IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Pere Villega
28.01.2024

Outline

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

Executive Summary

Summary of methodologies

- Data Collection: API, Web Scraping
- Data Wrangling
- Exploratory Data Analysis (EDA): SQL, Pandas, Maplotlib
- Interactive Visual Analysis: Folium, Plotly Dash
- Predictive Analysis (Classification)

Summary of all results

- Summary of Findings
- EDA Results
- Interactive Maps and Dashboards
- Predictive Outcomes
- Best hyperparameters for Logistic Regression, SVM, Decision Tree, KNN classifiers

Introduction

- **Project background and context**

This project aims to predict the successful landing of the Falcon 9 first stage, which is a key factor in determining the cost of a rocket launch.

We analyse characteristics of successful and failed launches. This data can help us determine which conditions impact a launch, to increase our success rate.

Problems you want to find answers

- Understand what variables affect the success for a launch and landing of the stage

- Understand the conditions that can help increase the success rate

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - Data was collected using the Space X REST API
  - It was complemented with data scrapped (Web scrapping) from Wikipedia
- Perform data wrangling
  - The data collected from the API is in JSON. The data from web scrapping is in HTML.
  - We perform a conversion step to transform both into Pandas dataframes.
- 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 Machine Learning algorithms for predictive analysis

Data Collection

SpaceX REST API

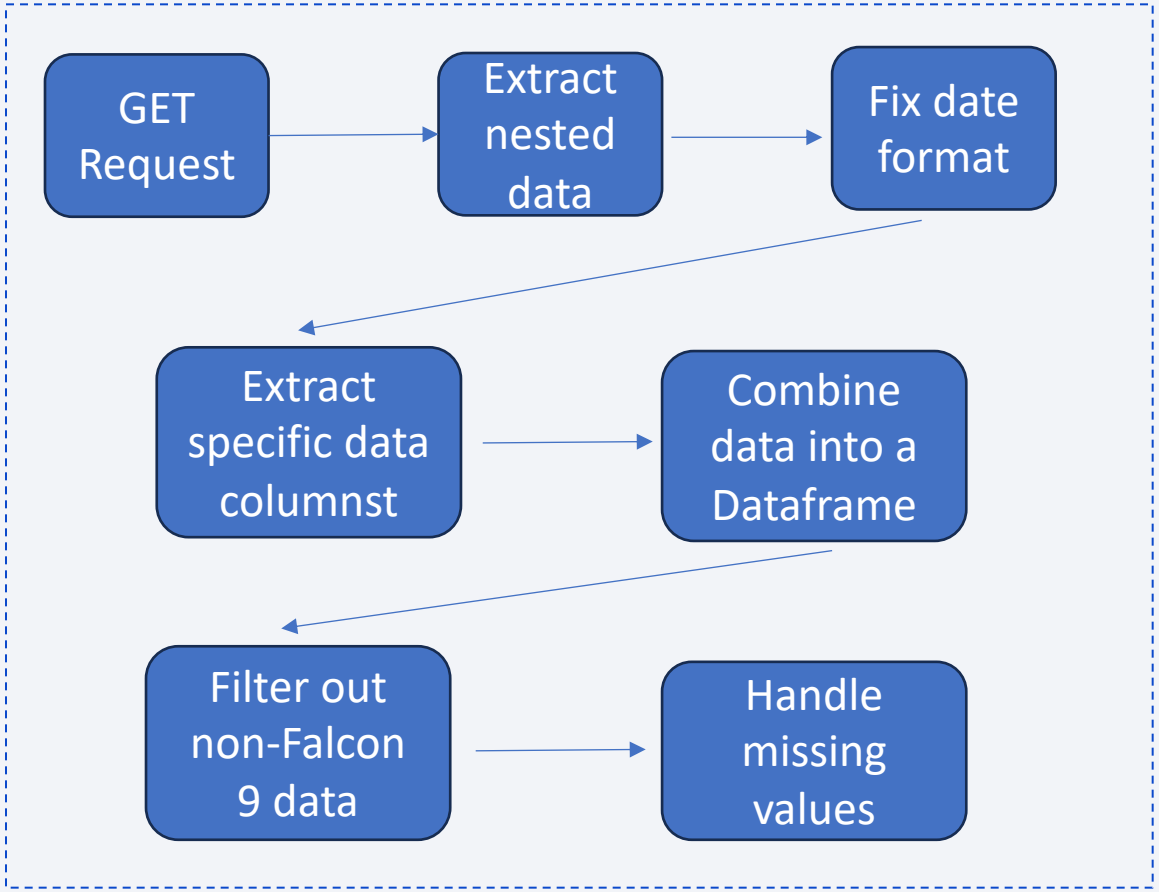
- The data was collected connecting to the public API endpoint
- The library used was Requests, and the data was obtained as JSON objects
- Data was later converted to Pandas dataframes

Web Scrapping

- Some data for launch records was scrapped from Wikipedia
- The library used was BeautifulSoup, to read HTML tables and parse its data
- Data was later converted to Pandas dataframes

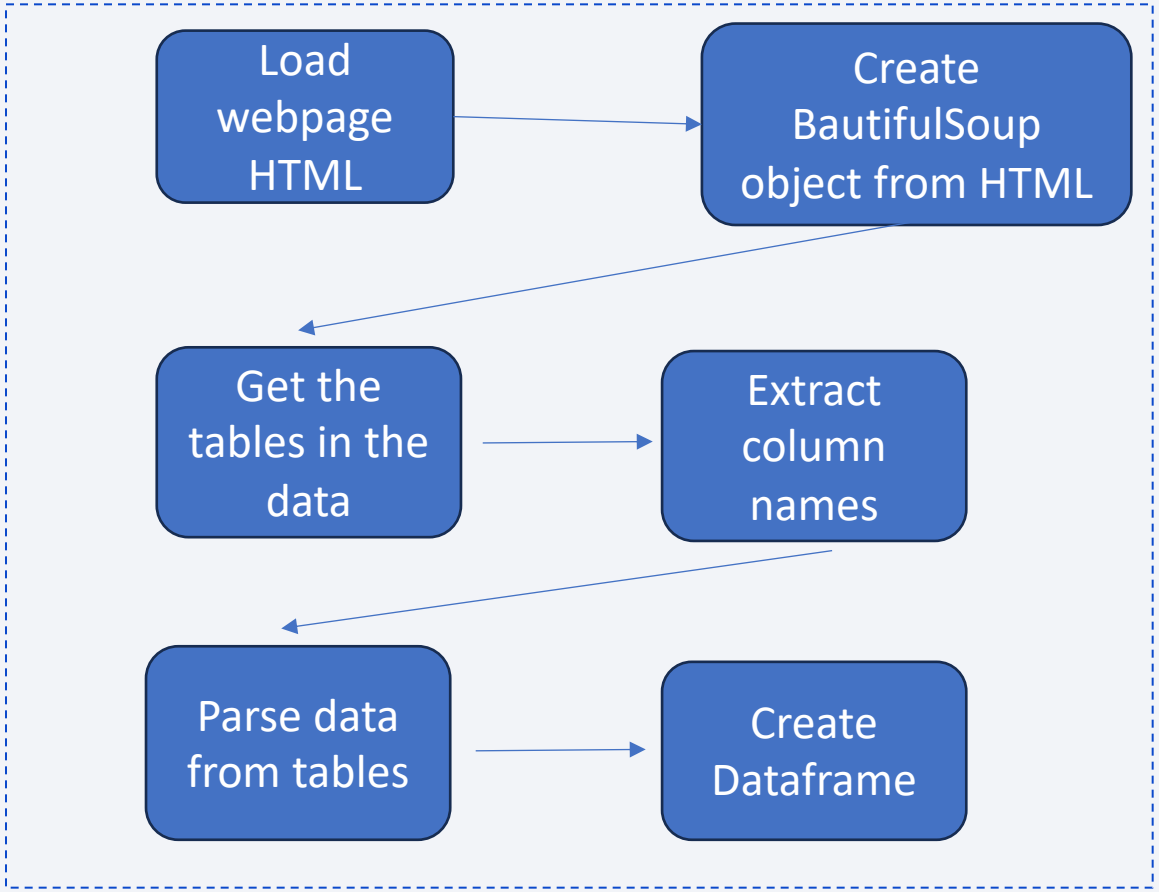
Data Collection – SpaceX API

- SpaceX publishes data via an API. That data is public
- Make a GET request to that API and receive the response
- Process the response (date formats, refine columns
- Combine columns into a Daraframe
- Link to notebook



