SPACEX AND THE FUTURE

DATA SCIENCE CAPTSONE

January 31, 2023

Arvin Godfrey Delos Reyes

OUTLINE

EXECUTIVE SUMMARY

Obtained, processed, and cleaned data from SpaceX API and SpaceX public data. Determined successful landing sets from the data. Made use of exploratory data analysis using SQL, and data visualization with folium maps, and dashboards. Gathered relevant columns to be used as features. Changed all categorical variables to binary using one hot encoding. Standardized data for machine learning modeling and used GridSearchCV to find best parameters for machine learning models. Lastly, visualized accuracy score of all models.

For the machine learning models four were used namely: Logistic Regression, Support Vector Machine, Decision Tree Classifier, and K Nearest Neighbors. The different models produced varying results with an average of 83.33% accuracy across all the models

INTRODUCTION

Project Background:

- Space Y is competing with Space X
- Space X provides readily accessible data
- They have been able to successfully record and recover Launched Rockets
- They are leading the race towards sustainable rocket launching and reusability.

Problem:

- **Space Y is aiming to find the success rate of recovering a stage 1 rocket through the use of a machine learning model with the appropriate accuracy.**

METHODOLGY

1. Data collection methodology:
    a. Combined data from SpaceX public API and SpaceX Wikipedia page
2. Perform data wrangling
    a. Classifying true landings as successful and unsuccessful otherwise
3. Perform exploratory data analysis using SQL
4. Create interactive visual analytics using Folium and Plotly Dash
5. Perform predictive analysis using classification models
    a. Tuned models using GridSearchCV

METHODOLOGY

OVERVIEW OF DATA COLLECTION, WRANGLING, VISUALIZATION, AND MODELING METHODS

DATA COLLECTION OVERVIEW

The data collection process used involves the API requests from Space X API and web scraping data from Space X's Wikipedia entry.

Space X API Data Columns:

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

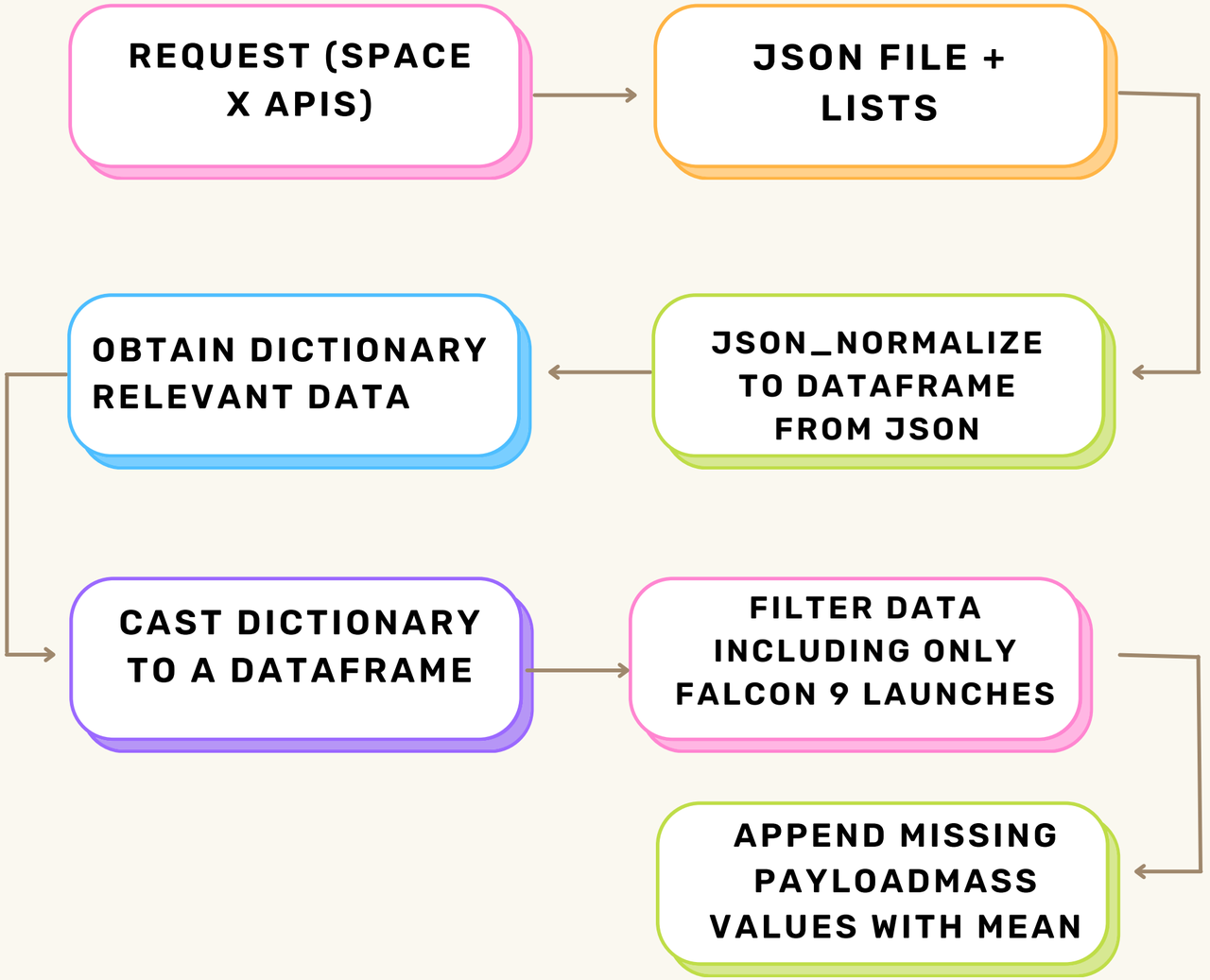
Wikipedia Webscraped Data Columns:

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

DATA COLLECTION - SPACE X API

GitHub URL:
https://github.com/99AGFDR/DataSci_IBM_Google_ProjectPortfolio/blob/main/Course10_FinalOutput/C10_Week1_01spacex_data_collection_api.ipynb

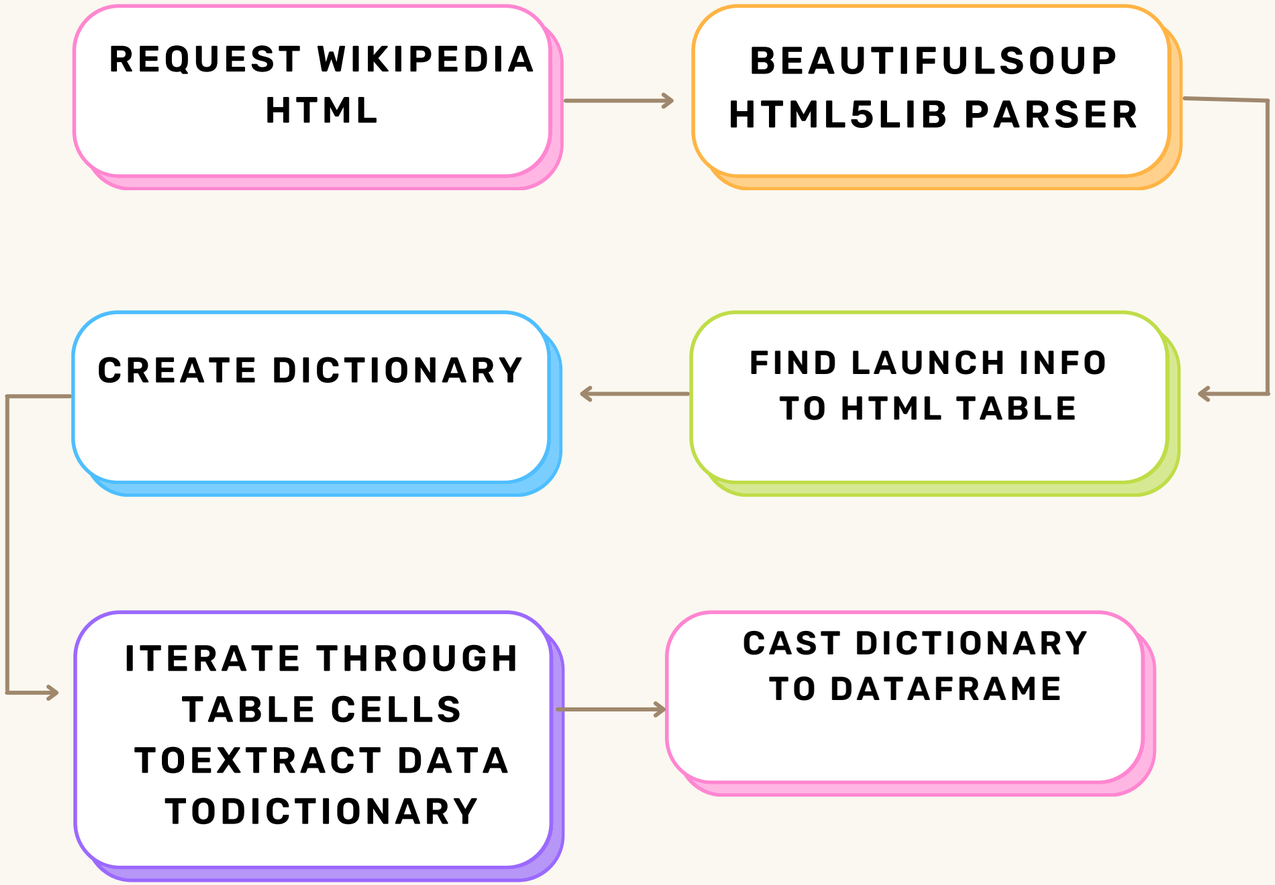
SPACE X API FLOWCHART



DATA COLLECTION - WEB SCRAPING

GitHub URL:
https://github.com/99AGFDR/DataSci_IBM_Google_ProjectPortfolio/blob/main/Course10_FinalOutput/C10_Week1_02spacex_WebScraping.ipynb

WEB SCRAPING FLOWCHART



DATA WRANGLING

- Develop training labels where a successful landing = 1 and failure = 0
- The result has two components on outcomes and landings
- Develop a training label for a class that encodes a mission outcome that is true to 1 and 0 when false.

