IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Johannes Hofmann
06.06.2025

Outline

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

Executive Summary

- Summary of methodologies
  - Data Collection through SpaceX API and web scraping
  - Data wrangling
  - Exploratory Data Analysis (EDA) with SQL and Data Visualization
  - Interactive Visual Analytics (dashboard)
  - Machine Learning Prediction
- Summary of all results
  - It was possible to collect valuable data from public sources
  - EDA allowed to identify the most important features for prediction
  - Logistic Regression, SVM, Decision Trees and KNN all performed equally well on this dataset.

Introduction

- Space X advertises Falcon 9 rocket launches with a cost of 62 million dollars
- Other providers cost upward of 165 million dollars each
- Much of the savings is because Space X can reuse the first stage.
- Therefore, if we can determine if the first stage will land, we can determine the cost of a launch.
- The goal of the project is to create a machine learning pipeline to predict if the first stage will land successfully.

Problems you want to find answers

- What is the nature and extent of the available data about SpaceX Falcon 9 first stage landings?
- Which machine learning model would work best to predict the outcome of a first stage landing from a future launch?
- Will a future Falcon 9 first stage landing be successful?

Section 1

Methodology

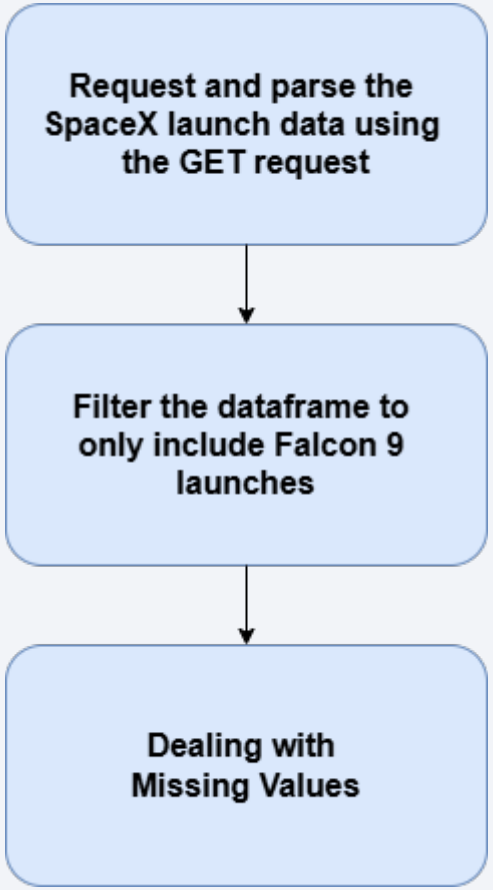
Methodology

Executive Summary

- Data collection methodology:
  - Data on the SpaceX Falcon 9 first stage landings was collected from a public API unaffiliated with SpaceX, and from a Wikipedia article.
- Perform data wrangling
  - Data was wrangled/cleaned (e.g. handling missing data) in preparation for visualizations, queries, and machine learning model training.
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - How to build, tune, evaluate classification models

