IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

<Victor Cruz>
<02/02/2025>

Outline

- Executive Summary : (3)
- Introduction : (4)
- Methodology : (5)
- Results : (16)
- Conclusion : (45)
- Appendix : (46)

Executive Summary

- Summary of methodologies :
- Gathering data from SpaceX public API and SpaceX Wikipedia page.
- Creating the "class" label column that classifies successful landings.
- Data exploration using SQL, visualization, folium maps. Collecting relevant columns to use as features.
- Casting all categorical variables to binary using one-hot encoding. Data standardization and use of GridSearchCV.
- Summary of all results :
- Machine learning models used: Logistic regression, Support vector machine, Decision tree classifier, and K-nearest neighbors.
- All models over-predicted successful landings with an accuracy rate of approximately 83.33%.

Introduction

- Project background and context :
- SpaceY wants to lead and compete with SpaceX.
- SpaceX has more capital and better prices.
- Problems you want to find answers :
- We create and train a machine learning model to predict successful stage recovery.

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - Gathering data from SpaceX API and SpaceX Wikipedia
- Perform data wrangling
  - We classify successful and unsuccessful landings
- 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

- Describe how data sets were collected.
- Data gathering is a combination of API requests from Space X public API and web scraping data from a table in Space X's Wikipedia entry.
- Flowchart of data collection from API and the flowchart of data collection from webscraping.
- You need to present your data collection process use key phrases and flowcharts

Space X API Data Columns:

FlightNumber, Date, BoosterVersion, PayloadMass, Orbit, LaunchSite, Outcome, Flights, GridFins,

Reused, Legs, LandingPad, Block, ReusedCount, Serial, Longitude, Latitude.

Wikipedia Webscrape Data Columns:

Flight No., Launch site, Payload, PayloadMass, Orbit, Customer, Launch outcome, Version Booster, Booster landing, Date, Time

Data Collection – SpaceX API

- Data collection with SpaceX API

- Github: https://github.com/VictorGonTec/Data-Science-Capstone/blob/main/Main/labs-jupyter-spacex-Data%20wrangling.ipynb

Request Spacex API -> Json File + list[Launch site, Booster Version, Payload] ->

Json_normalize DataFrame to Json form -> Dictionary { of data to use } -> Convert Dictionary to DataFrame

-> Filter Data Falcon 9 only -> Imputing missing data in PayloadMass with the mean()

Data Collection - Scraping

- Web scraping
  Data Collection

- GitHub URL:
  https://github.com/VictorGonTec/Data-Science-Capstone/blob/main/Main/jupyter-labs-webscraping.ipynb

Request HTML from Wlkipedia -> Beautiful Soup HTML5 parser ->Find Launch Tables rows -> Create dictionary {datasets} -> Iterate tables and extract dictionary {data entry} -> Casting dictionary {} to DataFrame.

Data Wrangling

- Describe how data were processed
- I created a training label with landing results where success = 1 and failure = 0 based on what I learned in past labs.
- The result column has two components: "Mission Outcome" and "Landing Location".
- New training label column "class" with a value of 1 if "Mission Outcome" is True and 0 otherwise.
- GitHub URL:https://github.com/VictorGonTec/Data-Science-Capstone/blob/main/Main/jupyter-labs-spacex-data-collection-api.ipynb

EDA with Data Visualization

- Summarize what charts were plotted and why you used those charts
- Exploratory data analysis performed on the variables Flight Number, Payload Mass, Launch Site, Orbit, Class, and Year.
- Flight Number vs. Payload Mass, Flight Number vs. Launch Site, Payload Mass vs. Launch Site, Orbit vs. Success Rate, Flight Number vs. Orbit, Payload vs. Orbit, and Annual Success Trend
- Plots Used:
- Scatter Plots,
- Line Charts,
- Bar Charts.
- GitHub URL :https://github.com/VictorGonTec/Data-Science-Capstone/blob/main/Main/SpaceX_Machine%20Learning%20Prediction_Part_5.ipynb

EDA with SQL

- Using bullet point format, summarize the SQL queries you performed
- IBM DB2 to load the Database.
- Queried using SQL Python integration.
- use of multiple queries to get a better understanding of the data.
- Queried information about launch site names, mission outcomes, various pay load sizes of customers and booster versions, and landing outcomes
- GitHub URL: https://github.com/VictorGonTec/Data-Science-Capstone/blob/main/Main/jupyter-labs-eda-sql-coursera_sqllite.ipynb

Build an Interactive Map with Folium

- Folium maps mark launch sites, successful and failed landings, and an example of proximity to key locations:
- We also visualize successful landings in relation to location to understand why launch sites are located in those places.
- GitHub URL : https://github.com/VictorGonTec/Data-Science-Capstone/blob/main/Main/Interactive%20Visual%20Analytics%20with%20Folium.ipynb

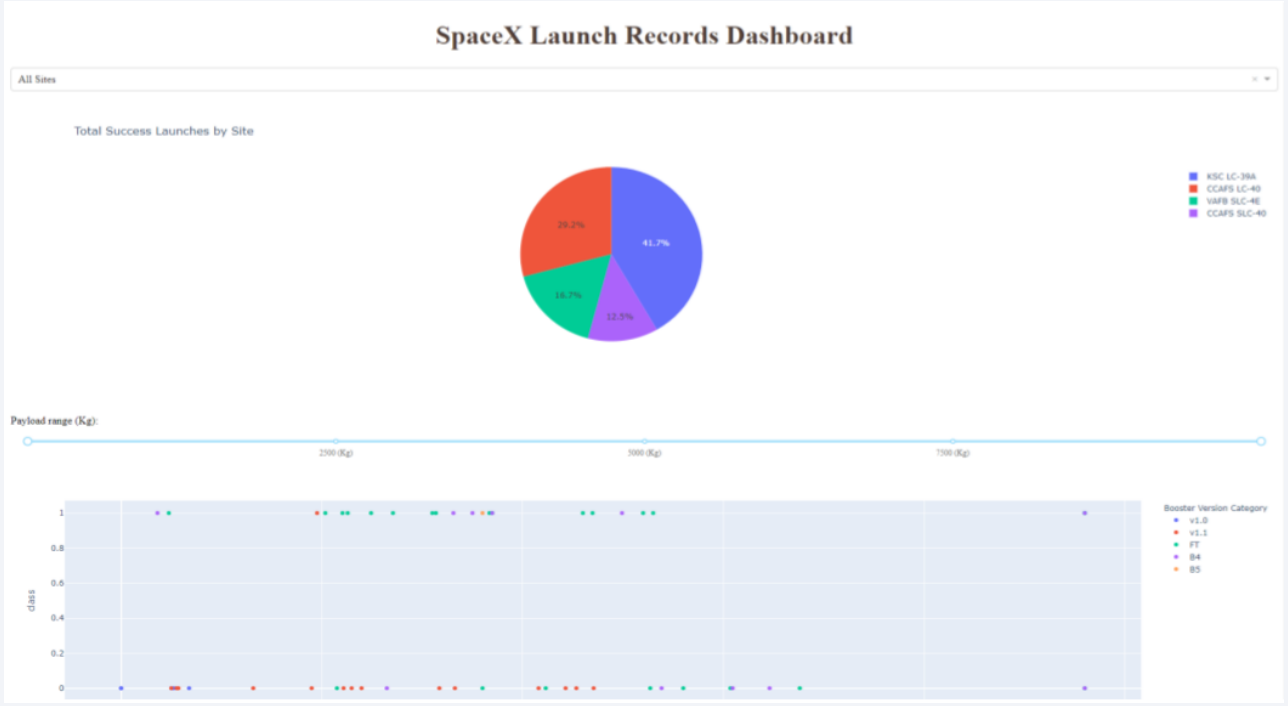
Build a Dashboard with Plotly Dash

- Dashboard includes a pie chart and a scatter plot.
- The Pie chart can be selected to show the distribution of successful landings across all launch sites and can be selected to show the success rates for individual launch sites.
- The Scatter plot takes two inputs: all sites or an individual site and the payload mass on a slider and the pie chart is used to visualize the launch site success rate
- GitHub URL: https://github.com/VictorGonTec/Data-Science-Capstone/blob/main/Main/spacex_dash_app.py

