IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

NGUYEN THUY DUONG
2023/12/20

Outline

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

Executive Summary

Summary of methodologies

Data Collection via APi, Data Wrangling

Exploratory Data analysis (EDA) with Data Visualization+ SQL

Visual Analytics and Dashboard with Folium

Machine Learning Prediction

Summary of all results

Exploratory Data Analysis result

Interactive analytics in screenshots

Predictive Analytics result

Introduction

Project background and context

The objective of this project is to predict the successful landing of the Falcon 9. The cost of launching a Falcon 9 rocket is $62 million, whereas launches by other providers can cost upwards of $165 million. SpaceX has the capability to reuse the first stage of the rocket. Hence, by determining the likelihood of the first stage landing successfully, we can estimate the cost of a launch. This information can be beneficial for companies looking to reduce costs.

https://spacenews.com/spacexs-new-price-chart-illustrates-performance-cost-of-reusability/

Problems you want to find answers

1. The factors determine if the rocket will land successfully
2. The interaction amongst various features that determine the success rate of a successful landing.
3. Operating conditions that needs to be in place to ensure a successful landing.

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:

    Using SpaceX REST API

- Perform data wrangling

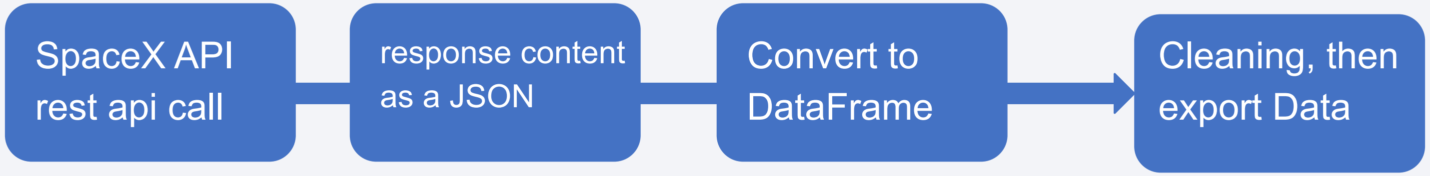
    Filtering, Handling missing value to prepare data for analysis

- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models

    Tune , Evaluate models to find the best model+parameters

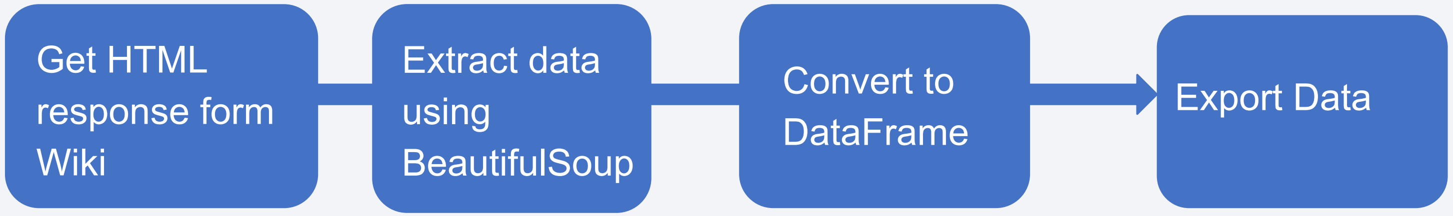
Data Collection

Collecting data using GET requests to the SpaceX API.



Web scraping from Wikipedia for Falcon 9 launch records with BeautifulSoup.

url: https://en.wikipedia.org/wiki/List_of_Falcon_9_and_Falcon_Heavy_launches



Data Collection – SpaceX API

- Use get request to the SpaceX API to collect data, then turn into a pandas dataframe

- The [GitHub URL](#)of the completed SpaceX API calls notebook

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

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

Check the content of the response

```
static_json_url='https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBM-DS0321EN-SkillsNe
```

We should see that the request was successfull with the 200 status response code

```
response.status_code
```

```
200
```

Now we decode the response content as a Json using `.json()` and turn it into a Pandas dataframe using `.json_normalize(`

```
# Use json_normalize meethod to convert the json result into a dataframe
res = requests.get(static_json_url)
print(res.content)
static_json_df = res.json()
data = pd.json_normalize(static_json_df)
```

b'[{"fairings": {"reused": false, "recovery_attempt": false, "recovered": false, "ships": []}, "links": {"pa
{"small": "https://images2.imgbox.com/3c/0e/T8iJcSN3_o.png", "large": "https://images2.imgbox.com/40/e3/Gyp
o.png"}, "reddit": {"campaign": null, "launch": null, "media": null, "recovery": null}, "flickr": {"small":

Data Collection - Scraping

- Applied web scraping to scrape Falcon 9 launch records with BeautifulSoup.
- Parsed the table and converted it into a pandas dataframe.

- The [GitHub URL]() of the completed web scraping notebook

```
static_url = "https://en.wikipedia.org/w/index.php?title=List_of_Falcon_9_and_Falcon_Heavy_launches&oldid=102768
```

Next, request the HTML page from the above URL and get a `response` object

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

```
200
```

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

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

Print the page title to verify if the `BeautifulSoup` object was created properly

```
# Use soup.title attribute
soup.title
```

```
<title>List of Falcon 9 and Falcon Heavy launches - Wikipedia</title>
```

Data Wrangling

How data were processed

In the dataset, there are several cases where the booster did not land successfully. We need to categorize variables: 1 for missions that successfully landed, and 0 for failures.

Calculating the number of launches for each site, occurrence of each orbit type.
Creating a landing outcome label from the outcome column, then export CSV file.

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

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

```
# Apply value_counts on Orbit column
df['Orbit'].value_counts()
```

```
GTO     27
ISS     21
```

```
# landing_outcomes = values on Outcome column
landing_outcomes = df['Outcome'].value_counts()
landing_outcomes
```

```
True ASDS      41
None None      19
True RTLS      14
```

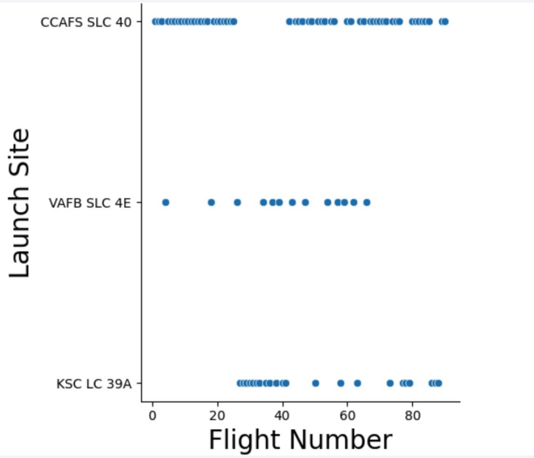
```
landing_class= df['Outcome'].apply(lambda x: 0 if x in bad_outcomes else 1)
landing_class.value_counts()
```

