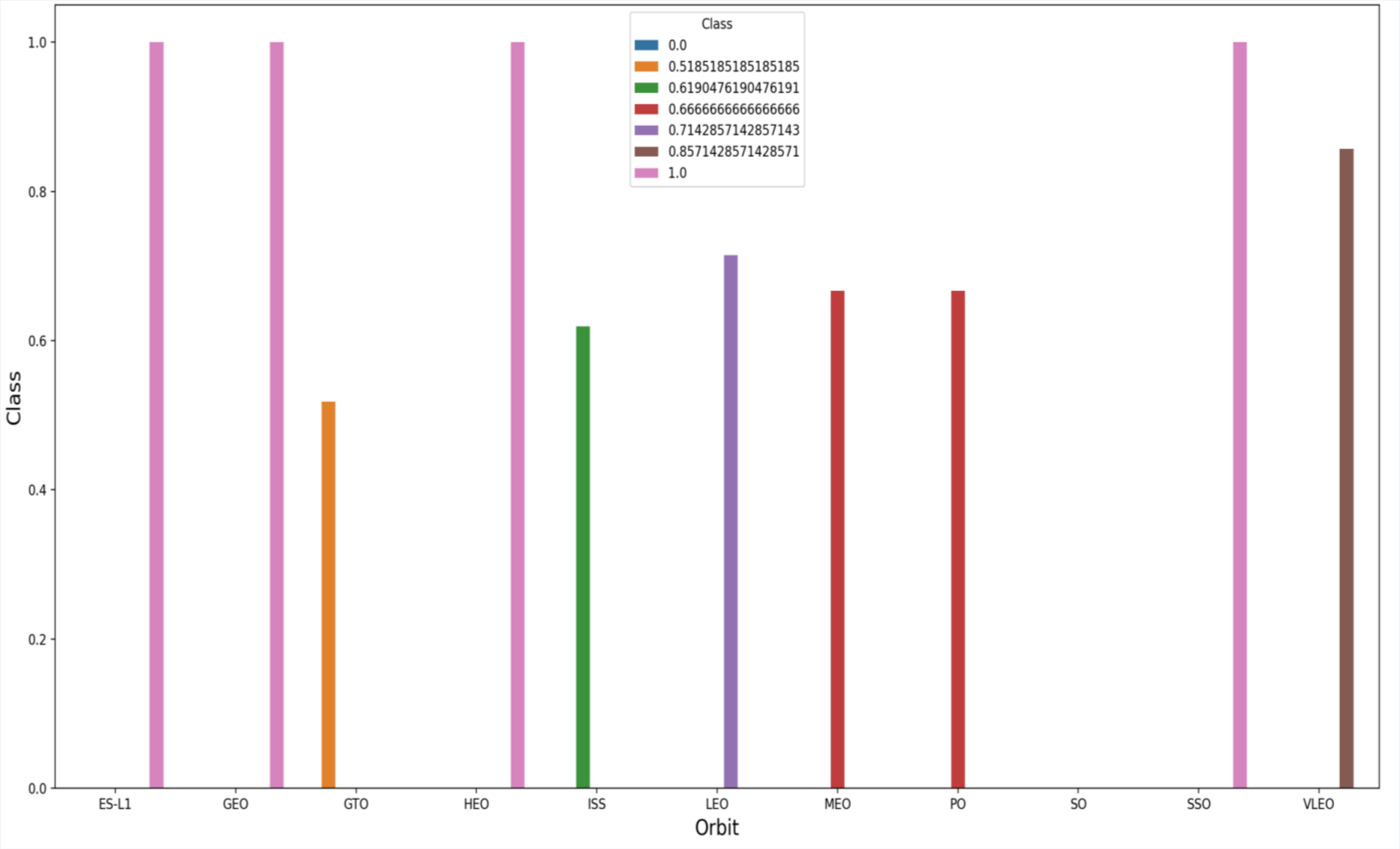
Payload vs. Launch Site



- Key observations:
  - **CCSFS SLC 40** – as payload mass increase, there is a high likelihood of rocket landing
  - **VAFB-SLC** – no rocket launched for payload greater than 10,000 kg (10 tonnes)
  - **KDC LC 39A** – no rocket was launched at payloads lesser than 2000 kg (2 tonnes)

