IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Noushin Naseri

Jan 2025

Outline

Executive Summary

Introduction

Methodology

Results

Conclusion

Executive Summary

Summary of methodologies

- Data collection from 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 result
- Interactive analysis in screenshots
- Predictive analytics result

Introduction

Project background and context

In this capstone, we will 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 determine if the rocket will land successfully?

The interaction amongst various features that determine the success rate of a successful landing.

What operating conditions needs to be in place to ensure a successful landing program.

Section 1

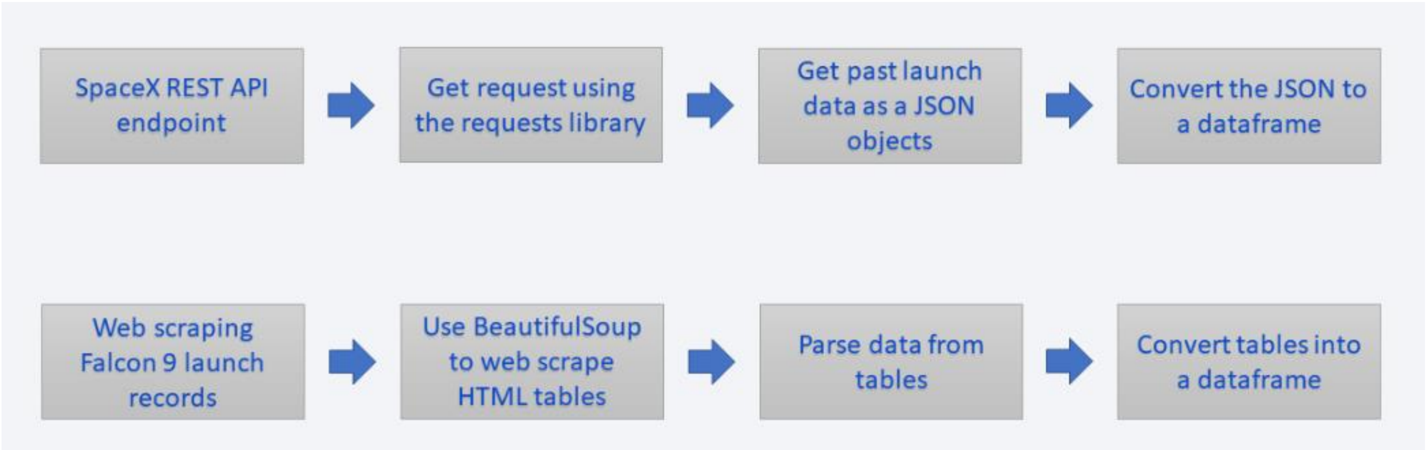
Methodology

Methodology

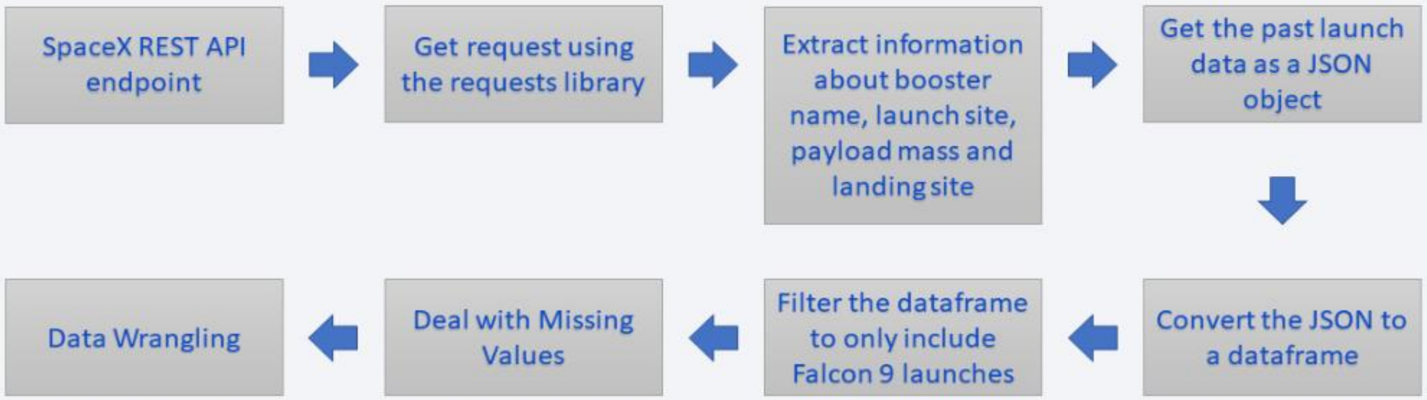
Executive Summary

- Data collection methodology:
    - Data was collected using SpaceX API and web scraping from Wikipedia
- Perform data wrangling
    - Data convert from Json and HTML to DF
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
    - Use of machine learning to determine if the first stage of Falcon 9 will land successfully

