IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Khalimonenko Nazar
25.10.2023

Outline

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

Executive Summary

Summary of methodologies

- *Data collection*
- *Data wrangling*
- *EDA with SQL*
- *Interactive analytics with Folium and Plotly Dash*
- *Predictive analysis using classifications models*

Summary of all results

- *Exploratory Data Analysis*
- *Visual analytics*
- *Predictive Analysis*

Introduction

Project background and context

The aim of this project is to predict whether or not the Falcon 9 main stage will land successfully. The cost of the Falcon 9 rocket is 62 million dollars while the other providers cost upward of 165 million dollars each. The difference in price is a result of reusable first stage of Falcon 9. So, if you can predict the likelihood of the first stage rocket successfully, you can determine the cost of a launch. In order to do this, some of Data Analysis methods were used.

Problems you want to find answers

- *Factors of successful outcome of the mission*
- *How can the best landing success rate be achieved?*

Section 1

Methodology

Methodology

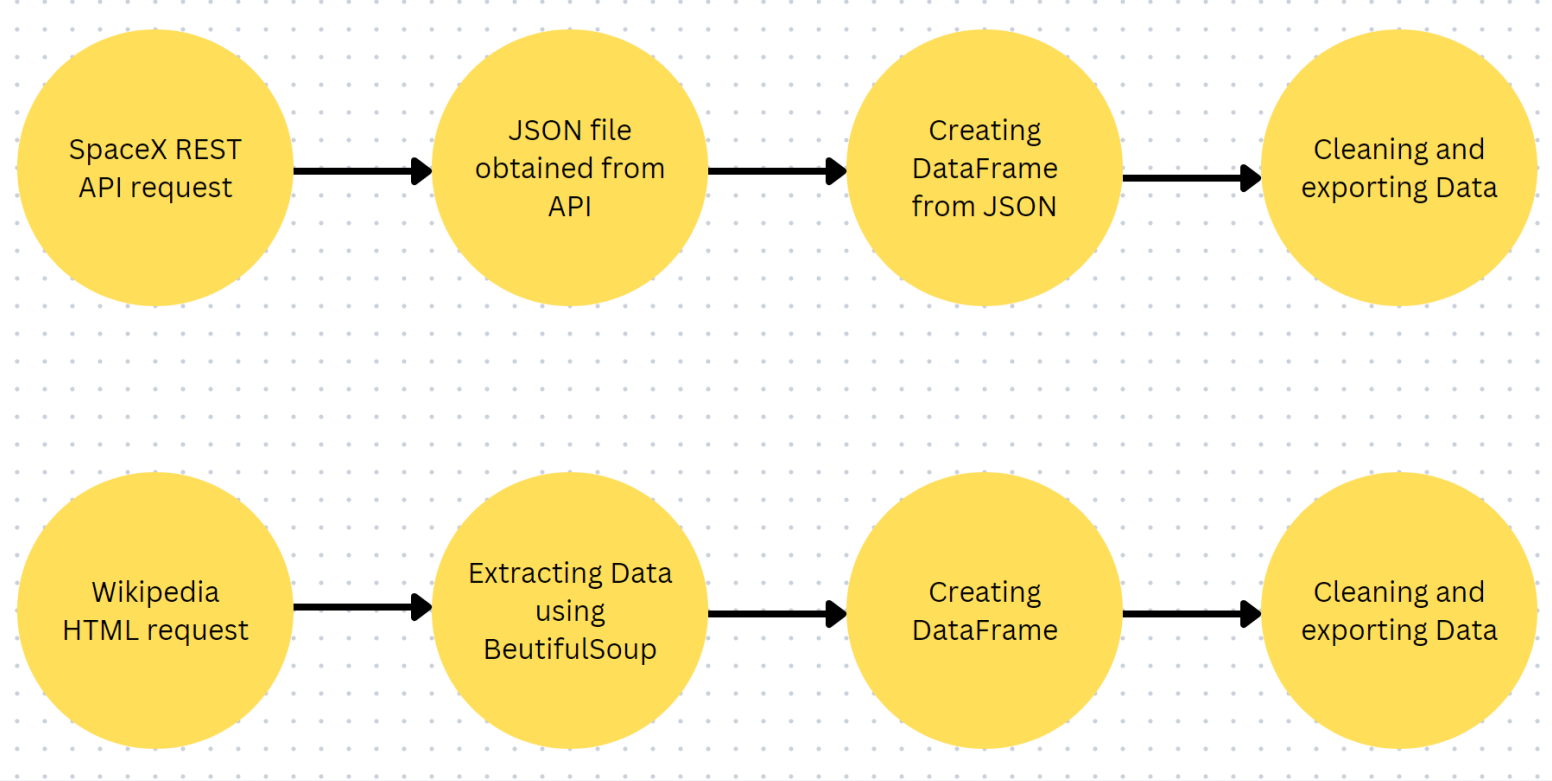
Executive Summary

- Data collection methodology:
  - SpaceX REST API, Web Scrapping
- Perform data wrangling
  - Deleting unnecessary columns, dropping null values, 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
  - Models such as LR, KNN and DT were tested and evaluated

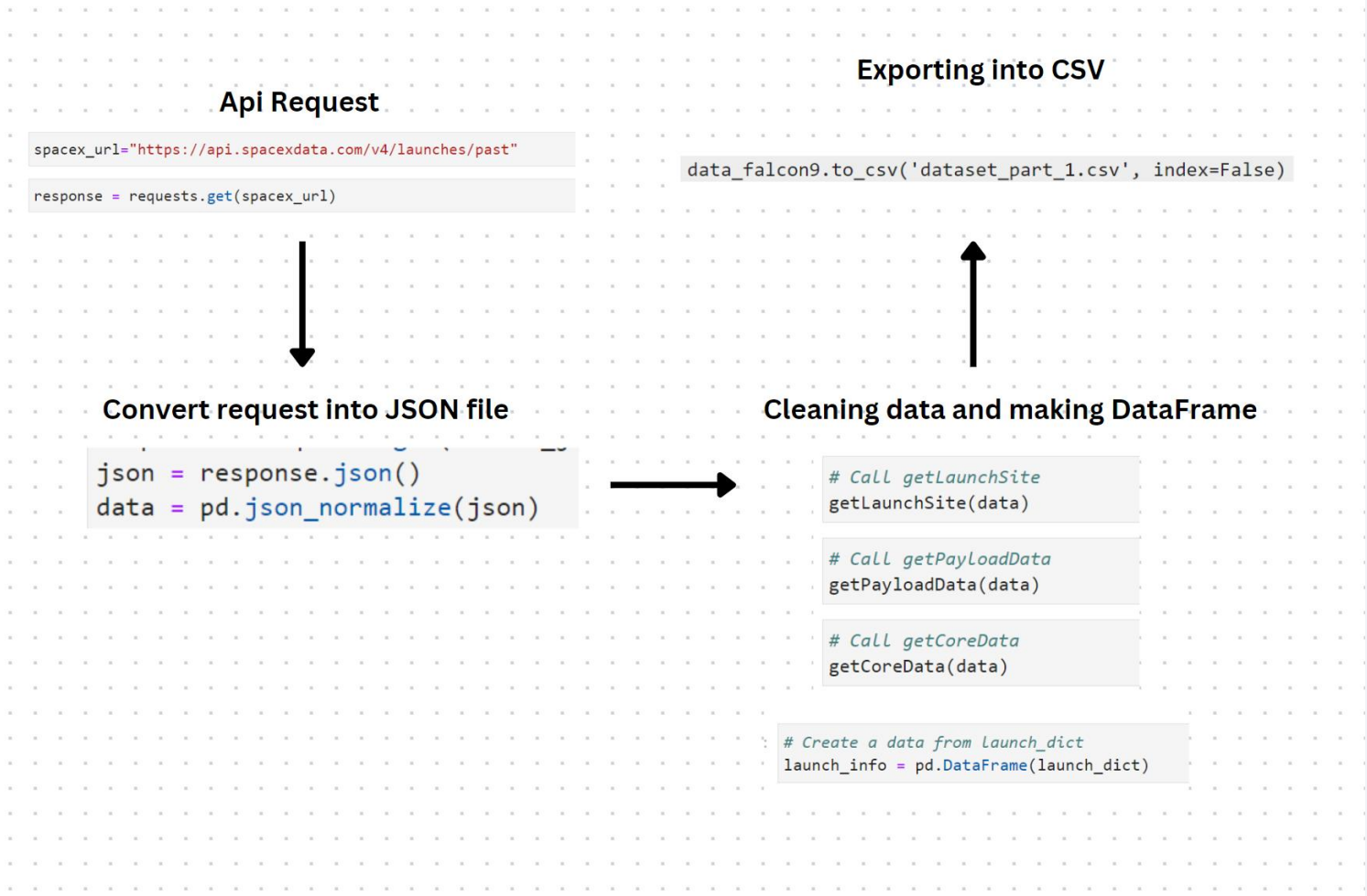
Data Collection

Resources the were used to collect data: SpaceX REST API, Wikipedia

Links: SpaceX API, Wikipedia page



Data Collection - SpaceX API



GitHub Link

Data Collection - Scraping

HTML Request and creating BS object

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

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

```
# Use BeautifulSoup() to create a BeautifulSoup object from a response
soup = BeautifulSoup(response.text, 'html.parser')
```