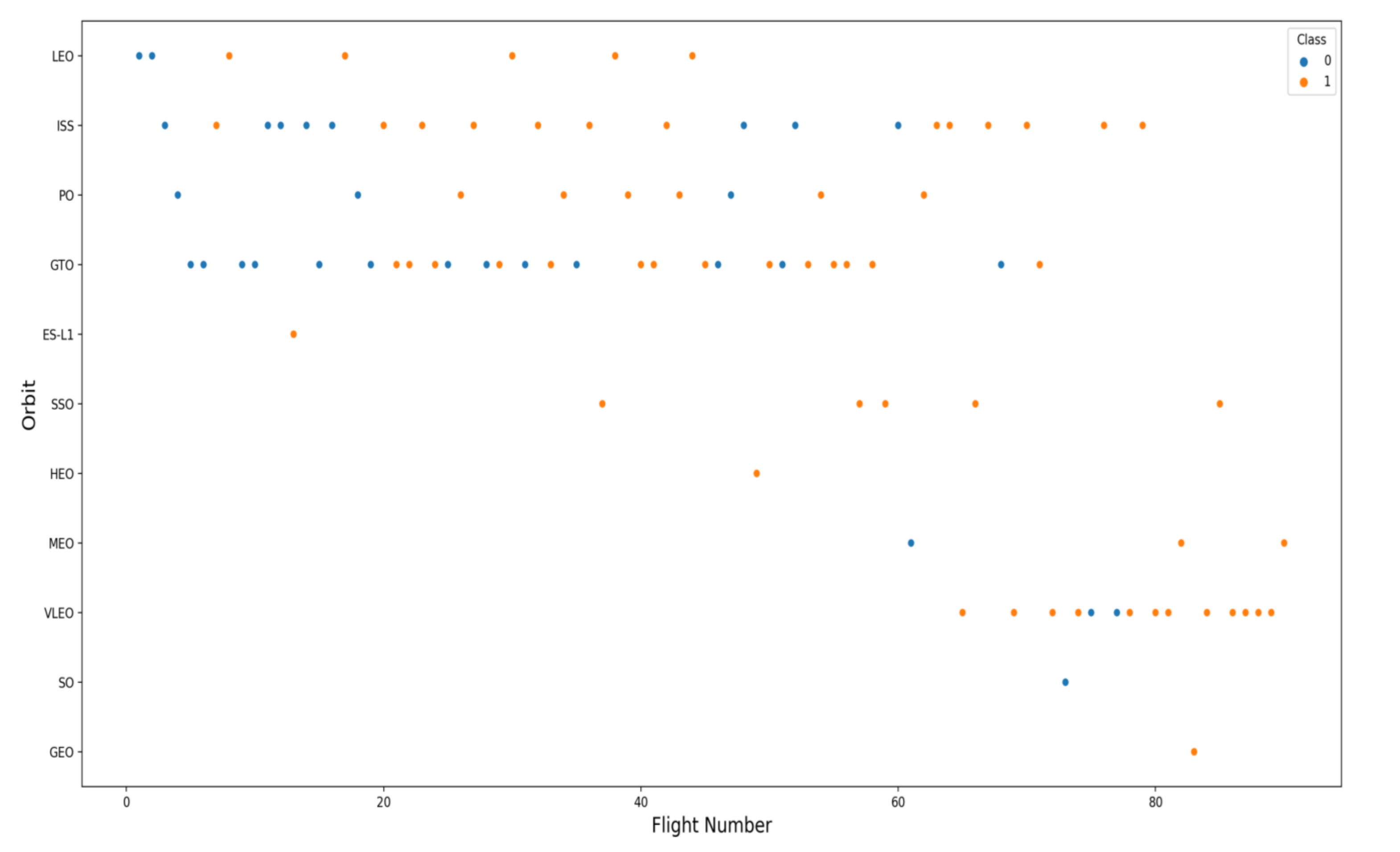
Success Rate vs. Orbit Type

- Key Observation: the following orbits have the highest success rate:
  - ES – L1
  - GEO
  - HEO
  - SSO

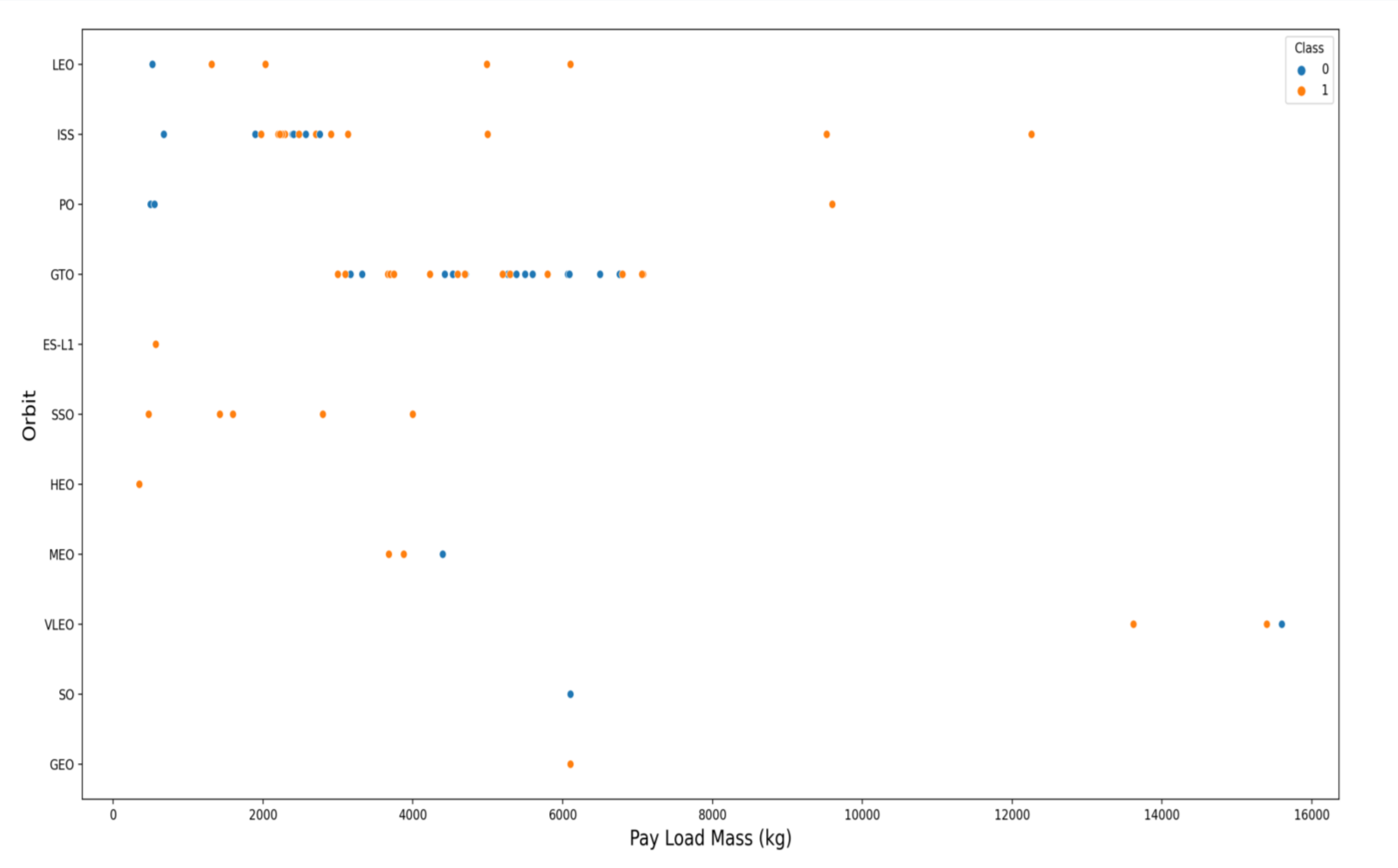


Flight Number vs. Orbit Type

- Key Observations:
  - Success rate of LEO orbit **<u>depends</u>** on the flight number
  - Success rate of GEO orbit **<u>does not depend on the flight</u>** number
  - Success rate of VLEO orbit is more prevalent when flight is more than 60

