Winning Space Race with Data Science

Panagiotis Kontos
14/10/2021
https://github.com/Panoskontos

Outline

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

Executive Summary

- **In this project we have :**
- Collected data from SpaceX API and SpaceX Wikipedia page.
- We have identified the successful & unsuccessful landings.
- Explored our data using Python, SQL, visualization, folium maps, and dashboards. Furthermore, we created the necessary Changed all categorical variables to binary using one hot encoding.
- Standardized our data and with GridSearchCV we found the best parameters for our Machine Learning Classification Model order to visualize and choose the most optimal model.

- **Summary of all results**
- Four machine learning models were produced:
- Logistic Regression, Support Vector Machine, Decision Tree Classifier, and K Nearest Neighbors.
- All produced similar results with accuracy rate of about 83% with KNN producing the best one at 86%..
- All models over predicted successful landings. More data is needed for better model determination and accuracy.

Introduction

Background:

- Commercial Space Age is Here
- Space X has best pricing ($62 million vs. $165 million)
- Largely due to ability to recover part of rocket (Stage 1)
- Space Y wants to compete with Space X
- Space Y that would like to compete with SpaceX

Problem

Our job is to **determine the price** of each launch.
We will **gather information** about Space X and create dashboards for our team.
We will also **determine** if SpaceX will reuse the first stage if the first stage will land successfully.
We will **train a machine learning model** and use public information to predict if SpaceX will reuse the first stage.

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:

  We gathered data from SpaceX public API and SpaceX Wikipedia page

- Perform data wrangling

  Cleaned our Data and classify true landings as successful and unsuccessful

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

  Build 4 classification models and tuned them using GridSearchCV

Data Collection

We used a combination of API requests from :

- Space X public API
- Web scraping data from Space X's Wikipedia Page.

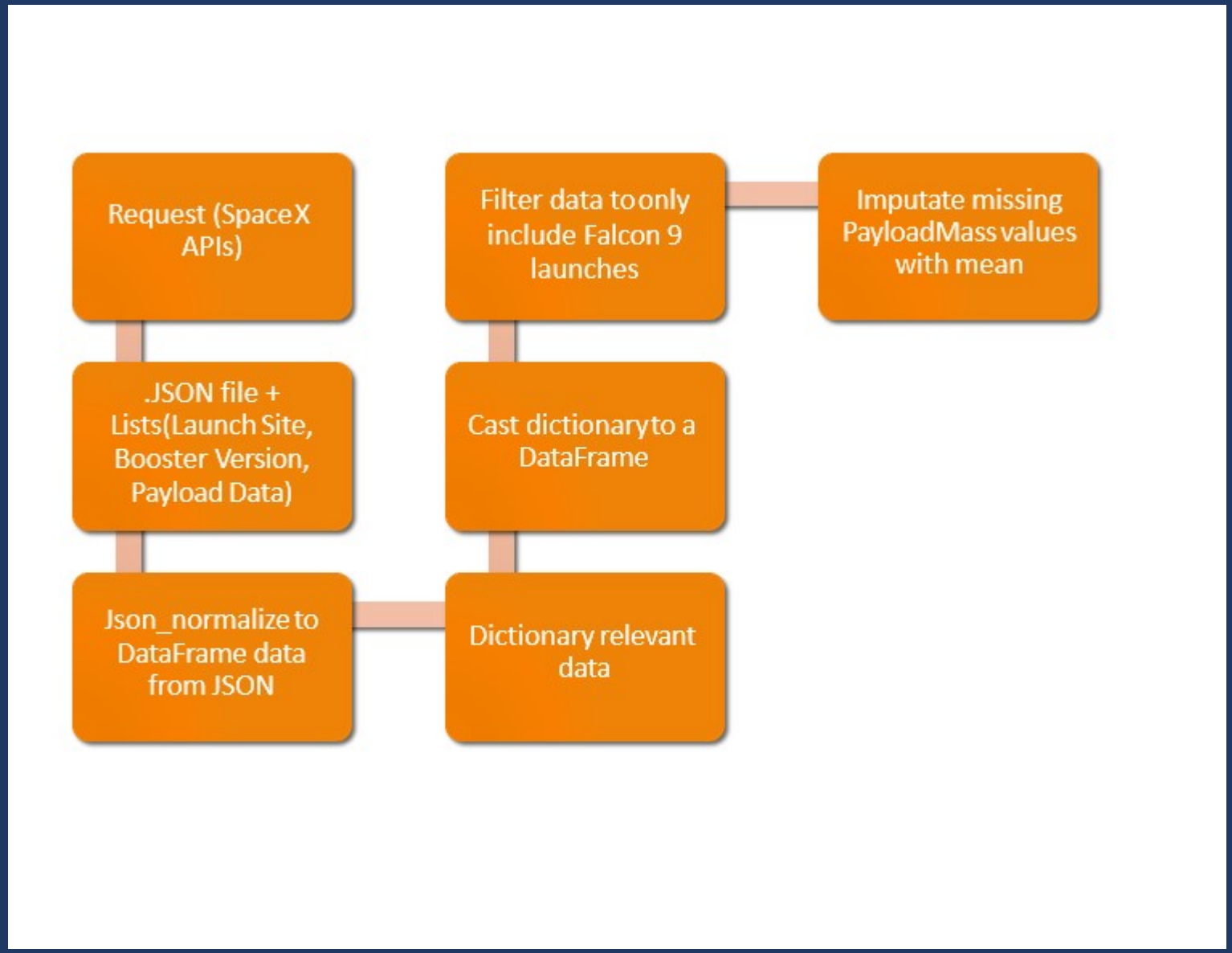
The next slides will show:

- The flowchart of data collection from API
- The flowchart of data collection from webscraping.

Space X API Data Columns:

FlightNumber, Date,
BoosterVersion,
PayloadMass, Orbit,
LaunchSite, Outcome,
Flights, GridFins,
Reused, Legs, LandingPad,
Block, ReusedCount, Serial,
Longitude, Latitude

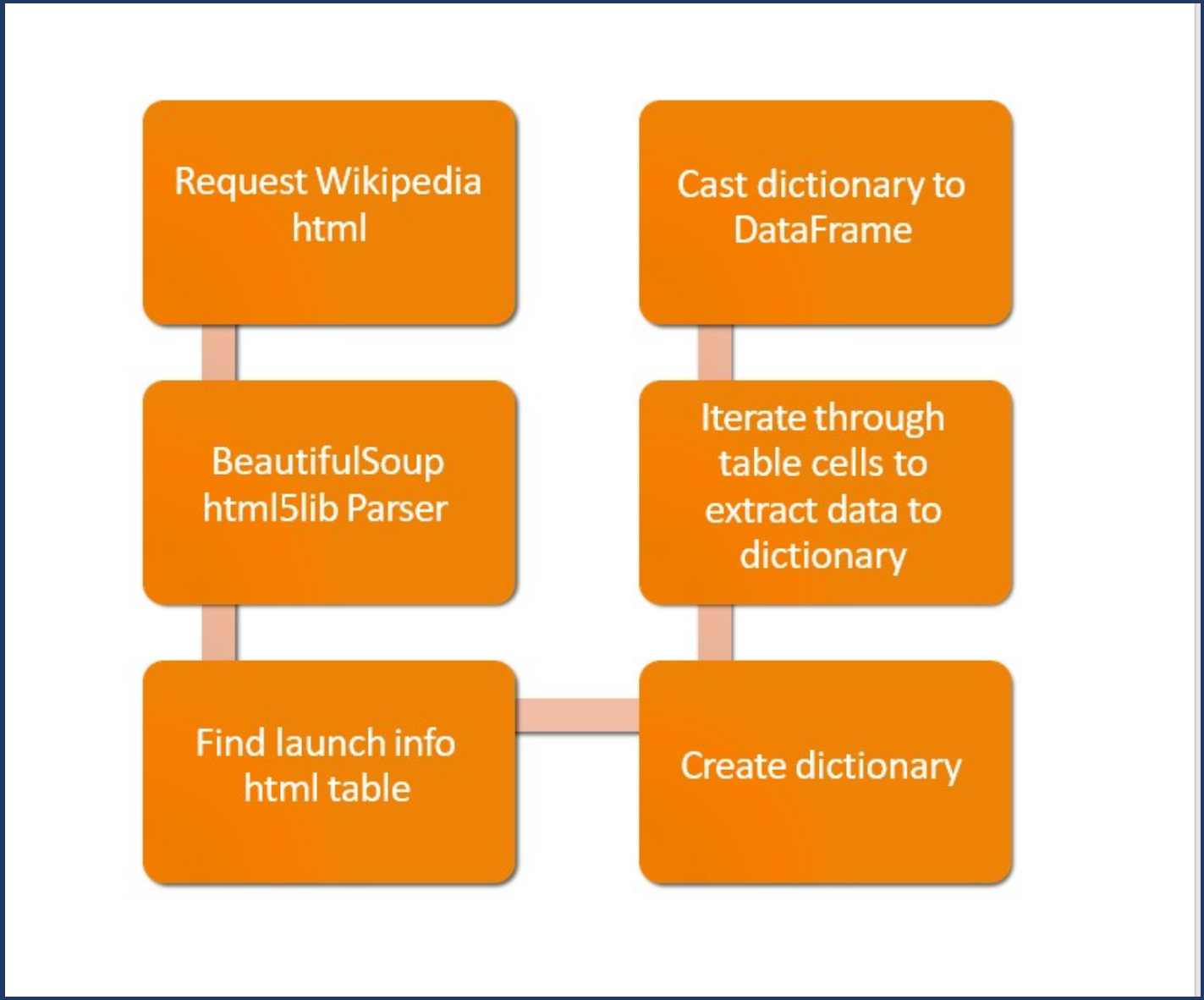
GitHub URL



Wikipedia Webscraping Data Columns:

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

GitHub URL



Data Wrangling

- Replace Nan values with the mean
- Create a training label with landing outcomes where successful = 1 & failure = 0.
- Outcome column has two components: ‘Mission Outcome’ ‘Landing Location’
- New training label column ‘class’ with a value of 1 if ‘Mission Outcome’ is True and 0 otherwise.
- Value Mapping:
  - True ASDS, True RTLS, & True Ocean – -> 1
  - None None, False ASDS, None ASDS, False Ocean, False RTLS --> 0

See more on GitHub

EDA with Data Visualization

Exploratory Data Analysis performed on variables:

- Flight Number
- Payload Mass
- Launch Site
- Orbit
- Class and Year.

