Space Venture with Data Science

Part 1

Github: Poornima Github

26/10/2025

Skills Network

IBM

OUTLINE

- Executive Summary
- Introduction
- Methodology
- Results
  - Visualization – Charts
  - Dashboard
- Discussion
  - Findings & Implications
- Conclusion
- Appendix

EXECUTIVE SUMMARY

- The study aims to predict Space X first stage successful landing using Space X data.
- Data Collection using Space X API
- Space X data collection by Web Scrapping
- Data Wrangling
- Exploratory data analysis (EDA)
  - EDA with SQL
  - EDA using Visualization Lab
- Interactive Visual Analytics using folium
- Interactive Dashboard with Plotly Dash
- Space X predictive Analysis using machine Learning
- Upload files to Git hub and present the findings in a presentation

INTRODUCTION

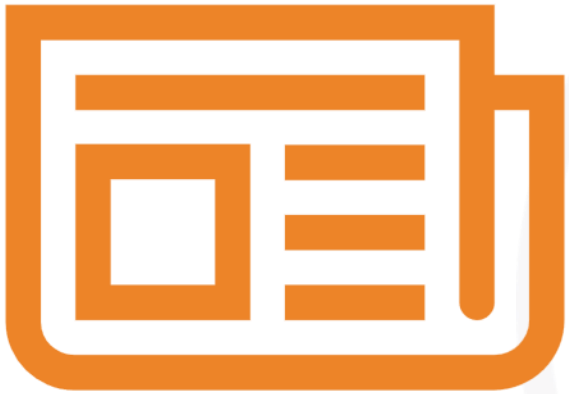
BACKGROUND

- It’s the era of Commercial Space Travel.
- Virgin galactic is providing suborbital space flights while Rocket lab is a satellite provider.
- Blue origin manufactures orbital and sub-orbital rockets
- Amongst all companies the most successful is Space X for it provides inexpensive space launches owing to the reuse of the first stage landing.

Problem Scenario

- Space Y wants to predict the reuse of first stage landing by Space X using the publicly available data.
- It involves Space X data collection using API, web scraping, data wrangling, EDA with SQL, data visualization, Developing dashboard and ML prediction

METHODOLOGY



- **Data Collection Methodology:**
  - Data collection from Space X public Domain And Wikipedia using API.
  - Data collection in performed using Python Beautiful Soup package.
  - The raw data is cleaned and transformed into an usable dataset by wrangling data using an API, Sampling data and dealing with NULLS.
- Perform Exploratory data analysis using SQL and Data Visualization
- Analyze launch site geo and proximities by developing an interactive map with launch site markers using Folium
- Develop a dashboard with Plotly Dash to visualize SPACE X data.
- Predictive analytics by building a ML pipeline to predict successful landing
  - Preprocessing
  - Data Standardization
  - Train test Split
- Test for models with best accuracy
  - Logistic Regression, K-Nearest Neighbor, Support Vector machine, Classifier, Decision Tree
- Tune models with Grid Search
- Output Confusion matrix.

Data Collection Methodology:

- **Data collection from Space X public Domain And Wikipedia using API.**
  - Request Space X API files
  - Json files containing information about payload delivered, launch specifications, landing specifications, and landing outcome.
  - Space X Falcon 9 Data Collection
- **Data collection Webscraping**
  - Data parsing using Beautiful Soup python package.
  - Find launch table info and build a dictionary
  - Cast dictionary to dataframe
  - Space X Falcon 9 Data Collection Web Scraping
- **Data wrangling** t
  - Transform data : The data is standardized and converted into a clean dataset that could be used for further processing.
  - Filter data to include only Falcon 9 launches
  - Space X Falcon 9 Data Collection Data Wrangling

Exploratory Data Analysis (EDA)

- **EDA using SQL:**
    - Loaded data set into SQLite 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
    - **Space X Falcon 9 EDA using SQL Notebook**

EDA WITH DATA VISUALIZATION

- EXPLORATORY DATA ANALYSIS WITH DATA VISUALIZATION
- DATA FEATURE ENGINEERING
- Space X data is downloaded and the dataset is converted into pandas data frame.
- Various relationships between first stage landing is compared in order to study the rate of successful first stage landing.
- The different relationships analyzed include
    - Flight number vs. Launch Site
    - Payload Mass vs. Launch Site
    - Success rate of each orbit
    - Flight Number and orbit type
    - Payload Mass and orbit type
    - Launch success yearly Trend
- Based on the observations features required to develop the prediction model are finalized.
- EDA with Data Visualization.

Build Interactive Map with Folium

- Lauch sites and geo are analyzed with Folium
- Mark launch site locations and proximities in an Interactive Map
- Analyze the map with the markers to discover patterns
- Finalize how to choose a launch site.
- Interactive Visual Analytics with Folium

Build Interactive Dashboard with Plotly Dash

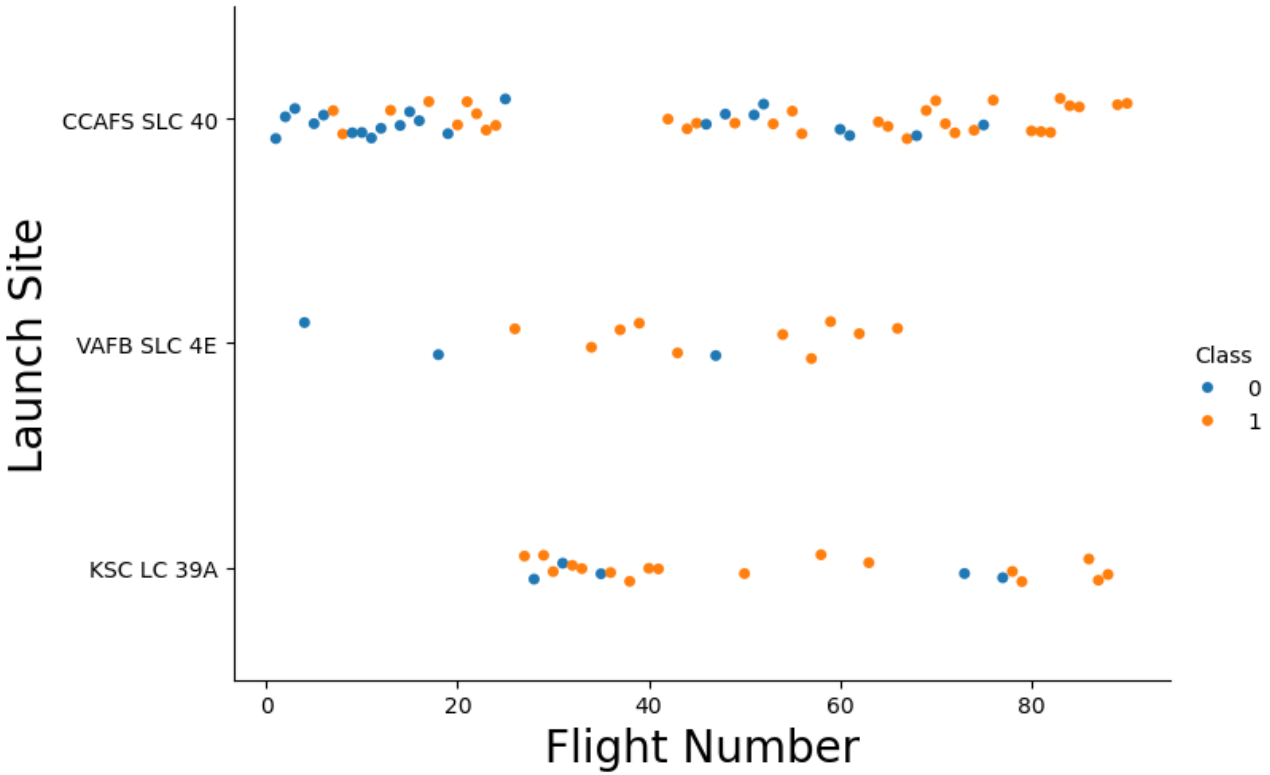
- Install required python libraries
- Download a skeleton application and a dataset
- Add launch site drop down component
- Add a call back function to render a success pie. Chart based on selected site dropdown
- Add a range slider to select payload
- Add a call back function to render success payload scatter chart
- Find insights visually from the plots
- Interactive Dashboard with Plotly Dash