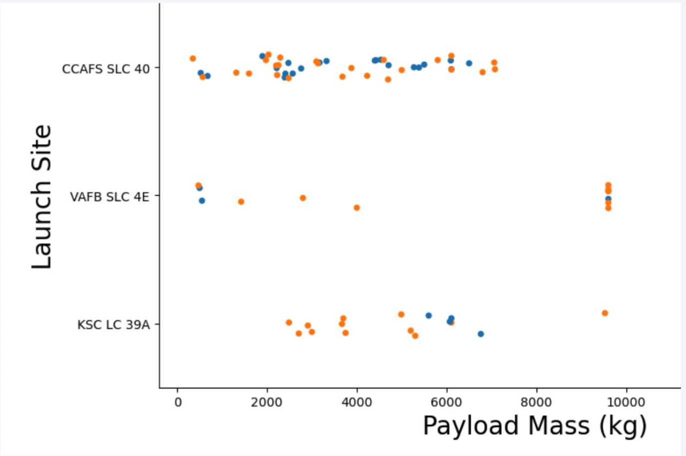
```
1    60
0    30
Name: Outcome, dtype: int64
```

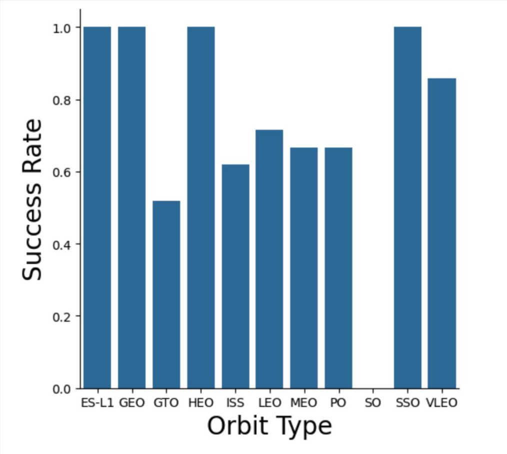
[The GitHub URL]

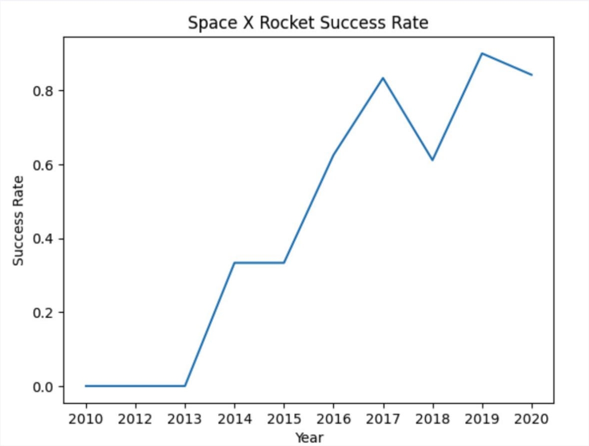
EDA with Data Visualization

- Explored the data by visualizing the relationship between data : flight number vs launch site, payload vs launch site, success rate of each orbit type, flight number vs orbit type, launch success vs yearly trend









[the GitHub URL]

EDA with SQL

SQL query tasks:

1. Displaying the names of the unique launch sites in the space mission.
2. Displaying 5 records where launch sites begin with the string 'CCA'.
3. Displaying the total payload mass carried by boosters launched by NASA (CRS).
4. Displaying the average payload mass carried by booster version F9 v1.1.
5. Listing the date of the first successful landing outcome on a ground pad.
6. Listing the names of boosters successful in drone ship landings with a payload mass between 4000 and 6000 kg.
7. Listing the total number of successful and failure mission outcomes.
8. Listing the names of booster versions that have carried the maximum payload mass.
9. Listing records showing the month names, failure landing outcomes, booster versions, and launch sites for the months in the year 2015.
10. Ranking the count of successful landing outcomes between the dates April 6, 2010, and March 20, 2017, in descending order.

[the GitHub URL]

Build an Interactive Map with Folium

▪All launch sites are marked on the Folium map with visual objects: markers, circles, and lines to indicate the success or failure of launches at each site.

▪The launch outcomes feature has been assigned1 for missions that successfully landed, and 0 for failures.

▪Launch sites with relatively high success rates have been identified and highlighted using color-labeled marker clusters.

▪Calculate distances between launch sites and their proximities, addressing questions about their proximity to railways, highways, coastlines, and cities.

The GitHub url:
https://github.com/duonglvjj/AppliedDataScienceCapstone/blob/bf66817fb6b8dae18f03c961e4e2590deea6f99b/Analysis%20with%20Folium.ipynb

Build a Dashboard with Plotly Dash

- Built an interactive dashboard with Plotly dash
- Plotted pie charts showing the total launches by a certain sites
- Plotted scatter graph showing the relationship with Outcome and Payload_Mass(Kg) for each booster version

The GitHub url:

Predictive Analysis (Classification)

- Summarize how you built, evaluated, improved, and found the best performing classification model
- You need present your model development process using key phrases and flowchart
- Add the GitHub URL of your completed predictive analysis lab, as an external reference and peer-review purpose

Results

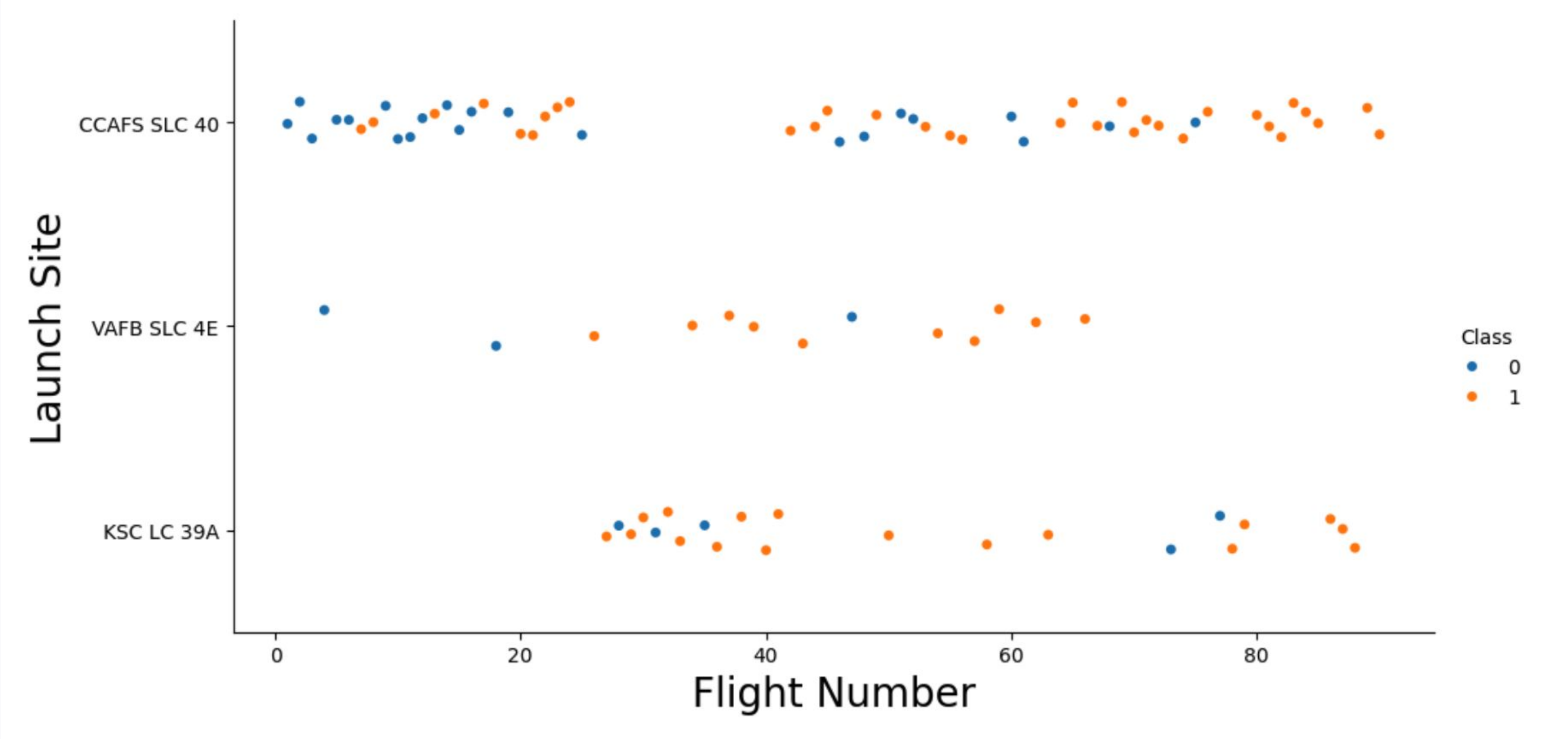
- Exploratory data analysis results
- Interactive analytics demo in screenshots
- Predictive analysis results

