IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Frank Kruger
12 Oct 2025

Outline

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

Executive Summary

- Methodologies
  - Data was collected from SpaceX's REST API and Wikipedia, focusing on the Falcon 9
  - The data was wrangled and feature engineering applied
  - Checked correlation amongst features such as orbit, payload mass, launch location and flight number (function of time)
  - Applied visualization techniques including scatter plots, maps, and a dashboard for dynamic interation
- Results
  - Factors that supported success included higher orbits, heavier payloads, launching from CCAFS LC-40.
  - The Support Vector Machine was the best performer to predict if the first stage will land successfully or not, although false positives could appear.

Introduction

- Space Y, founded by Allon Musk, wants to compete with Space X in the launch of payloads into orbit.
- The key question to answer is ‘what is the cost of each launch?’
- A key component of launch cost is the the cost of the first stage, which does most of the work. Space X has been able to keep costs down by having the first stage land back on Earth after use without crashing. That means a launch can cost 65 million dollars with successful landings rather that 165 million dollars if the first stage cannot be not reused.
- We will use machine learning and available Space X launch data to predict if Space X will reuse the first stage.

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - The data from Space X's launches was obtained from Space X's REST API and from the Falcon 9 Wikipedia web page
- Perform data wrangling
  - Launch data was checked for missing values. Labels were generated, and categorical features were one-hot encoded.
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - 4 classification models were applied, test and evaluated.

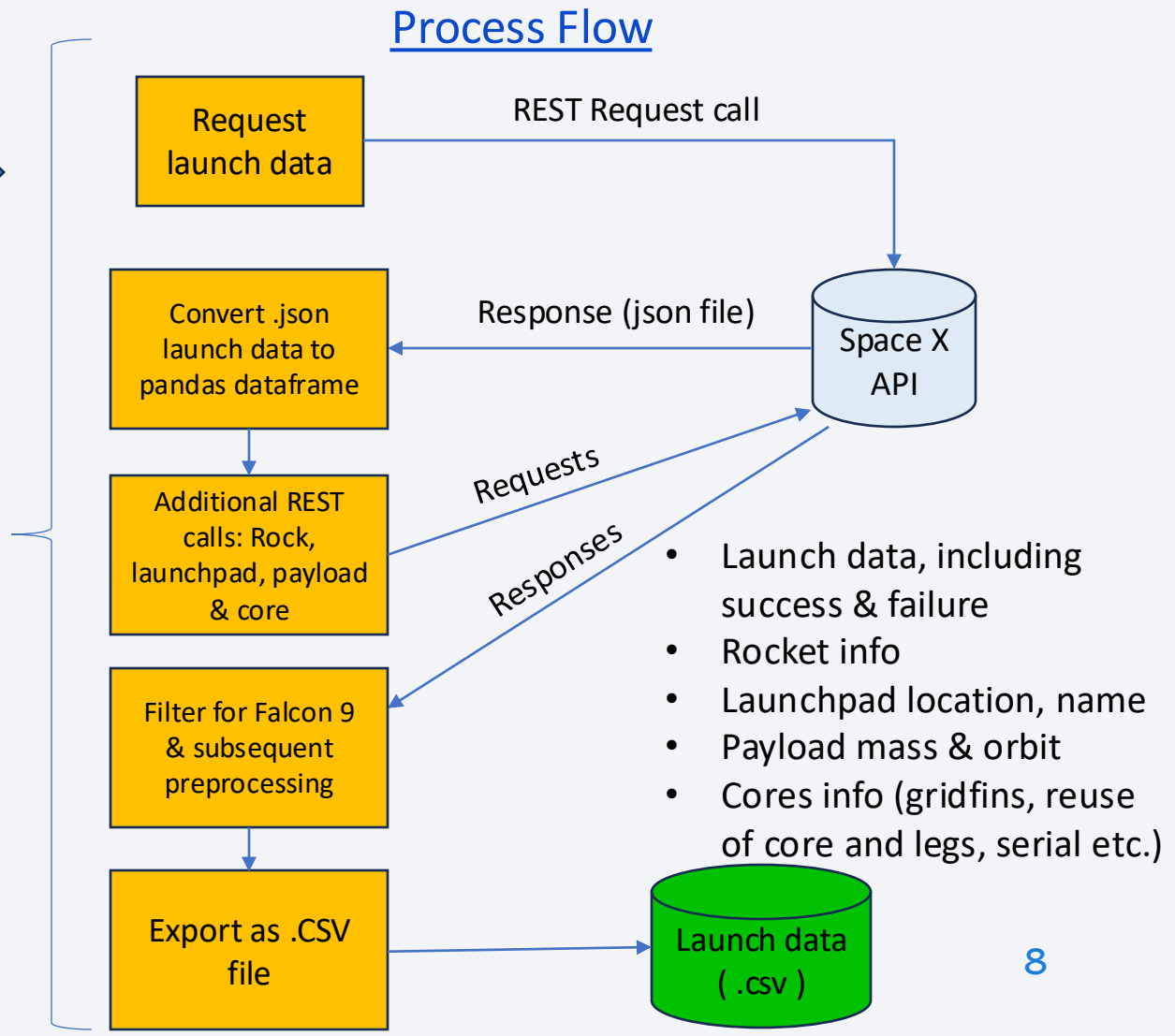
Data Collection

- For this step, two methods were used:
    1. Utilize the SpaceX REST API for the launch data
    2. Compile launch data from the Falcon 9 Wikipedia page
- The output of this process is then preprocessed further for analysis purposes.

Data Collection – SpaceX API

- SpaceX launch data was obtained from the SpaceX REST API
- Notebook at this URL executes the HTTP requests and response handling:

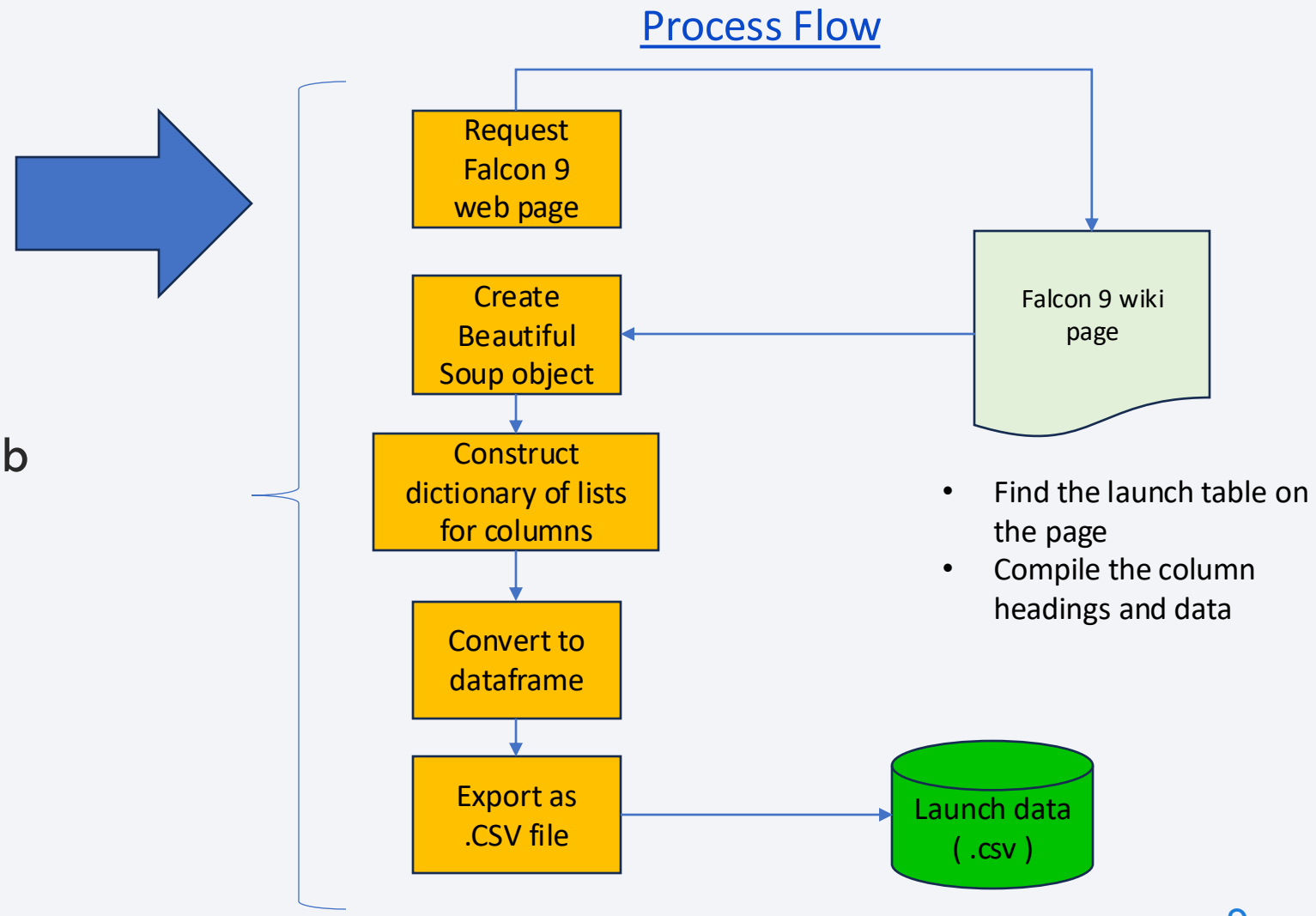
https://github.com/fkruger-tech/data-science-capstone/blob/main/jupyter-labs-spacex-data-collection-api.ipynb



Data Collection - Scraping

- In this step, we use the Wikipedia page for the Falcon 9 rocket to compile the launch data
- The notebook to carry out this process is at this Github URL:

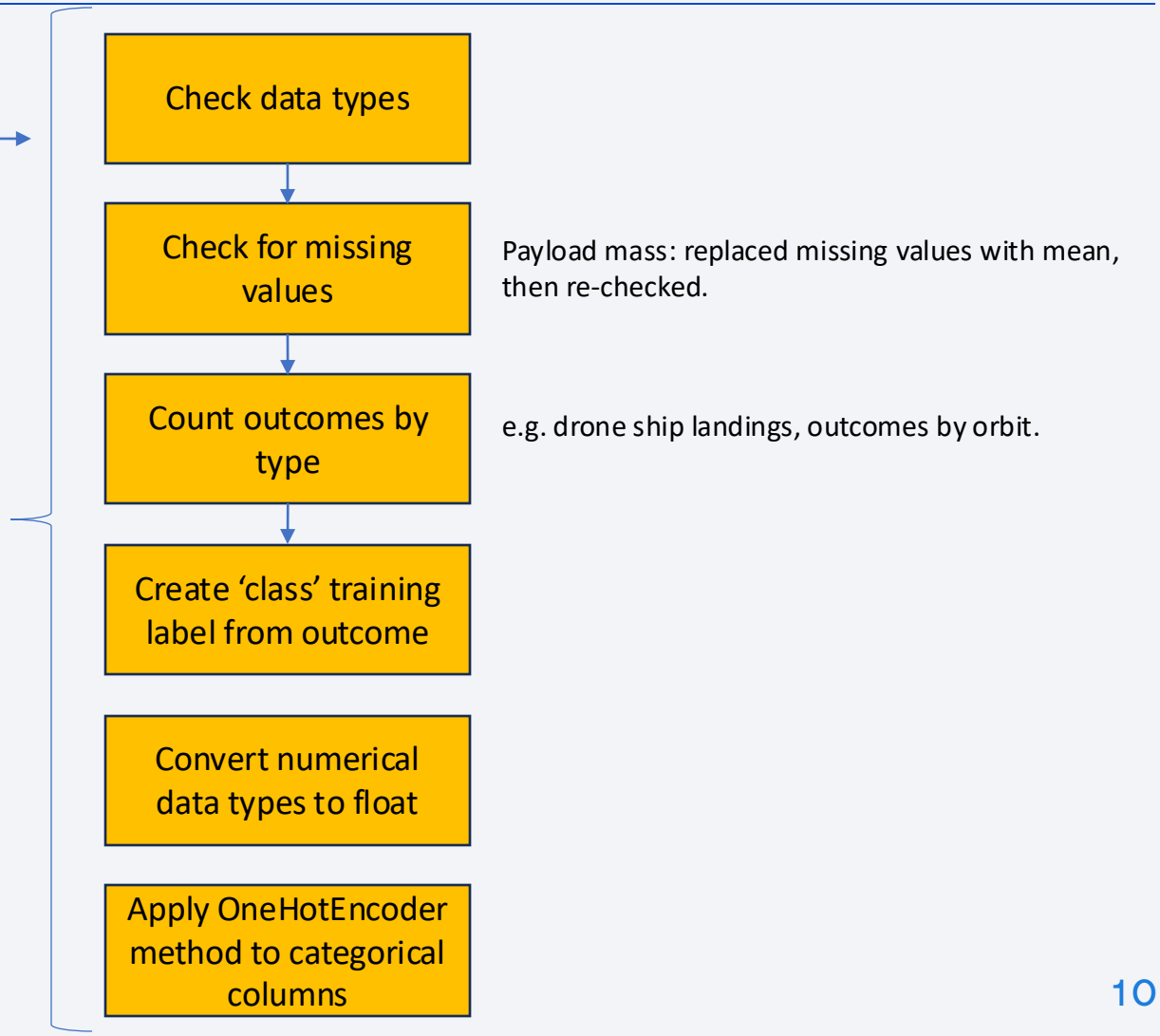
https://github.com/fkruger-tech/data-science-capstone/blob/main/jupyter-labs-webscraping.ipynb