Predictive Analytics with Machine Learning

- Load engineered datasets and define target and scaling.
- Define target and features.
- Feature Scaling and train test split.
- Trained with four different models : Logistic Regression, Support Vector Machine, Decision Tree and K Nearest Neighbour
- Finalize model hyperparameters
- Choose model via cross-validation.
- Evaluate on hold out test data
- Machine Learning Prediction Lab

RESULTS

EDA with Visualization

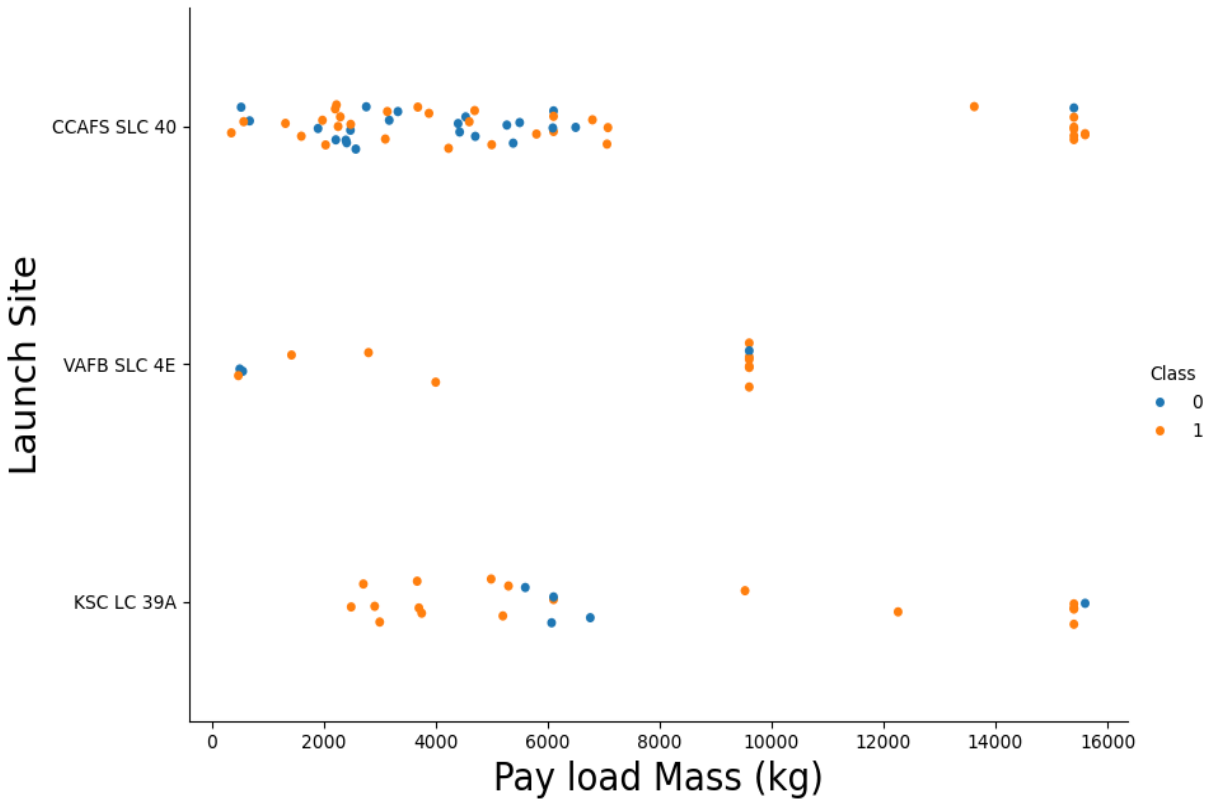


The graph shows the number of successful and failure from the launchsites and its trend with increasing flight number.

It is observed that

- Success Rate Improves Over Time
- Launch site performance trend
  - **CCAFS SLC 40 i**s the most frequently used launch site with success rate increasing over time.
  - **KSC LC 39A** is used in mid to high flight numbers and showed maximum success rate
  - **VAFB SLC 4E** Fewer launches and more failures

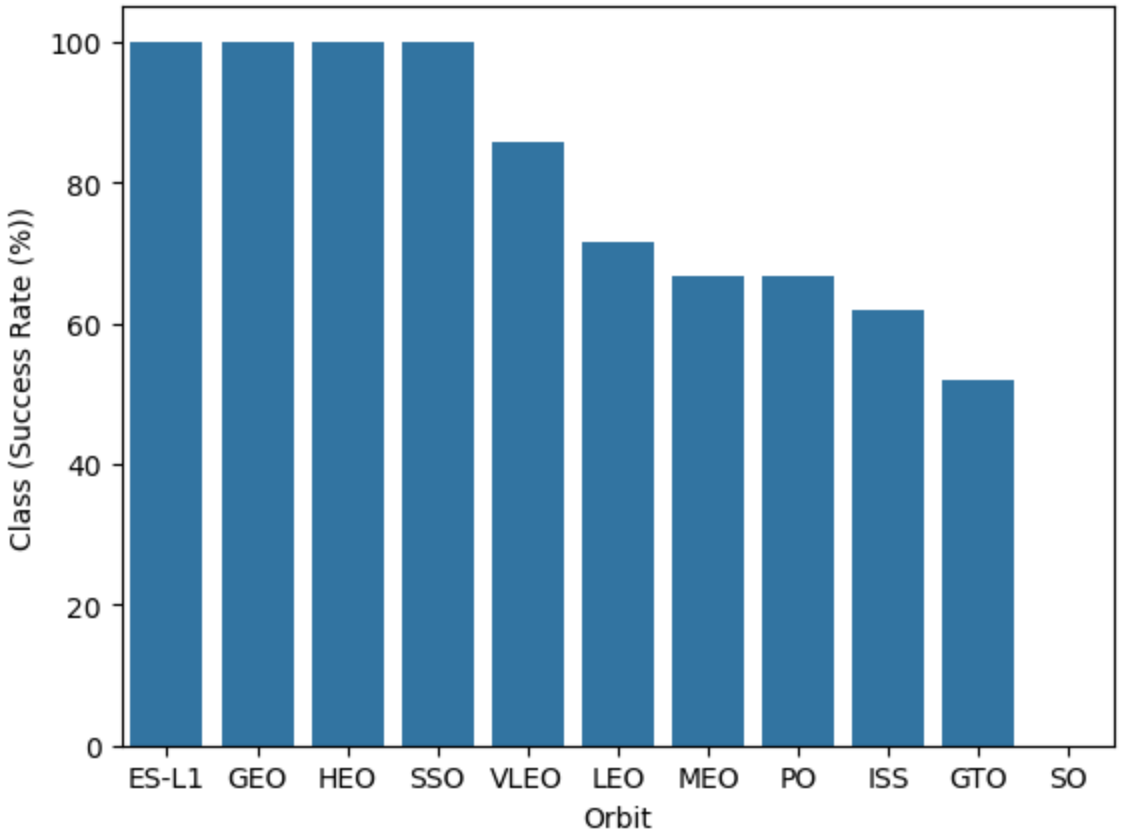
Relationship between Payload mass and Launch site