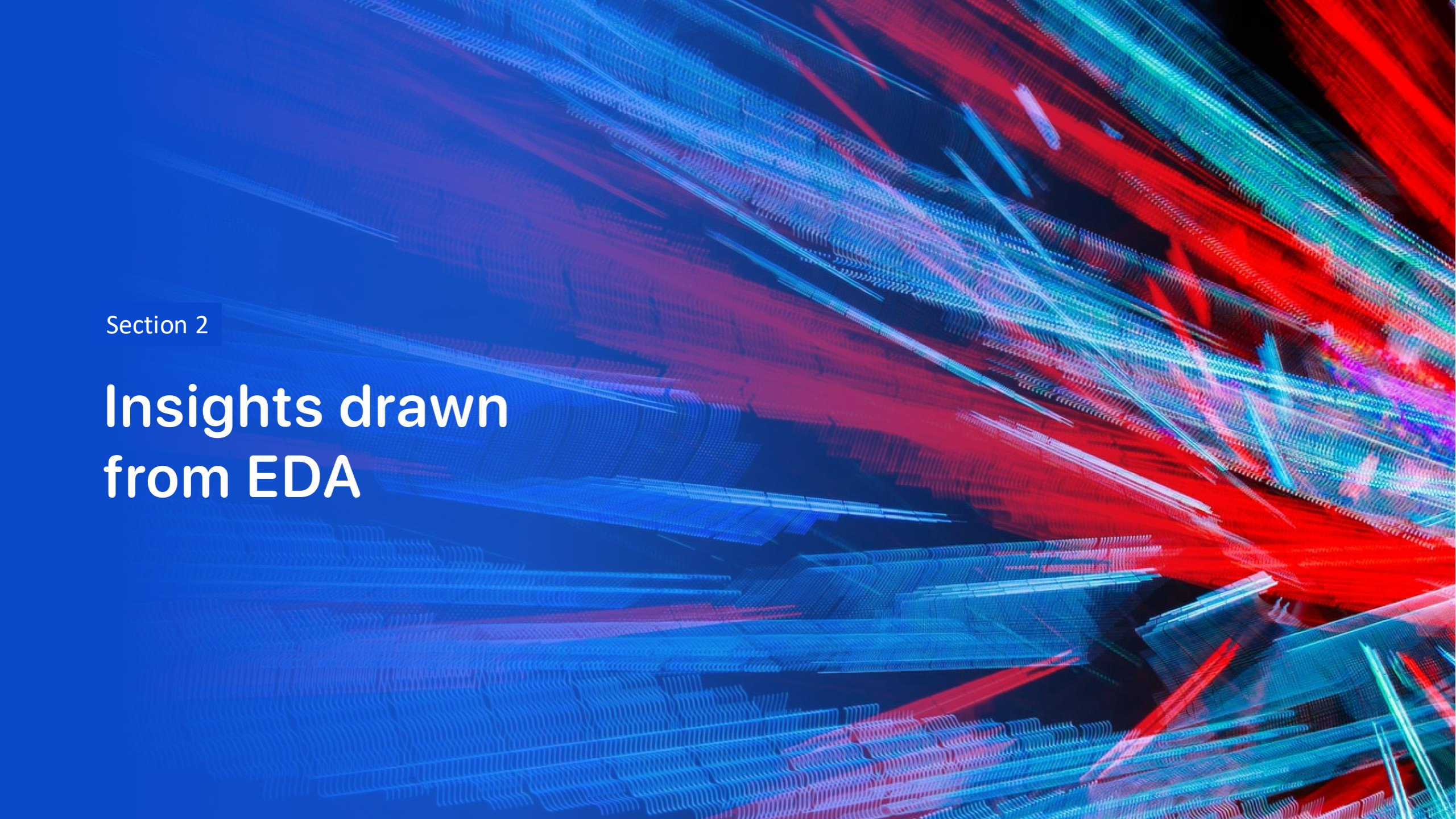
Predictive Analysis (Classification)

- Split the label class column -> Fit the features with StandardScaler() ->
- Train_Test_Split() the data -> Apply GridSearchCV() -> GridSearchCV() on Logisteic Regression, DecisionTree, KNigbors and SVM. -> score(Test data) -> ConfusionMatrix(All Models) -> BarPlot over all score models.
- GitHub URL: https://github.com/VictorGonTec/Data-Science-Capstone/blob/main/Main/SpaceX_Machine%20Learning%20Prediction_Part_5.ipynb

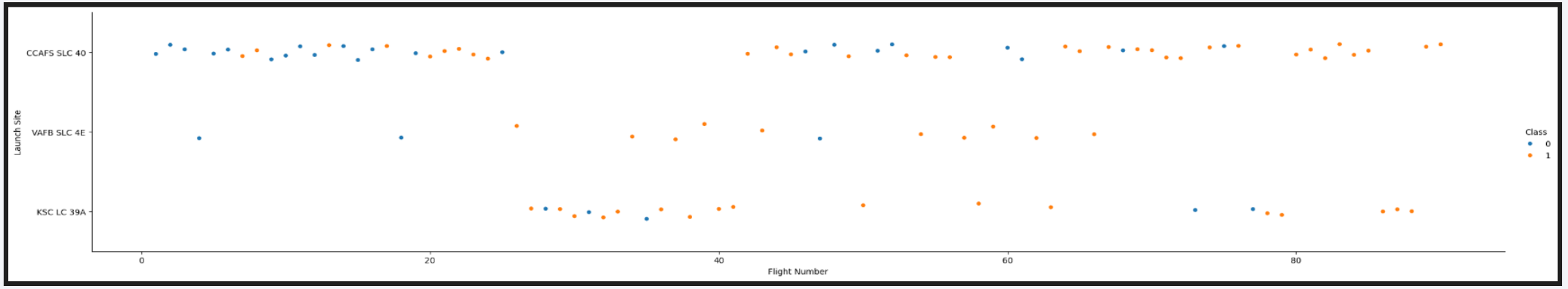
Results



Plotly Dashboard Previeu accuracy of 83.33% of de model result


Section 2
Insights drawn from EDA

Flight Number vs. Launch Site

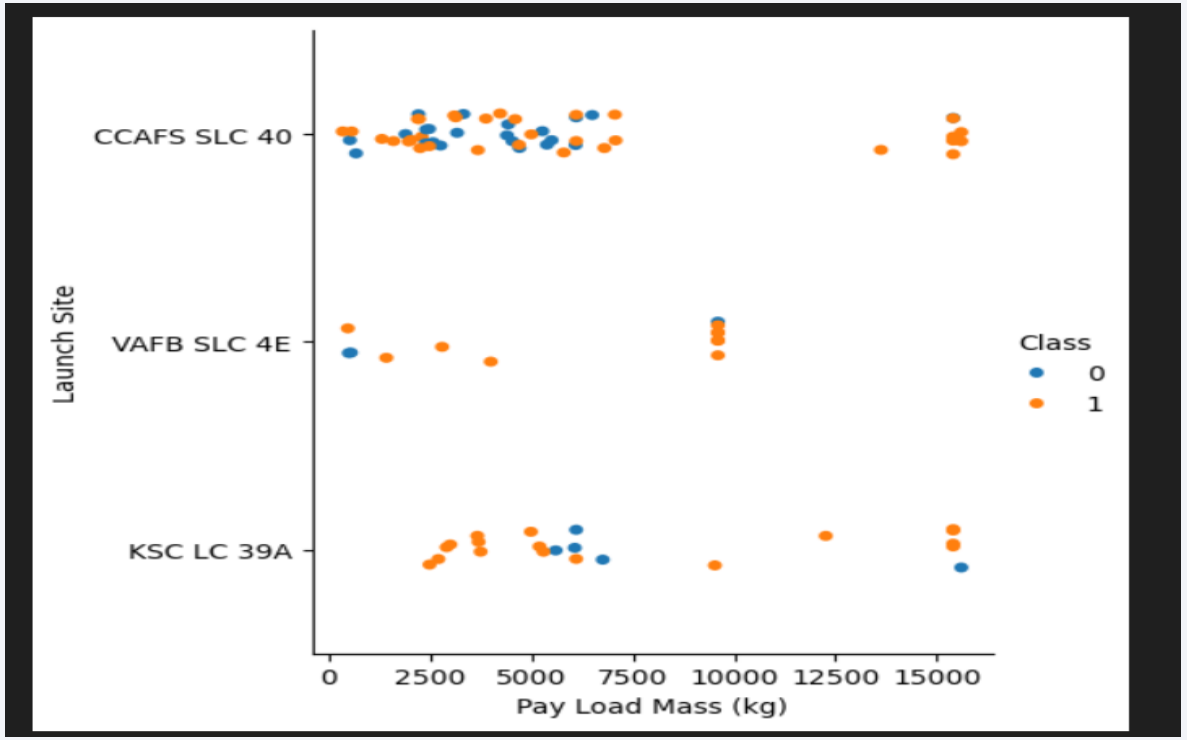


Blue Represent Successful launch and Orange unsuccessful launch.