Creating a DataFrame by parsing HTML tables

```
extracted_row = 0
#Extract each table
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()
        else:
            flag=False
        #get table element
        row=rows.find_all('td')
```

```
df= pd.DataFrame({ key:pd.Series(value) for key, value in launch_dict.items() })
```

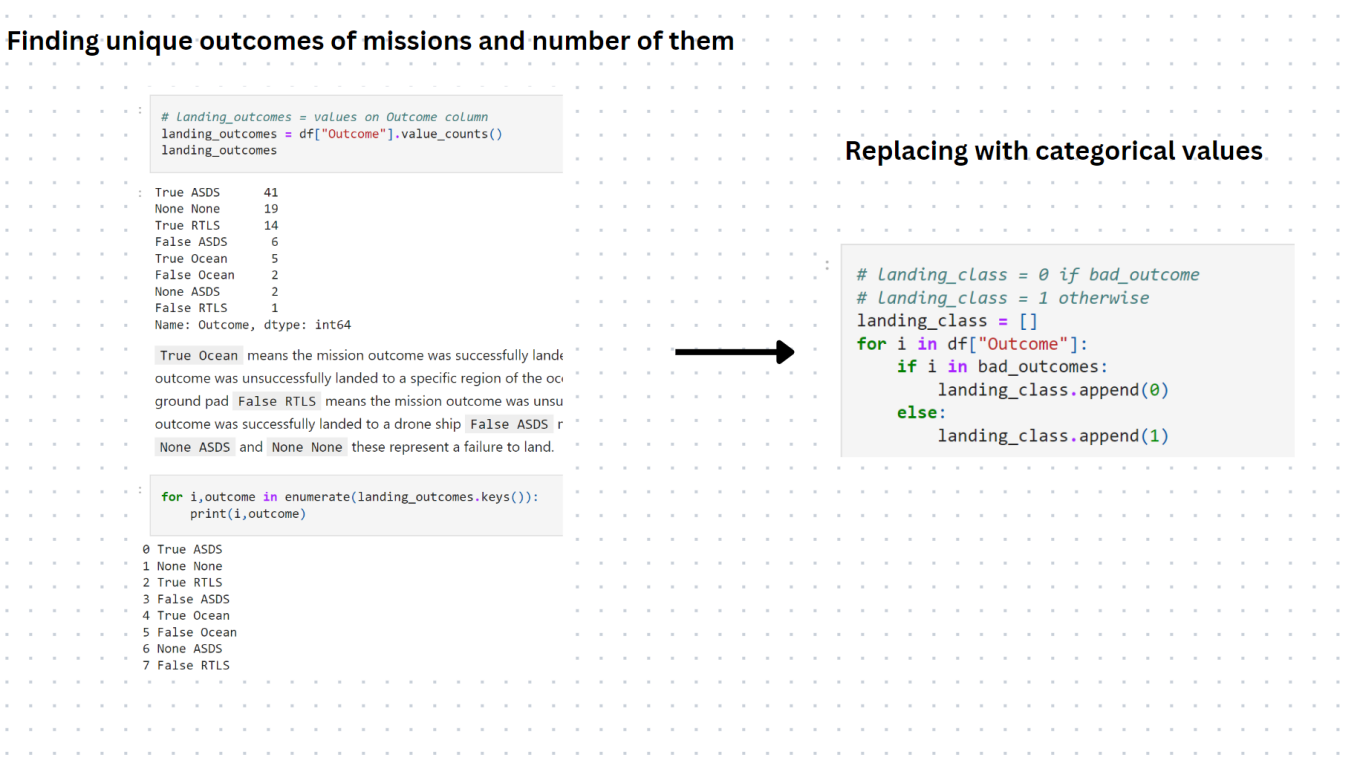
Exporting into CSV

```
df.to_csv('spacex_web_scraped.csv', index=False)
```

GitHub Link

Data Wrangling

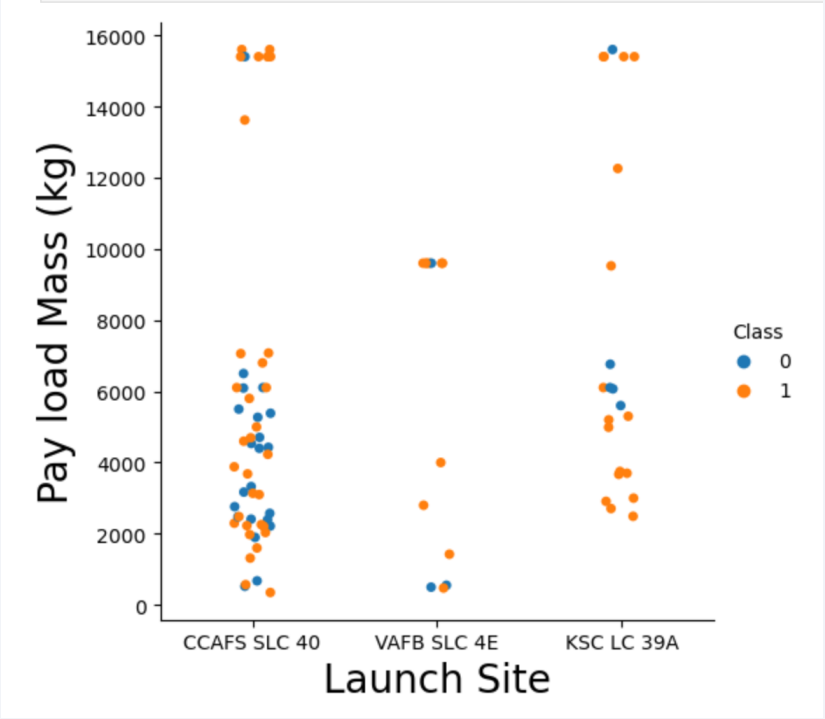
In order to complete further analysis we need to make information about the result of landing (successful/unsuccessful) more readable for ML models. So we replace it with categorical values.

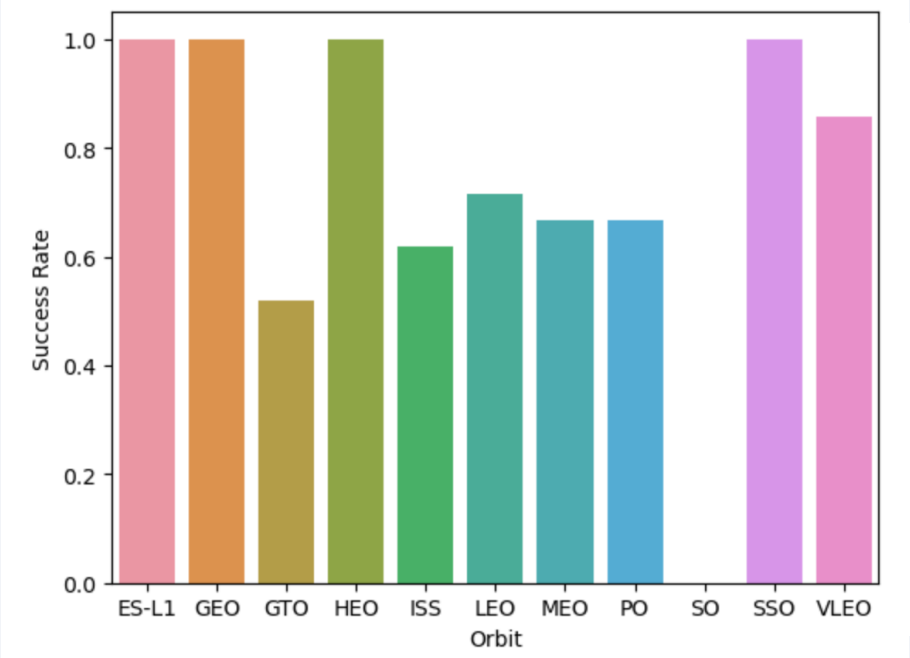


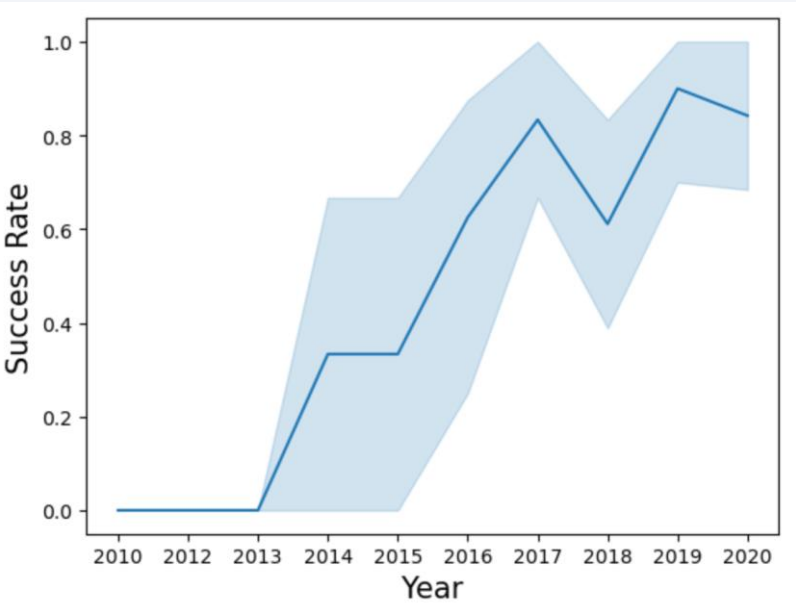
GitHub Link

EDA with Data Visualization

In order to show relation between some values plots like scatter plot, bar graph, line plot were used







GitHub Link

EDA with SQL

During EDA with SQL we used following queries:

- *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.*
- *List the records which will display the month names, failure landing_outcomes in drone ship ,booster versions, launch_site for the months 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.*

GitHub Link

Build an Interactive Map with Folium

Objects like circles, markers, etc. were added to make map more interactive and to visualize some geological factors the might affect the result of a mission.