GitHub URL

Plots Used:

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 Success Yearly Trend

Scatter plots, line charts, and bar plots were the best tool to use in order to compare relationships and find correlation between variables so that they could be used in training the machine learning model

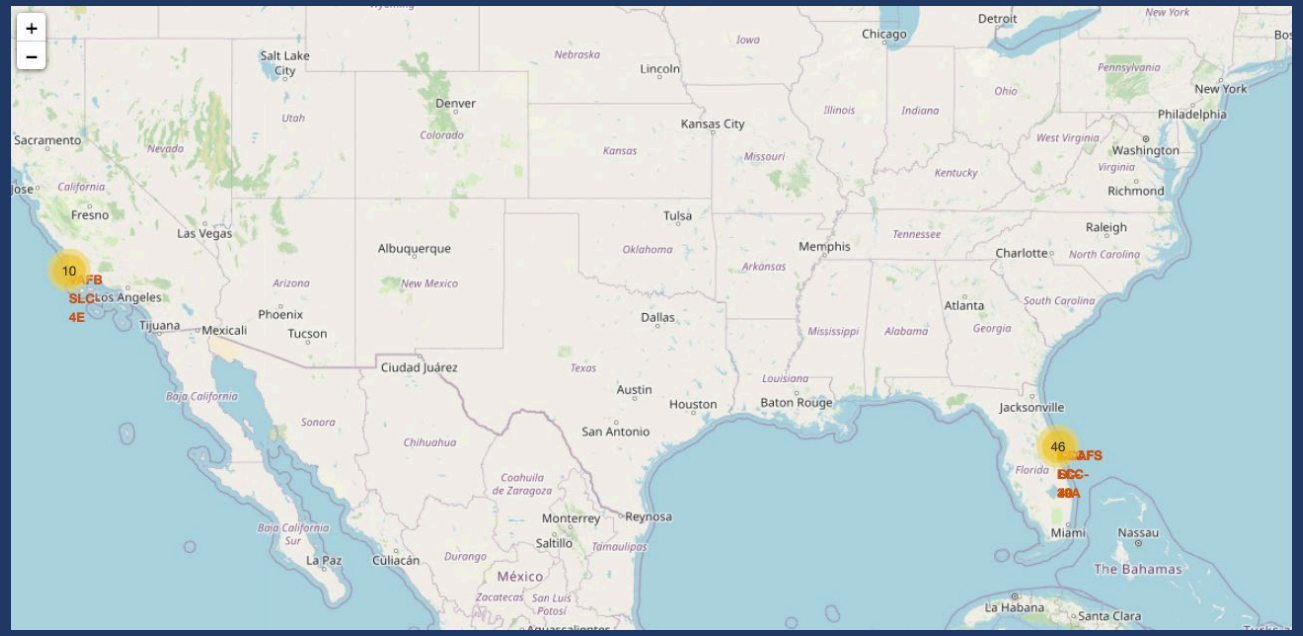
EDA with SQL

- Loaded data set into IBM DB2 Database.
- Queried using SQL Python integration.
- Queries were made to get a better understanding of the dataset.
- Queried information about launch site names, mission outcomes, various pay load sizes of customers and booster versions, and landing outcomes

GitHub URL

Build an Interactive Map with Folium

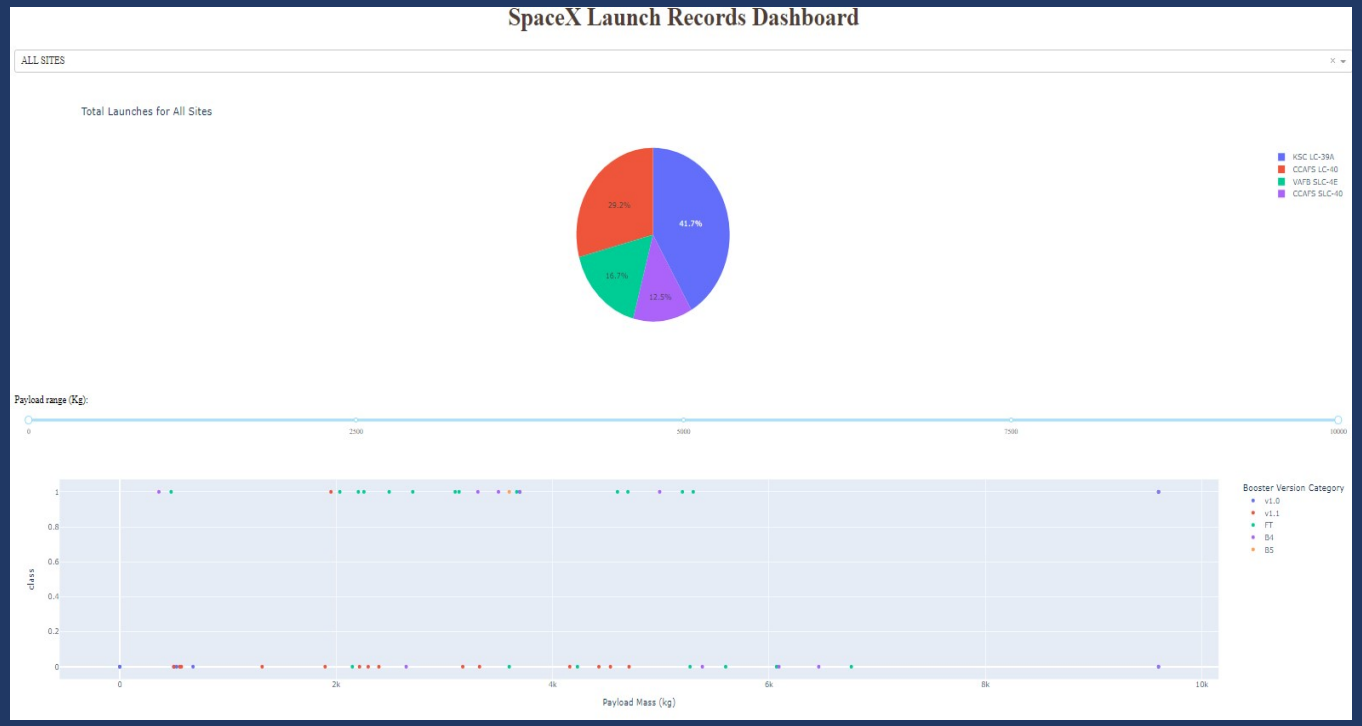
- Folium maps mark Launch Sites, successful and unsuccessful landings, and a proximity example to key locations: Railway, Highway, Coast, and City.
- This allows us to understand why launch sites may be located and **where** they are.
- Also we **visualized** successful landings relative to location.



GitHub URL

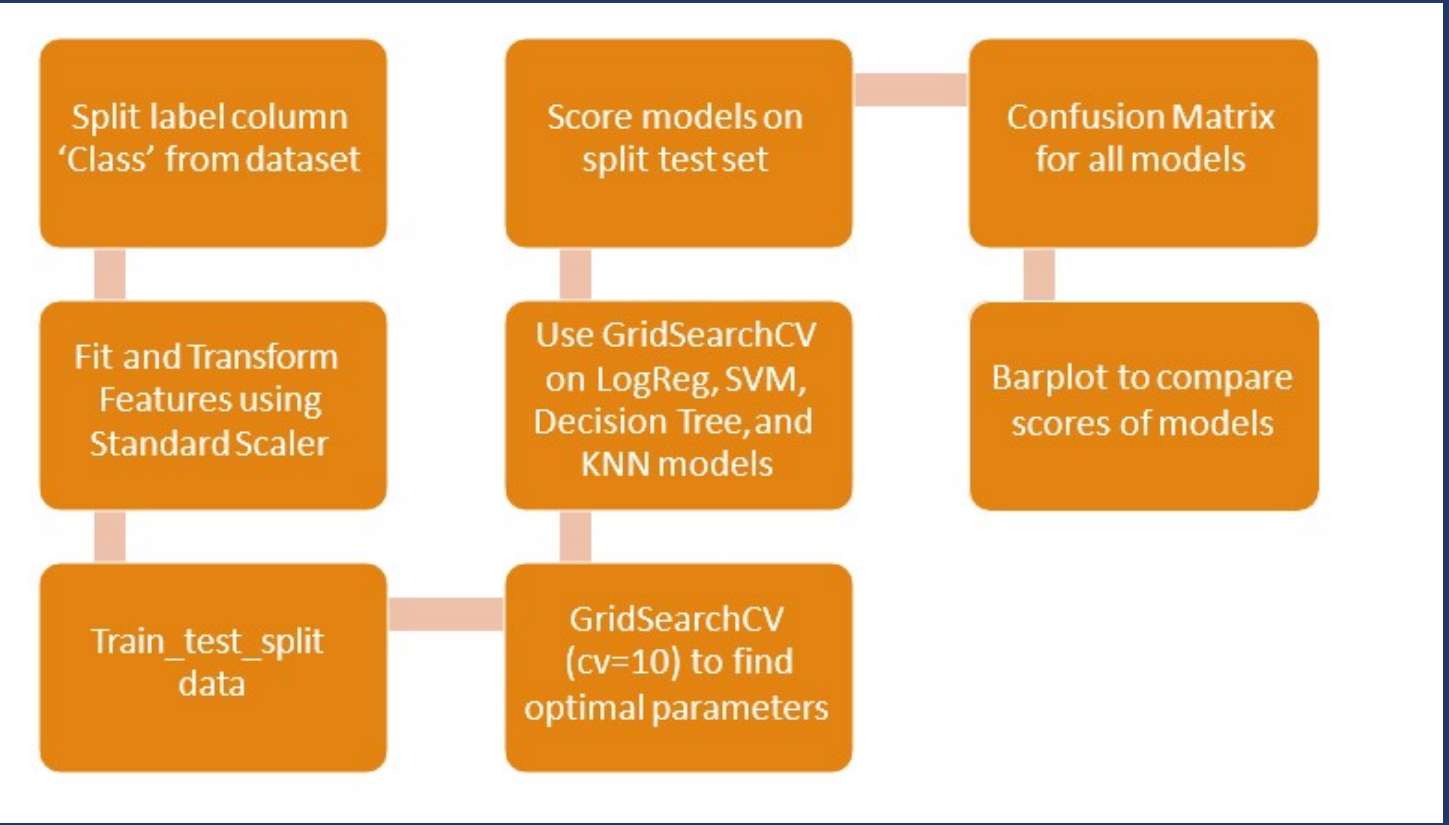
Build a Dashboard with Plotly Dash

- Dashboard includes a **pie chart** and a **scatter plot**.
- Pie chart can be selected to show distribution of successful landings across all launch sites and can be selected to show individual launch site success rates.
- Scatter plot takes two inputs: All sites or individual site and payload mass on a slider between 0 and 10000 kg.
- The pie chart is used to visualize launch site success rate.
- The scatter plot can help us see how success varies across launch sites, payload mass, and booster version category.