From the graph it is observed that

- Launch sites handle different payload masses
  - SLC 40 handles wide range of payload and is used for general launch operations
  - LC 39A is designed for high mass mission. It is capable of handling very high payload.
  - VAFB is used for polar orbit missions
- Success decreases with heavier payloads as heavier payloads have lesser fuel for landing.
- Moderate payloads resulted in an increase in success rate. Boosters are more recoverable when moderate payload is used.

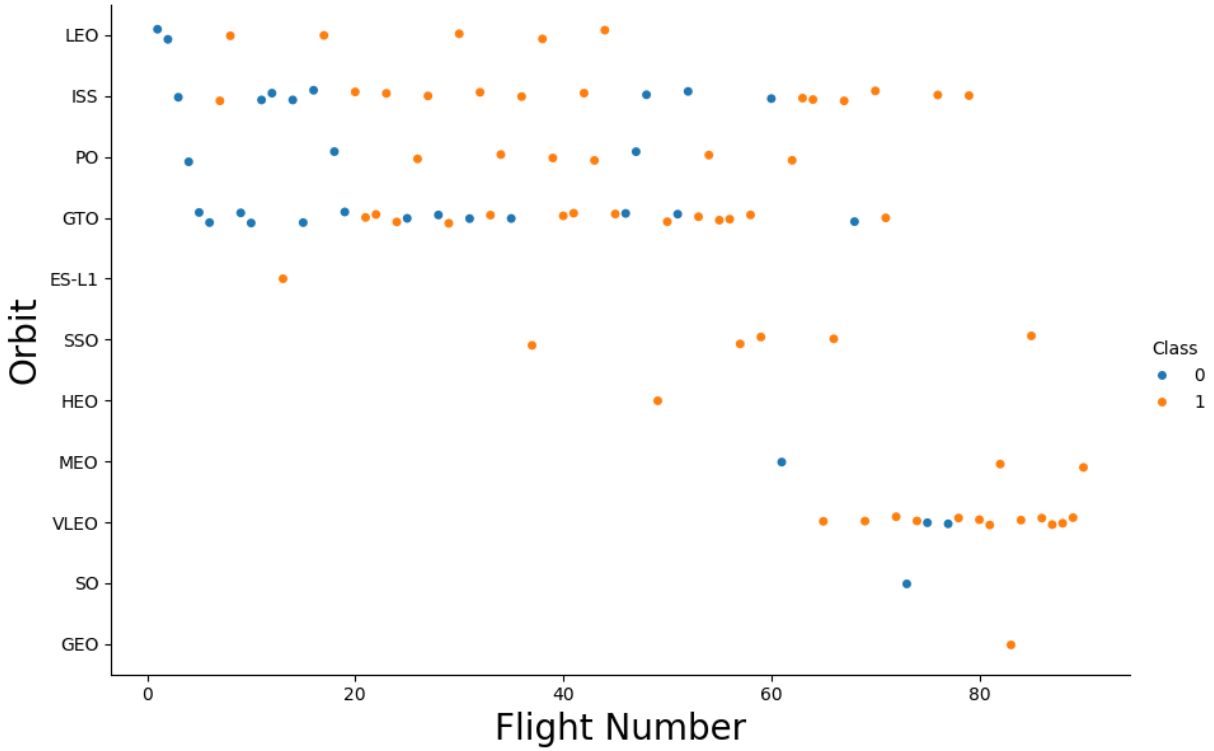
Relationship between success rate of each orbit type



From the Graph it is evident that:

- Mission such as ES-L1, GEO, HEO, SSO with high values had a higher success rate.
- Missions such as VLEO, LEO, MEO, PO, ISS with mid-range values covered the common operational orbits such as the earth observation missions, International space station, satellite launch.
- GTO and So with very heavy payloads had very low success rates.

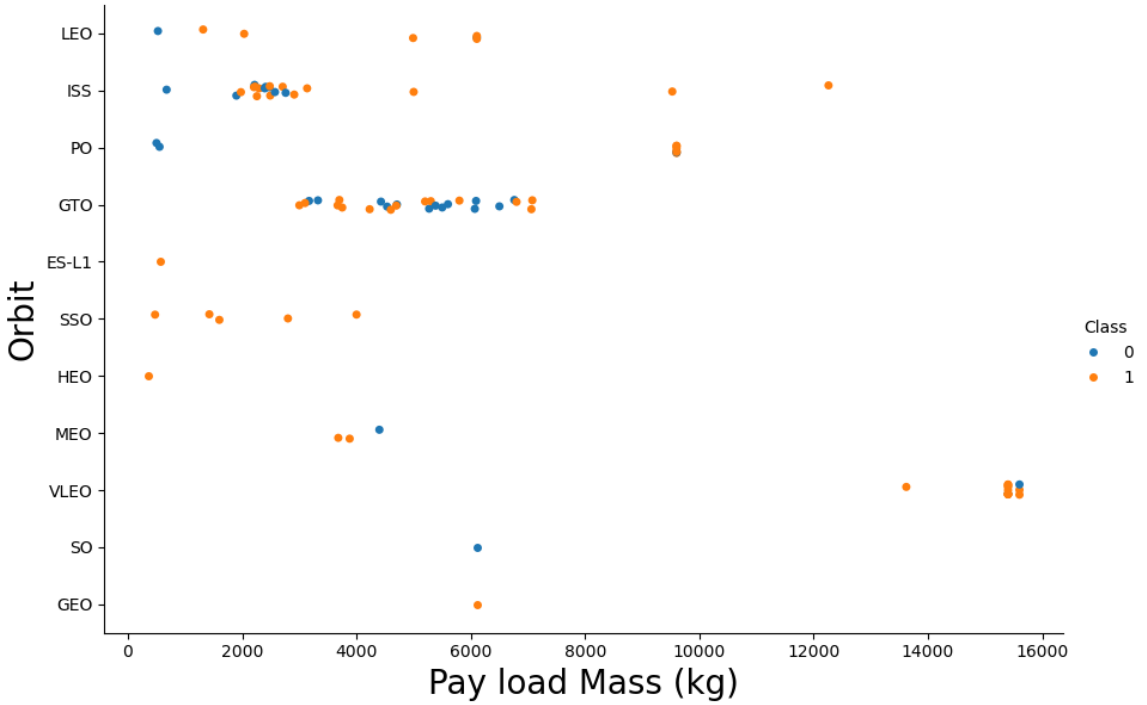
Relationship between Flight Number and Orbit type



