IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Rocío Rego Sierra
March 2023

Outline

Executive Summary

- Summary of methodologies
  - Data Collection using web scraping and SpaceX API
  - Data Collection with Web Scraping
  - Data Wrangling
  - Exploratory Data Analysis with SQL
  - Exploratory Data Analysis with Data Visualization
  - Interactive Visual Analytics with Folium
  - Machine Learning Prediction

- Summary of all results
  - Exploratory Data Analysis results
  - EDA allowed to identify which features are the best to predict success of launchings
  - Machine Learning Prediction showed the best model to predict the important characteristics, using all collected data

Introduction

- Project background and context

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

- Problems you want to find answers
  - What factors will make the rocket land successfully?
  - The effect of each relationship of rocket variables on the outcome
  - What conditions does SpaceX have to achieve to get the best results and ensure the best rocket success landing rate

Section 1

Methodology

Methodology

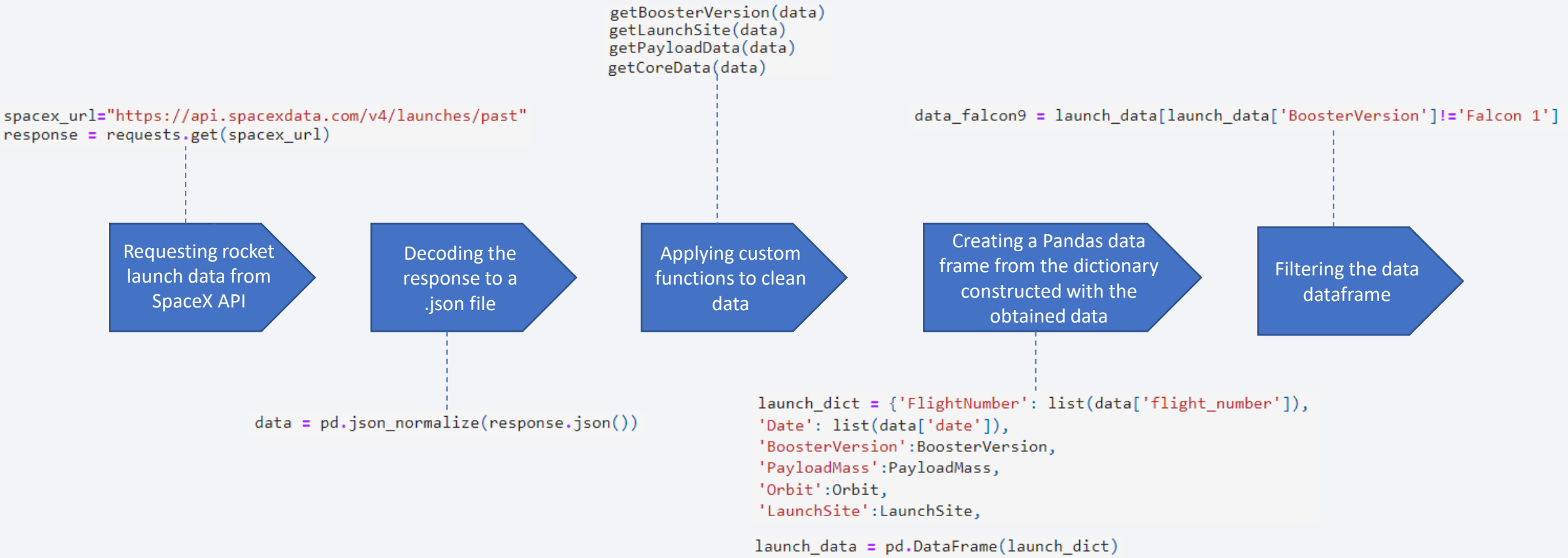
Executive Summary

- Data collection methodology:
  - Data was collected using SpaceX API
  - Also, performing web scraping from a Wikipedia
- Perform data wrangling
  - Data was processed applying one-hot encoding data fields for Machine Learning and dropping irrelevant columns
- Perform exploratory data analysis (EDA) using visualization and SQL
  - Scatter and bar graphs to show patterns between data
- Perform interactive visual analytics using Folium and Plotly Dash
  - Using Folium and Plotly Dash Visualizations
- Perform predictive analysis using classification models
  - Build and evaluate classification models

Data Collection

- How data sets were collected?
  - The data collection process was done by making GET requests to the SpaceX API
  - Similarly, another way to get the data is WebScraping Wikipedia using BeautifulSoup

Data Collection – SpaceX API



Github notebook URL

Data Collection - Scraping

```
static_url = "https://en.wikipedia.org/w/index.php?title=List_of_Falcon_9_and_Falcon_Heavy_launches&oldid=1027686922"
data  = requests.get(static_url).text
```

Requesting the HTML page from the URL

Creating BeautifulSoup object from the HTML response

```
soup = BeautifulSoup(data)
```

```
html_tables=soup.find_all("table")
html_tables
```

Finding all tables on the wiki page

Extracting column name one by one

```
ths = first_launch_table.find_all('th')
for th in ths:
    name = extract_column_from_header(th)
    if name is not None and len(name) > 0:
        column_names.append(name)
```

```
launch_dict= dict.fromkeys(column_names)
```

Creating a data frame by parsing the launch HTML tables

Github notebook URL

Data Wrangling

```
df['LaunchSite'].value_counts()
```

Determining the number of launches on each site

Determining the number and occurrence of each orbit in the column 'Orbit'

```
df['Orbit'].value_counts()
```

```
GTO     27
ISS     21
VLEO    14
PO       9
LEO      7
SSO      5
MEO      3
ES-L1    1
HEO      1
SO       1
GEO      1
```

```
landing_outcomes = df['Outcome'].value_counts()
for i,outcome in enumerate(landing_outcomes.keys()):
    print(i,outcome)
```

```
0 True ASDS
1 None None
2 True RTLS
3 False ASDS
4 True Ocean
5 False Ocean
6 None ASDS
7 False RTLS
```

Calculating the number and occurence of mission outcome per orbit type

Creating a landing outcome label from Outcome column

```
df['Class']=landing_class
df[['Class']].head(8)
```

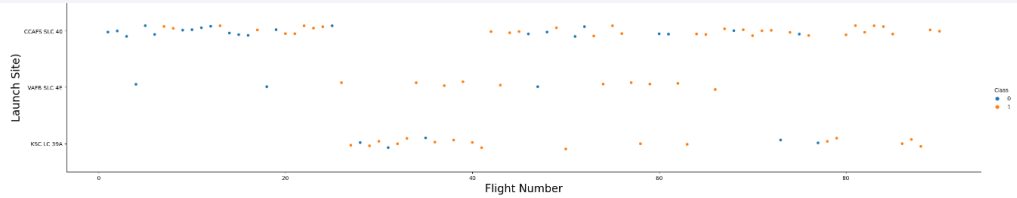
[Github notebook URL]

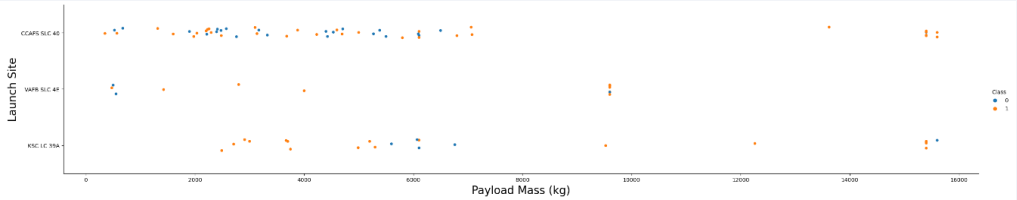
EDA with Data Visualization

Scatter Graphs Drawn

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

Scatter plots show dependency of attributes on each other. Once a pattern is determined from the graphs it's easy to predict factors that will lead to the maximum probability of success in both outcome and landing.