Following objects were added:

- *Mark all launch sites on a map*
- *Mark the success/failed launches for each site on the map*
- *Calculate the distances between a launch site to its proximities*

GitHub Link

Build a Dashboard with Plotly Dash

By using Dash we created some plots like pie chart, scatter plot to visualize some relations between vales.

For example:

- *Pie chart for visualizing success ratio for each launch site*
- *Relation between Payload mass and success ratio*

GitHub Link

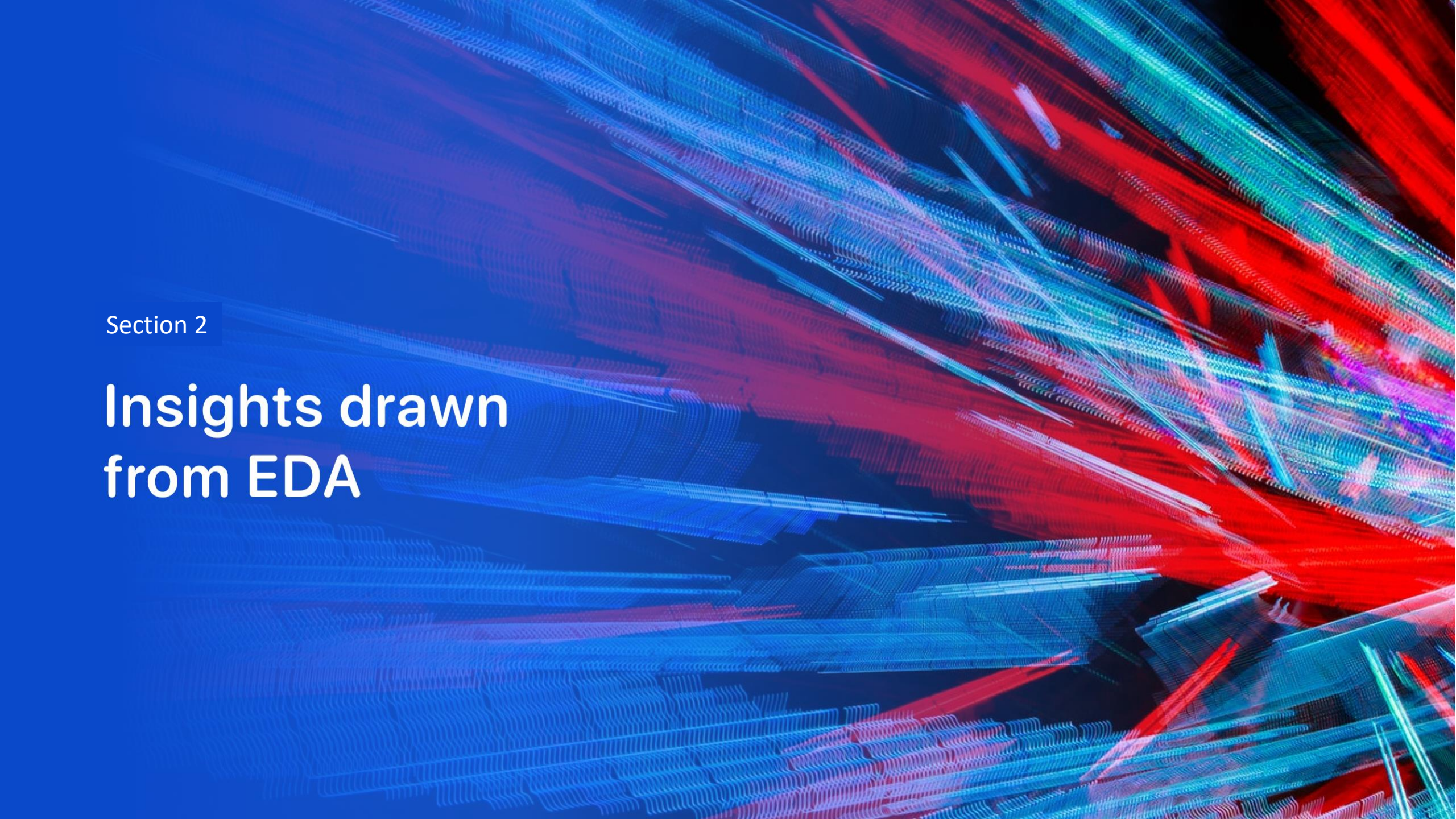
Predictive Analysis (Classification)

- Preparing data
  - Data normalization
  - Splitting data into training and testing sets
- Model building
  - Creating Ml models (LR, KNN, SVM, DT) by using GridSearchCV
  - Getting the best hyperparameters for models
  - Training models with training set
- Model evaluation
  - Finding accuracy for each model on testing test
  - Plotting Confusion Matrix
- Comparison
  - Models comparison by their accuracy rate