Data Collection

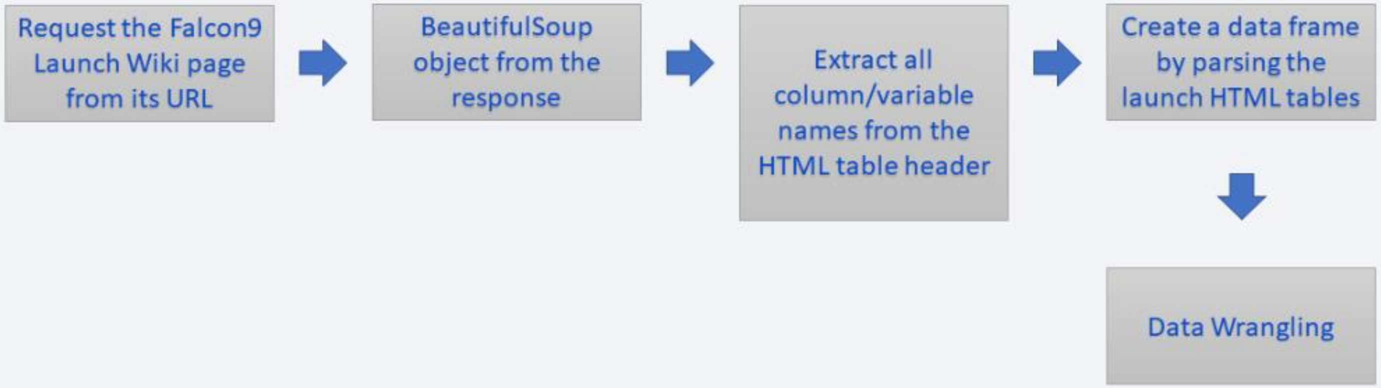


Data Collection – SpaceX API



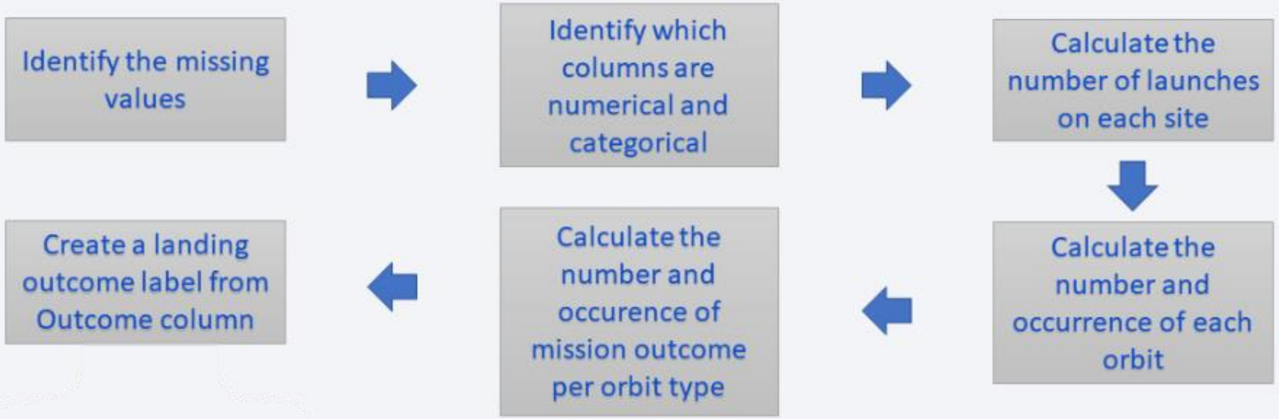
https://github.com/Noushin-Naseri/winning-space-race/blob/main/jupyter-labs-spacex-data-collection-api.ipynb

Data Collection - Scraping



https://github.com/Noushin-Naseri/winning-space-race/blob/main/jupyter-labs-webscraping.ipynb

Data Wrangling



https://github.com/Noushin-Naseri/winning-space-race/blob/main/labs-jupyter-spacex-Data%20wrangling.ipynb

EDA with Data Visualization

We EDA by visualizing the relationship between flight number and launch site, payload and launch site, success rate of each orbit type, flight number and orbit type, the launch success yearly trend.

https://github.com/Noushin-Naseri/winning-space-race/blob/main/edadataviz.ipynb

EDA with SQL

We applied EDA with SQL to get insight from the data. We wrote queries to find out for instance:

- ✓ The name of unique launch sites in the space mission
- ✓ The total payload mass carried by boosters launched by NASA
- ✓ The average payload mass carried by booster version F9 v1.1
- ✓ The total number of successful and failure mission outcomes
- ✓ The failed landing outcomes in drone ship, their booster version and launch sit names

https://github.com/Noushin-Naseri/winning-space-race/blob/main/jupyter-labs-eda-sql-coursera_sqllite.ipynb

Build an Interactive Map with Folium

01

We marked all launch sites, and added map objects such as markers, circles, lines to mark the success or failure of launches for each site on the folium map.

02

We assigned the feature launch outcomes to class 0 and 1

03

Using the color_labeled marker clusters, we identified which launch sites have relatively high success rate

04

We calculated the distances between a launch site to its proximities

Build a Dashboard with Plotly Dash

We built an interactive dashboard with Plotly dash

We plotted pie charts showing the total launches by a certain sites

We plotted scatter graph showing the relationship with outcome and payload mass for the different booster version

Predictive Analysis (Classification)

We loaded the data using numpy and pandas, transformed the data, split our data into training and testing

We built different machine learning models and tune different hyperparameters using GridSearchCV

We used accuracy as the metric for our model, improved the model using feature engineering and algorithm tuning

We found the best performing classification model

https://github.com/Noushin-Naseri/winning-space-race/blob/main/SpaceX_Machine%20Learning%20Prediction_Part_5.ipynb

Results

EXPLORATORY DATA ANALYSIS RESULTS

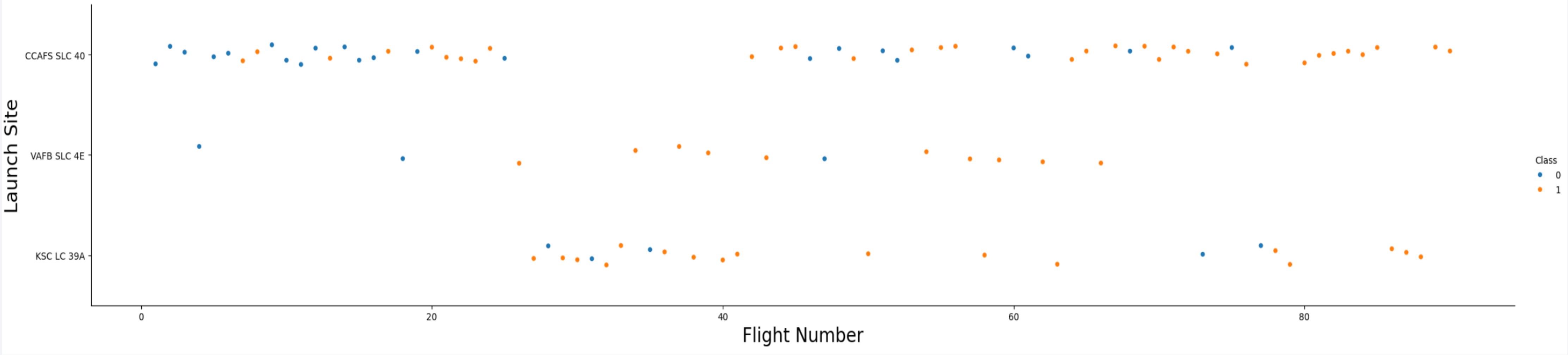
INTERACTIVE ANALYTICS DEMO IN SCREENSHOTS