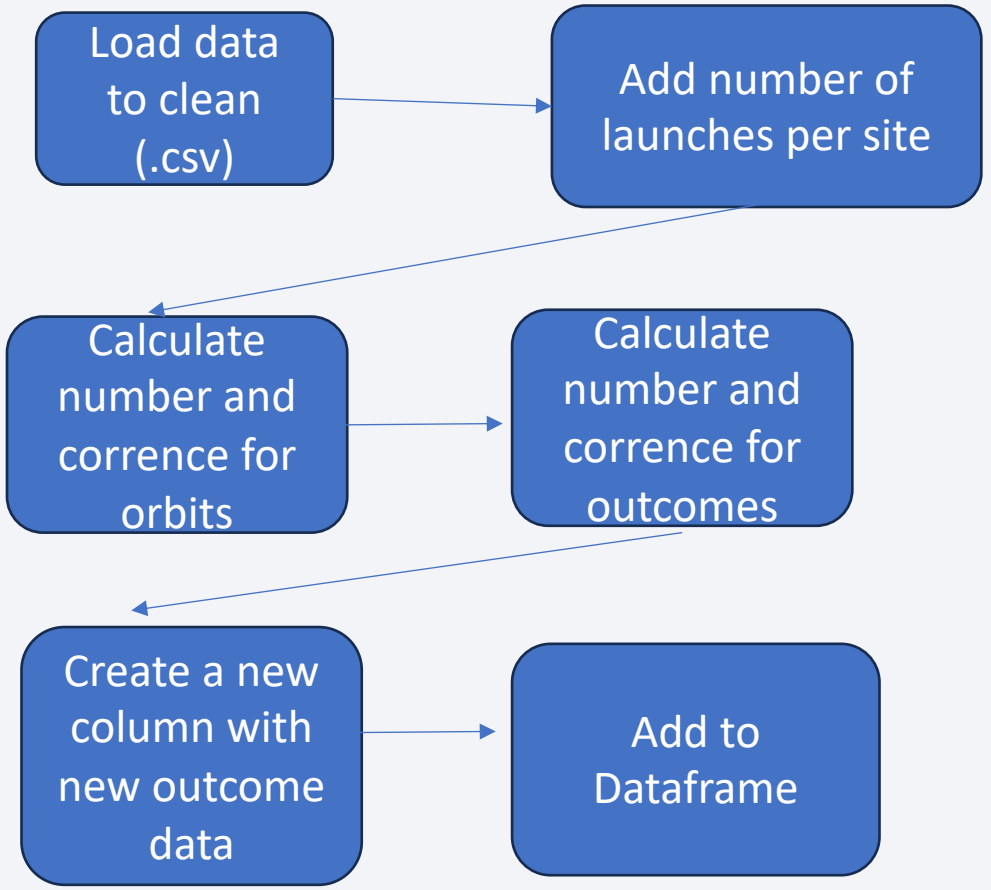
Data Collection - Scraping

- Wikipedia offers data on SpaceX launches
- Scrap tables using BeautifulSoup to get launch data
- Convert the data into a Pandas Dataframe
- Link to notebook



Data Wrangling

- Data from previous sections needs to be cleaned for processing
- Some fields need clean-up (launch sites, orbit types, outcomes)
- Some types were converted to binary classification (0 or 1 values)
- Cleaned data was added to the Dataframe
- Link to notebook



EDA with Data Visualization

The following charts are used:

- Scatterplot for mission outcome by launch site and flight number
- Scatterplot for mission outcome by launch site and payload
- Bar chart for mission outcome by orbit type
- Scatterplot for mission outcome by orbit type and flight number
- Scatterplot for mission outcome by orbit type and payload
- Line plot for mission outcome by year

- Link to notebook

EDA with SQL

Queries using SQL created for:

- Launch sites (keywords: distinct, like '%', limit)
- Payload masses (keywords: sum, avg)
- Dates (keywords: min)
- Booster types (keywords: count, group by, max, distinct, limit)
- Mission outcomes (keywords: like '%', count, group by, order by)

- Link to notebook

Build an Interactive Map with Folium

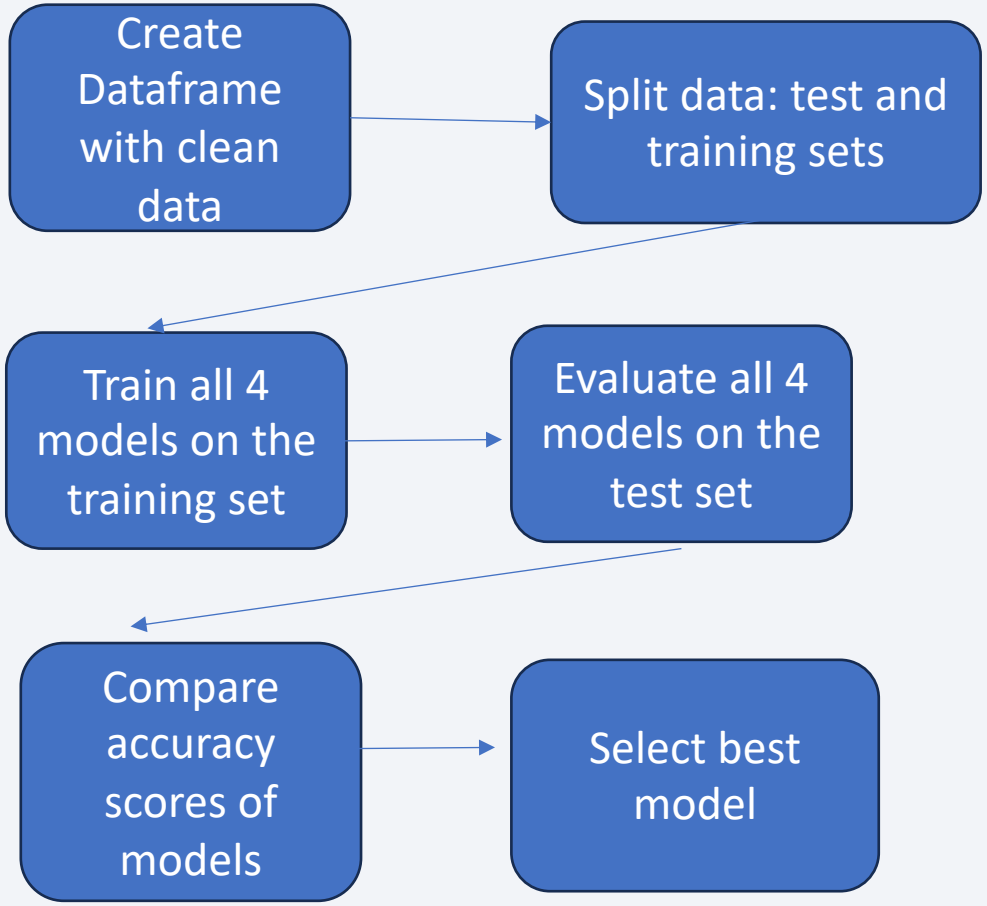
- Markers added for launch sites and NASA Johnson Space Center
- Circles added at the launch sites
- Lines added to show distance to nearby features
- Link to notebook

Build a Dashboard with Plotly Dash

- Added an input dropdown to select a launch site (there are 4)
- Added a slider to select different payload ranges
- The pie chart shows the split between success and failed missions on the selected site
- The scatterplot shows the landings split by payload mass, booster, and outcome
- Link to notebook

