IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Vishal Murmu
22-03-2025

Outline

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

Executive Summary

- 4 Models were incorporated to determine the best model based on GridSearchCV
- Decision Tree model ad the highest accuracy

Introduction

- SpaceX has gained worldwide attention for a series of historic milestones.
- It is the only private company ever to return a spacecraft from low-earth orbit, which it first accomplished in December 2010. SpaceX advertises Falcon 9 rocket launches on its website with a cost of 62 million dollars whereas other providers cost upward of 165 million dollars each, much of the savings is because Space X 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.
- By performing the analysis, we will be able to determine the parameters (factors) which maximizes the ‘success’ outcomes of the landing

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - Publicly available Dataset
- Perform data wrangling
  - Data filtered to only contain Falcon 9 launches
  - Launch outcome was reclassified as binary variable (0 – Failure, 1- Success)
- 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

- Use API calls, launch datasets were retrieved and segregated
- Booster Version had both Falcon 1 and Falcon 9 data. Thus, data was filtered to contain only Falcon 9 launches
- *PayLoadMass* variable had 5 missing values thus it was replaced with the mean of the remaining Payload values using mean() and replace() functions
- Webscraping was done to retrieve launch outcomes
- Launch Outcome was finally converted into binary values (where, 0 is Failure and 1 is Success)

Data Collection – SpaceX API

GETTING LAUNCH IDs

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

APPENDING BOOSTER DATA

response = requests.get("https://api.spacexdata.com/v4/rockets/"+str(x)).json()

APPENDING LAUNCH SITE DATA

response = requests.get("https://api.spacexdata.com/v4/launchpads/"+str(x)).json()

Longitude.append(response['longitude'])

Latitude.append(response['latitude'])

LaunchSite.append(response['name'])

APPENDING PAYLOAD DATA

response = requests.get("https://api.spacexdata.com/v4/payloads/"+load).json()

APPENDING CORE DATA

response = requests.get("https://api.spacexdata.com/v4/cores/"+core['core']).json()

Outcome.append(str(core['landing_success'])+' '+str(core['landing_type']))

Flights.append(core['flight'])

GridFins.append(core['gridfins'])

Reused.append(core['reused'])

Legs.append(core['legs'])

LandingPad.append(core['landpad'])

Github Link: Data Collection

Data Collection - Scraping

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

API Calling Wikipedia page

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

Creating soup object

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

Retrieving Table and Column Names

Looping through the soup object and appending data

```
for table_number,table in enumerate(soup.find_all('table',"wikitable plainrowheaders collapsible")):
   # get table row
    for rows in table.find_all("tr"):
        #check to see if first table heading is as number corresponding to launch a number
        if rows.th:
            if rows.th.string:
                flight_number=rows.th.string.strip()
                flag=flight_number.isdigit()
```

Github Link: [Web Scraping]

Data Wrangling

Identifying datatypes of data variables

```
df.dtypes
```

```
FlightNumber        int64
Date               object
BoosterVersion     object
PayloadMass       float64
Orbit              object
LaunchSite         object
Outcome            object
Flights             int64
GridFins             bool
Reused               bool
Legs                 bool
LandingPad         object
Block             float64
ReusedCount         int64
Serial             object
Longitude         float64
Latitude          float64
dtype: object
```

Identifying number of facilities

```
# Apply value_counts() on column LaunchSite
df["LaunchSite"].value_counts()
```

```
LaunchSite
CCAFS SLC 40    55
KSC LC 39A      22
VAFB SLC 4E     13
Name: count, dtype: int64
```

Converting target variable to binary values

```
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
```

```
bad_outcomes=set(landing_outcomes.keys()[[1,3,5,6,7]])
bad_outcomes
```

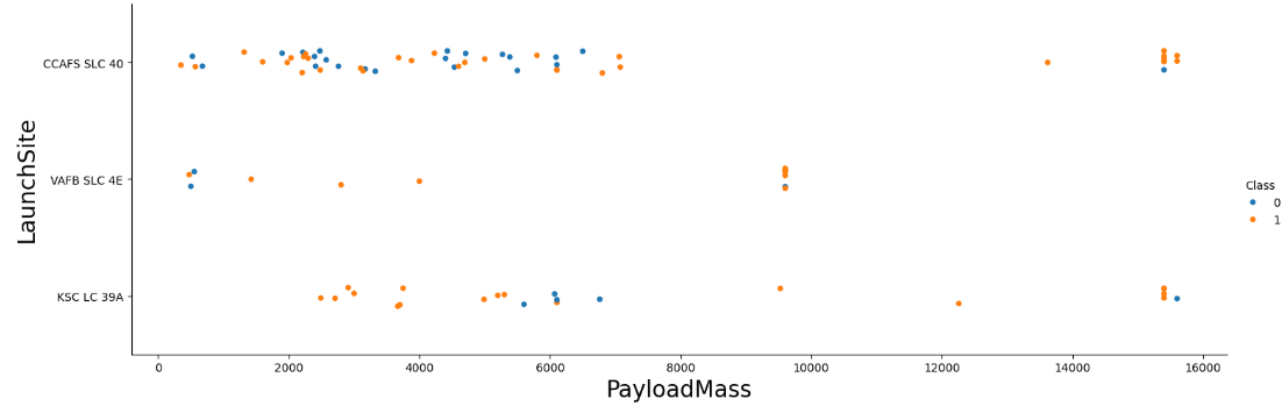
```
{'False ASDS', 'False Ocean', 'False RTLS', 'None ASDS', 'None None'}
```

```
# landing_class = 0 if bad_outcome
# landing_class = 1 otherwise

landing_class = []

for i in df["Outcome"]:
    if i in bad_outcomes:
        landing_class.append(0)
    else:
        landing_class.append(1)
landing_class
```

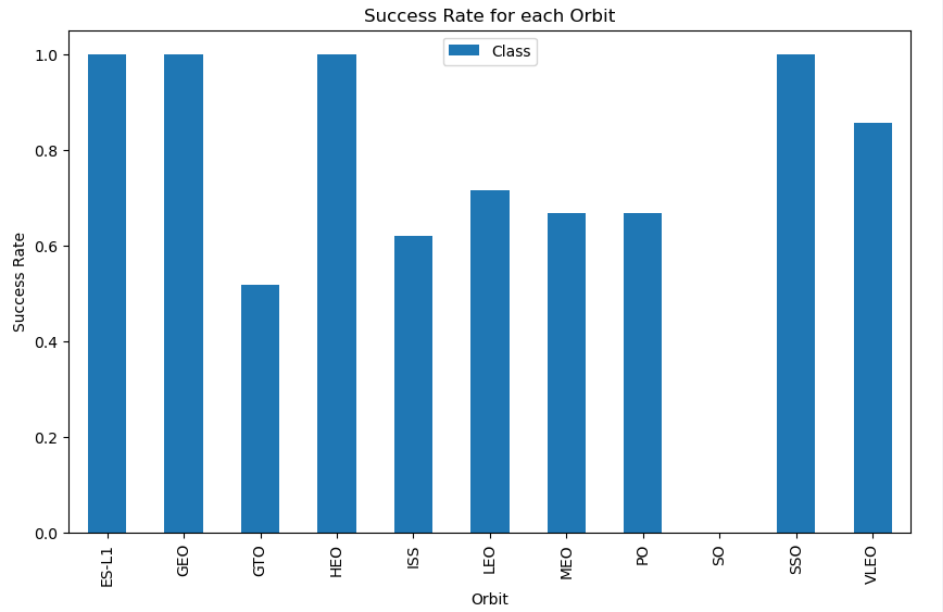
Github: Data Wrangling

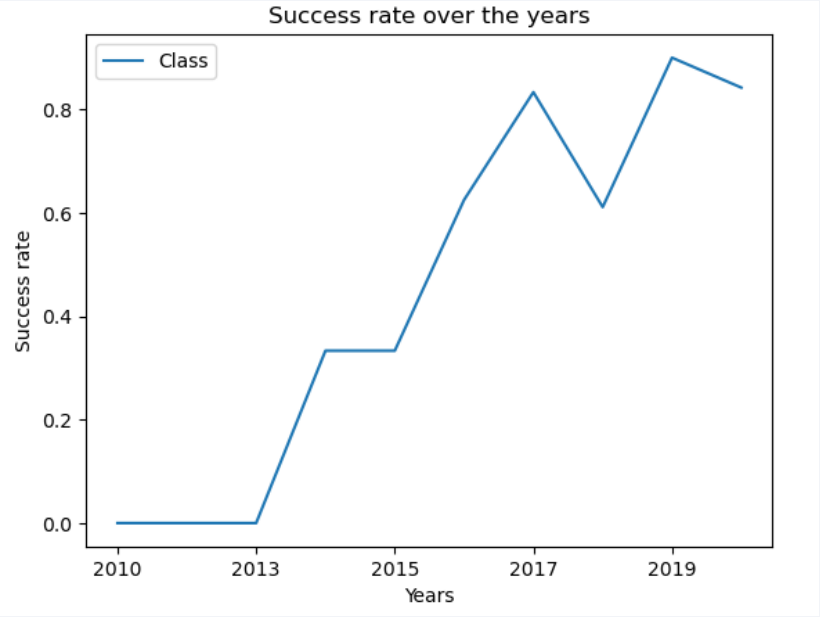
EDA with Data Visualization

Scatter Plot



Bar Chart





Line Chart

Github: [Data Visualization]

EDA with SQL

```
%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

```
%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
2010-06-04	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	7:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success	No attempt
2012-10-08	0:35:00	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

```
%%sql
SELECT Landing_Outcome, COUNT(Landing_Outcome) AS COUNT
FROM SPACEXTBL
GROUP BY Landing_Outcome
HAVING Date > '2010-06-04' AND
Date < '2017-03-20'
ORDER BY COUNT(Landing_Outcome) DESC
```