Value Mapping:

The values are set to 1 when they are of the following: True ASDS, True RTLS, & True Ocean

The values are set to 0 when they are of the following: None None, False ASDS, None ASDS, False Ocean, False RTLS

GitHub URL:

https://github.com/99AGFDR/DataSci_IBM_Google_ProjectPortfolio/blob/main/Course10_FinalOutput/C10_Week1_03spacex-Data%20wrangling.ipynb

EDA WITH DATA VISUALIZATION

- Exploratory Data Analysis performed on variables Flight Number, Payload Mass, Launch Site, Orbit, Class, and Year.

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
- Success Yearly Trend
- Those graphs were used to determine an existing relationship between variables in order to use the data in applicable machine-learning models

GitHub URL:
https://github.com/99AGFDR/DataSci_IBM_Google_ProjectPortfolio/blob/main/Course10_FinalOutput/C10_Week2_02eda-dataviz.ipynb

EDA WITH SQL

- Prepared and uploaded the different datasets into the IBM DB2 Database.
- Created queries using SQL Python integration through magic SQL functions.
- Queries were made to get a better understanding of the dataset.
- Queried the necessary information regarding launch site names, mission outcomes, various payload sizes of customers, booster versions, and landing outcomes

Github URL:
https://github.com/99AGFDR/DataSci_IBM_Google_ProjectPortfolio/blob/main/Course10_FinalOutput/C10_Week2_01eda-sql-sqllite.ipynb

BUILD AN INTERACTIVE MAP WITH FOLIUM

- Created Folium maps that marked launch sites, successful and unsuccessful landings, and proximity to locations such as Railway, Highway, Coast, and City.

- This allows us to understand where launch sites are located. Through visualization, a better understanding of the different places and conditions necessary for a rocket launch is presented

Github URL:
https://github.com/99AGFDR/DataSci_IBM_Google_ProjectPortfolio/blob/main/Course10_FinalOutput/C10_Week3_01LaunchSite.ipynb

BUILD A DASHBOARD WITH PLOTLY DASH

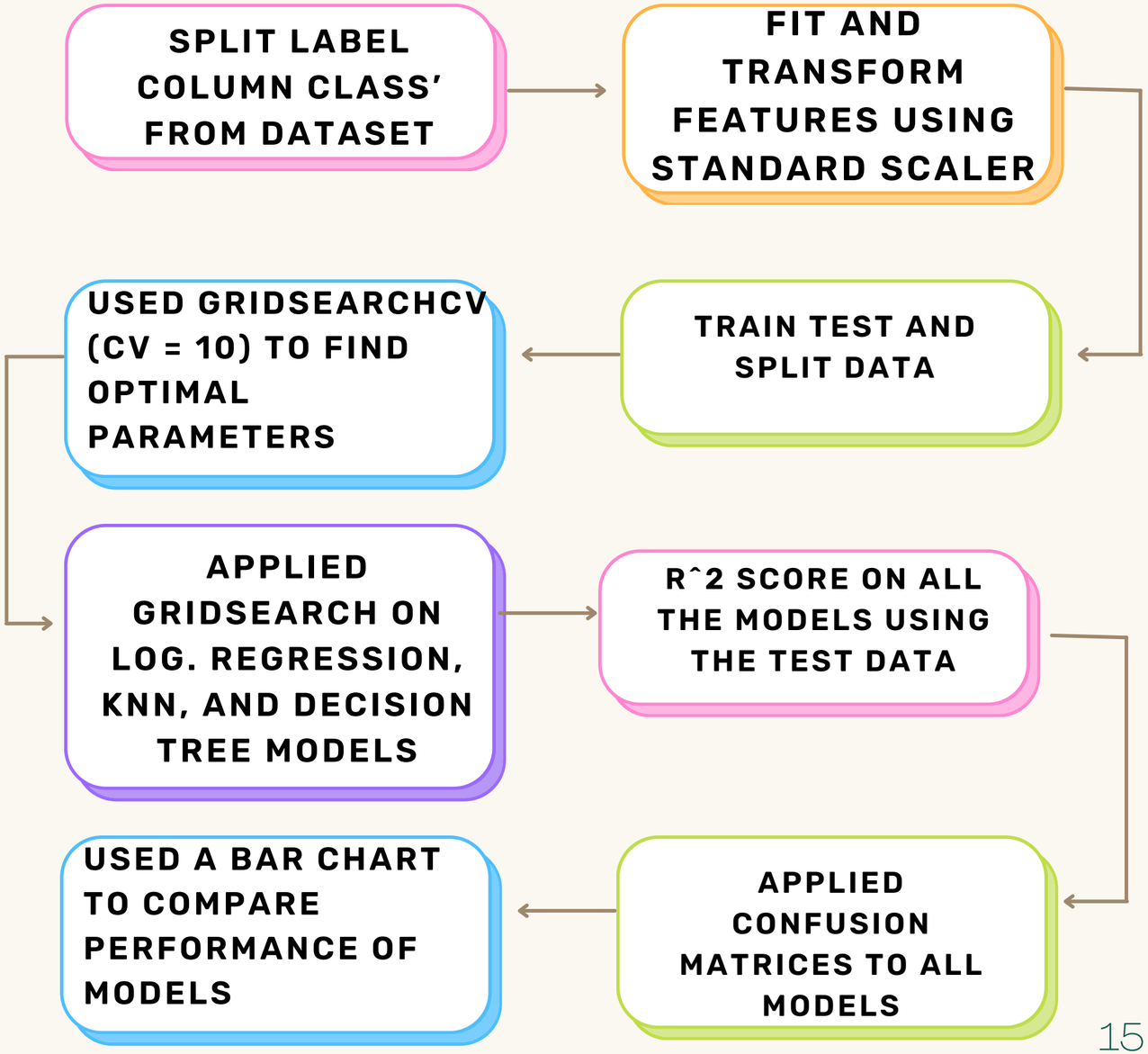
- Dashboard includes a pie chart and a scatter plot.
- Pie chart can be selected to show the distribution of successful landings across all launch sites and can be selected to show individual launch site success rates.
- Scatter plot takes two inputs: All sites or individual sites and payload mass on a slider between 0 and 10000 kg.
- The pie chart is used to visualize the 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:
https://github.com/99AGFDR/DataSci_IBM_Google_ProjectPortfolio/blob/main/Course10_FinalOutput/C10_Week3_02Dash_App_SpaceX.py

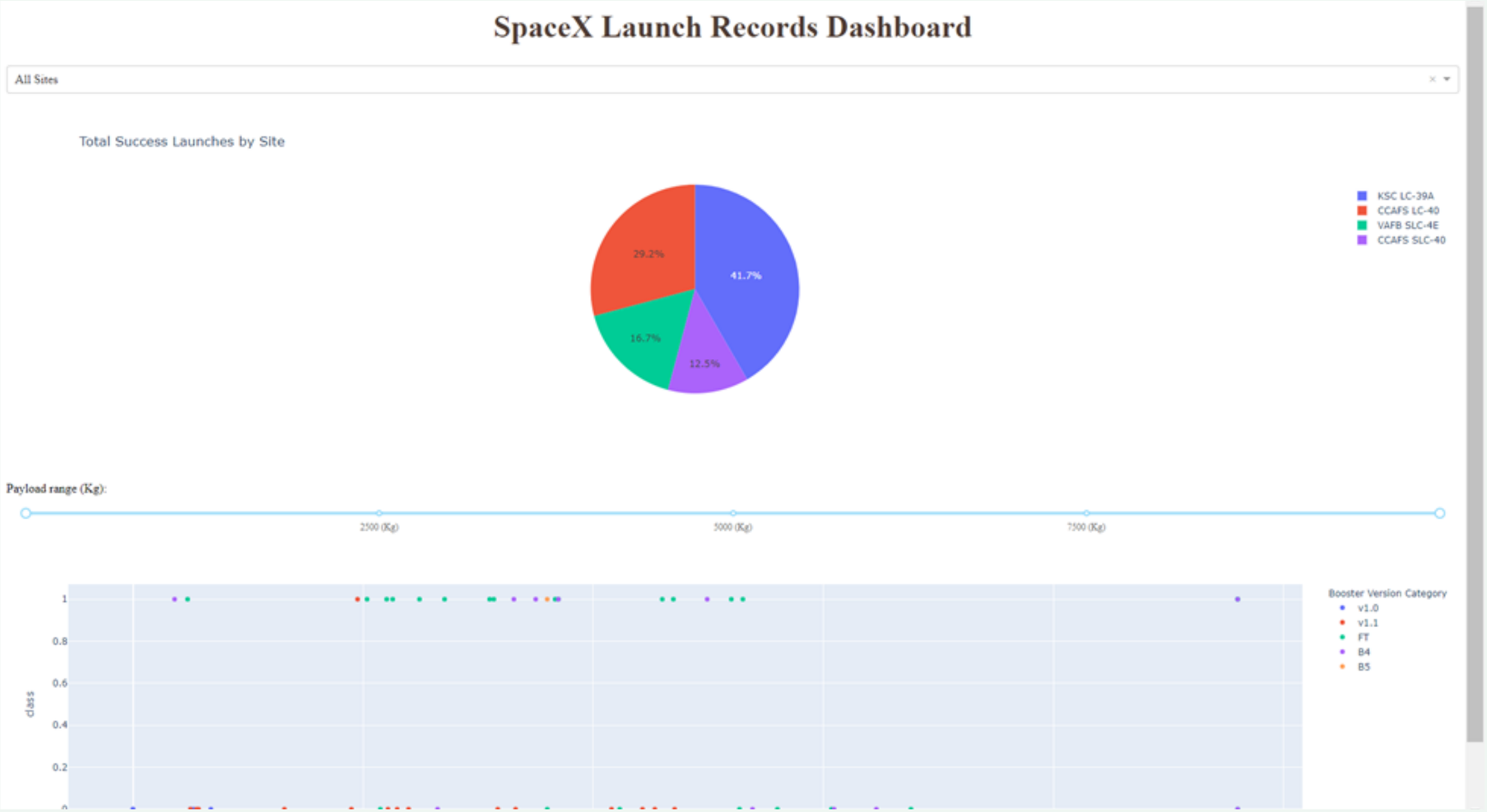
PREDICTIVE ANALYSIS (CLASSIFICATION)