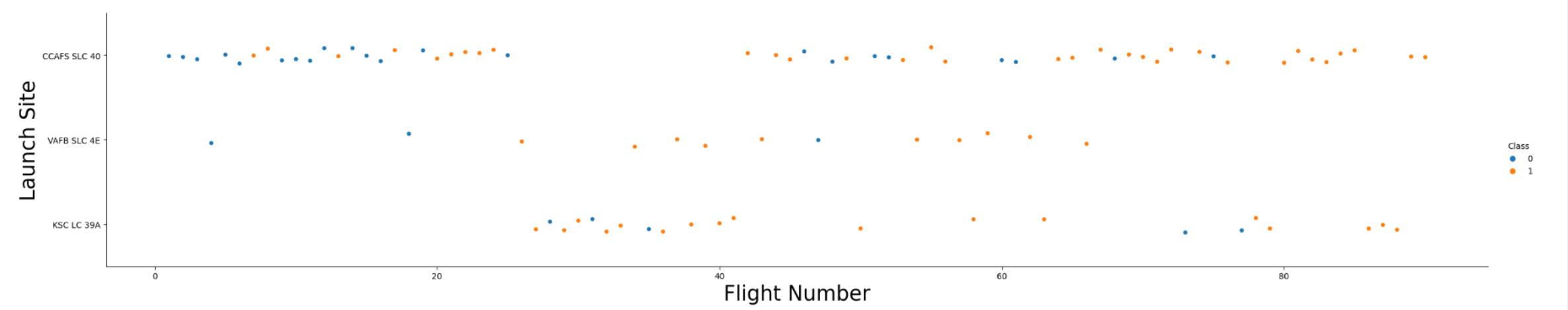
GitHub

Results

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

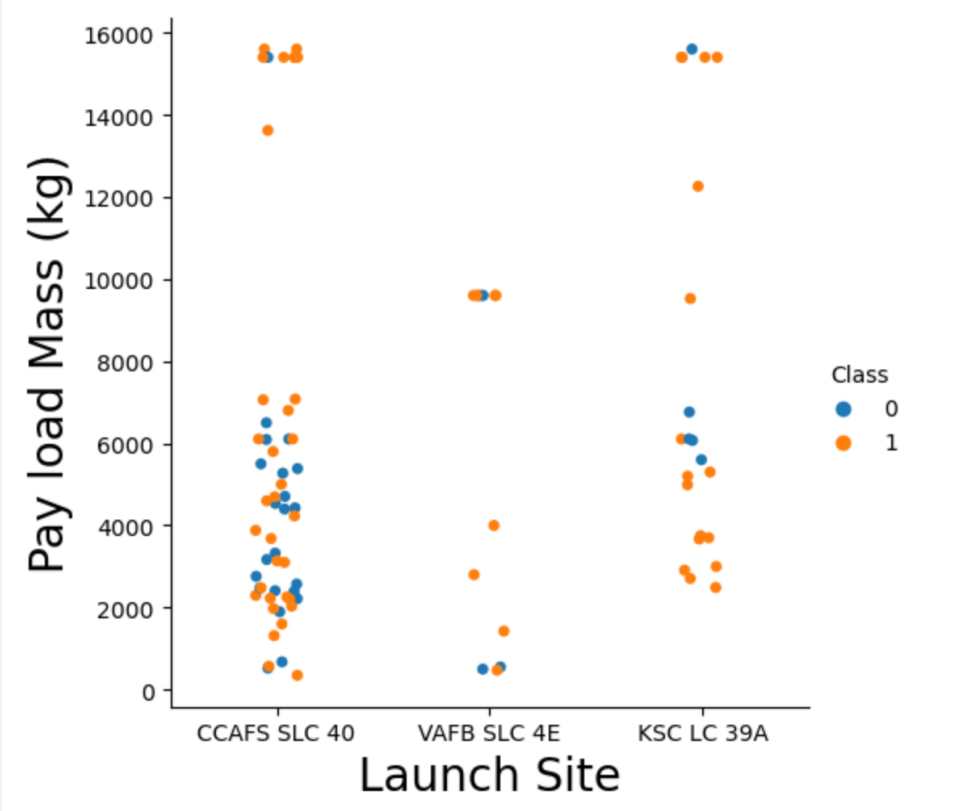

Section 2
Insights drawn from EDA

Flight Number vs. Launch Site



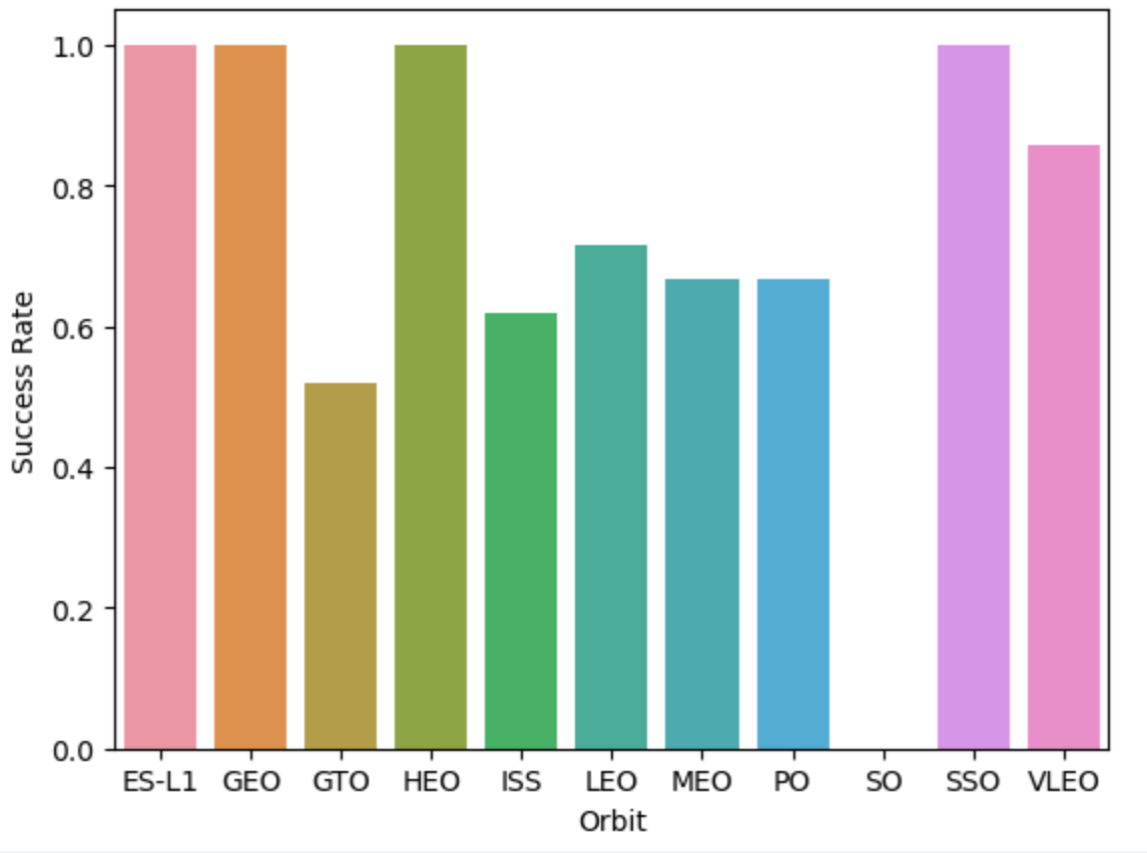
It can be seen that with increasing number of flights the success rate on each launch sites is also increasing.

Payload vs. Launch Site



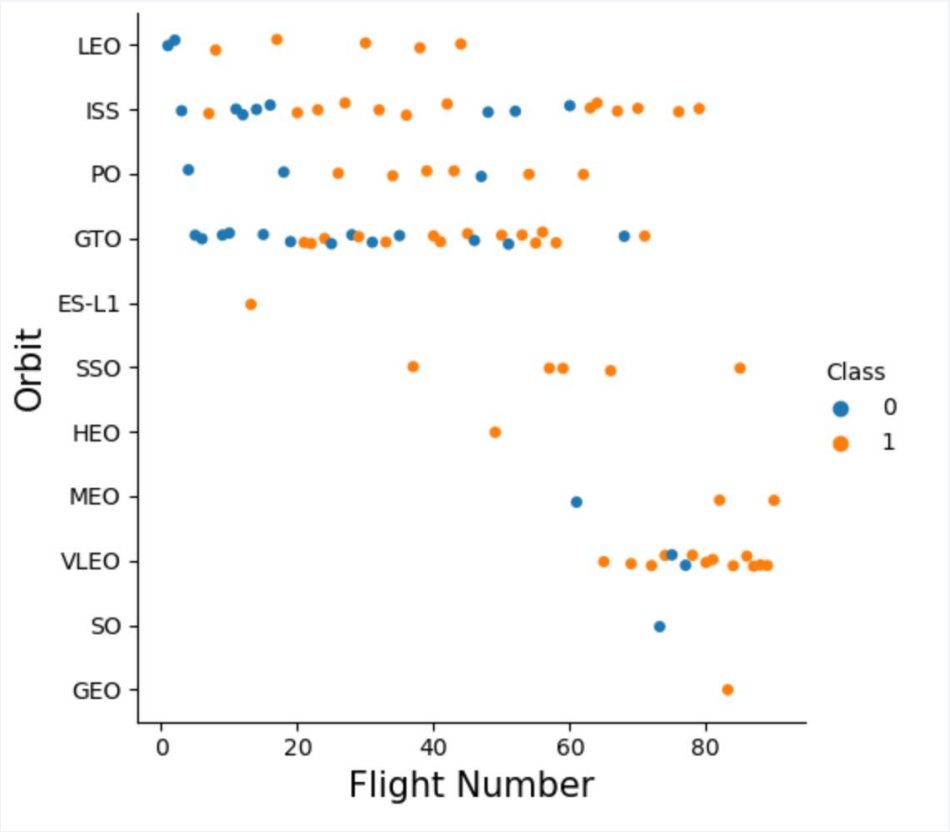
We can see that CCAFS SLC 40 has higher success rate at PLM of 14000-16000 while KSC LC 39A has higher success rate at PLM of 2000-6000

Success Rate vs. Orbit Type



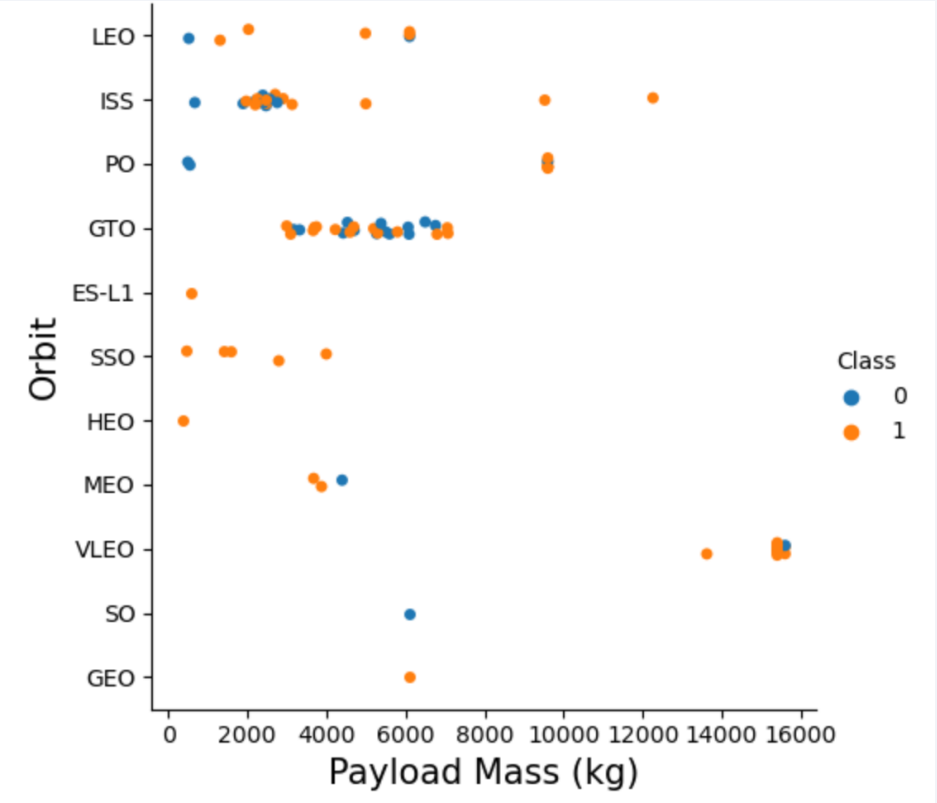
Orbits ES-L1, GEO, HEO, SSO have the highest success rate