From the figure it can be observed that

- Mission success improves with time across *all* orbit types.
- Strong evidence of learning curve and booster reusability advancements
- GTO remains more challenging for recovery
- Orbit types diversify with increase in flight numbers

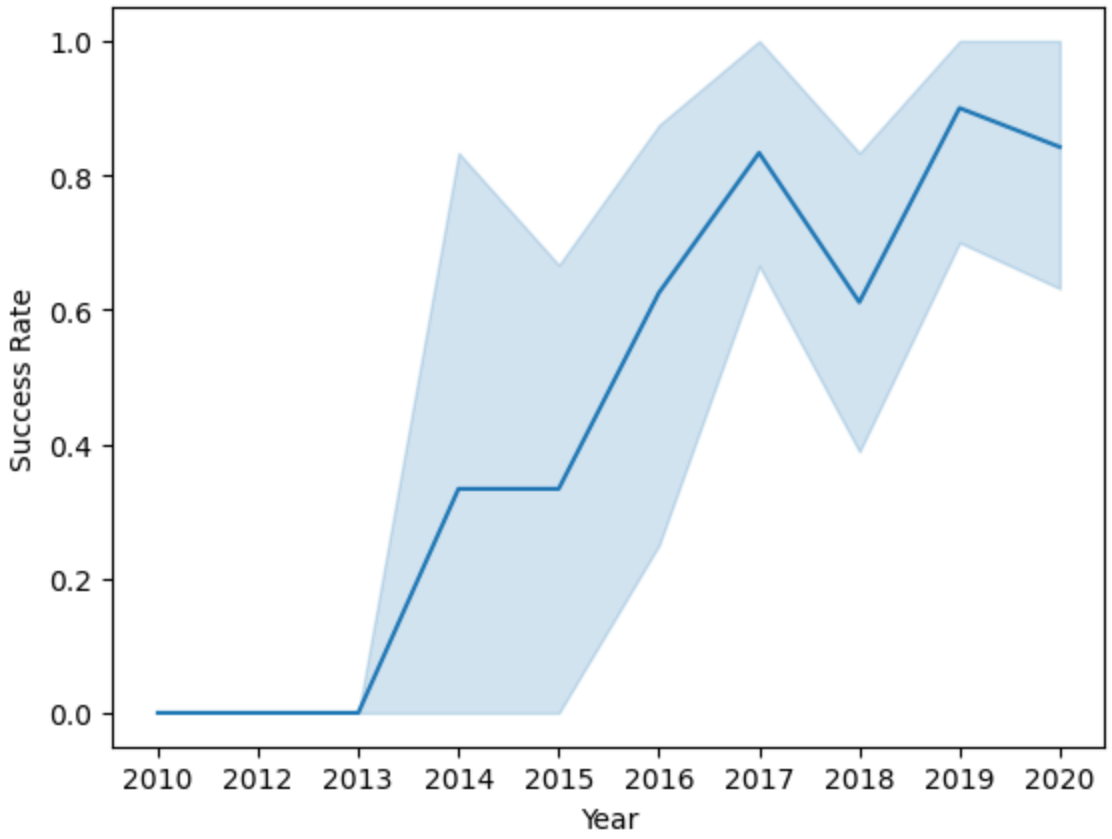
Relationship between Payload Mass and Orbit type



From the figure it is derived that:

- Heavy payloads need more fuel to reach orbit → less fuel left for return burn → higher chance of landing failure.
- Orbit type influences payload mission and outcome
- Cluster patterns reveal mission specialization

Visualize the launch success yearly trend



The figure depicts:

- **Early years** (2010-2013) had no success when booster recovery experiments were still in practise.
- Major f0o0cus was o0n launch success rather than recovering first stage.
- **2014-2017 breakthrough phase** where success rate started raising. Controlled oceanic landing attempts.
- 2017–2018: Strong performance with a dip probably due to complex missions.
- 2019–2020: Consistent maturity nearly perfect success and reuse.

EDA with SQL

Unique Launch sites are queried using SQL and the output is given below

Task 1

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

In [10]:
```
%sql SELECT DISTINCT LAUNCH_SITE as "Launch_Sites" FROM SPACEXTBL;
```

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

Out[10]:

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

Records from the Launch site with the string 'CCA" is displayed

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.
```

[11]:

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

The query is used to find the total payload mass in kgs.

Task 3

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

[12]:
```
%sql SELECT SUM(PAYLOAD_MASS__KG_) as "Total Payload Mass(Kgs)", Customer FROM 'SPACEXTBL' WHERE Customer = 'NASA (CRS)';
```

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

[12]:

Total Payload Mass(Kgs)	Customer
45596	NASA (CRS)

Average Payload mass carried by booster version F9 v1.1

Task 4

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

[13]:
```
%sql SELECT AVG(PAYLOAD_MASS__KG_) as "Payload Mass Kgs", Customer, Booster_Version FROM 'SPACEXTBL' WHERE Booster_Version LIKE '
```

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

[13]:

Payload Mass Kgs	Customer	Booster_Version
2534.6666666666665	MDA	F9 v1.1 B1003

First successful ground landing date

Task 5

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

Hint:Use min function

[20]:
```
%sql SELECT MIN("Date") FROM SPACEXTBL WHERE "Landing_Outcome" = 'Success (ground pad)';
```

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

[20]:

MIN("Date")
2015-12-22

- This query returns the first successful ground pad landing date.
- First ground pad landing wasn’t until the end of 2015.
- Successful landings in general appear starting 2014.

Successful drone ship landing with payload between 4000 and 6000

[27]:
```
%%sql
SELECT DISTINCT Booster_Version
FROM SPACEXTBL
WHERE "Landing_Outcome" = 'Success (drone ship)'
  AND Payload_Mass__kg_ > 4000
  AND Payload_Mass__kg_ < 6000;
```

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

[27]:

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

Total number of successful and failure mission outcomes

Task 7

List the total number of successful and failure mission outcomes

[28]:
```
%sql SELECT "Mission_Outcome", COUNT("Mission_Outcome") as Total FROM SPACEXTBL GROUP BY "Mission_Outcome";
```

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

[28]:

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

Total number of successful and failure mission outcomes record

[26]:

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

Boosters that carried maximum payload

Task 8

List all the booster_versions that have carried the maximum payload mass, using a subquery with a suitable aggregate function.

[29]:
```
%sql SELECT "Booster_Version",Payload, "PAYLOAD_MASS__KG_" FROM SPACEXTBL WHERE "PAYLOAD_MASS__KG_" = (SELECT MAX("PAYLOAD_MASS__
```

[29]:

Booster_Version	Payload	PAYLOAD_MASS__KG_
F9 B5 B1048.4	Starlink 1 v1.0, SpaceX CRS-19	15600
F9 B5 B1049.4	Starlink 2 v1.0, Crew Dragon in-flight abort test	15600
F9 B5 B1051.3	Starlink 3 v1.0, Starlink 4 v1.0	15600
F9 B5 B1056.4	Starlink 4 v1.0, SpaceX CRS-20	15600
F9 B5 B1048.5	Starlink 5 v1.0, Starlink 6 v1.0	15600
F9 B5 B1051.4	Starlink 6 v1.0, Crew Dragon Demo-2	15600
F9 B5 B1049.5	Starlink 7 v1.0, Starlink 8 v1.0	15600
F9 B5 B1060.2	Starlink 11 v1.0, Starlink 12 v1.0	15600
F9 B5 B1058.3	Starlink 12 v1.0, Starlink 13 v1.0	15600
F9 B5 B1051.6	Starlink 13 v1.0, Starlink 14 v1.0	15600
F9 B5 B1060.3	Starlink 14 v1.0, GPS III-04	15600
F9 B5 B1049.7	Starlink 15 v1.0, SpaceX CRS-21	15600

Failed landing ship records