GitHub URL:
https://github.com/99AGFDR/DataSci_IBM_Google_ProjectPortfolio/blob/main/Course10_FinalOutput/C10_Week5_02ML_Application.ipynb

PREDICTIVE ANALYSIS (CLASSIFICATION) FLOWCHART



RESULT

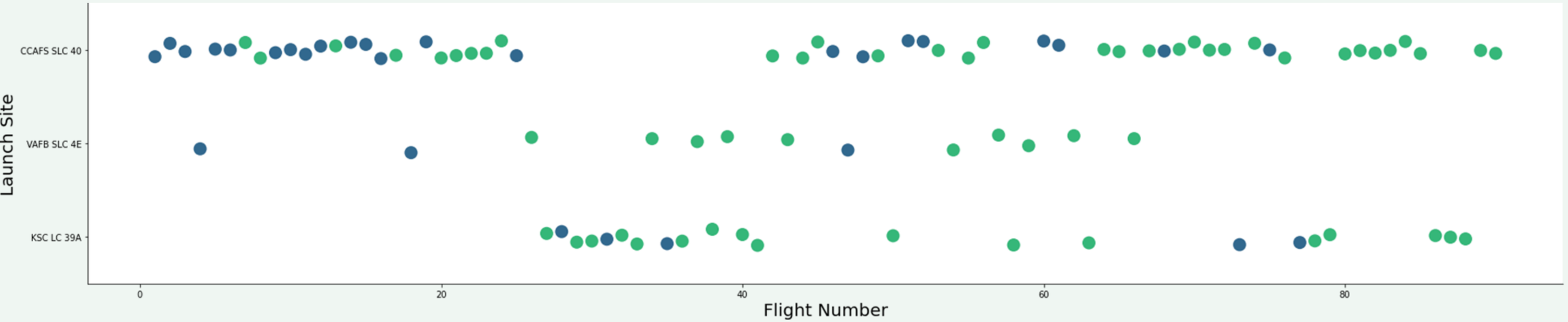


Preview of the dashboard. The following slides will be about the different results obtained from the exploratory data analysis using python and SQL. With the model results having a 83% accuracy rating so far.

EDA WITH VISUALIZATION

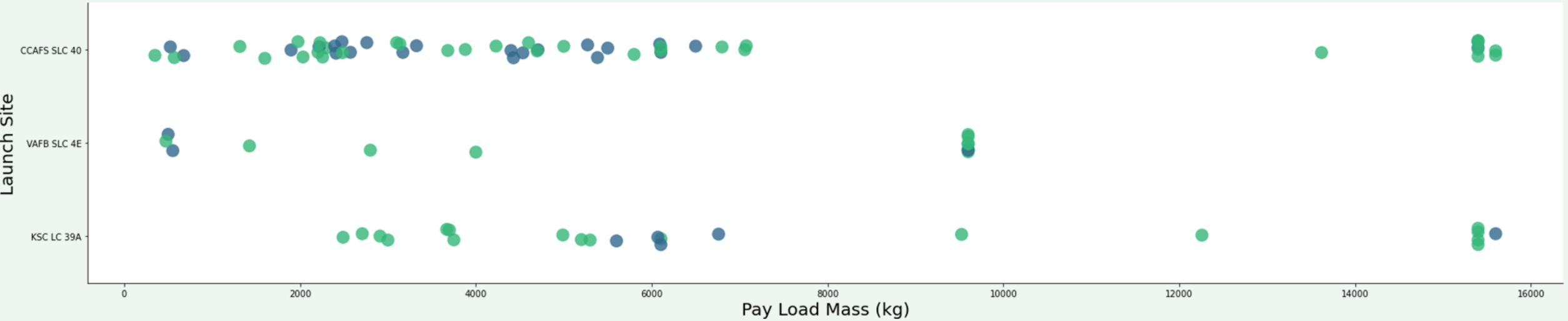
EXPLORATORY DATA ANALYSIS WITH PYTHON

FLIGHT NUMBER VS. LAUNCH SITE



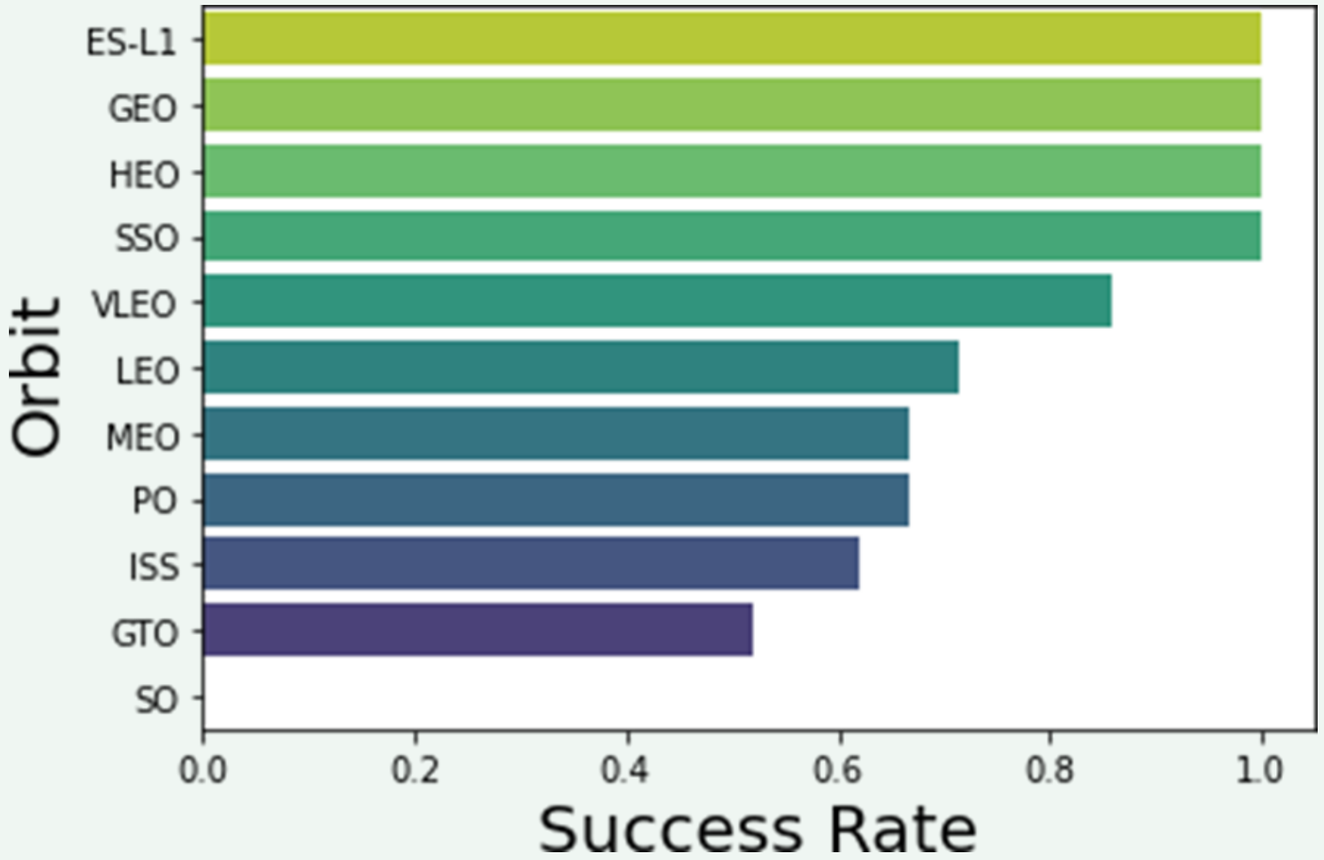
- Green indicates successful launch; Purple indicates unsuccessful launch.
- Briefly the graph shows that as more flights are undertaken the more they are successful flights rather than unsuccessful flights

PAYLOAD VS. LAUNCH SITE



- Green indicates successful launch; Purple indicates unsuccessful launch.
- Briefly the graph shows that there are no clear patterns towards payload mass and launch site more data is needed to confirm emerging trends and patterns such as at the highest mass are mostly successes and in the ranges between 0 to 6000 data is skewed slightly towards successes.