Data Wrangling

- Data wrangling process
- URL for notebook that deals with data wrangling step is:

https://github.com/fkruger-tech/data-science-capstone/blob/main/labs-jupyter-spacex-Data%20wrangling.ipynb



EDA with Data Visualization

This step's purpose is to get a better understanding of the data that has been collected through visualization. The charts listed here were used for this purpose.

Chart type	Chart	Purpose
scatter	Flight Number vs PayloadMass	To see how success changes as time progresses
scatter-category	Flight Number vs Launch Site	To see how success varies across launch sites over time.
Scatter	Launch site vs payload mass	Visualize relationship between payload mass and site
bar	Success rate by Orbit	To visualize the success rate for different orbits
scatter	Flight Number vs Orbit	To see how success varies over time and orbit
scatter	Pay load mass vs Orbit	To see relationship between payload and orbit type
line	Success by year	To see how success rate changed over time

- Notebook with these charts is at this URL:

https://github.com/fkruger-tech/data-science-capstone/blob/main/edadataviz.ipynb

EDA with SQL

The SQL queries used to better understand the data collected are:

- List the launch sites from the SPACEX table
- List the first 5 launch sites that start with 'CCA'
- Display the total mass carried by boosters launched by NASA (CRS)
- Display average payload mass carried by booster F9 v1.1
- List the date the first successful ground pad landing was achieved
- List the booster versions that had successful drone ship landings with payload between 4000 and 6000 kg.
- Summarize the # successful and failure mission outcomes
- List the boosters that carried the maximum payload mass
- List the drone ship failures with month for 2015 with booster version and launch site
- Rank the landing outcomes by number of occurrences between 2010-06-04 and 2017-03-20
- The Notebook used for the SQL statements is at this URL:

https://github.com/fkruger-tech/data-science-capstone/blob/main/jupyter-labs-eda-sql-coursera_sqllite.ipynb

Build an Interactive Map with Folium

Objects were added to a folium map:

- Circles to indicate the launch sites
- Marker Clusters were added to indicate success and failures. Clusters were used since many of the data points had the same location and seeing the volumes at each common location was necessary.
- Mouse Position was added so that we can dynamically see the coordinates as we explore the map.
- Icons and lines were added to support calculating distances

The notebook to visualize the location of the launch sites is at URL:

https://github.com/fkruger-tech/data-science-capstone/blob/main/lab_jupyter_launch_site_location.ipynb

Build a Dashboard with Plotly Dash

The dashboard consists of the following:

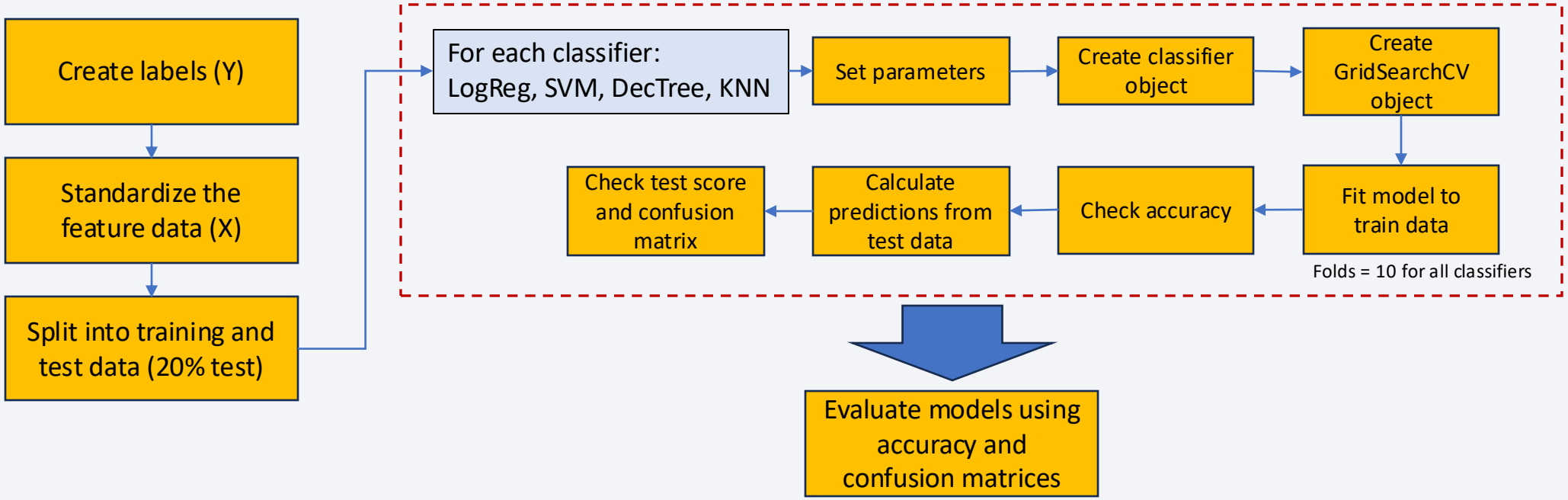
- A pie chart that describes the successes for all launch sites, or the success rates for each launch site
- A scatter plot of success or failure by payload mass, based on which launch site is selected (or all of them)

The idea behind this dashboard is to dynamically explore success patterns based on launch sites and payload mass.

The dashboard app is at this URL:

https://github.com/fkruger-tech/data-science-capstone/blob/main/spacex-dash-app.py

Predictive Analysis (Classification) - Process

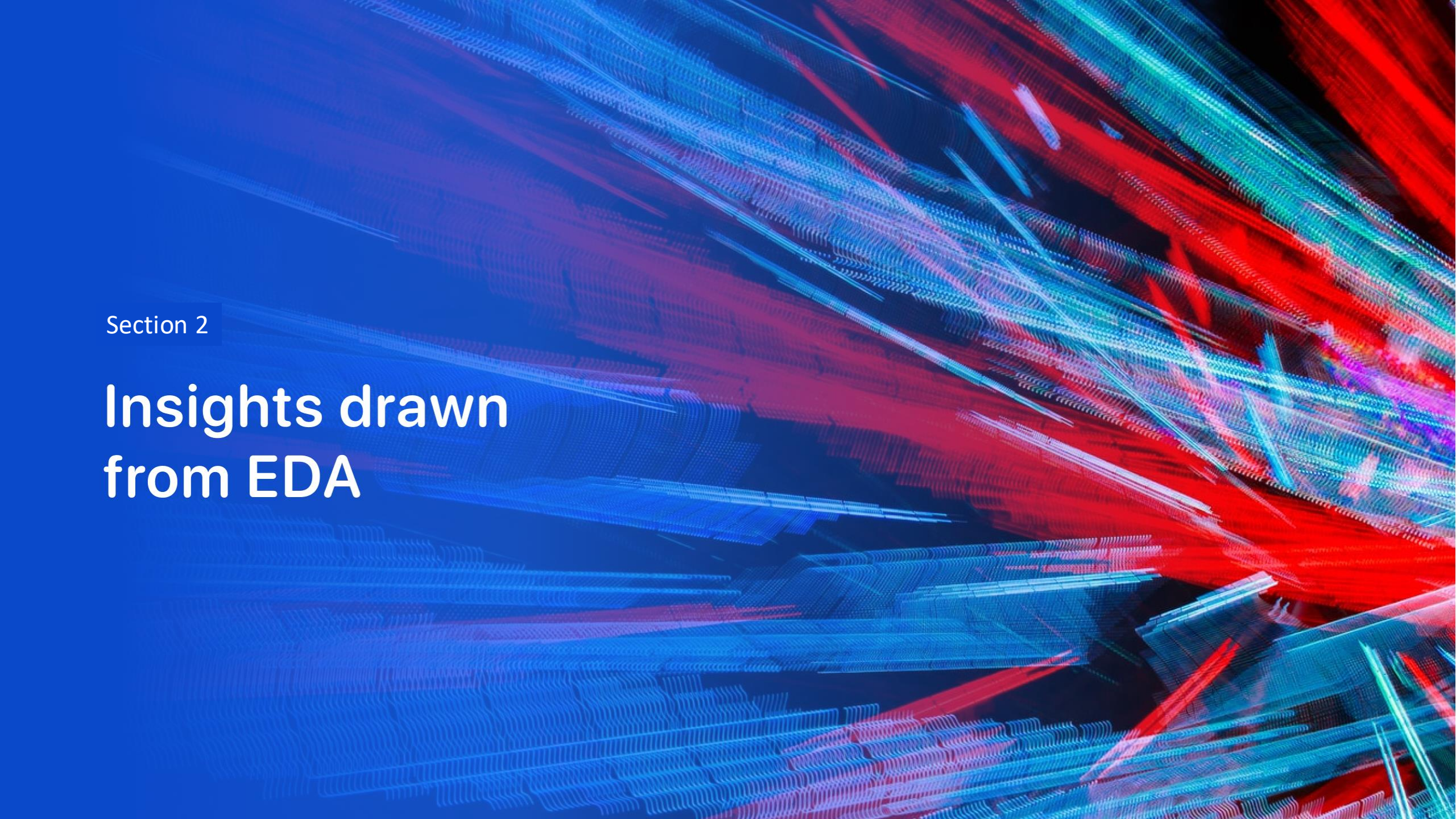


- URL for the notebook that walks through this process is:

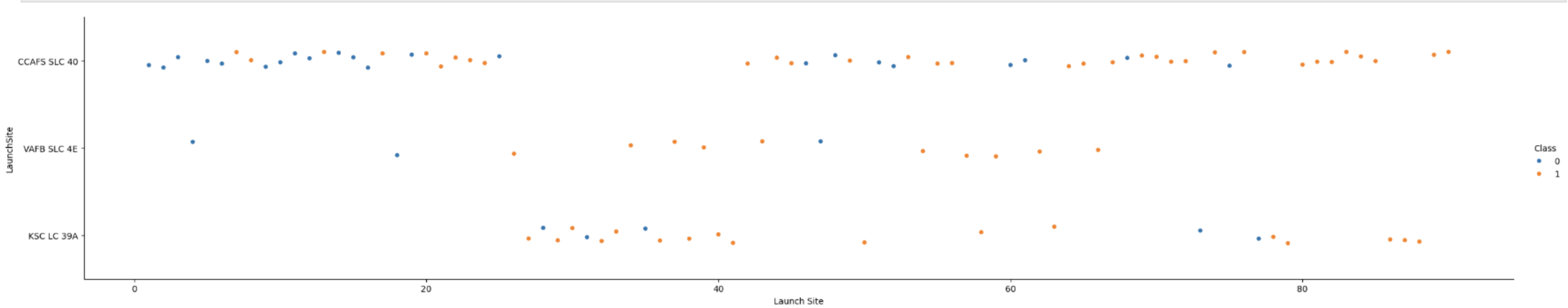
https://github.com/fkruger-tech/data-science-capstone/blob/main/SpaceX_Machine%20Learning%20Prediction_Part_5.ipynb

Results

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


Section 2
Insights drawn from EDA

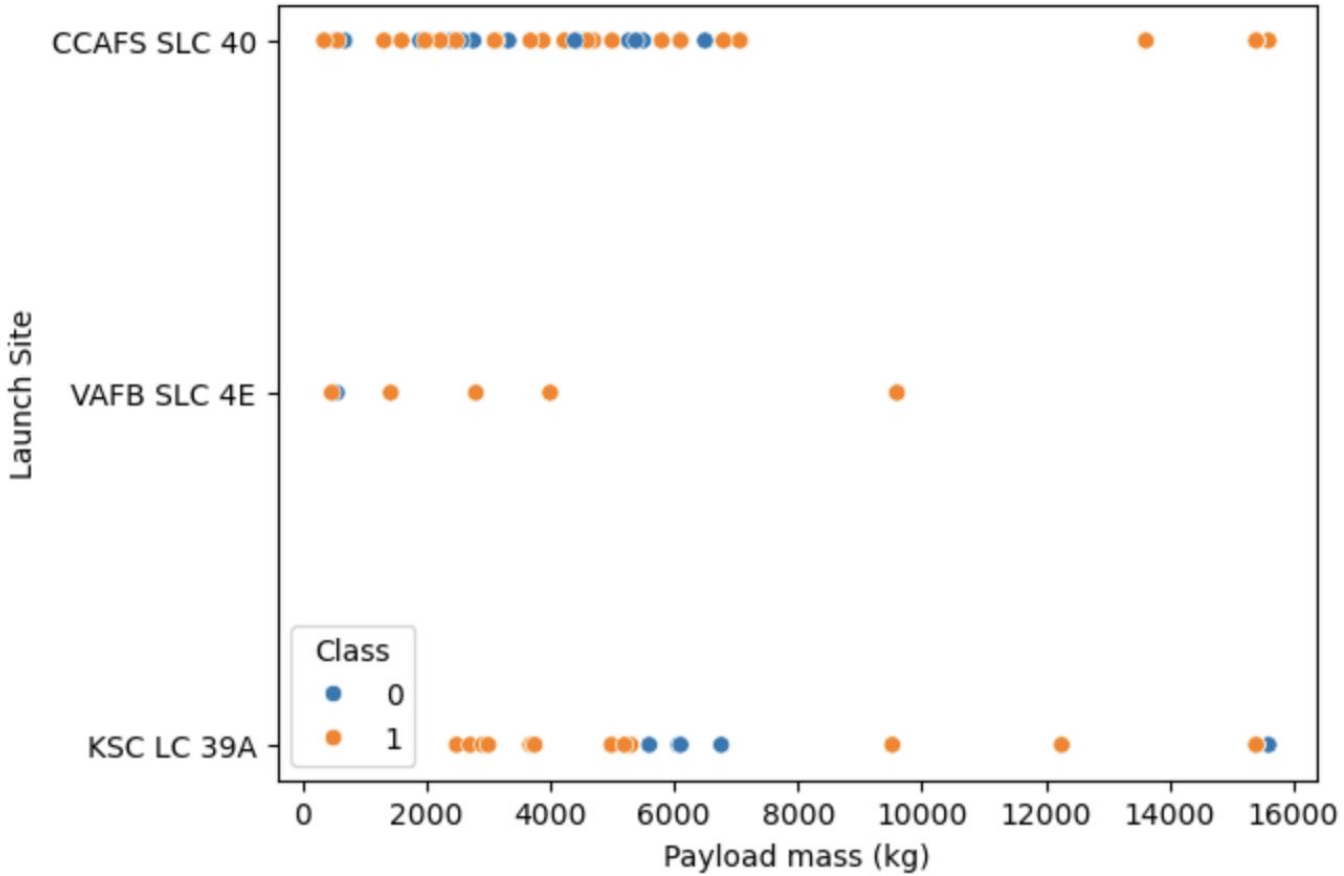
Flight Number vs. Launch Site



Now try to explain the patterns you found in the Flight Number vs. Launch Site scatter point plots.

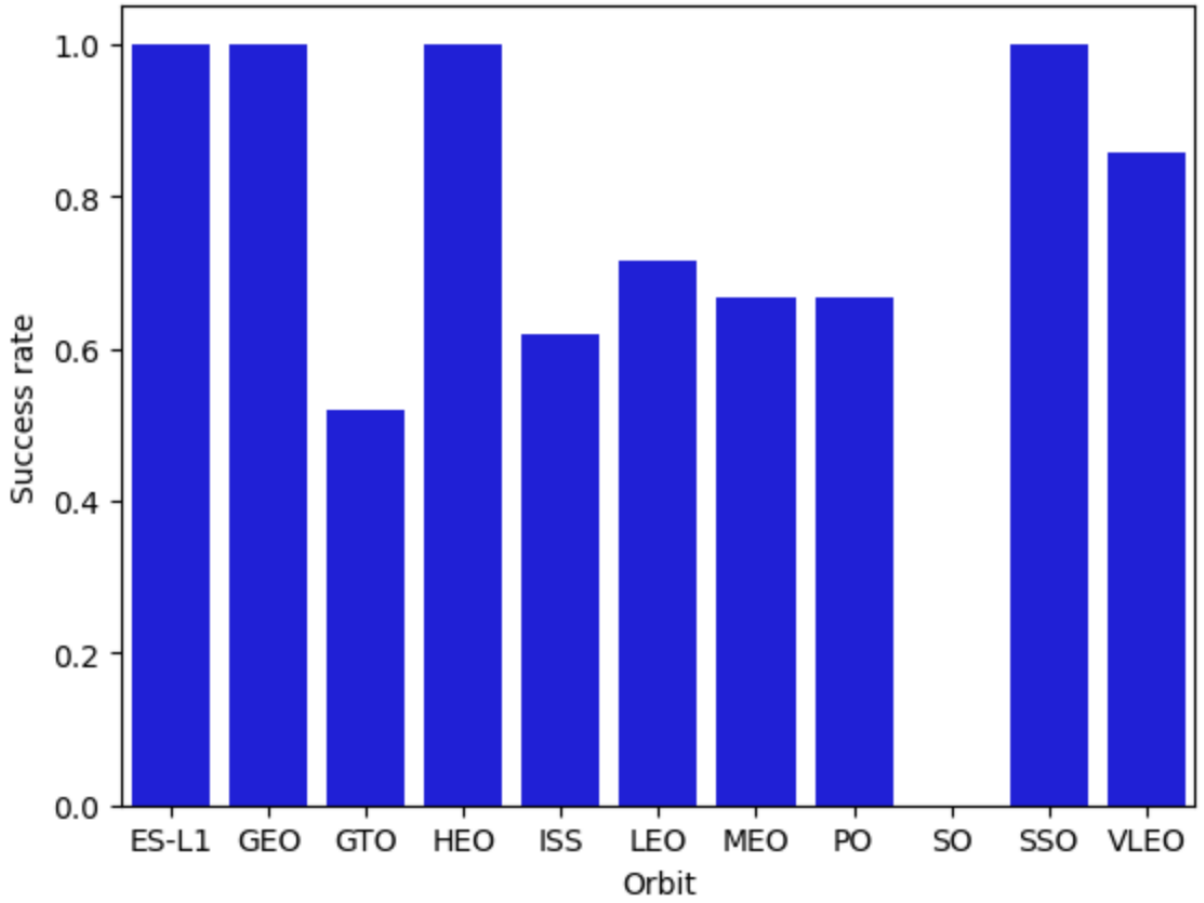
Many of the earlier failures were from the CCAFS SLC 40 launch site. Launches from KSC LC 39A had successes right from its start. There are not many data points for VAFB SLC 4E to interpret, but early successes from this site shifted to consistent failures later.

Payload vs. Launch Site



Now if you observe Payload Mass Vs. Launch Site scatter point chart you will find for the VAFB-SLC launchsite there are no rockets launched for heavypayload mass(greater than 10000).

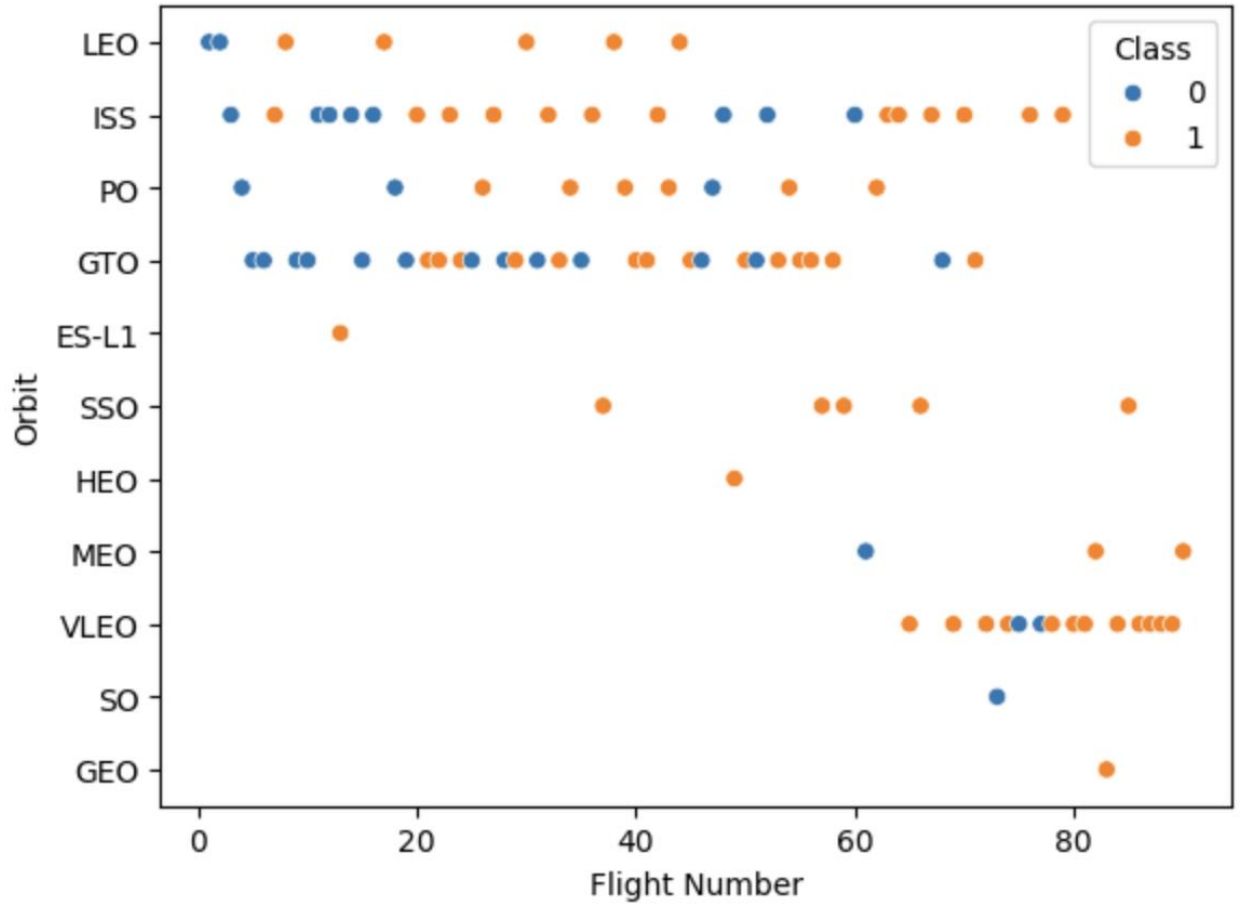
Success Rate vs. Orbit Type



Analyze the plotted bar chart to identify which orbits have the highest success rates.