```
[23]: %%sql
SELECT
    substr(Date,1,4) AS Year,
    substr(Date,6,2) AS Month,
    Booster_Version,
    Launch_Site,
    Payload,
    PAYLOAD_MASS__KG_,
    Mission_Outcome,
    Landing_Outcome
FROM SPACEXTBL
WHERE substr(Date,1,4)='2015'
  AND Landing_Outcome='Failure (drone ship)'
LIMIT 2;

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

Done.

:

Month	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Mission_Outcome	Landing_Outcome
01	F9 v1.1 B1012	CCAFS LC-40	SpaceX CRS-5	2395	Success	Failure (drone ship)
04	F9 v1.1 B1015	CCAFS LC-40	SpaceX CRS-6	1898	Success	Failure (drone ship)

LANDING OUTCOME

Landing_Outcome	Outcome_Count
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

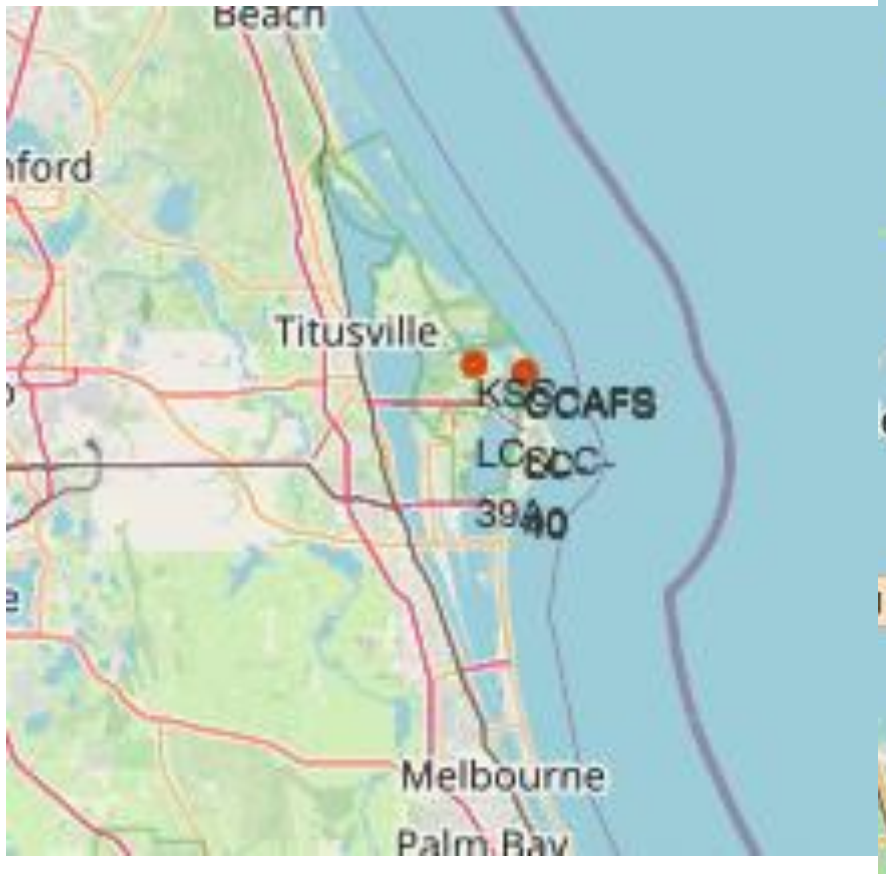
INTERACTIVE MAPS WITH FOLIUM

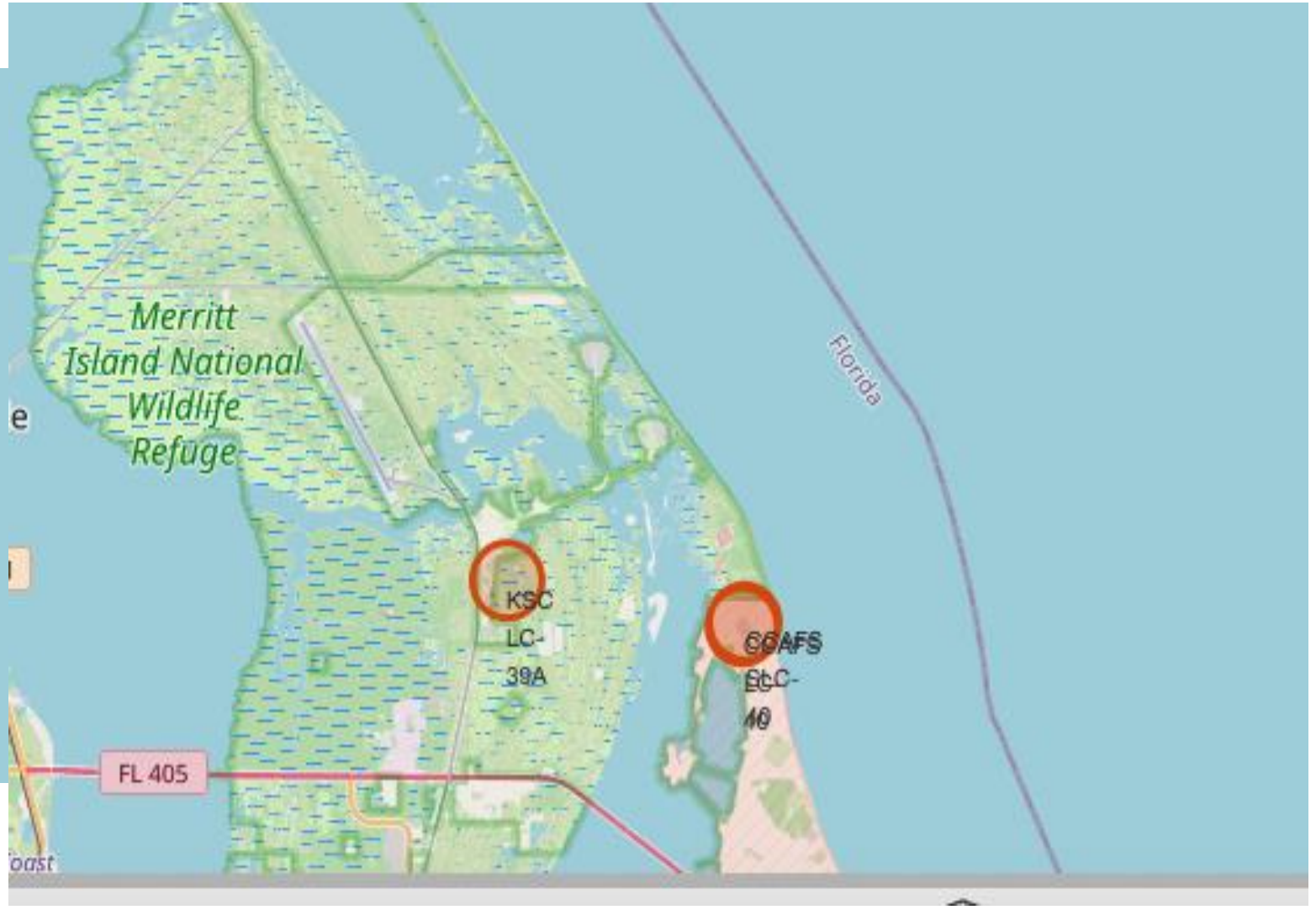
COORDINATES FOR EACH SITE

	Launch Site	Lat	Long
0	CCAFS LC-40	28.562302	-80.577356
1	CCAFS SLC-40	28.563197	-80.576820
2	KSC LC-39A	28.573255	-80.646895
3	VAFB SLC-4E	34.632834	-120.610745

Initial center location to be NASA Johnson Space Center at Houston, Texas.