Section 2

Insights drawn from EDA

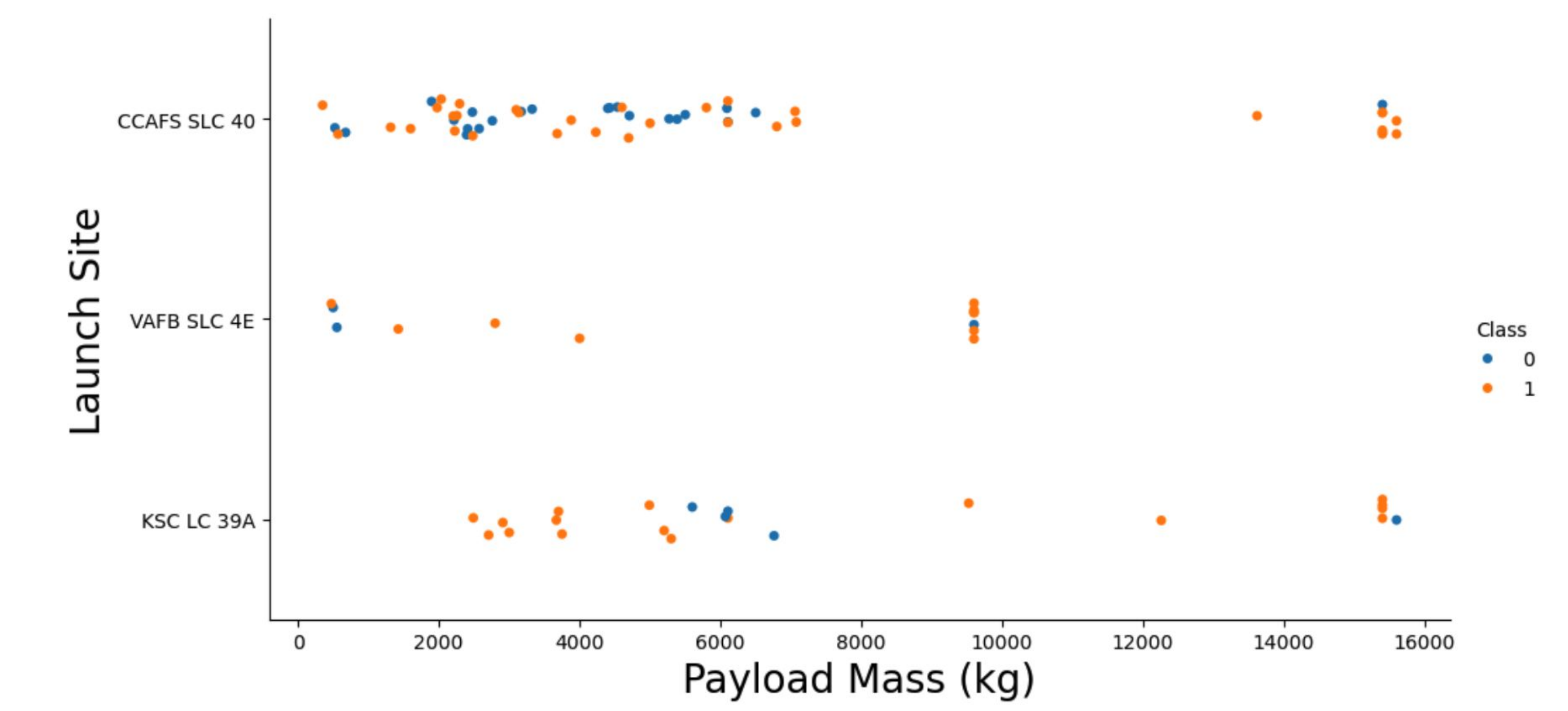
Flight Number vs. Launch Site

The larger the flight amount at a lunch site, the greater the success rate at a launch site



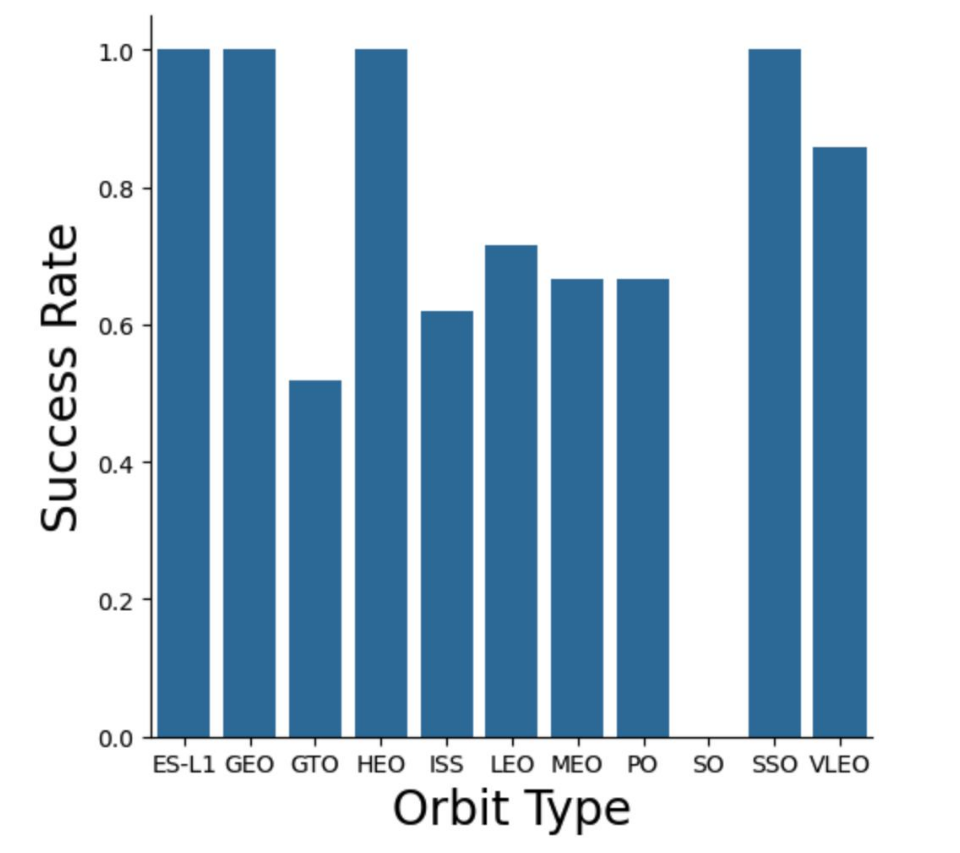
Payload vs. Launch Site

The greater the payload mass for launch site CCAFS SLC 40 the higher the success rate of the rocket