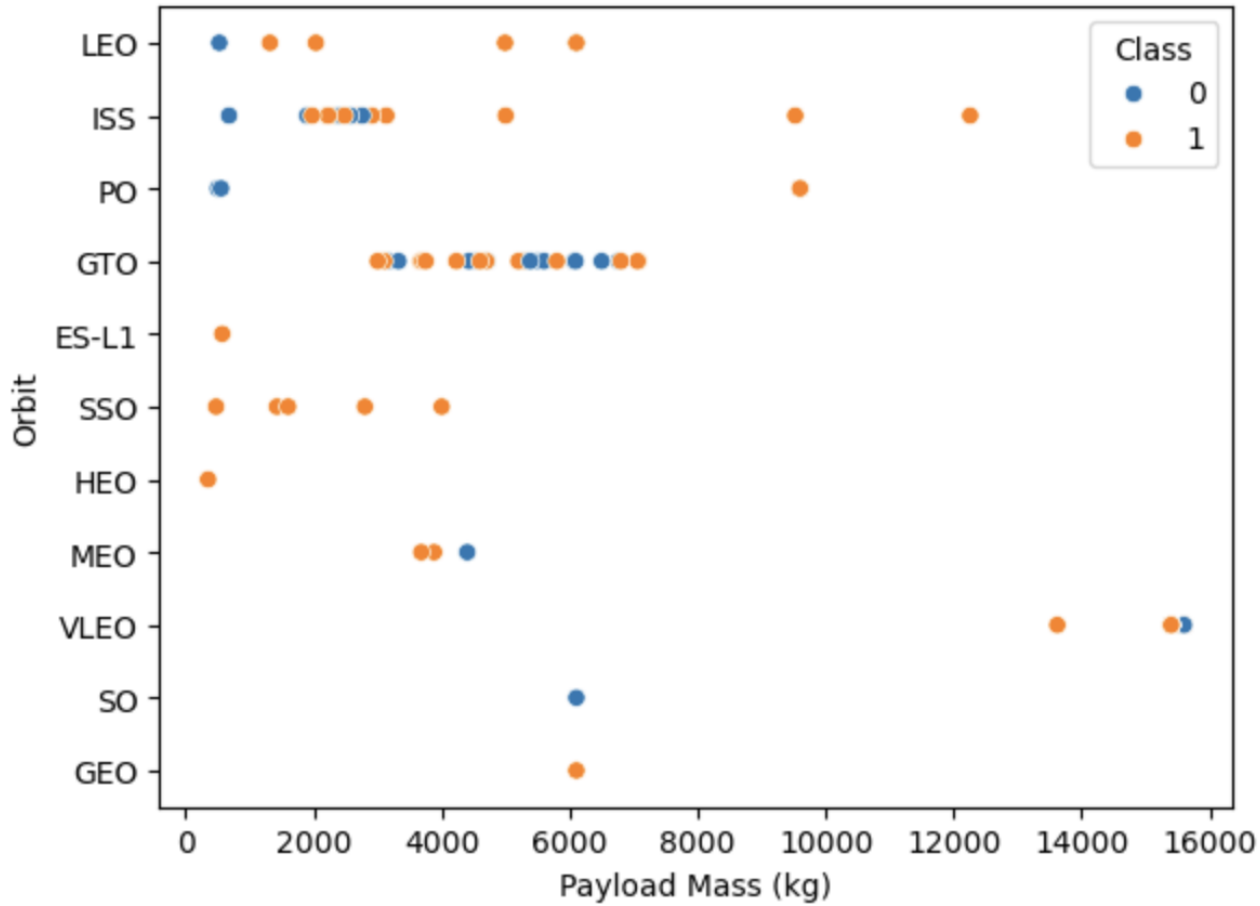
ES-L1, GEO, HEO and SSO have perfect success rates. GTO, ISS, LEO, MEO and PO have poorer success rates, less than 0.7.

Flight Number vs. Orbit Type



You can observe that in the LEO orbit, success seems to be related to the number of flights. Conversely, in the GTO orbit, there appears to be no relationship between flight number and success.

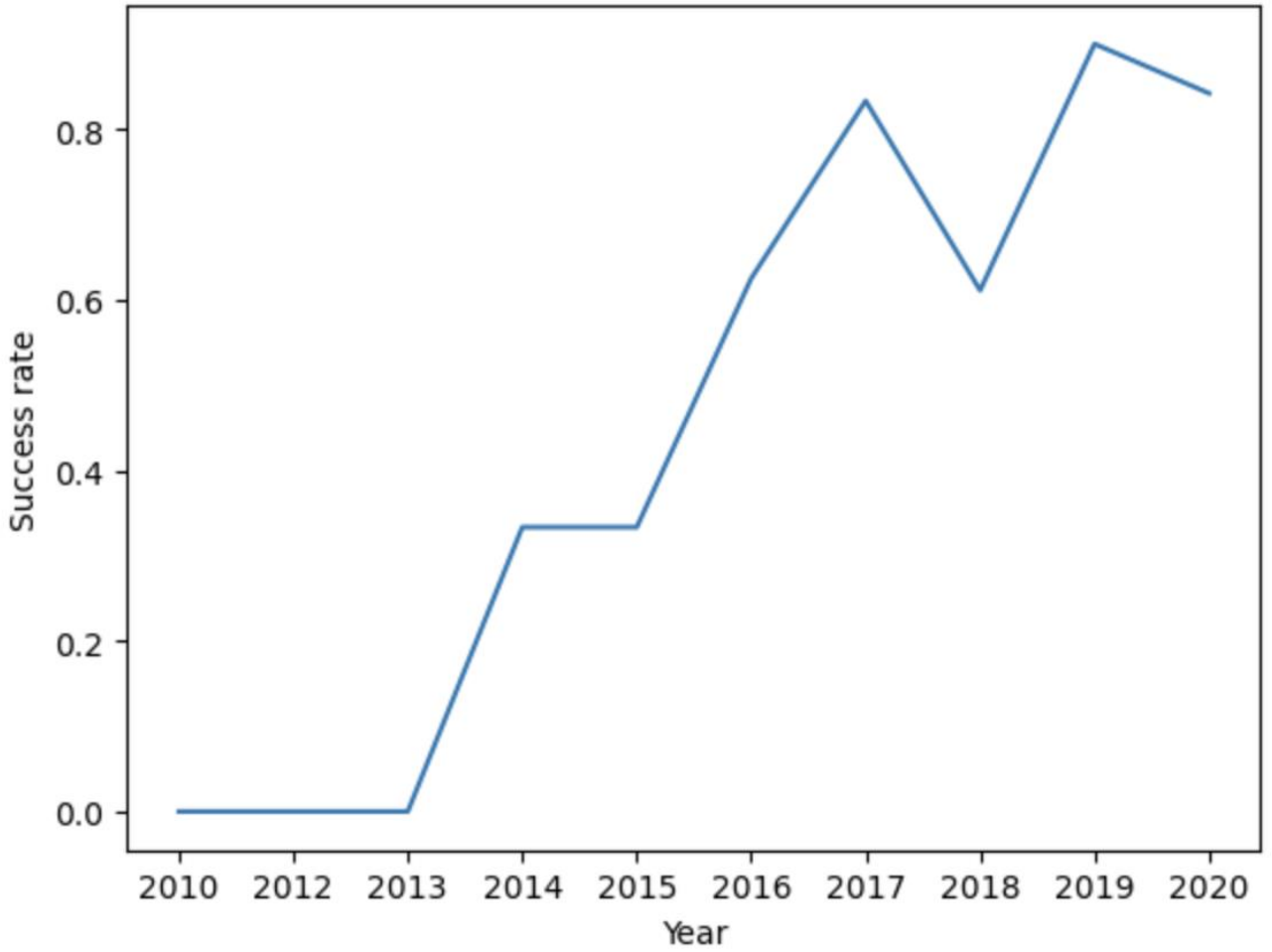
Payload vs. Orbit Type



With heavy payloads the successful landing or positive landing rate are more for Polar,LEO and ISS.

However, for GTO, it's difficult to distinguish between successful and unsuccessful landings as both outcomes are present.

Launch Success Yearly Trend



you can observe that the sucess rate since 2013 kept increasing till 2020

All Launch Site Names

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

These are the launch sites, by obtaining the unique launch site names from the SpaceX table.

Launch Site Names Begin with 'CCA'

- This is the first 5 of records with launch site names that start with CCA, using like 'CCA%' in the SQL statement.

Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Orbit	Customer	Mission_Outcome	Landing_
2010-06-04	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success	Failure (p
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 (p
2012-05-22	7:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success	N
2012-10-08	0:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500	LEO (ISS)	NASA (CRS)	Success	N
2013-03-01	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677	LEO (ISS)	NASA (CRS)	Success	N

Total Payload Mass

- This total payload mass for customer NASA (CRS) is calculated by sum grouped by customer in the SpaceX table, picking only NASA (CRS)

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

Average Payload Mass by F9 v1.1

- This is the average payload mass carried by booster version F9 v1.1
- Where clause for Booster Version used in SQL statement here.

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

First Successful Ground Landing Date

- The earliest date for a successful landing outcome on a ground pad was selected using the MIN SQL function.

min("Date")
2015-12-22

Successful Drone Ship Landing with Payload between 4000 and 6000

- The names of boosters which have successfully landed on drone ship and had payload mass greater than 4000 but less than 6000

- Used the distinct clause to list each booster version only once.

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

Total Number of Successful and Failure Mission Outcomes

- The total number of successful and failure mission outcomes
- This is a count of date, grouped by mission outcomes from the SpaceX table

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

Boosters Carried Maximum Payload

- The names of the booster which have carried the maximum payload mass; turned out to be 15,600kg

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 max payload is calculated first; then boosters which have this value are listed.