SUCCESS RATE VS. ORBIT TYPE

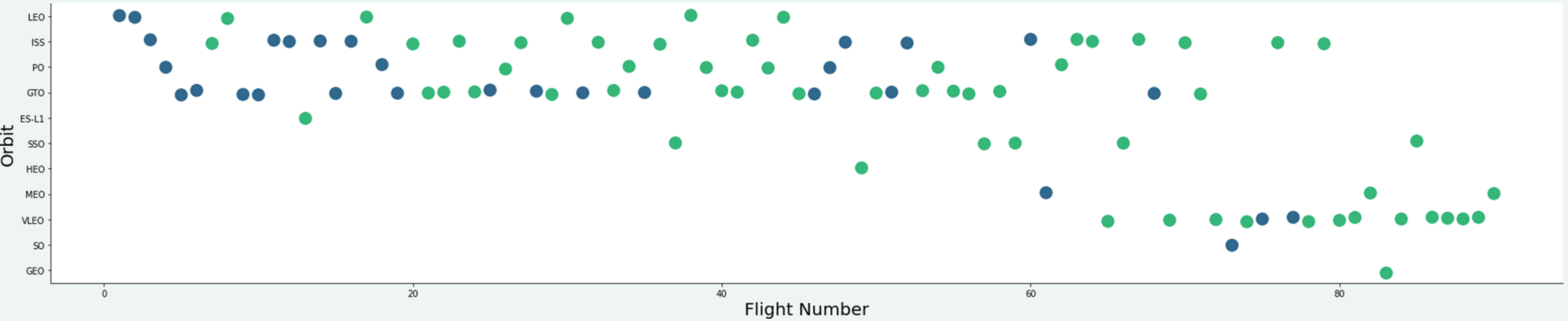


Description of Probability Markers:

- 0 as 0%
- 0.5 as 50%
- 1 as 100%

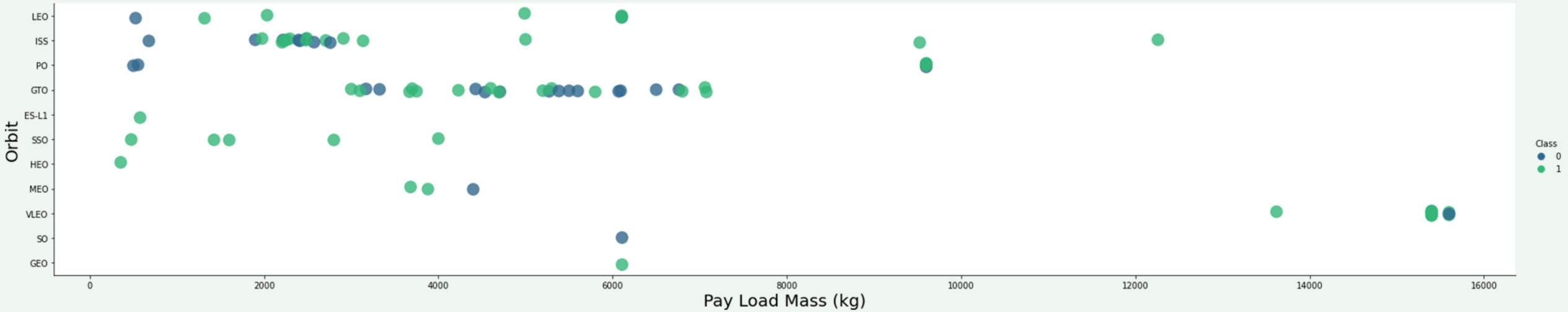
- ES-L1 (1), GEO (1), HEO (1), and SSO (5) has 100% success rate. (sample sizes in parenthesis)
- VLEO (14) has decent success rate and attempts greater than 85%
- SO (1) has 0% success rate
- GTO (27) has around a 50% success rate and is the largest sample size

FLIGHT NUMBER VS. ORBIT TYPE



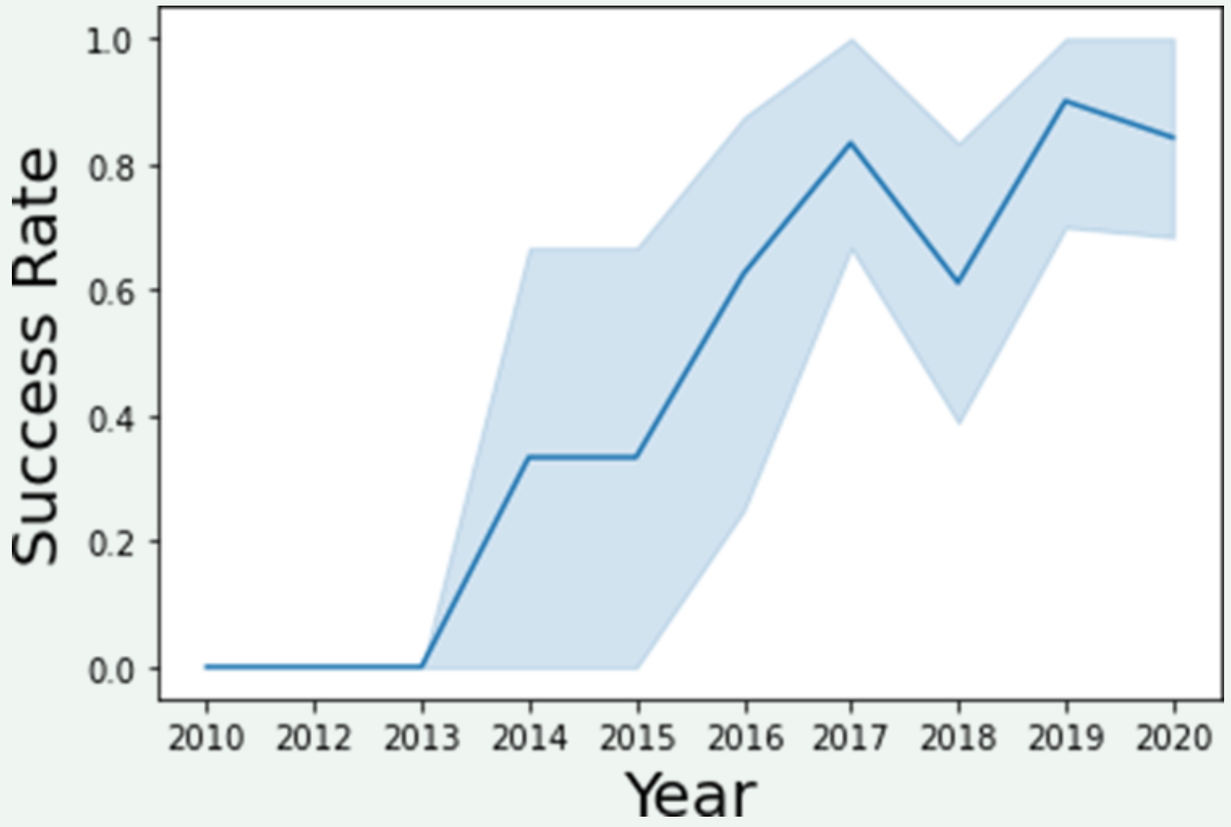
- Green indicates successful launch; Purple indicates unsuccessful launch.
- Launch Orbit preferences changed over Flight Number.
- Launch Outcome seems to correlate with this preference.
- SpaceX started with LEO orbits which saw moderate success LEO and returned to VLEO in recent launches SpaceX appears to perform better in lower orbits or Sun-synchronous orbits

PAYLOAD VS. ORBIT TYPE



- Green indicates successful launch; Purple indicates unsuccessful launch.
- Payload mass correlates with orbit.
- LEO and SSO have relatively low payload mass.
- The other most successful orbit VLEO only has payload mass values in the higher end of the range
- More data is needed to determine relationships to evenly split data markers

LAUNCH SUCCESS YEARLY TREND



- There has been a marked increase in successful launches that spiked after the year 2013 and gradually increased ever since with a slight dip in 2018
- Success in recent years are around equal to or greater than 80%

EDA WITH SQL

EXPLORATORY DATA ANALYSIS WITH IBM DB2 DATABASE

IGENERATING QUERIES IN PYTHON WITH SQLALCHEMY

ALL LAUNCH SITE NAMES

Display the names of the unique launch sites in the space mission

In [7]:
```
%%sql

SELECT DISTINCT("LAUNCH_SITE")
FROM "SPACEXTBL"
```

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

Out[7]:

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

- A query of unique launch site names from the database was made.
- CCAFS SLC-40 and CCAFSSLC-40 represent the same launch site with a possible data entry error.
- Only 3 unique launch site values: CCAFS SLC-40, KSC LC-39A, and VAFB SLC-4E

LAUNCH SITE NAMES BEGINNING WITH `CCA`

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

In [11]:
```
%%sql

SELECT "LAUNCH_SITE"
FROM "SPACEXTBL"
WHERE "LAUNCH_SITE" like '%CCA%'
ORDER BY "LAUNCH_SITE"
LIMIT 5
```

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

Out[11]:

Launch_Site
CCAFS LC-40
CCAFS LC-40
CCAFS LC-40
CCAFS LC-40
CCAFS LC-40

- The first 5 records of launch sites with names beginning with CCA.

TOTAL PAYLOAD MASS FROM NASA

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

In [24]:
```
%%sql

SELECT "CUSTOMER", SUM("PAYLOAD_MASS__KG_")
FROM "SPACEXTBL"
WHERE "CUSTOMER" = "NASA (CRS)"
```

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

Out[24]:

Customer	SUM("PAYLOAD_MASS__KG_")
NASA (CRS)	45596

- The SQL query sums the total payload mass in kg as NASA was the customer.
- CRS stands for Commercial Resupply Services that seems to indicate that the payloads were probably sent to the International Space Station (ISS).