Flight Number vs. Orbit Type



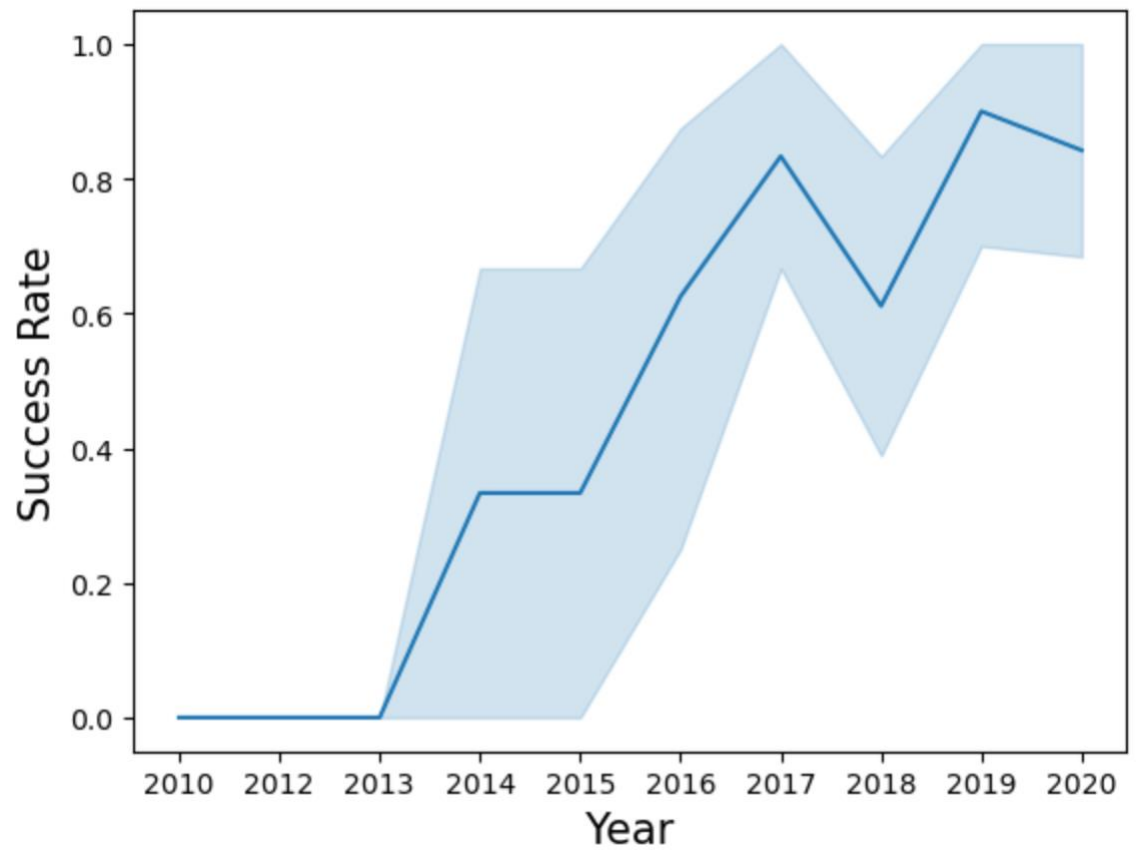
Orbits like LEO and PO have visible relation between success rate and number of flights.

Payload vs. Orbit Type



Success rate is higher with heavy payloads for LEO, ISS and PO

Launch Success Yearly Trend



Success rate has been increasing during 2013-2017 and 2018-2019

All Launch Site Names

```
In [7]: %sql SELECT DISTINCT("Launch_Site") FROM SPACEXTABLE
 * sqlite:///my_data1.db
Done.
```

Out[7]:

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

By using SELECT DISTINCT() only unique values of a column will be shown

Launch Site Names Begin with 'CCA'

In [24]:
```
%sql SELECT * FROM SPACEXTABLE WHERE "Launch_Site" LIKE '%CCA%' LIMIT 5
```

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

Out[24]:

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

By using method LIKE “” we can filter values by their filling. LIMIT means that only a certain number of first rows will be sown.

Total Payload Mass

Display the total payload mass carried by boosters launched by NASA (CRS)

```
%sql SELECT SUM(PAYLOAD_MASS__KG_) AS 'Total payload mass' FROM SPACEXTABLE
```

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

Total payload mass
619967

Method SUM() will summarize all values in a column

Average Payload Mass by F9 v1.1

Display average payload mass carried by booster version F9 v1.1

```
%sql SELECT AVG(PAYLOAD_MASS__KG_) AS 'Average payload mass F9 v1.1' FROM SPACEXTABLE WHERE Booster_Version LIKE 'F9 v1.1%'
```

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

Average payload mass F9 v1.1
2534.6666666666665

Method AVG() will calculate the average of values in a column

First Successful Ground Landing Date

```
%sql SELECT MIN(Date) AS 'Date of first successful landing' FROM SPACEXTABLE WHERE Landing_Outcome LIKE 'Success%' LIMIT 1
```

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

Date of first successful landing
2015-12-22

Method MIN() will find the minimal value in a column

Successful Drone Ship Landing with Payload between 4000 and 6000

List the names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000

In [67]:
```
%sql SELECT DISTINCT(Booster_Version) FROM SPACEXTABLE WHERE  PAYLOAD_MASS__KG_ BETWEEN 4000 AND 6000 AND Landing_Outcome
```

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

Out[67]:

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

Method WHERE [COLUMN] BETWEEN *int1* AND *int2* will only shown rows where values in a [COLUMN] are in range *int1* and *int2*

Total Number of Successful and Failure Mission Outcomes

```
In [66]: %sql SELECT COUNT(Mission_Outcome) FROM SPACEXTABLE
 * sqlite:///my_data1.db
Done.
Out[66]: COUNT(Mission_Outcome)
         101
```

Method COUNT() will count the number of rows

Boosters Carried Maximum Payload

```
In [70]: %sql SELECT DISTINCT(Booster_Version) FROM SPACEXTABLE WHERE PAYLOAD_MASS__KG_ IN (SELECT MAX(PAYLOAD_MASS__KG_) FROM SPACEX
 * sqlite:///my_data1.db
Done.
```

Out[70]:

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

In this query we used subquery to obtain information about max payload

2015 Launch Records

In [75]:
```
%sql SELECT SUBSTR(Date, 6, 2) AS 'Month', Landing_Outcome AS 'Landing Outcome', Booster_Version AS 'Booster version', Laun
```

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

Out[75]:

Month	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

In this query SUBSTR() was used to obtain month and year from DATE column

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

```
In [79]: %sql SELECT Landing_Outcome AS 'Landing Outcome', COUNT(*) AS 'Volume' FROM SPACEXTABLE WHERE Date BETWEEN '2010-06-04' AND
 * sqlite:///my_data1.db
Done.
```

Out[79]:

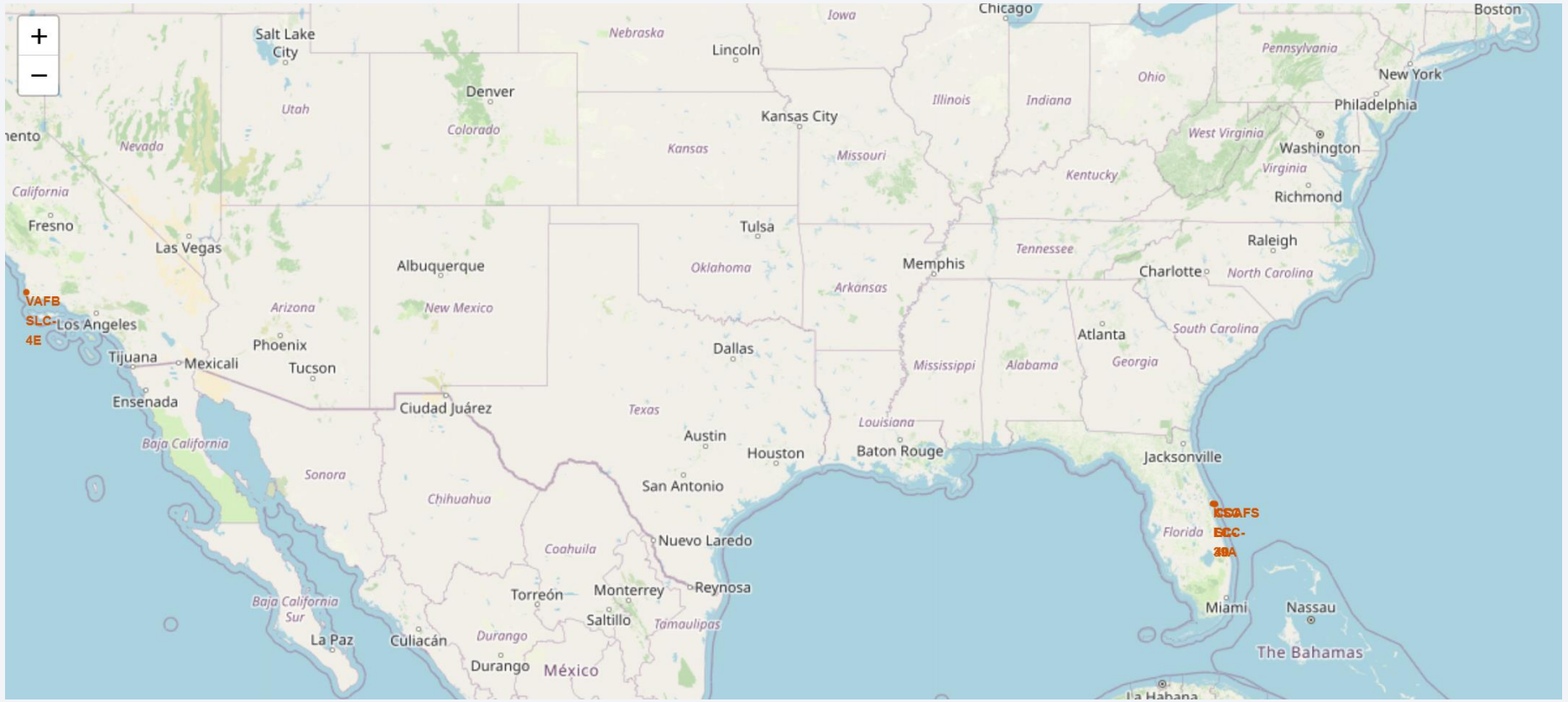
Landing Outcome	Volume
No attempt	10
Success (ground pad)	5
Success (drone ship)	5
Failure (drone ship)	5
Controlled (ocean)	3
Uncontrolled (ocean)	2
Precluded (drone ship)	1
Failure (parachute)	1

In order to show Volume column in descending order we used function ORDER BY [COLUMN] DESC

Section 3

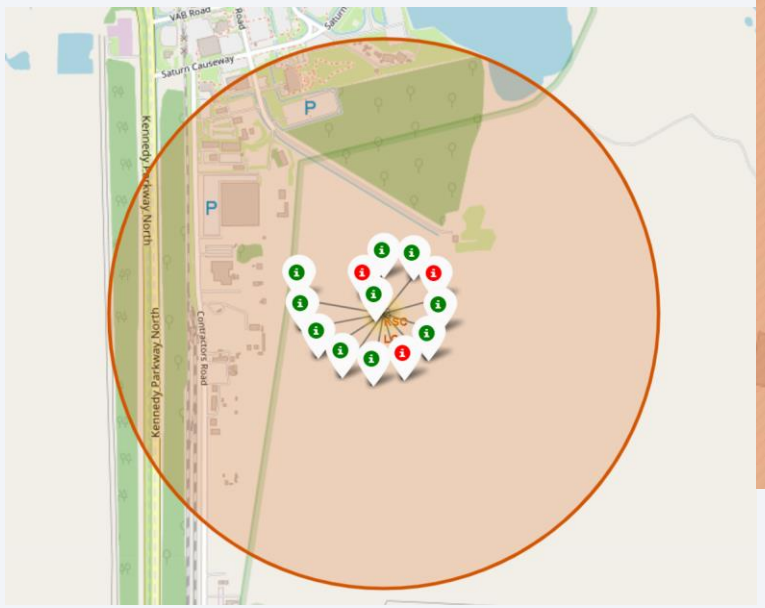
Launch Sites Proximities Analysis

Launch sites on a map

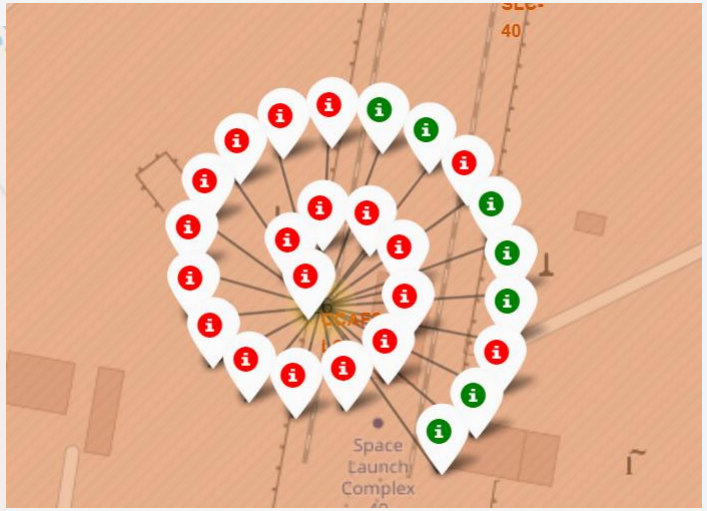


As we can see all of the launch sites are located on Merritt island and between Los Angeles and Fresno