Success Rate vs. Orbit Type

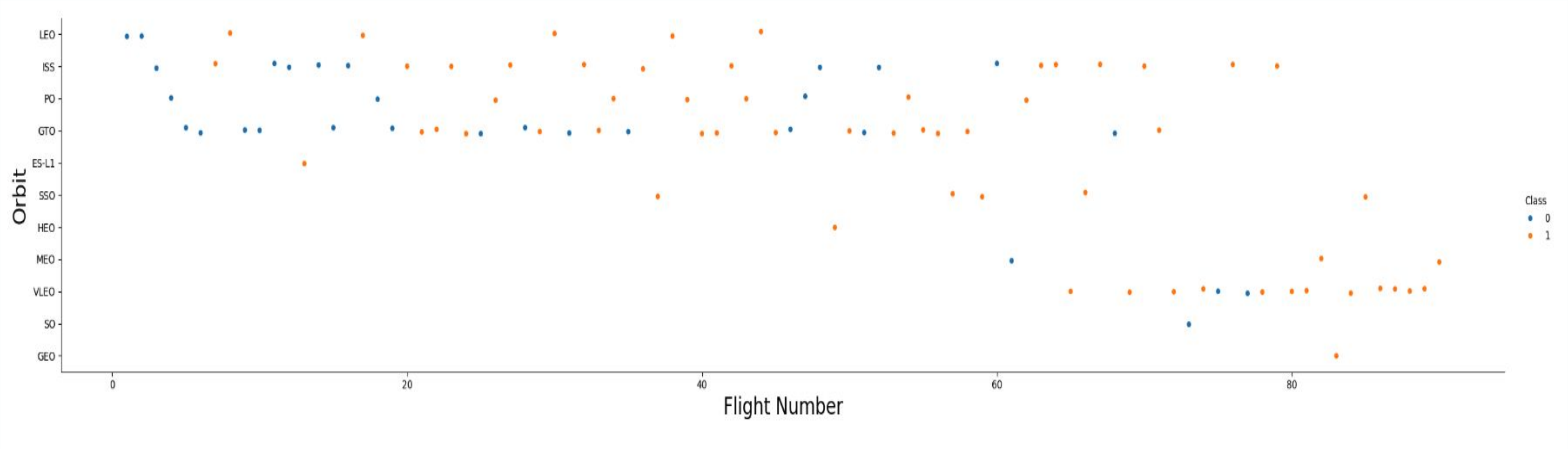
ES-L1, GEO, HEO,SSO,VLEO had the most success rate



Flight Number vs. Orbit Type

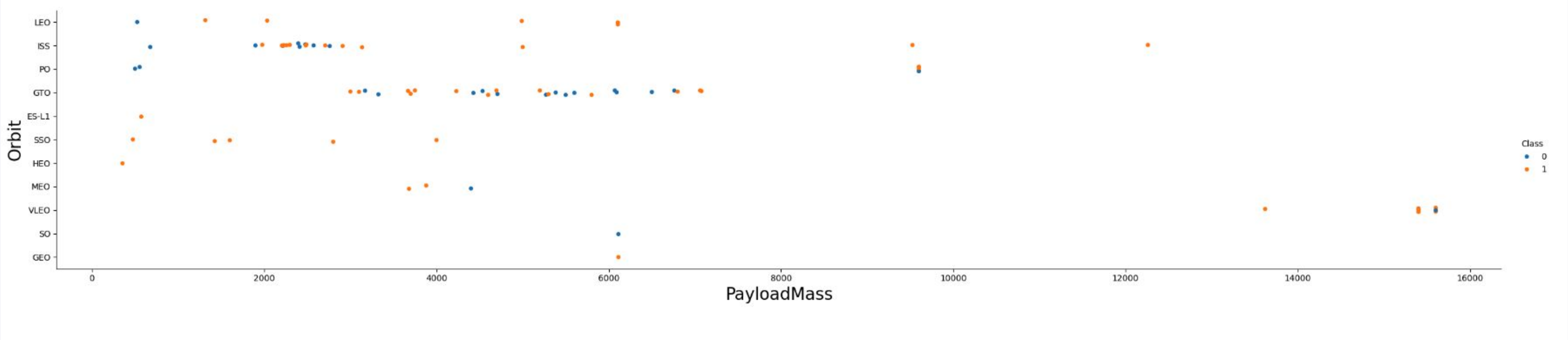
In the LEO orbit, success is related to the number of flights