Launch Site Locations

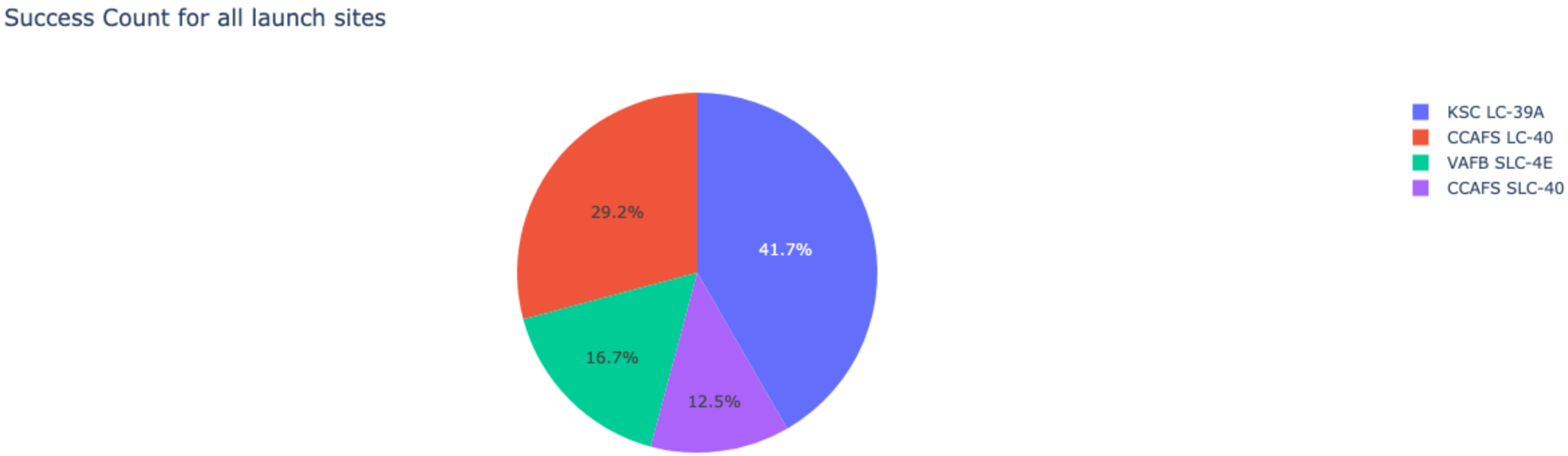


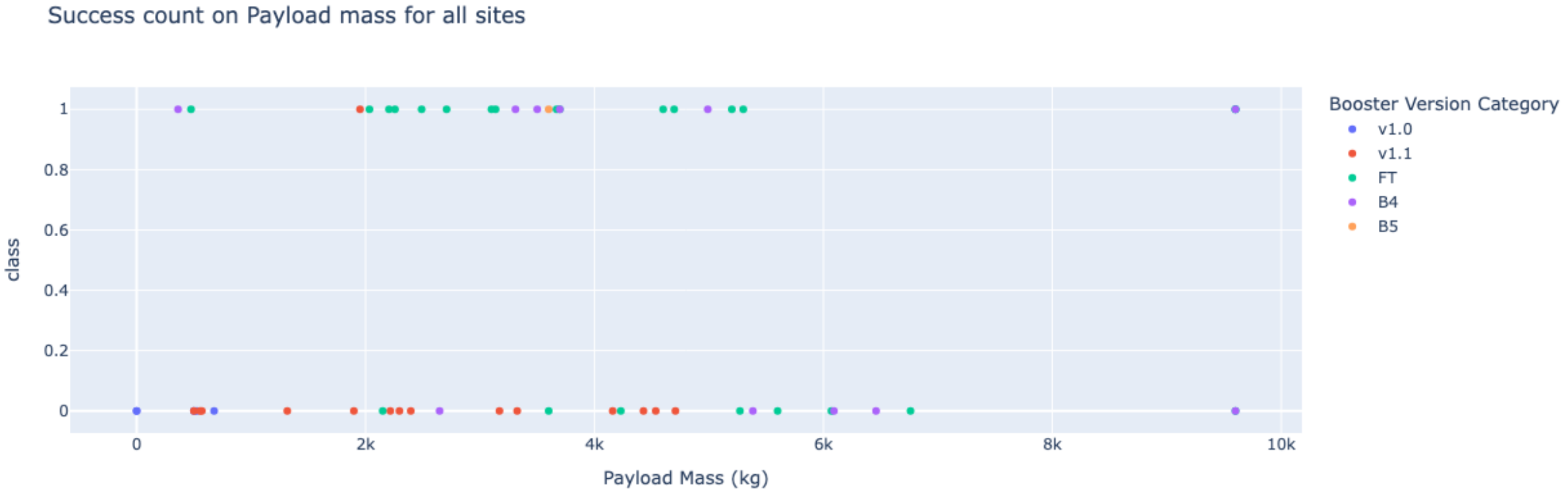


Launch Site Locations

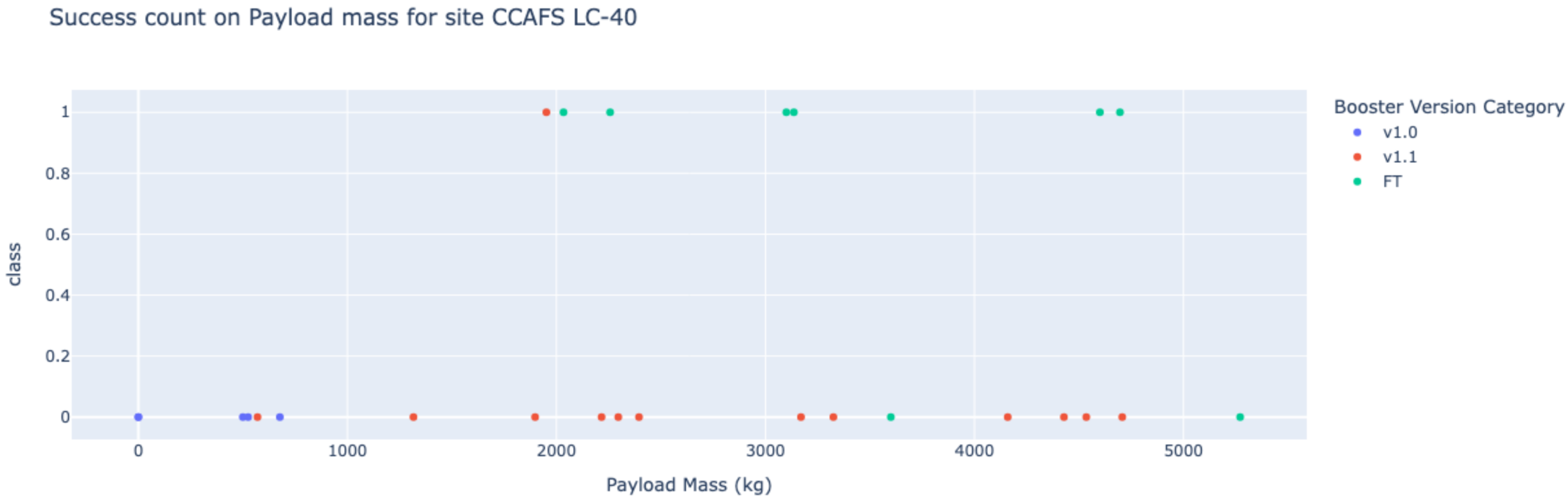
Color coded launch markers

Dashboard with Plotly Dash

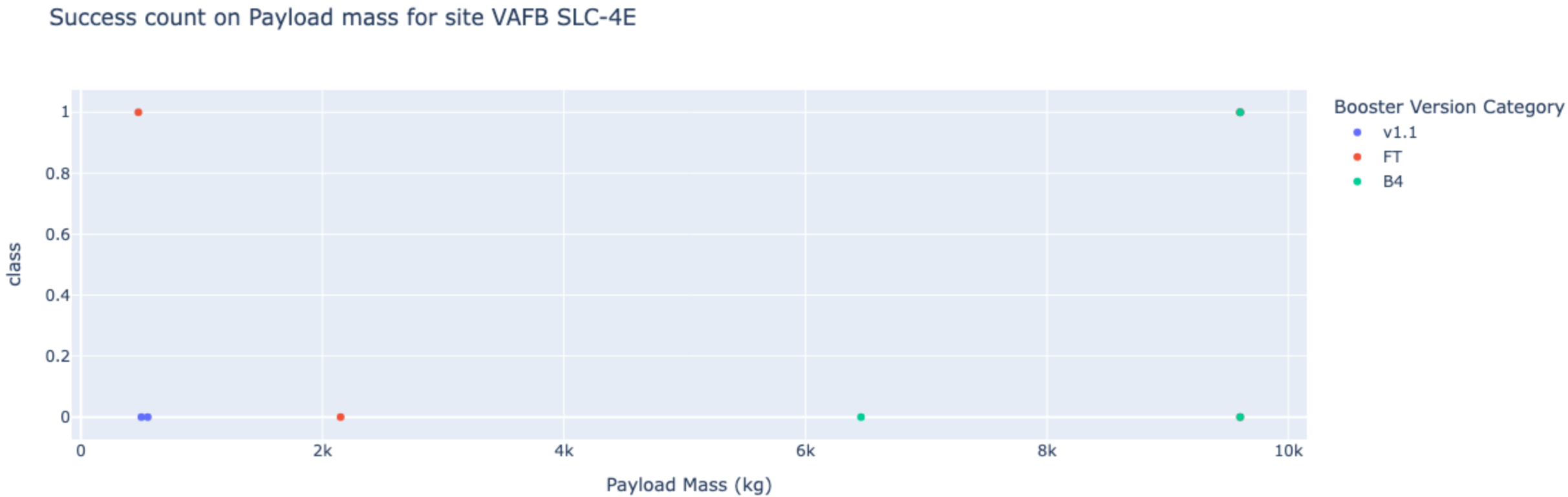