Marking the success/failed launches for each site on the map

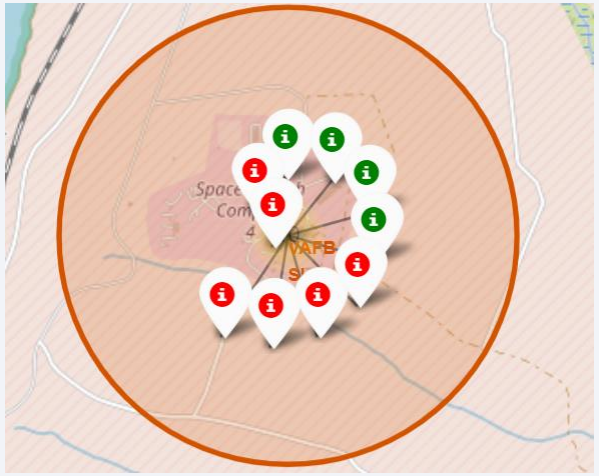


KSC LC 39-A



CCAFS LC-40

CCAFC SLC-40

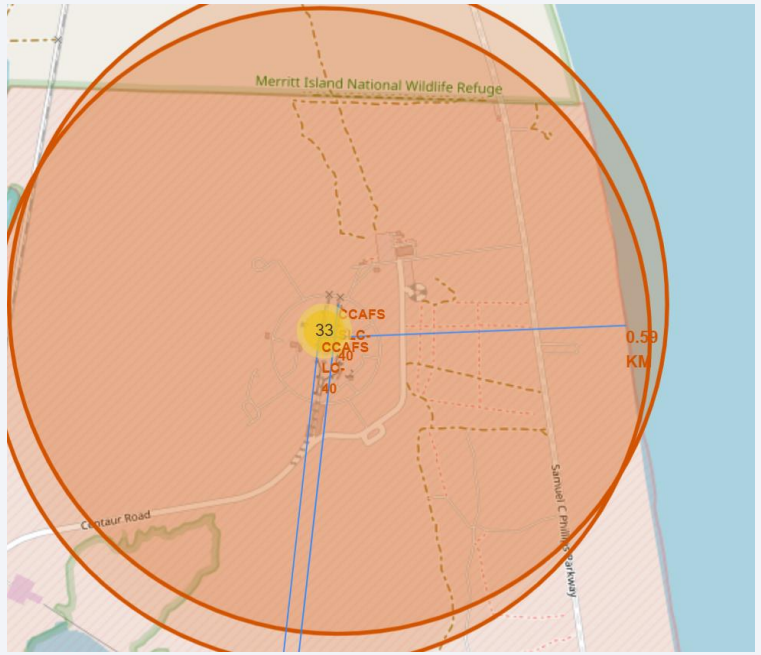


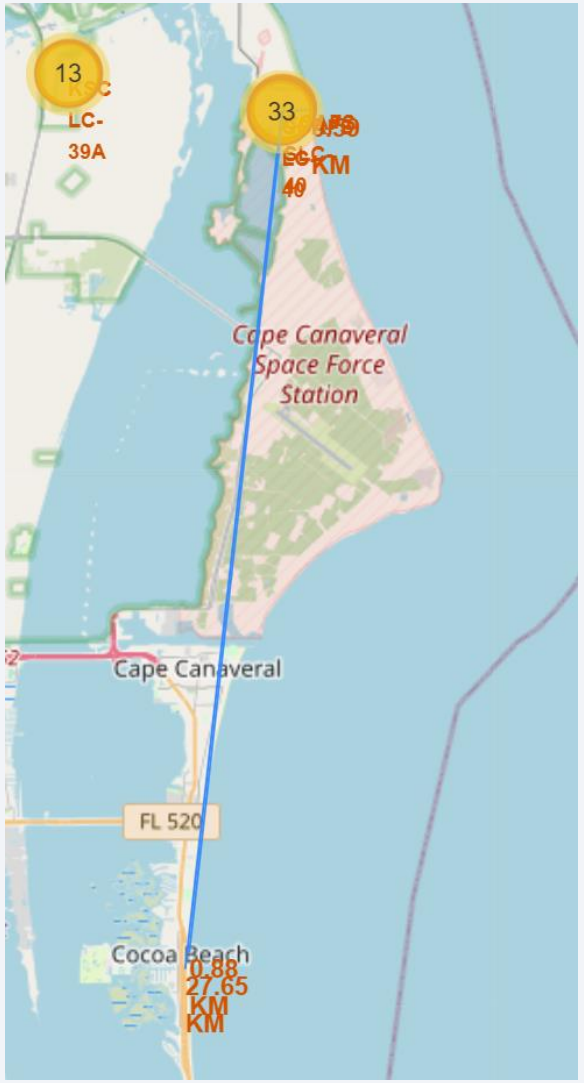
VAFB SLC 40-E

Green markers – successful launch/ red markers – failure

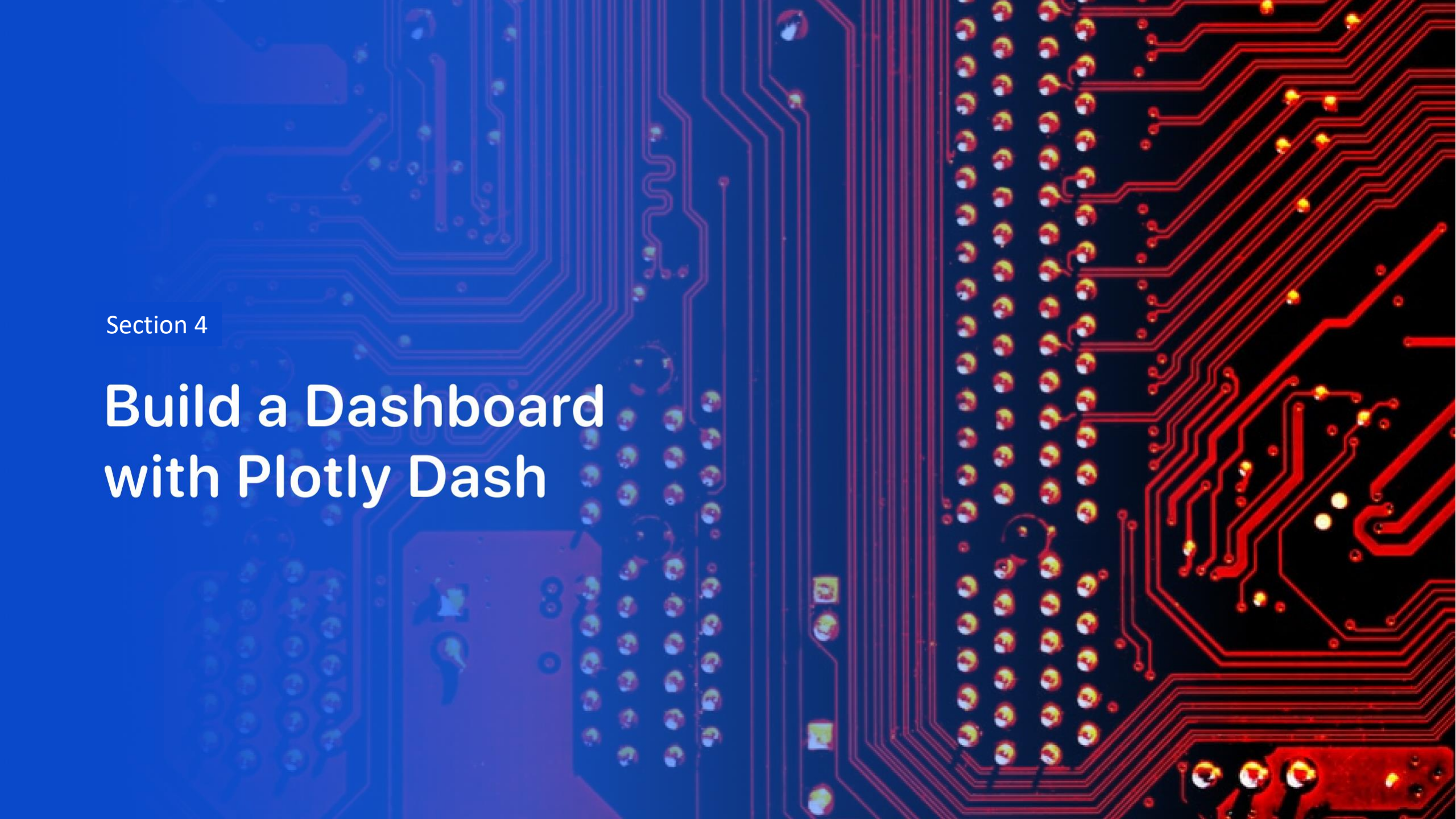
We can observer that KSC LC 39-A and CCAFS LC-40 have most launches. KSC LC 39-A has higher success rate Than CCAFS LC-40

Calculate the distances between a launch site to its proximities

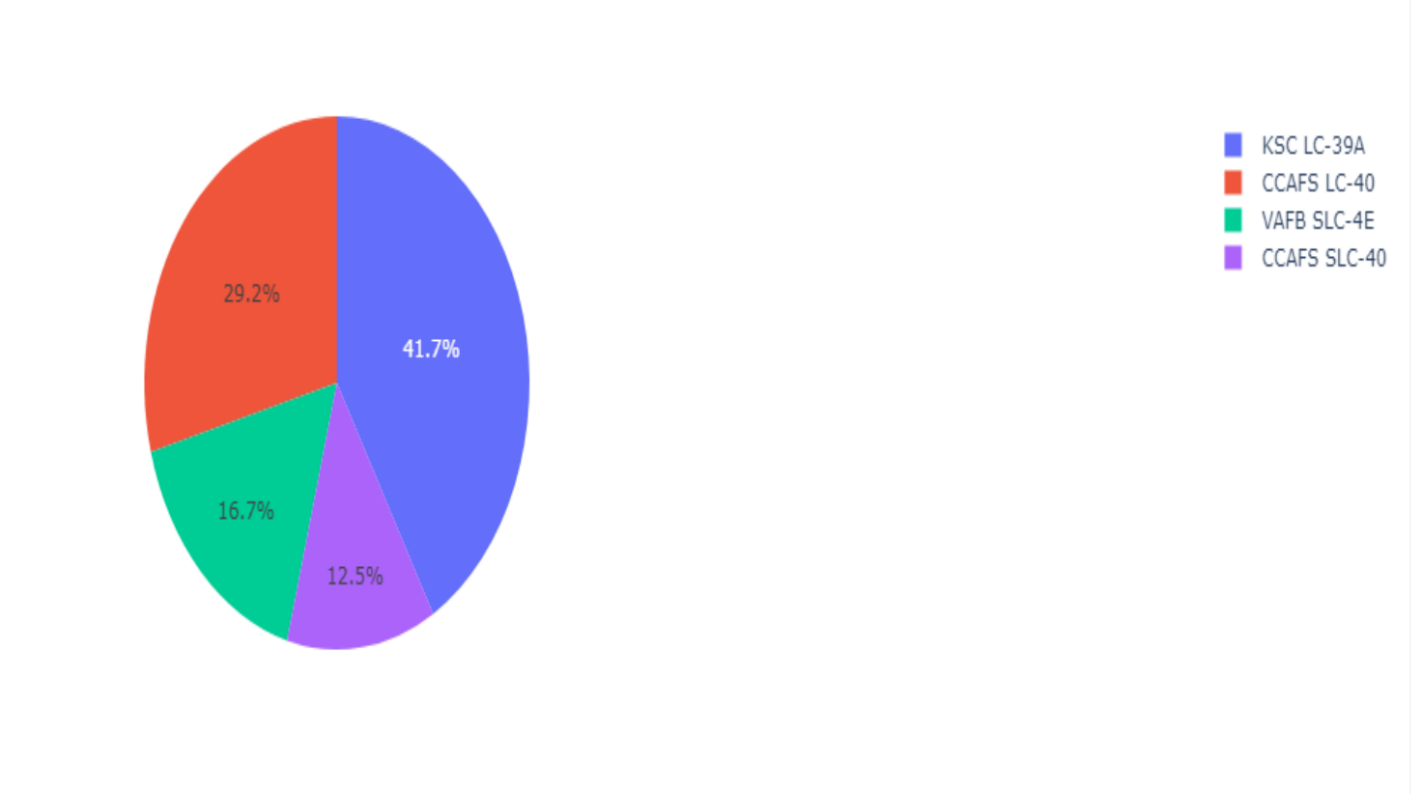




All of the launch sites are located near the coastline and railway
and far away from cities