In the GTO orbit, there is no relationship between flight number and the orbit



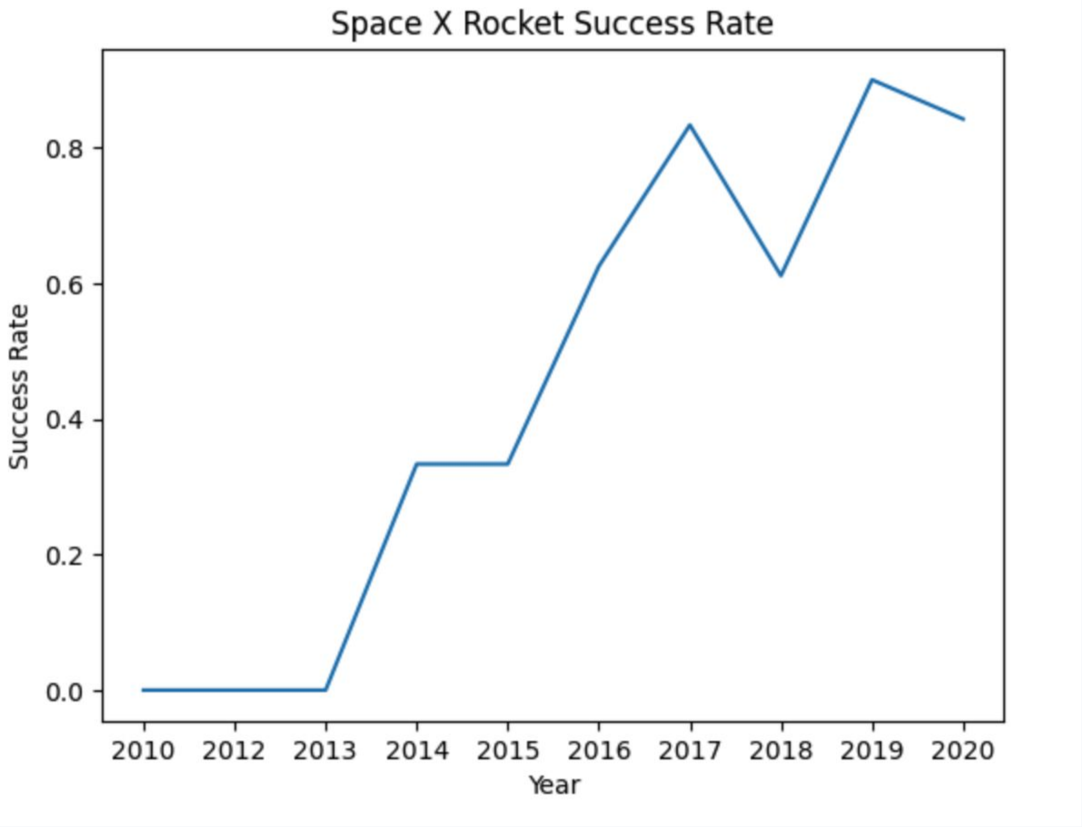
Payload vs. Orbit Type

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



Launch Success Yearly Trend

The sucess rate since 2013 kept increasing till 2020



All Launch Site Names

Use DISTINCT

```
[SQL: SELECT Unique(LAUNCH_SITE) FROM SPACEXTBL;]
(Background on this error at: http://sqlalche.me/e/13/e3q8
```

```
%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

Task 2

Launch Site Names Begin with 'CCA'

Display 5 records where launch sites begin with the string 'CCA'

Task 2

Display 5 records where launch sites begin with the string '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	L
2010-06-04	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success	
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	
2012-05-22	7:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success	
2012-10-08	0:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500	LEO (ISS)	NASA (CRS)	Success	
2013-03-01	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677	LEO (ISS)	NASA (CRS)	Success	

Total Payload Mass

- Calculate the total payload carried by boosters from NASA

```
%sql SELECT SUM(PAYLOAD_MASS__KG_) FROM SPACEXTBL WHERE Customer LIKE 'NASA (CRS)';
```

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

SUM(PAYLOAD_MASS__KG_)
45596

Average Payload Mass by F9 v1.1

Calculate the average payload mass carried by booster version F9 v1.1

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

```
%sql SELECT avg(PAYLOAD_MASS__KG_) from SPACEXTBL where Booster_Version LIKE 'F9 v1.1%';
```

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

avg(PAYLOAD_MASS__KG_)
2534.666666666665

First Successful Ground Landing Date

Find the dates of the first successful landing outcome on ground pad

Hint:Use min function

```
%sql SELECT min(date) FROM SPACEXTBL where Mission_Outcome = 'Success' and Landing_Outcome like '%ground pad%'
```

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

min(date)
2015-12-22

Successful Drone Ship Landing with Payload between 4000 and 6000

List the 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 SPACEXTBL
    WHERE Landing_Outcome like '%drone%' and PAYLOAD_MASS__KG_> 4000 and PAYLOAD_MASS__KG_ <6000
```

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

Booster_Version
F9 FT B1020
F9 FT B1022
F9 FT B1026
F9 FT B1021.2
F9 FT B1031.2

Total Number of Successful and Failure Mission Outcomes

Calculate the total number of successful and failure mission outcomes

```
]: %sql SELECT COUNT(*) FROM SPACEXTBL WHERE mission_outcome like '%success%'

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

COUNT(*)
100

```
]: %sql SELECT COUNT(*) FROM SPACEXTBL WHERE mission_outcome like '%failure%'

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

COUNT(*)
1

Boosters Carried Maximum Payload

List the names of the booster which have carried the maximum payload mass

```
%%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

List the failed landing_outcomes in drone ship, their booster versions, and launch site names for in year 2015

```
%%sql SELECT substr(Date,6,2) as month,DATE,Booster_Version,Launch_Site,Landing_Outcome FROM SPACEXTBL
    WHERE substr(Date,0,5)='2015'and Landing_Outcome like'%Failure%'
```

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

month	Date	Booster_Version	Launch_Site	Landing_Outcome
10	2015-10-01	F9 v1.1 B1012	CCAFS LC-40	Failure (drone ship)
04	2015-04-14	F9 v1.1 B1015	CCAFS LC-40	Failure (drone ship)

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

- 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