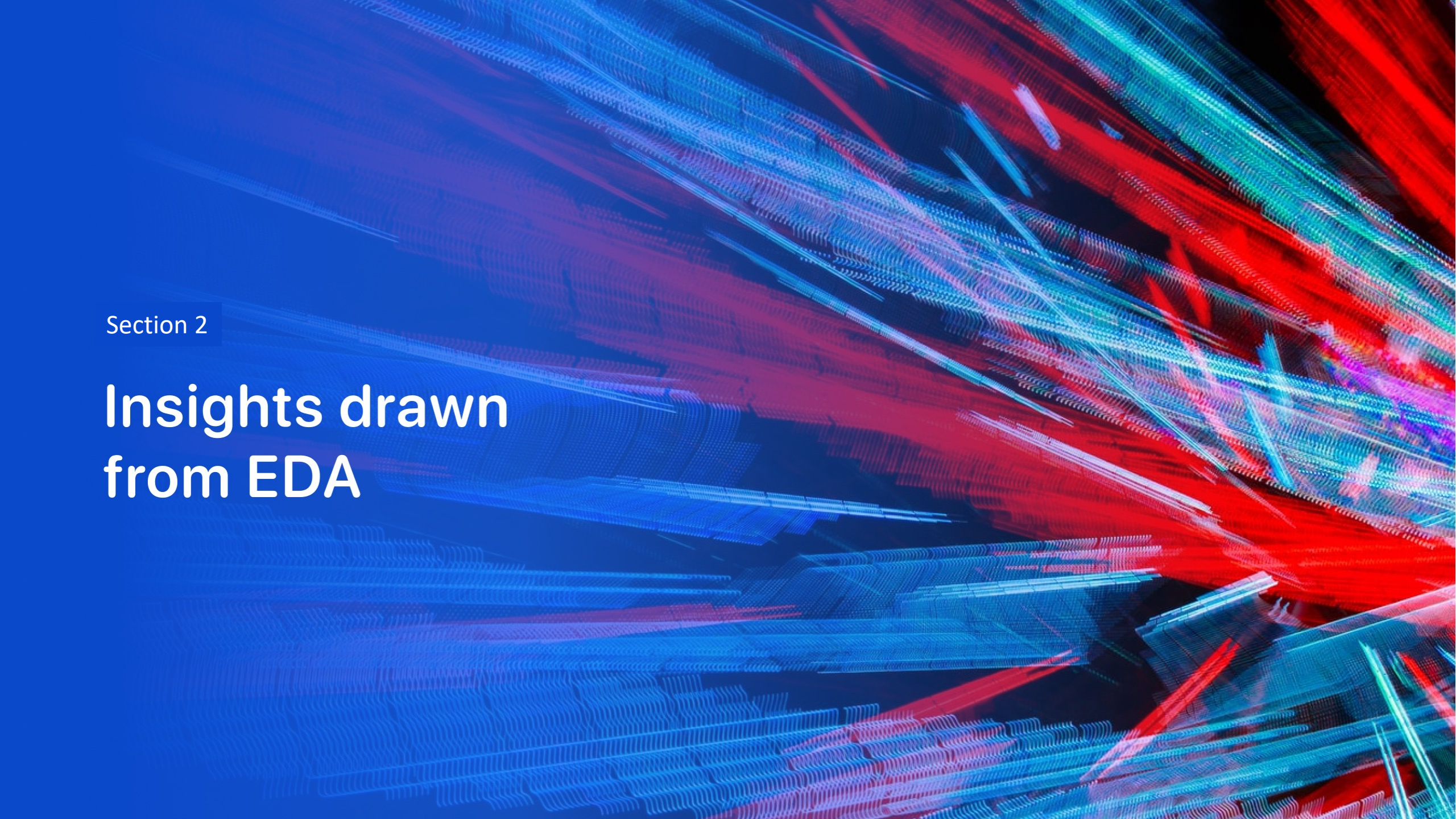
Predictive Analysis (Classification)

- Dataset was split into training and testing sets
- Different models were trained: Logistic Regression, SVM, Decision Tree, and KNN.
- For each mode, we found the best hyper-parameters
- Each model was scored for accuracy using the test set
- Link to notebook



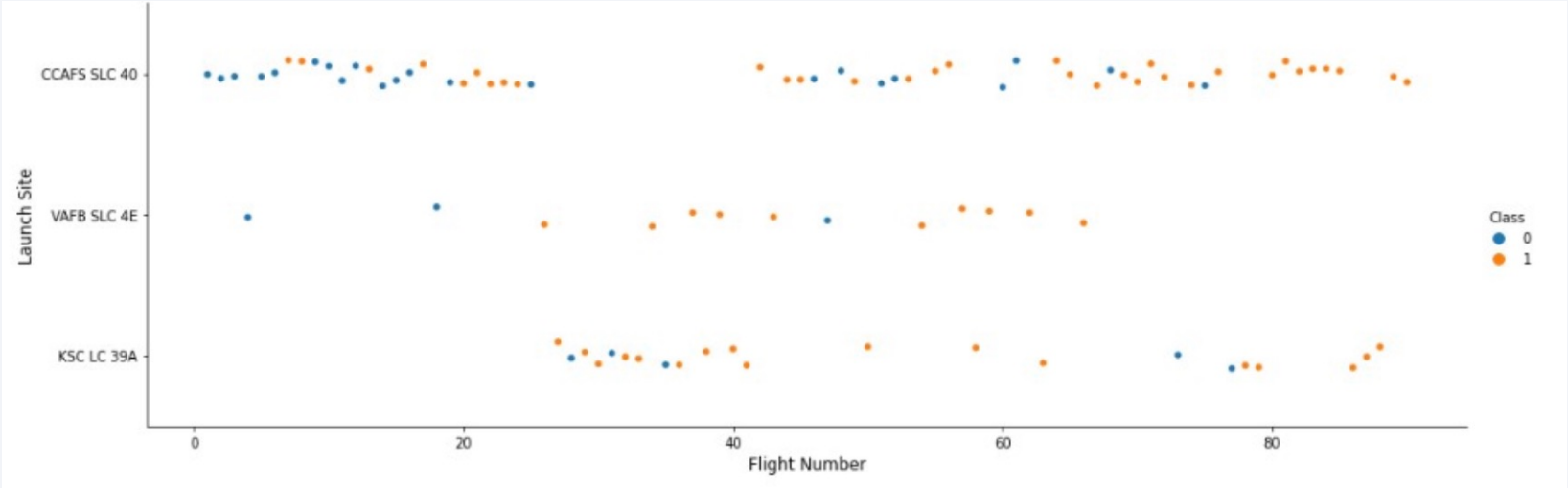
Results

- Exploratory data analysis results
- Interactive analytics demo in screenshots
- Predictive analysis results


Section 2
Insights drawn from EDA

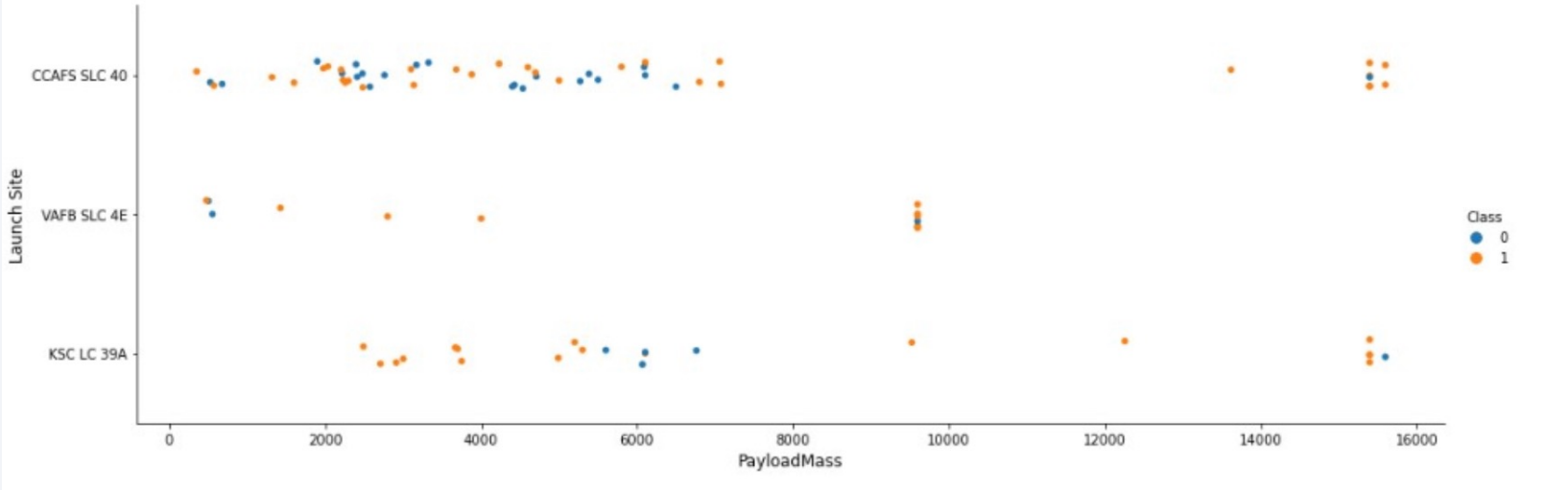
Flight Number vs. Launch Site

- Success rate is different per launch site
- Success rate increases over time, as more launches happen