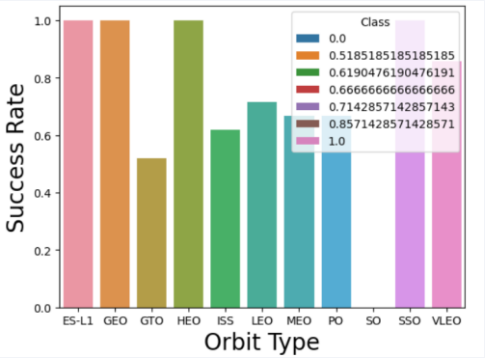


Github notebook URL

Bar Graphs Drawn

- Success Rate vs. Orbit Type

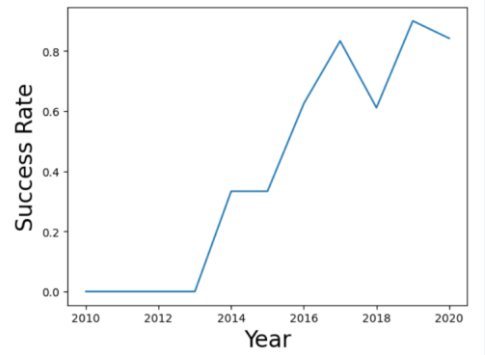
Bar graphs make easier the task of interpreting the relationship between attributes. Using these graphs it is easy to determine which orbits have a higher probability of success.



Line Graphs Drawn

- Launch Success Yearly Trend

Line graphs are very useful as they clearly show a certain trend and help in forecasting.



EDA with SQL

SQL queries performed:

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

Github notebook URL

Build an Interactive Map with Folium

Map objects created and added to a folium map:

- Map Marker: create a simple stock Leaflet marker on the map, with optional popup text or Vincent visualization
- Icon Marker: creates an Icon object that will be rendered using Leaflet.awesome-markers
- Circle Marker: class for drawing circle overlays on a map
- PolyLine: draw polyline overlays on a map
- Marker Cluster Object: provides beautiful animated Marker Clustering functionality for maps

Github notebook URL

Build a Dashboard with Plotly Dash

Plots/graphs and interactions added to a dashboard:

- Dash and its components: provide all of the available HTML tags as user-friendly Python classes
- Pandas: fetch values from CSV and creates a dataframe
- Plotly: plot the graphs with interactive Plotly library
- Dropdown: create a dropdown for launch sites
- Rangeslider: create a rangeslider for Payload Mass range selection
- Pie Chart: create the pie graph for Success Percentage display
- Scatter Chart: create the scatter graph for Correlation display

Github notebook URL

Predictive Analysis (Classification)

Model Building

- Load the dataframe
- Standardize the data
- Split the data into training and test datasets
- Check the number of samples
- Decide on the machine learning algorithm to use
- Set the parameters and algorithms to GridSearchCV
- Fit the datasets into the GridSearchCV objects and train the dataset

Model Evaluation

- Check the accuracy for each model
- Get tuned hyperparameters for each type of algorithms
- Plot Confusion Matrix

Model Improving

- Feature Engineering
- Algorithm Tuning

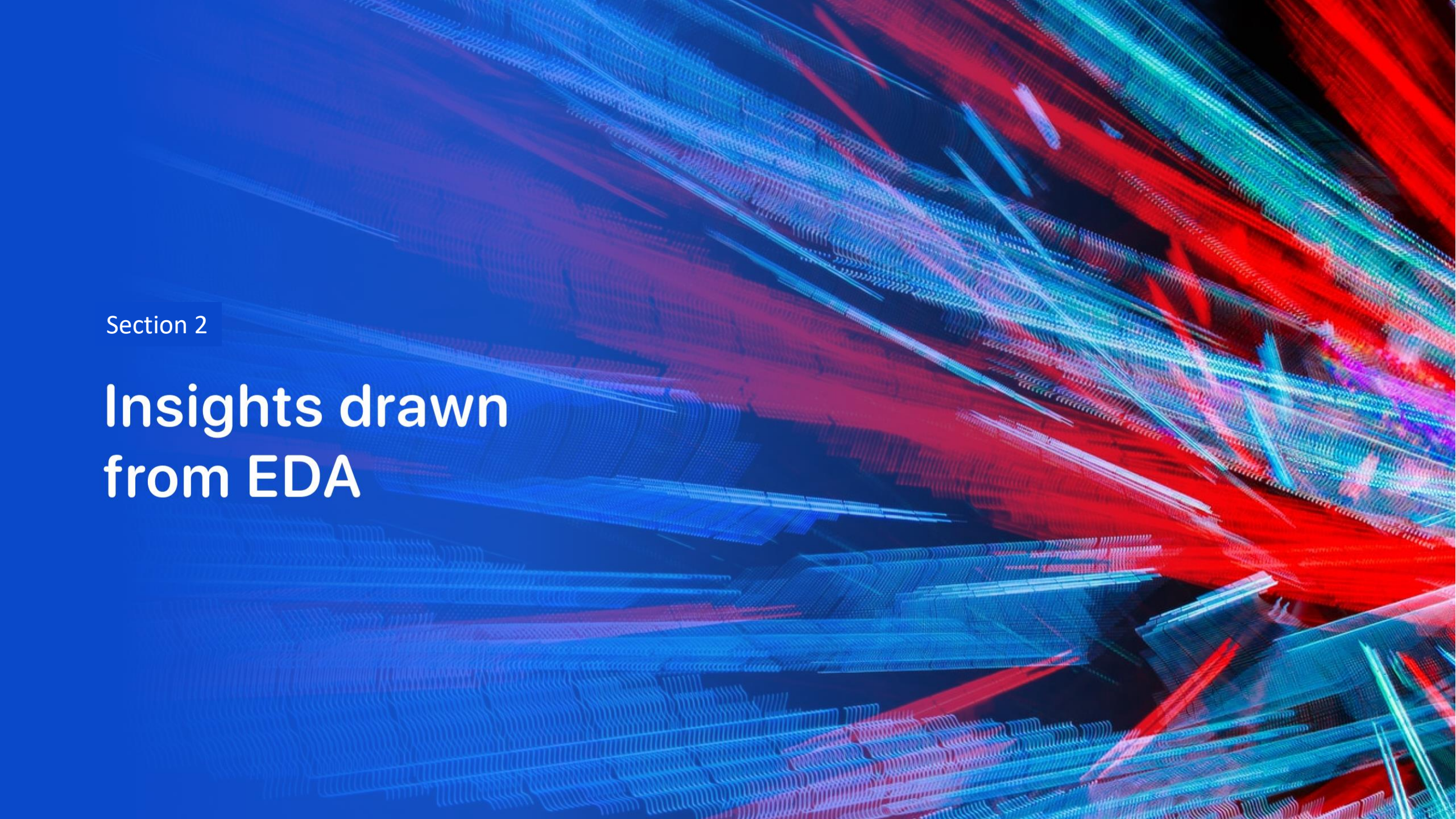
Search of the best performing classification model

The best performing model is chosen from among the models with the best accuracy score