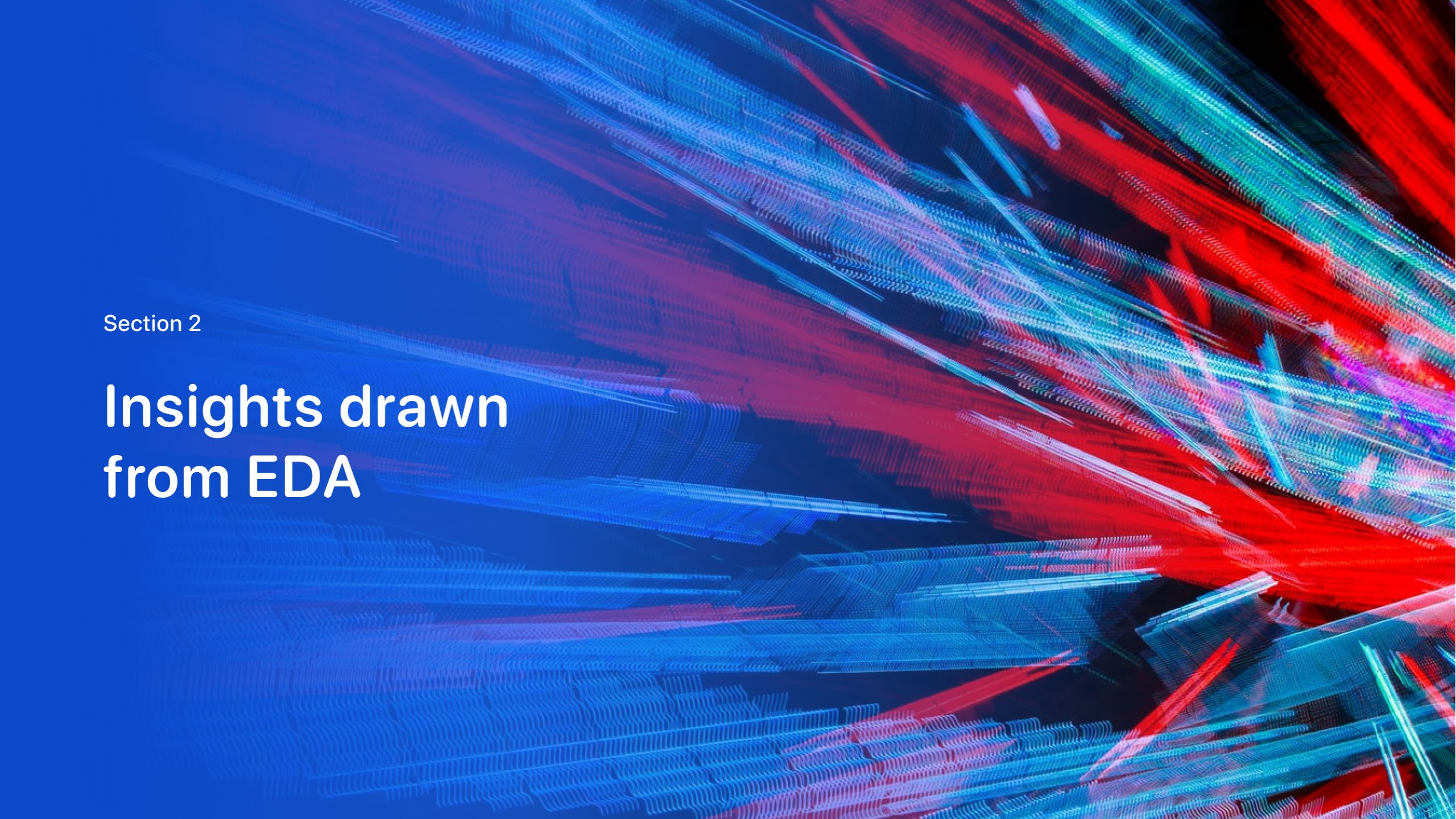


GITHUB URL

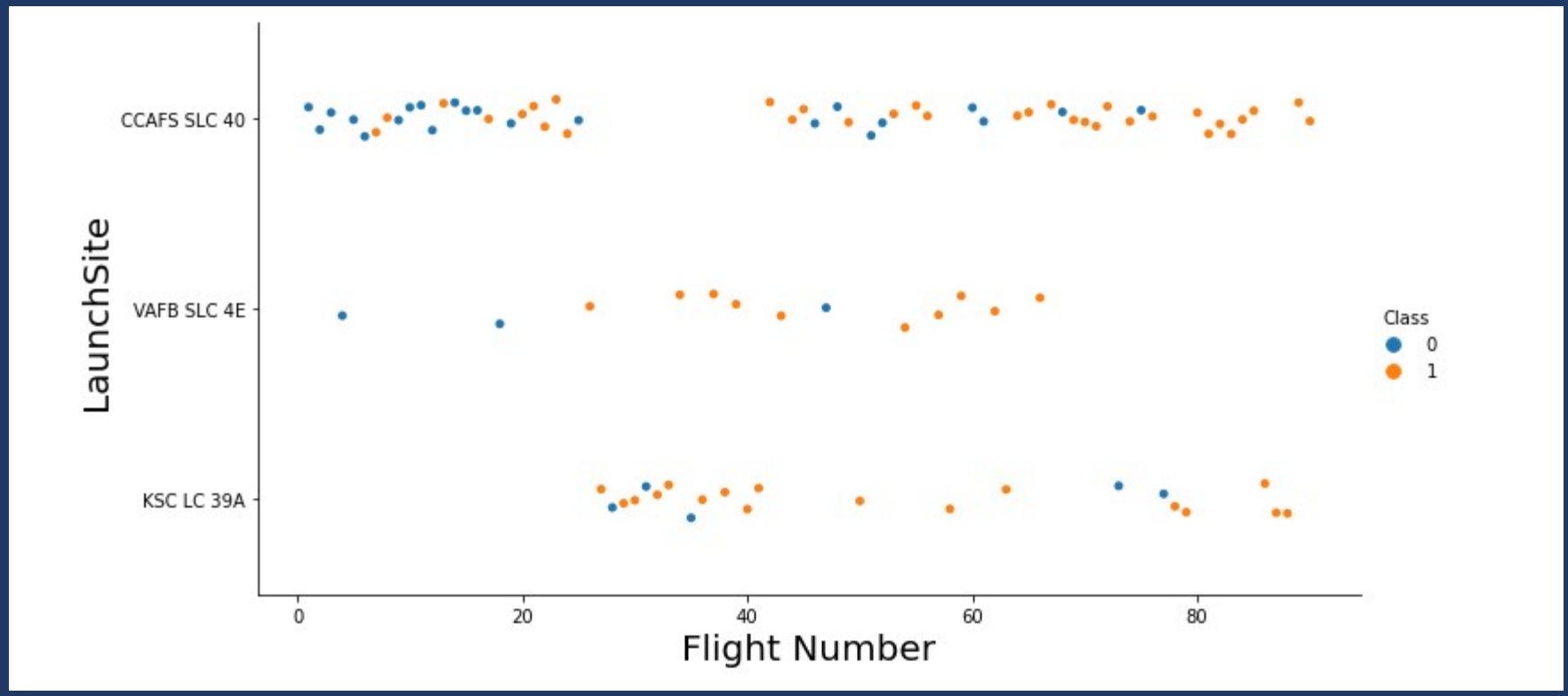
Predictive Analysis (Classification)