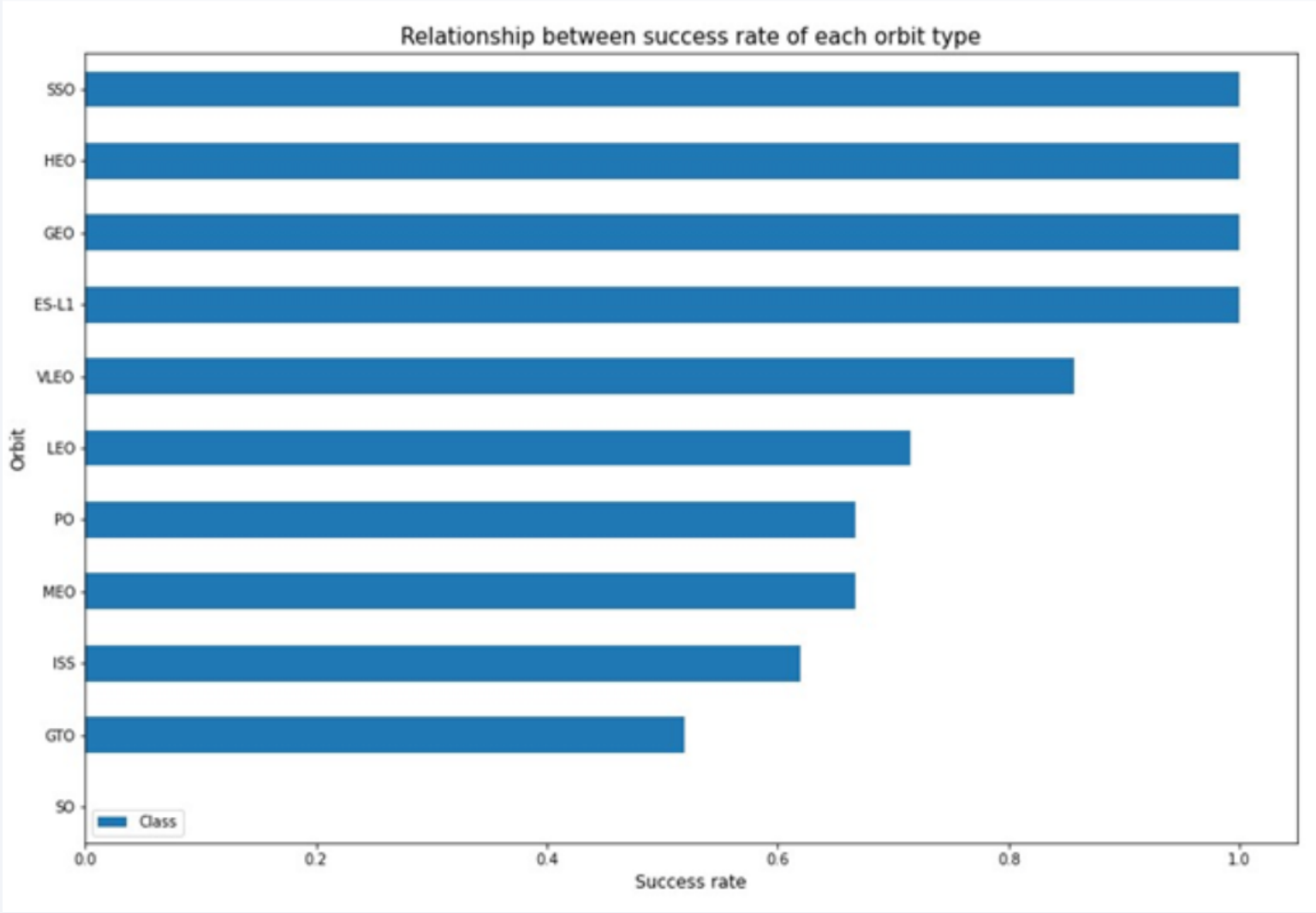
Payload vs. Launch Site

- Not all payload masses are launched from all sites.
- Success isn't necessarily linked to payload mass across all sites, but does in specific sites