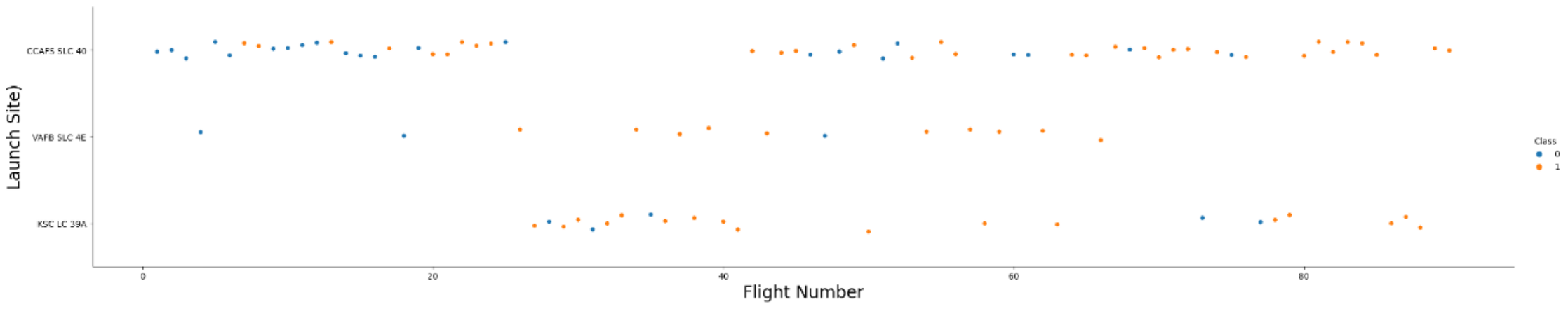
Github notebook URL

Results

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

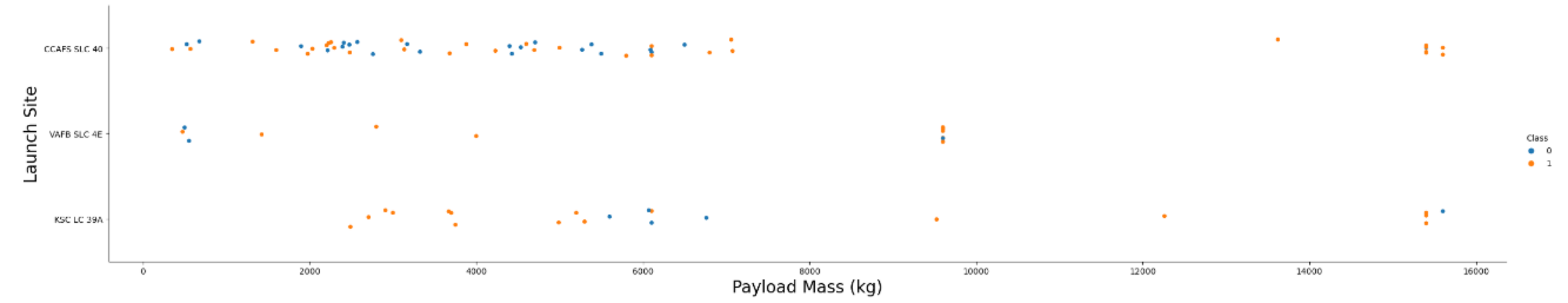

Section 2
Insights drawn from EDA

Flight Number vs. Launch Site



- According to the plot, the best launch site is CCAF5 SLC 40, where most of recent launches were successful.
- The larger the flight amount at a launch site, the greater the success rate at a launch site.

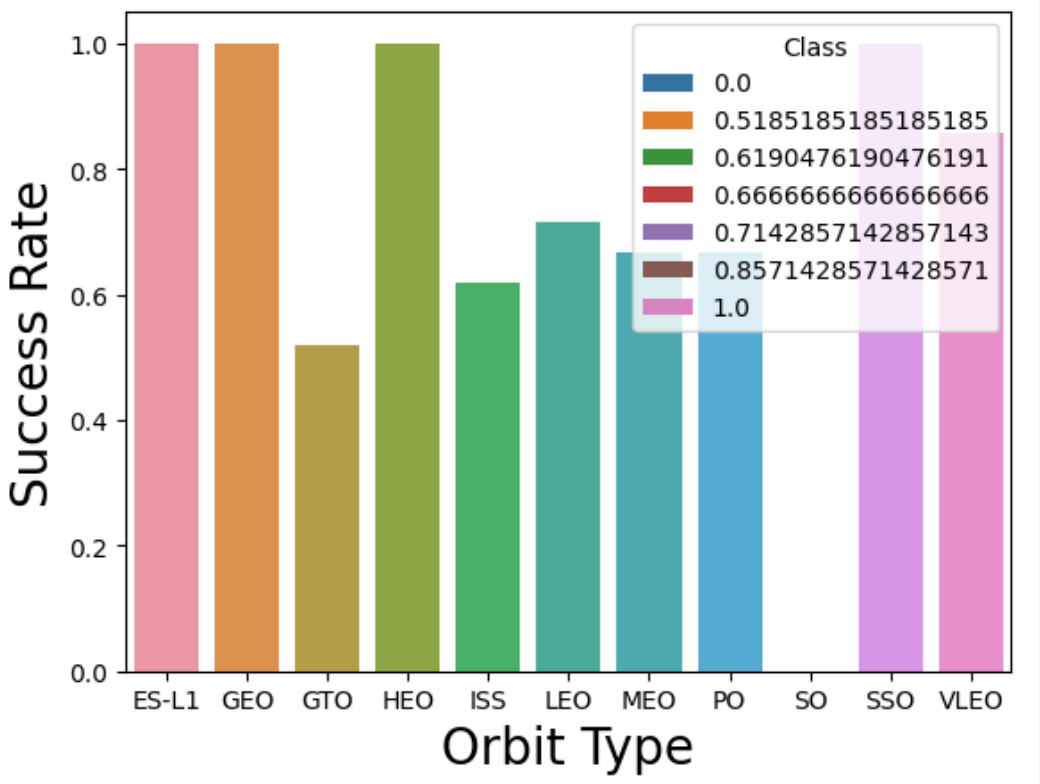
Payload vs. Launch Site



The greater the payload mass (greater tan 7000 kg), the higher success rae for the rocket.