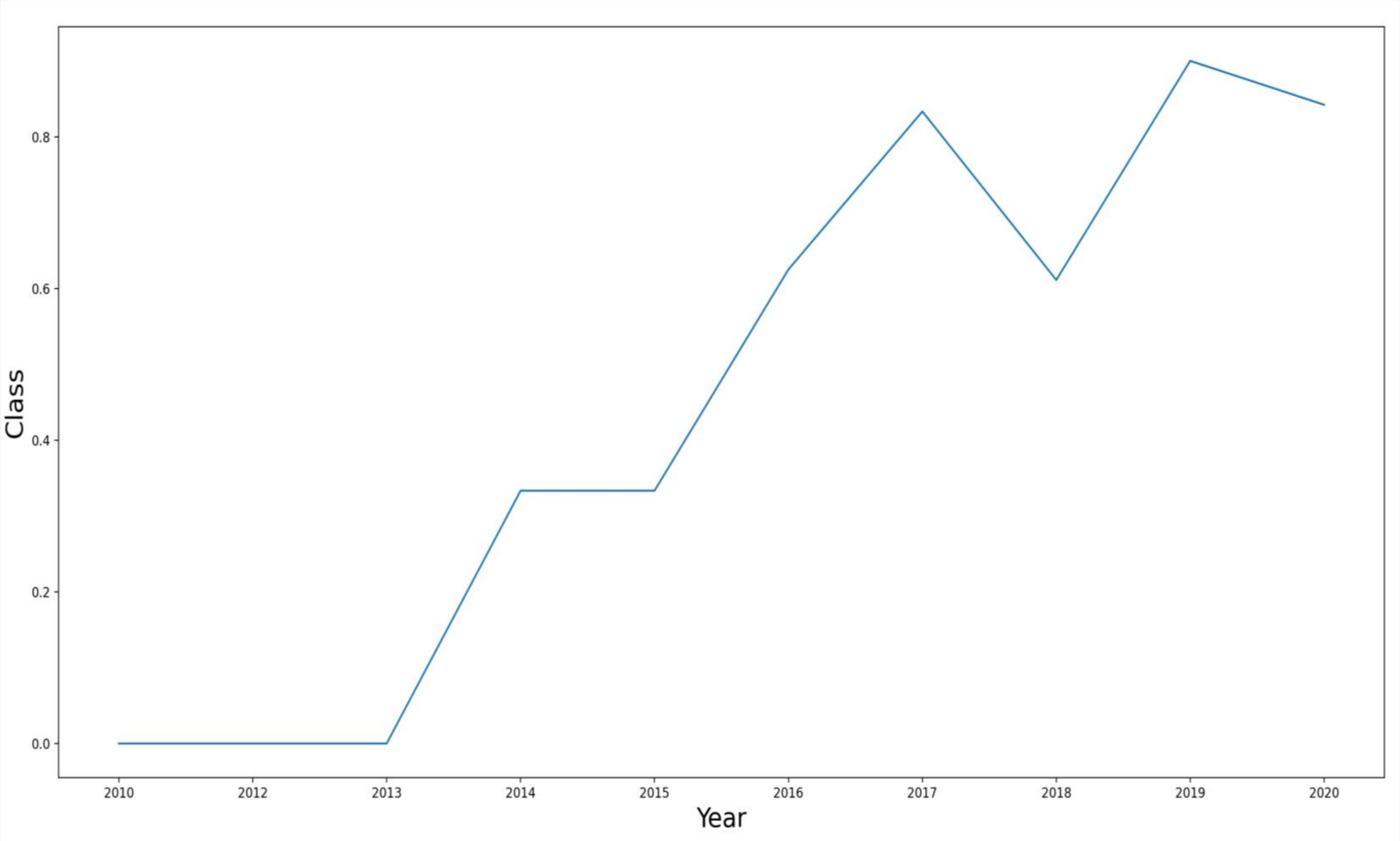


Payload vs. Orbit Type



- Key Observations:
  - Success rates are more prominent at heavier payload (>10000 kg) for LEO, ISS and Polar orbits
  - There is no observable relationship between orbit GTO and Payload Mass

Launch Success Yearly Trend



- The line chart shown is the average success rate for each year from 2013 to 2020
- Progressively increasing with minor fluctuations (e.g., major “dip” in 2018)

All Launch Site Names

- As shown in the query below, the resulting table was the unique launch site names and was able to extract this information using `DISTINCT( )` function.

```
%sql select distinct(launch_site) from SPACEXTBL
```

```
 * sqlite:///my_data1.db
Done.
```

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

Launch Site Names Begin with 'CCA'

- Query and results are shown.
- The `LIKE` command was used to filter through the columns for the launch site names that starts with the string 'CCA'.
- the `LIMIT` function only displayed a specific number of records (rows)

```
%%sql
select * from SPACEXTBL
where launch_site like "CCA%"
limit 5
```

```
* sqlite:///my_data1.db
Done.
```

Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Orbit	Customer	Mission_Outcome	Landing_Outcome
2010-04-06	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success	Failure (parachute)
2010-08-12	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
2012-08-10	00:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500	LEO (ISS)	NASA (CRS)	Success	No attempt
2013-01-03	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677	LEO (ISS)	NASA (CRS)	Success	No attempt

Total Payload Mass

- Query and result are shown below.
- Total payload mass was calculated using the `SUM` aggregation function and constraining the result to `customer` as "NASA (CRS)"
- Total payload mass is 45,596 kg (~46 tones)

```
%%sql
select sum(payload_mass__kg_) as 'Total Payload Mass of Boosters launched by NASA (CRS)'
from SPACEXTBL
where customer = 'NASA (CRS)'
```

```
 * sqlite:///my_data1.db
Done.
```

Total Payload Mass of Boosters launched by NASA (CRS)
45596

Average Payload Mass by F9 v1.1

- Query and result are shown below.
- Average payload mass was calculated using the `AVG` aggregation function and constraining the result to `booster_version` as "F9 v1.1"
- Average payload mass is 2,928 kg (~3 tones)

```
%%sql
select avg(payload_mass__kg_) as 'Average Payload Mass - Booster Version F9 v1.1'
from SPACEXTBL
where booster_version = 'F9 v1.1'
```

```
 * sqlite:///my_data1.db
Done.
```

Average Payload Mass - Booster Version F9 v1.1
2928.4

First Successful Ground Landing Date

- Query and result are shown below
- Usage of `MIN` aggregation function to find the earliest date constraint with the \`landing_outcome\` as “Success (ground pad)”
- First successful ground pad landing occurred on December 22, 2015

```
%%sql
select min(Date) as 'Date of First Successful Landing in a Ground Pad'
from SPACEXTBL
where landing_outcome = 'Success (ground pad)'
```

```
 * sqlite:///my_data1.db
Done.
```

Date of First Successful Landing in a Ground Pad
2015-12-22

Successful Drone Ship Landing with Payload between 4000 and 6000

- Query and result are shown below
- Used the conditional function AND to capture the successful drone ship outcome and the payload mass constraint

```
%%sql
select booster_version from SPACEXTBL
where landing_outcome ='Success (drone ship)' and payload_mass__kg_ between 4000 and 6000
```

```
 * sqlite:///my_data1.db
Done.
```

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

Total Number of Successful and Failure Mission Outcomes

- Query and result are shown below
- Total success and fail mission outcomes are 61 and 10 respectively
- `COUNT` aggregate function was used in addition to constraining the `landing_outcome` with words that starts with "Success" and "Failure"

```
#Sucessful Missions
%sql select count(landing_outcome) as 'Number of Successful Mission' from SPACEXTBL where landing_outcome like 'Success%'
```

```
 * sqlite:///my_data1.db
Done.
```

Number of Successful Mission
61

```
#Failure Missions
%sql select count(landing_outcome) as 'Number of Failure Mission' from SPACEXTBL where landing_outcome like 'Failure%'
```

```
 * sqlite:///my_data1.db
Done.
```