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

Landing_Outcome	COUNT
No attempt	21
Success (drone ship)	14
Success (ground pad)	9
Failure (drone ship)	5
Controlled (ocean)	5
Uncontrolled (ocean)	2
Precluded (drone ship)	1

```
%%sql
SELECT COUNT(Landing_Outcome) AS Count,
CASE
WHEN Landing_Outcome LIKE ('%Success%') THEN 'Success'
WHEN Landing_Outcome LIKE ('%None%') THEN 'Failure'
WHEN Landing_Outcome LIKE ('%Failure%') THEN 'Failure'
ELSE 'Failure'
END AS Outcome
FROM SPACEXTBL
GROUP BY Outcome
```

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

Count	Outcome
40	Failure
61	Success

```
%%sql
SELECT DISTINCT Booster_Version FROM SPACEXTBL WHERE
PAYLOAD_MASS__KG_ = (SELECT MAX(PAYLOAD_MASS__KG_) FROM SPACEXTBL) ORDER BY Booster_Version
```

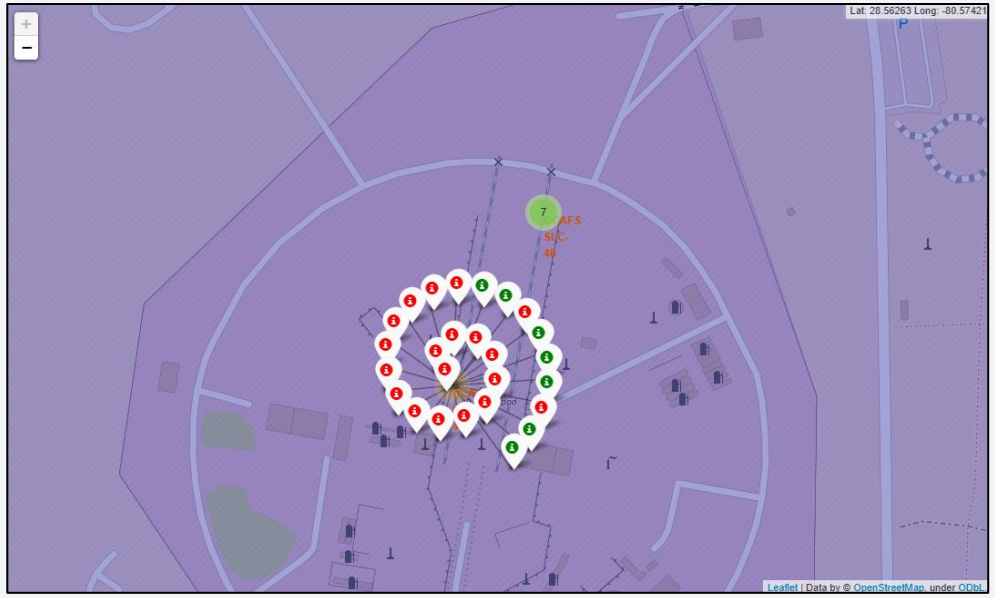
```
 * sqlite:///my_data1.db
Done.
```

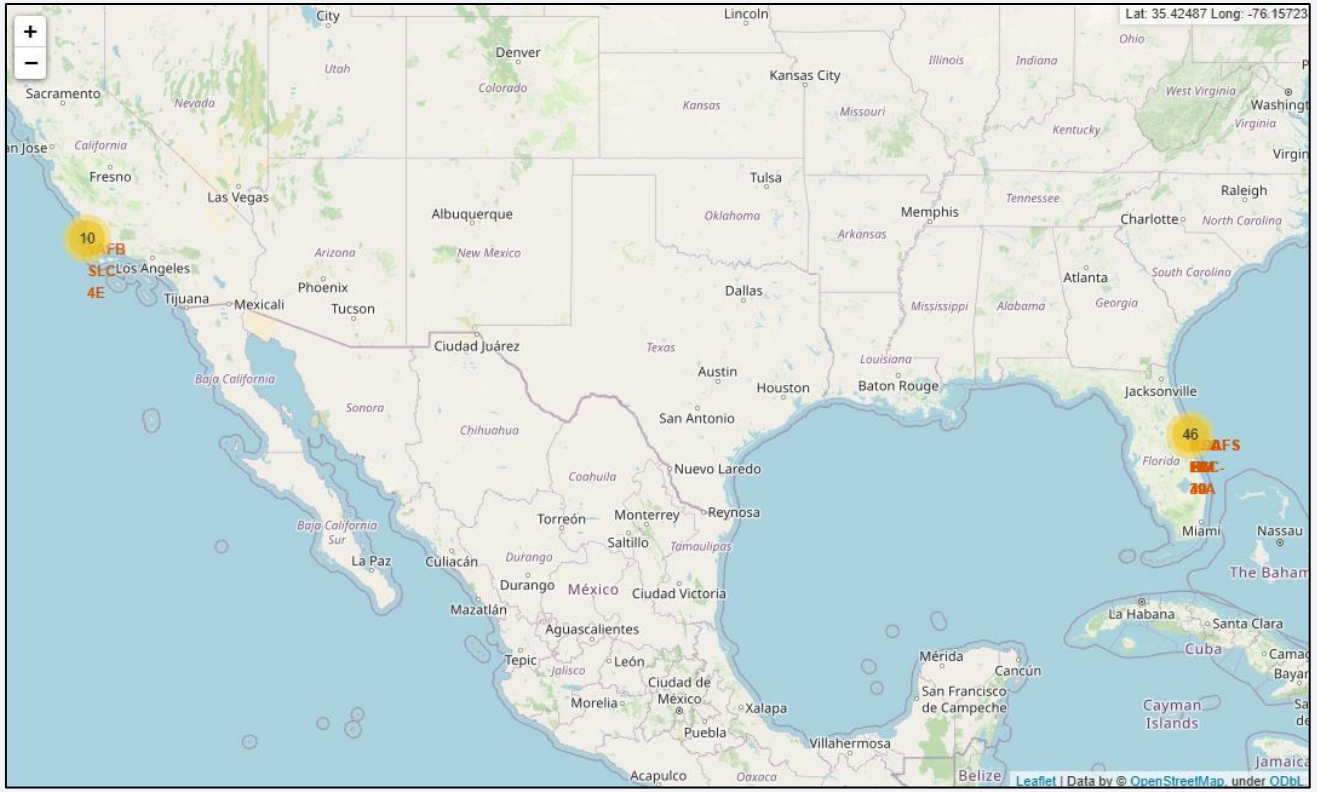
Booster_Version
F9 B5 B1048.4
F9 B5 B1048.5
F9 B5 B1049.4
F9 B5 B1049.5
F9 B5 B1049.7
F9 B5 B1051.3
F9 B5 B1051.4
F9 B5 B1051.6
F9 B5 B1056.4
F9 B5 B1058.3
F9 B5 B1060.2
F9 B5 B1060.3