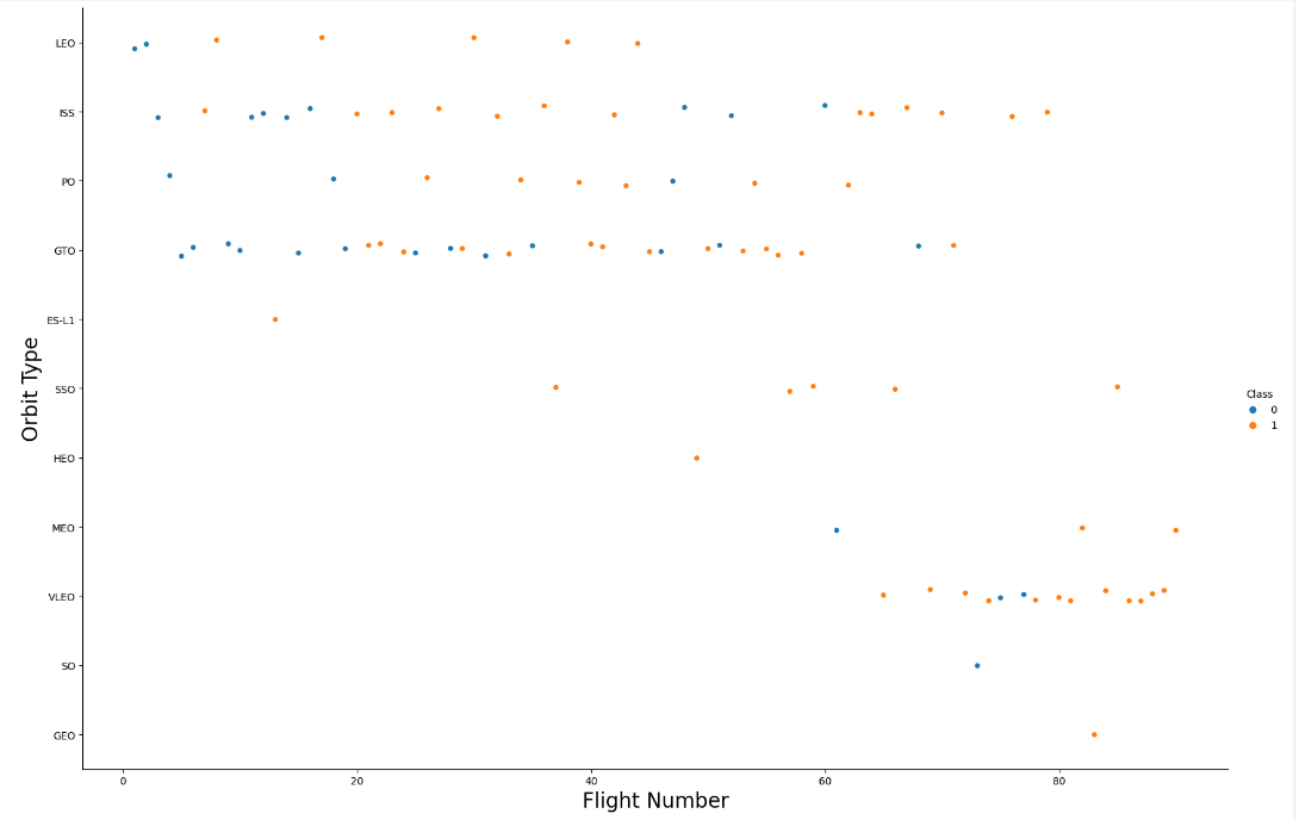
Success Rate vs. Orbit Type

GEO, HEO, SSO, ES-L1 orbits has the best success rate



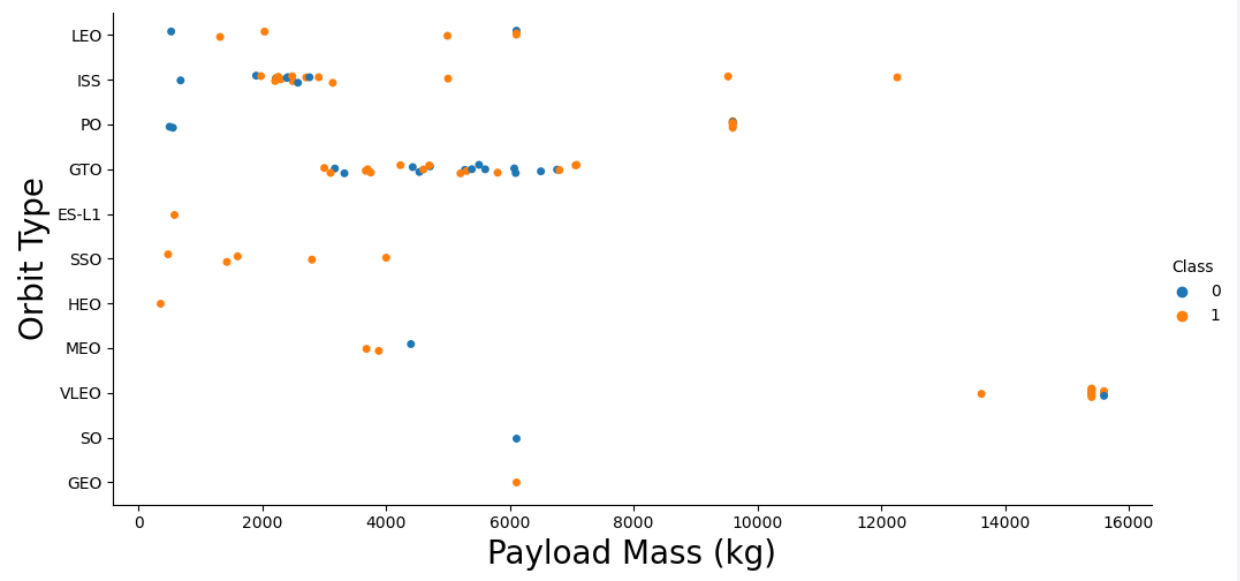
Flight Number vs. Orbit Type

- In the case of LEO orbit, it can be observed that the success rate increases at the same time as the number of flights.
- In all other cases, there is no relationship between successes and the number of flights



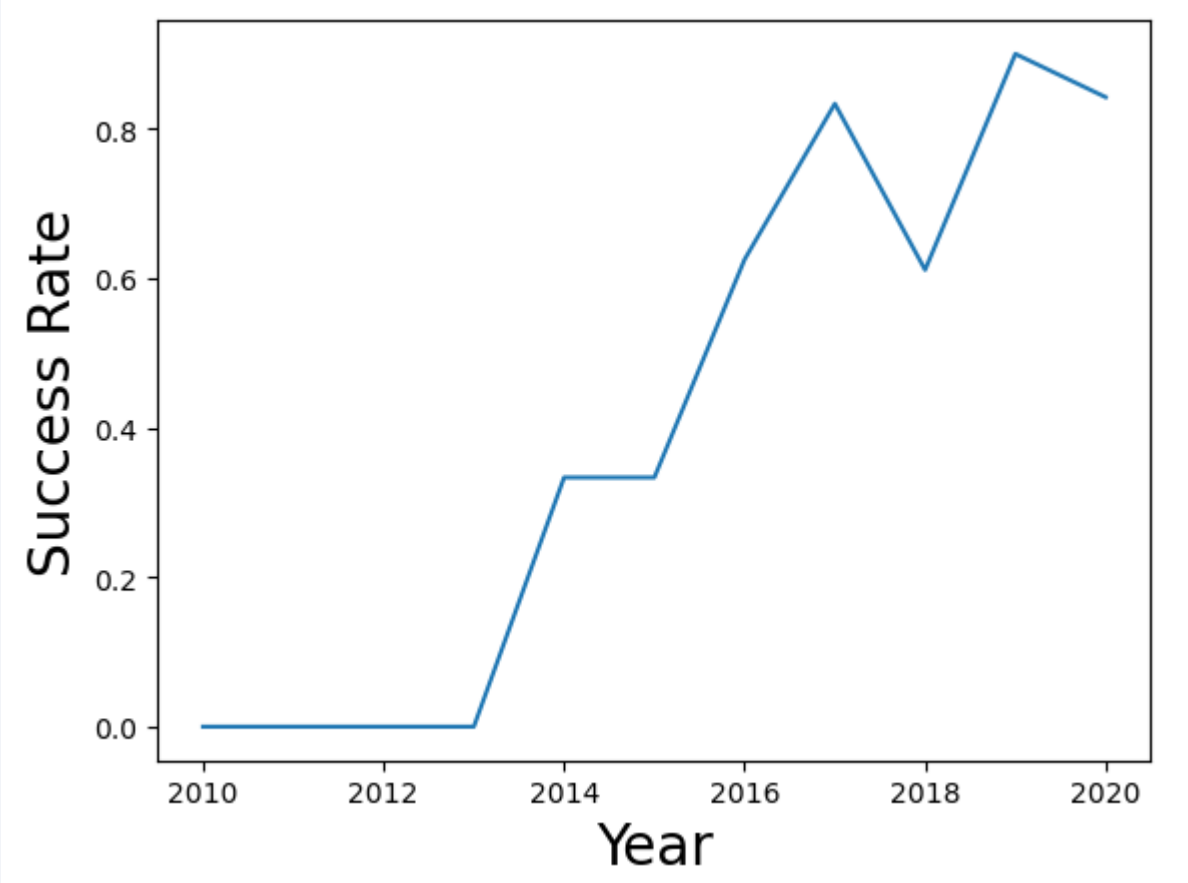
Payload vs. Orbit Type

It can be observed that the higher the payload, the higher the landing success in PO, LEO and ISS orbits



Launch Success Yearly Trend

It is clear that the success rate rises sharply from 2013 onwards, then dips in 2018 and continues to rise thereafter.



All Launch Site Names

Names of the unique launch sites:

```
%sql select launch_site, count(*) as launches_per_site from spacex group by launch_site
```

launch_site	launches_per_site
CCAFS LC-40	26
CCAFS SLC-40	34
KSC LC-39A	25
VAFB SLC-4E	16

The names are obtained by grouping the launch_site using **group by** and counting the occurrences of each with **count**

Launch Site Names Begin with 'CCA'

5 records where launch sites begin with ‘CCA’:

```
%sql select * from spacex 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

All rows are filtered to get only those where launch_site starts with 'CCA' using the **like 'CCA%'** condition. Then, to get only 5 entries, **limit 5** is used.

Total Payload Mass

Total payload carried by boosters from NASA:

```
%sql select sum(payload_mass__kg_) as total_payload_mass_kg from spacex where customer = 'NASA (CRS)'
```

total_payload_mass_kg
45596

Using the **sum()** operator, sum the payload_mass__kg of all entries where the customer matches the desired, using the condition **where customer = 'NASA (CRS)'**

Average Payload Mass by F9 v1.1

Average payload mass carried by booster version F9 v1.1:

```
%sql select avg(payload_mass__kg_) as payload_mass_kg_avg from spacex where booster_version = 'F9 v1.1'
```

payload_mass_kg_avg
2928

Using the **avg()** operator, the average payload_mass__kg of all entries where the booster_version matches what is desired is calculated, using the condition **where booster_version = 'F9 v1.1'**

First Successful Ground Landing Date

Dates of the first successful landing outcome on ground pad:

```
%sql select DATE from spacex where landing__outcome like '%ground pad%' order by DATE asc limit 1
```

DATE
2015-12-22

It is obtained by filtering the entries with the condition **where landing__outcome like '%ground pad%'**, in order to keep only the desired rows. Then, they are sorted using **order by date asc**. Finally, only the oldest date is retained by using **limit 1**

Successful Drone Ship Landing with Payload between 4000 and 6000

Names of boosters which have successfully landed on drone ship and had payload mass greater than 4000 but less than 6000:

```
%sql select booster_version from spacex where landing__outcome like '%Success (drone ship)%' and payload_mass__kg_ between 400 and 6000
```

booster_version
F9 FT B1021.1
F9 FT B1022
F9 FT B1023.1
F9 FT B1026
F9 FT B1021.2
F9 FT B1029.2
F9 FT B1038.1
F9 FT B1031.2
F9 B4 B1042.1
F9 B5 B1046.1

Se filtran las entradas usando la condición **where landing__outcome like '%Success (drone ship)%'** y **payload_mass__kg_ between 400 and 6000**

Total Number of Successful and Failure Mission Outcomes

Total number of successful and failure mission outcomes:

```
%sql select mission_outcome, count(*) as num_mission_outcome from spacex group by mission_outcome
```

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

The entries are grouped according to the mission_outcome column using **group by mission_outcome**, and then the number of entries corresponding to each of the values obtained from mission_outcome is counted, using the **count(*)** operator

Boosters Carried Maximum Payload

Names of the booster which have carried the maximum payload mass:

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

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

Entries are filtered based on the payload_mass__kg_ column, using the condition **where payload_mass__kg_ = (select max(payload_mass__kg_) from spacex)**. The value to satisfy the condition is a subquery in which the **max()** operator is used to calculate the maximum value of payload_mass__kg_ in the database

2015 Launch Records

Failed landing_outcomes in drone ship, their booster versions, and launch site names for in year 2015:

```
%sql select DATE, landing__outcome, booster_version, launch_site from spacex where landing__outcome like '%Failure (drone ship)%' and DATE like '2015%
```

DATE	landing__outcome	booster_version	launch_site
2015-01-10	Failure (drone ship)	F9 v1.1 B1012	CCAFS LC-40
2015-04-14	Failure (drone ship)	F9 v1.1 B1015	CCAFS LC-40

date, landing__outcome, booster_version and launch_site are displayed for entries that meet the condition **where landing__outcome like '%Failure (drone ship)%' and DATE like '2015%'**

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

Ranking of 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:

```
%sql select landing__outcome, count(*) as landing_outcome_num from spacex group by landing__outcome order by landing_outcome_num desc
```

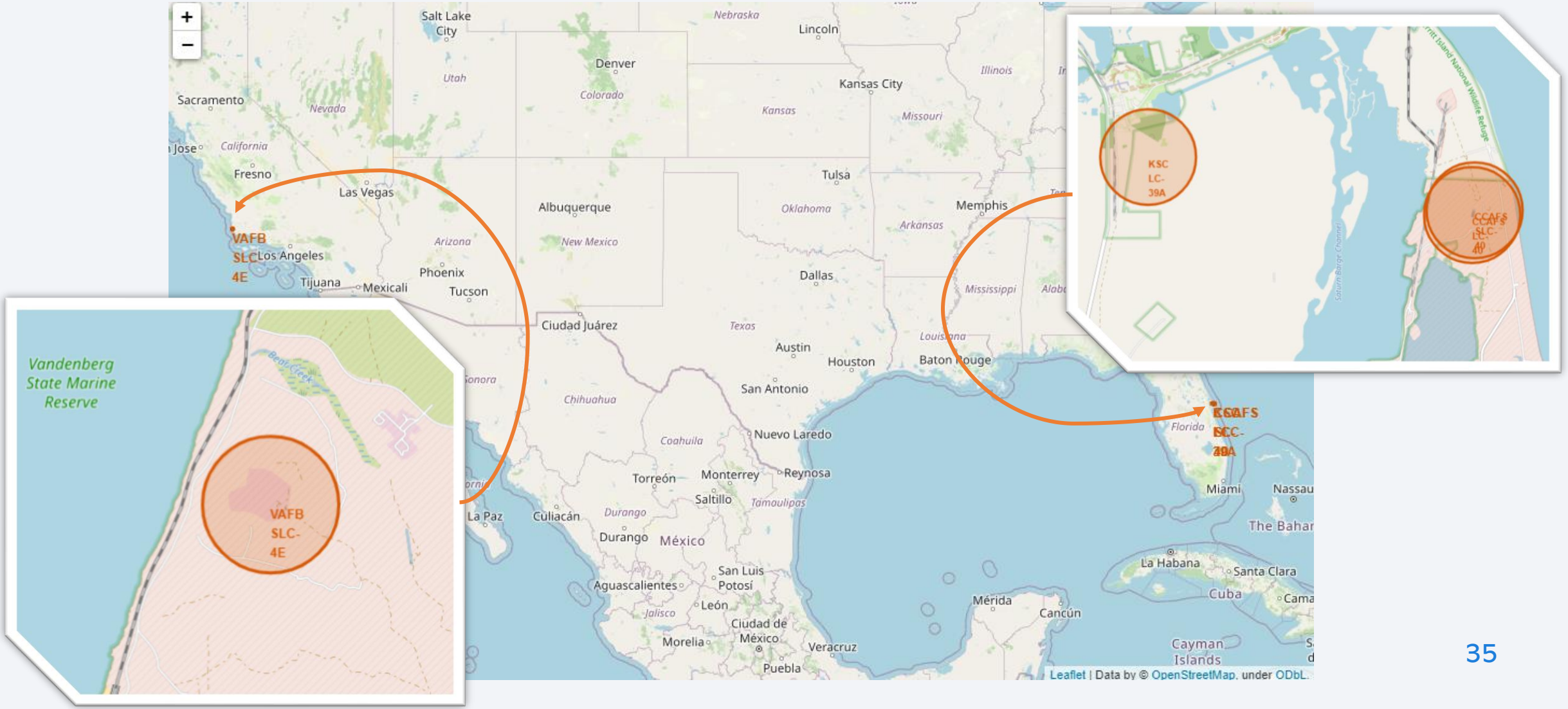
landing__outcome	landing_outcome_num
Success	38
No attempt	22
Success (drone ship)	14
Success (ground pad)	9
Controlled (ocean)	5
Failure (drone ship)	5
Failure	3
Failure (parachute)	2
Uncontrolled (ocean)	2
Precluded (drone ship)	1

landing__outcome are grouped using **group by landing__outcome** and the number of entries in each landing__outcome is counted using the **count(*)** operator. To undertake the ranking, the data obtained is sorted from highest to lowest using **order by landing_outcome_num desc**

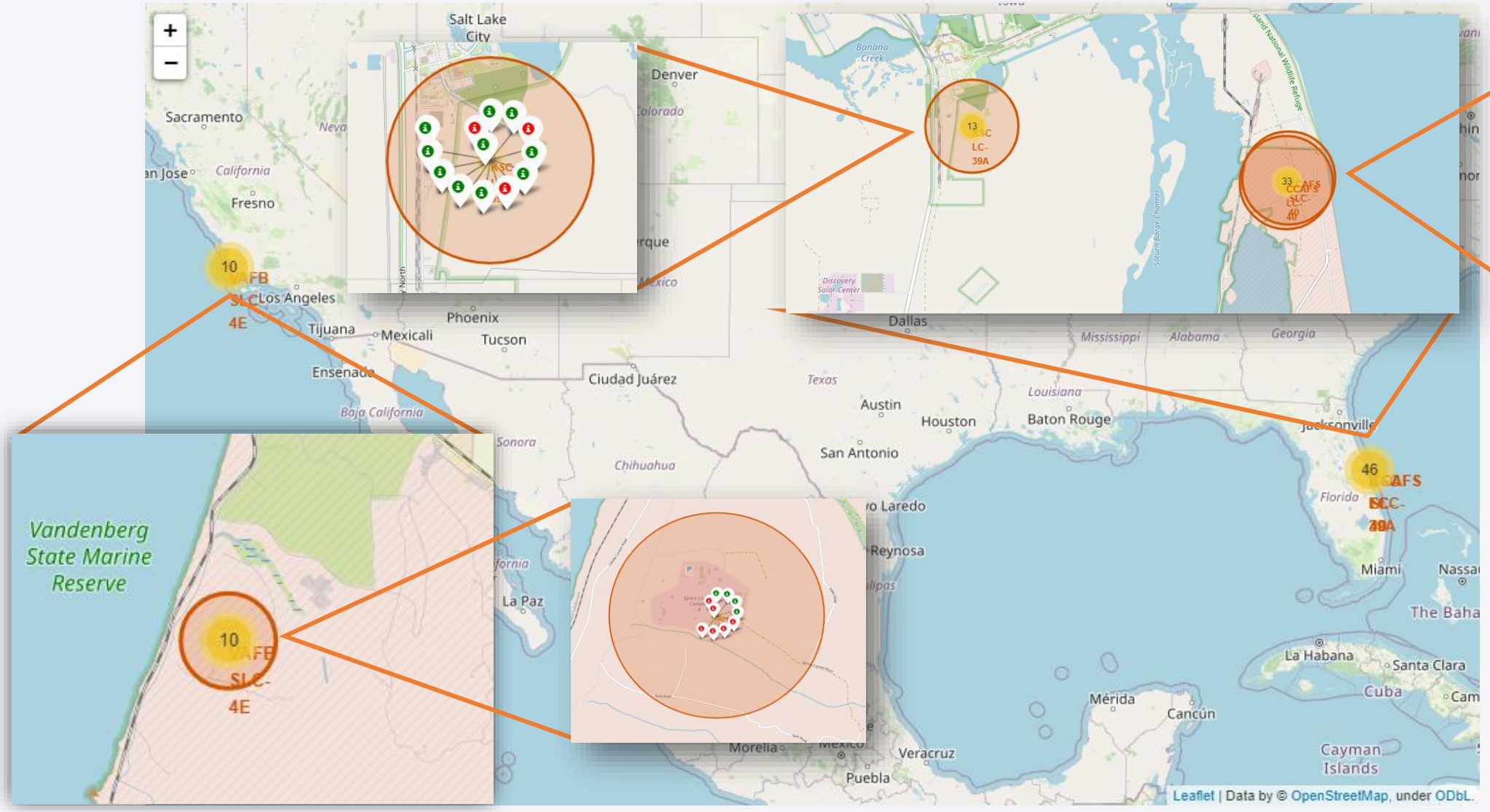
Section 3

Launch Sites Proximities Analysis

All launch sites location markers on a global map



Color-labeled launch outcomes on the map