Number of Failure Mission
10

Boosters Carried Maximum Payload

- Query and result are shown
- A *subquery* was used to determine the `booster_version` that has the maximum payload mass (e.g., MAX( ) aggregation command)

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

```
* sqlite:///my_data1.db
Done.
```

Booster_Version
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

- Used the `SUBSTR( )` command to extract specific characters in a string. For this case, it was used to extract the **month** portion of the `Date` column
- `WHERE` command was used to filter year of 2015 and landing outcome as “Failure (drone ship)”

```
%%sql
select substr(Date,6,2) as 'Month Number', landing_outcome, booster_version, launch_site
from SPACEXTBL
where substr(Date,1,4) = '2015' and landing_outcome like 'Failure (drone ship)'
```

```
 * sqlite:///my_data1.db
Done.
```

Month Number	Landing_Outcome	Booster_Version	Launch_Site
10	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

- Query and result are shown
- Used `GROUP BY` and `ORDER BY` functions as well as the `AND` conditional operator to sort and summarize the query

```
%%sql
select landing_outcome, count(landing_outcome)
from SPACEXTBL
where Date between '2010-06-04' and '2017-03-20'
group by landing_outcome
order by count(landing_outcome) desc
```

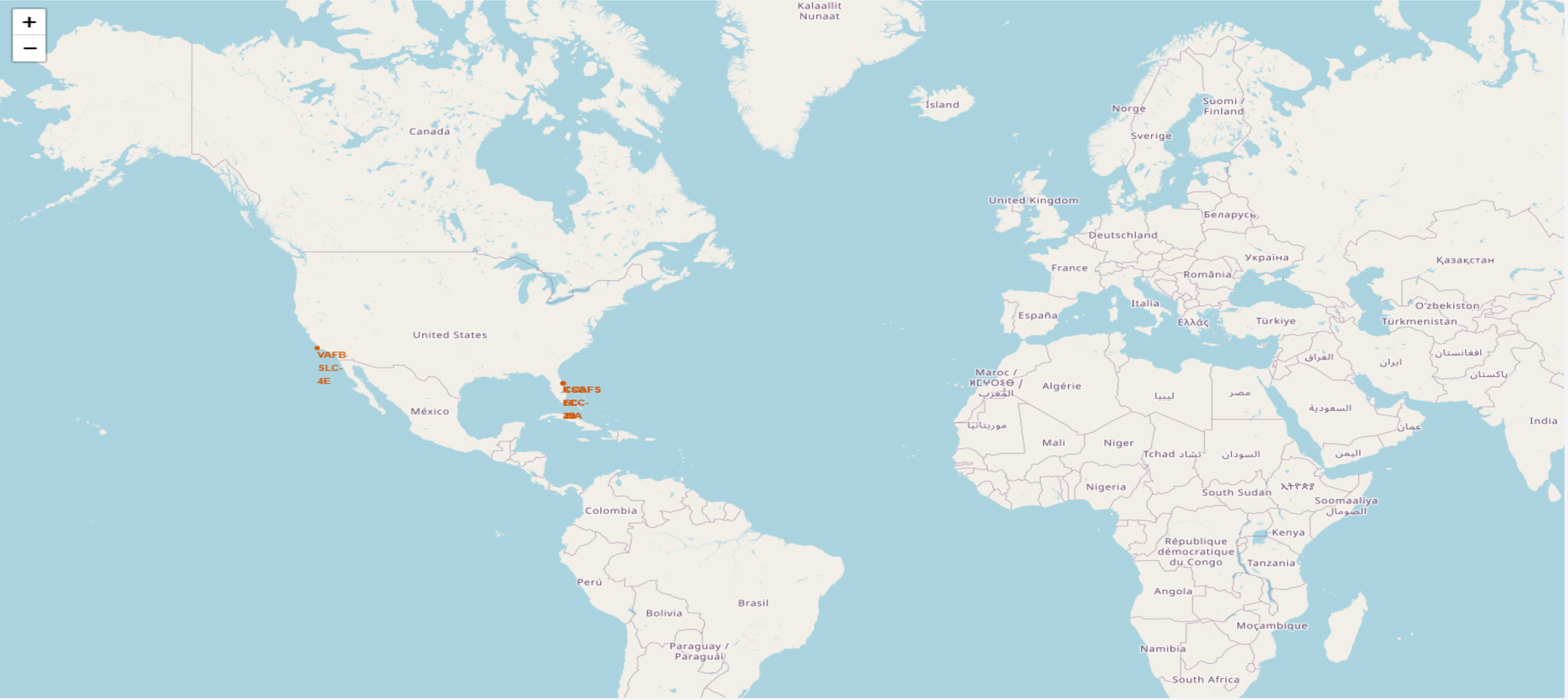
```
 * sqlite:///my_data1.db
Done.
```

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

Section 3

Launch Sites Proximities Analysis

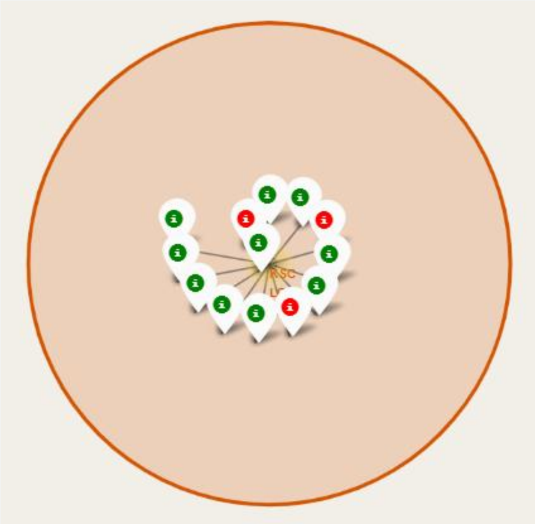
SpaceX Launch Sites – Global Map



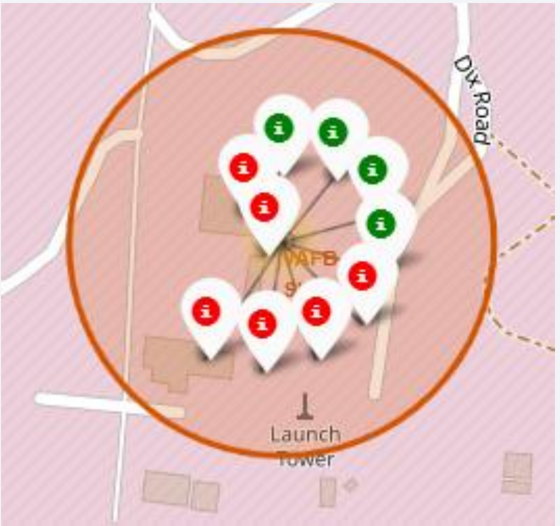
- Launch sites are located on east and west coast of USA (i.e., state of California (CA) and Florida (FL)) with only 1 launch site in CA and the rest are in FL