Github: EDA with SQL

Build an Interactive Map with Folium

Folium map created to visually identify success of each launch site

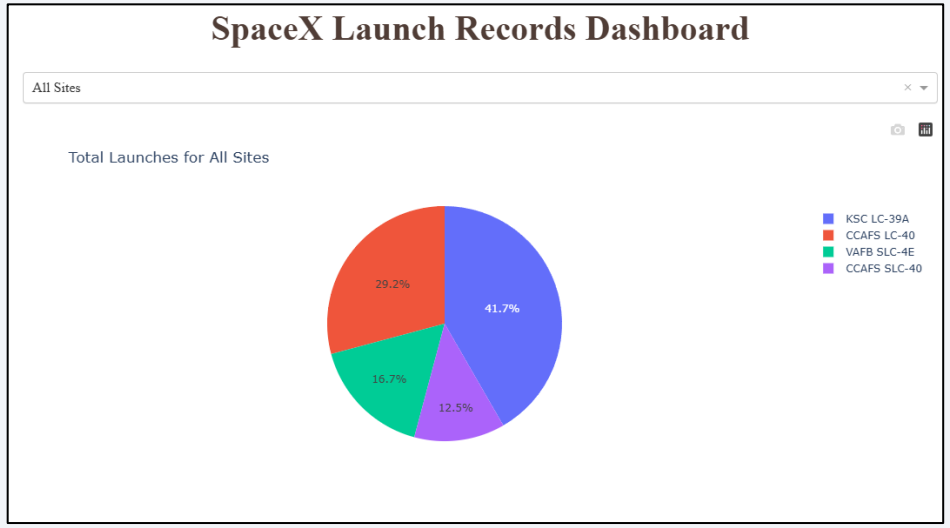


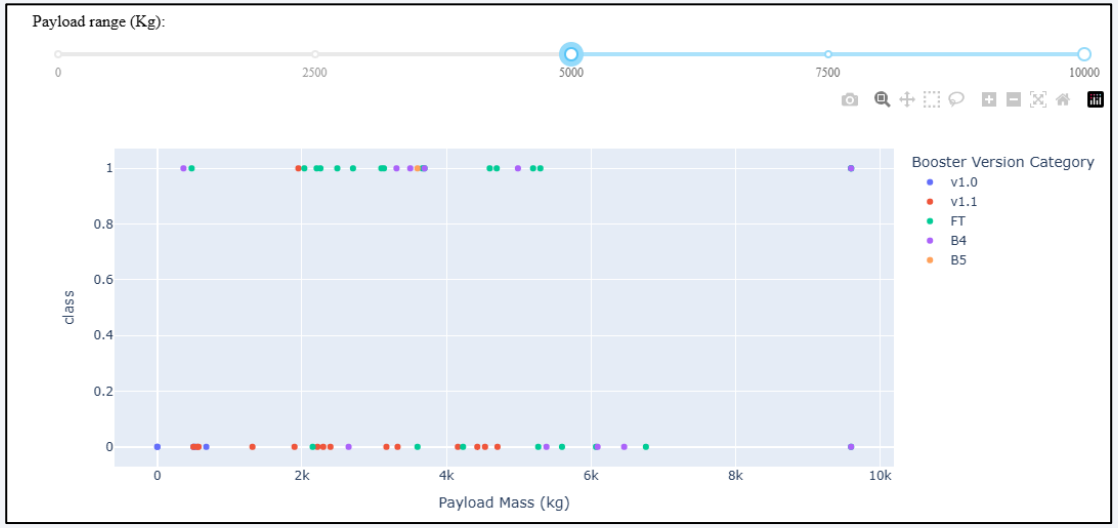


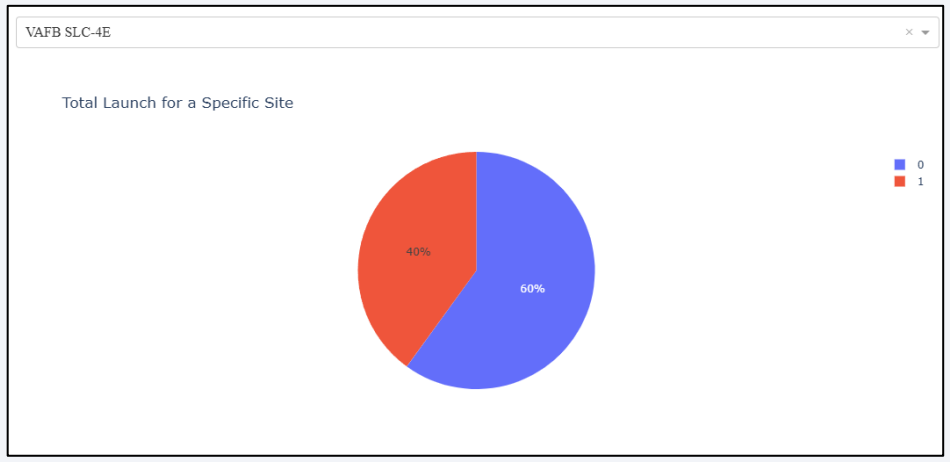
Github: [Folium Plots]

Build a Dashboard with Plotly Dash

Plotly dash was built to identify the success rate of each site







Github: [Plotly Dash]

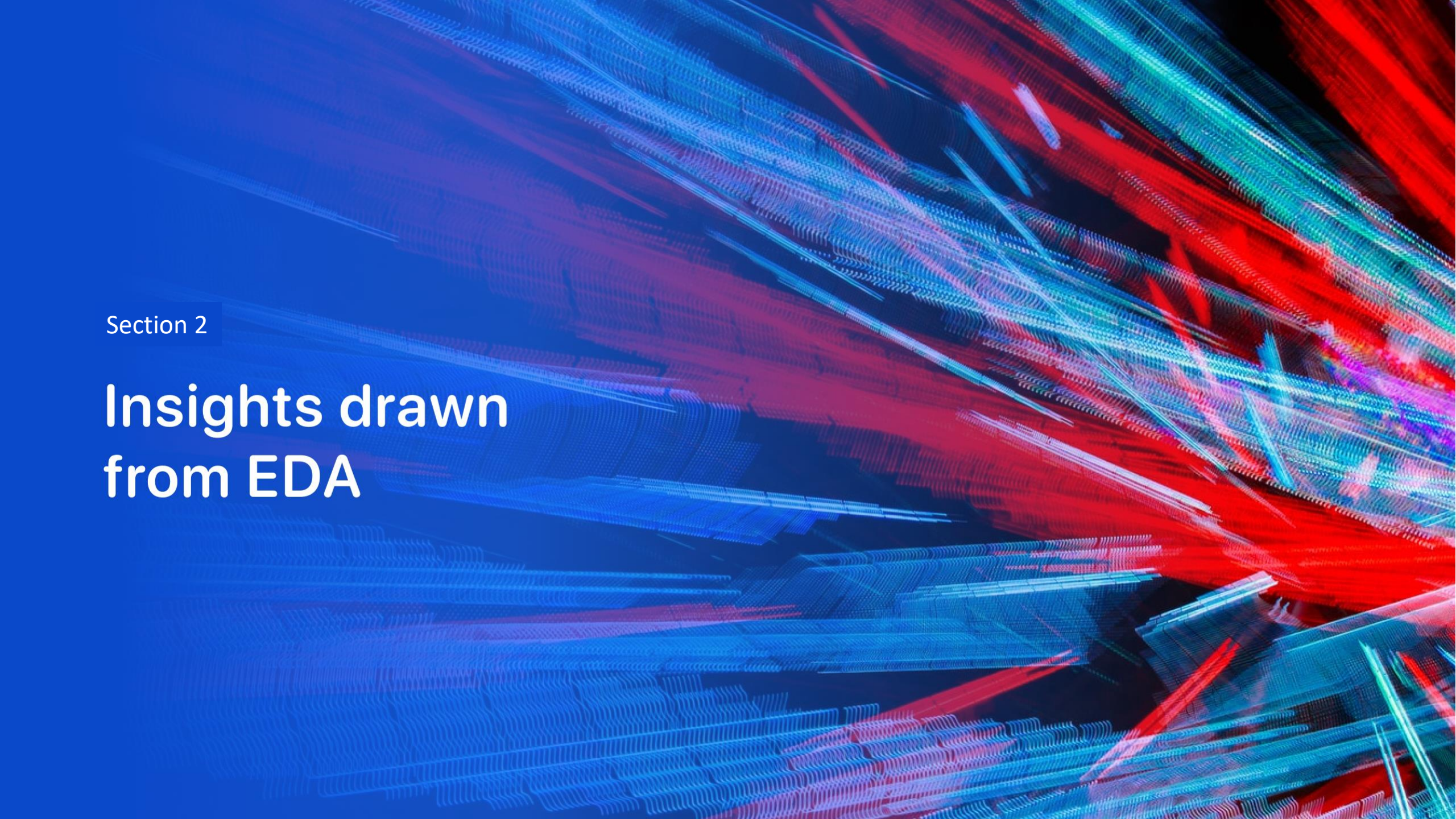
Predictive Analysis (Classification)

- Below models were used to build predictive analysis:
  - Logarithmic Regression
  - SVM
  - Decision Tree
  - KNN
- GridSearchCV was used to determine best parameters for each model and consequently calculate accuracy
- Among the models mentioned, Decision Tree model has the highest accuracy