PREDICTIVE ANALYSIS RESULTS

Section 2

Insights drawn from EDA

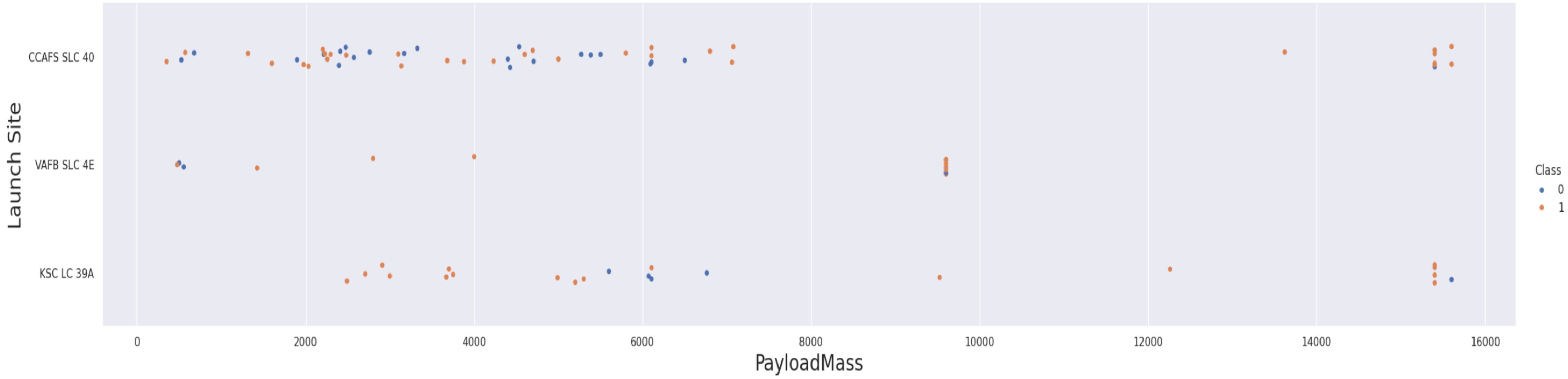
Flight Number vs. Launch Site



From the plot, we found that the larger the flight mount at a lunch site, the greater the success rate at a launch site.

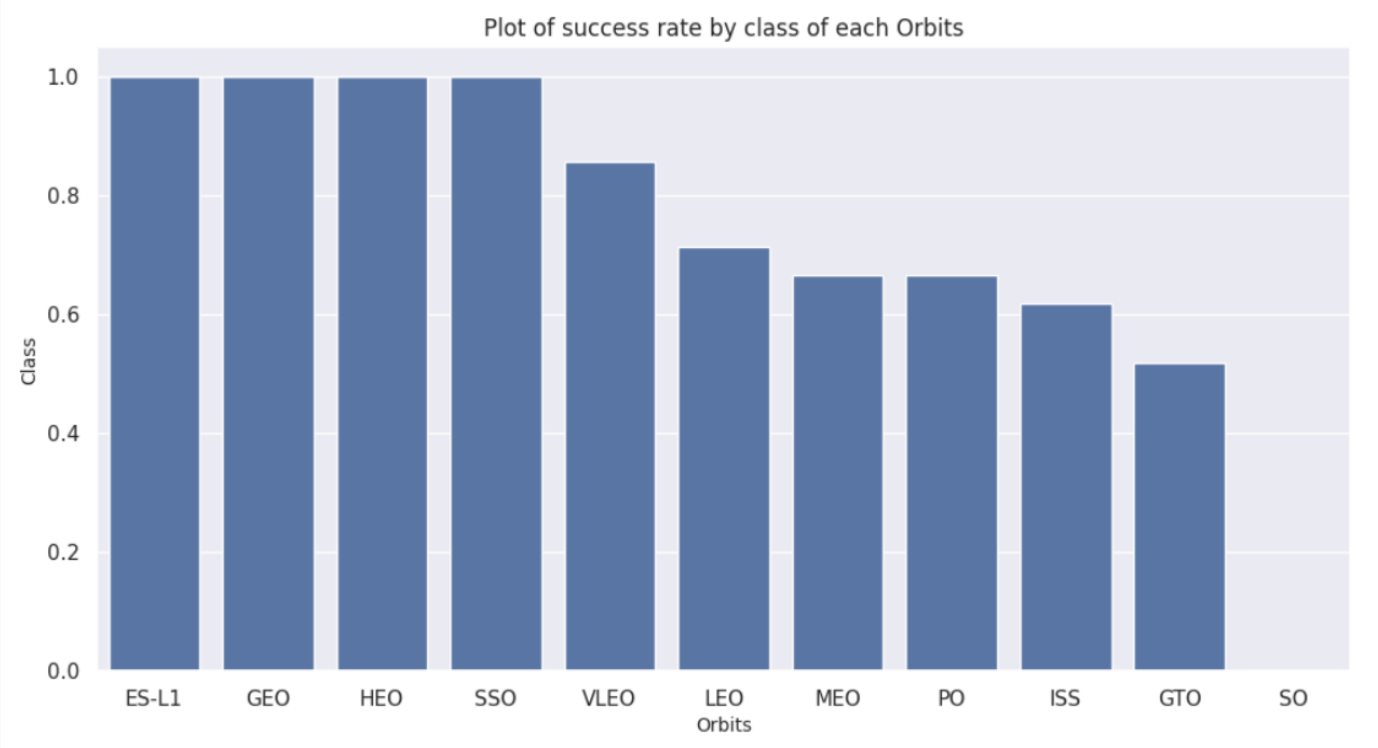
Payload vs. Launch Site

The greater the playload mass for launch site CCAFS SLC 40 the higher the success rate for the rocket.