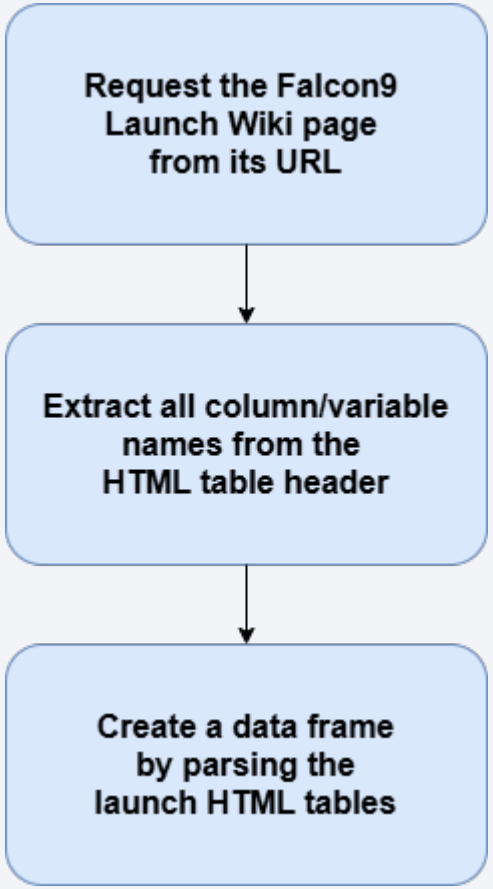
Data Collection – SpaceX API

- Data from past launches was collected using the get request function to the SpaceX API.
- The response content was parsed to a JSON object using .json() and turn it into a Pandas dataframe using .json_normalize()
- Data was filtered to only include Falcon 9 lunches
- Missing data for payload mass was replaced with a mean value
- **GitHub URL**



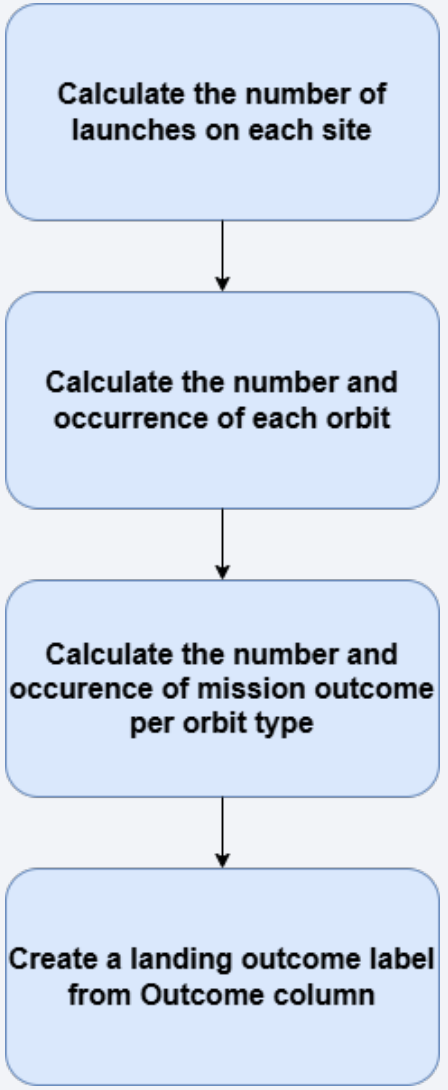
Data Collection – Web Scraping

- Web scraping Falcon 9 launch records using BeautifulSoup.
- Iterate through the HTML Table elements to extract the colume names
- Create a dictionary from the Soup object
- Parse the dictionary to a pandas data frame
- **GitHub URL**



Data Wrangling

- The CSV file was loaded to a data frame.
- The launch sites, orbit types and mission outcomes were processed and reformatted.
- The mission outcome types were converted to a binary classification (onehot encoding) where 1 represented the Falcon 9 first stage landing being a success and 0 represented a failure.
- The new mission outcome classification column was added to the DataFrame.
- **GitHub URL**



EDA with Data Visualization

The following charts were created to look at Launch Site trends

- Catplot: FlightNumber vs. PayloadMass (hue parameter set to ‘class’)
- Catplot: FlightNumber vs LaunchSite (hue parameter set to ‘class’)
- Catplot: LaunchSite vs PayloadMass (hue parameter set to ‘class’)
- Barchart: Class vs Orbit type (Class refers to mission outcome)
- Catplot: Orbit type vs FlightNumber (hue parameter set to ‘class’)
- Catplot: Orbit type vs PayloadMass (hue parameter set to ‘class’)
- Line plot: Class vs Date