Success count on Payload mass for all sites
class
1
0.8
0.6
0.4
0.2
0
0
2k
4k
6k
8k
10k
Payload Mass (kg)
Booster Version Category
v1.0
v1.1
FT
B4
B5

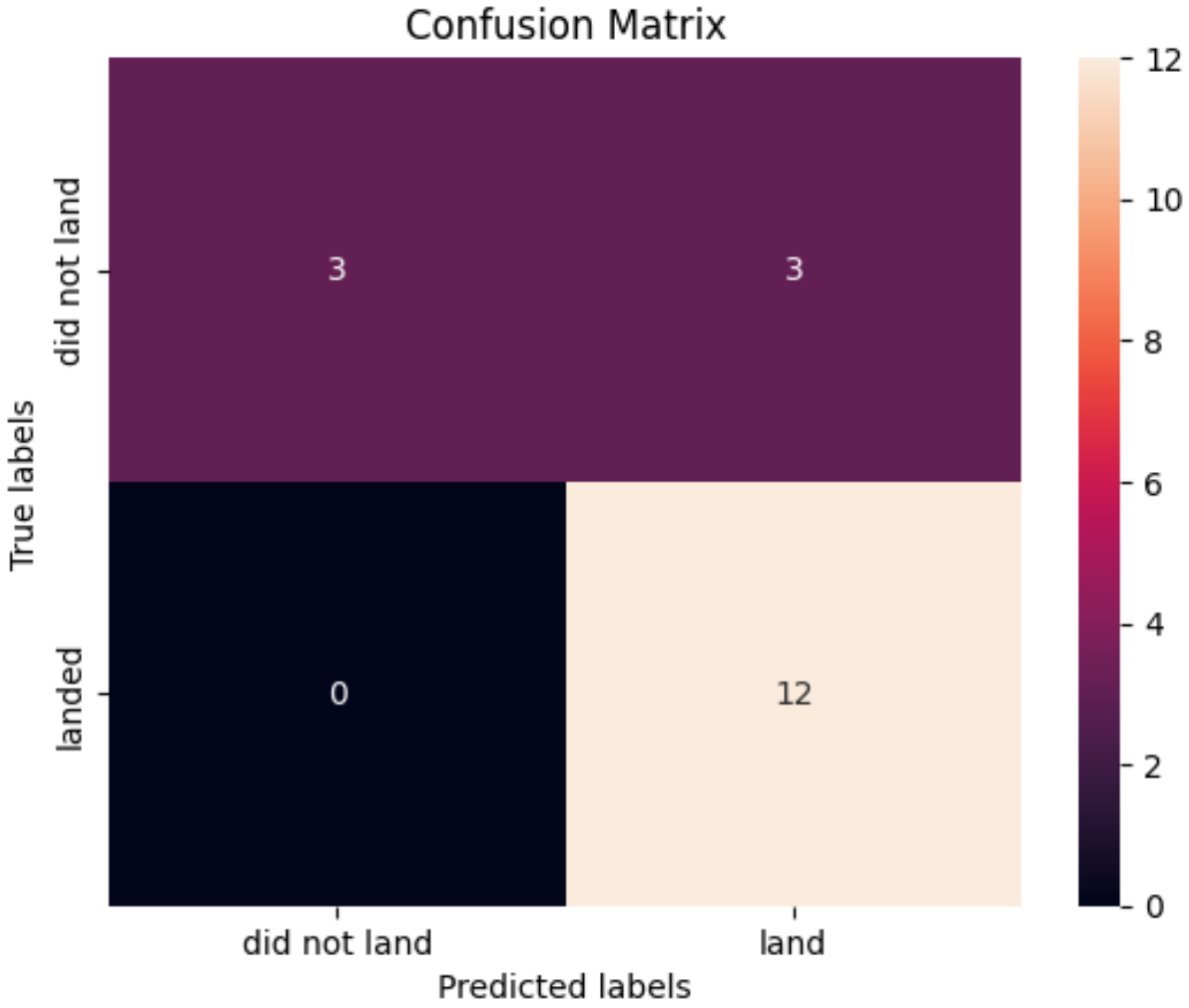
Total Success Launches for site CCAFS LC-40