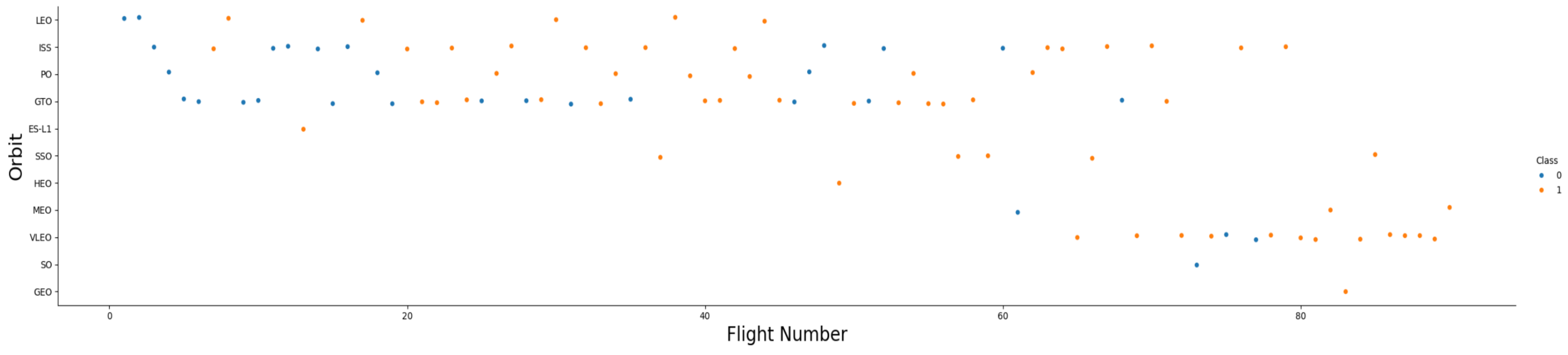


Success Rate vs. Orbit Type

We can see that ES-L1, GEO, HEO, SSO had the most success rate.

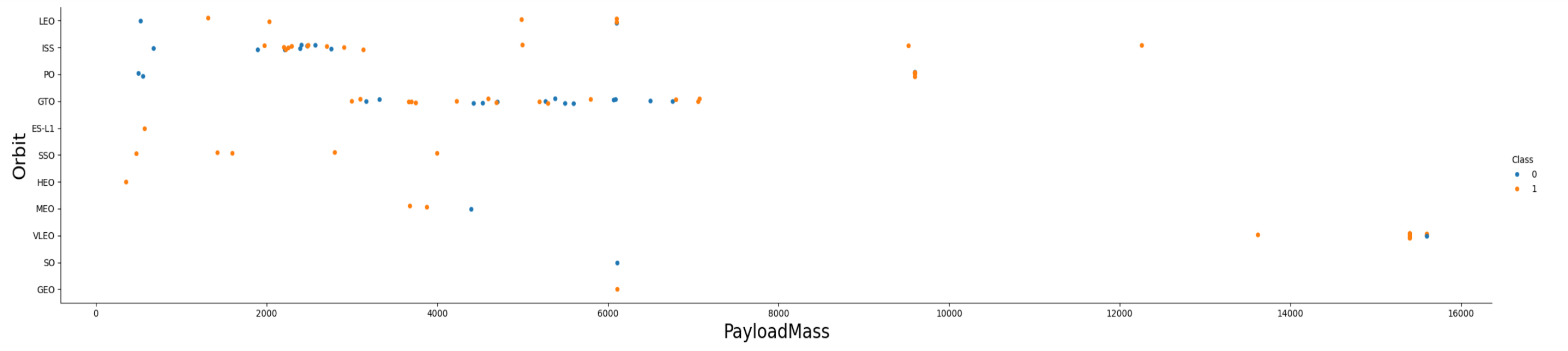


Flight Number vs. Orbit Type



The plot below shows the flight number vs. orbit type. We observe that in the LEO orbit, success is related to the number of flights whereas in the GTO orbit, there is no relationship between flight number and the orbit.

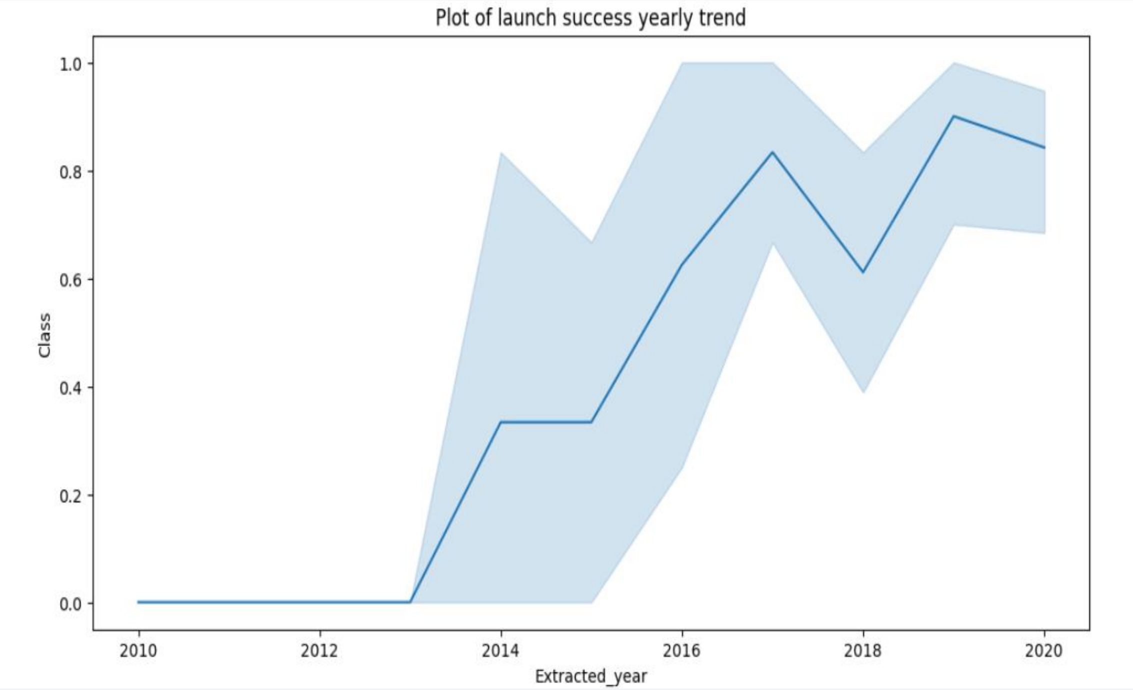
Payload vs. Orbit Type



- We can observe that with heavy payloads, the successful landing are more for PO, LEO and ISS orbits.