- **GitHub URL**

EDA with SQL

- SQL queries performed
  - Display the names of the unique launch sites in the space mission
  - Display 5 records where launch sites begin with the string 'KSC‘
  - Display the total payload mass carried by boosters launched by NASA (CRS)
  - Display average payload mass carried by booster version F9 v1.1
  - List the date where the successful landing outcome in drone ship was achieved.
  - List the names of the boosters which have success in ground pad and have payload mass greater than 4000 but less than 6000
  - List the total number of successful and failure mission outcomes

EDA with SQL

- SQL queries performed (continued)
  - List all the booster versions that have carried the maximum payload mass
  - List the records which will display the month names, successful landing outcomes in ground pad ,booster versions, launch site for the months in year 2017
  - 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

- **GitHub URL**

Build an Interactive Map with Folium

- Map objects were created and added to the Folium map
- Markers /Marker Clusters were added for launch sites
- Circles were added for the launch sites.
- Lines were added to show the distance to the nearby features:
  - Distance from CCAFS LC-40 to the coastline
  - Distance from CCAFS LC-40 to the railway
  - Distance from CCAFS LC-40 to road

- **GitHub URL**

Build a Dashboard with Plotly Dash

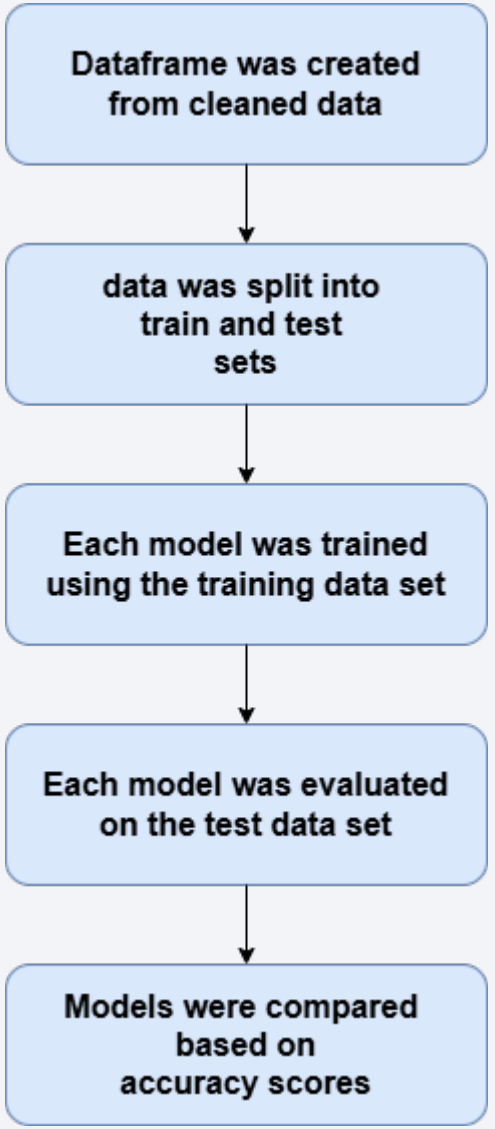
The Plotly dashboard included a dropdown menu to select data from one specific or from all launch sites to display on the pie chart and scatterplot.

- For a specific launch site, the pie chart displays the distribution of successful and failed first stage landings for that site.
- For all launch sites, the pie chart displays the distribution of successful first stage landings between all the sites.
- The input slider is used to filter the payload masses for the scatterplot.
- The scatterplot displays the distribution of Falcon 9 first stage landings split by payload mass, mission outcome and by booster version category.