AVERAGE PAYLOAD MASS BY F9 V1.1

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

In [50]:
```
%%sql

SELECT "BOOSTER_VERSION", AVG("PAYLOAD_MASS__KG_")
FROM "SPACEXTBL"
WHERE "BOOSTER_VERSION" = "F9 v1.1";
```

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

Out[50]:

Booster_Version	AVG("PAYLOAD_MASS__KG_")
F9 v1.1	2928.4

- The SQL query calculates the average of the payload mass of the F9 V1.1
- The result shows an average of 2928.4

FIRST SUCCESSFUL GROUND PAD LANDING DATE

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

Hint:Use min function

In [51]:
```
%%sql

SELECT min("DATE"), "Landing _Outcome"
FROM "SPACEXTBL"
WHERE "Landing _Outcome" = "Success (ground pad)";
```

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

Out[51]:

min("DATE")	Landing _Outcome
01-05-2017	Success (ground pad)

- The SQL query shows the first date where a successful ground pad landing occured.
- The result shows the date to be around the year 2017.

SUCCESSFUL DRONE SHIP LANDING WITH PAYLOAD BETWEEN 4000 AND 6000

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

Hint:Use min function

In [51]:
```
%%sql

SELECT min("DATE"), "Landing _Outcome"
FROM "SPACEXTBL"
WHERE "Landing _Outcome" = "Success (ground pad)";
```

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

Out[51]:

min("DATE")	Landing _Outcome
01-05-2017	Success (ground pad)

- The SQL query shows the first date where a successful ground pad landing occured.
- The result shows the date to be around the year 2017.

TOTAL NUMBER OF EACH MISSION OUTCOME

In [56]:

```
%%sql

SELECT "MISSION_OUTCOME", COUNT("MISSION_OUTCOME") AS Num_Outcome
FROM "SPACEXTBL"
GROUP BY "MISSION_OUTCOME";
```

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

Out[56]:

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

- The SQL query shows the different mission outcomes and their number
- There are 99 successes and 1 failure indicating that there is a 99% success rate with all launches so far.
- The multiple success category indicates a data entry error and the unclear status is peculiar.

BOOSTERS THAT CARRIED MAXIMUM PAYLOAD

In [59]:

```
%%sql

SELECT "BOOSTER_VERSION", "PAYLOAD_MASS__KG_"
FROM "SPACEXTBL"
WHERE "PAYLOAD_MASS__KG_" =
    (SELECT MAX("PAYLOAD_MASS__KG_")
    FROM "SPACEXTBL");
```

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

Out[59]:

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

- The SQL query shows the different maximum payload carrying capacity of each booster version.
- These booster versions are similar and are of the F9 B5 B10xx.x variation.
- This indicates that payload mass correlates with the booster version used.

2015 FAILED DRONE SHIP LANDING RECORDS

In [78]:

```
%%sql

SELECT "BOOSTER_VERSION", "LAUNCH_SITE", substr(Date, 4, 2), substr(Date,7,4)
FROM "SPACEXTBL"
WHERE ("Landing _OUTCOME" = 'Failure (drone ship)') and (substr(Date,7,4) = '2015')
```

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

Out[78]:

Booster_Version	Launch_Site	substr(Date, 4, 2)	substr(Date,7,4)
F9 v1.1 B1012	CCAFS LC-40	01	2015
F9 v1.1 B1015	CCAFS LC-40	04	2015

- The SQL query shows the months in 2015 where there was a failure in the landing on a drone ship.
- It is clear that there are two instances in January and April where the failure occurred and are of the F9 variation.

RANK LANDING OUTCOMES BETWEEN 2010-06-04 AND 2017-03-20

In [18]:

```
%%sql

SELECT "Landing _OUTCOME", COUNT(*) as NO_OUTCOME
FROM "SPACEXTBL"
WHERE ("Landing _OUTCOME" like "%Success%") and ("DATE" between "04-06-2010"and "20-03-2017")
GROUP BY "Landing _OUTCOME"
ORDER BY "NO_OUTCOME" desc
```

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

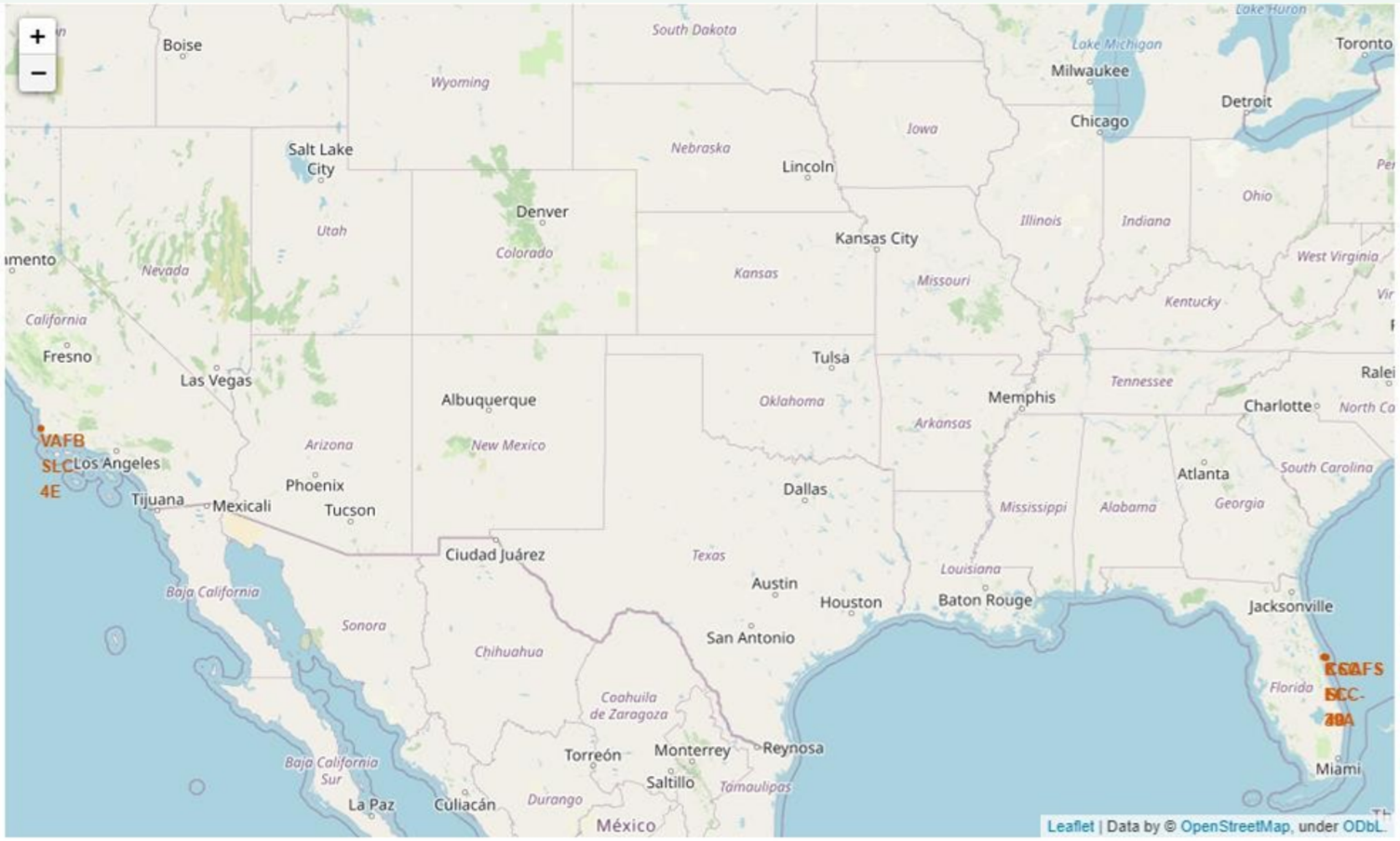
Out[18]:

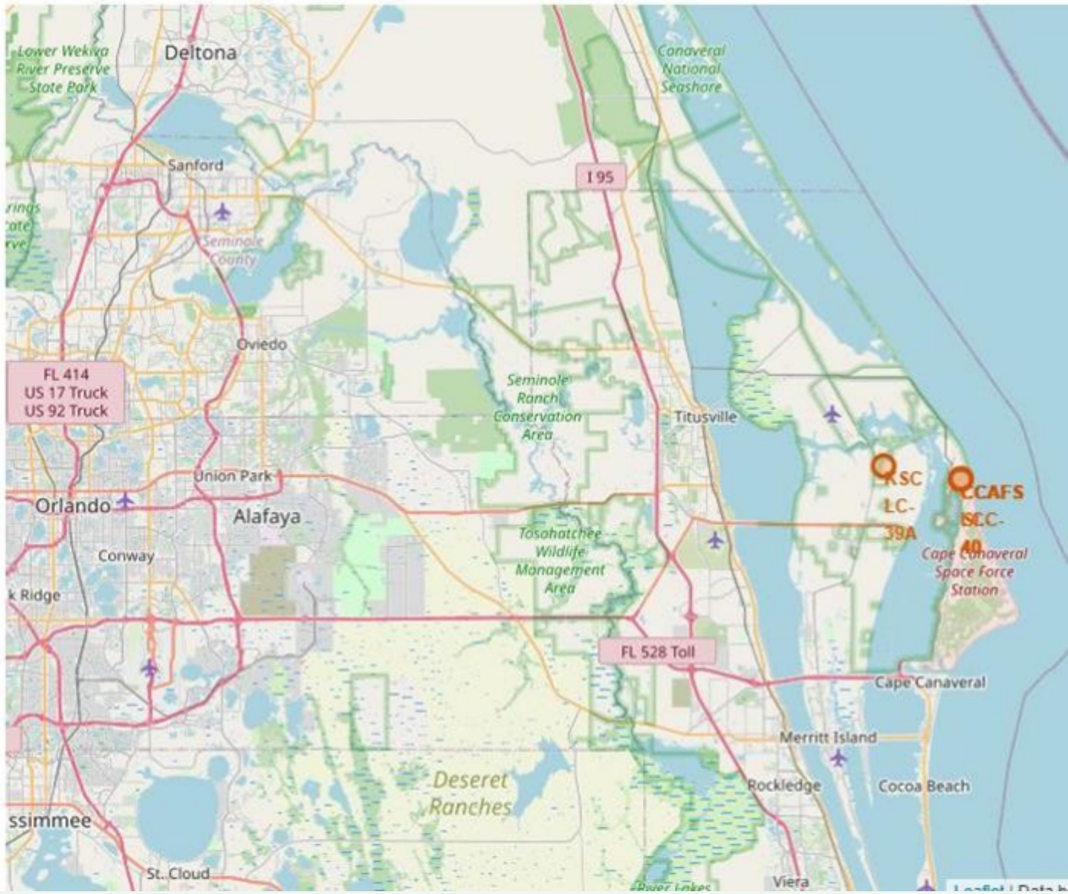
Landing _Outcome	NO_OUTCOME
Success	20
Success (drone ship)	8
Success (ground pad)	6