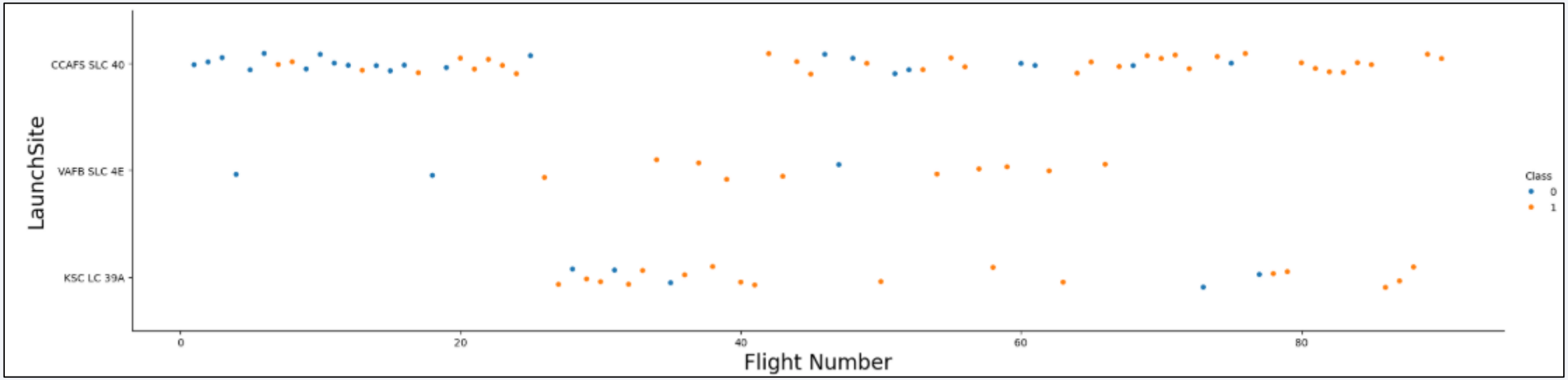
Github: Predictive Analysis

Results

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

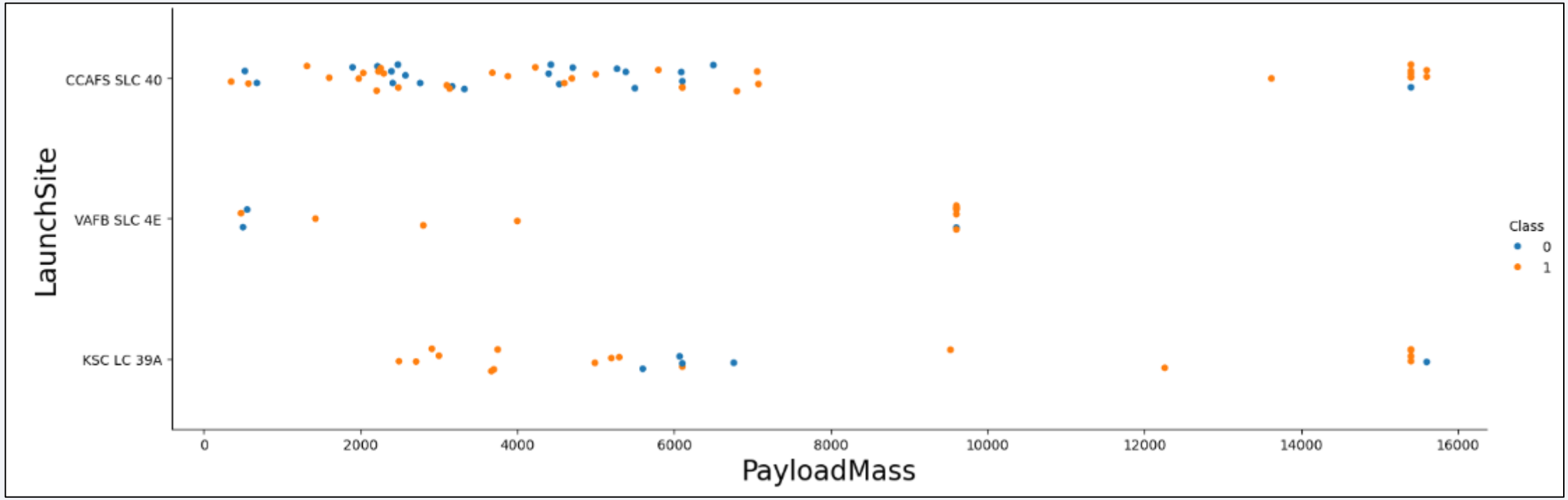

Section 2
Insights drawn from EDA

Flight Number vs. Launch Site



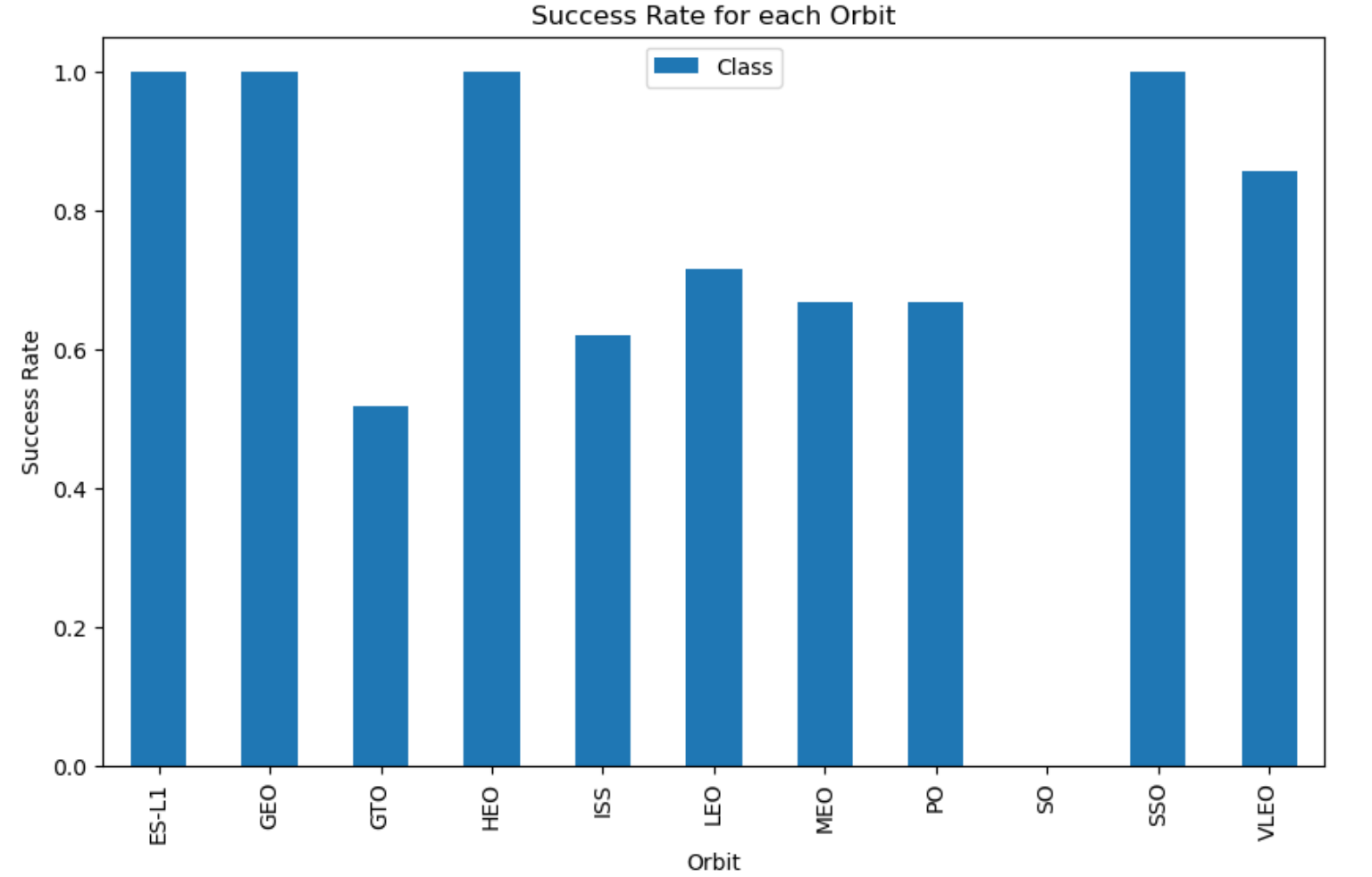
Based on the above scatter plot, for all the launch sites, success rate was higher in the launches after flight number 40.

Payload vs. Launch Site



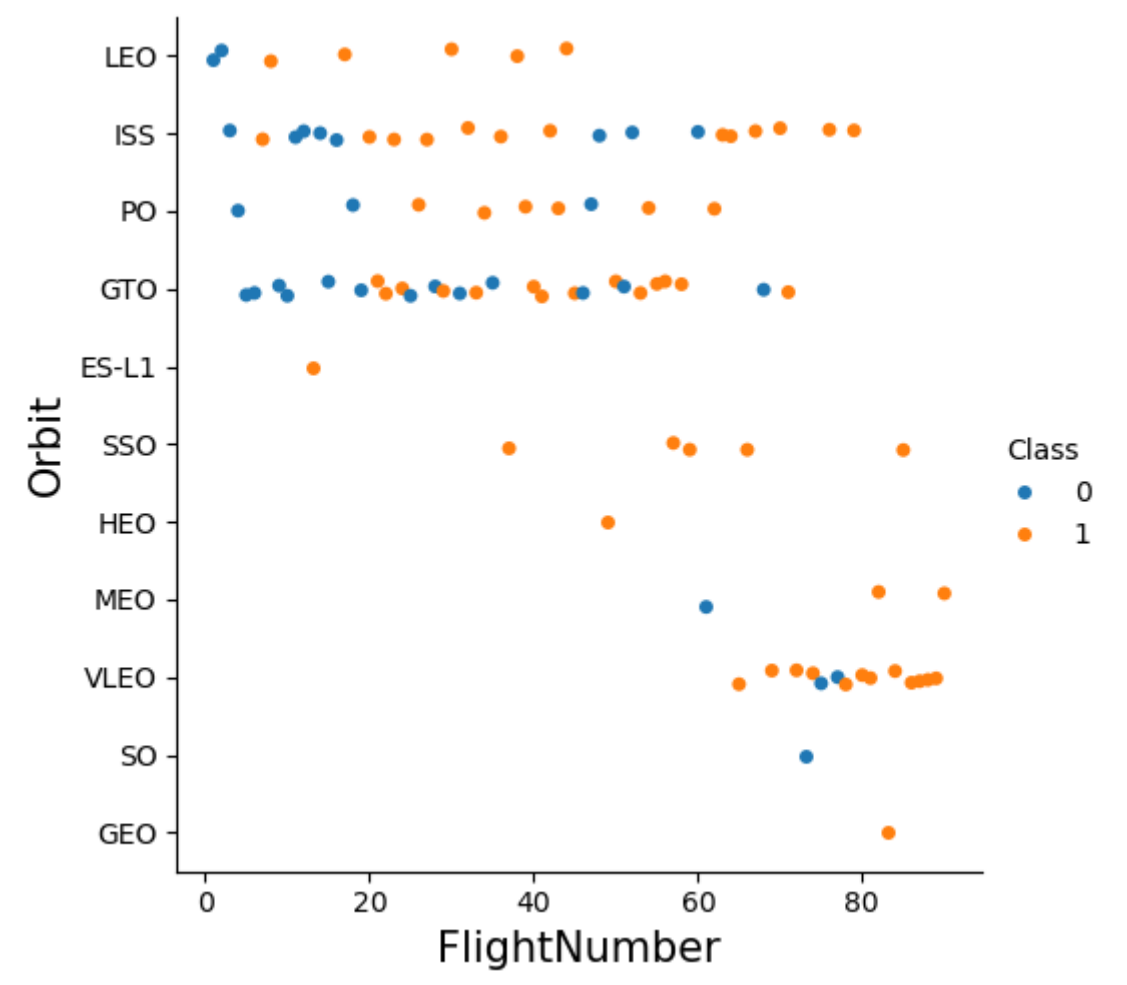
Based on the above scatter plot, for all the launch sites, success rate was higher when the Payload mass was greater than 8000 kg