- **GitHub URL**

Predictive Analysis (Classification)

- Creation of the target variable Y as a NumPy array from the column “Class”
- Standardization of the feature variables X
- The function train_test_split was used to split the data X and Y into training and test data
- The following machine learning models were trained
  - SVM (Support Vector Machine)
  - Decision Tree
  - KNN (k-Nearest Neighbors)
  - Logistic Regression
- Hyper parameters were evaluated using GridSearchCV
- Using the best hyper parameters, each model was scored on accuracy
- **GitHub URL**



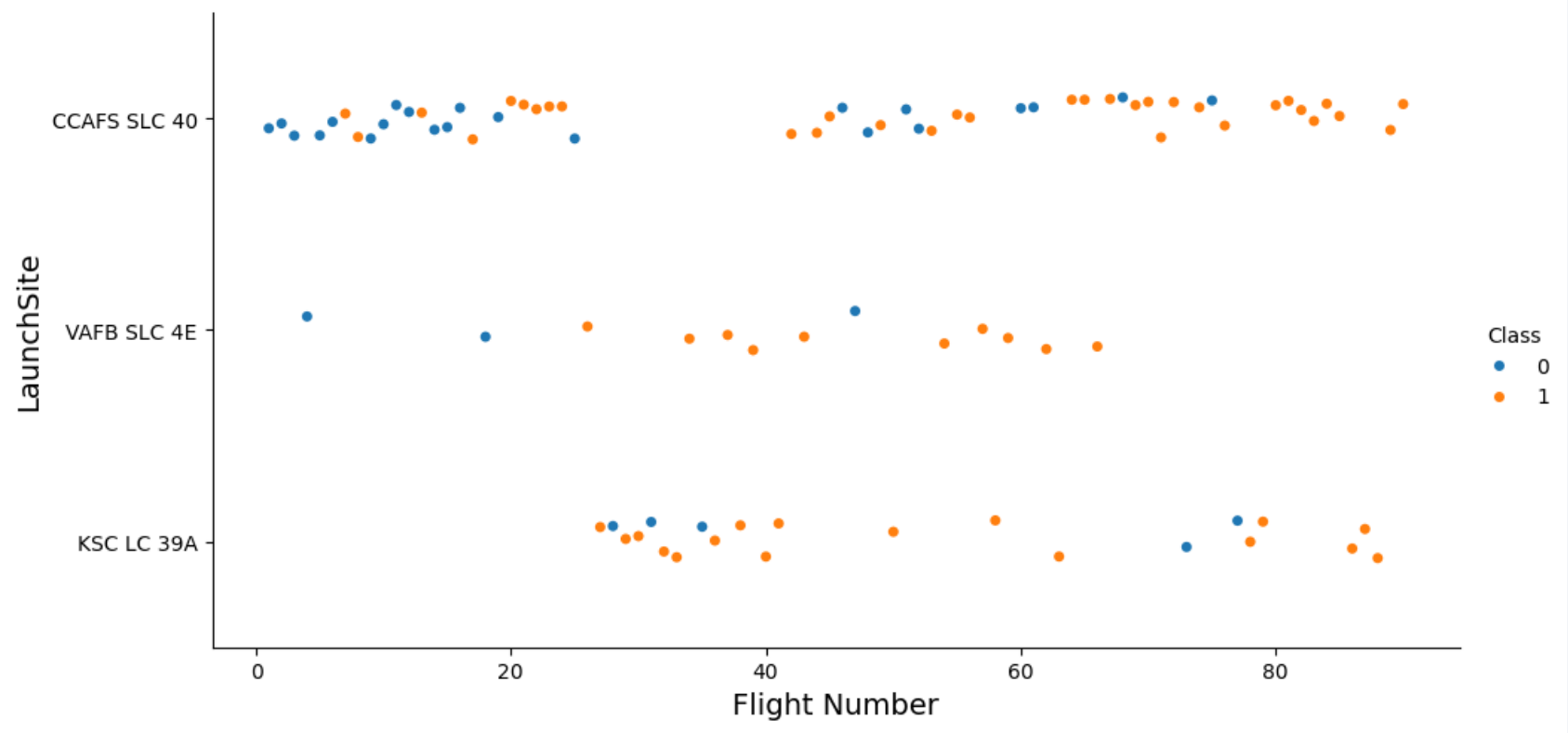
Results

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

Section 2

Insights drawn from EDA

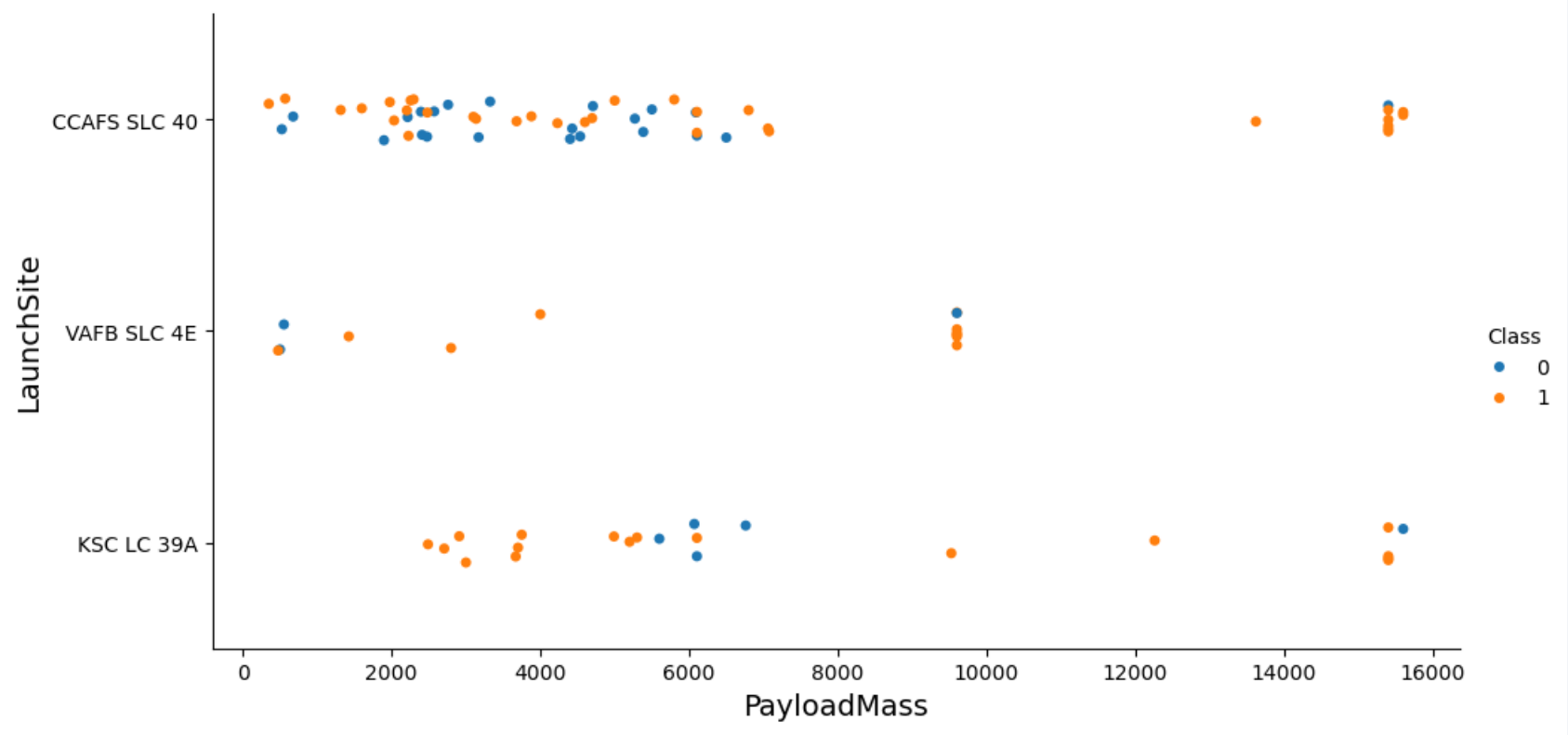
Flight Number vs. Launch Site



Class = 1 : Successful
Class = 0 : Unsuccessful

- Success rate improves over time for every launch site. Especially for CCAFS SLC40
- Success rate varies between launch sites