Launch Success Yearly Trend

From the plot, we can observe that success rate since 2013 kept on increasing till 2020.



All Launch Site Names

We used the key word DISTINCT to show only unique launch sites from the SpaceX data.

```
%sql SELECT DISTINCT "Launch_Site" FROM SPACEXTBL
```

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

Launch Site Names Begin with 'CCA'

We Find 5 records where launch sites 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

We 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

We calculate the 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_")
2928.4

First Successful Ground Landing Date

We find the dates of the first successful landing outcome on ground pad

```
%sql SELECT MIN(Date) FROM SPACEXTBL where "Landing_Outcome" like "Success (ground pad)"
```

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

MIN(Date)
2015-12-22

Successful Drone Ship Landing with Payload between 4000 and 6000

We 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 "Success (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

We calculate the total number of successful and failure mission outcomes

```
%sql SELECT count("Mission_Outcome") FROM SPACEXTBL WHERE Mission_Outcome LIKE 'Success%'
```

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

count("Mission_Outcome")
100

```
%sql SELECT count("Mission_Outcome") FROM SPACEXTBL WHERE Mission_Outcome LIKE 'Failure%'
```

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

count("Mission_Outcome")
1

Boosters Carried Maximum Payload

We list the names of the booster which have carried the maximum payload mass

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

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

Booster_Version	PAYLOAD_MASS__KG_
F9 B5 B1048.4	15600
F9 B5 B1048.5	15600
F9 B5 B1049.4	15600
F9 B5 B1049.5	15600
F9 B5 B1049.7	15600
F9 B5 B1051.3	15600
F9 B5 B1051.4	15600
F9 B5 B1051.6	15600
F9 B5 B1056.4	15600
F9 B5 B1058.3	15600
F9 B5 B1060.2	15600
F9 B5 B1060.3	15600

2015 Launch Records

We 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), "Booster_Version", "Launch_Site", "Landing_Outcome" FROM SPACEXTBL WHERE Landing_Outcome like 'Failure (drone ship)' AND substr(Date,0,5)='2015'
```

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

substr(Date, 6,2)	Booster_Version	Launch_Site	Landing_Outcome
01	F9 v1.1 B1012	CCAFS LC-40	Failure (drone ship)
04	F9 v1.1 B1015	CCAFS LC-40	Failure (drone ship)

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

We 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") FROM SPACEXTBL WHERE DATE BETWEEN '2010-06-04' AND '2017-03-20' GROUP BY "Landing_Outcome" ORDER BY COUNT("Landing_Outcome") DESC
```