```
%%sql SELECT Landing_Outcome, COUNT(Landing_Outcome)AS TOTAL FROM SPACEXTBL
    WHERE DATE BETWEEN '2010-06-04' AND '2017-03-20' GROUP BY Landing_Outcome ORDER BY TOTAL DESC
```

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

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

Section 3

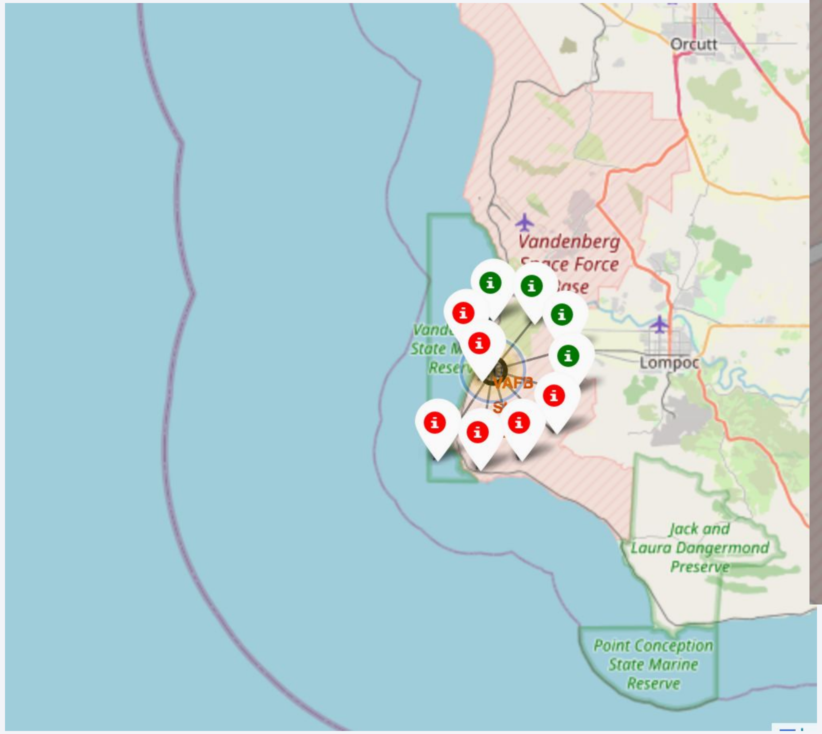
Launch Sites Proximities Analysis

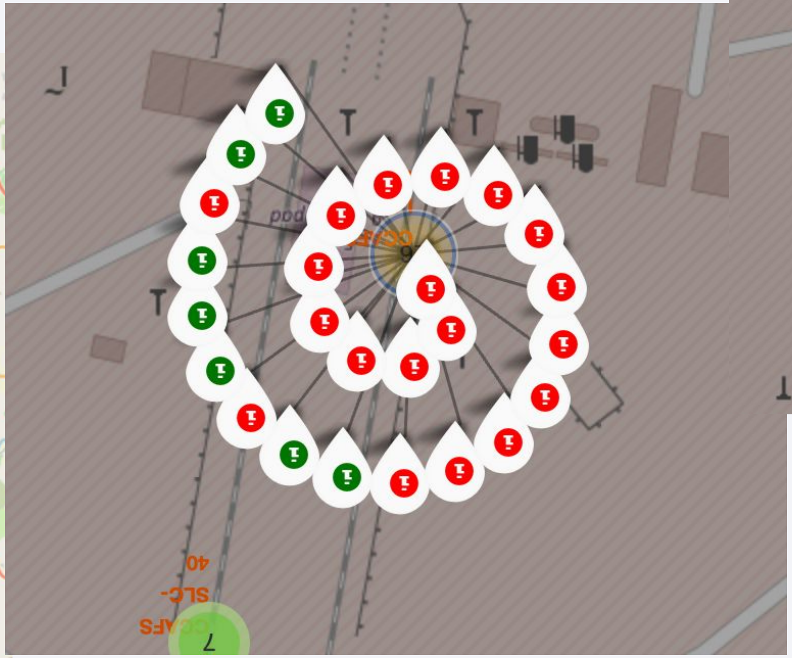
Global map

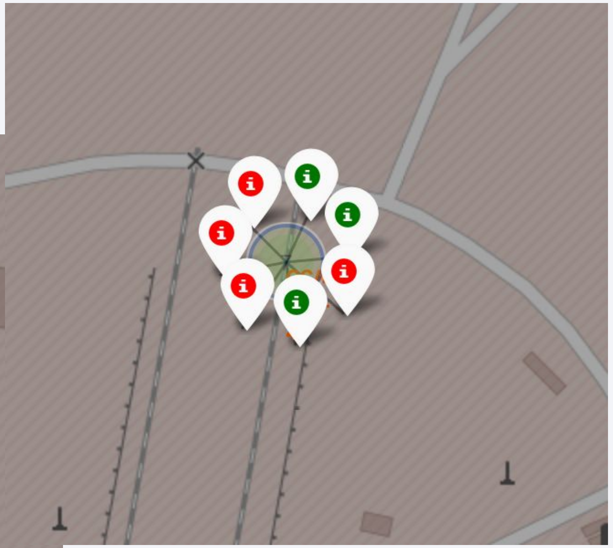
The SpaceX launch_sites are in US : Florida, CA

Launche sites with color

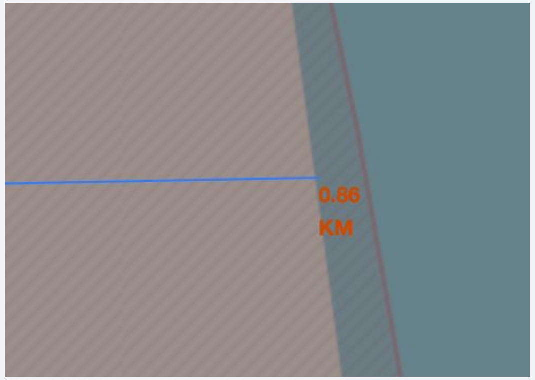
Green is successful launches, Red is Failures

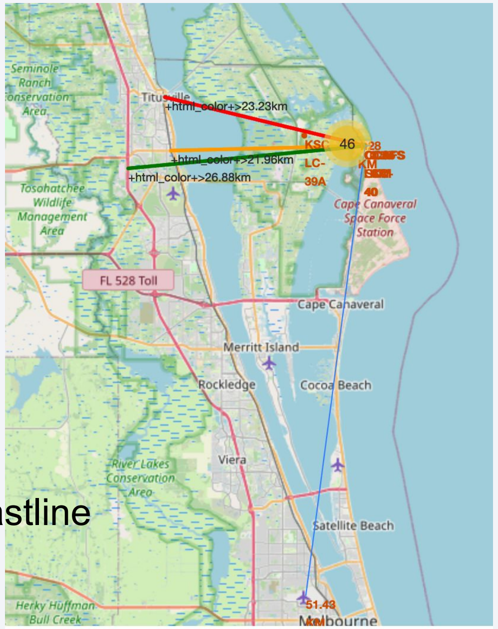


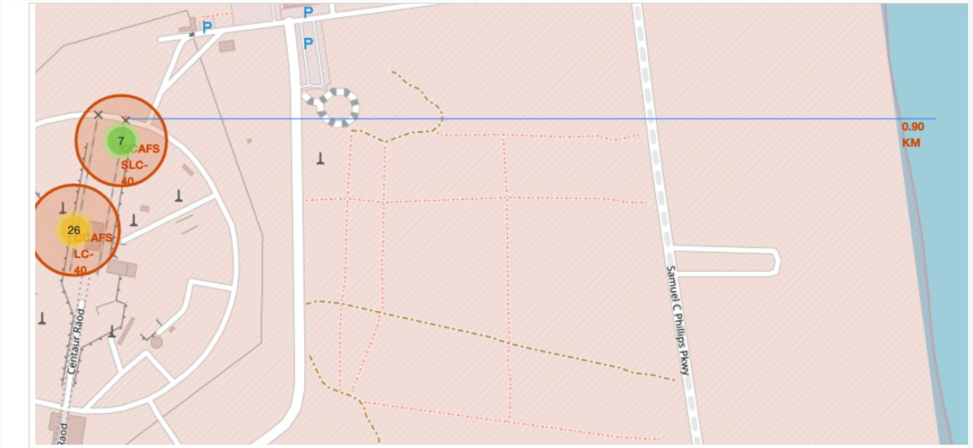


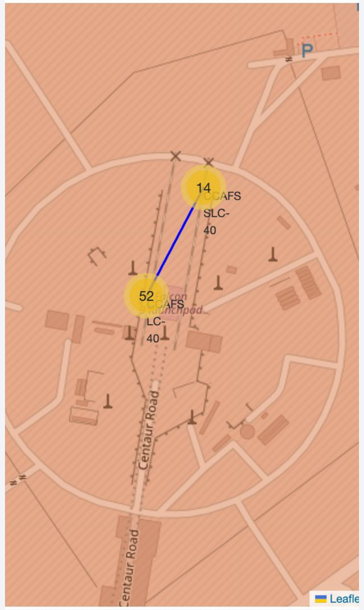


Launch site distance to landmark









Distance to coastline

Distance between highway

Distance to coast

After you plot distance lines to the proximities, you can answer the followi

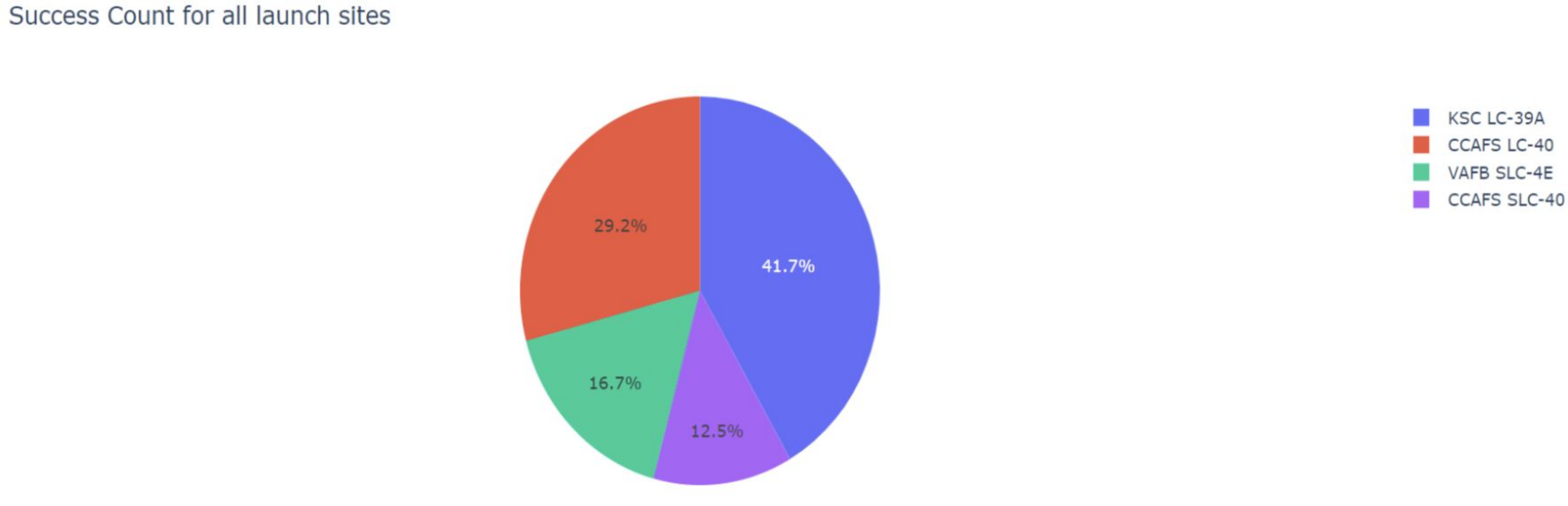
- Are launch sites in close proximity to railways? -> NO
- Are launch sites in close proximity to highways? -> NO