Success Rate vs. Orbit Type



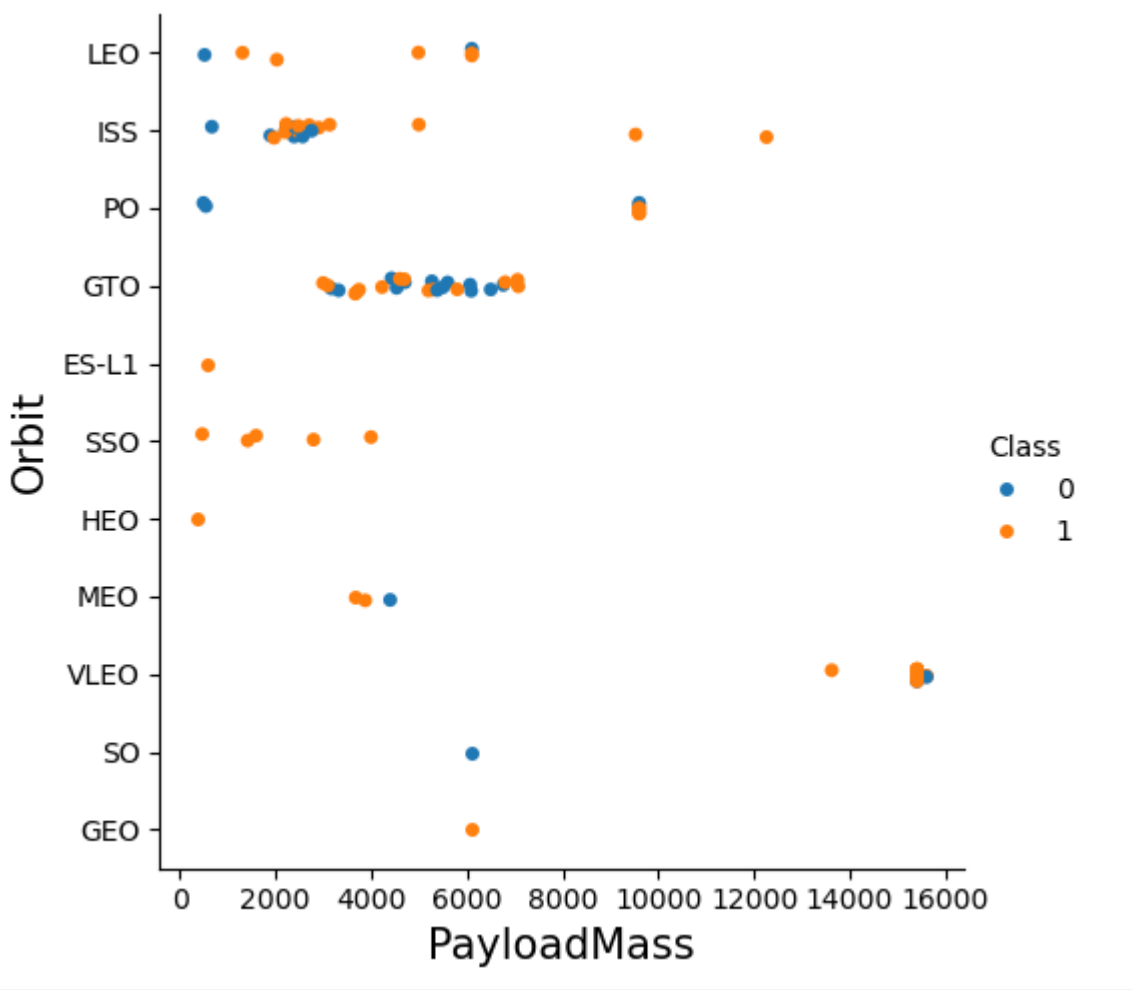
Success rate Orbits SO, GTO, ISS, LEO, MEO, PO was fairly low, whereas,
Success rate of ES-L1, GEO, HEO, SSO was 100%

Flight Number vs. Orbit Type



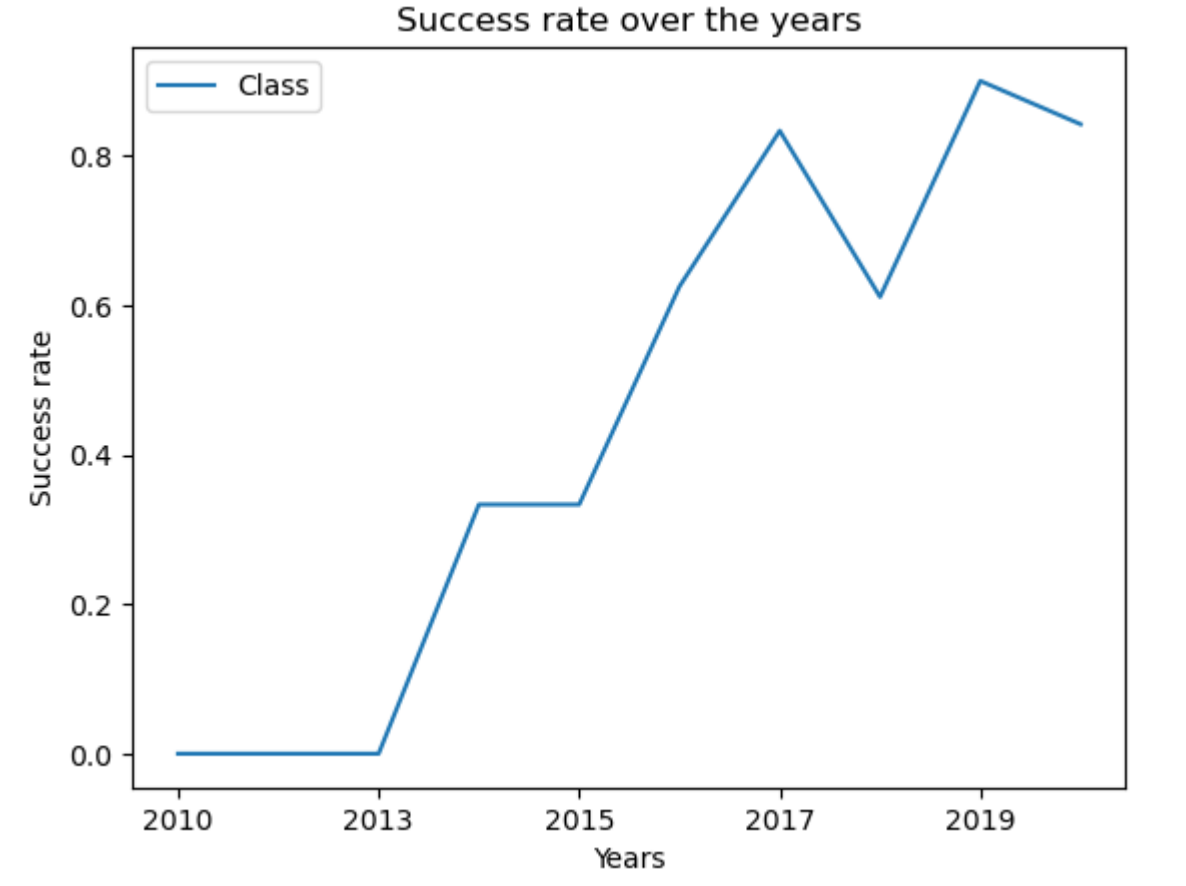
In the LEO orbit the Success appears related to the number of flights; on the other hand, there seems to be no relationship between flight number when in GTO orbit.

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 here.

Launch Success Yearly Trend



Success rate since 2013 kept increasing till 2017 (stable in 2014) and after 2015 it started increasing.

All Launch Site Names

```
%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

There are 4 launch sites from which Falcon 9 was launched

Launch Site Names Begin with 'CCA'

```
%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
2010-06-04	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	7:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success	No attempt
2012-10-08	0:35:00	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 FROM SPACEXTBL WHERE CUSTOMER LIKE ('%NASA (CRS)%')
```

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

TOTAL_PAYLOAD
48213

Average Payload Mass by F9 v1.1

```
%sql SELECT AVG(PAYLOAD_MASS__KG_) AS AVG_PAYLOAD FROM SPACEXTBL WHERE Booster_Version LIKE ('%F9 v1.1%')
```

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

AVG_PAYLOAD
2534.6666666666665

First Successful Ground Landing Date

```
%sql SELECT Date FROM SPACEXTBL WHERE Landing_Outcome LIKE ('%Success%') AND Landing_Outcome LIKE ('%ground pad%') ORDER BY DATE LIMIT 1
```

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

Date
2015-12-22

Successful Drone Ship Landing with Payload between 4000 and 6000

```
%%sql
SELECT DISTINCT Booster_Version FROM SPACEXTBL WHERE Landing_Outcome LIKE ('%Success%') AND Landing_Outcome LIKE ('%drone ship%')
AND PAYLOAD_MASS__KG_ > 4000 AND PAYLOAD_MASS__KG_ < 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

```
%%sql
SELECT COUNT(Landing_Outcome) AS Count,
CASE
WHEN Landing_Outcome LIKE ('%Success%') THEN 'Success'
WHEN Landing_Outcome LIKE ('%None%') THEN 'Failure'
WHEN Landing_Outcome LIKE ('%Failure%') THEN 'Failure'
ELSE 'Failure'
END AS Outcome
FROM SPACEXTBL
GROUP BY Outcome
```

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

Count	Outcome
40	Failure
61	Success

Boosters Carried Maximum Payload

```
%%sql
SELECT DISTINCT Booster_Version FROM SPACEXTBL WHERE
PAYLOAD_MASS__KG_ = (SELECT MAX(PAYLOAD_MASS__KG_) FROM SPACEXTBL) ORDER BY Booster_Version
```

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

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

2015 Launch Records

```
%%sql
SELECT substr(Date,6,2) AS MONTH, Landing_Outcome, Booster_Version, Launch_Site FROM SPACEXTBL
WHERE substr(Date,0,5)=='2015' AND
Landing_Outcome LIKE '%Failure%' AND
Landing_Outcome LIKE '%drone ship%'
```

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

```
%%sql
SELECT Landing_Outcome, COUNT(Landing_Outcome) AS COUNT
FROM SPACEXTBL
GROUP BY Landing_Outcome
HAVING Date > '2010-06-04' AND
Date < '2017-03-20'
ORDER BY COUNT(Landing_Outcome) DESC
```