- The SQL query shows the count of the different successful landing outcomes.
- There are three types of successful landing outcomes: success, drone ship, and ground pad landings.
- There is a total of 34 landings that were a success.

INTERACTIVE MAP WITH FOLIUM

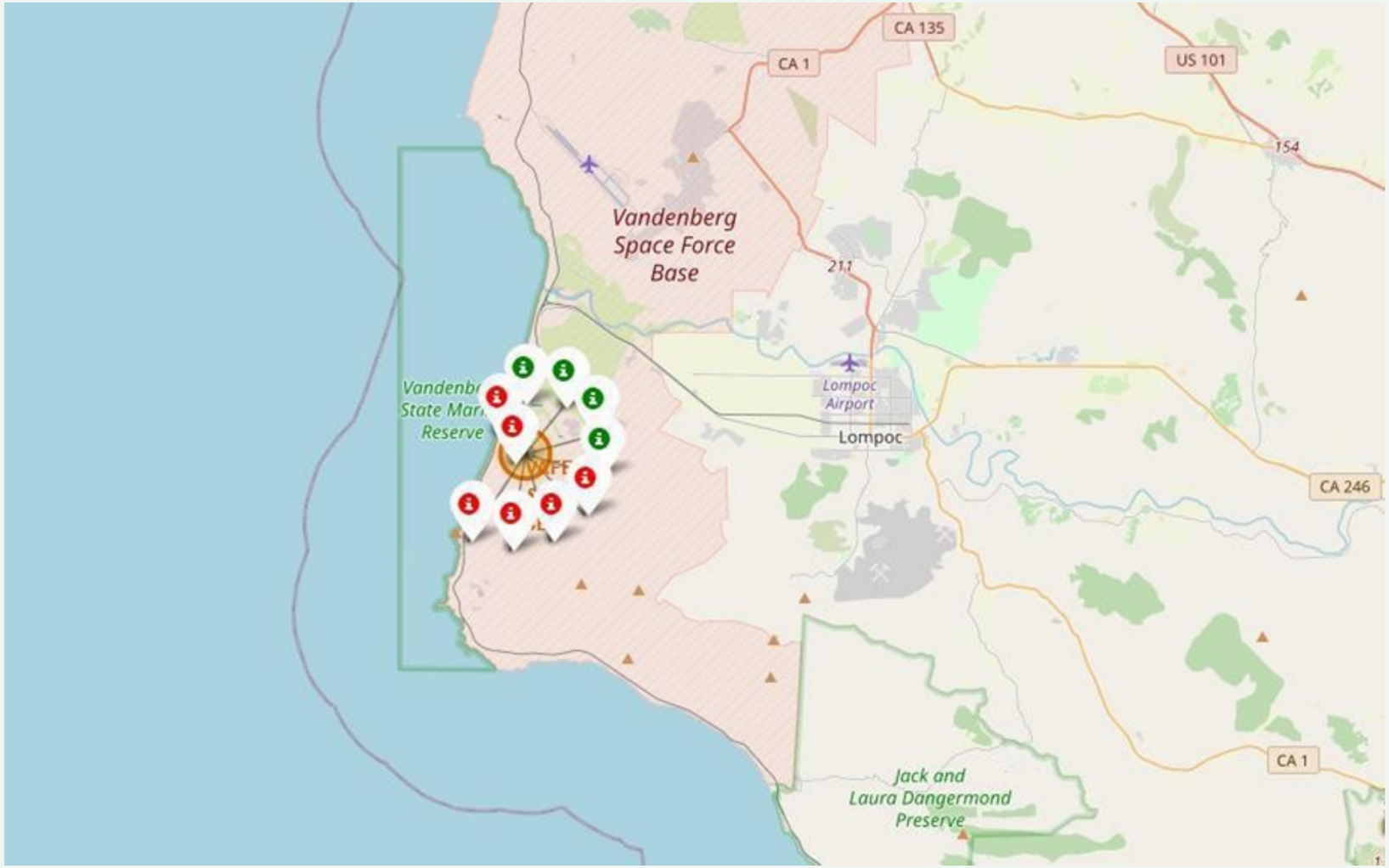
LAUNCH SITE LOCATIONS





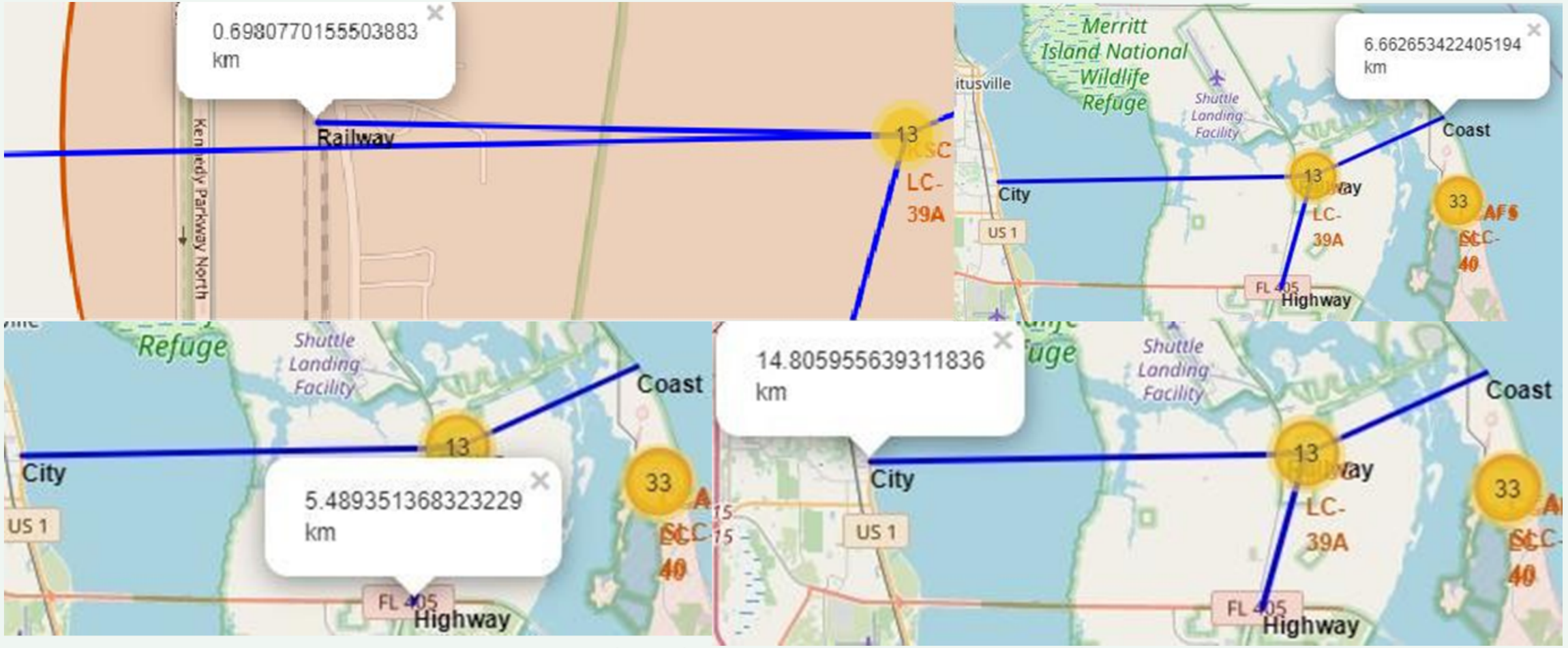
- The left map shows all launch sites on the US map. On the right map it shows the two Florida launch sites. All launch sites are at coastal areas.