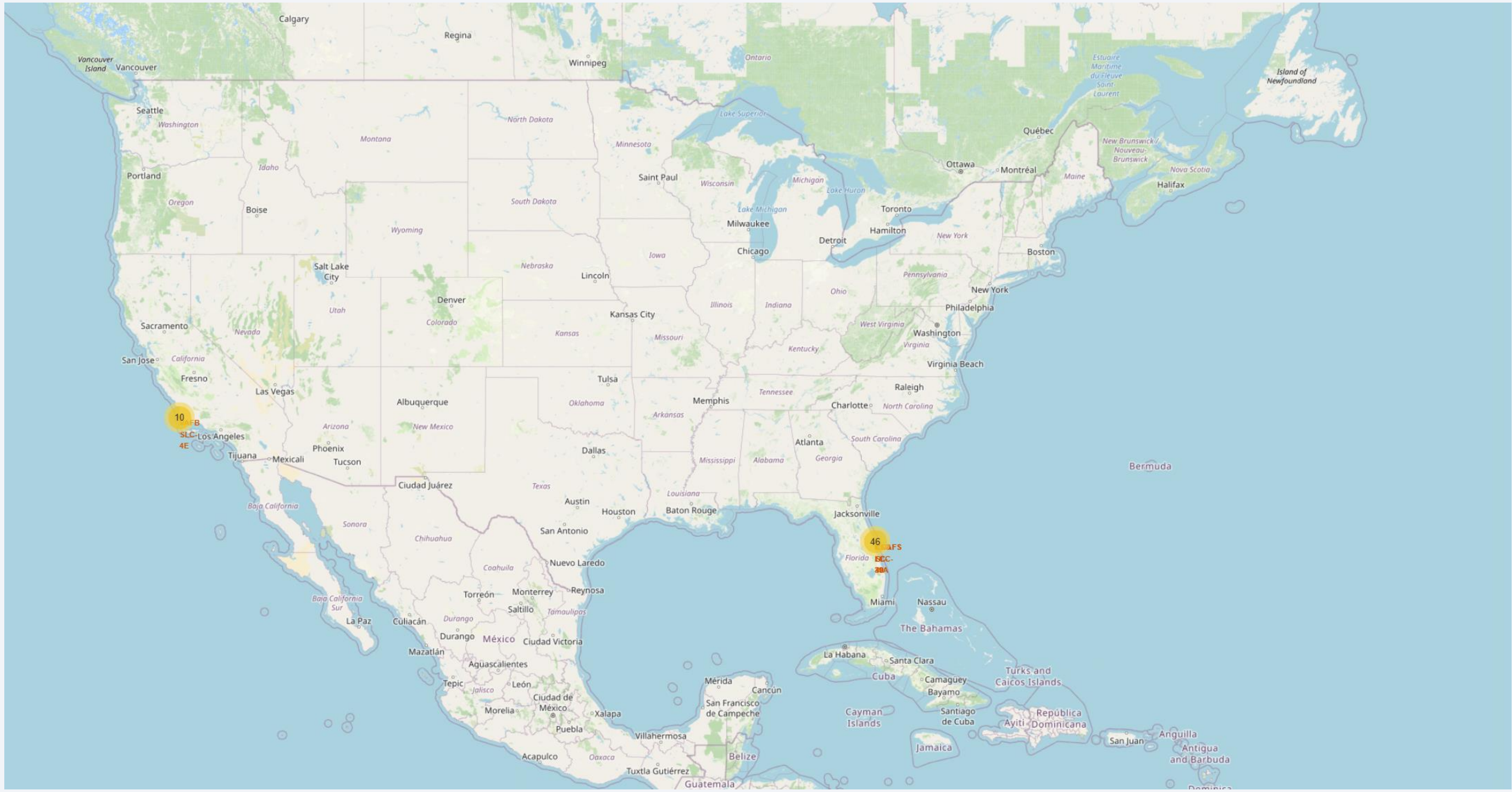
```
 * sqlite:///my_data1.db
Done.
```

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

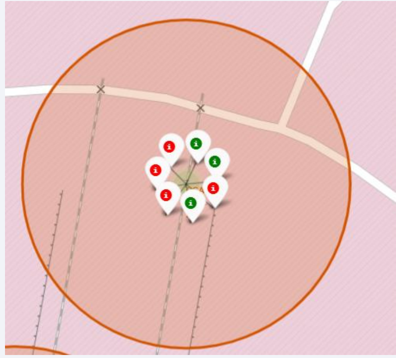
Section 3

Launch Sites Proximities Analysis

all launch sites' location markers on a global map



the color-labeled launch outcomes on the map

Florida launch sites

California launch sites

Green marker shows successful launches and red marker shows failures

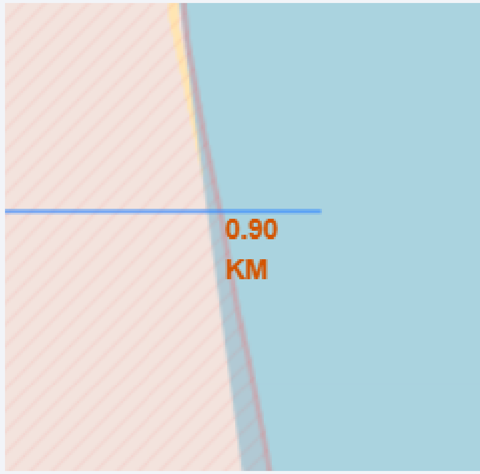
The distance from the launch site to its vicinity is calculated and displayed

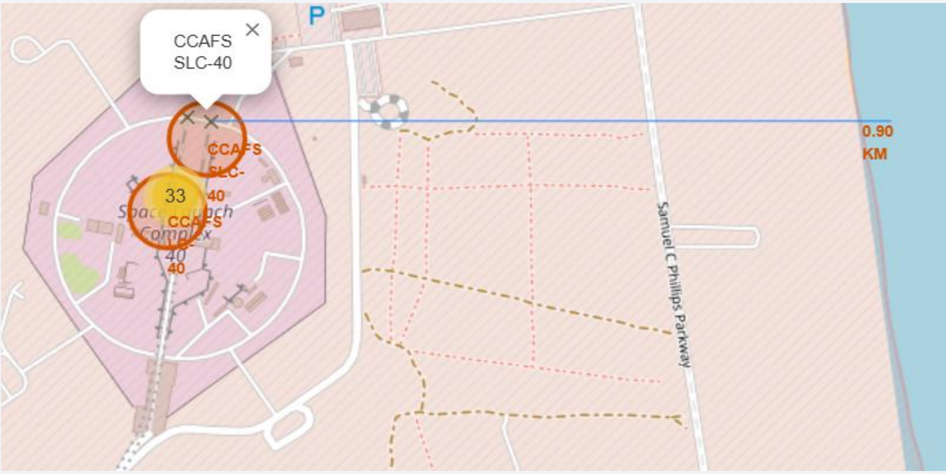
Are launch sites in close proximity to railways? No