Graphic suggests an increase in success rate over time (indicated in Flight Number).

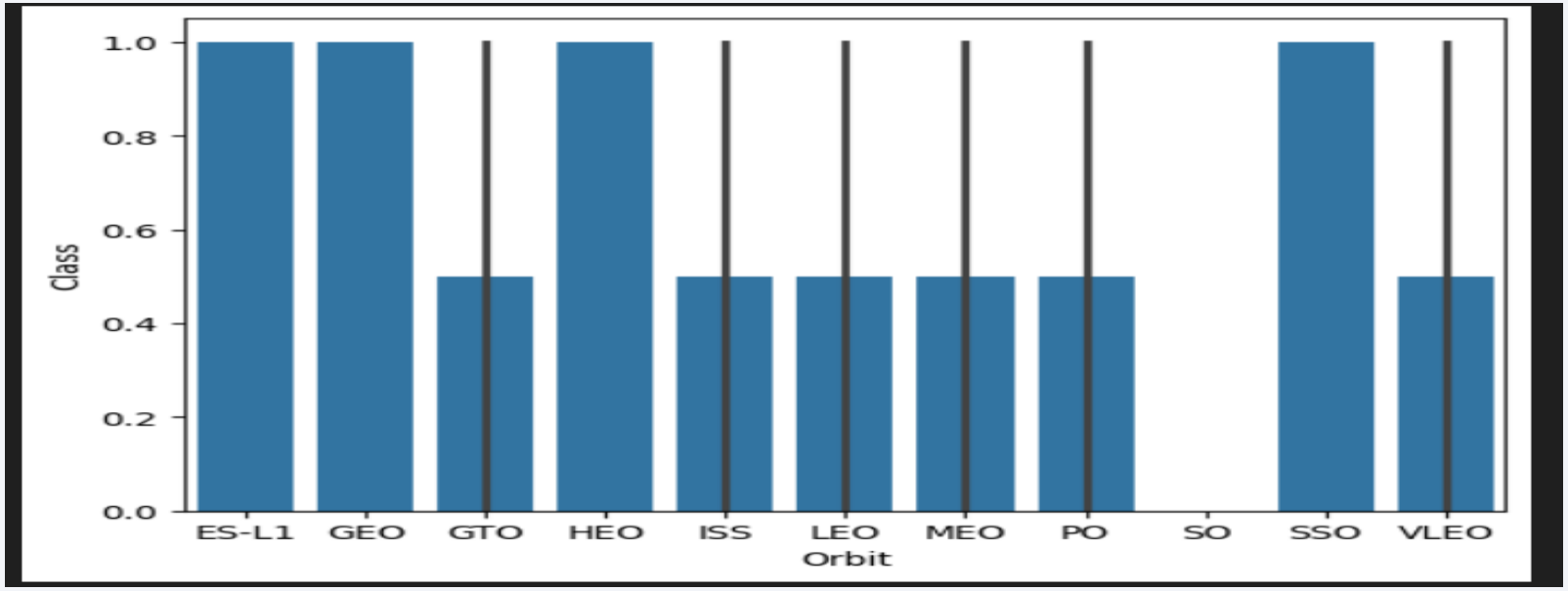
CCAFS appears to be the main launch site as it has the most volume.

Payload vs. Launch Site



- Blue Represent Successful launch and Orange unsuccessful launch.
- Payload mass appears to fall mostly between 0-6000 kg.
- Different launch sites also seem to use different payload mass.

Success Rate vs. Orbit Type



ES-L1 (1), GEO (1), HEO (1) tienen una tasa de éxito del 100 % (los tamaños de muestra están entre paréntesis).

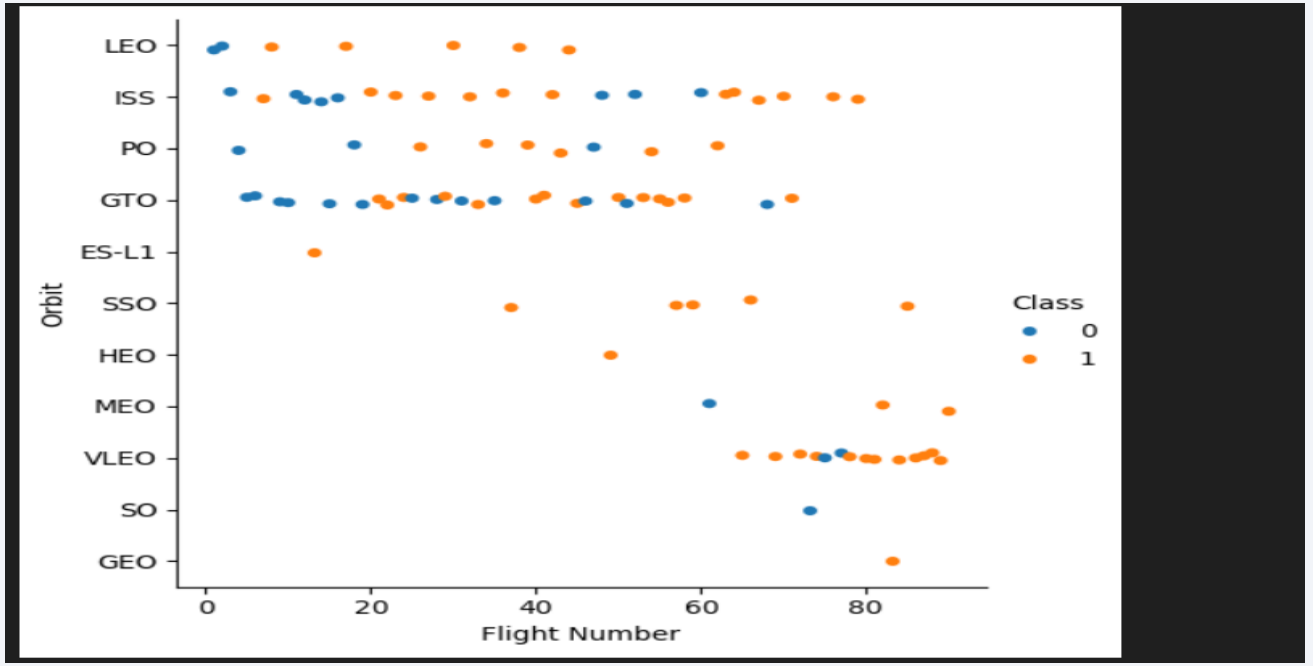
SSO (5) tiene una tasa de éxito del 100 %.

VLEO (14) tiene una tasa de éxito y unos intentos aceptables.

SO (1) tiene una tasa de éxito del 0 %.