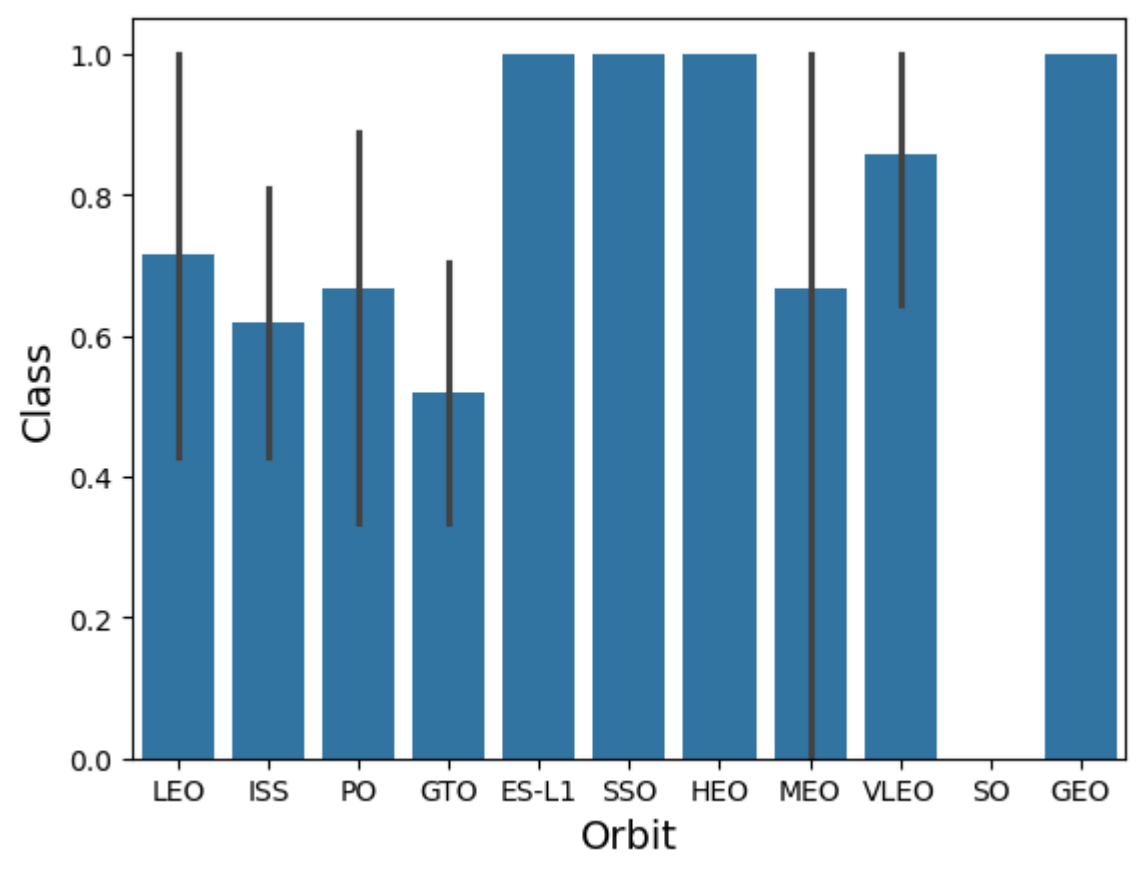
Payload vs. Launch Site



Class = 1 : Successful
Class = 0 : Unsuccessful

- Launches with PayloadMass above 9000 kg are more likely to be successful
- Payload mass above 10000 kg semms to be only possible at CCAFS SLC 40 and KSC LC39A