GITHUB URL

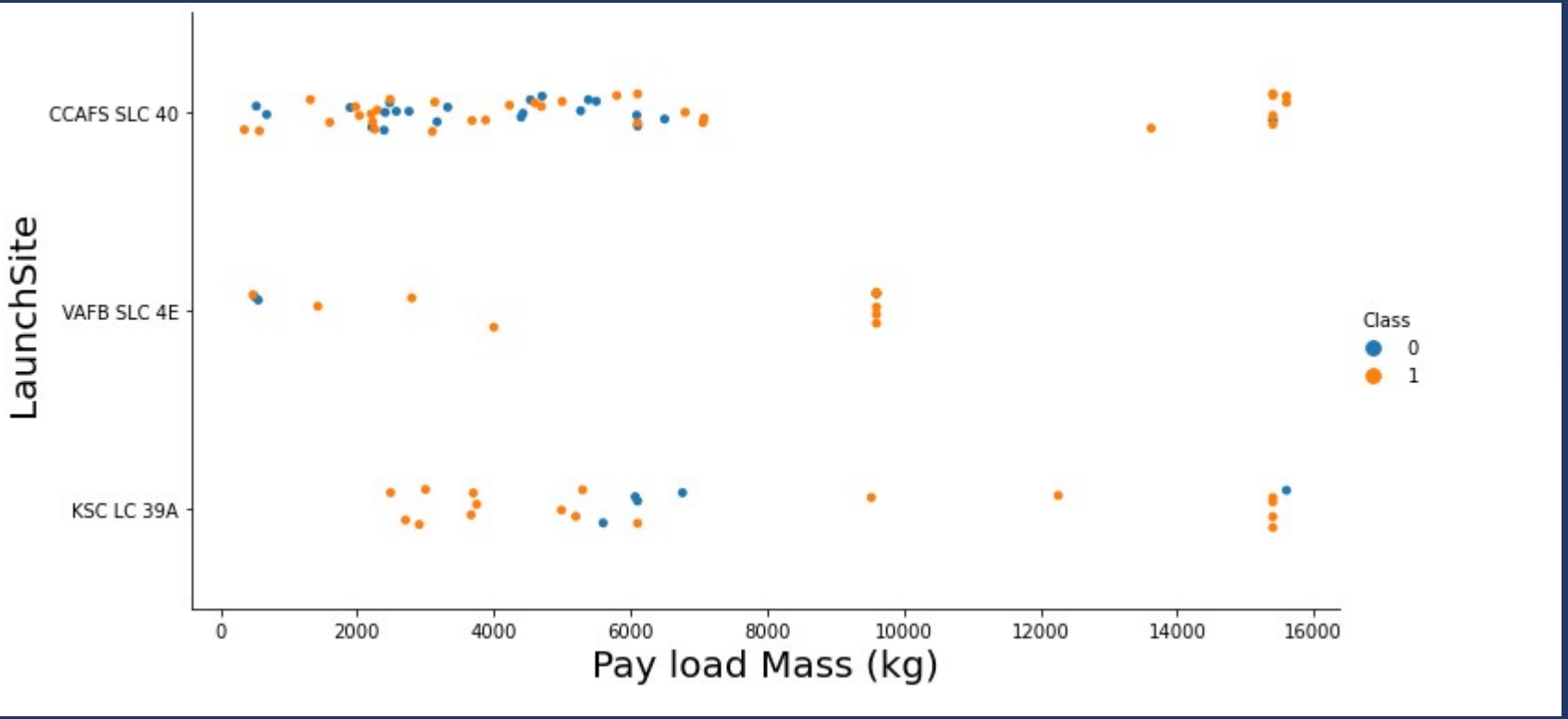

Section 2
Insights drawn
from EDA

Flight Number vs. Launch Site



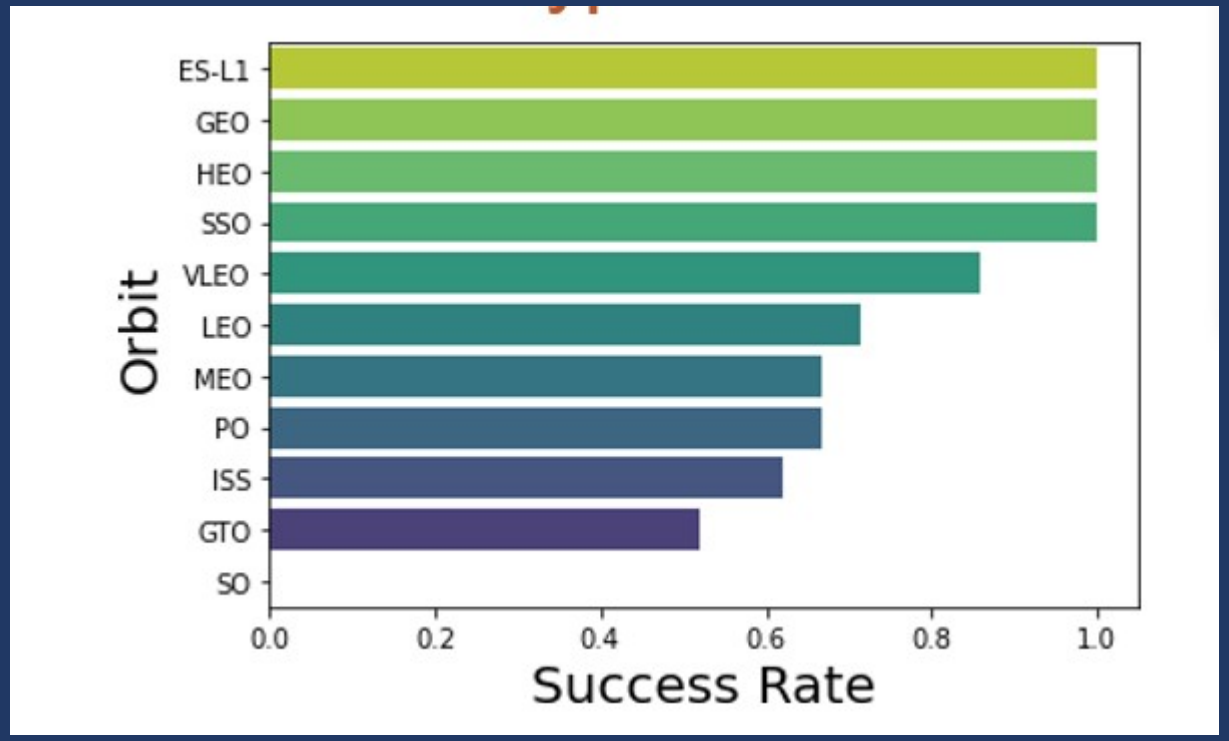
Graphic suggests an increase in success rate over time (indicated in Flight Number). Likely a big breakthrough around flight 20 which significantly increased success rate. CCAFS appears to be the main launch site as it has the most volume.

Payload vs. Launch Site



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



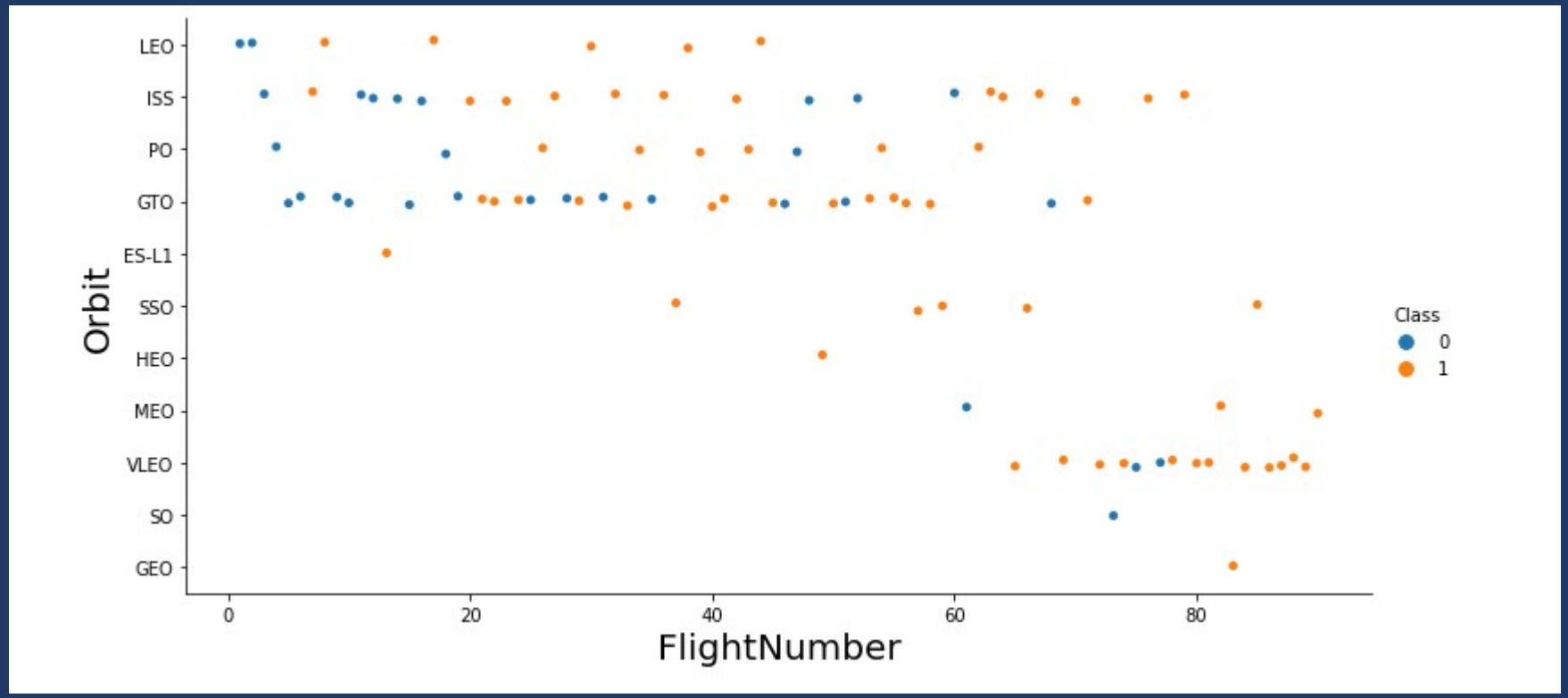
ES-L1 , GEO , HEO , SSO have 100% success rate

VLEO has decent success rate and attempts

SO has 0% success rate

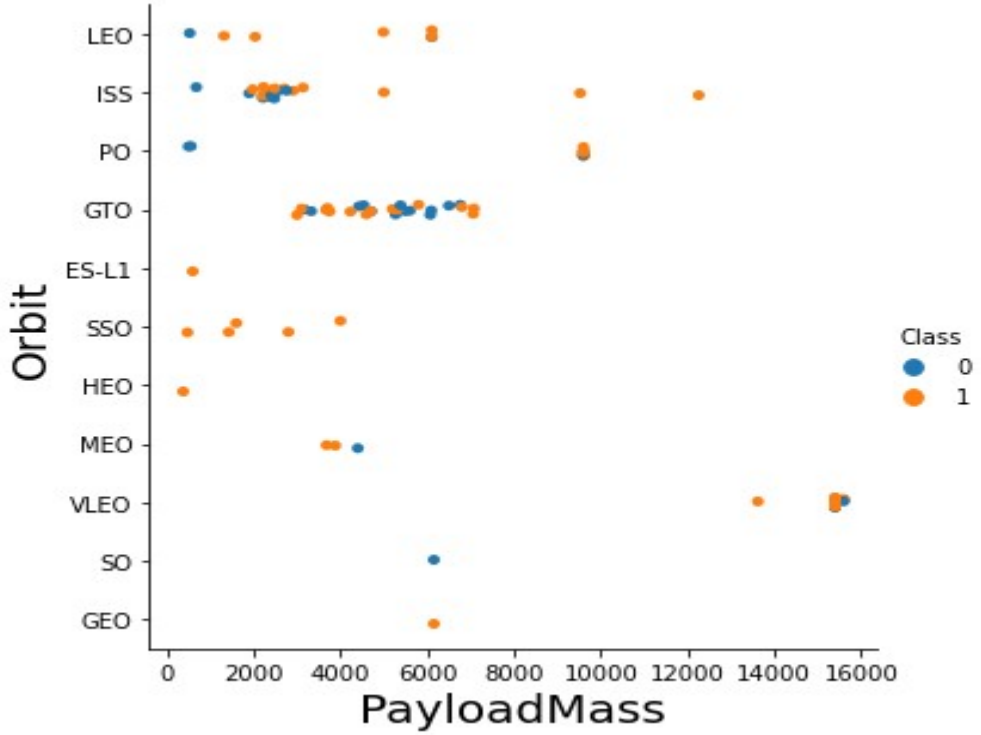
GTO has moderate success rate

Flight Number vs. Orbit Type



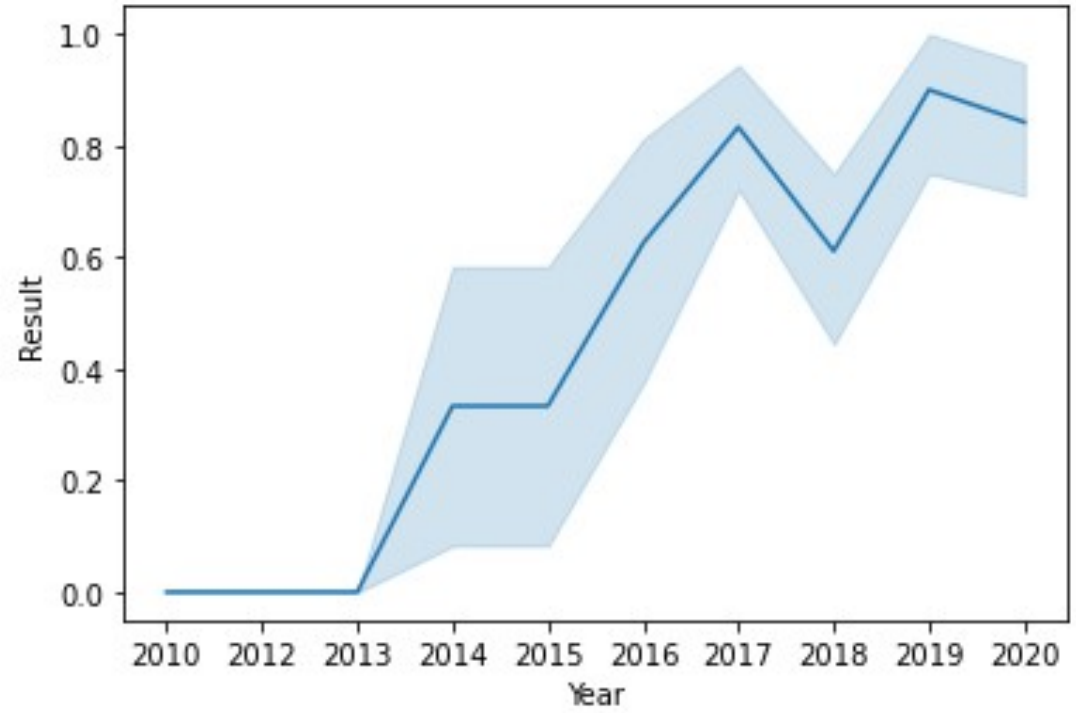
We see that in the LEO orbit the Success appears related to the number of flights; on the other hand, there seems to be no relationship between flight number when in GTO orbit.