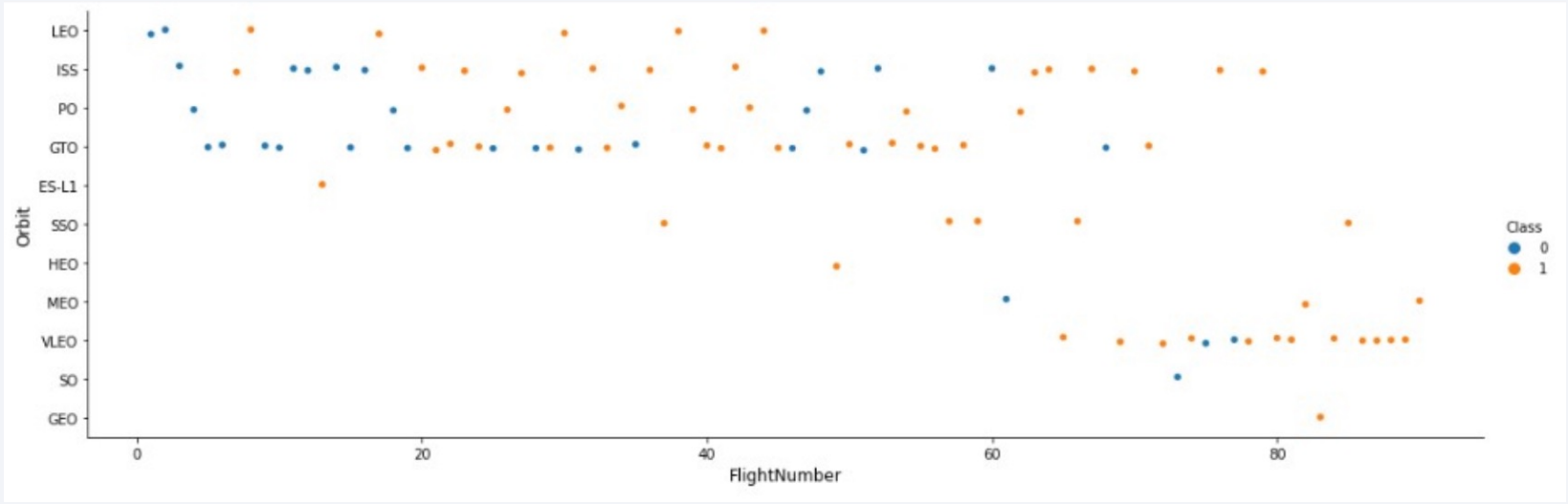
Success Rate vs. Orbit Type

- 4 orbits have perfect success rate
- SO orbits have no successes



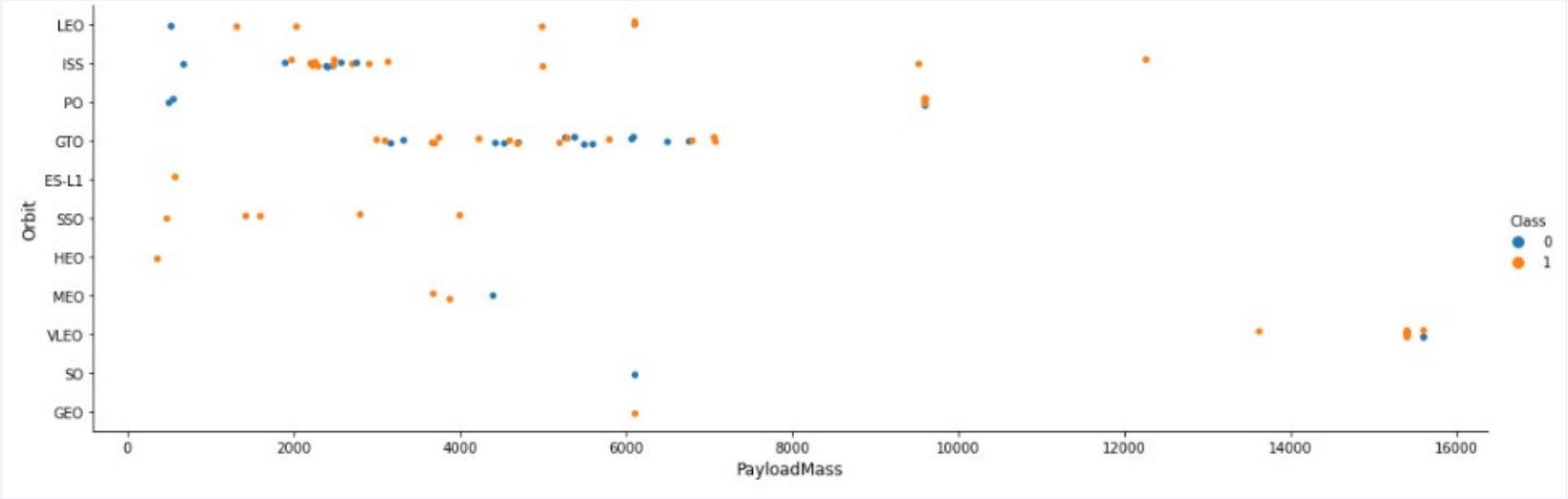
Flight Number vs. Orbit Type

- The more launches on an orbit, the most likely is we have a success
- Some orbits have few launches, so success or failure is not relevant (not enough data)



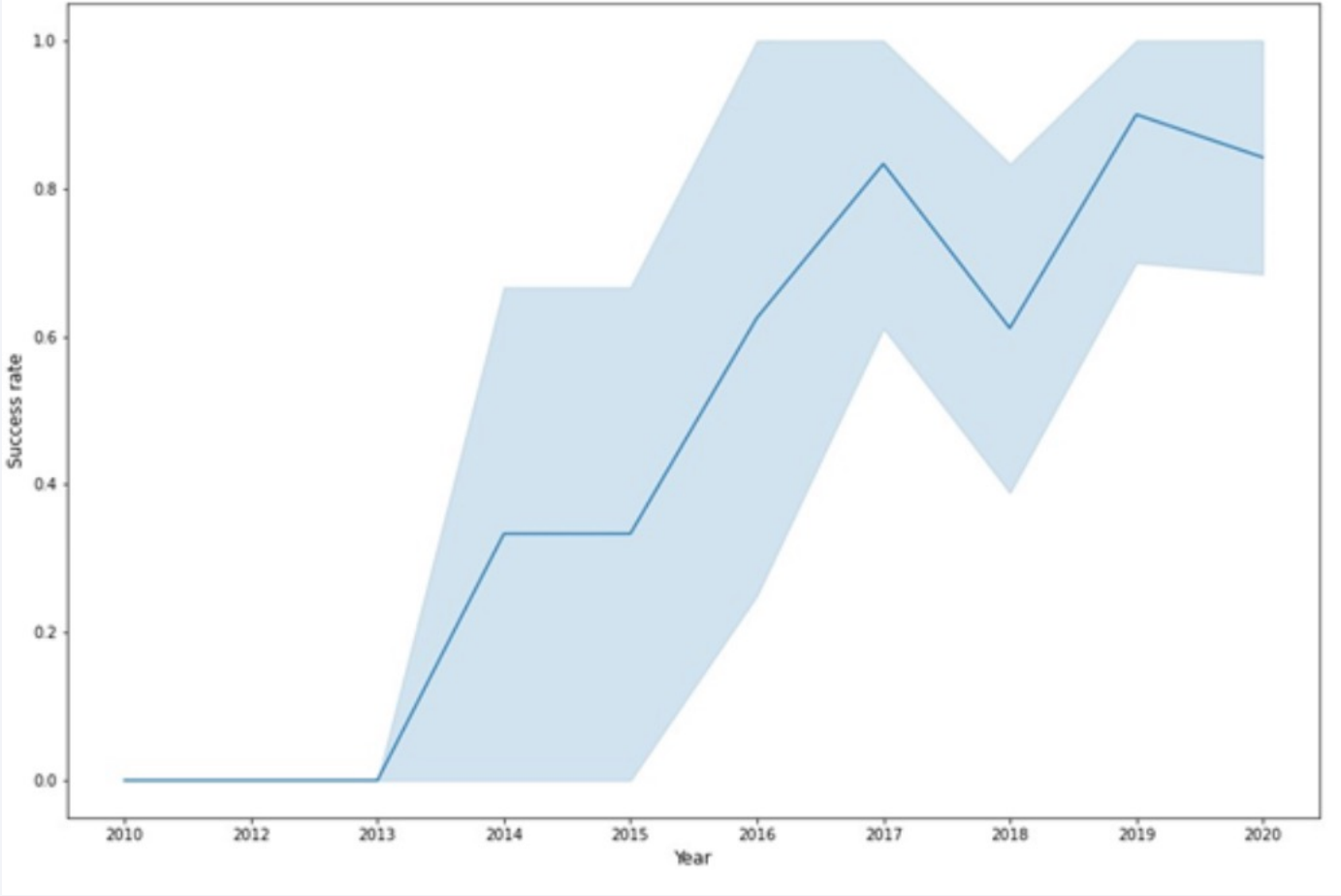
Payload vs. Orbit Type

- Most of the launches are below 8,000 Kg
- Heavy launches have a high success rate, but they are few
- GTO has a mix of success and failures across all weights



Launch Success Yearly Trend

- Success rate has, in general, increased
- There was a set-back in 2018
- Overall, success rate in 2020 is much higher



All Launch Site Names

- Find the names of the unique launch sites. We use `distinct` to ensure unique names.
- Query: %sql SELECT DISTINCT (Launch_site) FROM SPACEXTBL;
- Result:

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

Launch Site Names Begin with 'CCA'

- Find 5 records where launch sites begin with `CCA`. We use `like` to filter similar names, and `limit` to filter the output to 5.
- Query: %sql SELECT * FROM SPACEXTBL WHERE launch_site LIKE 'CCA%' LIMIT 5;
- Result:

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

Total Payload Mass

- Calculate the total payload carried by boosters from NASA. We use a `where` clause to filter by `NASA`. We use `sum` to add the `payload_mass_kg_` columns that match the `where` filter
- Query: %sql SELECT SUM(payload_mass_kg_) AS TOTAL_PAYLOAD_MASS FROM SPACEXTBL WHERE customer= 'NASA (CRS) ' ;
- Result:

total_payload_mass
45596

Average Payload Mass by F9 v1.1

- Calculate the average payload mass carried by booster version F9 v1.1. We use a `where` condition to select the booster, and `avg` to calculate the value
- Query: %sql SELECT AVG(payload _mass_kg_) AS AVG_PAYLOAD_MASS FROM SPACEXTBL WHERE booster_version= 'F9 v1.1';
- Result:

avg_payload_mass
2928

First Successful Ground Landing Date

- Find the dates of the first successful landing outcome on ground pad. We use a `where` filter and `min` to find the earliest date.
- Query: %sql SELECT MIN(DATE) AS first_successful _landing FROM SPACEXTBL WHERE landing_outcome = 'Success (ground pad)';
- Result:

first_successful_landing
2015-12-22

Successful Drone Ship Landing with Payload between 4000 and 6000

- List the names of boosters which have successfully landed on drone ship and had payload mass greater than 4000 but less than 6000. We use `between` in our `where` filter to select values in the range we want.
- Query: %sql SELECT booster_version, payload__mass_kg_, landing_outcome FROM SPACEXTBL WHERE landing_outcome='Success (drone ship)' AND (payload _mass_kg_ BETWEEN 4000 AND 6000) ;
- Result:

booster_version	payload_mass__kg_	landing_outcome
F9 FT B1022	4696	Success (drone ship)
F9 FT B1026	4600	Success (drone ship)
F9 FT B1021.2	5300	Success (drone ship)
F9 FT B1031.2	5200	Success (drone ship)

Total Number of Successful and Failure Mission Outcomes

- Calculate the total number of successful and failure mission outcomes. We use `group by` to group outcomes by its potential values, and `count` to get the number of occurrences of each value
- Query: %sql SELECT mission_outcome, COUNT (mission_outcome) AS TOTAL FROM SPACEXTBL GROUP BY mission_outcome;
- Result:

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

Boosters Carried Maximum Payload

- List the names of the booster which have carried the maximum payload mass. First we use a subquery with `max` to get the maximum payload mass. Then we use a `where` filter to select entries that match that value.
- Query: %sql SELECT booster_version, payload_mass__kg_ FROM SPACEXDATASET WHERE payload_mass__kg_ = (SELECT max(payload_mass__kg_) FROM SPACEXDATASET);
- Result:

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

2015 Launch Records

- List the failed landing_outcomes in drone ship, their booster versions, and launch site names for in year 2015. We use a `like` filter to get the outcome, and the `year` function to filter dates by the right year.
- Query: %sql SELECT landing_outcome, booster_version, launch_site, DATE FROM SPACEXTBL WHERE landing_outcome LIKE '%Failure (drone ship)%' AND YEAR(DATE) = 2015;
- Result:

landing_outcome	booster_version	launch_site	DATE
Failure (drone ship)	F9 v1.1 B1012	CCAFS LC-40	2015-01-10
Failure (drone ship)	F9 v1.1 B1015	CCAFS LC-40	2015-04-14

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

- 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. We use `group by` and `count` to get the count. We use `Order by` to get the right ordering.
- Query: %sql SELECT landing_outcome, COUNT(landing_outcome) AS "total" FROM SPACEXTBL WHERE (DATE BETWEEN ' 2010-06-04' AND '2017-03-20') GROUP BY landing_outcome ORDER BY COUNT(landing_outcome) DESC
- Result:

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

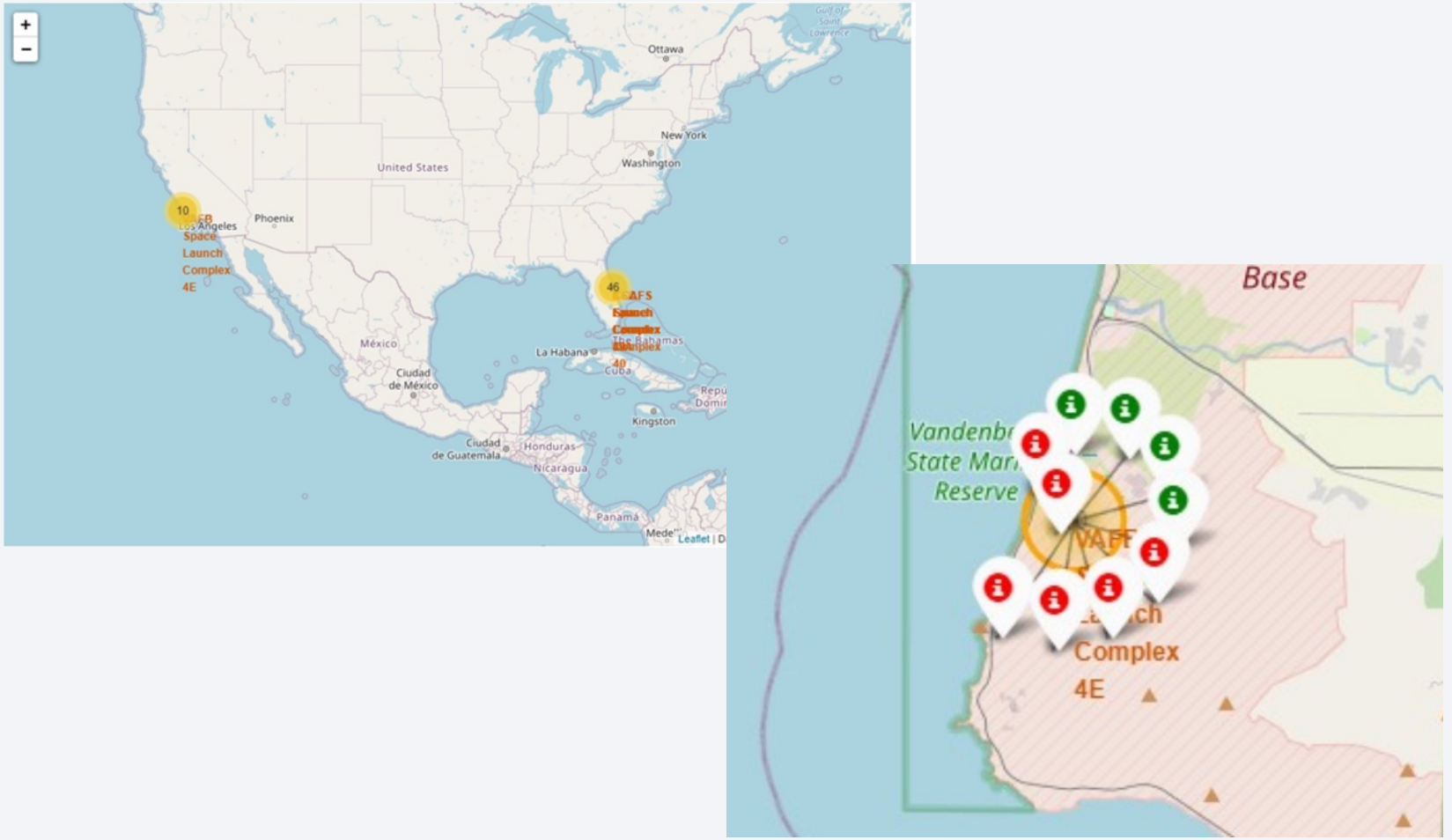

Section 3
Launch Sites
Proximities Analysis

All Launch Sites

- All launch sites are near the coast
- KSC and CCAFS are close to each other and overlap on this view