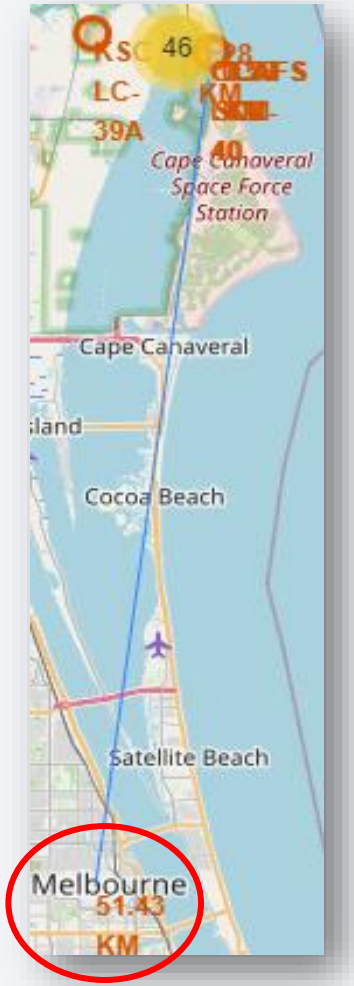


Successful launches

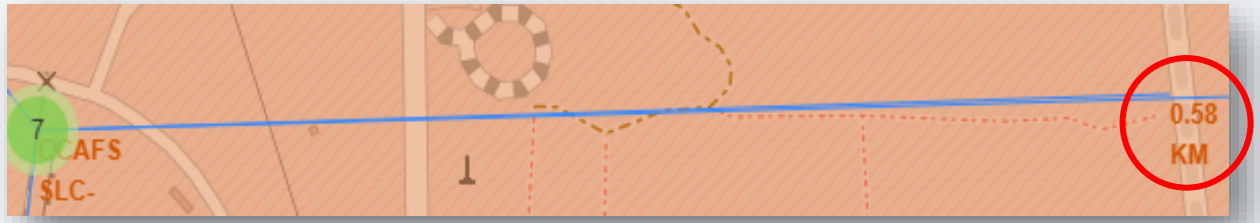
Failed launches

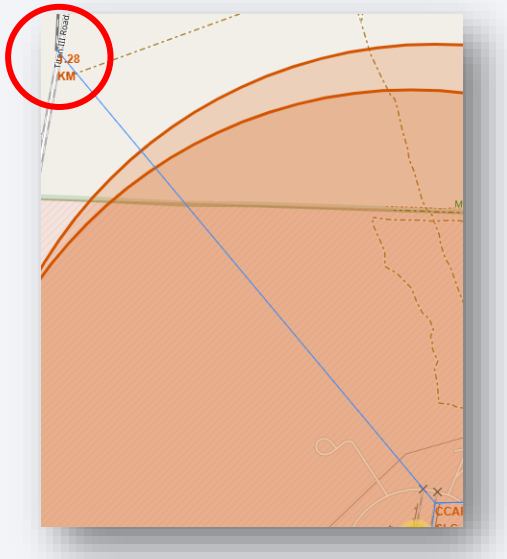
Launch site distances from railway, highway & coastline

Closest city



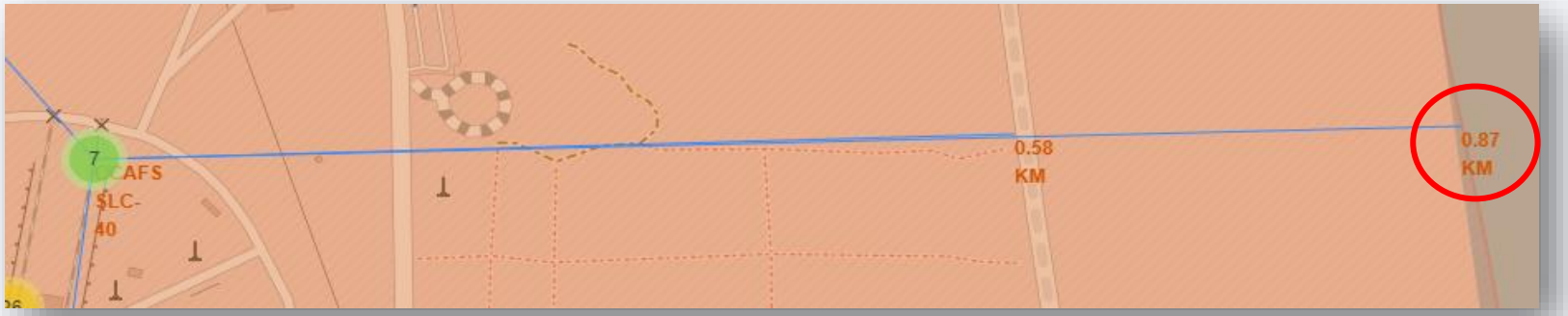
Closest highway

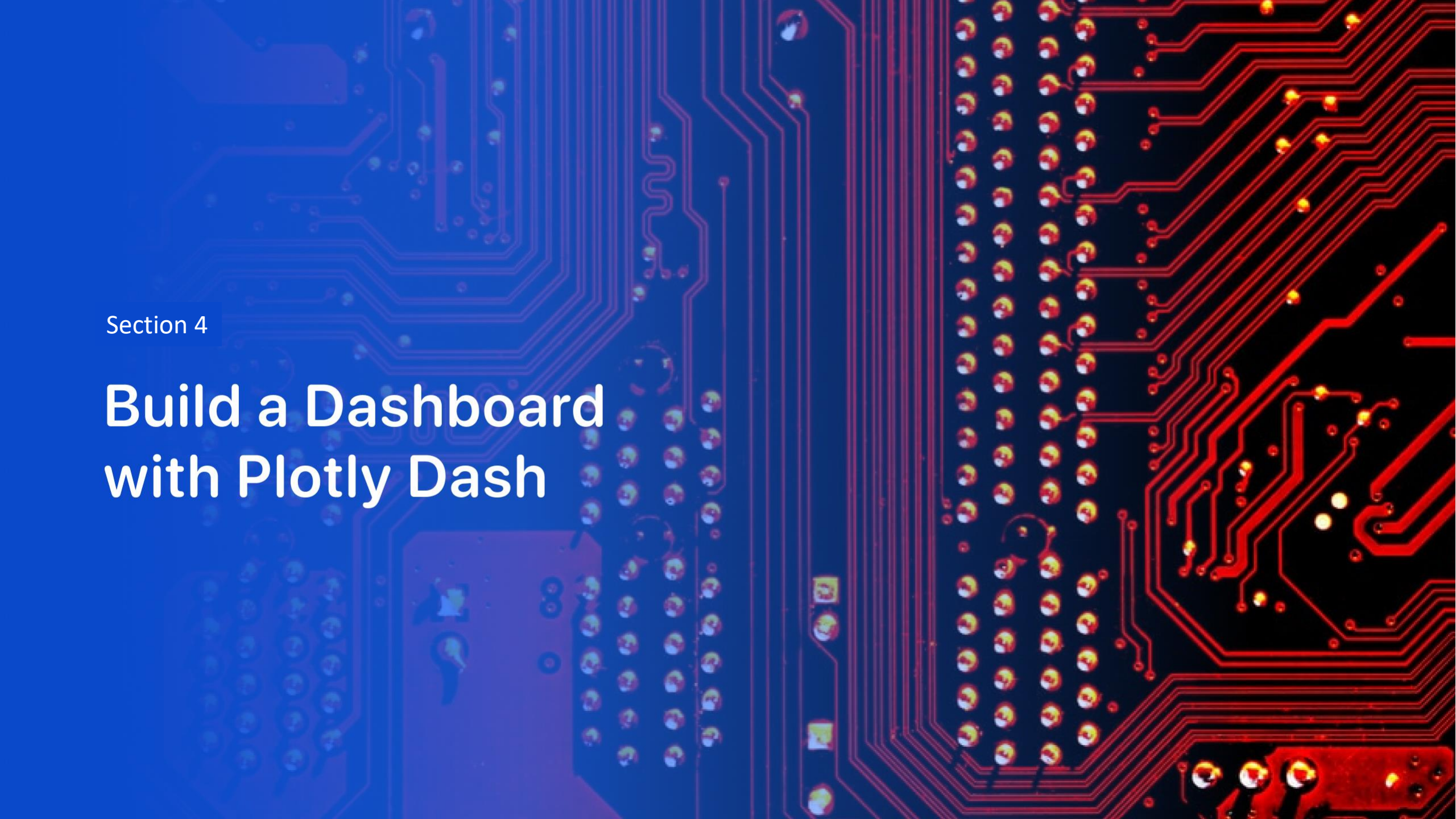




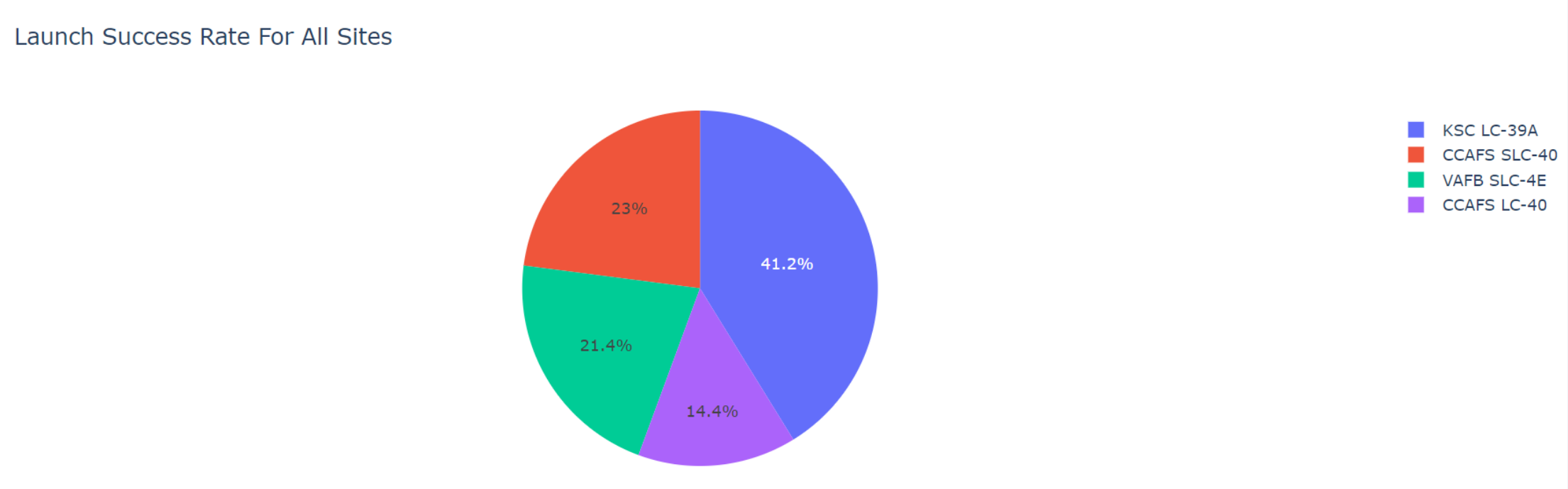
Closest railway

Closest coastline



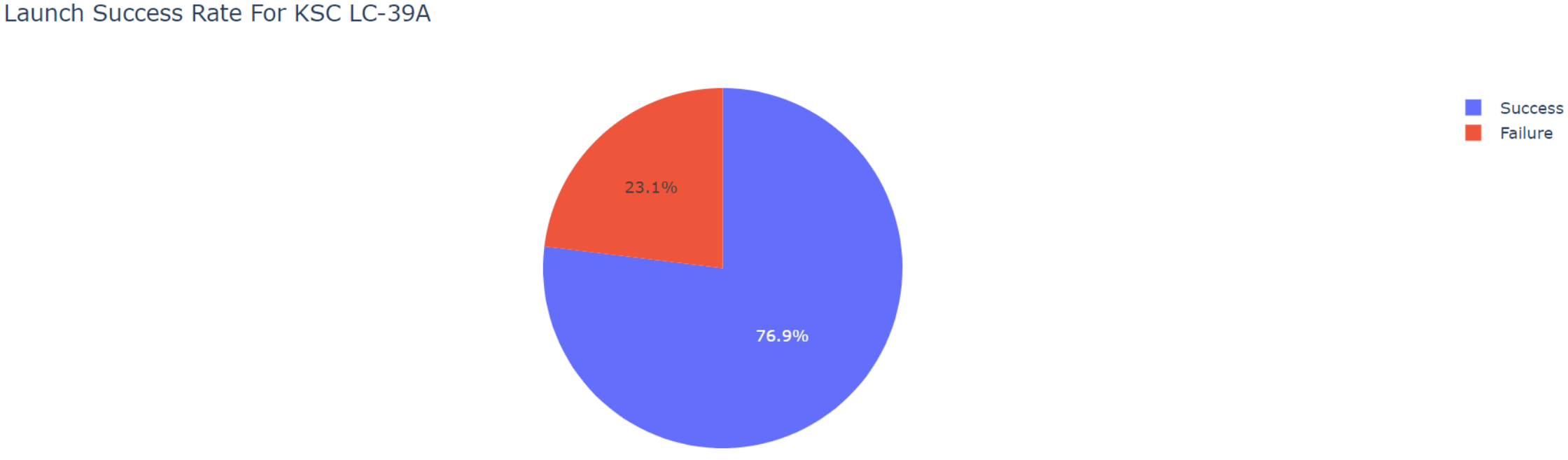

Section 4
Build a Dashboard with Plotly Dash

Launch success count for all sites



KSC LC-39A has the highest percentage of successful launches from all sites

Launch site with highest launch success ratio

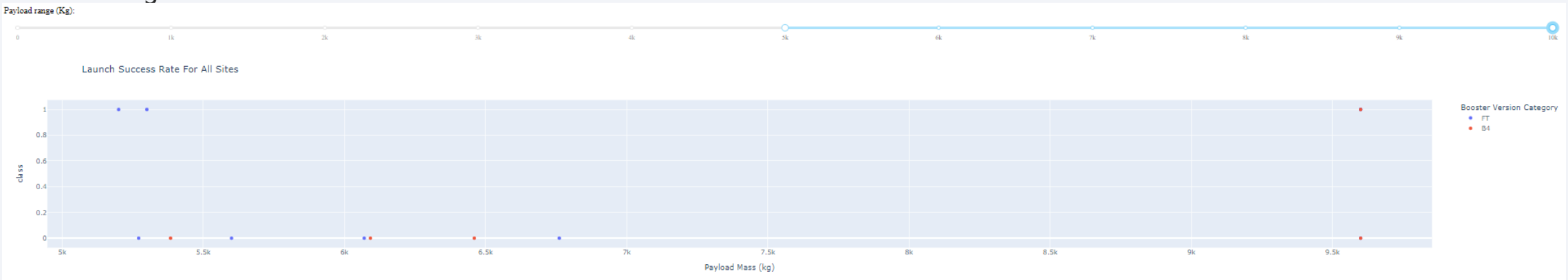


KSC LC-39A achieves a 76.9% success rate while getting a 23.1% failure rate

Payload vs. Launch Outcome scatter plot for all sites

0k - 4k range

4k - 10k range



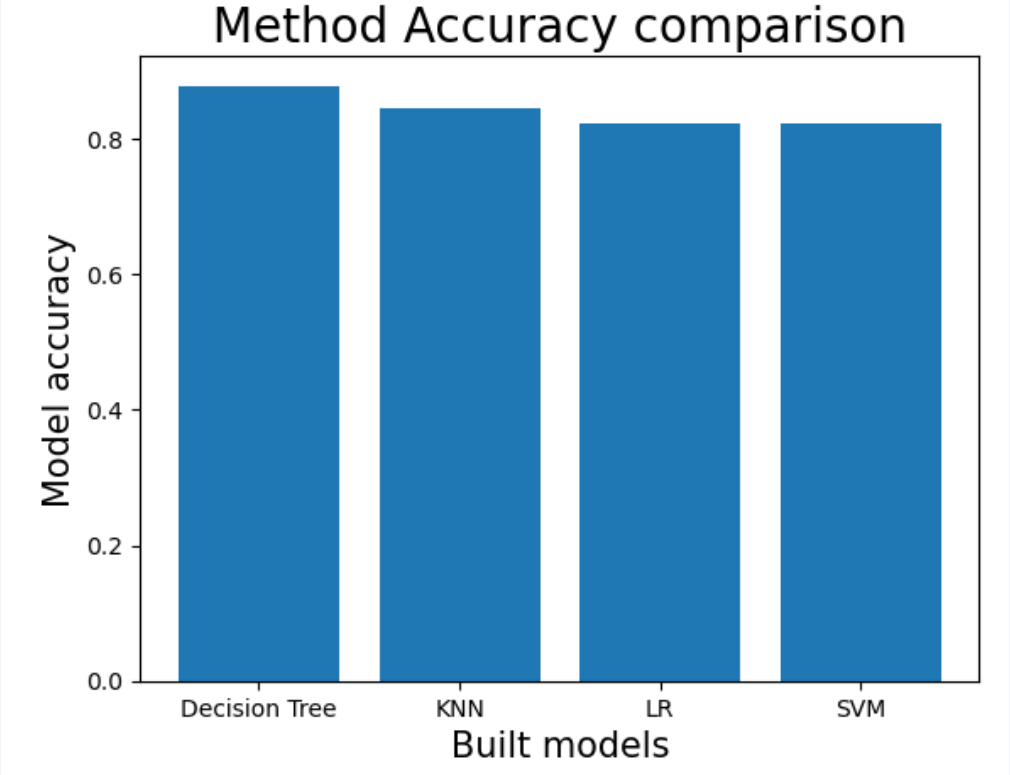
Section 5