2015 Launch Records

- The failed landing_outcomes in drone ship, their booster versions, and launch site names for in year 2015
- The year and month are calculated from the date column using substrings and a case statement.

year	month	Landing_Outcome	Booster_Version	Launch_Site
2015	Jan	Failure (drone ship)	F9 v1.1 B1012	CCAFS LC-40
2015	Apr	Failure (drone ship)	F9 v1.1 B1015	CCAFS LC-40

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

- Ranked 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

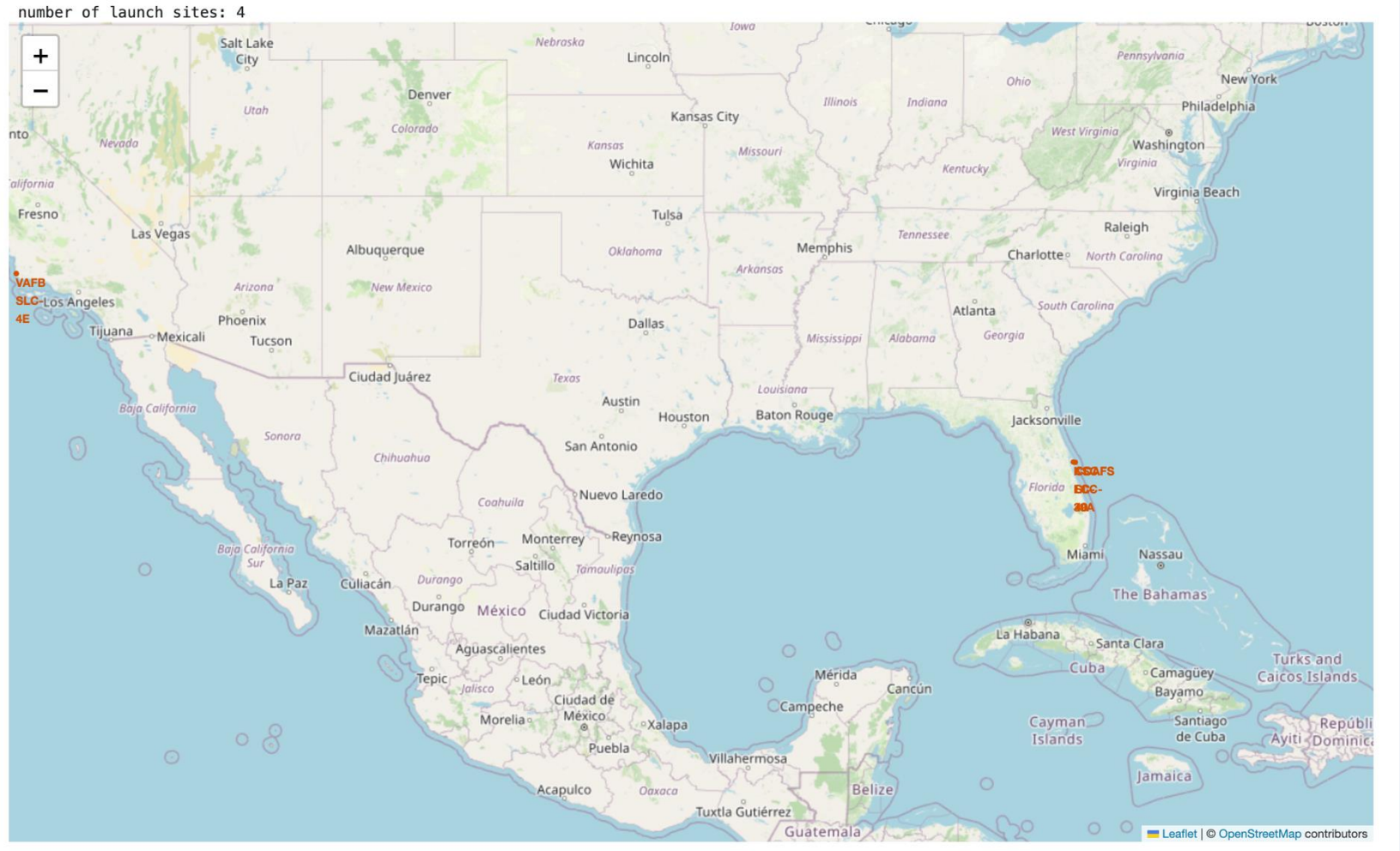
- A subquery was used to calculate the number of instances grouped by landing outcome.

Landing_Outcome	no_landing_outcomes
Success	38
No attempt	21
Success (drone ship)	14
Success (ground pad)	9
Controlled (ocean)	5
Failure (drone ship)	5
Failure	3
Failure (parachute)	2
Uncontrolled (ocean)	2
No attempt	1
Precluded (drone ship)	1

Section 3

Launch Sites Proximities Analysis

All Launch Sites Marked on a Folium Map

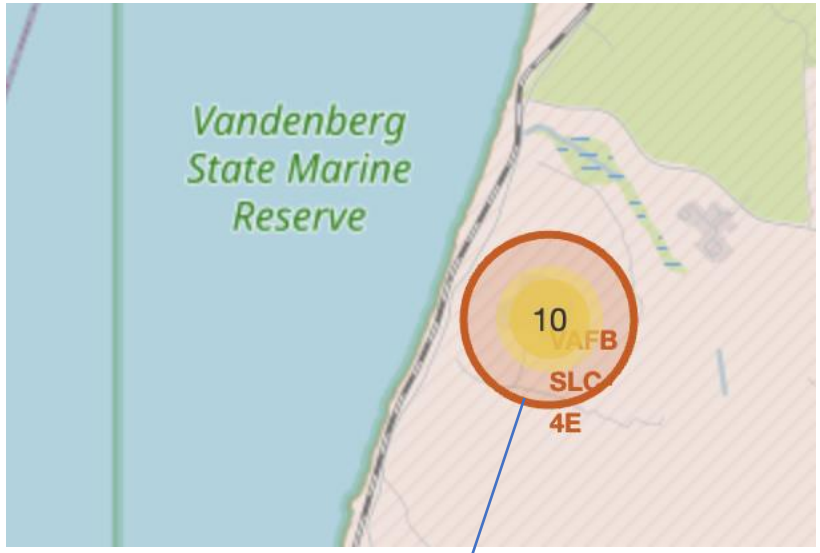


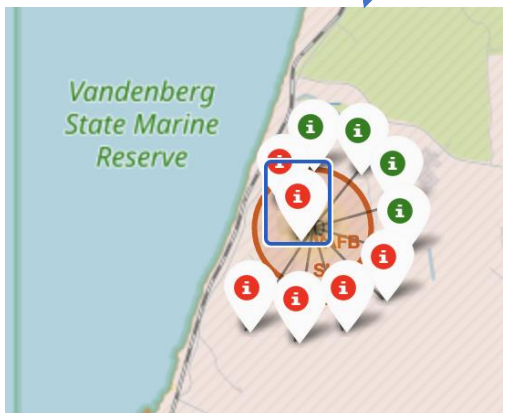
- The launch sites are either on the east coast or west coast of the United States.
- The launch sites are near the southern-most points in the U.S.

At each launch site:
- Success
- Failure

A closer look at each of the launch sites…

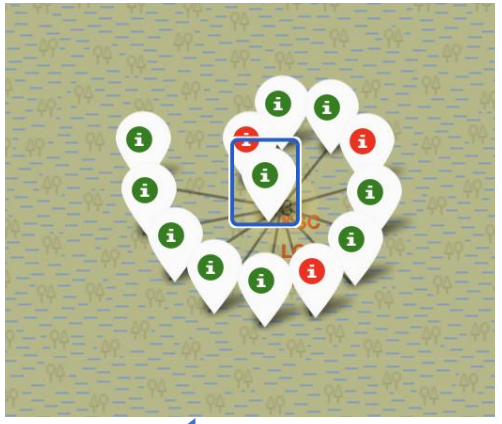
West Coast: VAFB SLC-4E





More failures as time went on

East Coast



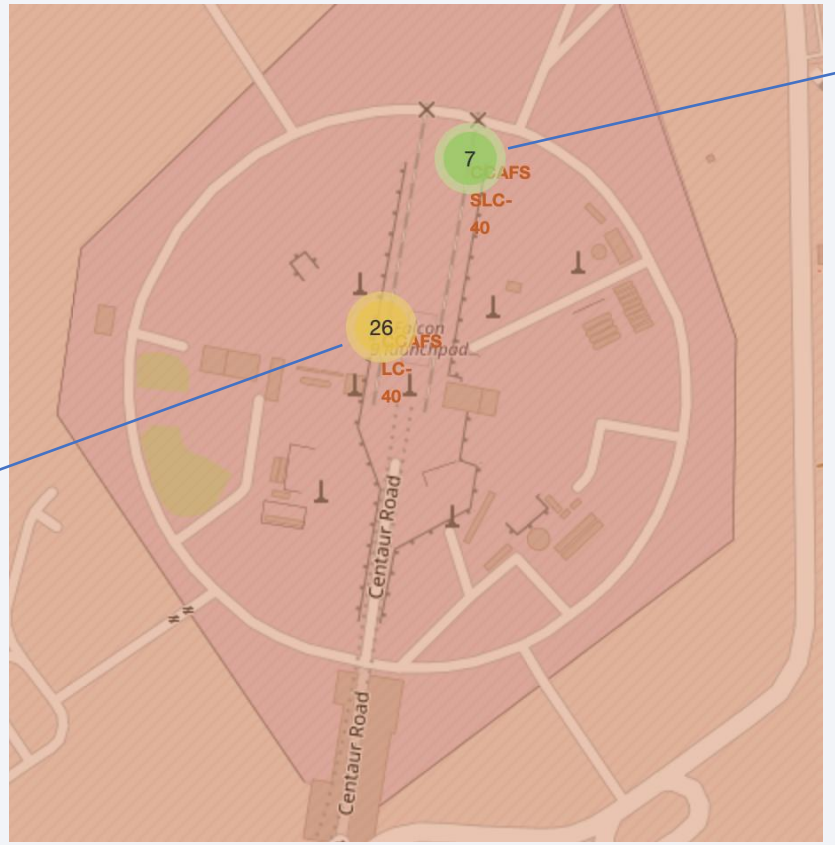
Successes more consistent as time went on