COLOR-CODED LAUNCHMARKERS



- The map indicates the different failed and successful landings the area near Vandenberg space force base. The example indicates VAFB SLC-4E that shows 4 successful landings and 6 failed landings

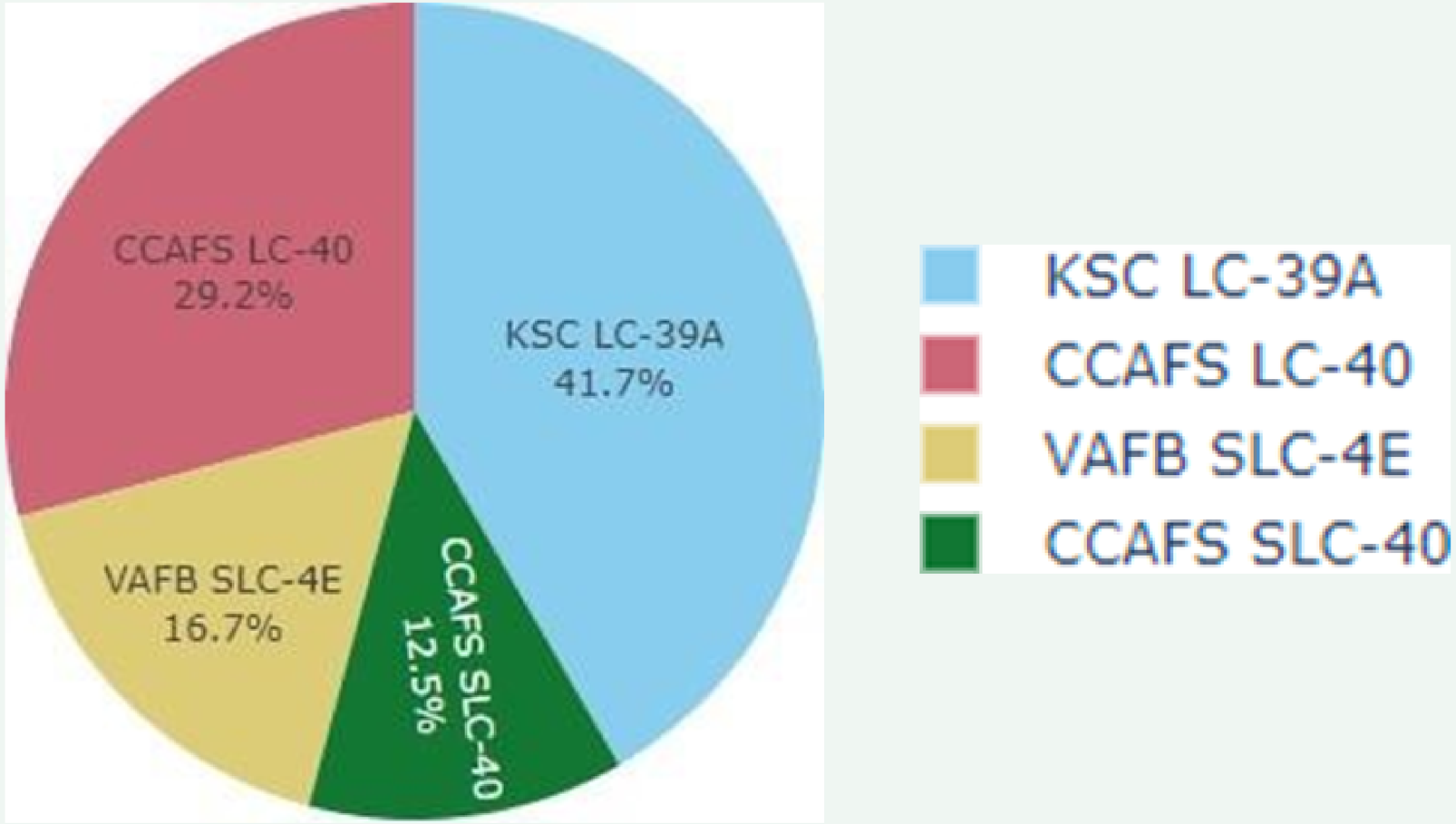
KEY LOCATION PROXIMITIES



Using KSC LC-39A as an example, launch sites are close to railways for supply transportation. Launch sites are close to highways, coasts and relatively far from cities so that launch failures can land in the sea to avoid rockets falling on densely populated areas.

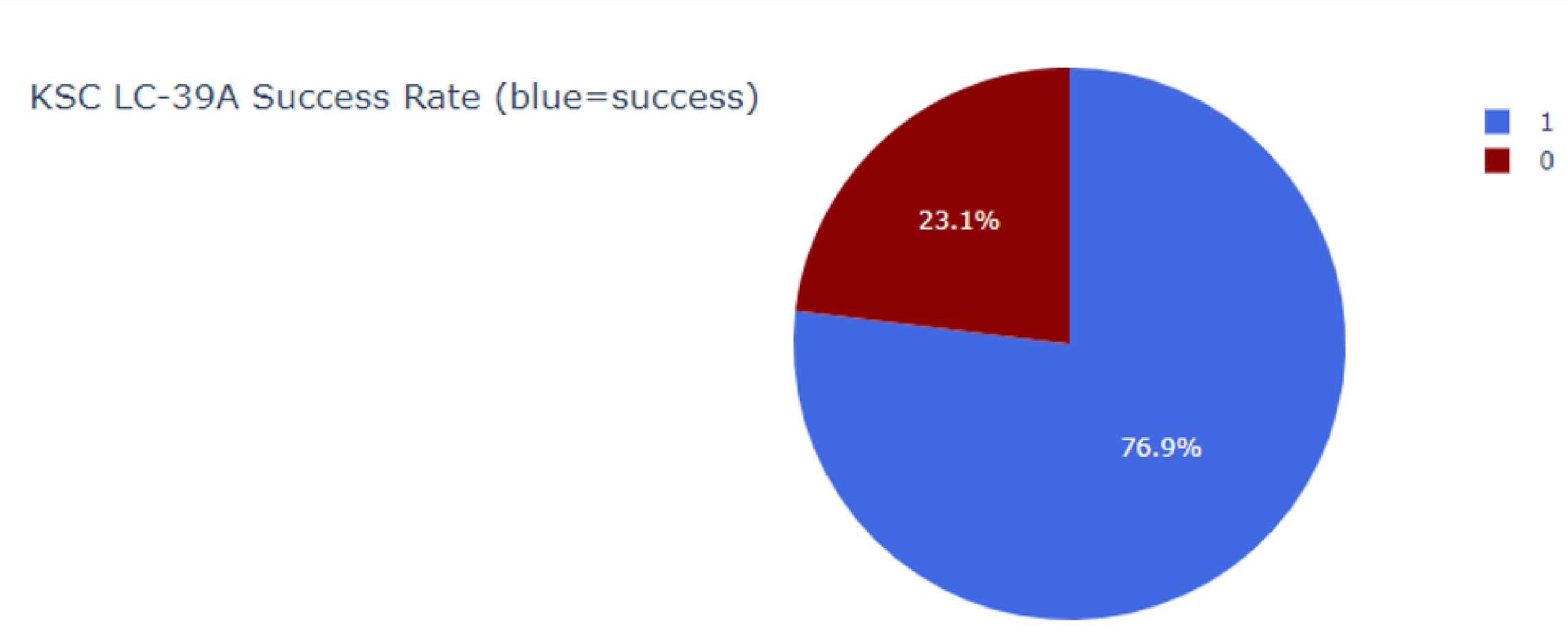
BUILD A DASHBOARD WITH PLOTLYDASH

SUCCESSFUL LAUNCHES ACROSS LAUNCHSITES



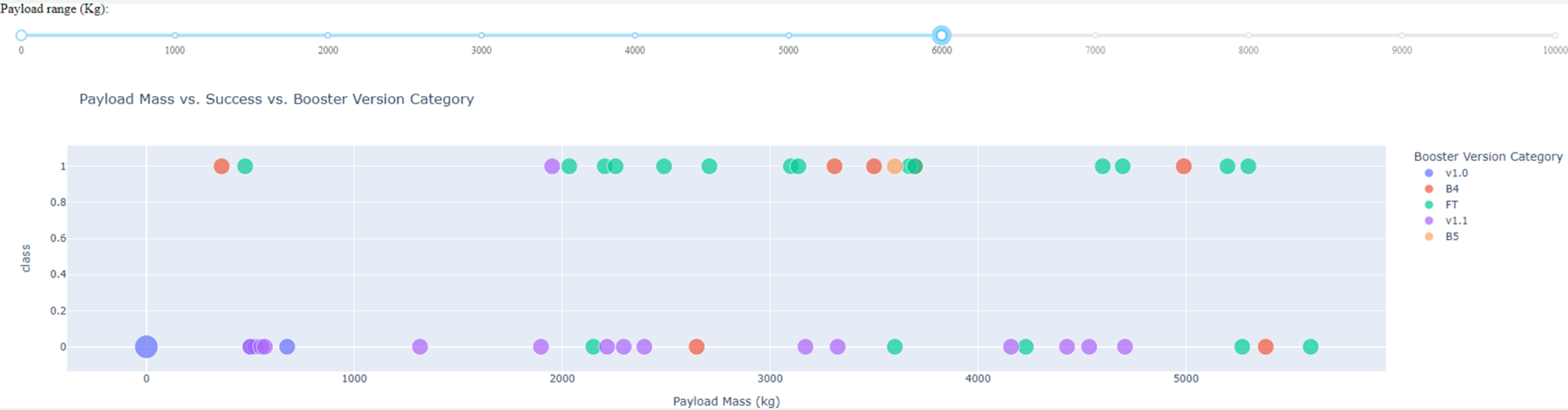
This is the distribution of successful landings across all launch sites. CCAFS LC-40 is the old name of CCAFS SLC-40 so CCAFS and KSC have the same amount of successful landings, but a majority of the successful landings where performed before the name change. VAFB has the smallest share of successful landings. This may be due to smaller sample and increase in difficulty of launching in the west coast.