Success/Fail Launches

[FL] KSC LC-39A

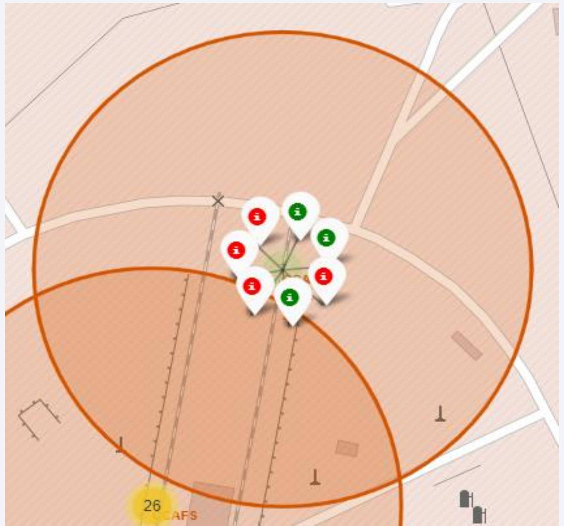


[CA] VAFB SLC-4E



[FL] CCAFS LC-40

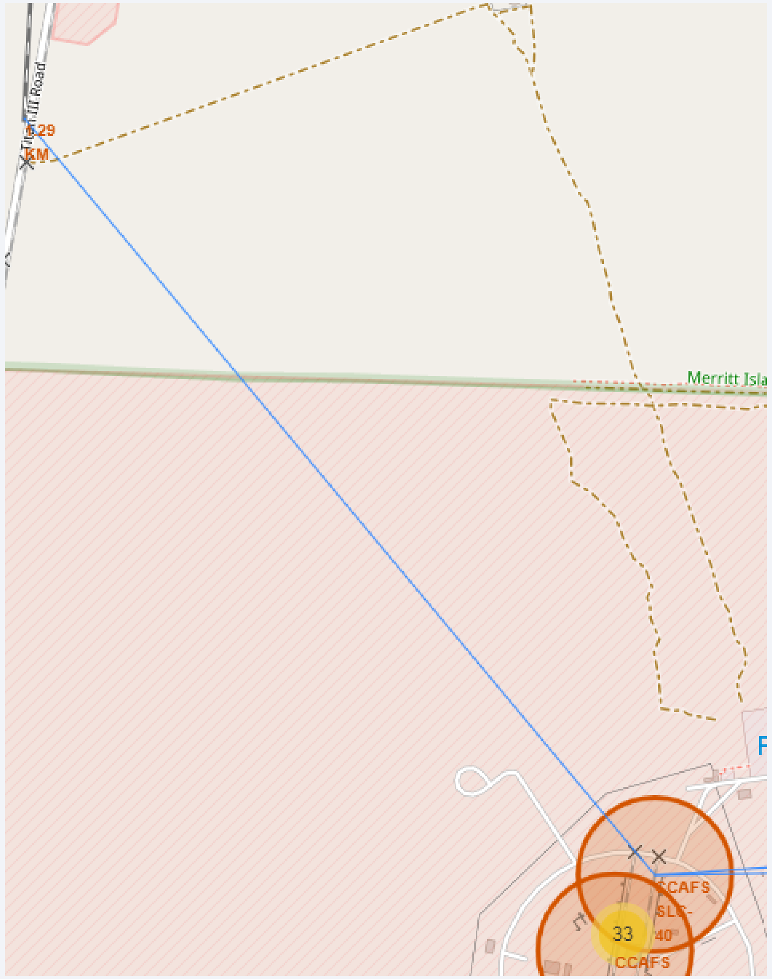
[FL] CCAFS SLC-40



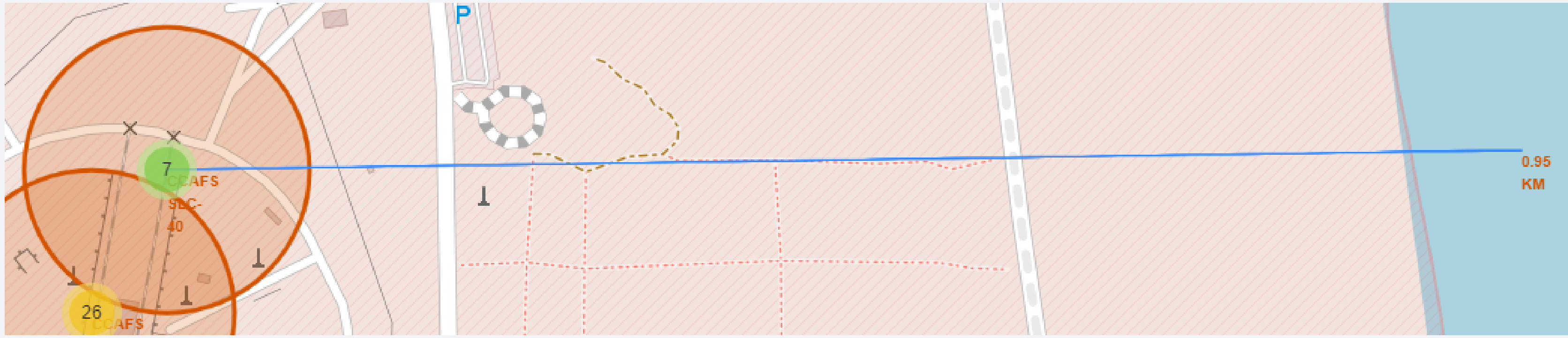
- For each launch site, **green markers** are designated as **<u>successful mission</u>** while red markers are designated as **<u>failure mission</u>**.