KSC LC-39A

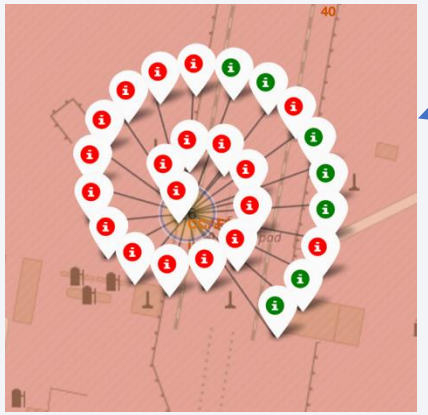
Details
Next slide

Launch Sites – East Coast Continued…

CCAFS SLC-40

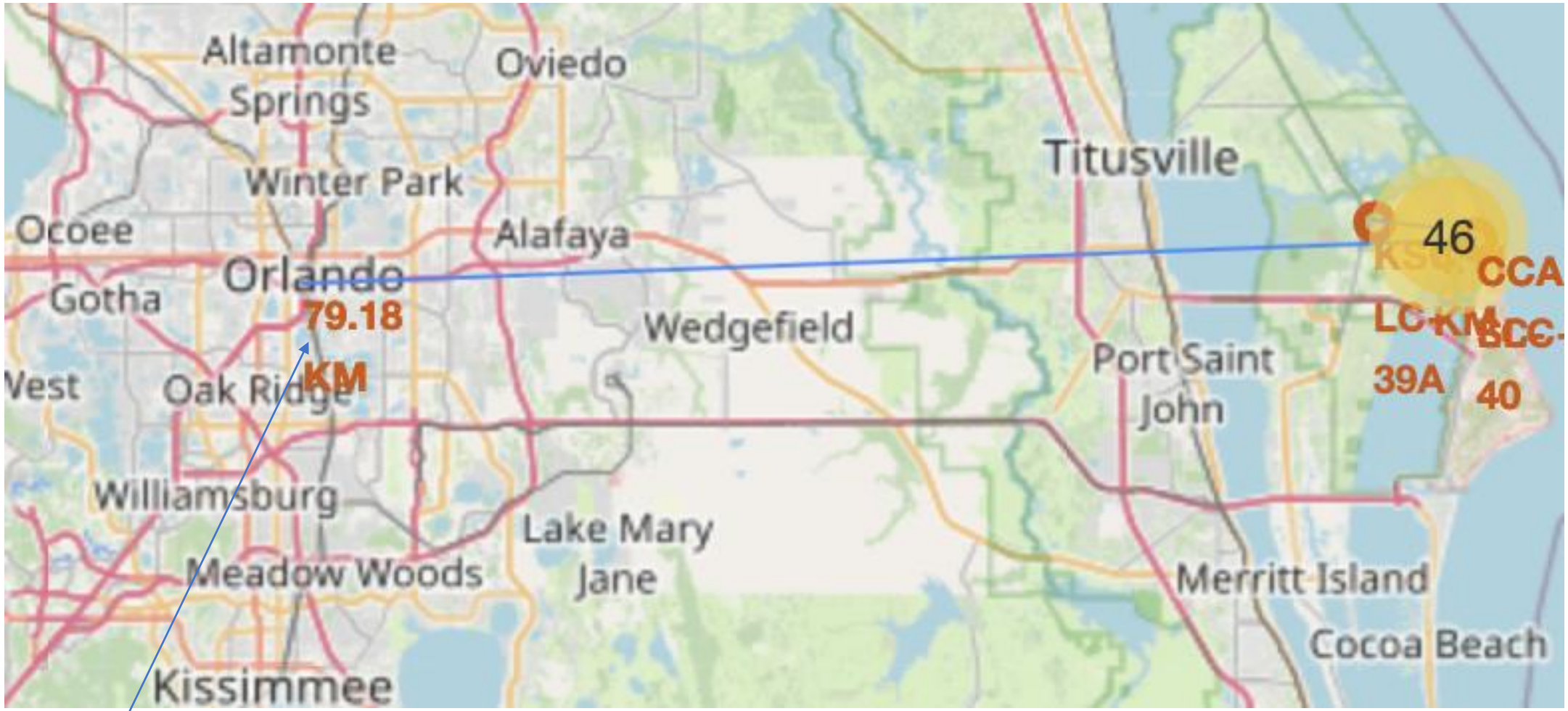


CCAFS LC-40

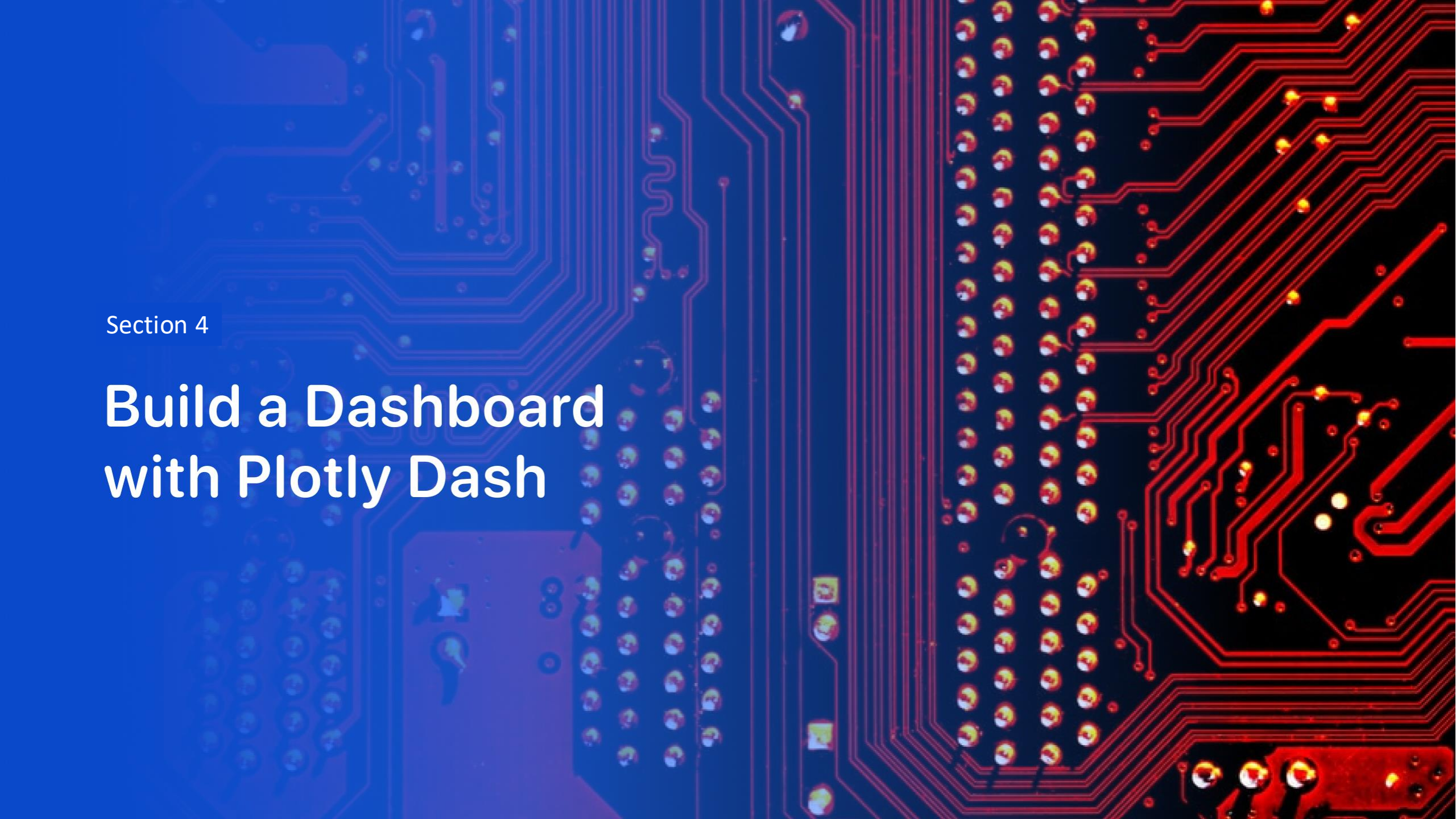


How close are we to the east coast?

How close are we to Orlando, FL?

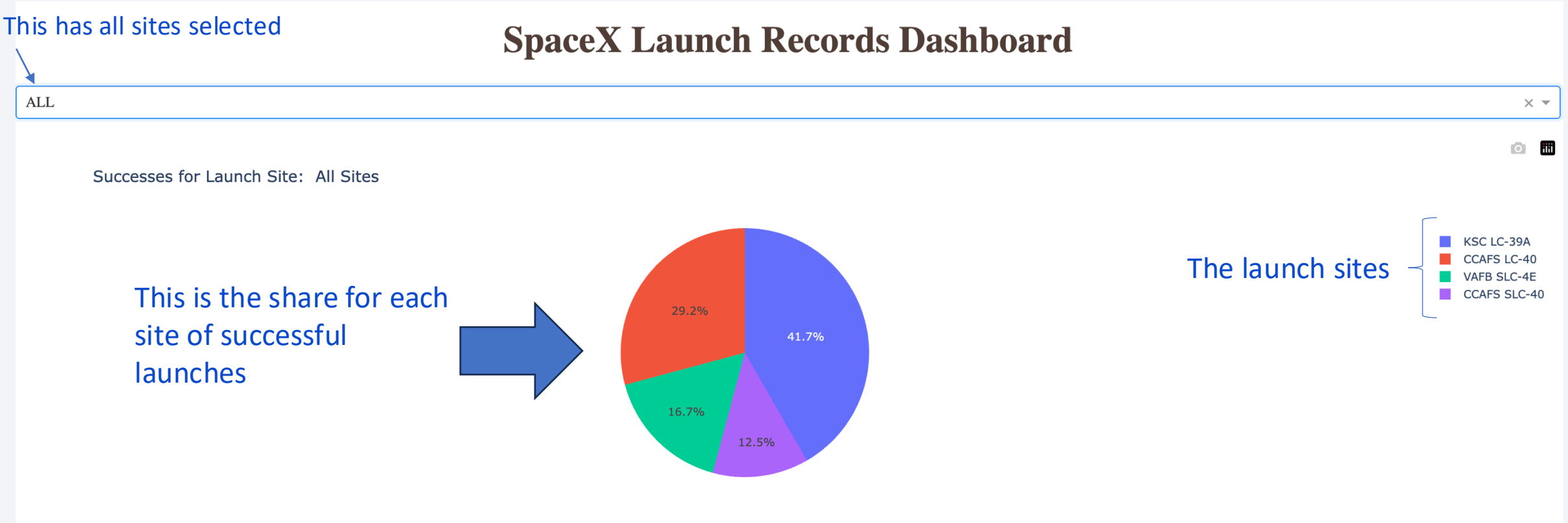


79km away.