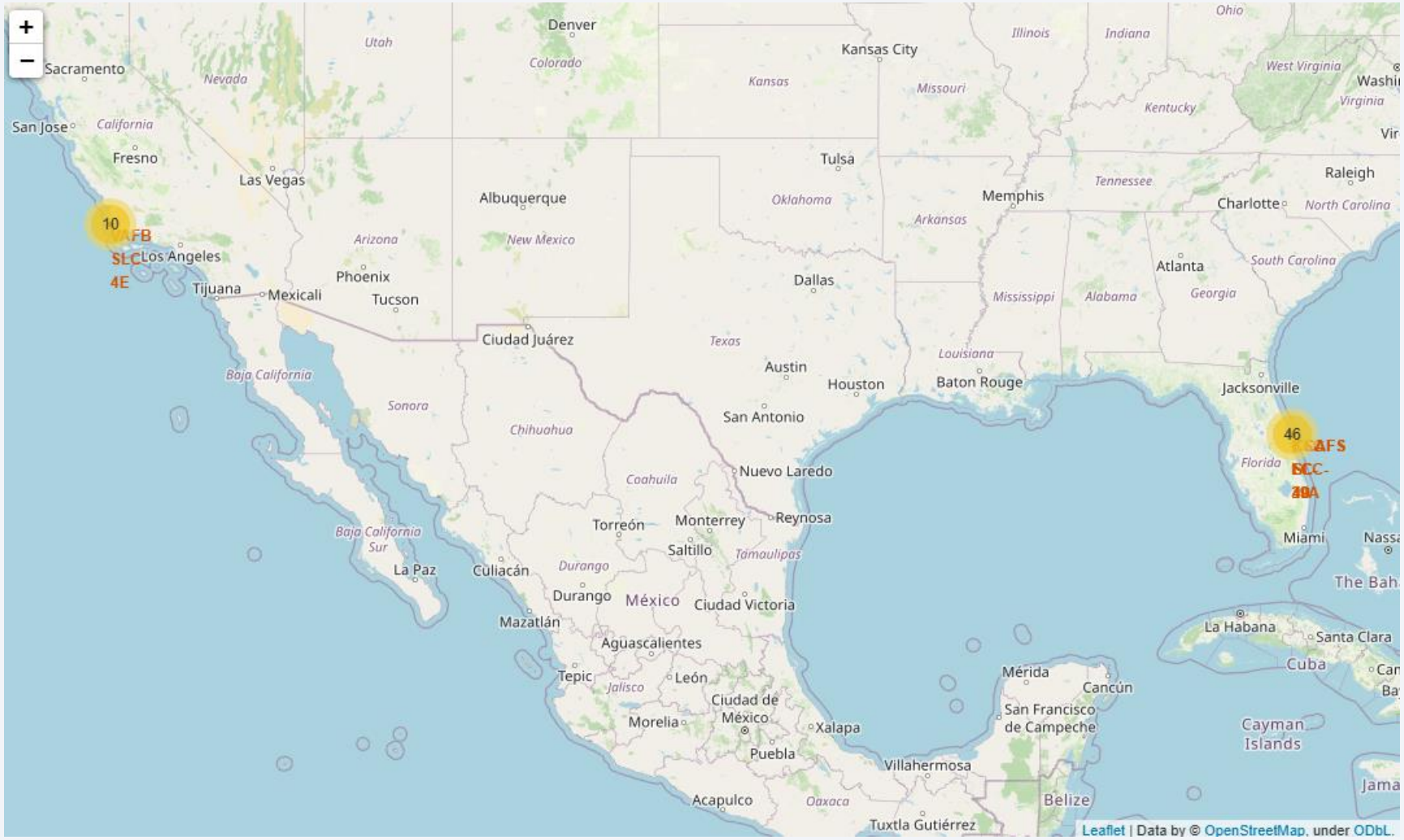
```
 * sqlite:///my_data1.db
Done.
```

Landing_Outcome	COUNT
No attempt	21
Success (drone ship)	14
Success (ground pad)	9
Failure (drone ship)	5
Controlled (ocean)	5
Uncontrolled (ocean)	2
Precluded (drone ship)	1

Section 3

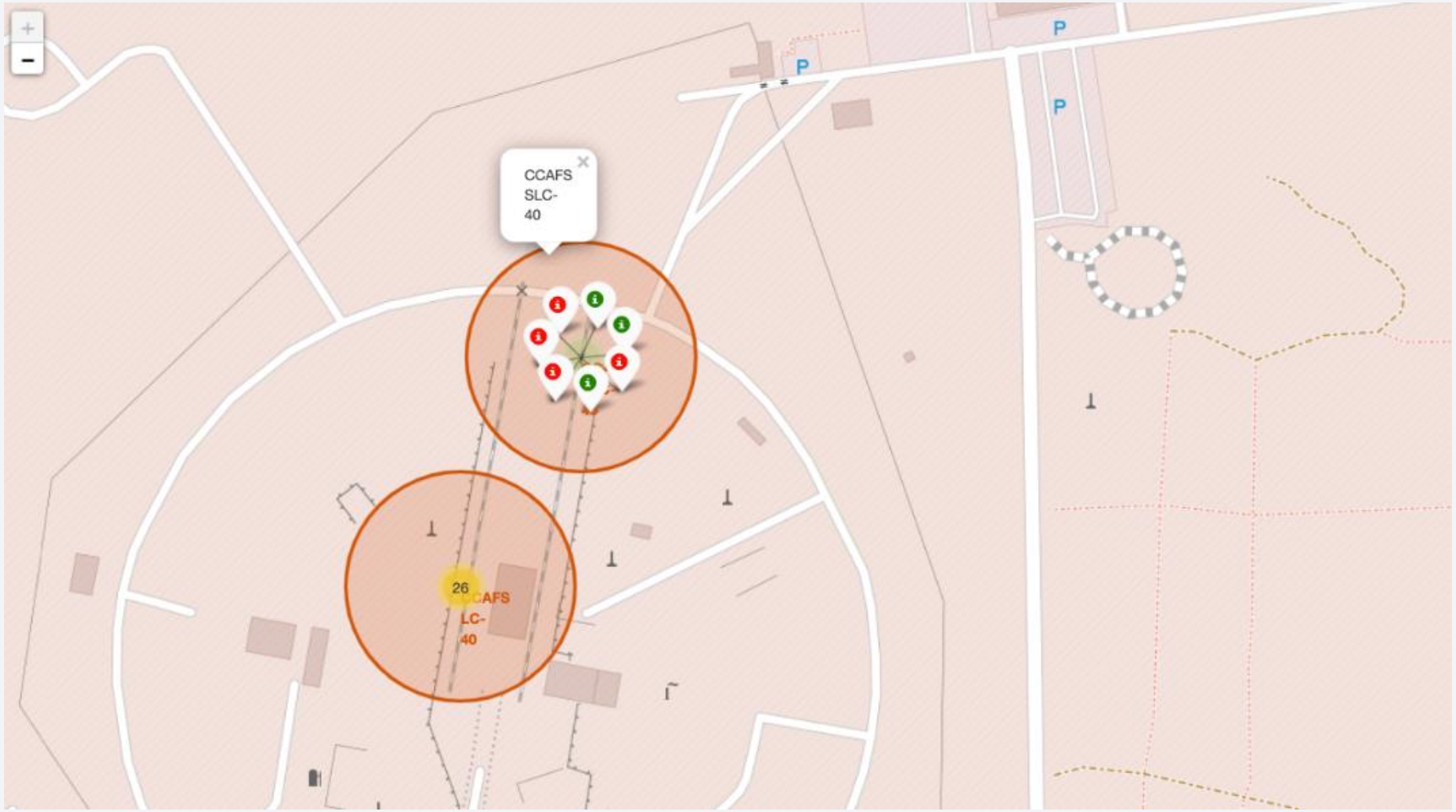
Launch Sites Proximities Analysis

Launch Site and Count of Launches



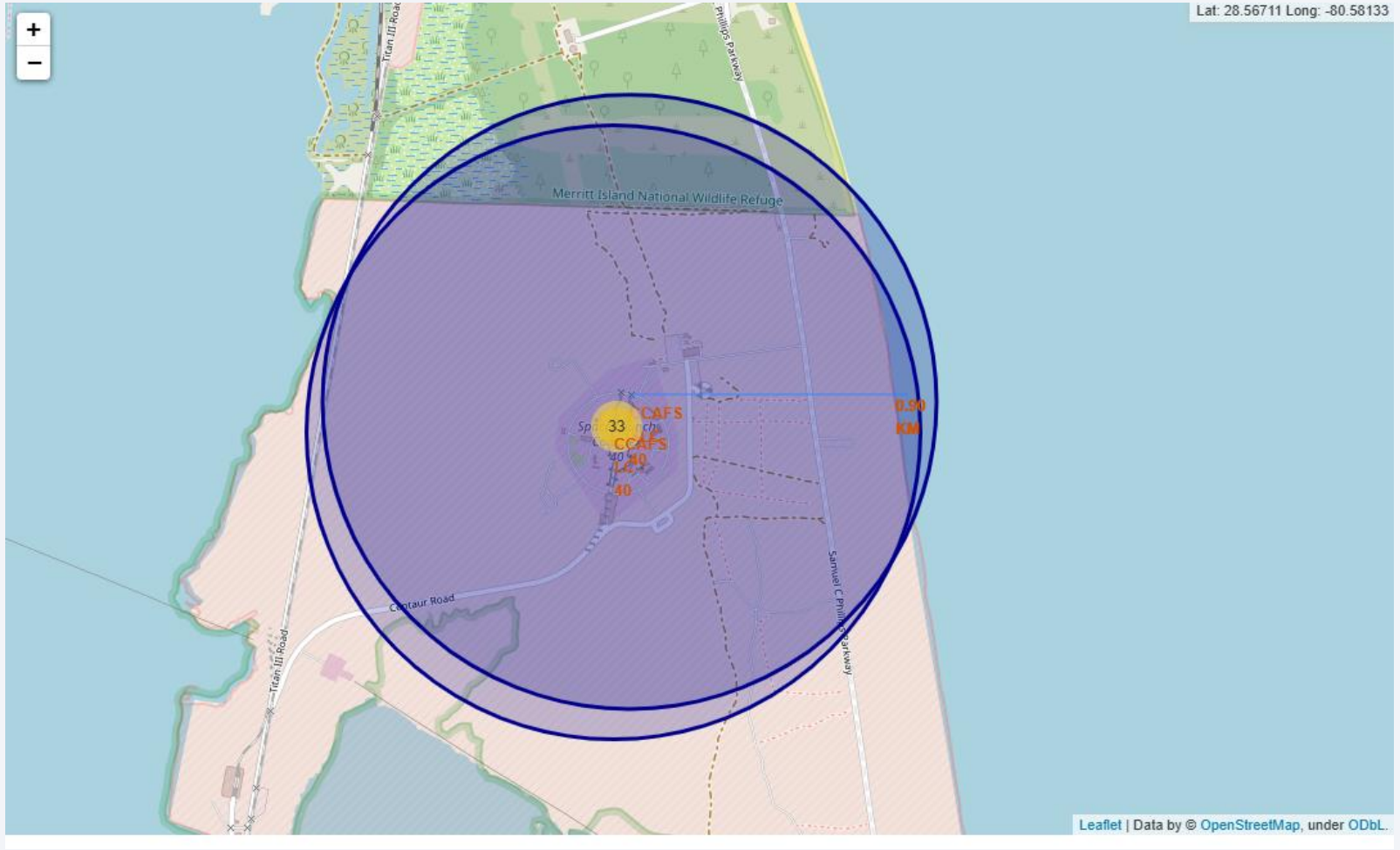
The folium map highlights launch site locations and number of launchs in the marker