- Are launch sites in close proximity to coastline? -> YES
- Do launch sites keep certain distance away from cities? ->YES

Section 4

Build a Dashboard with Plotly Dash

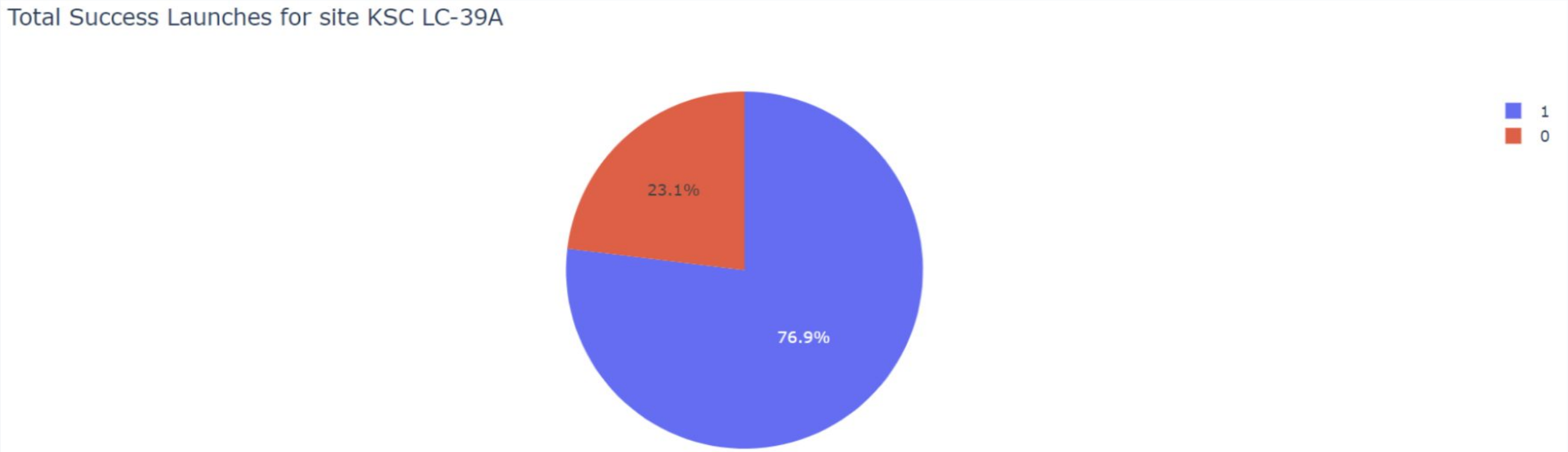
Success count for all sites

The KSC LC-39A has the most successful launches



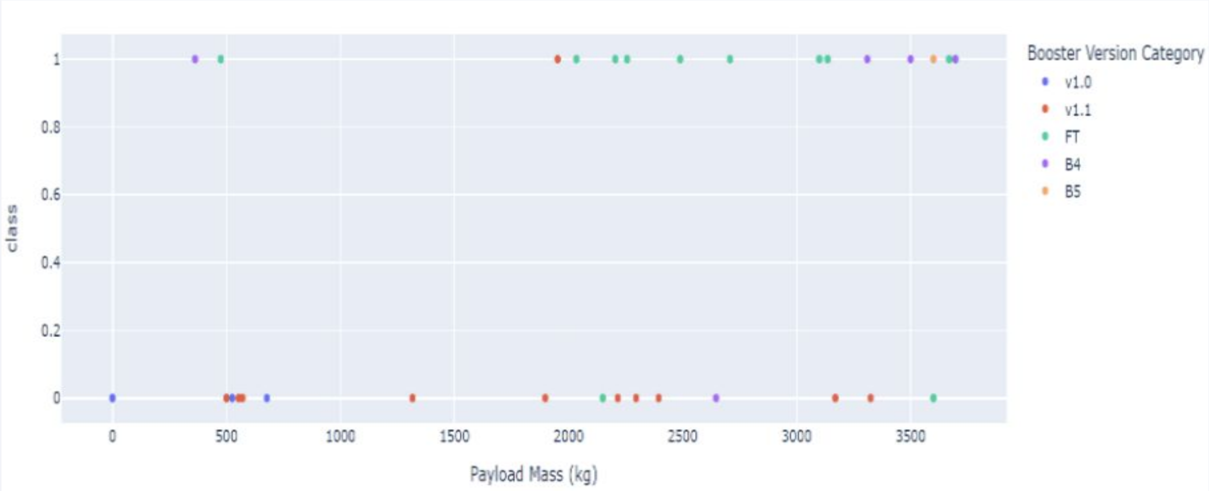
Launche site with the highest success ratio

The KSC LC-39A has 76.9% success rate, and 23.1% failure rate

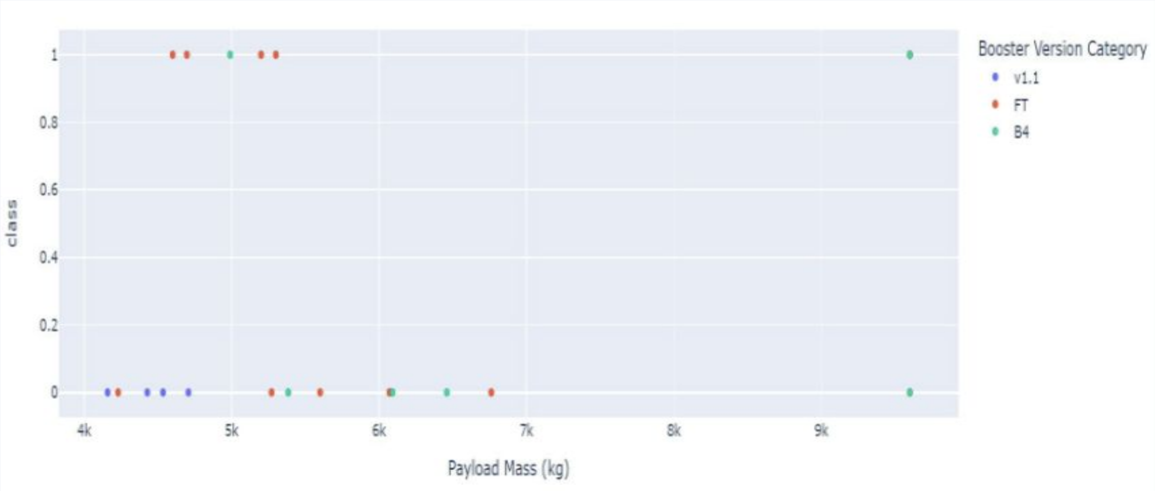


Payload vs Launch outcome

The success rate of pay payload 0~4000kg is higher than pay load 4000~10000kg



payload 0~4000kg



payload 4000~10000kg

Section 5

Predictive Analysis (Classification)

Classification Accuracy

Best model is DecisionTree with a score of 0.8607142857142858

Find the method performs best:

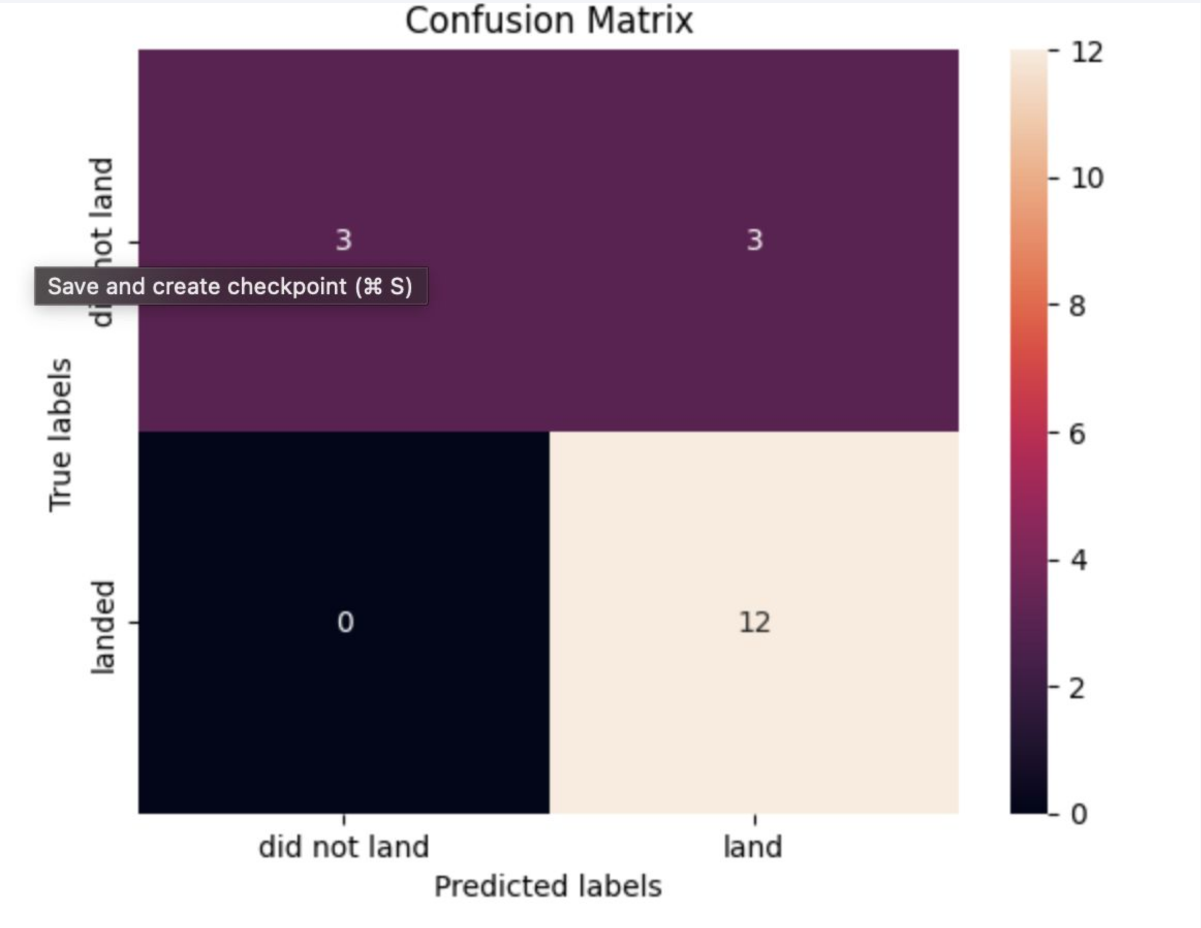
```
]:
models = {'KNeighbors':knn_cv.best_score_,
              'DecisionTree':tree_cv.best_score_,
              'LogisticRegression':logreg_cv.best_score_,
              'SupportVector': svm_cv.best_score_}