HIGHEST SUCCESS RATE LAUNCHSITE



Among the rockets launched, KSC LC-39A has the highest success rate of 76.9% and 23.1% for failed landing rate.

PAYLOAD MASS VS. SUCCESS VS. BOOSTER VERSION CATEGORY

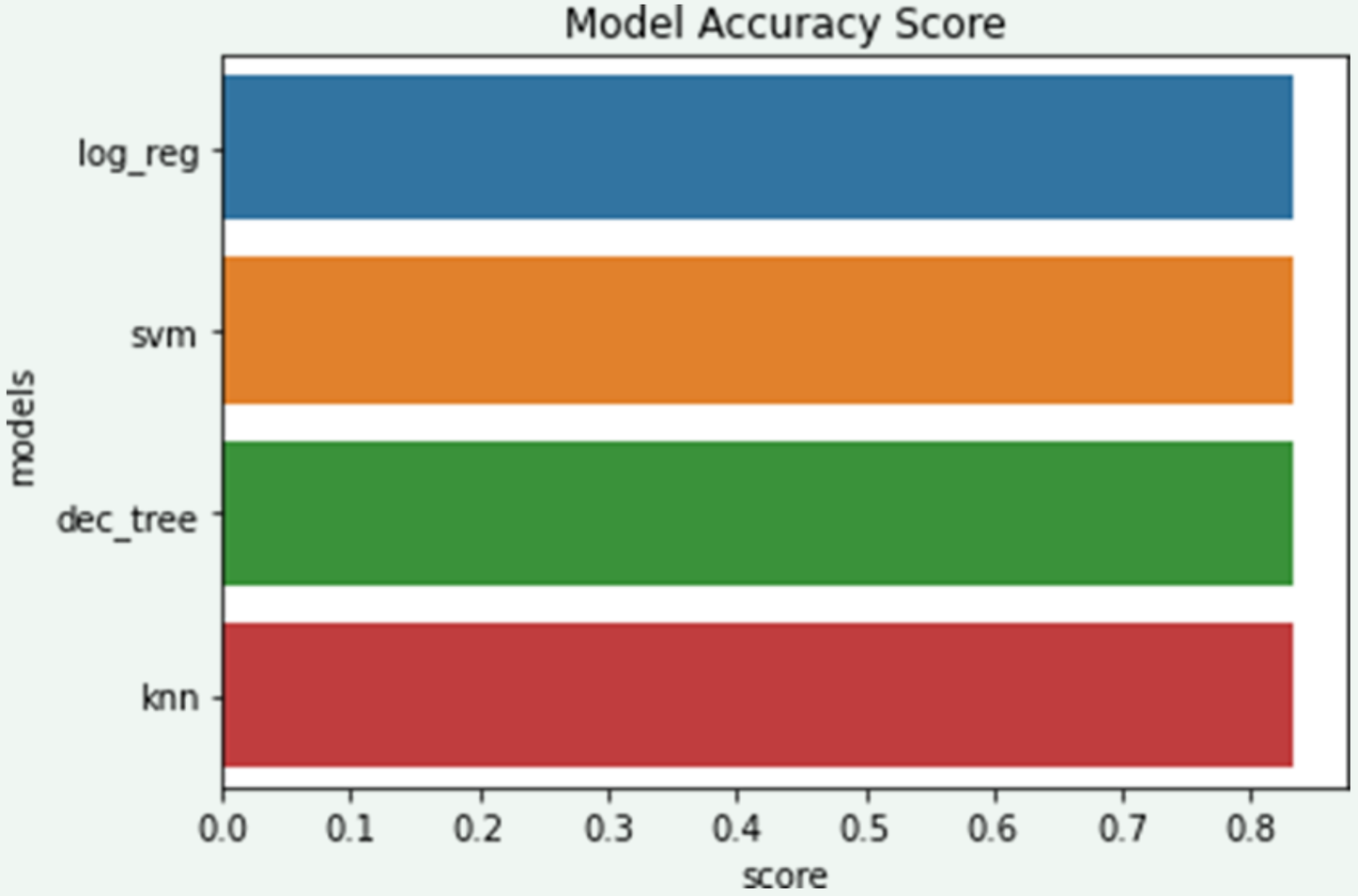


The dashboard shows a Payload range selector. The selector was set from 0 to 10,000 Kg. Class indicates 1 for a successful landing and 0 for a failed landing. The scatter plot also accounts for booster version category in color and number of launches in point size. In this particular range of 0-6000, interestingly there are two failed landings with payloads of zero kg.

PREDICTIVE ANALYSIS (CLASSIFICATION)

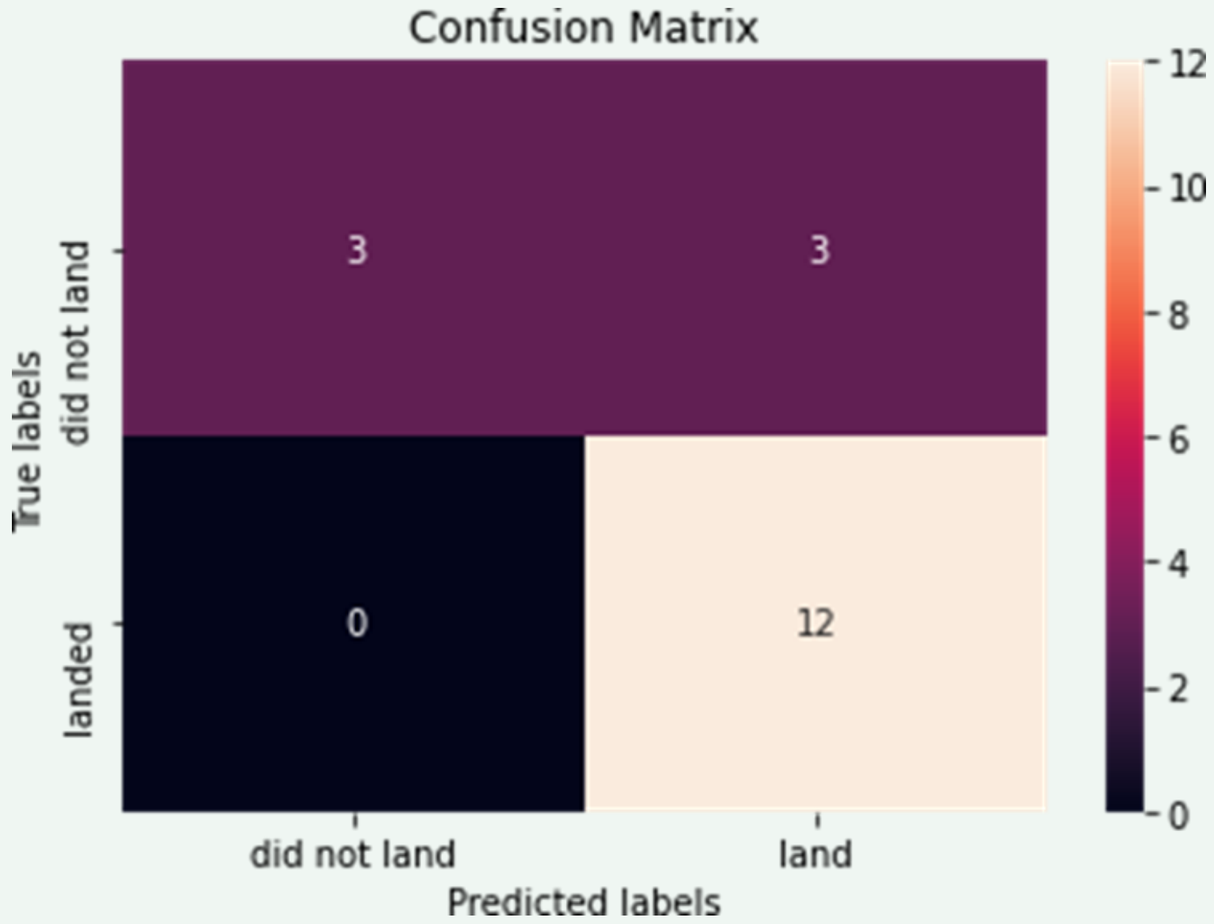
GRIDSEARCHCV(CV=10) ON LOGISTIC REGRESSION, SVM, DECISION TREE, AND KNN

CLASSIFICATION ACCURACY



The models had the same accuracy score of 83.33% using the test set data. There is a need for more data to further establish and differentiate the best results in order to find the optimal model.

CONFUSION MATRIX



Correct predictions are on a diagonal from top left to bottom right.

All the models performed accordingly for the test set data. The models predicted indicated 12 successful landings (true positive) . The models predicted 3 unsuccessful landings (true negative).
The models predicted 3 successful landings when the label was unsuccessful landings (false positives).

CONCLUSION

- A machine learning model was developed for Space Y.
- The goal of the model is to predict when Stage 1 will successfully land to save ~$100 million USD
- The data from SpaceX was used in particular their API and Webscraped the Wikipedia data available
- Developed data labels and used the IBM DB2 database to store the datasets.
- Developed different machine learning models with an accuracy of 83%
- Allon Mask of SpaceY can 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.
- If possible more data should be collected to better determine the best machine learning model and improve accuracy.

APPENDIX

GitHub repository URL:
https://github.com/99AGFDR/DataSci_IBM_Google_ProjectPortfolio/tree/main/Course10_FinalOutput

Special Thanks to all the instructors and professionals who made it possible to accomplish the IBM Data Science Professional Certificate.

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

THANK YOU

FOR LISTENING

January 31, 2023
Arvin Godfrey Delos Reyes