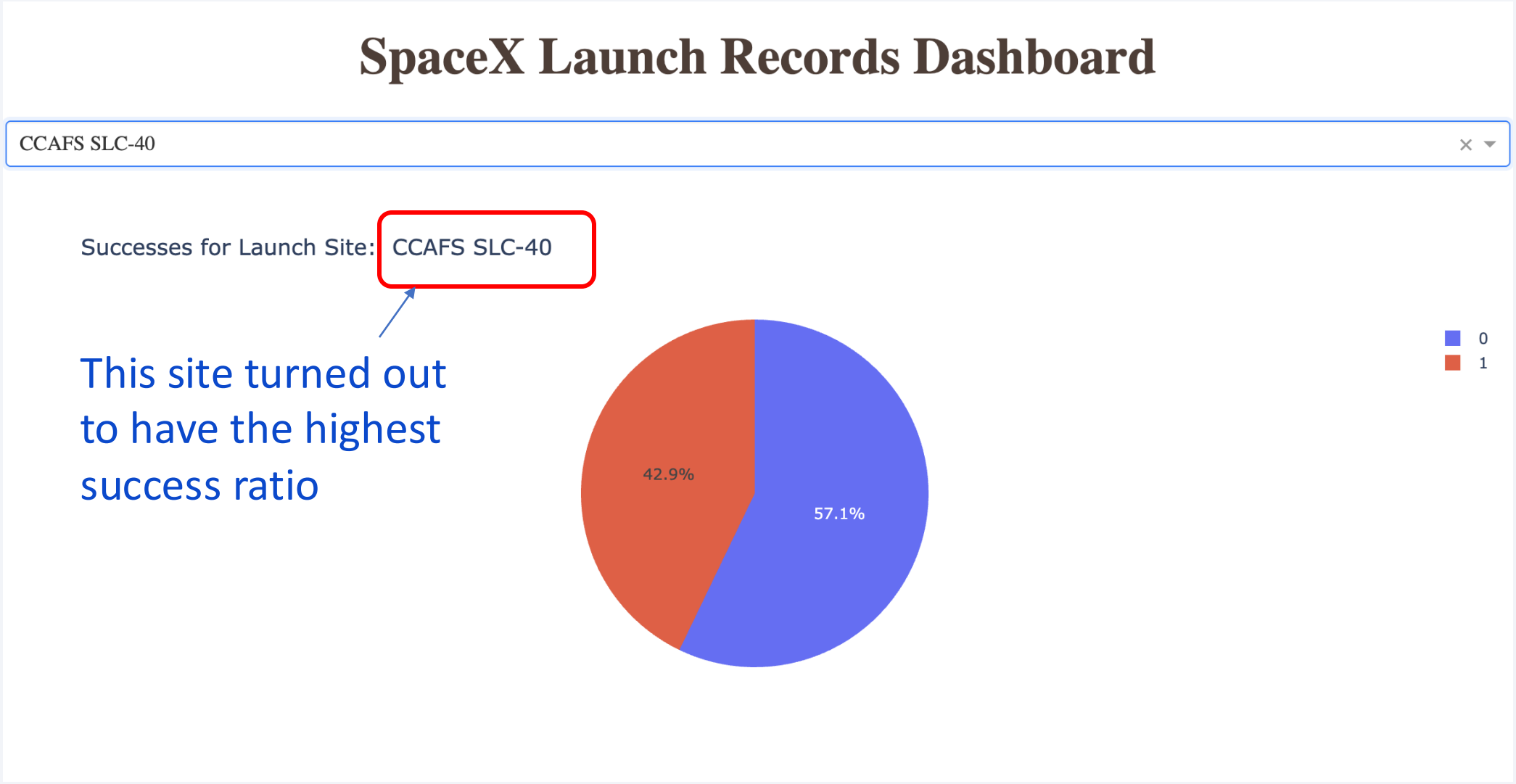

Section 4
Build a Dashboard with Plotly Dash

SpaceX Launch Records Dashboard

This has all sites selected

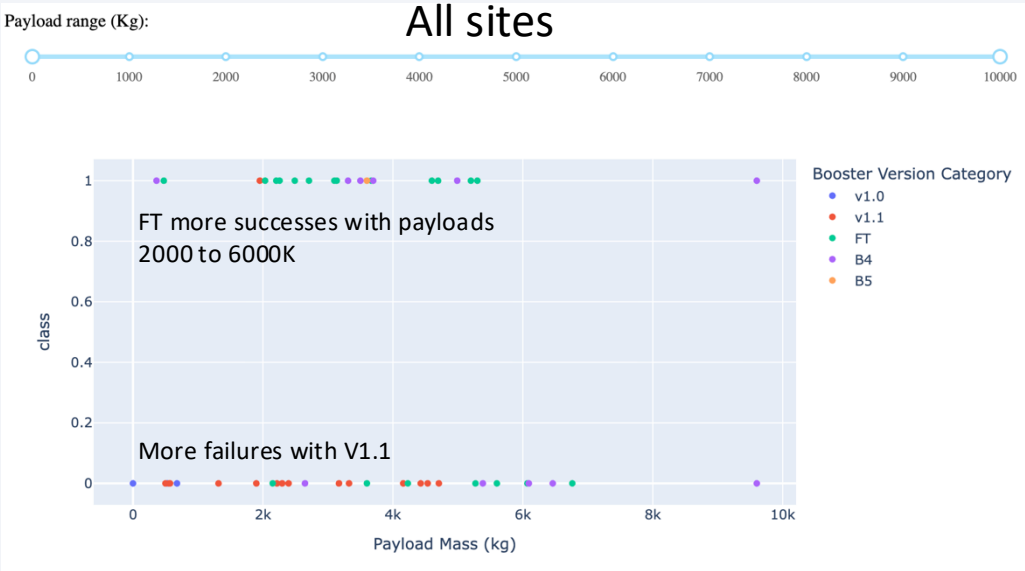


SpaceX Launch Records Dashboard – highest launch success ratio

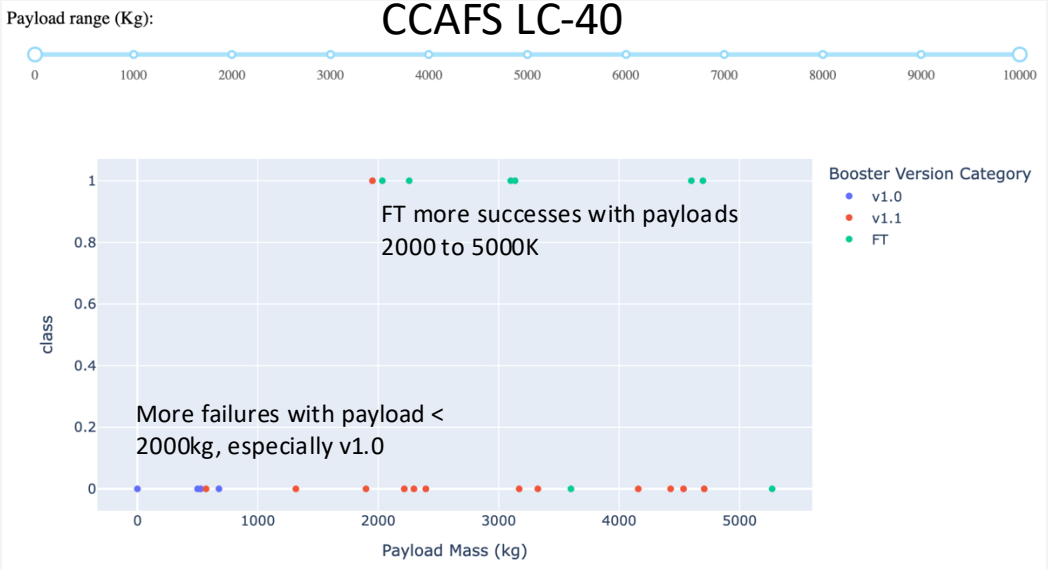


SpaceX Dashboard – Payload vs Outcome

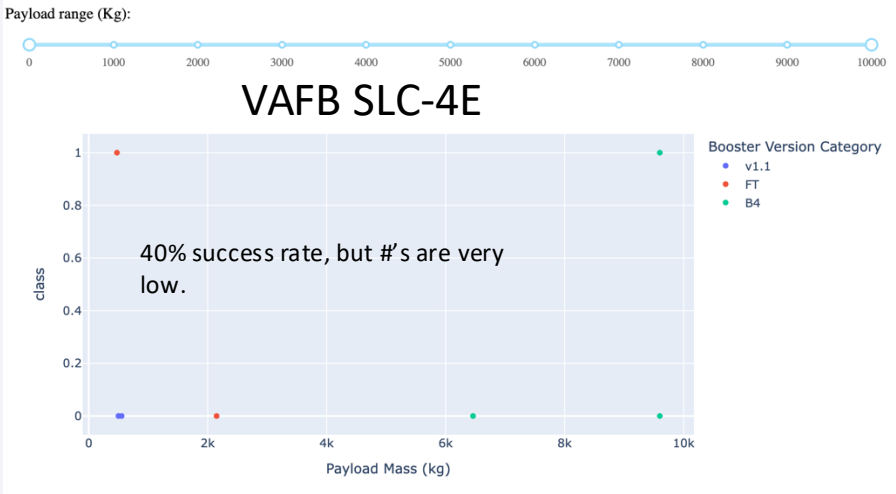
All sites



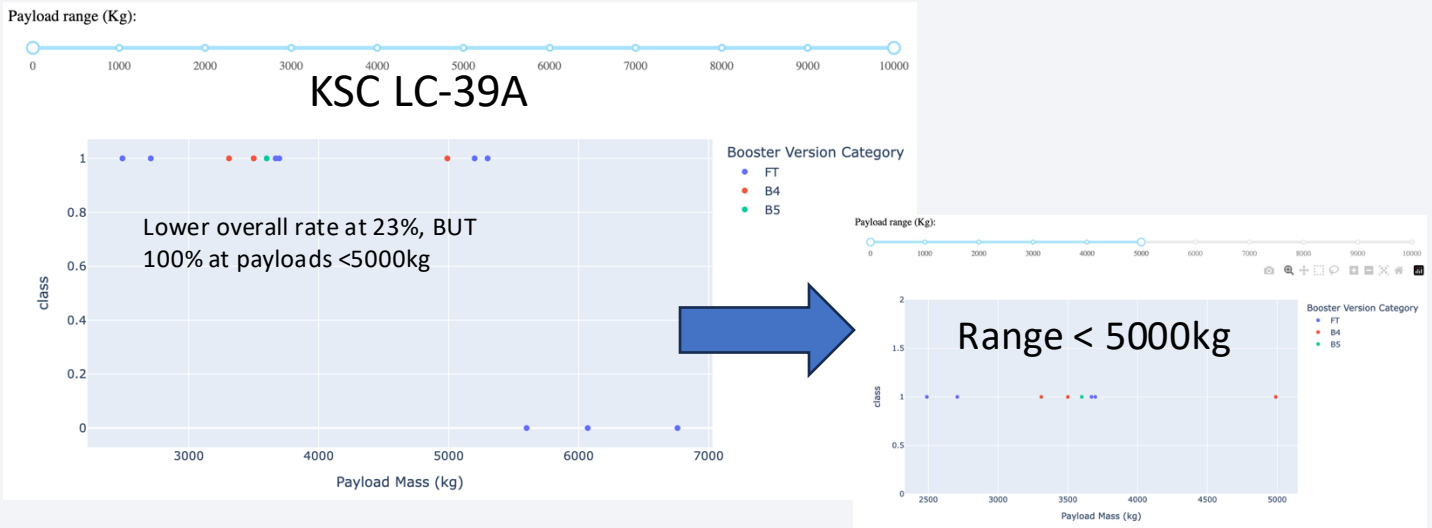
CCAFS LC-40



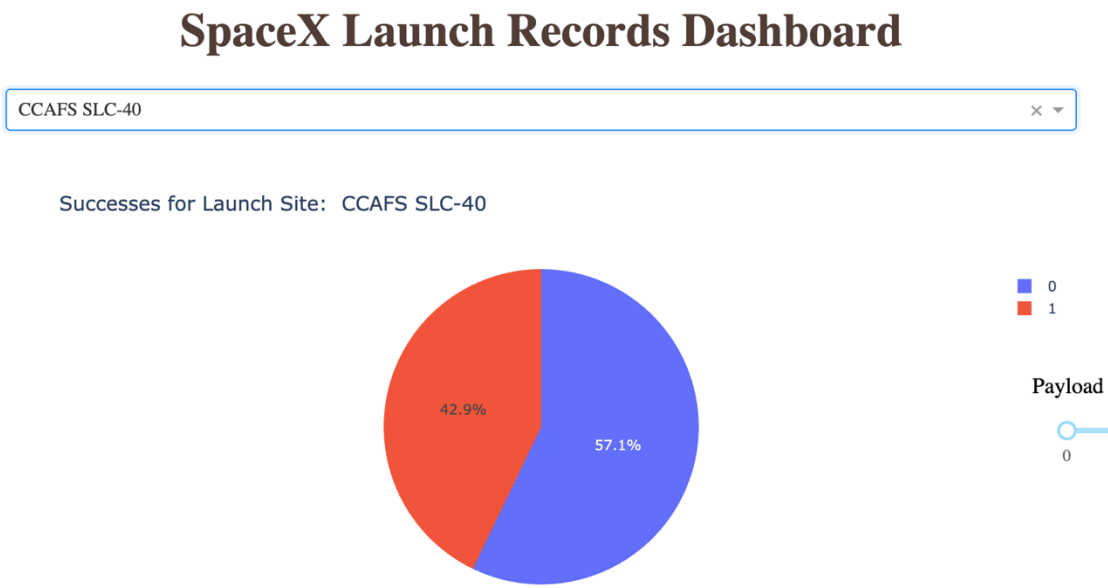
VAFB SLC-4E



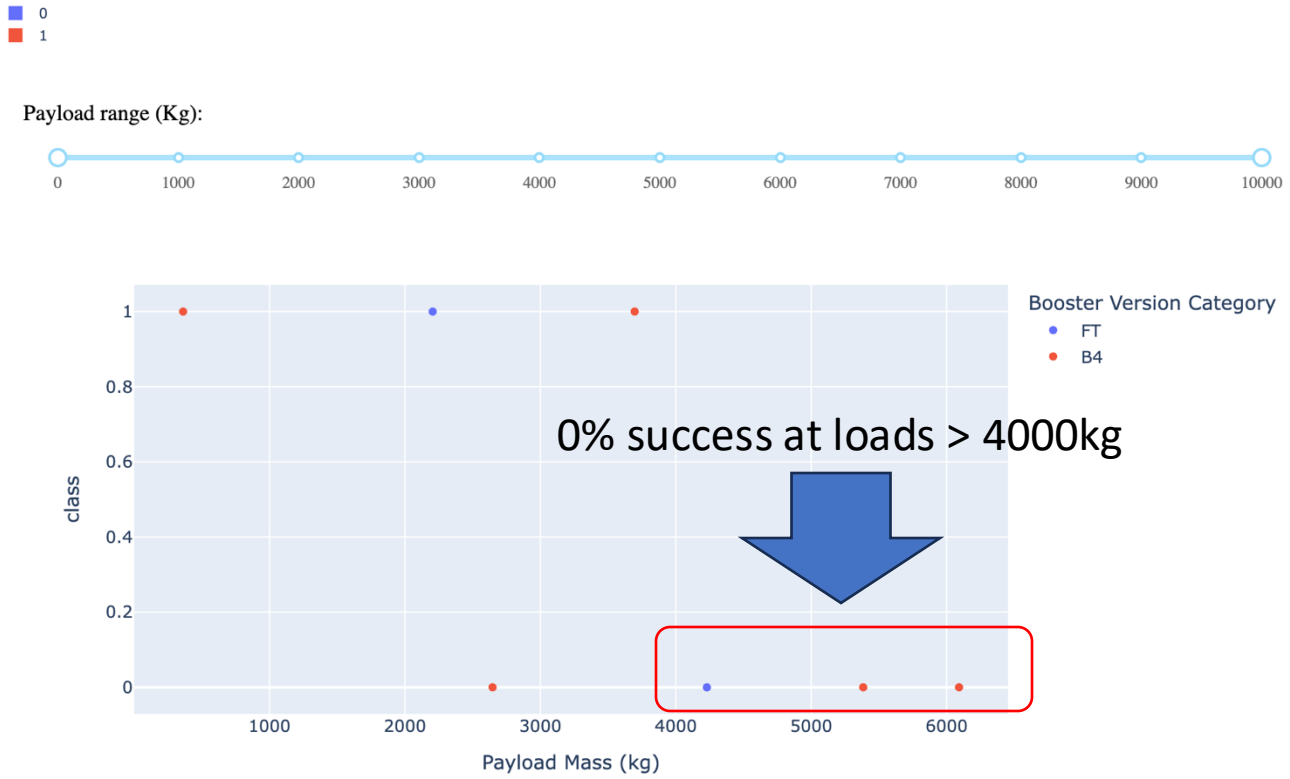
KSC LC-39A



SpaceX Dashboard – Payload vs Outcome



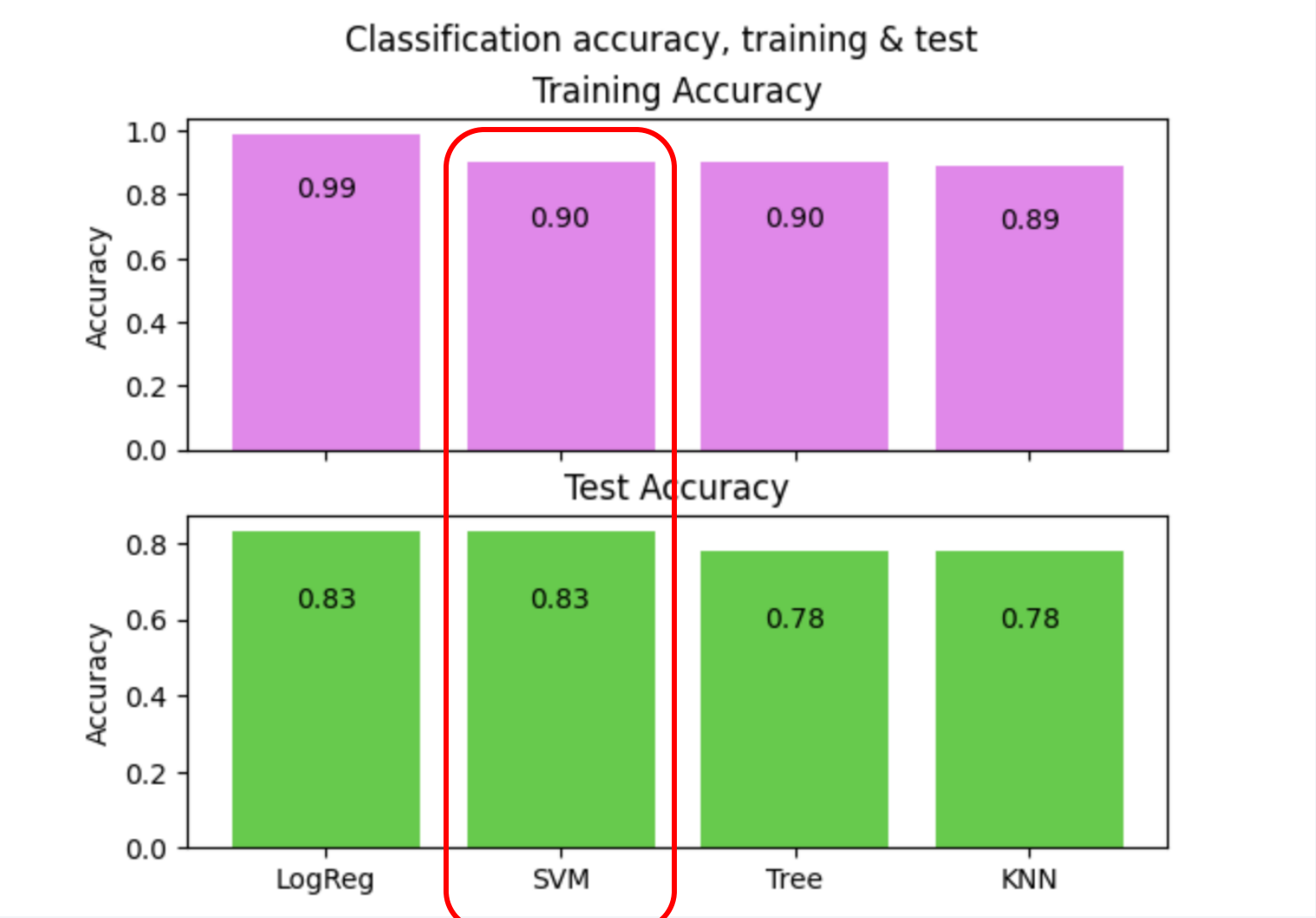
For this site, highest success rate at 42.9%, but only 7 records



Section 5

Predictive Analysis (Classification)

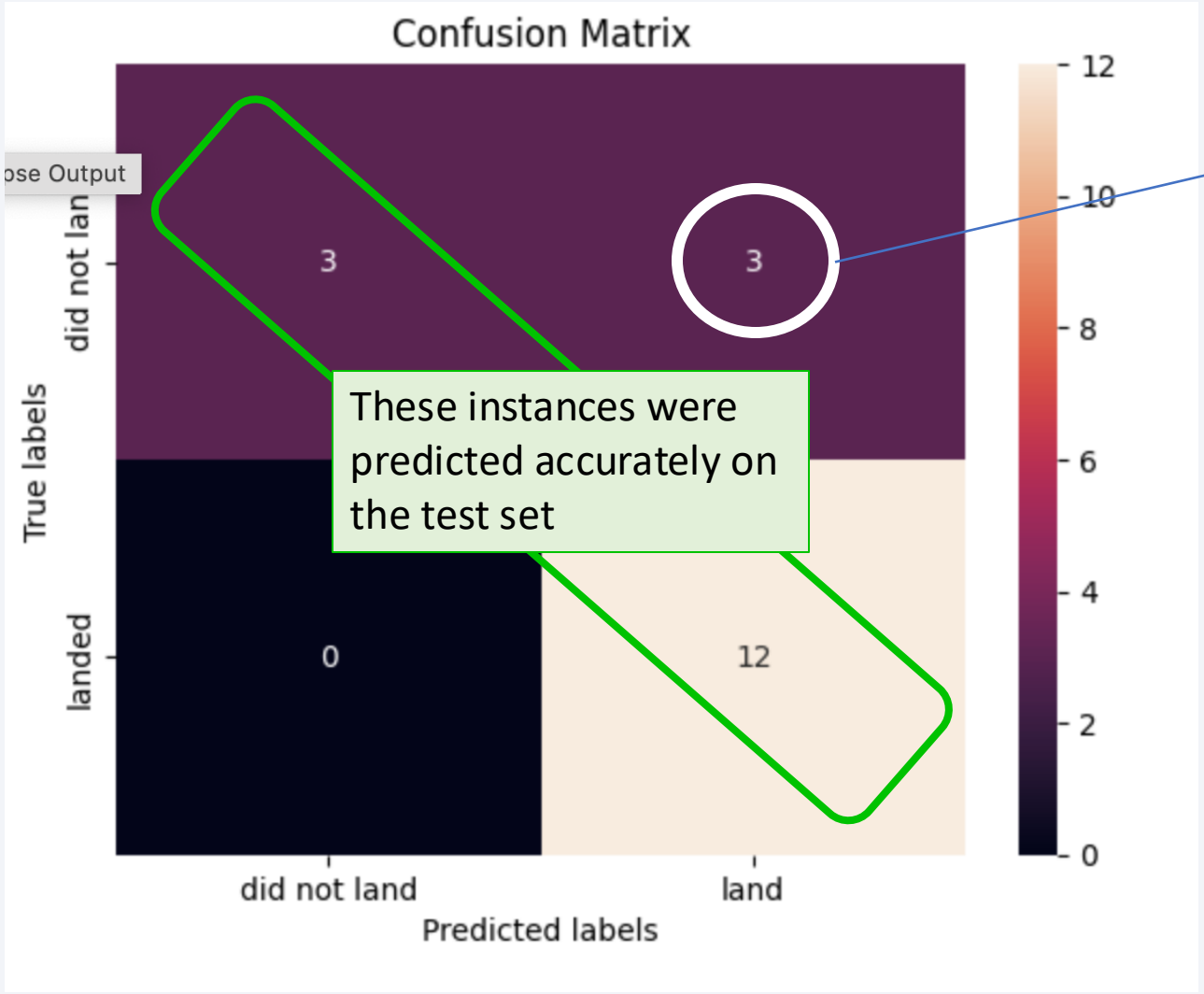
Classification Accuracy



Although the Logistic Regression has the highest training accuracy, there is a big drop from training to test accuracy, suggesting overfitting.

Therefore, the Support Vector Machine had the best overall performance in terms of classification accuracy.

Confusion Matrix – Support Vector Machine



In the test set, 3 false positives were indicated.
→ Showing the rocket would land, but in fact it did not land.

Conclusions

- Rockets bound for higher orbits (ES-L1, GEO, HEO, SSO) tend to have better success rates
- Heavier payloads tend to have better success with the Ploar, LEO and ISS orbits
- Overall success rate has improved to over 80% by 2020
- Although there have been a number of first stage rockets that failed to land successfully, overall mission success has been overwhelming.
- The launch site CCAFS LC-40 had early failures, but improved over time with larget payloads.