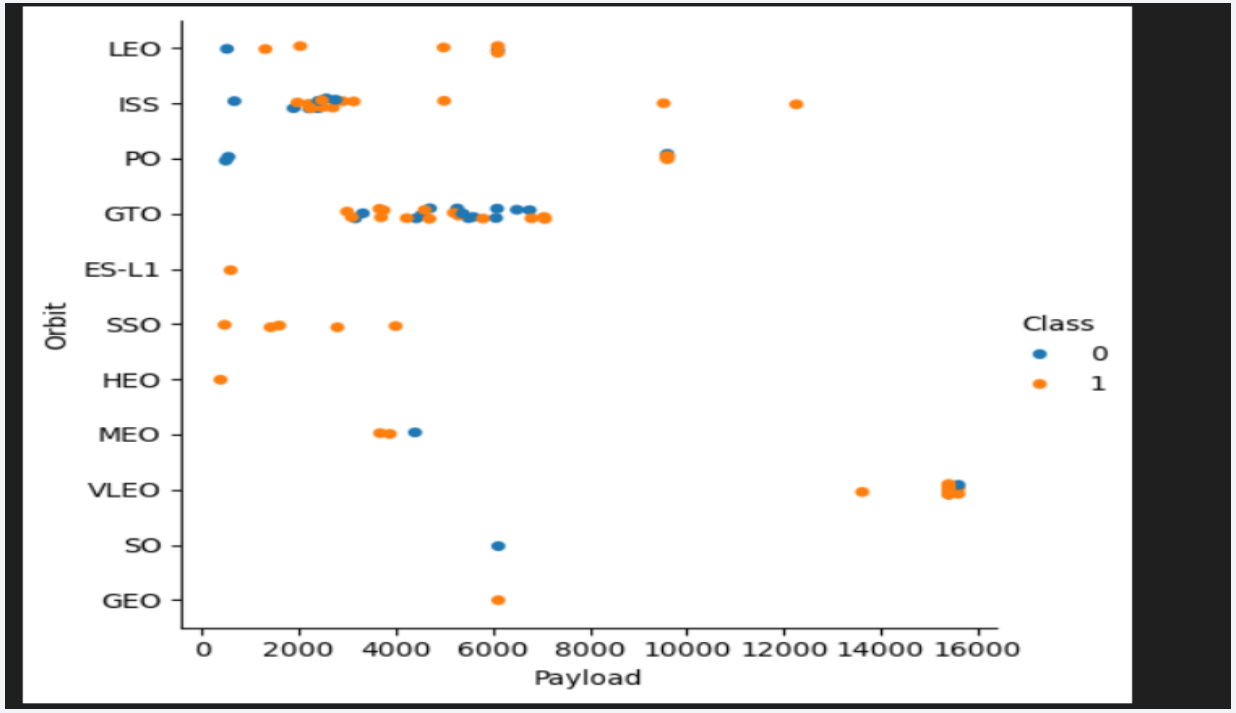
GTO (27) tiene una tasa de éxito de alrededor del 50 %, pero la muestra más grande.

Flight Number vs. Orbit Type



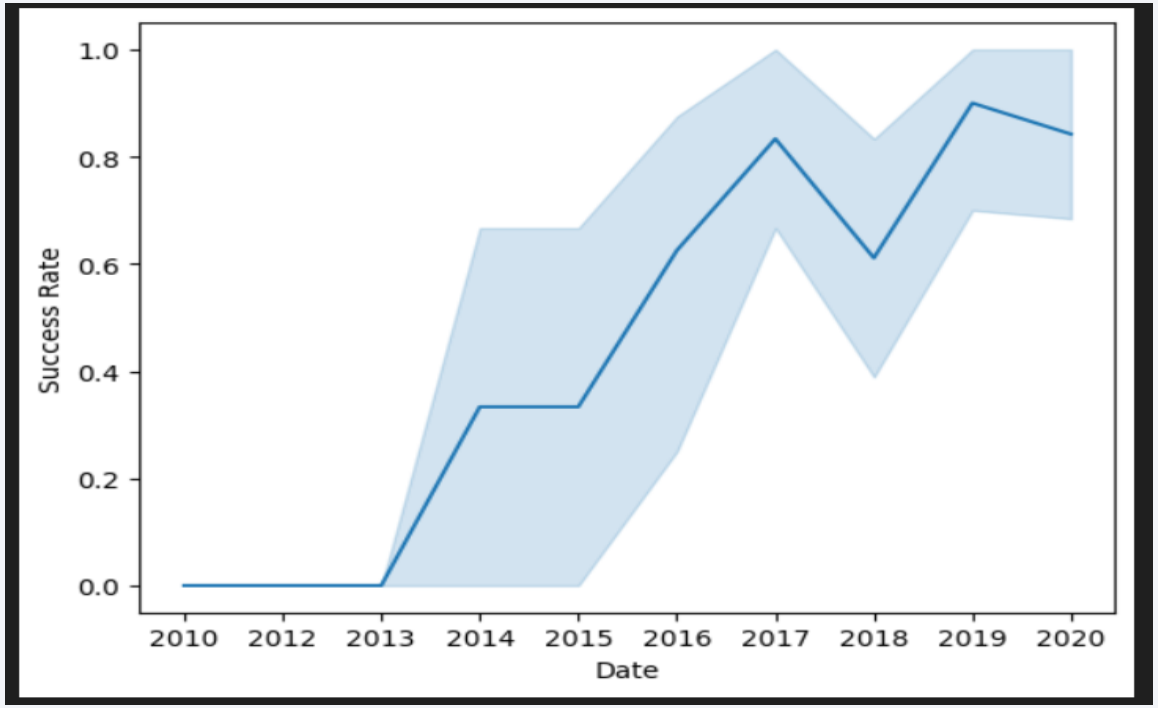
- Launch orbit preferences changed depending on flight number.
- Launch outcome appears to be related to this preference.
- SpaceX started with LEO orbits that were moderately successful and returned to VLEO in recent launches.

Payload vs. Orbit Type



- Payload mass appears to correlate with LEO orbit.
- SSO appear to have relatively low payload mass.
- VLEO only has payload mass values at the high end of the range.

Launch Success Yearly Trend



- Success in recent years around 80%.
- Success generally increases over time since 2013 with a slight drop in 2018

All Launch Site Names

```
%sql select distinct Launch_Site from SPACEXTABLE
```

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

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

- Check the database for unique launch site names.
- It is likely that CCAFS SLC-40 and CCAFSSLC-40 represent the same launch site with data entry errors.
- CCAFS LC-40 was the previous name.
- Most likely there are only 3 unique launch_site values:
- CCAFS SLC-40, KSC LC-39A, VAFB SLC-4E

Launch Site Names Begin with 'CCA'

```
%sql select * from SPACEXTABLE where Launch_Site like '%CCA%' limit 5
```

Python

```
 * 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

Find 5 records where launch sites begin with `CCA`

Total Payload Mass

```
%sql select sum(PAYLOAD_MASS__KG_) as Total from SPACEXTABLE where customer== 'NASA (CRS)'
```

Python

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

Total
45596

Calculate the total payload carried by boosters from NASA

This query sums the total payload mass in kg where NASA was the customer.

Average Payload Mass by F9 v1.1

```
%%sql select avg(PAYLOAD_MASS__KG_) as average
    from SPACEXTABLE where Booster_Version == 'F9 v1.1'
```

Python

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

average
2928.4

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

Average payload mass of F9 1.1 is on the low end of our payload mass range

First Successful Ground Landing Date

```
%%sql select min(Date) from SPACEXTABLE
    where Landing_Outcome == 'Success (ground pad)'
```

Python

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

min(Date)
2015-12-22

This query returns the first successful ground pad landing date.

First ground pad landing was around 2015.

Successful landings in general appear starting 2014

Successful Drone Ship Landing with Payload between 4000 and 6000

```
%%sql select Booster_Version From SPACEXTABLE
    where Landing_Outcome == 'Success (drone ship)' and PAYLOAD_MASS__KG_ > 4000 and PAYLOAD_MASS__KG_ < 6000
```

Python

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

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

- This query returns the four booster versions that had successful drone ship landings and a payload mass between 4000 and 6000 non inclusively.

Total Number of Successful and Failure Mission Outcomes

```
%%sql select Mission_Outcome, count(Mission_Outcome) as Total from SPACEXTABLE
    group by Mission_Outcome
```

Python

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

Mission_Outcome	Total
Failure (in flight)	1
Success	98
Success	1
Success (payload status unclear)	1

- This query returns a count of each mission outcome.
- SpaceX appears to achieve its mission outcome nearly 99% of the time.
- This means that most of the landing failures are intended.

Boosters Carried Maximum Payload

```
%%sql select distinct Booster_Version from SPACEXTABLE
    where PAYLOAD_MASS__KG_ == (select max(PAYLOAD_MASS__KG_) from SPACEXTABLE)
```

Python

```
 * 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

- This query returns the booster versions that carried the largest payload mass of 15600 kg.
- These booster versions are very similar and are all of the F9 B5 B10XXX variety.