Launch Site Proximities

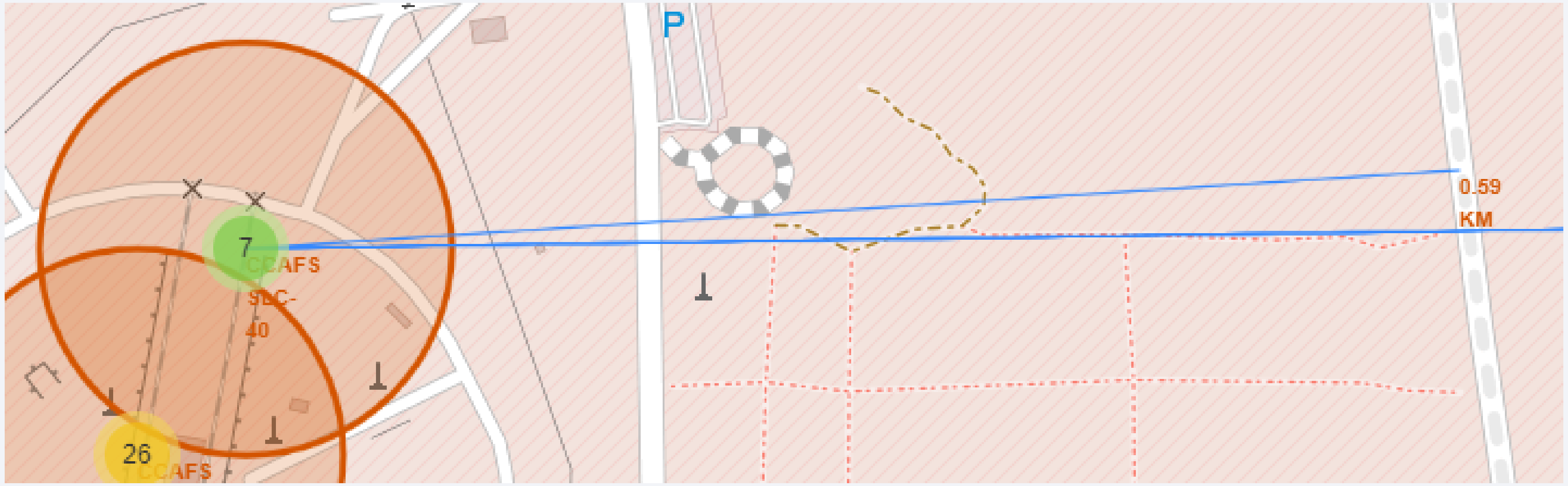
- It is important to determine the approximate distance from the launch site to key point of interest such as highways, coasts or City centers to determine any implications due any launch activities



Launch Site CCAFS SLC-40
Distance to nearest rail = 1.29 km



Launch Site CCAFS SLC-40
Distance to nearest coastline = 0.95 km

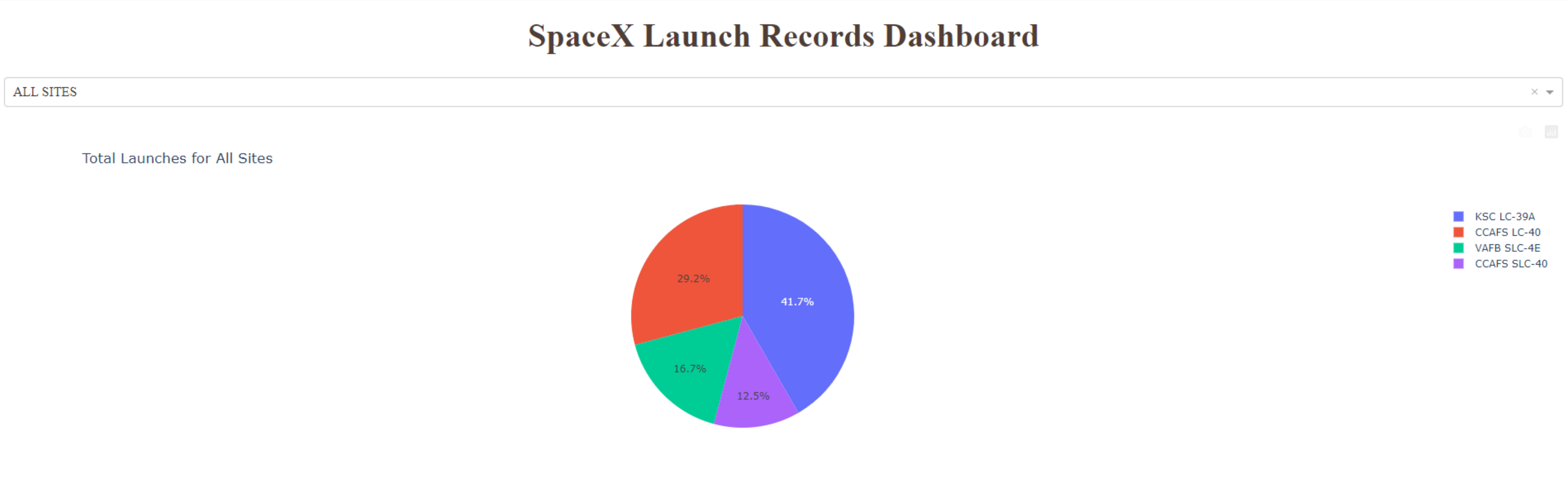


Launch Site CCAFS SLC-40
Distance to nearest highway = 0.59 km

Section 4

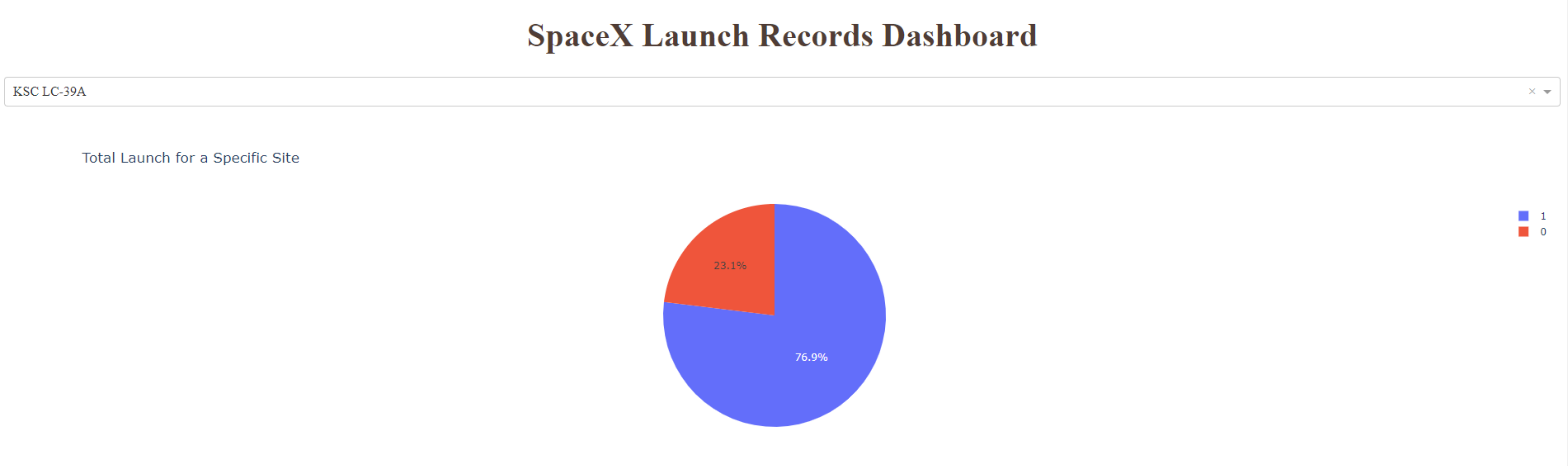
Build a Dashboard with Plotly Dash

Distribution of Launches per Launch Sites



- Of all the existing launch sites, **Site KSC LC – 39A** has the most rocket launches