bestalgorithm = max(models, key=models.get)
print('Best model is', bestalgorithm,'with a score of', models[bestalgorithm])
if bestalgorithm == 'DecisionTree':
    print('Best params is :', tree_cv.best_params_)
if bestalgorithm == 'KNeighbors':
    print('Best params is :', knn_cv.best_params_)
if bestalgorithm == 'LogisticRegression':
    print('Best params is :', logreg_cv.best_params_)
if bestalgorithm == 'SupportVector':
    print('Best params is :', svm_cv.best_params_)
```

```
Best model is DecisionTree with a score of 0.8607142857142858
Best params is : {'criterion': 'gini', 'max_depth': 4, 'max_features': 'sqrt', 'min_samples_leaf': 2, 'min_samples_split': 10, 'spl
itter': 'random'}
```

Confusion Matrix

- The classifier can distinguish between different classes.
- The major issue is false positives; the model incorrectly predicted the 1st stage booster to land in 3 out of 18 samples in the test set.



Conclusions

1. The larger the number of flights at a launch site, the higher the success rate at that site.
2. The launch success rate began to improve from 2013 to 2020.
3. Orbits designated as ES-L1, GEO, HEO, SSO, and VLEO had the highest success rates.
4. KSC LC-39A had the most successful launches compared to other sites.
5. Decision tree classifier is the best machine learning algorithm for this task.

Appendix

Notebooks to recreate the dataset, perform the analysis, and develop the models:

https://github.com/duonglvjj/AppliedDataScienceCapstone.git

Coursera offer this course online:
https://www.coursera.org/professional-certificates/ibm-data-science

Thank you!