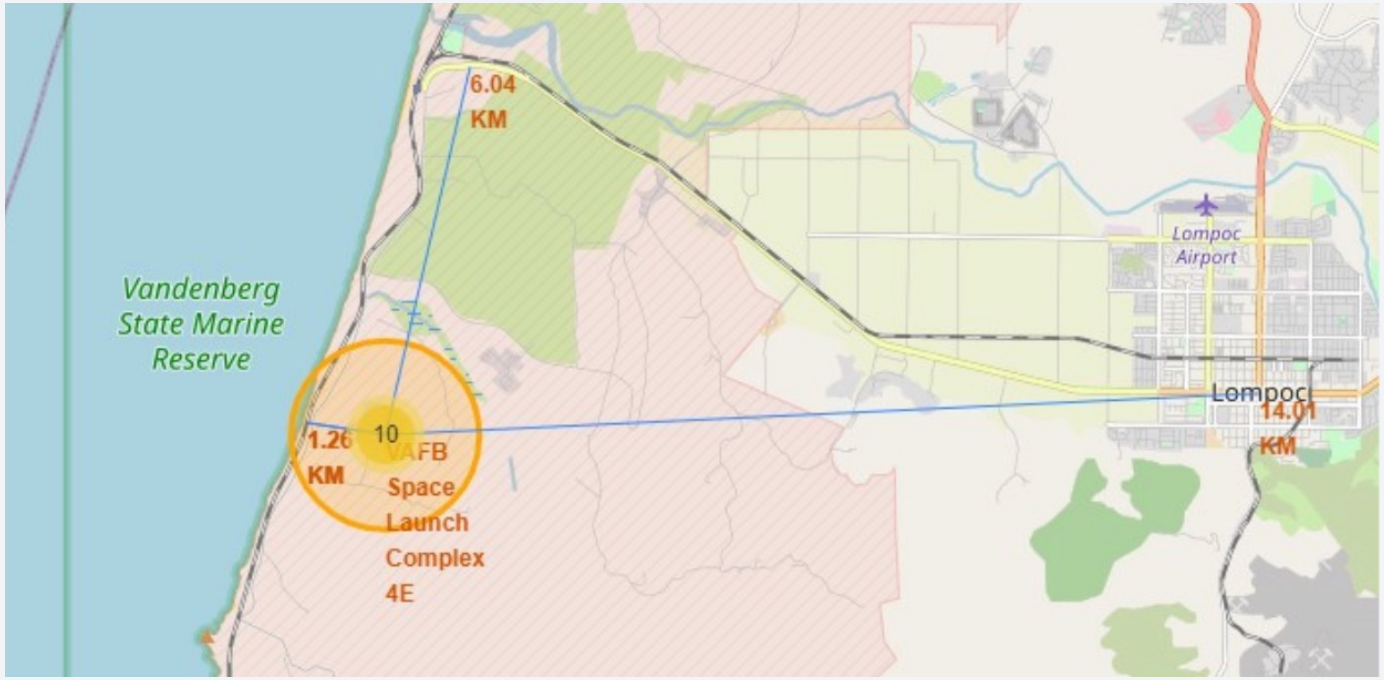
Success and Failures on Each Site

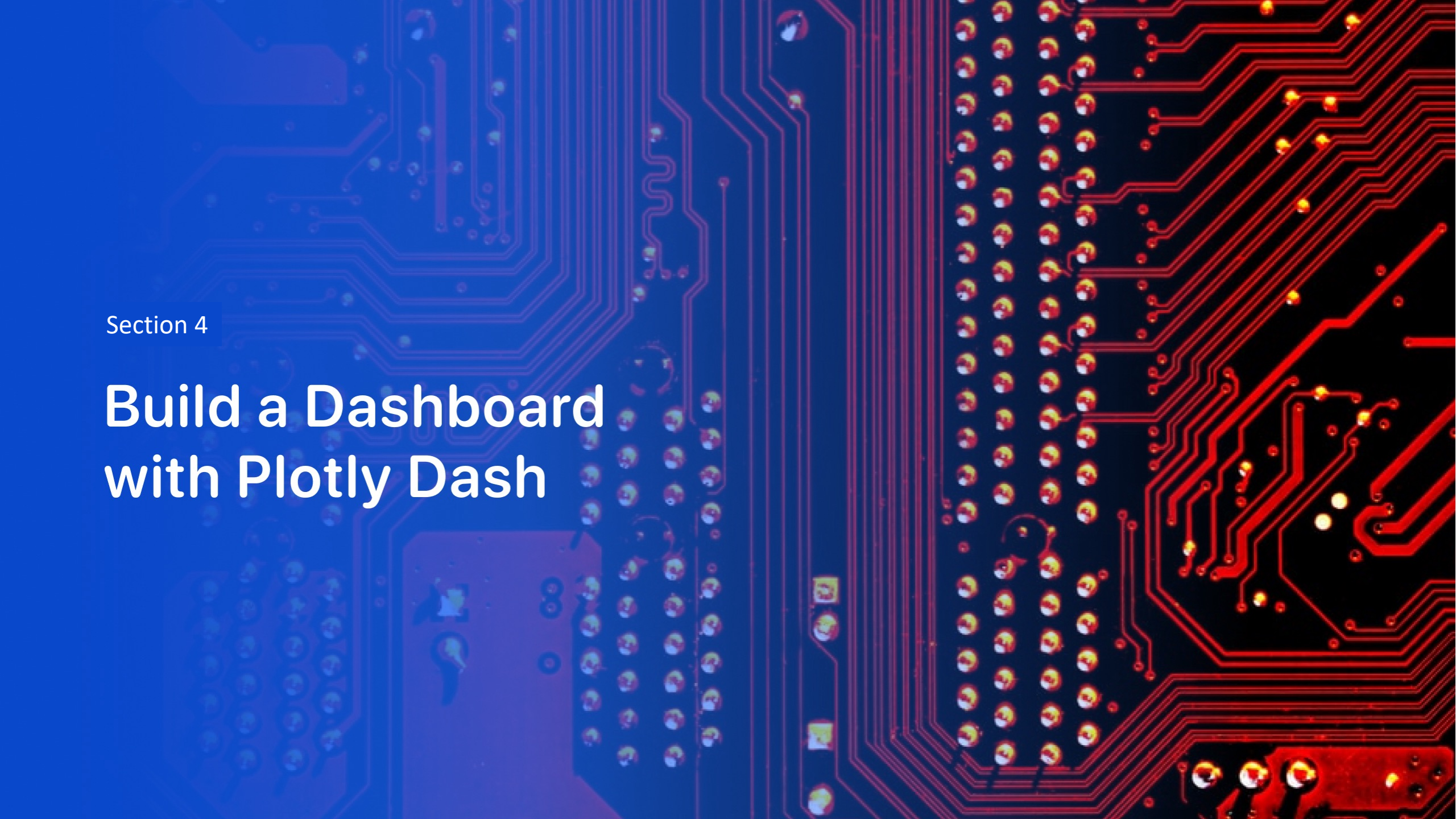
- The view clusters launches, showing a numeric count of the number of launches per location
- On zoom, we can see individual markers for success (green) or failure (red)



Distance to Proximities

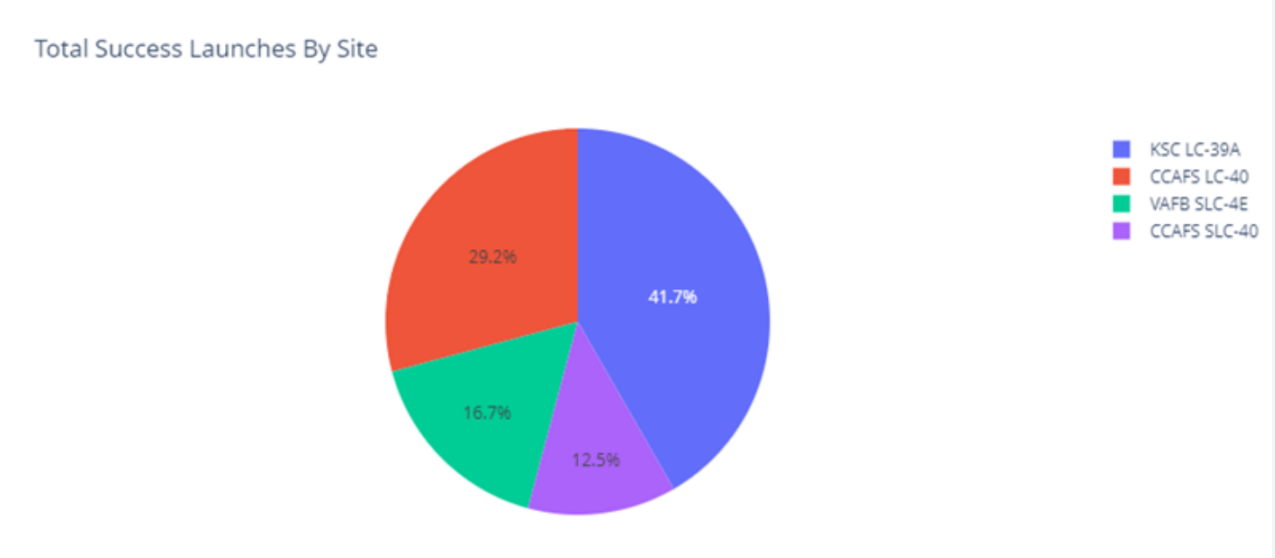
- Launch sites are in isolated areas
- VAFB is 1.26 Km from the coastline
- It is 6.04 Km from a rail station
- It is 14.01 Km from a city