2015 Launch Records

```
%%sql select
    case substr(Date,6,2)
    when '01' then 'January'
    when '02' then 'February'
    when '03' then 'March'
    when '04' then 'April'
    when '05' then 'May'
    when '06' then 'June'
    when '07' then 'July'
    when '08' then 'August'
    when '09' then 'September'
    when '10' then 'October'
    when '11' then 'November'
    when '12' then 'December'
    end as month_name,Landing_Outcome,Booster_Version,Launch_Site From SPACEXTABLE
    where substr(Date,0,5) = '2015'
```

Python

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

month_name	Landing_Outcome	Booster_Version	Launch_Site
January	Failure (drone ship)	F9 v1.1 B1012	CCAFS LC-40
February	Controlled (ocean)	F9 v1.1 B1013	CCAFS LC-40
March	No attempt	F9 v1.1 B1014	CCAFS LC-40
April	Failure (drone ship)	F9 v1.1 B1015	CCAFS LC-40
April	No attempt	F9 v1.1 B1016	CCAFS LC-40
June	Precluded (drone ship)	F9 v1.1 B1018	CCAFS LC-40
December	Success (ground pad)	F9 FT B1019	CCAFS LC-40

This query represetn a list of the failed landing_outcomes in drone ship, their booster versions, and launch site names for in year 2015

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

```
%%sql select Landing_Outcome, count(Landing_Outcome) as Total
    from SPACEXTABLE
    where date between '2010-06-04' and '2017-03-20'
    group by Landing_Outcome order by Total desc
```

Python

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

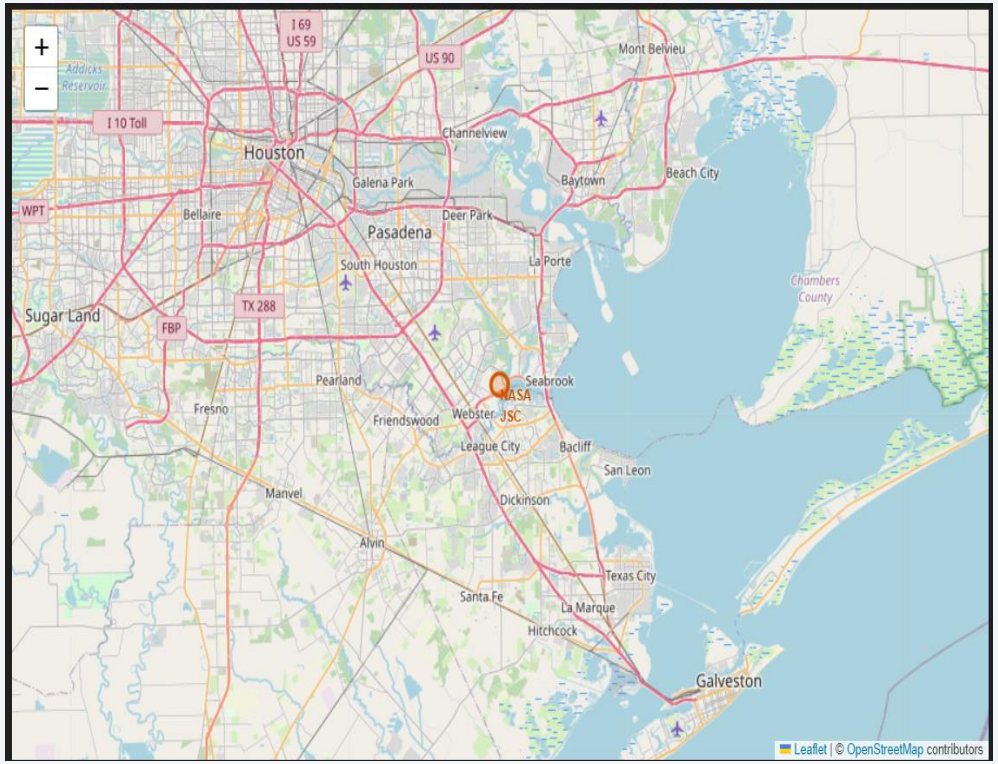
Landing_Outcome	Total
No attempt	10
Success (drone ship)	5
Failure (drone ship)	5
Success (ground pad)	3
Controlled (ocean)	3
Uncontrolled (ocean)	2
Failure (parachute)	2
Precluded (drone ship)	1

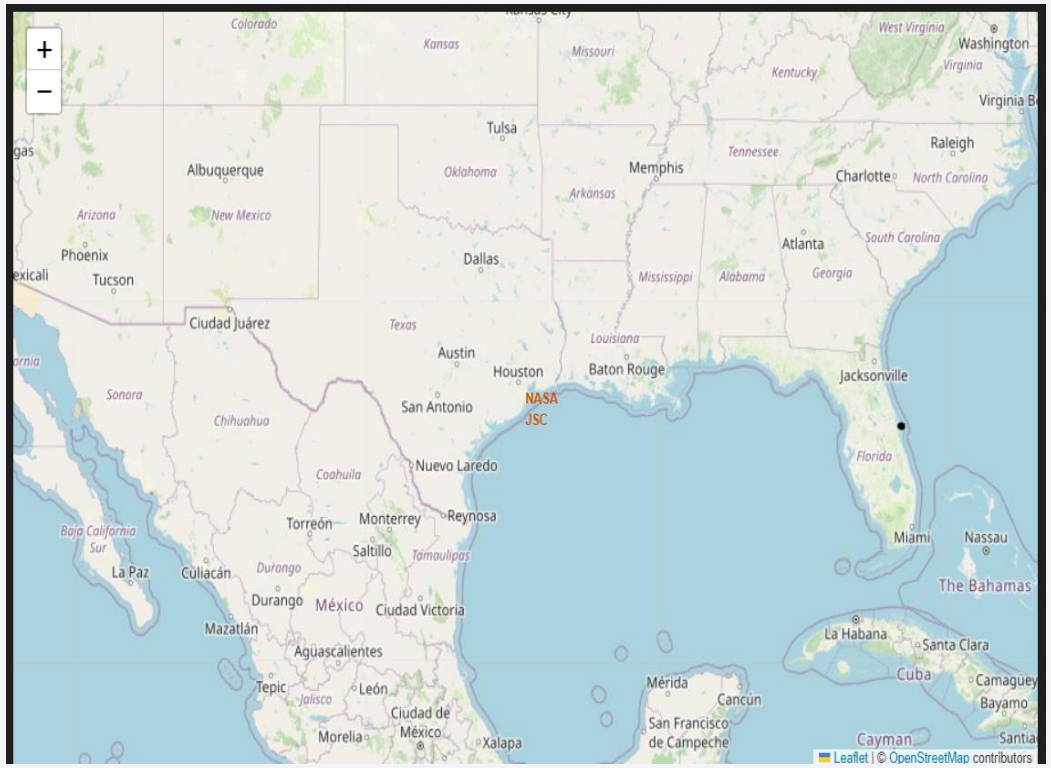
- This is rank of the count of landing outcomes of Failure drone ship and Success ground pad between the date 2010-06-04 and 2017-03-20, in descending order