Success count on Payload mass for site CCAFS LC-40
class
1
0.8
0.6
0.4
0.2
0
0
1000
2000
3000
4000
5000
Payload Mass (kg)
Booster Version Category
v1.0
v1.1
FT

Total Success Launches for site VAFB SLC-4E


Success count on Payload mass for site VAFB SLC-4E
class
1
0.8
0.6
0.4
0.2
0
0
2k
4k
6k
8k
10k
Payload Mass (kg)
Booster Version Category
v1.1
FT
B4

Logistic Regression



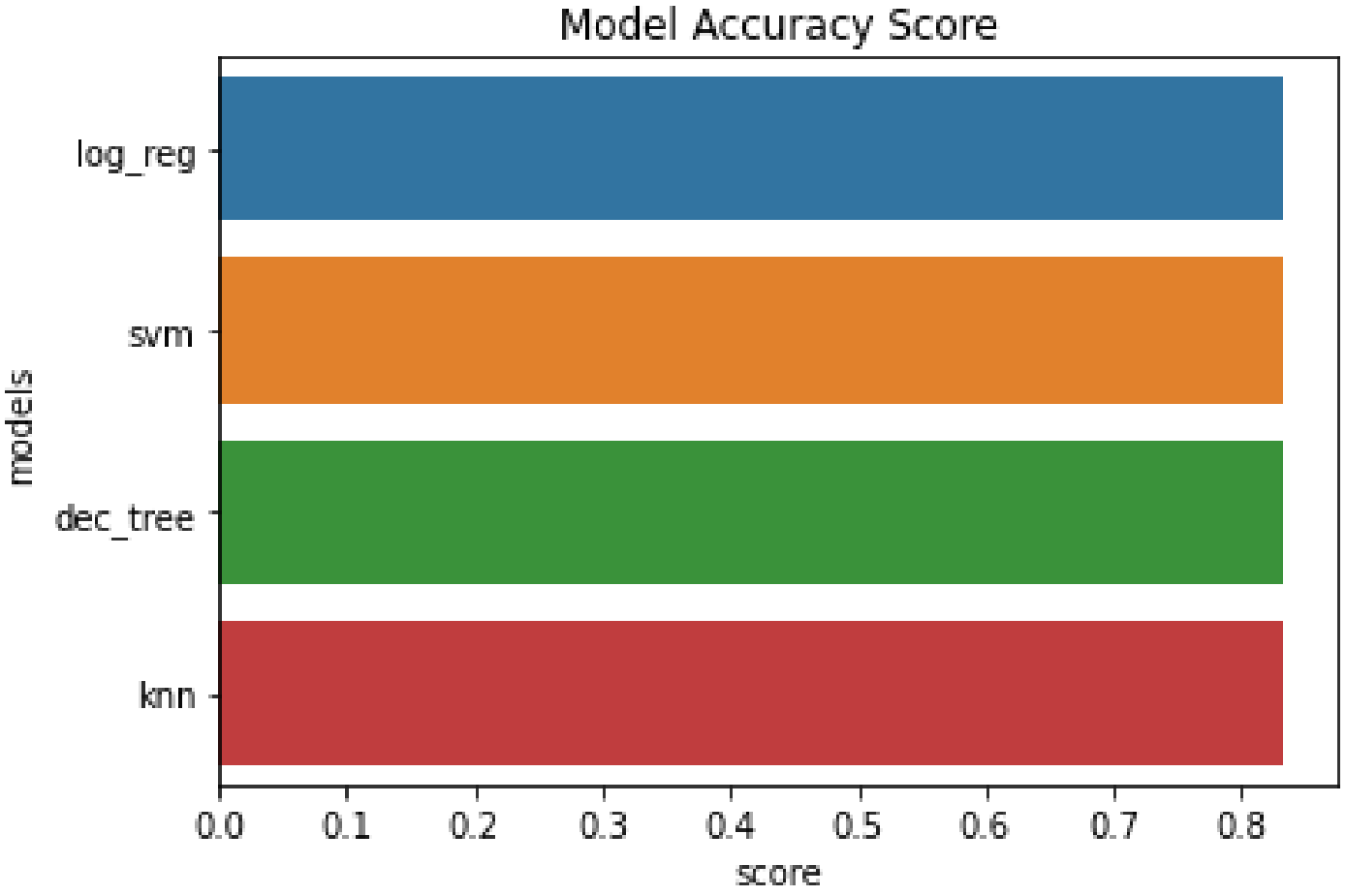
Since all models performed the same for the test set, the confusion matrix is the same across all models.

The models predicted 12 successful landings when the true label was successful landing.

The models predicted 3 unsuccessful landings when the true label was unsuccessful landing.

The models predicted 3 successful landings when the true label was unsuccessful landings (false positives).

Our models over predict successful landings.



All models had virtually the same accuracy on the test set at 83.33% accuracy. It should be noted that test size is small at only sample size of 18. This can cause large variance in accuracy results, such as those in Decision Tree Classifier model in repeated runs.

CONCLUSION

- Our task: to develop a machine learning model for Space Y who wants to bid against SpaceX
- The goal of model is to predict when Stage 1 will successfully land to save ~$100 million USD
- Used data from a public SpaceX API and web scraping SpaceX Wikipedia page
- Created data labels and stored data into a DB2 SQL database
- Created a dashboard for visualization
- We created a machine learning model 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