Section 4
Build a Dashboard with Plotly Dash

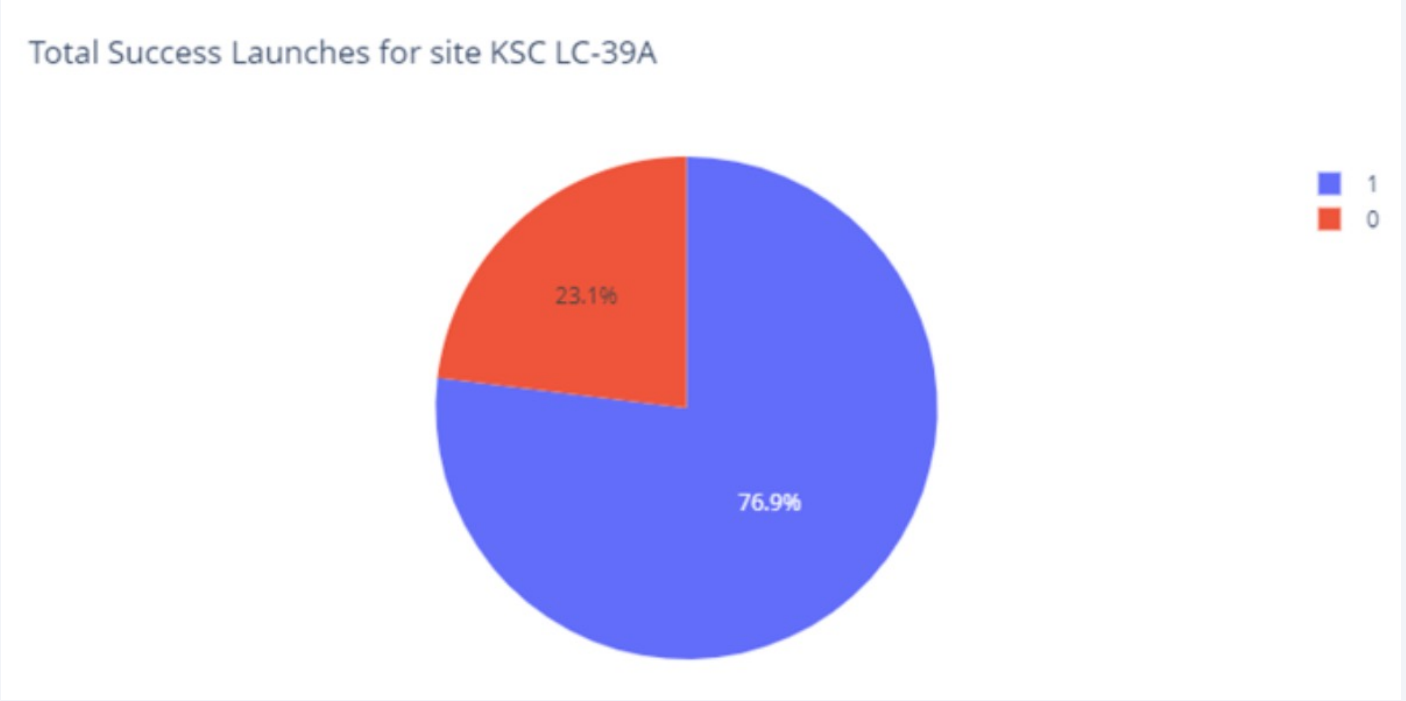
Total Success by Launch Site

- The pie chart shows all successful launches on all sites
- Most successful site is KSC LC-39A



KSC LC-39A

- The pie chart show success and failures for KSC LC-39A
- The success rate is 76.9%



Payload vs Launch Outcome

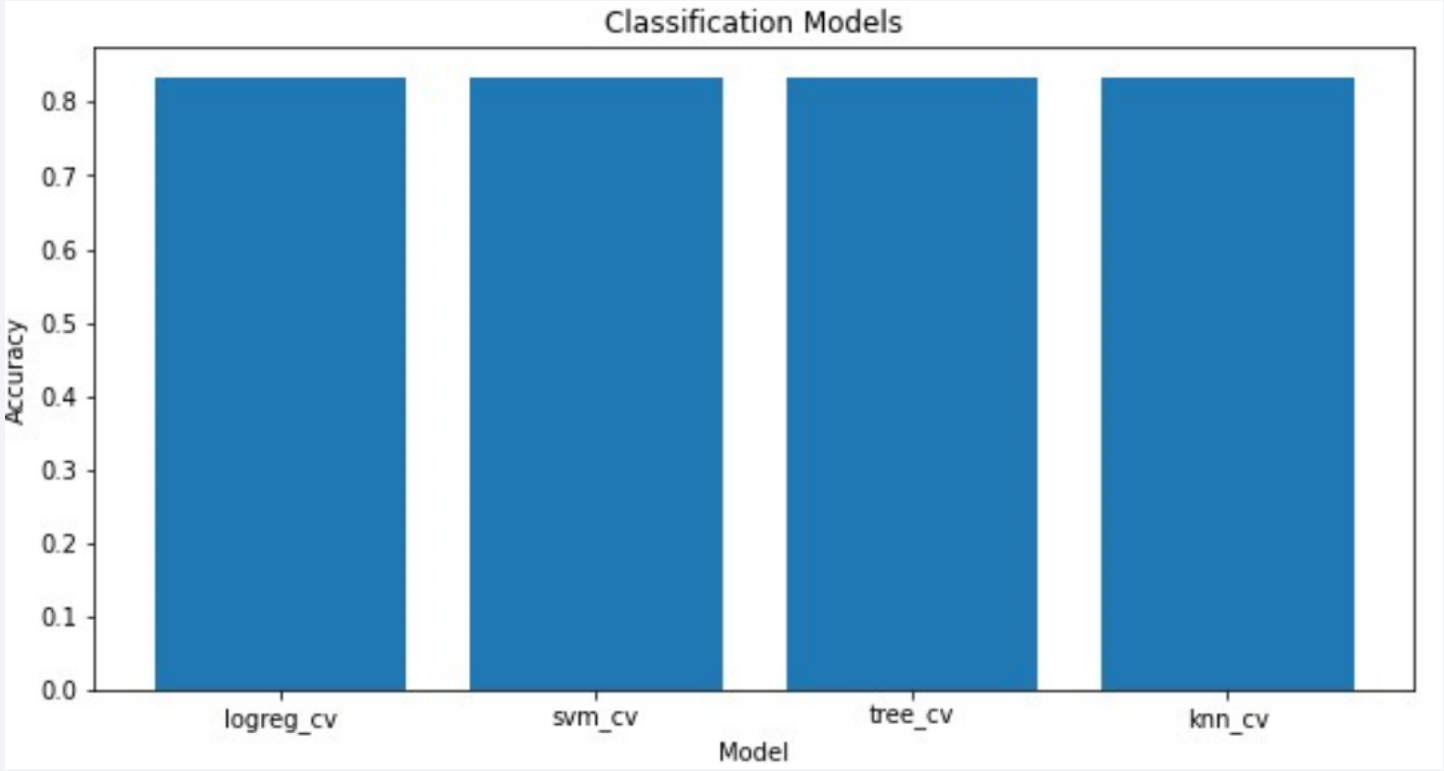
- The selected ranges are 0-2500, 2500-5000, and >5000

Section 5

Predictive Analysis (Classification)

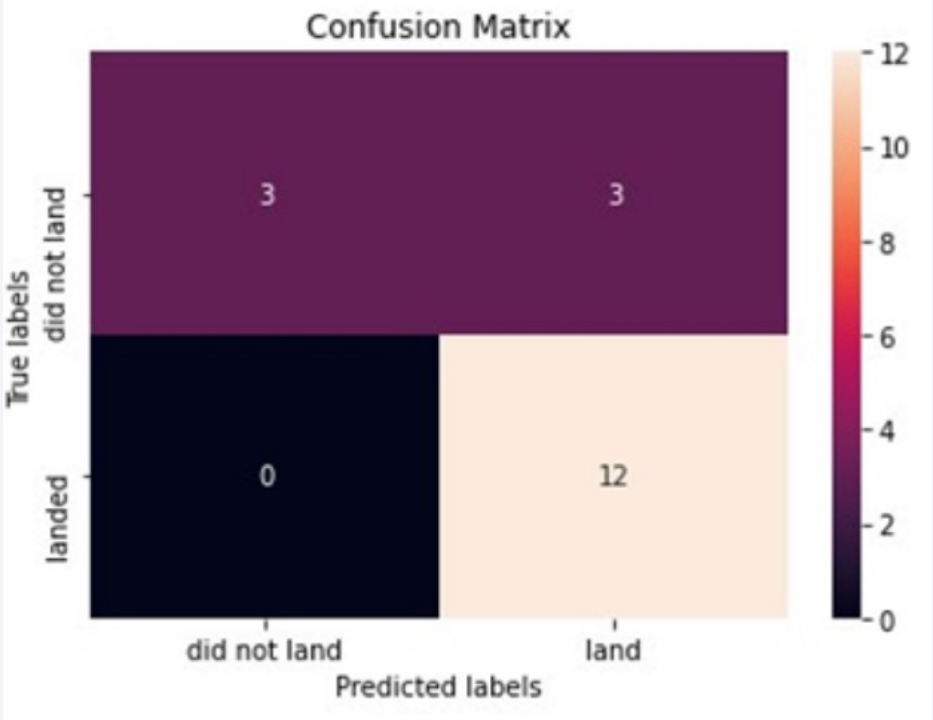
Classification Accuracy

- All models had comparable accuracy



Confusion Matrix

- Confusion matrix for Logistic Regression model
- It shows 3 false positives (top right) and 0 false negatives (bottom left)



Conclusions

- Landing outcomes have improved over time
- Machine learning models can predict the outcome, although not perfectly
- Some combinations of location, orbit, and payload size are performing much better than others

Appendix

- Links to notebooks have been included in the relevant slides

Thank you!