Payload vs. Orbit Type



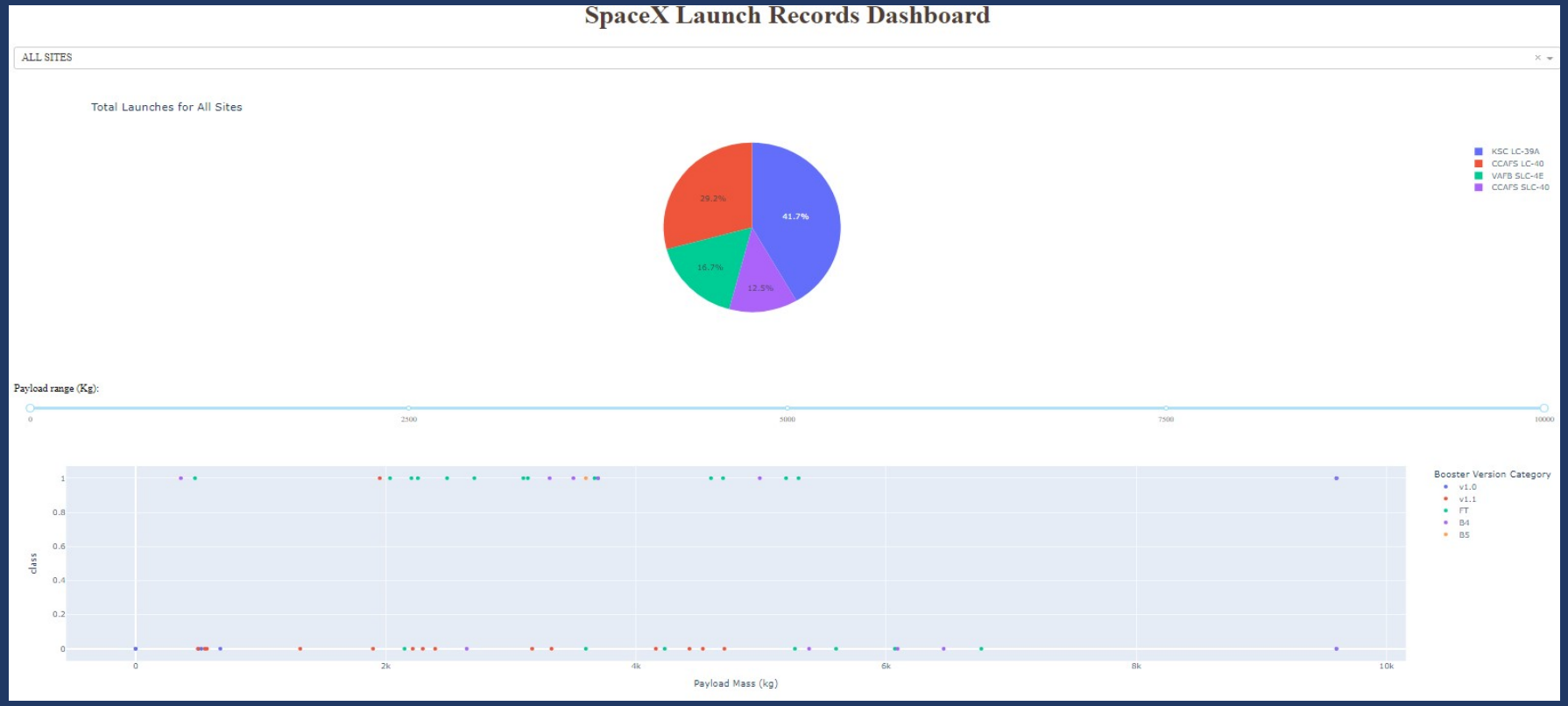
With heavy payloads the successful landing or positive landing rate are more for Polar, LEO and ISS. However for GTO we cannot distinguish this well as both positive landing rate and negative landing(unsuccessful mission) are both there here.

Launch Success Yearly Trend



We observe that the success rate since 2013 kept increasing till 2020
It is also evident that last years success rate is approximately 80%

Results



This is a preview of the Plotly dashboard. The following sides will show the results of EDA with visualization, EDA with SQL, Interactive Map with Folium, and finally the results of our model with about 83% accuracy.

Launch Site Names Begin with 'CCA'

```
In [4]: %sql SELECT DISTINCT launch_site from SPACEX1
 * ibm_db_sa://rmx39209:***@ba99a9e6-d59e-4883-8fc0-d6a8c9f7a08f.c1ogj3sd0tgtu0lqc
Done.
```

Out[4]:

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

As the result illustrate the unique launch sites are 4

5 Records where launch sites begin with the 'CCA'

```
In [19]: %sql SELECT * from SPACEX1 WHERE launch_site LIKE 'CCA%' LIMIT 5
 * ibm_db_sa://rmx39209:***@ba99a9e6-d59e-4883-8fc0-d6a8c9f7a08f.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud:31321/BLUDB
Done.
```

Out[19]:

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	07:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success	No attempt
2012-10-08	00:35:00	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

All five of the records has the same launch site and we also notice the different info we can get from the table.

Total Payload Mass

```
In [6]: %sql SELECT SUM(payload_mass__kg_) FROM SPACEX1 WHERE customer = 'NASA (CRS)'
 * ibm_db_sa://rmx39209:***@ba99a9e6-d59e-4883-8fc0-d6a8c9f7a08f.c1ogj3sd0tgtu0lqde00.da
Done.
Out[6]:
1
45596
```

Total payload mass in kg where customer is NASA (CRS)
CRS stands for *Cargo Resupply*

Average Payload Mass by F9 v1.1

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

```
In [7]: %sql SELECT AVG(payload_mass__kg_) FROM SPACEX1 WHERE booster_version = 'F9 v1.1'
 * ibm_db_sa://rmx39209:***@ba99a9e6-d59e-4883-8fc0-d6a8c9f7a08f.c1ogj3sd0tgtu0lqde00.databas
Done.
Out[7]:
1
2928
```

It seems like version F9 v1.1 carries on average 2928 kg

First Successful Ground Landing Date

List the date when the first successful landing outcome in ground pad was acheived.

Hint:Use min function

```
In [8]: %sql SELECT MIN(DATE) FROM SPACEX1 WHERE landing__outcome = 'Success (ground pad)'
```

```
 * ibm_db_sa://rmx39209:***@ba99a9e6-d59e-4883-8fc0-d6a8c9f7a08f.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud:31321/BLUDB
Done.
```

Out[8]:

1
2015-12-22

After years of hard work in 22-12-2015 first successful landing with ground pad was achieved.

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 [9]:
```
%sql SELECT booster_version FROM SPACEX1 WHERE landing__outcome = 'Success (drone ship)' AND payload_mass__kg_ BETWEEN 4000 AND 6000
```

```
 * ibm_db_sa://rmx39209:***@ba99a9e6-d59e-4883-8fc0-d6a8c9f7a08f.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud:31321/BLUDB
Done.
```

Out[9]:

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

Only four boosters with moderate Payload have landed successfully in drone ship

Total Number of Successful and Failure Mission Outcomes

```
In [10]: %sql SELECT mission_outcome,COUNT(*) AS numbers FROM SPACEX1 GROUP BY mission_outcome
 * ibm_db_sa://rmx39209:***@ba99a9e6-d59e-4883-8fc0-d6a8c9f7a08f.c1ogj3sd0tgtu0lqde00.databases.ap
Done.
```

Out[10]:

mission_outcome	numbers
Failure (in flight)	1
Success	99
Success (payload status unclear)	1

99% of missions where a success with only 1 % being a failures.

Boosters Carried Maximum Payload

```
In [11]: %sql SELECT booster_version FROM SPACEX1 where payload_mass__kg_ = (SELECT MAX(payload_mass__kg_) FROM SPACEX1)
 * ibm_db_sa://rmx39209:***@ba99a9e6-d59e-4883-8fc0-d6a8c9f7a08f.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud:31321/BLUDB
Done.
```

Out[11]:

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

The results showed us that 12% of Boosters carried maximum weight with most of them being type 4.

2015 Launch Records

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

In [12]:
```
%sql SELECT booster_version,launch_site* FROM SPACEX1 WHERE landing__outcome = 'Failure (drone ship)' AND YEAR(DATE) = 2015
```

```
 * ibm_db_sa://rmx39209:***@ba99a9e6-d59e-4883-8fc0-d6a8c9f7a08f.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud:31321/BLUDB
Done.
```

Out[12]:

booster_version	launch_site
F9 v1.1 B1012	CCAFS LC-40
F9 v1.1 B1015	CCAFS LC-40

In 2015 it is noticed that there were two landing outcomes in drone ship.