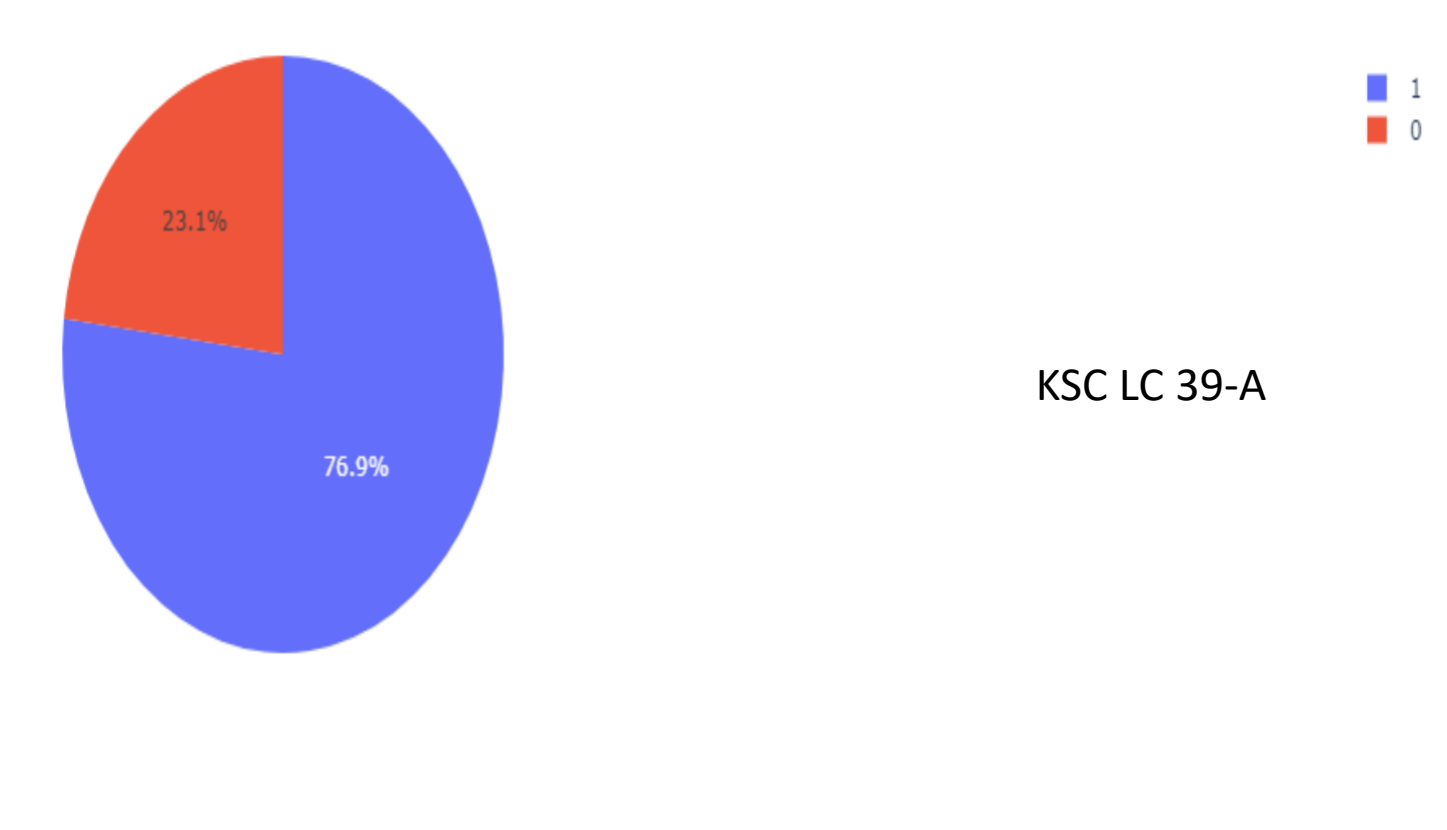

Section 4
Build a Dashboard with Plotly Dash

Total success launches for all sites



As we can observe, KSC LC-39A has the most success launches while CCAFS SLC-40 has the lowest

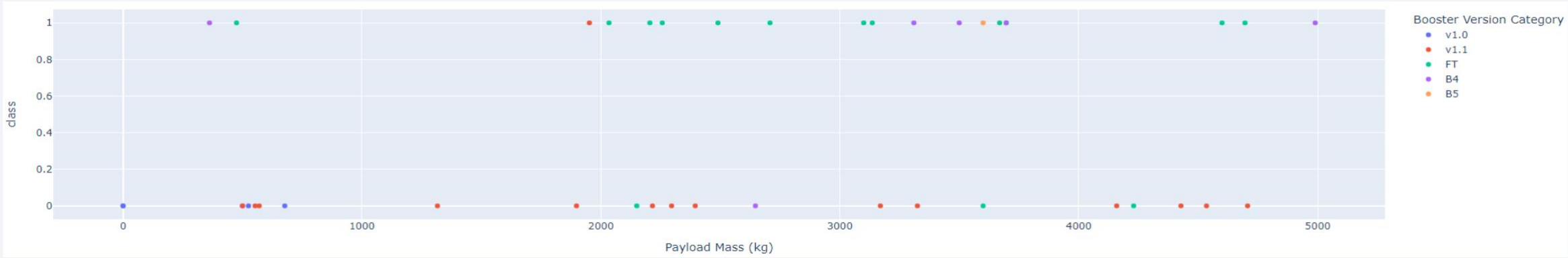
Launch site with highest success ratio



KSC LC 39-A has success rate of 76.9%

Payload vs. Launch Outcome scatter plot for all sites

0-5000 KG



5000-10000 KG

Success rate of low-weighted payloads (<5000KG) is much higher than for heavy-weighted (10000 KG)

Section 5

Predictive Analysis (Classification)

Classification Accuracy

```
logreg_cv.score(X_test, Y_test)
```

```
0.8333333333333334
```

```
svm_cv.score(X_test, Y_test)
```

```
0.8333333333333334
```

```
tree_cv.score(X_test, Y_test)
```

```
0.8333333333333334
```

```
knn_cv.score(X_test, Y_test)
```

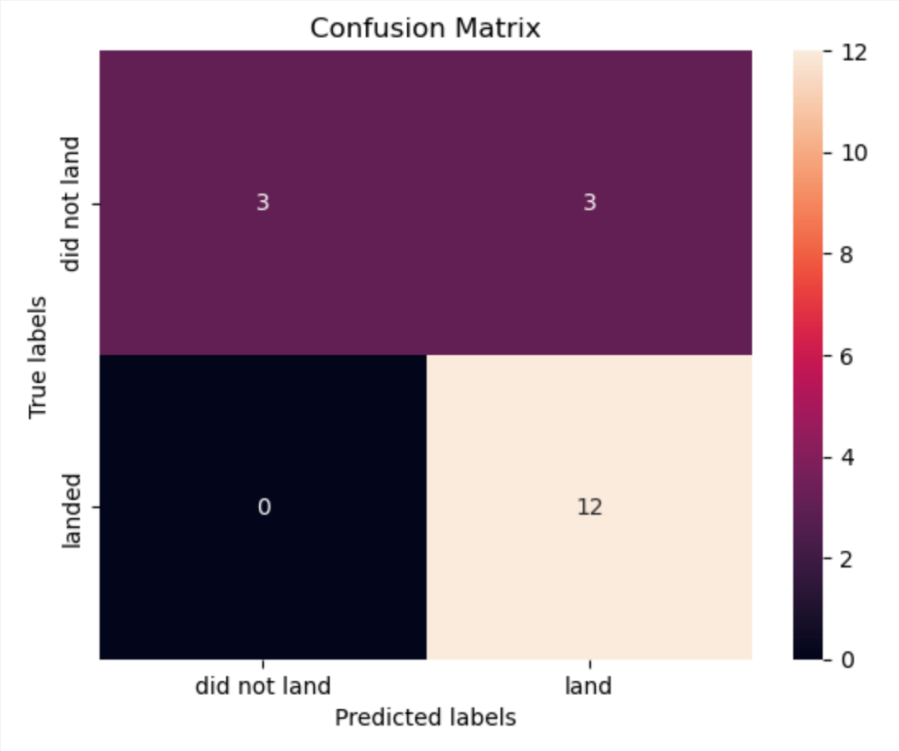
```
0.8333333333333334
```

```
print("tuned hpyerparameters :(best parameters) ",tree_cv.best_params_)
print("accuracy :",tree_cv.best_score_)
```

```
tuned hpyerparameters :(best parameters)  {'criterion': 'entropy', 'max_d
'random'}
accuracy : 0.8875
```

As we can see, all of the models (LR, SVM, DT, KNN) has near the same accuracy on the test data. But Decision Tree Has higher accuracy on training data than others

Confusion Matrix



		Predicted	
		Negative **(N)** -	Positive **(P)** +
Actual	Negative -	True Negative **(TN)**	**False Positive (FP) Type I Error**
	Positive +	**False Negative (FN) Type II Error**	True Positive **(TP)**

All of the models have the same confusion matrices. The main problem is False Positive

Conclusions

- KSC LC 39-A has the most successful launches compared to all of the other launch sites
- Orbits SSO, GEO, HEO and ES-L1 have the highest success rate
- Low-weighted payloads have much higher success rate than heavy-weighted
- Success rate for SpaceX increased greatly after 2013
- All of the ML models have the same accuracy rate on testing set (0.83). Decision Tree has the highest accuracy on training set of 0.8875

Appendix

Great online design platform CANVA

Official documentation for Scikit

Thank you!