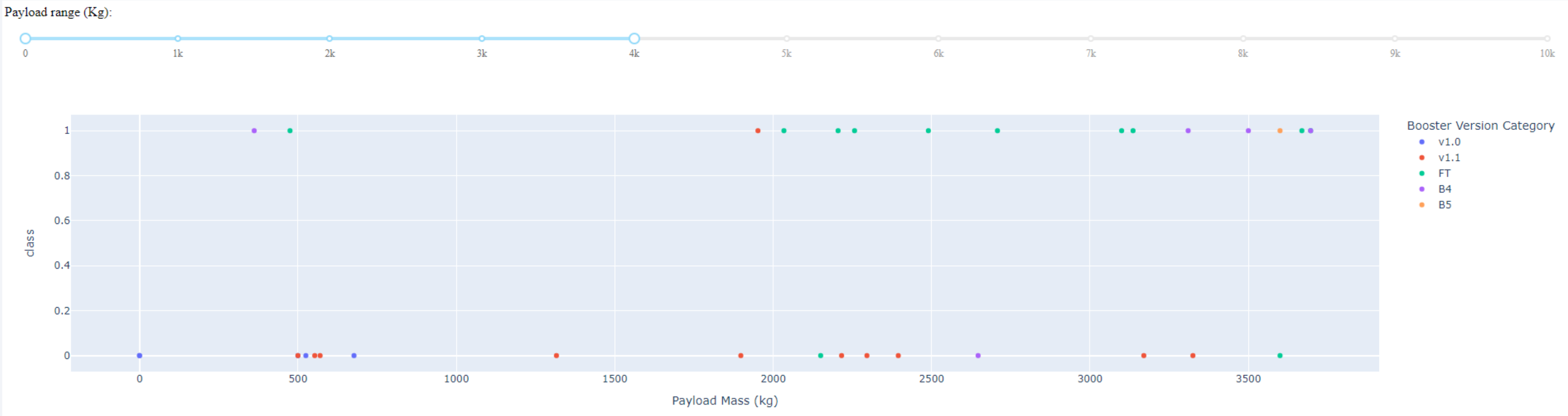
KSC LC-39A Success Rate



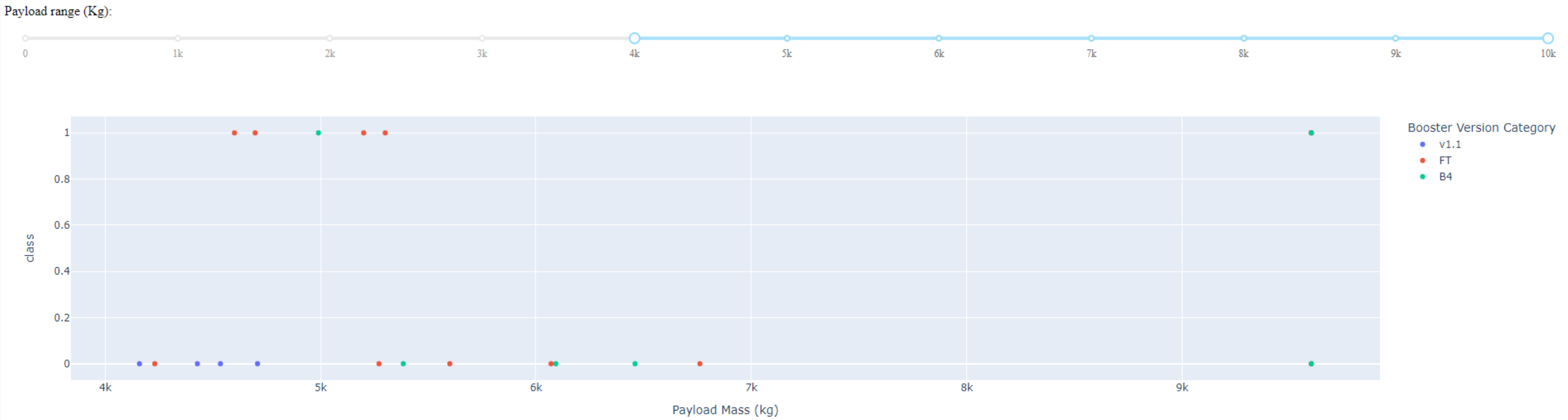
- Launch site KSC LC – 39A has about ~77% success rate in terms of landing the rocket’s first stage

Payload Range Scatter Plot

Low Payload
(0 to 4,000 kg)



Heavy Payload
(4,000 to 10,000kg)