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

```
In [13]: %sql SELECT landing__outcome,COUNT(*) AS NUMBERS FROM SPACEX1 WHERE DATE>'2010-06-04' AND  DATE < '2017-03-20' GROUP BY landing__outcome ORDER BY NUMBERS DESC
 * ibm_db_sa://rmx39209:***@ba99a9e6-d59e-4883-8fc0-d6a8c9f7a08f.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud:31321/BLUDB
Done.
```

Out[13]:

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

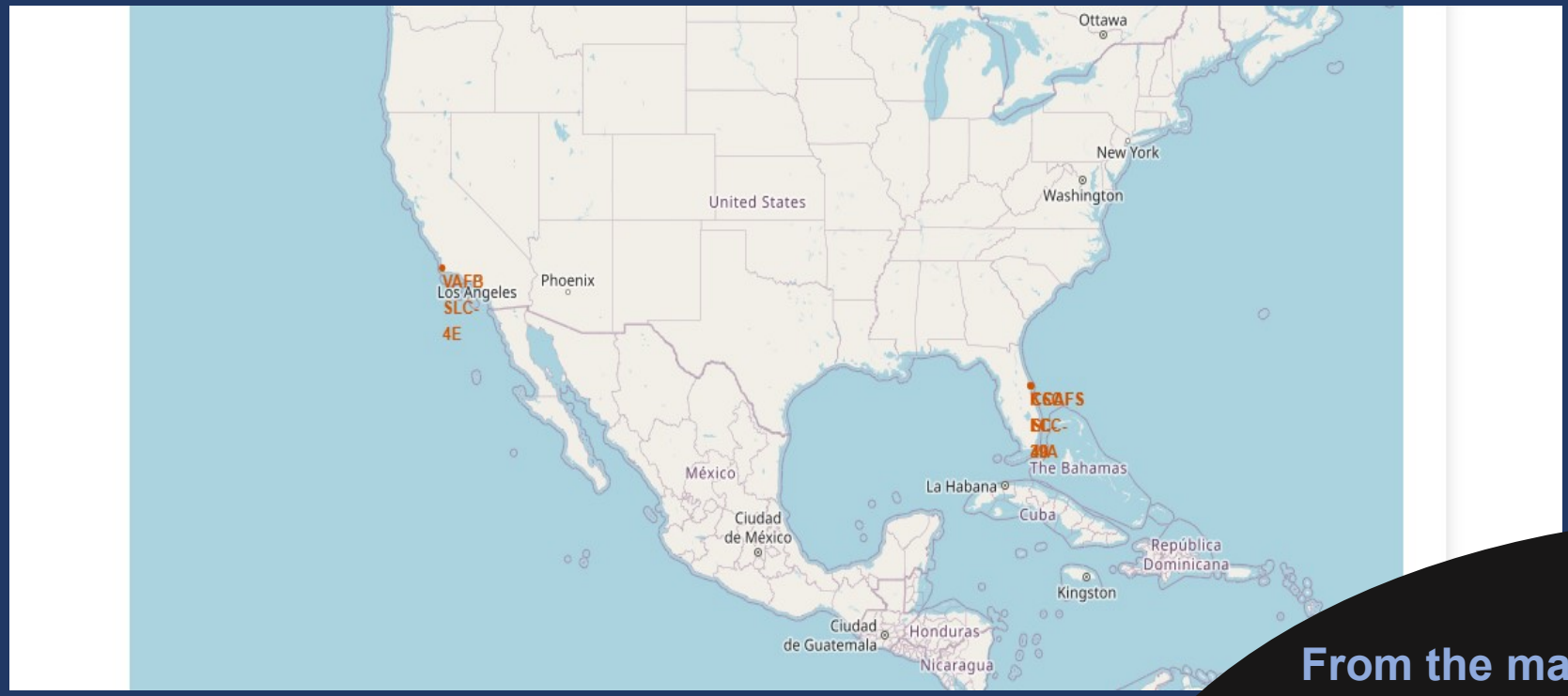
This data frame represents the landing outcomes between 2010-06-04 and 2017-03-20 .

It is evident that most outcomes were ‘No attempt’ with drone ship landings being also popular.

Section 4

Launch Sites Proximities Analysis

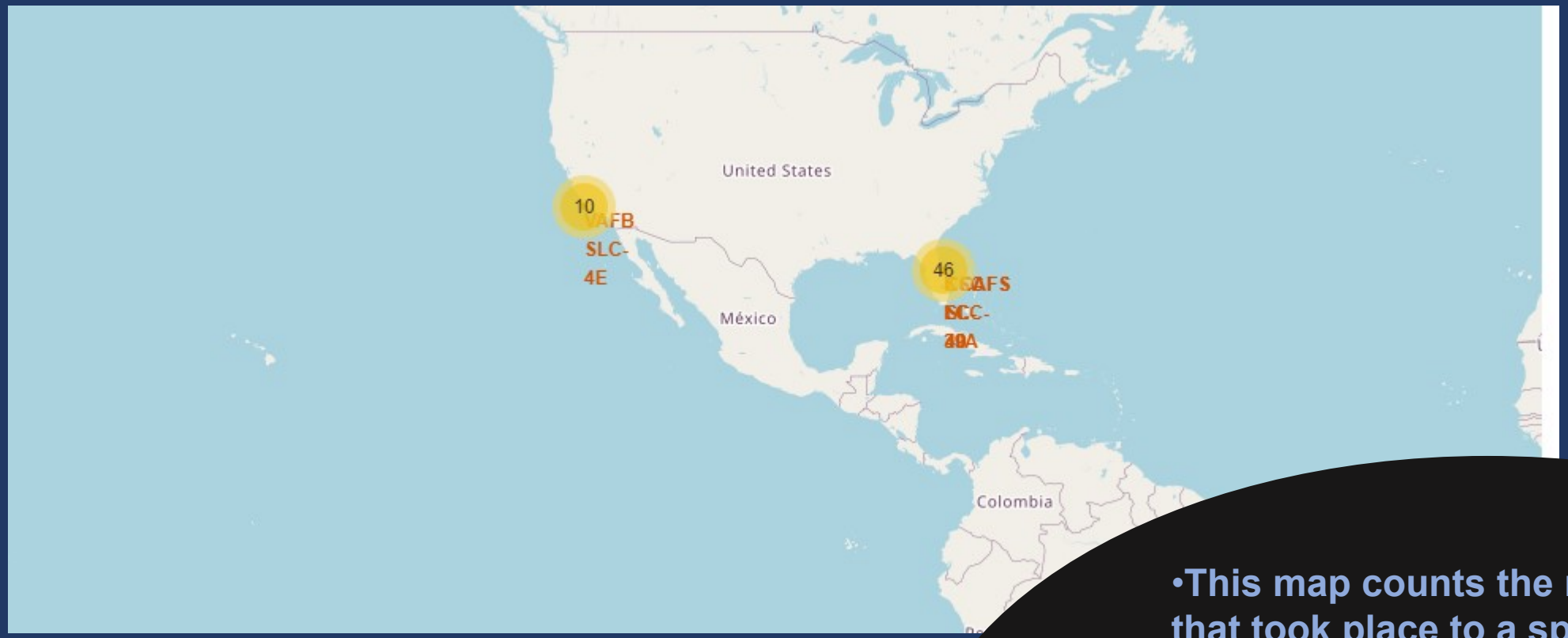
Launch Sites Locations



From the map we notice the following things:

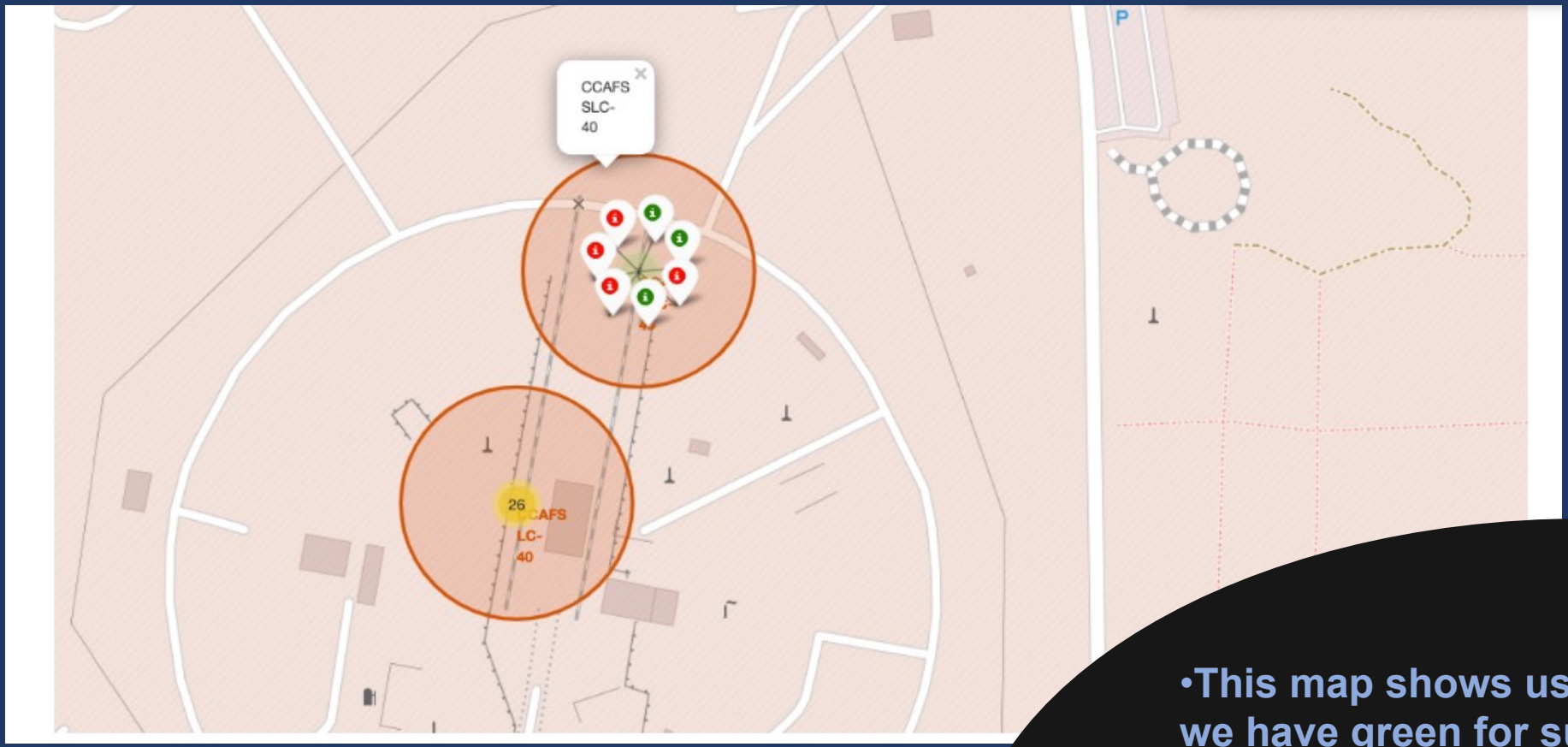
- All launch sites are in proximity to the Equator line
- All launch sites are in very close proximity to the coast for safety reasons in the event of failure