Section 3

Launch Sites Proximities Analysis

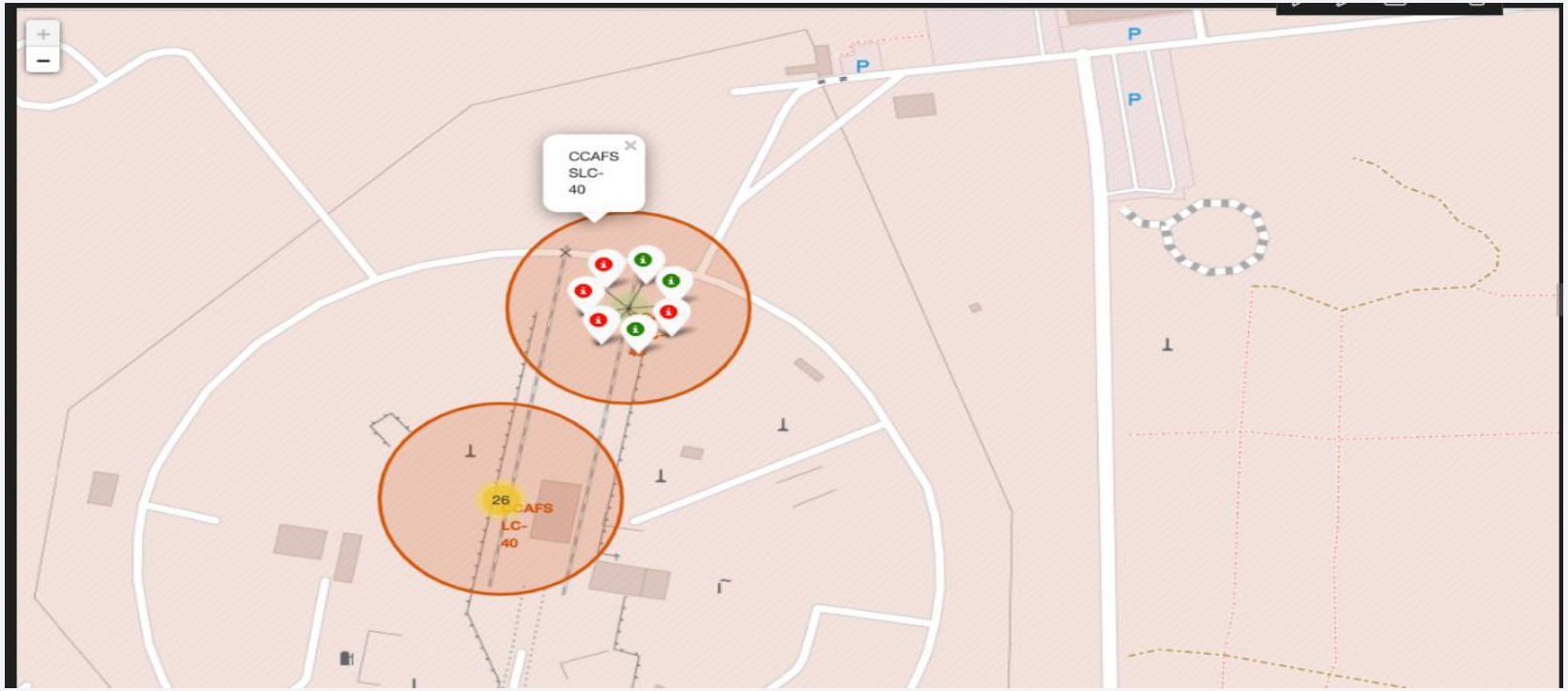
Launch Site Location





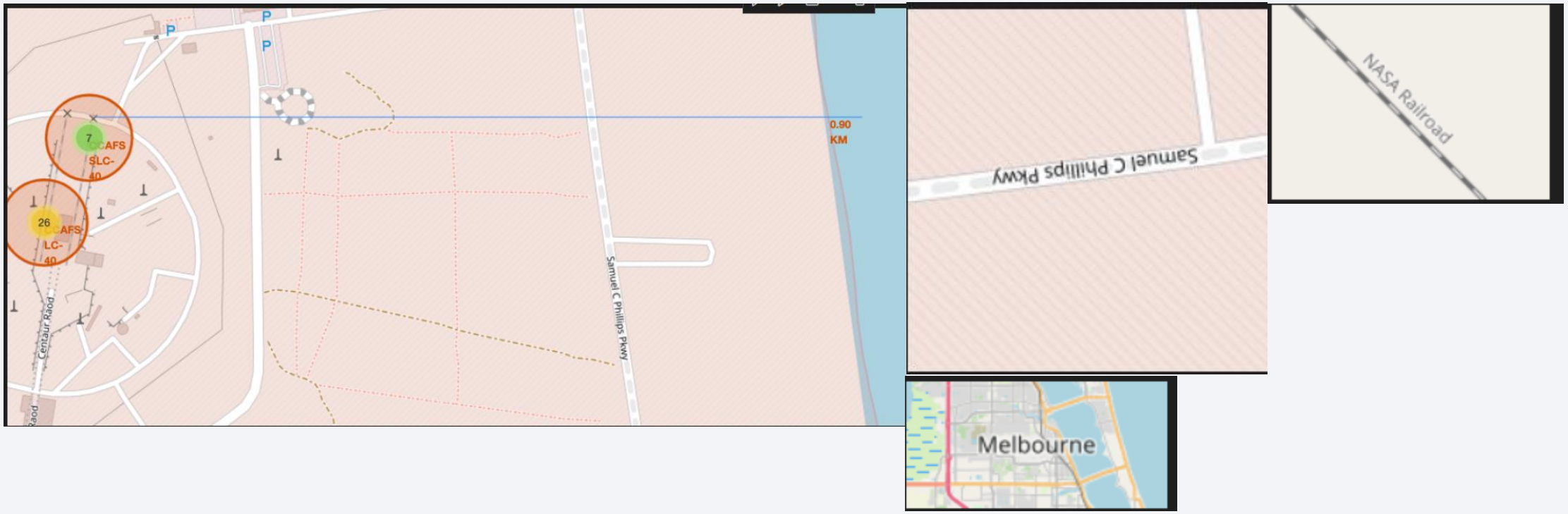
- The right map shows all launch sites relative US map. The left map shows the two Houston launch sites since they are very close to each other. All launch sites are near the ocean.