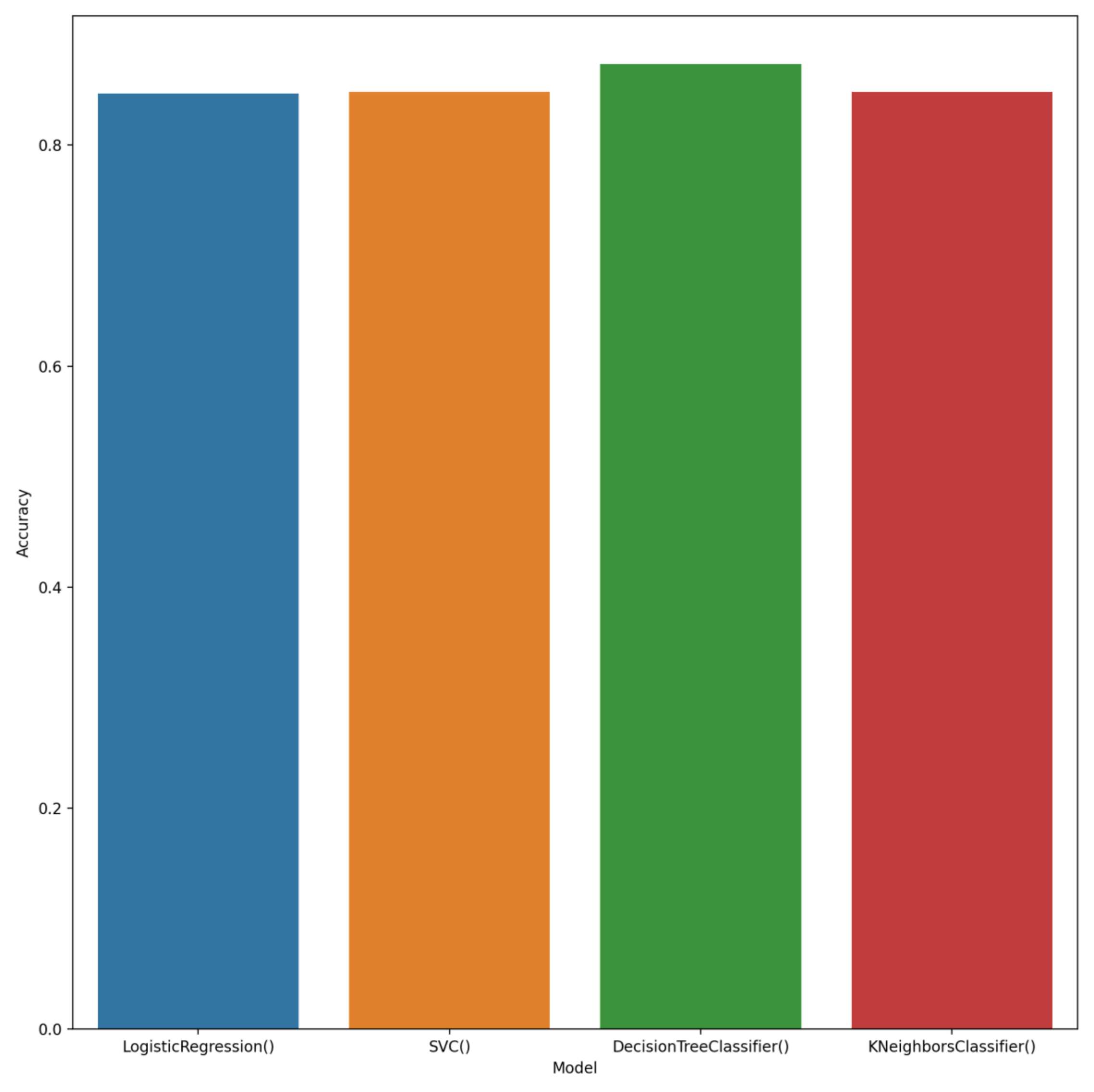


- When observing both Low and Heavy Payload weight, it can be inferred that lower payload weight can be attributed to a higher chance of landing success compared to heavier pay load

Section 5

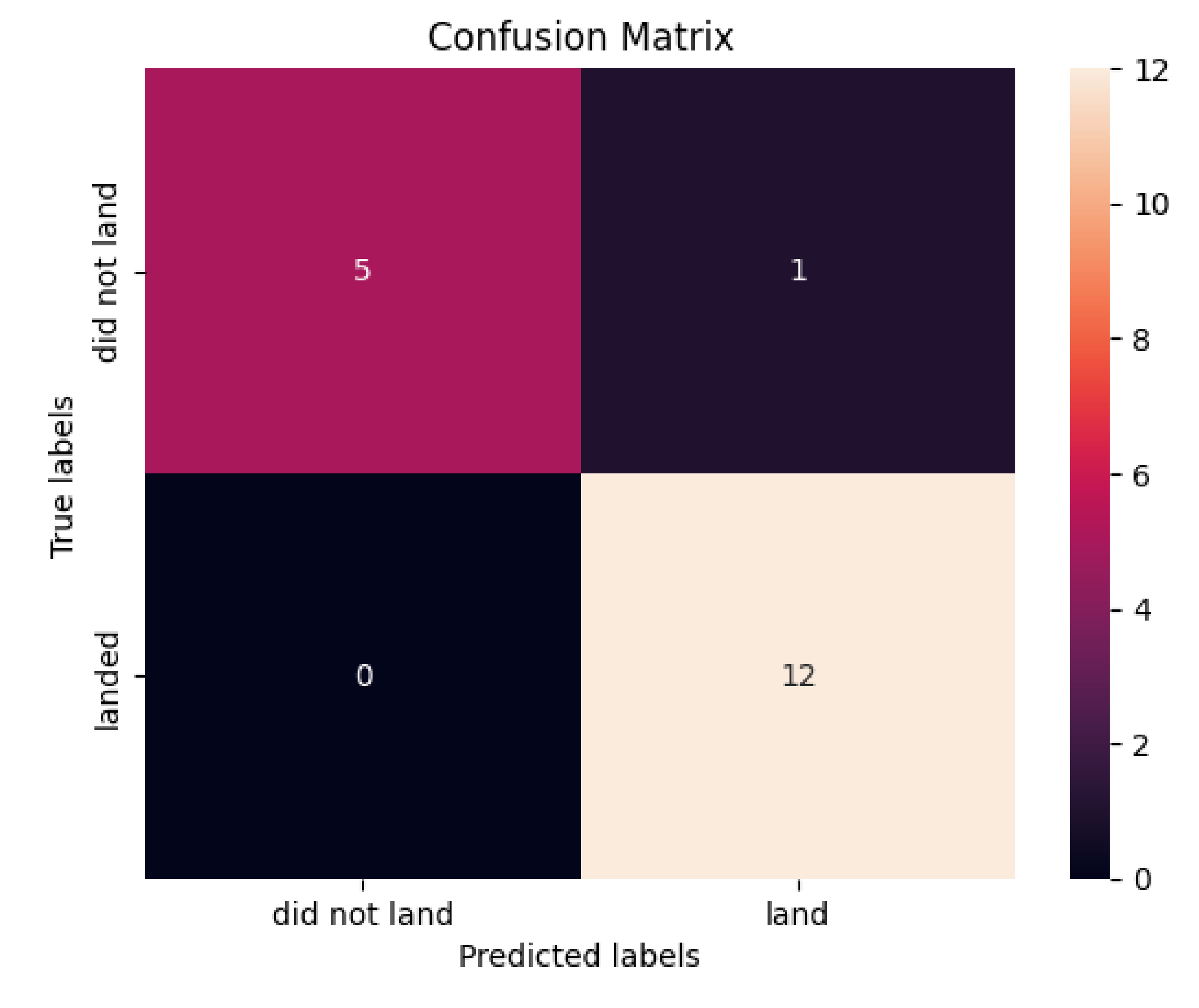
Predictive Analysis (Classification)

Classification Accuracy



- A bar chart summarizing the accuracy of each classification model is given where the **Decision Tree algorithm** is the most accurate (e.g., 87% accurate)

Confusion Matrix



- Shown is the confusion matrix for the most accurate classification model which is the Decision Tree which shows low false positive (i.e., true values indicating rocket **did not** land but predicted value yielded that the rocket **did** land)

Conclusions

- Greater number fights = greater success rate
- Orbits that have the highest success rate: ES-L1, GEO, HEO, SSO
- Success rate is likely for orbits that accommodate for heavier payloads such as LEO, ISS, and Polar
- Generally, average landing success rate had been progressively increasing through the year with a major dip on the year 2018
- Of all the launch sites investigated, Site KDC LC-39 has the highest launch success rate
- Lastly, Decision Tree algorithm is the most accurate classification algorithm (87% accurate) in predicting whether the first stage of SpaceX Falcon 9 rocket will successfully land

Thank you!