Predictive Analysis (Classification)

Classification Accuracy

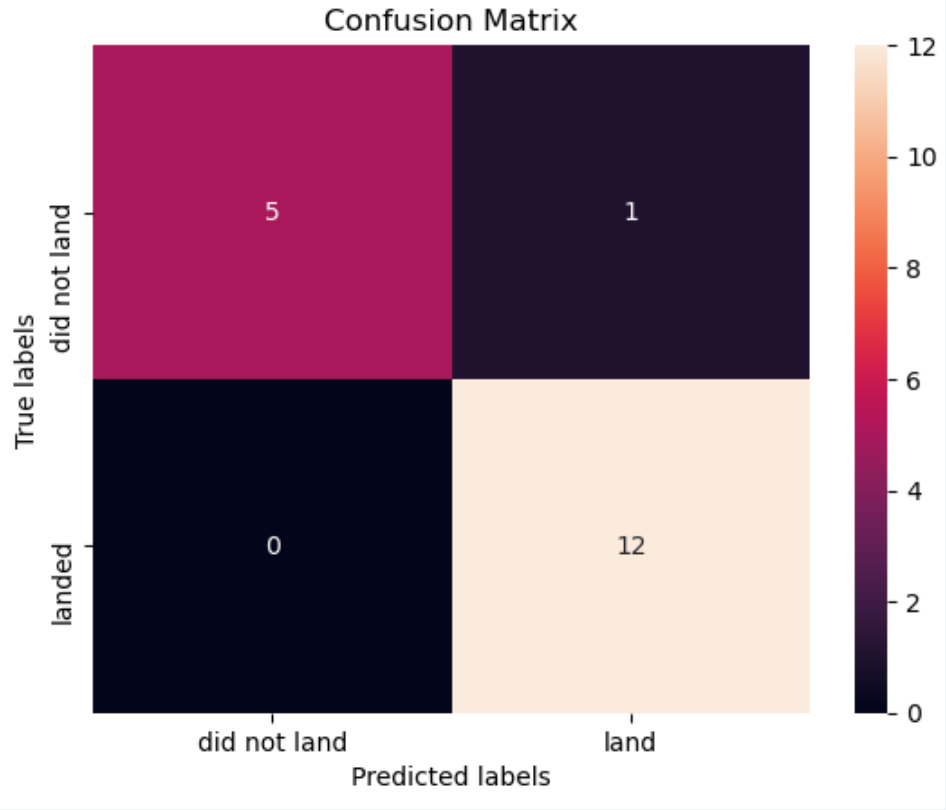
- In the right margin, a bar chart with the different models built and their corresponding precision value is shown

- The model with the highest classification accuracy is the Decision Tree Classificator with an accuracy of 87.78%



Confusion Matrix

In the right margin, the confusion matrix of the Decision Tree model is shown



Conclusions

It can be concluded that:

- The success rates for SpaceX launches is directly proportional time in years they will eventually perfect the launches
- It can be seen that KSC LC-39A had the most successful launches from all the sites
- Low weighted payloads perform better than the heavier payloads
- Orbit GEO, HEO, SSO, ES-L1 has the best Success Rate
- The Decision tree classifier is the best machine learning algorithm for this task

Thank you!