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



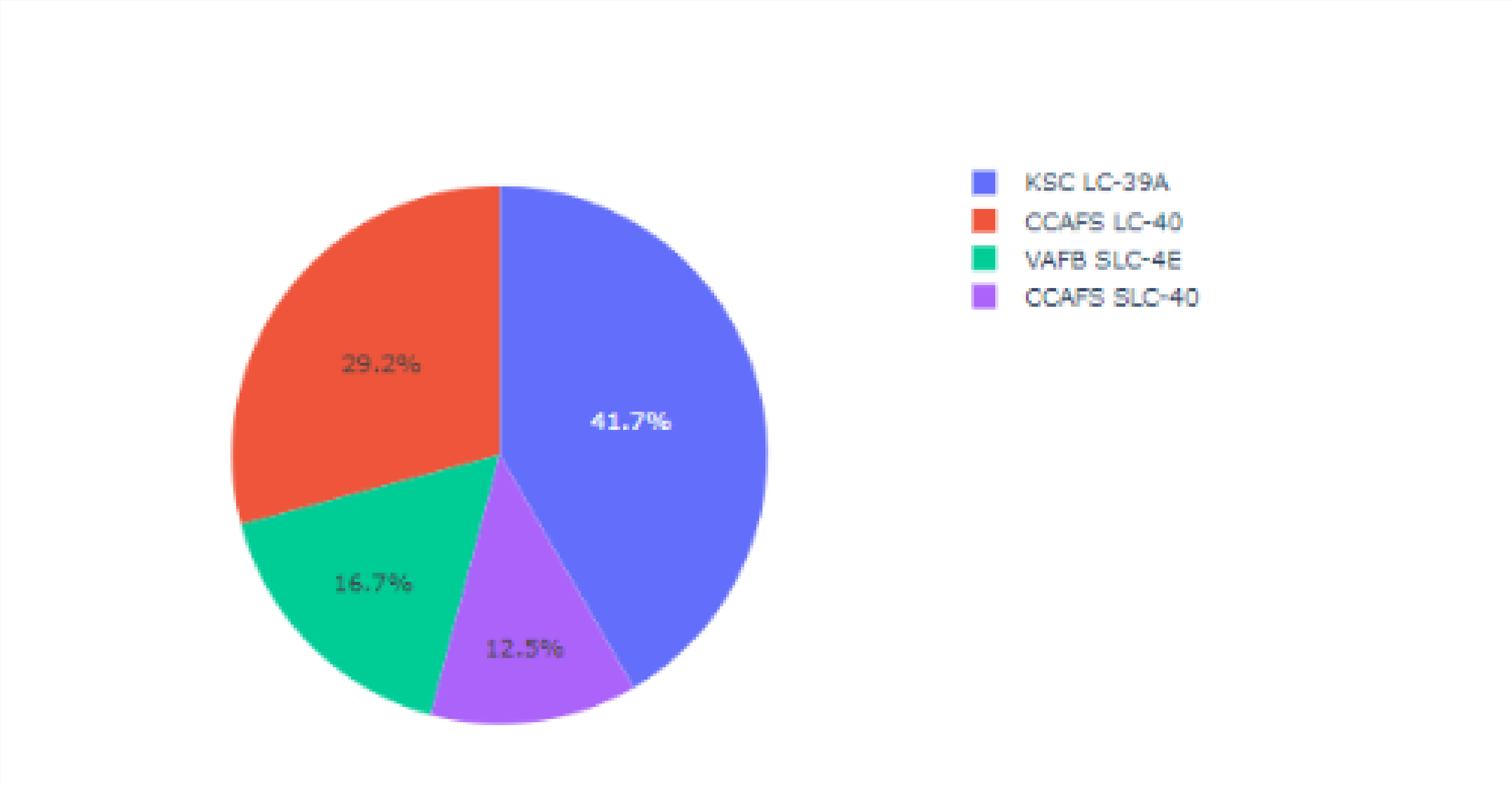


Section 4

Build a Dashboard with Plotly Dash

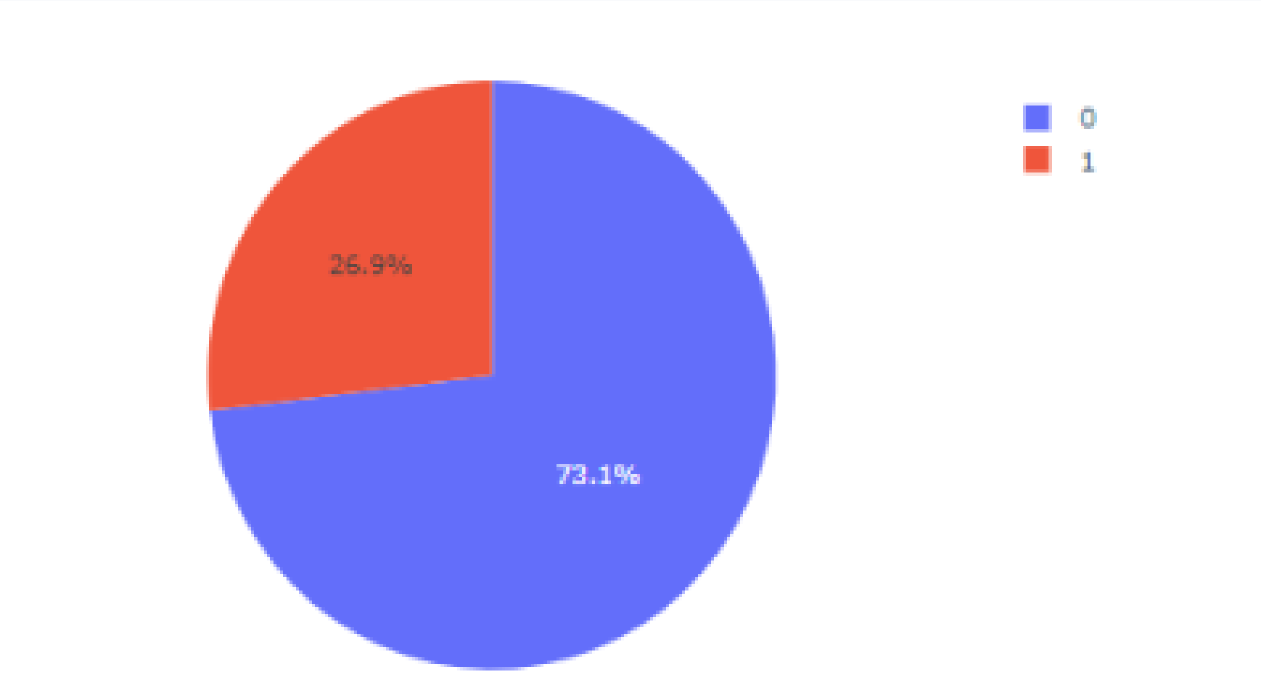
launch success count for all sites, in a pie chart