Number of Lunches



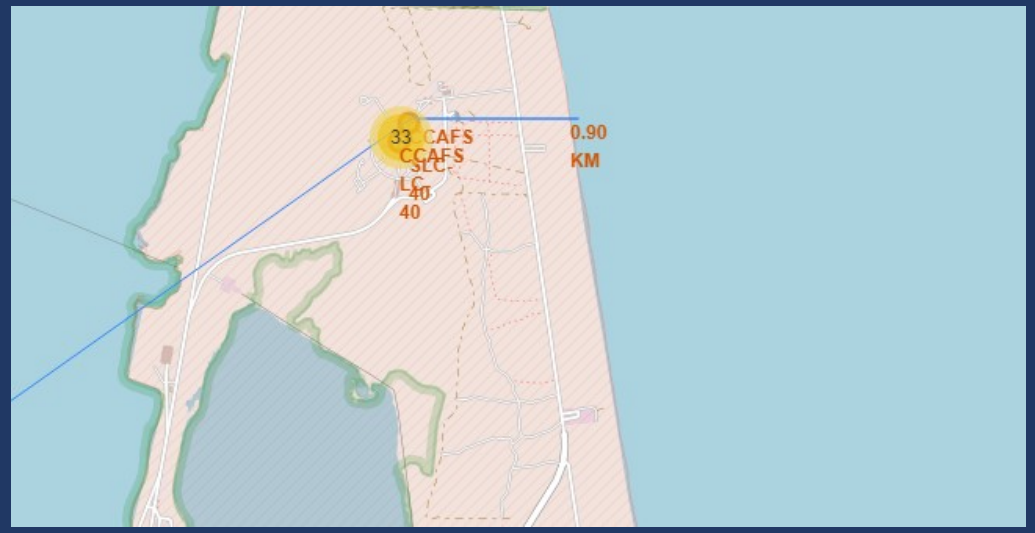
- This map counts the number of lunches that took place to a specific area
- It is clear that most lunches 80% take place in East Coast

Lunches Outcomes

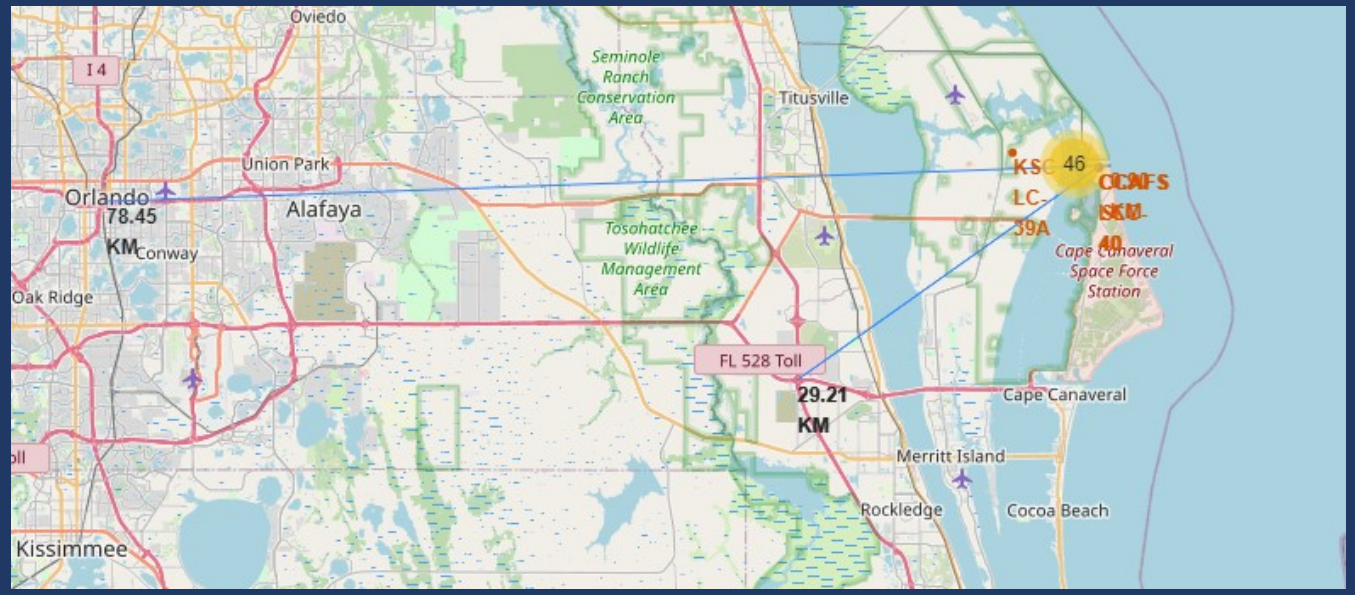


- This map shows us the lunch outcome we have green for success and red for failure