Color Marker Label

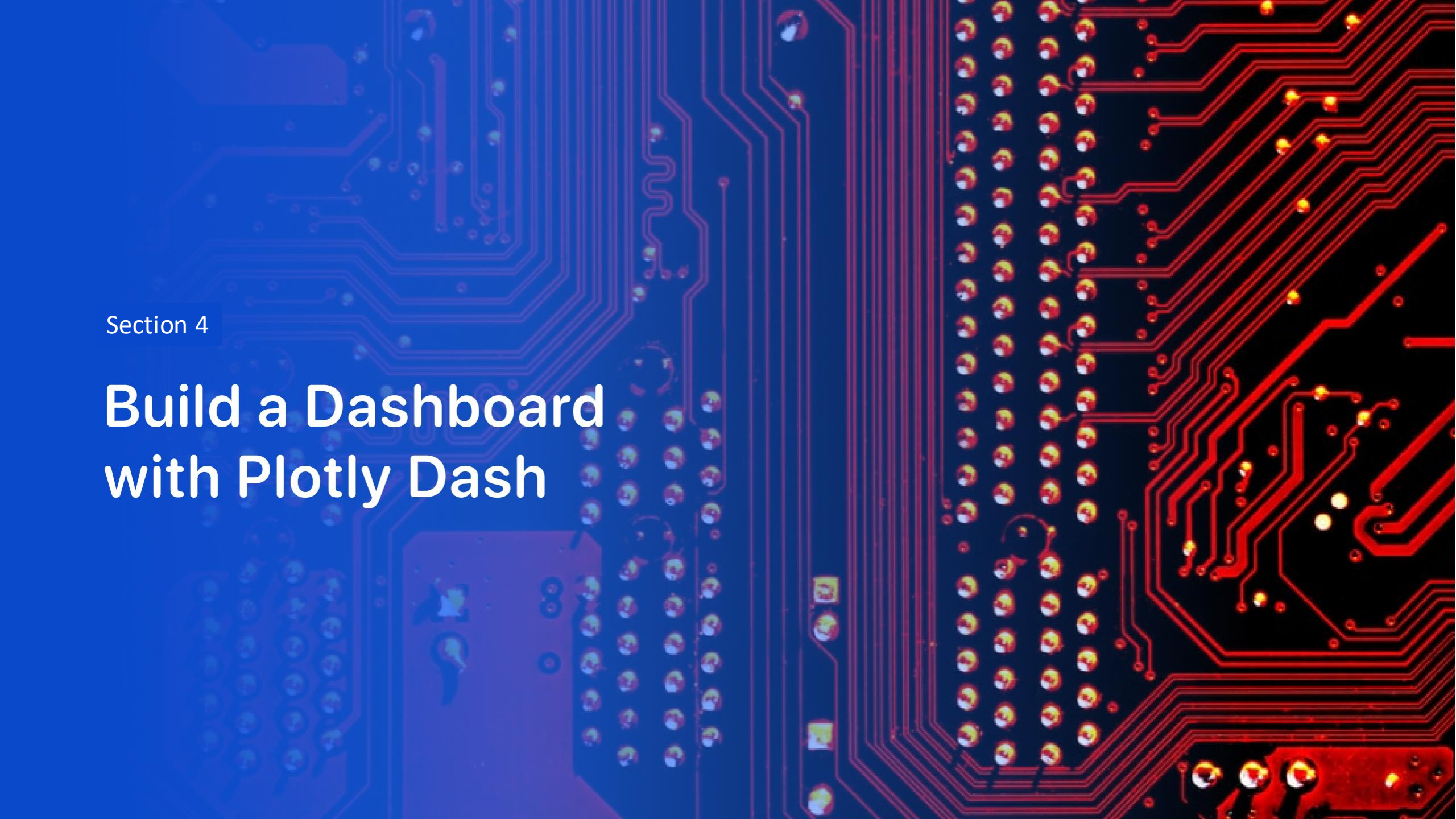


Explore the folium map and make a proper screenshot to show the color-labeled launch outcomes on the map

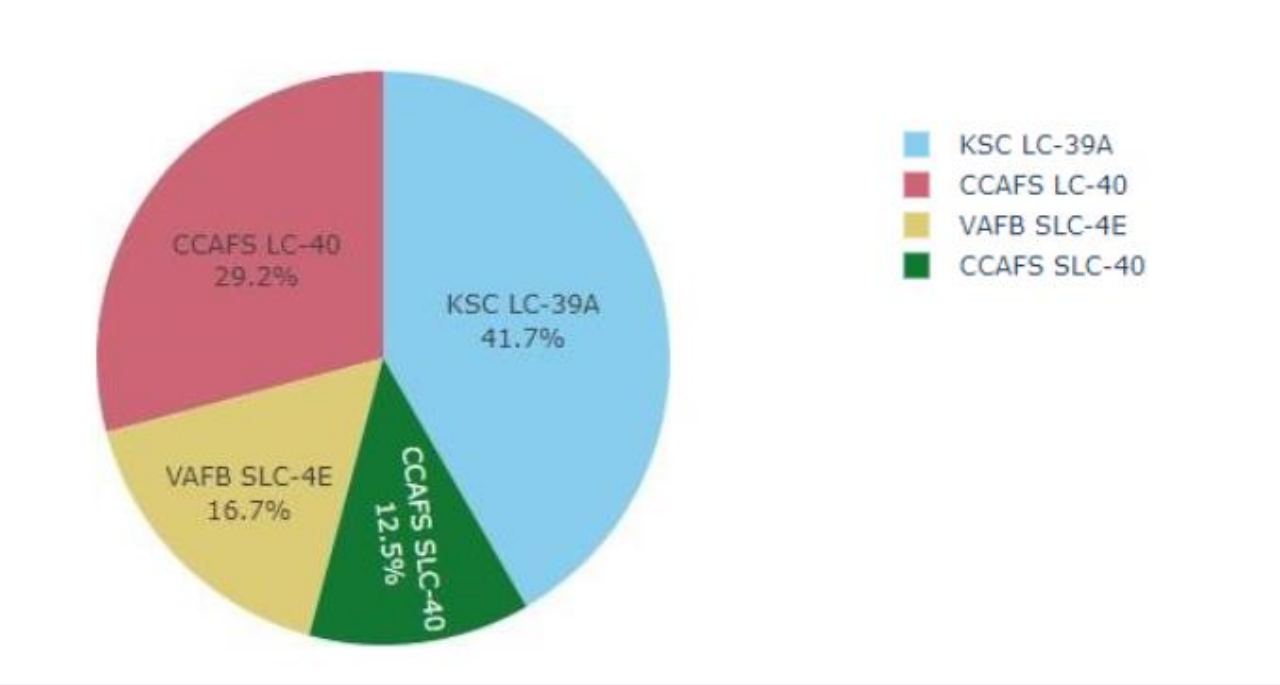
Location Proximities



Explore the generated folium map and show the screenshot of a selected launch site to its proximities such as railway, highway, coastline, with distance calculated and displayed


Section 4
Build a Dashboard with Plotly Dash

Successful Launches



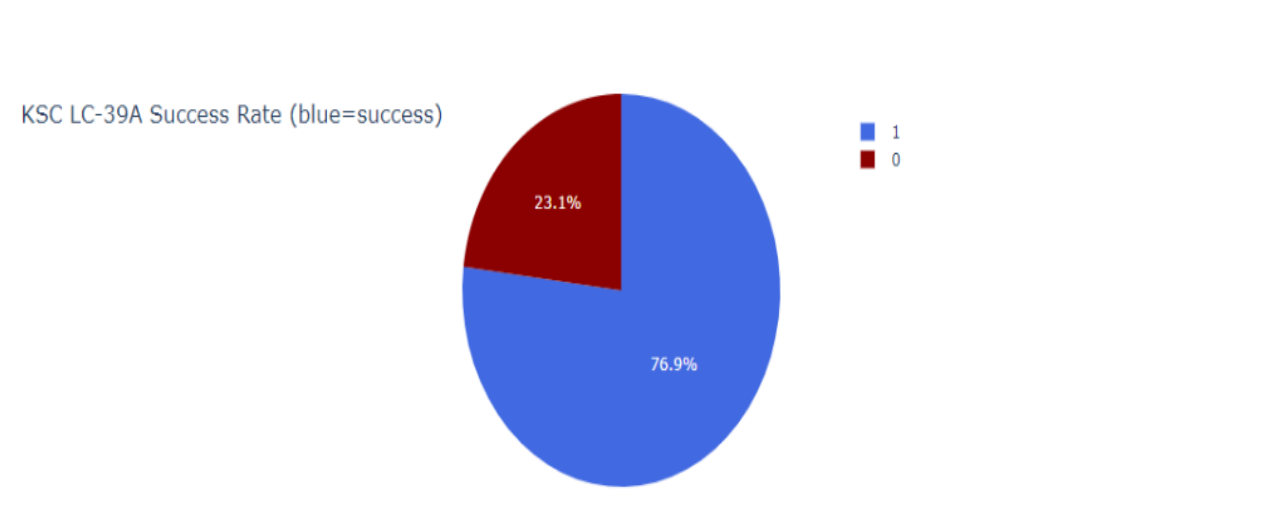
This is the distribution of successful landings across all launch sites.

CCAFS LC-40 is the former name for CCAFS SLC-40.

CCAFS and KSC have the same number of successful landings.

VAFB has the lowest proportion of successful landings.