Color-labeled markers in marker clusters

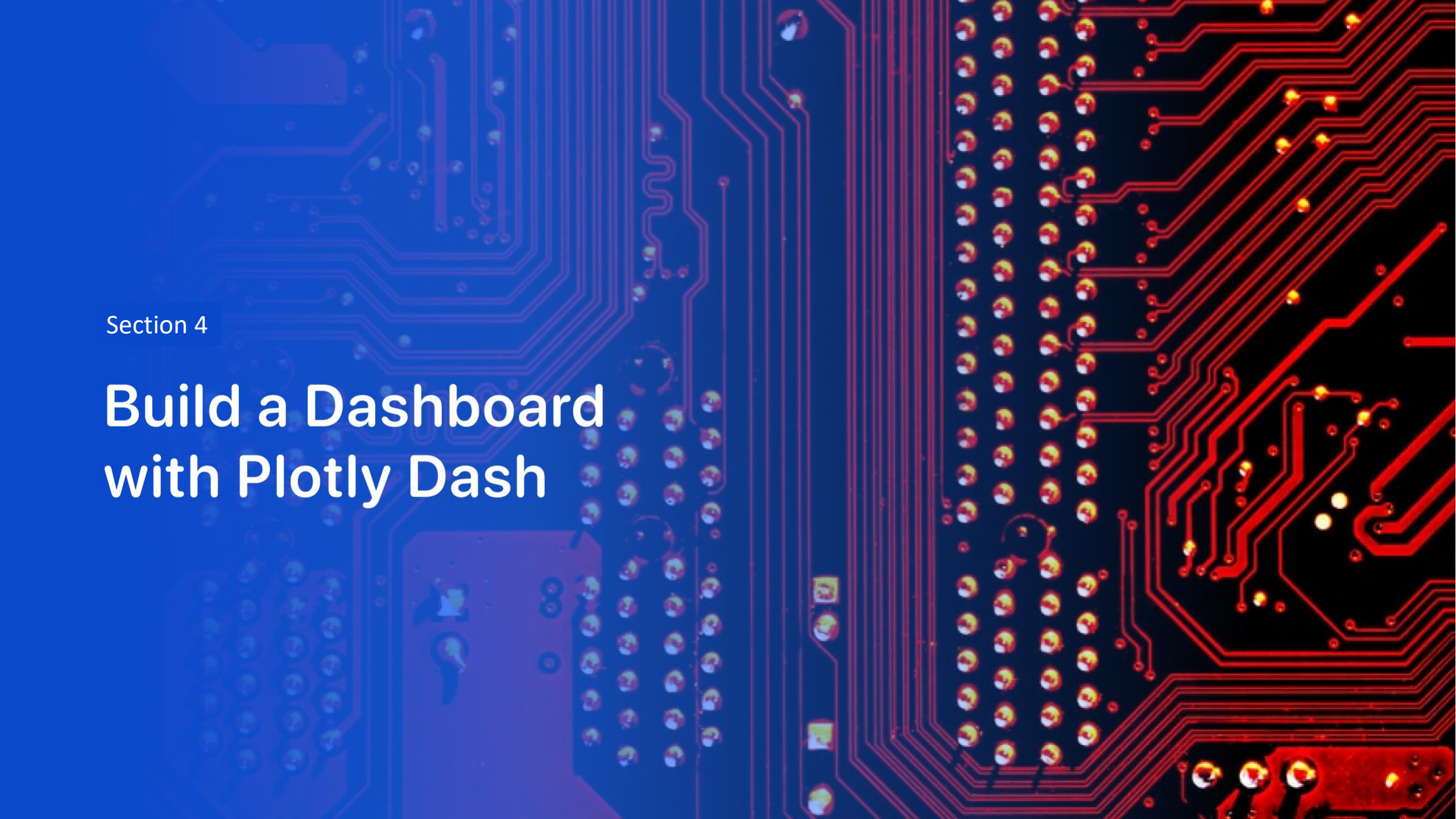


In this Folium output, Markers are individually marked as Success (Green) and Failure (Red). In this way, it will easier to identify visually the success rate of each launch site

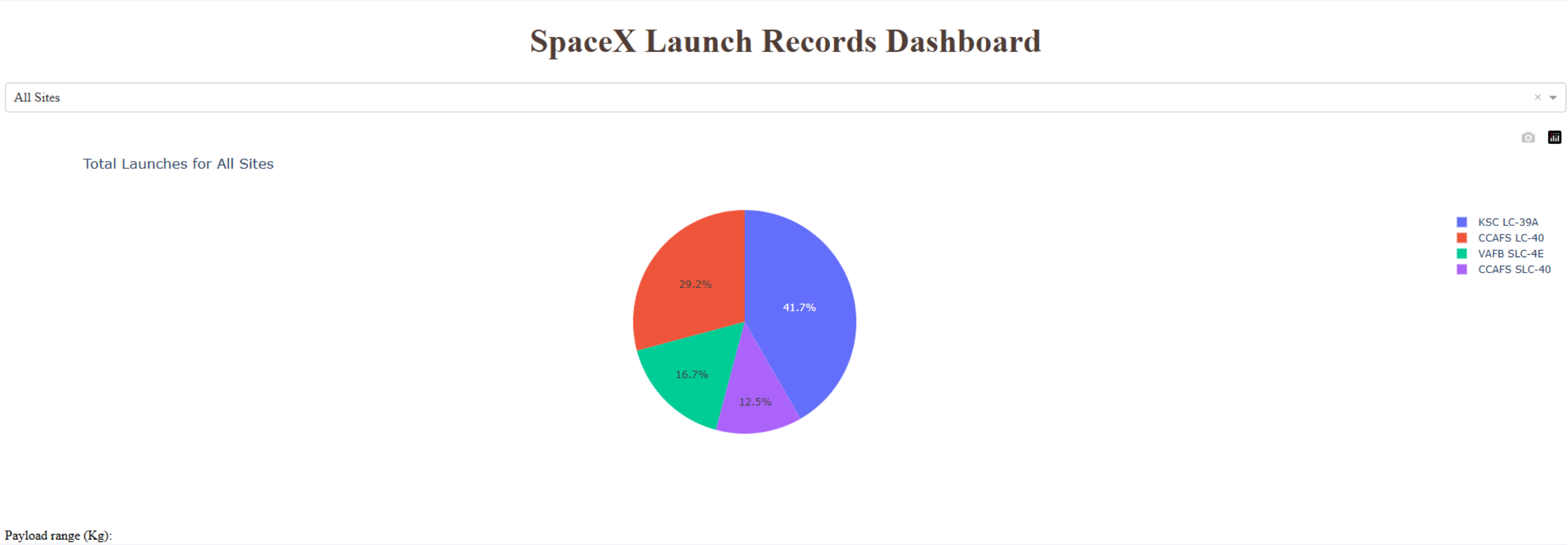
Proximity of launch site from map features



In this Folium output, distance from a particular map feature (coastline) is highlighted. This can be useful to determine the distance of launch sites to features like highway, railway and coastline