KSC LC-39A had the most successful launches from all the sites



pie chart for the launch site with highest launch success ratio

KSC LC-39A achieved a 73.1% successful rate and 26.9% failure rate

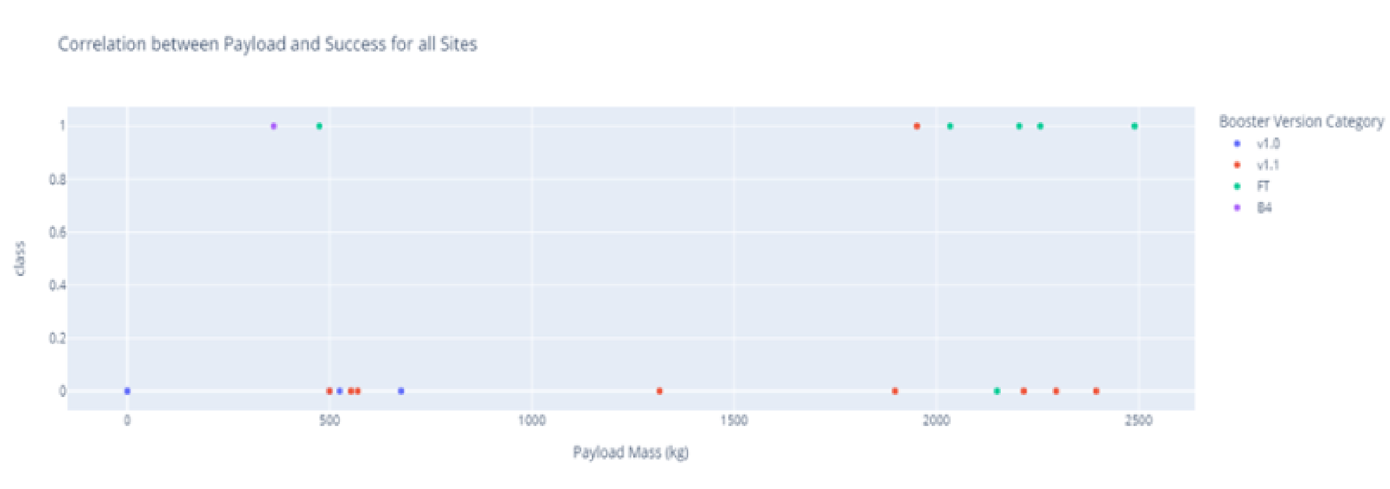


Payload vs. Launch Outcome scatter plot for all sites

Payload vs. Launch Outcome scatter plot for all sites, with different payload selected in the range slider:

- [0-2,500]

- [2,500-5,000]

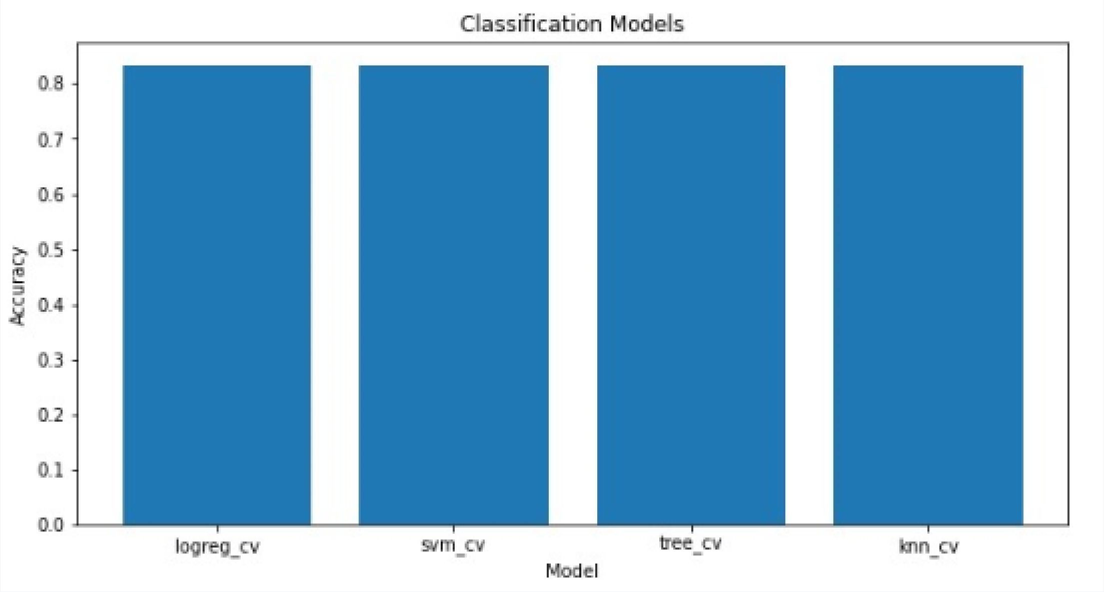


Section 5

Predictive Analysis (Classification)

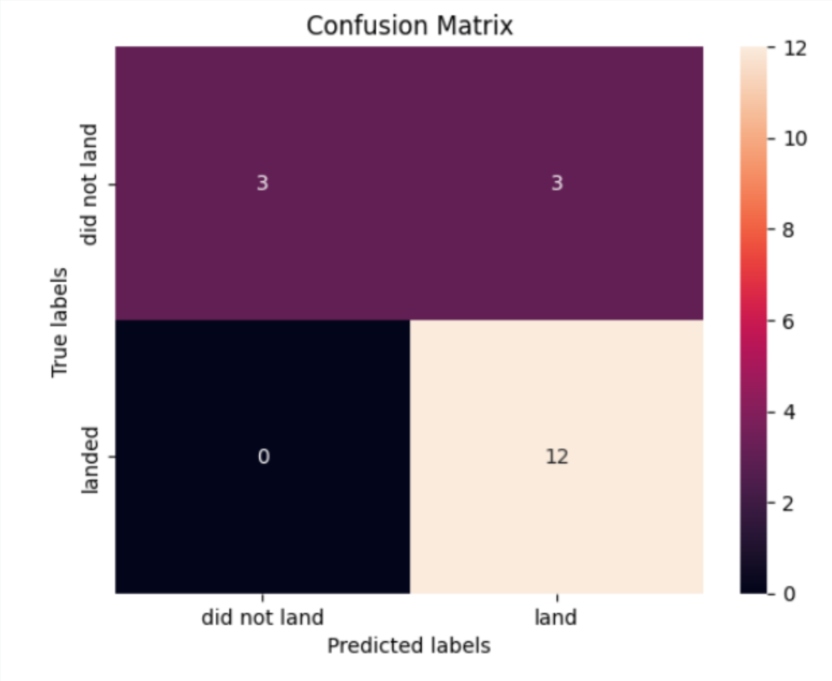
Classification Accuracy

the best performing model is related to decision tree model equal to 0.87 is shown in the chart:



Confusion Matrix

- confusion matrix of the best performing model that is related to decision tree model equal to 0.87 is shown below:



Conclusions

We can conclude that:

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

✓Launch success rate started to increase in 2013 till 2020

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

✓KSC LC-39A had the most successful launches of any sites

✓The decision tree classifier is the best machine learning algorithm for this task

Thank you!