Launch sites proximities

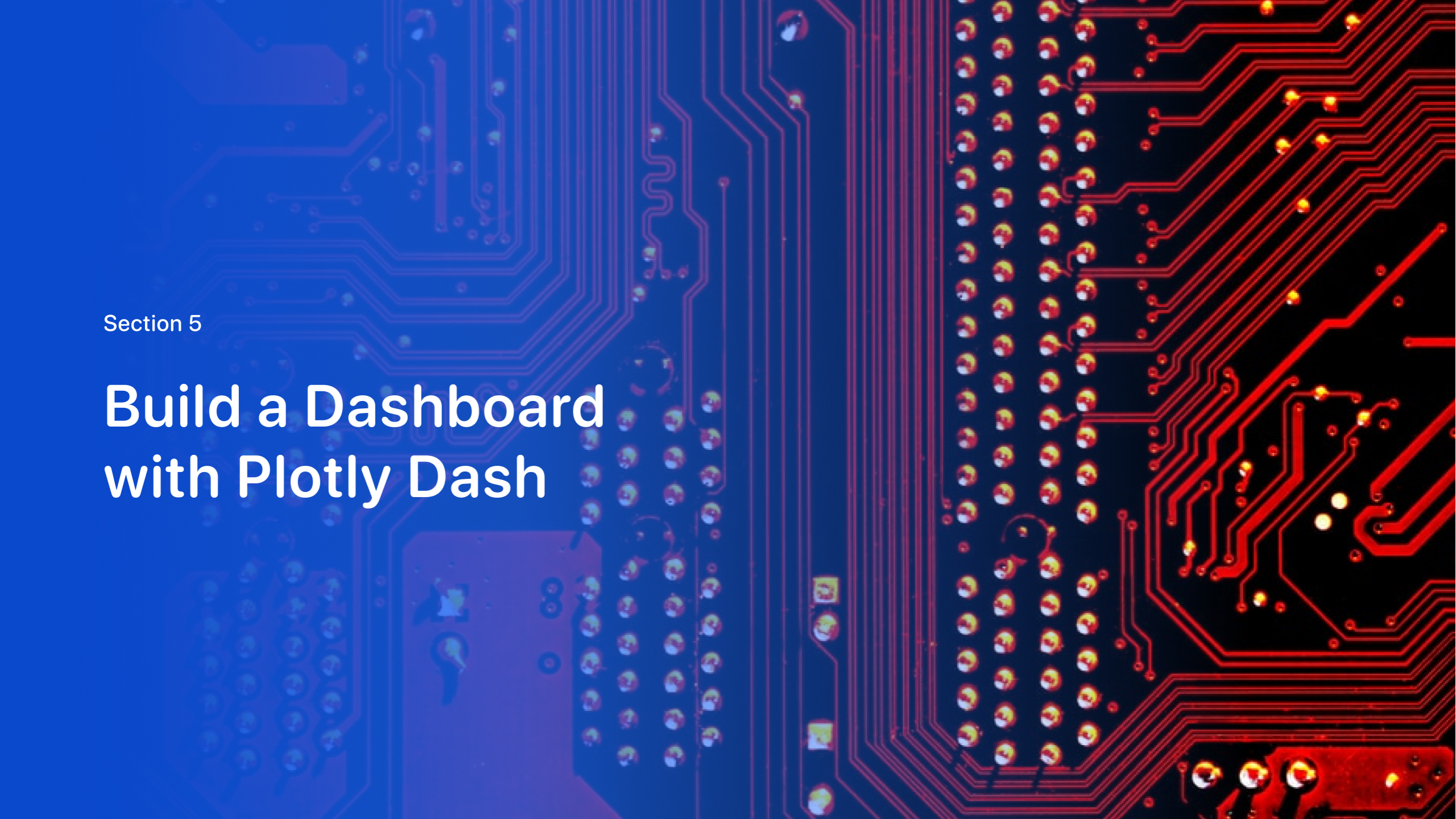


Distance to coastline – 0.9 KM



Distance to Highway– 29.21 KM

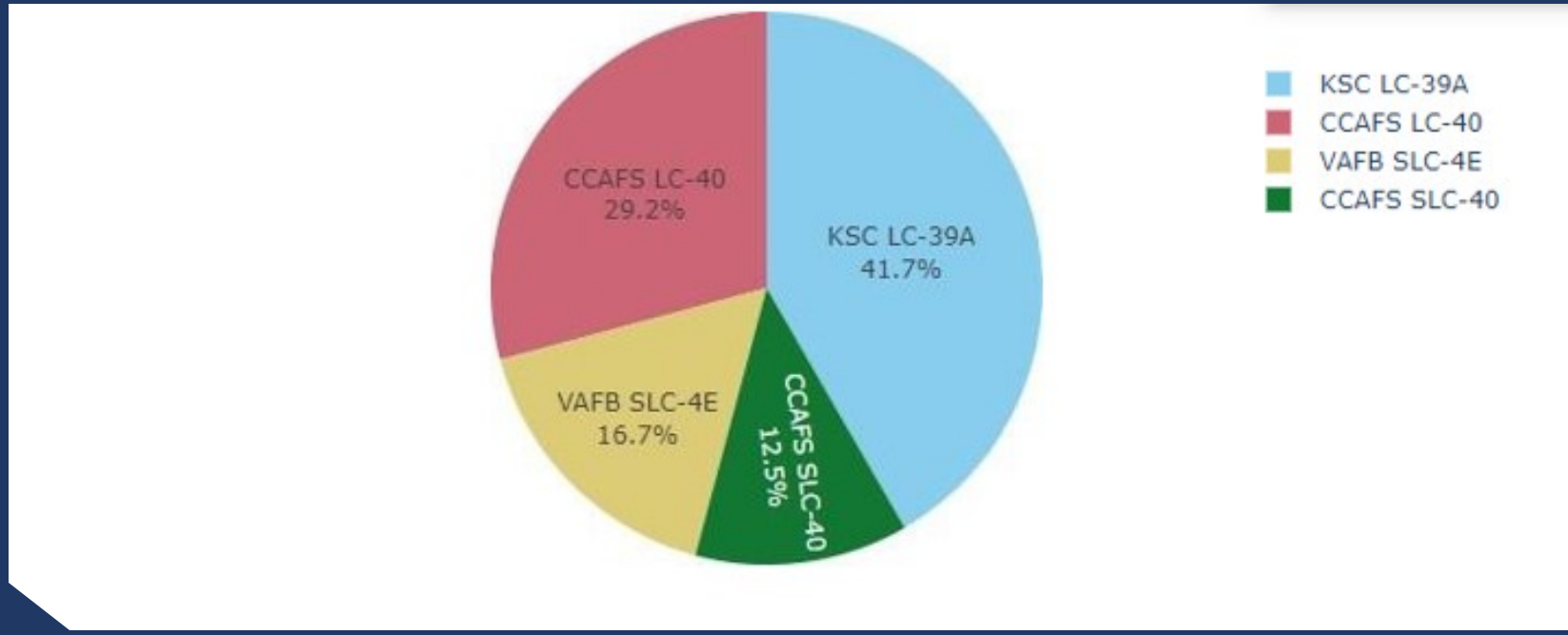
Distance to City– 78.45 KM



Section 5

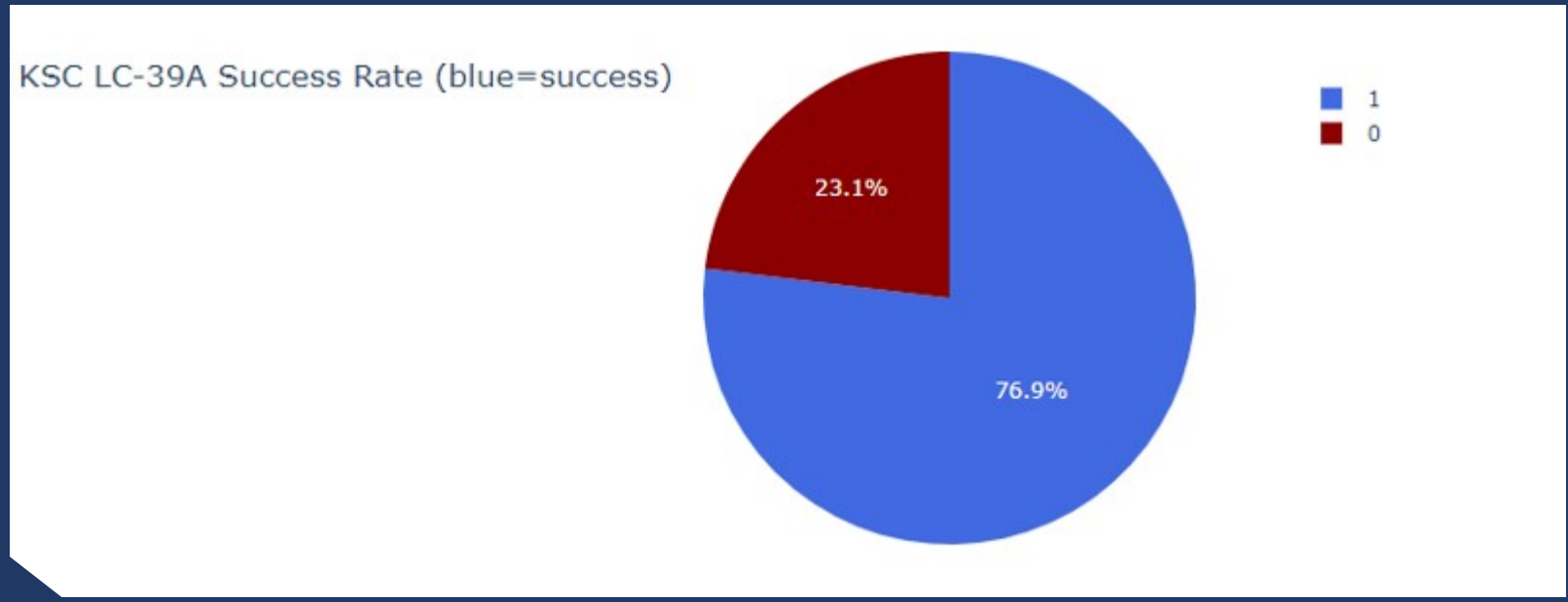
Build a Dashboard with Plotly Dash

Successful Launches Across Launch Sites



This is the distribution of successful landings across all launch sites. Majority of the successful landings where performed by KSC LC 39A.

Highest Success Rate Launch Site



KSC LC 39A has also the highest success rate with only 3 unsuccessful landings

Payload Mass vs. Success vs. Booster Version Category

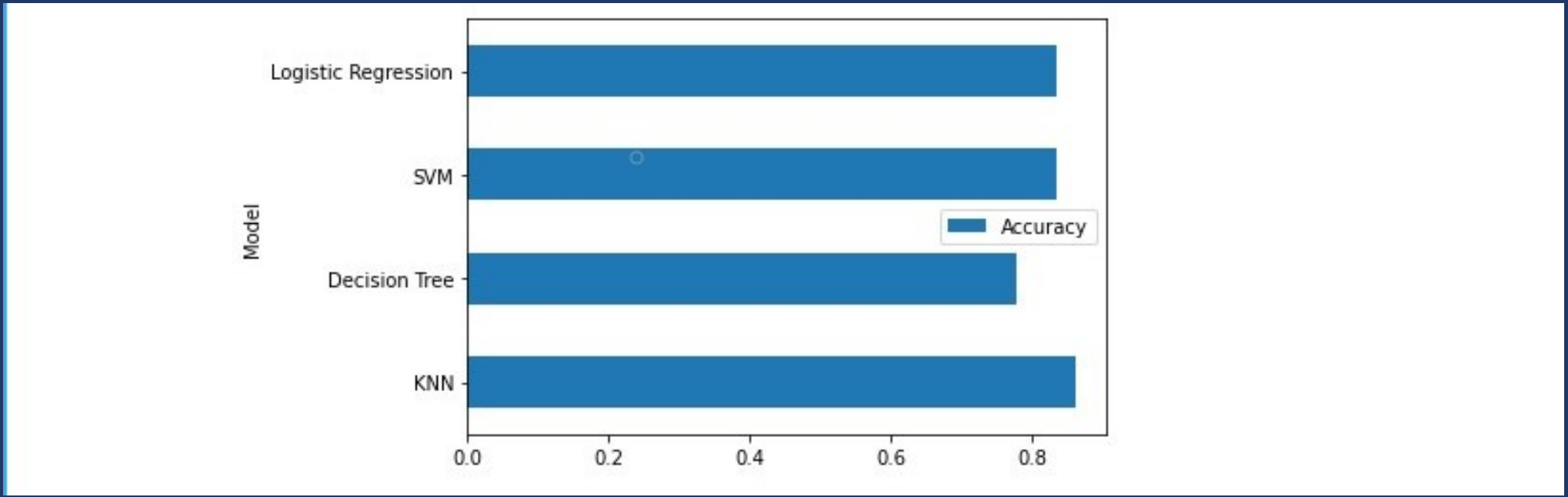
Class = 1 successful landing.
Class = 0 failed landing.

Also it seems like Boosters B4 & FT are the more successful ones

Section 6

Predictive Analysis (Classification)

Classification Accuracy

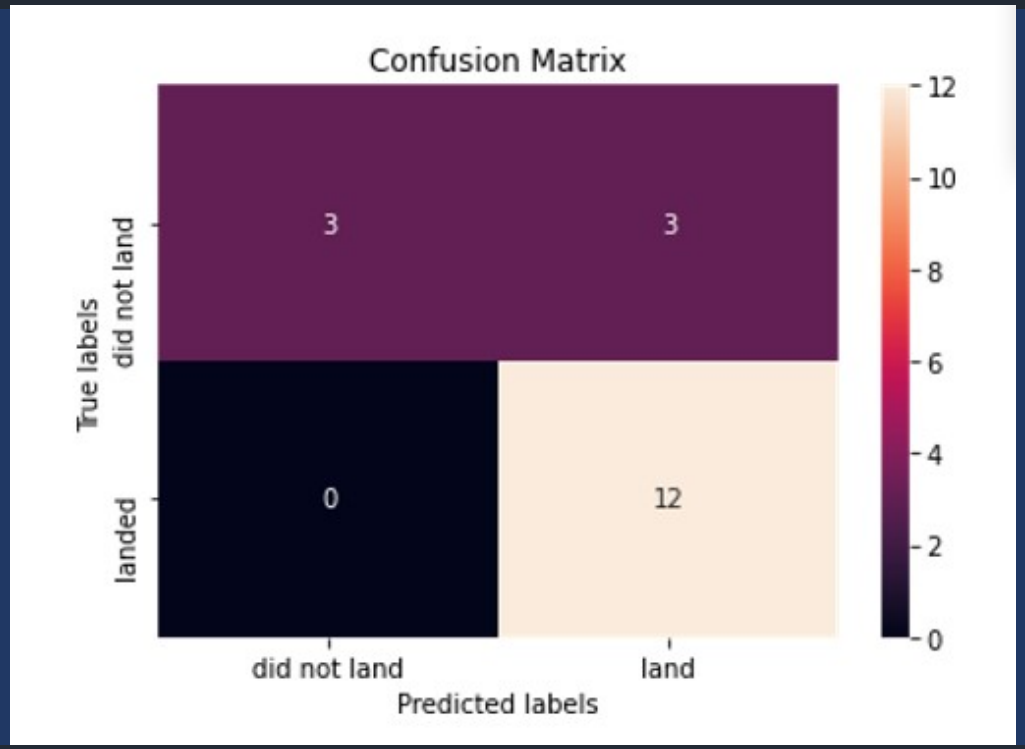


At the test set all models had decent accuracy (over 70%).

However, the most significant one was **KNN with almost 87%** accuracy.

It should be noted that test set was pretty small (18 samples) this can cause large variance in our results, such as those in Decision Tree Classifier model in repeated runs. It is possible that we need more data to determine the best model.

Confusion Matrix



Confusion Matrix was almost identical with all models.

This Matrix shows that Type I and Type II errors.

Our model has only Type II errors meaning that it predicted that it will not land but in reality it landed

Conclusions

- Our task was to develop a machine learning model for Space Y who wants to compete against SpaceX
- Our goal was to create a model to predict when Stage 1 will successfully land to save almost $100 million USD
- We gathered data from SpaceX API and with web scraping from SpaceX Wikipedia page
- We created data labels and stored data into a DB2 SQL database
- Created a dashboard for Interactive visualization
- We created a machine learning model with an accuracy of 83%
- Finaly, we are now able to use this model to predict with relatively high accuracy whether a launch will have a successful Stage 1 landing before launch to determine whether the launch should be made or not
- We recommend that more data should be collected to optimize the best machine learning model and improve accuracy

Appendix

Special Thanks to all Instructors:

Instructors: Rav Ahuja, Alex Aklson, Aije Egwaikhide, Svetlana Levitan, Romeo Kienzler, Polong Lin, Joseph Santarcangelo, Azim Hirjani, Hima Vasudevan, Saishruthi Swaminathan, Saeed Aghabozorgi, Yan Luo

GitHub URL for Full Project

https://www.coursera.org/professional-certificates/ibm-data-science?&instructors#instructors

SPACEX API URL

Thank you!