- The Support Vector Machine Classifier turned out to be best at predicting if the Falcon 9 rocket would land based on the features available.

Appendix

Since SQLLite did not support month names, use this case statement:

```
%%sql
select
    substr(Date,0,5) as "year",
    (case
        when substr(Date,6,2) = '01' then 'Jan'
        when substr(Date,6,2) = '02' then 'Feb'
        when substr(Date,6,2) = '03' then 'Mar'
        when substr(Date,6,2) = '04' then 'Apr'
        when substr(Date,6,2) = '05' then 'May'
        when substr(Date,6,2) = '06' then 'Jun'
        when substr(Date,6,2) = '07' then 'Jul'
        when substr(Date,6,2) = '08' then 'Aug'
        when substr(Date,6,2) = '09' then 'Sep'
        when substr(Date,6,2) = '10' then 'Oct'
        when substr(Date,6,2) = '11' then 'Nov'
        else 'Dec' end) as month,
    "Landing_Outcome",
    "Booster_Version",
    "Launch_Site"
from SPACEXTABLE
where "Landing_Outcome" = 'Failure (drone ship)' and
    substr(Date,0,5) = '2015'
```

Appendix

For max_features in the Decision Tree Classifier, the ‘auto’ option will now cause an error.
Replaced it with ‘log2’. The first option was to leave it as ‘sqrt’ but accuracy was poor.

```
parameters = {'criterion': ['gini', 'entropy'],
     'splitter': ['best', 'random'],
     'max_depth': [2*n for n in range(1,10)],
     'max_features': ['log2'],  # removed 'auto' as this was causing an error and the fit failed. 'auto' deprecat
     'min_samples_leaf': [1, 2, 4],
     'min_samples_split': [2, 5, 10]}

# changed max_features from sqrt to log2. This decreased the training accuracy a little from 0.91 to 0.89, but
# improved test accuracy from 0.61 to 0.83.

tree = DecisionTreeClassifier()
```

Thank you!