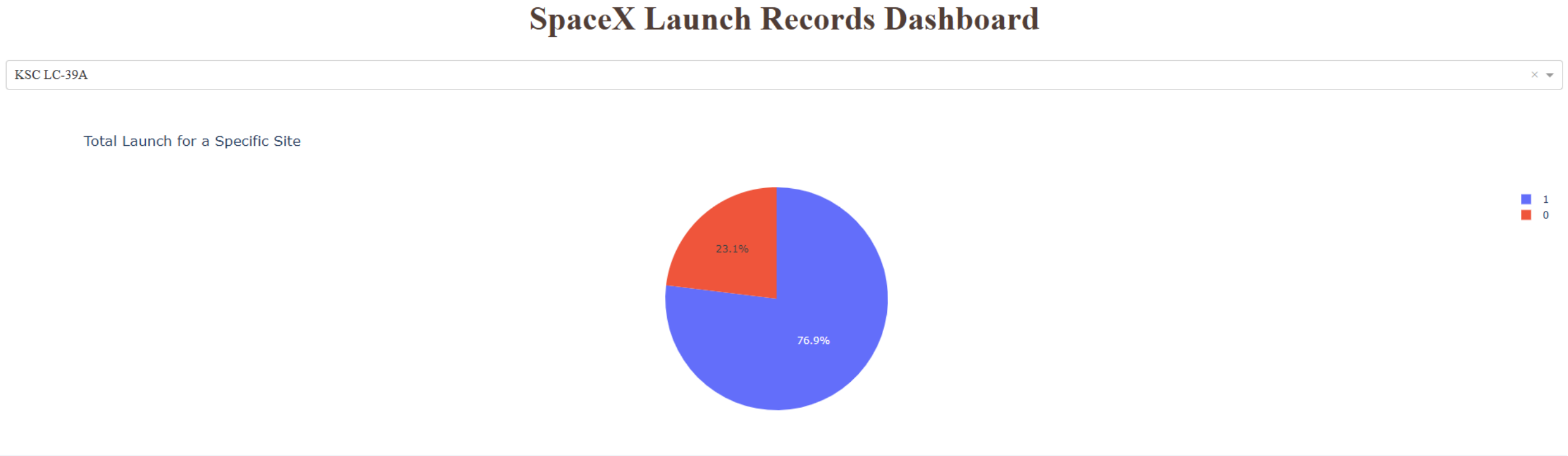

Section 4
Build a Dashboard with Plotly Dash

Distribution of launch



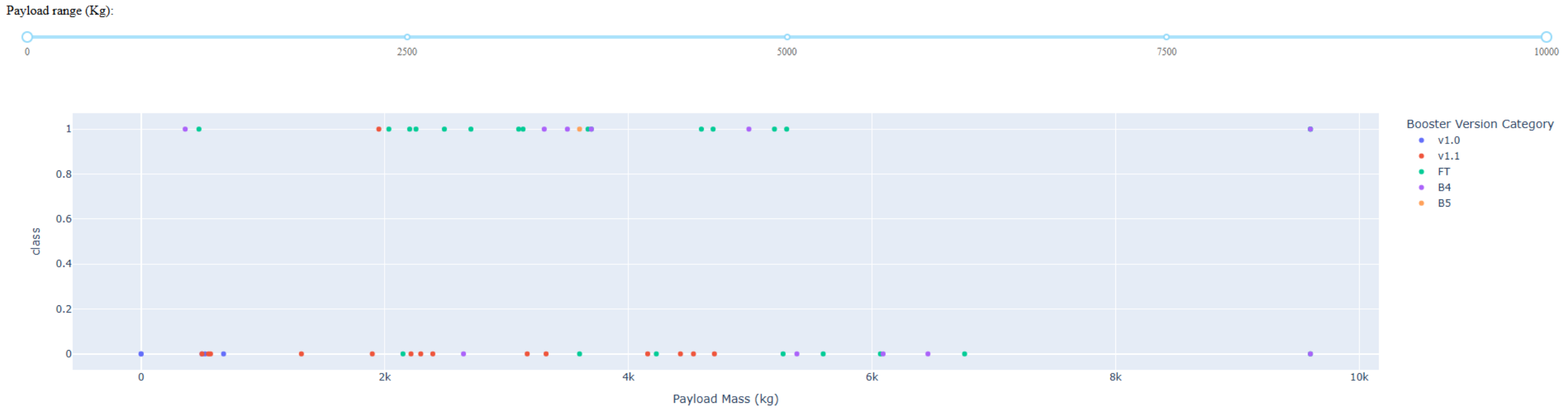
Majority of launches are done at site KSC LC-39A (~42%) and CCAFS LC-40 (~30%)

Launch Site with Highest Success Rate



KSC LC-39A has the highest success rate

Payload vs Outcome Scatter Plot

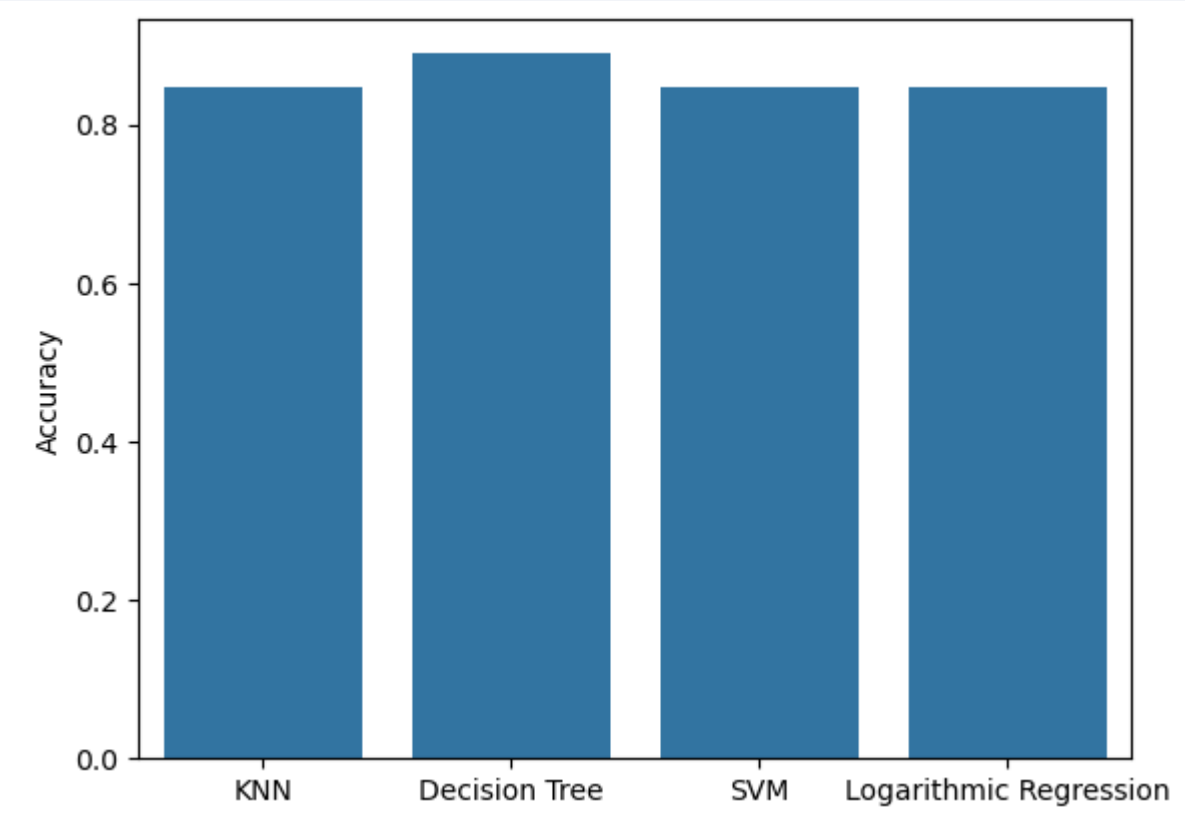


Although there is not a visible difference in the success rate across varied payload mass, it worthwhile to note that Booster version ‘FT’ has the highest success rate

Section 5

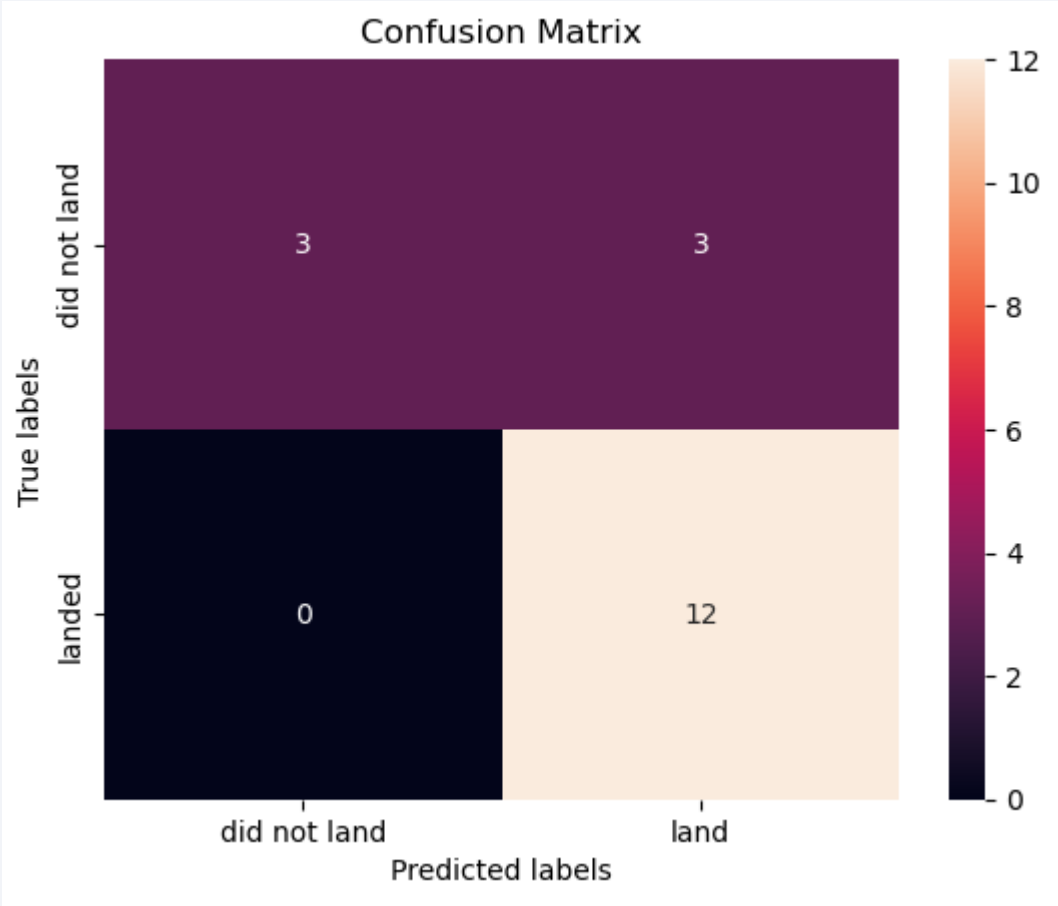
Predictive Analysis (Classification)

Classification Accuracy



Decision Tree has the highest accuracy

Confusion Matrix



The figure highlights confusion matrix of the Decision tree model.

The model accurately calculated the predicted ‘success’ outcomes against true ‘success’ outcomes (High recall)

However, the model falsely predicted few ‘failure’ outcomes as ‘success’ (Lower Precision)

Conclusions

- In this study, various models were incorporated to determine the rocket would land successfully or not.
- Key parameters to consider to maximize the success outcomes are:
  - Booster version
  - Launch site

Appendix

- N/A

Thank you!