Highest Succes Launches

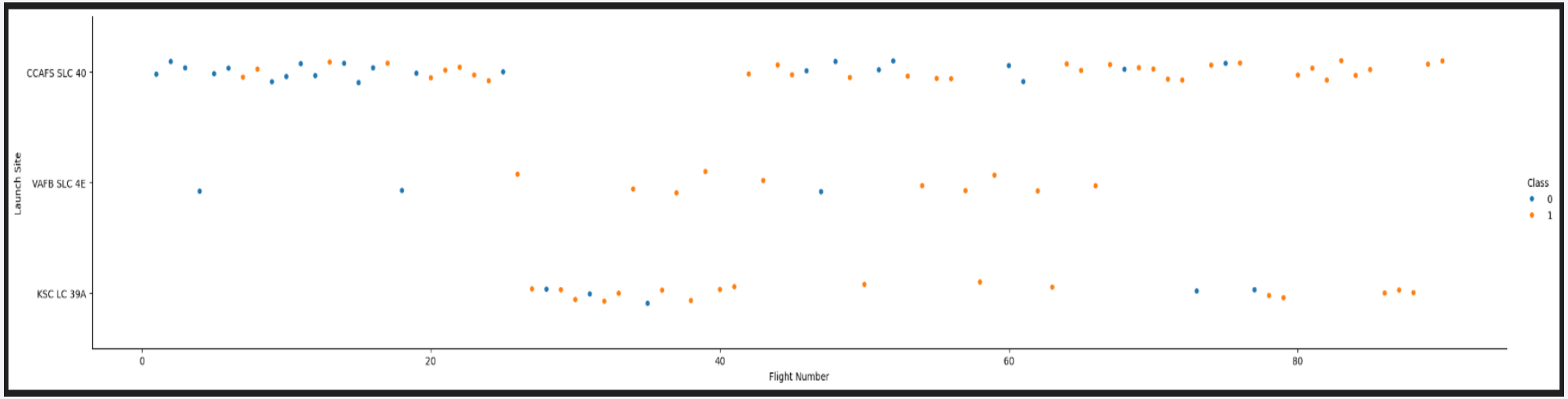


We can see that the KSC LC-39A has the highest success rate.

10 successful landings.

3 failed landings.

Payload vs Launch Outcome

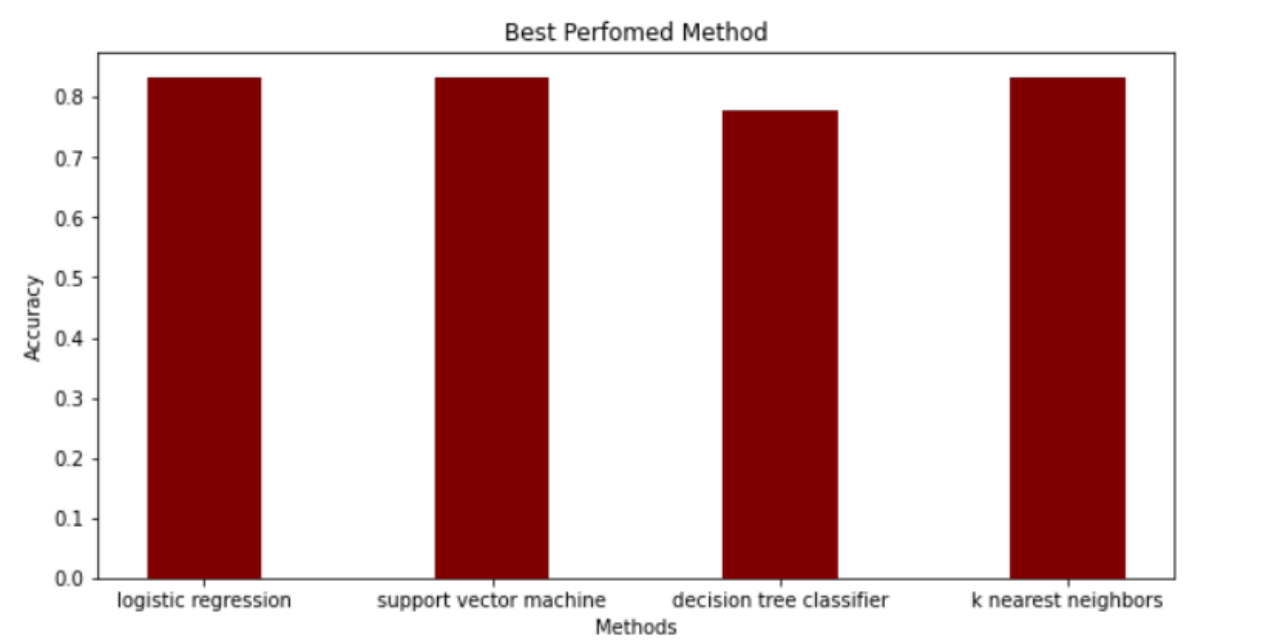


We can see the relationship between Payload vs. Launch Outcome for all sites.

Section 5

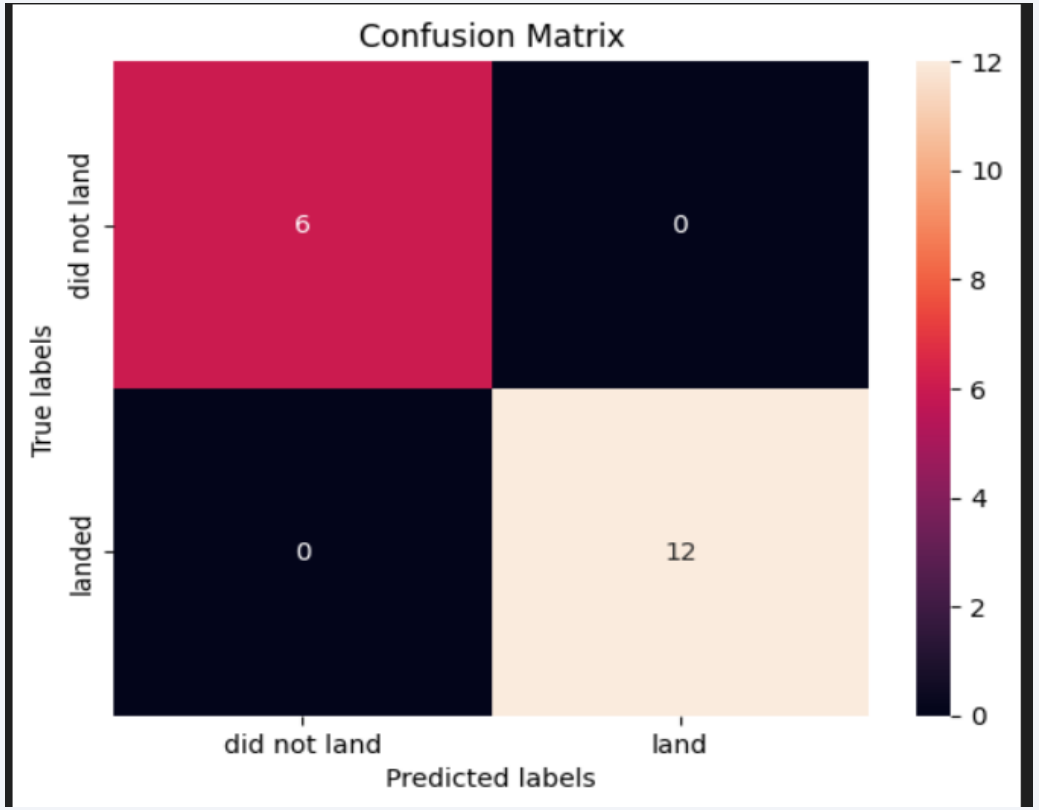
Predictive Analysis (Classification)

Classification Accuracy



- All models have the same accuracy in the tests, with an accuracy of 83.33%.
- The sample size is only 18.
- This can cause a large variation in the accuracy results.

Confusion Matrix



The confusion matrix is the same for all models, because all models performed equally for the test set.

12 successful landings when the true label was successful landing.

3 failed landings when the true label was failed landing.

Conclusions

- Point 1: We have developed a machine learning model for SpaceY, which wants to compete with SpaceX.
- Point 2: Goal of the model is to predict when Stage 1 will successfully land in order to save around $100 million.
- Point 3 :We have also used data from a public SpaceX API and extracted data from SpaceX's Wikipedia page.
- Point 4 :We created a machine learning model with an accuracy of 83.33%
- Allon Mask CEO of SpaceY can use this model to predict with relatively high accuracy whether a launch will have a successful Stage 1 landing.

Appendix

- Github Repository URL: https://github.com/VictorGonTec/Data-Science-Capstone/tree/main/Main

Thank you!