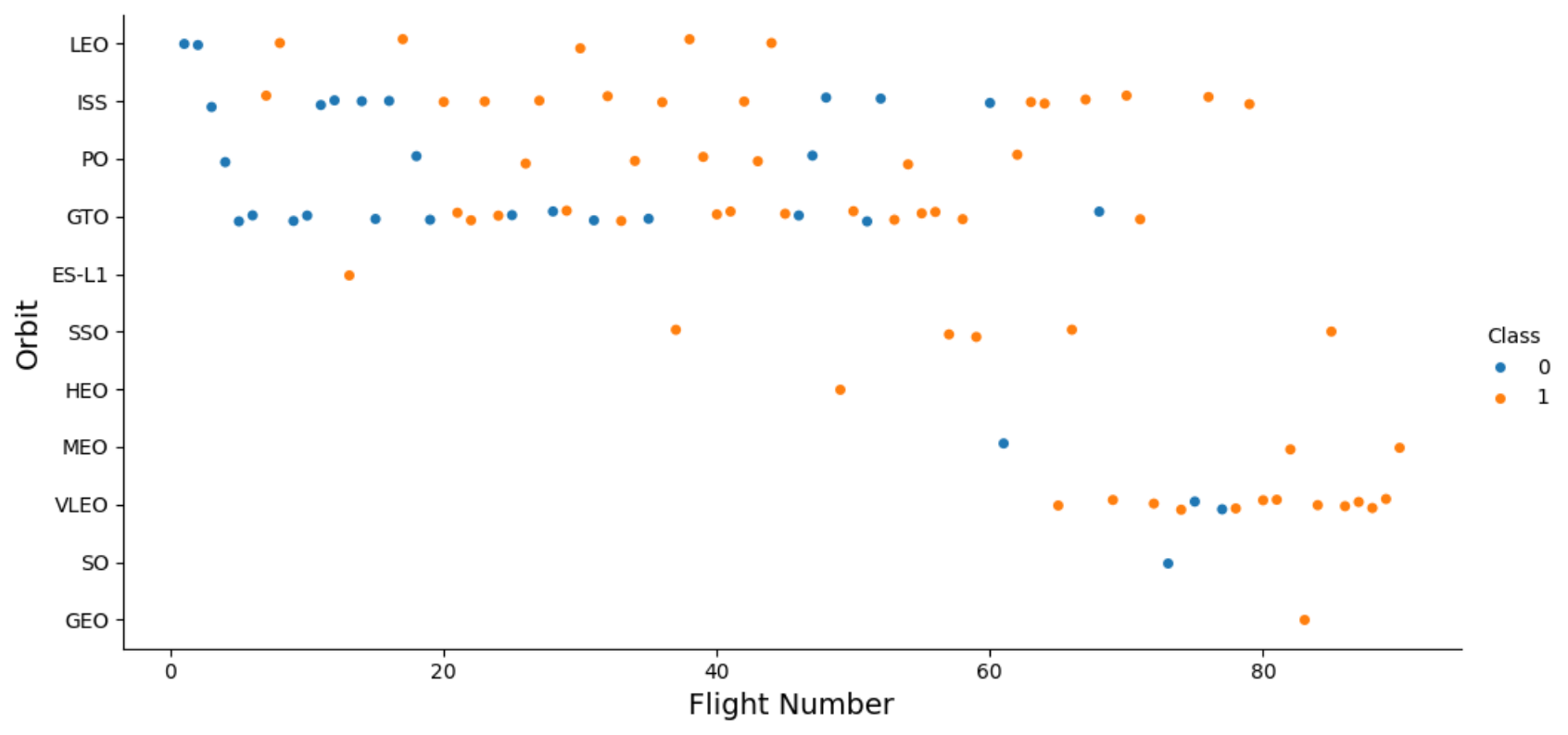
Success Rate vs. Orbit Type



Class = 1 : Successful
Class = 0 : Unsuccessful

- ES-L1, SSO, HEO and GEO orbits have no failed first stage landings.
- But bar chart needs to be interpreted with number of launches per orbit type

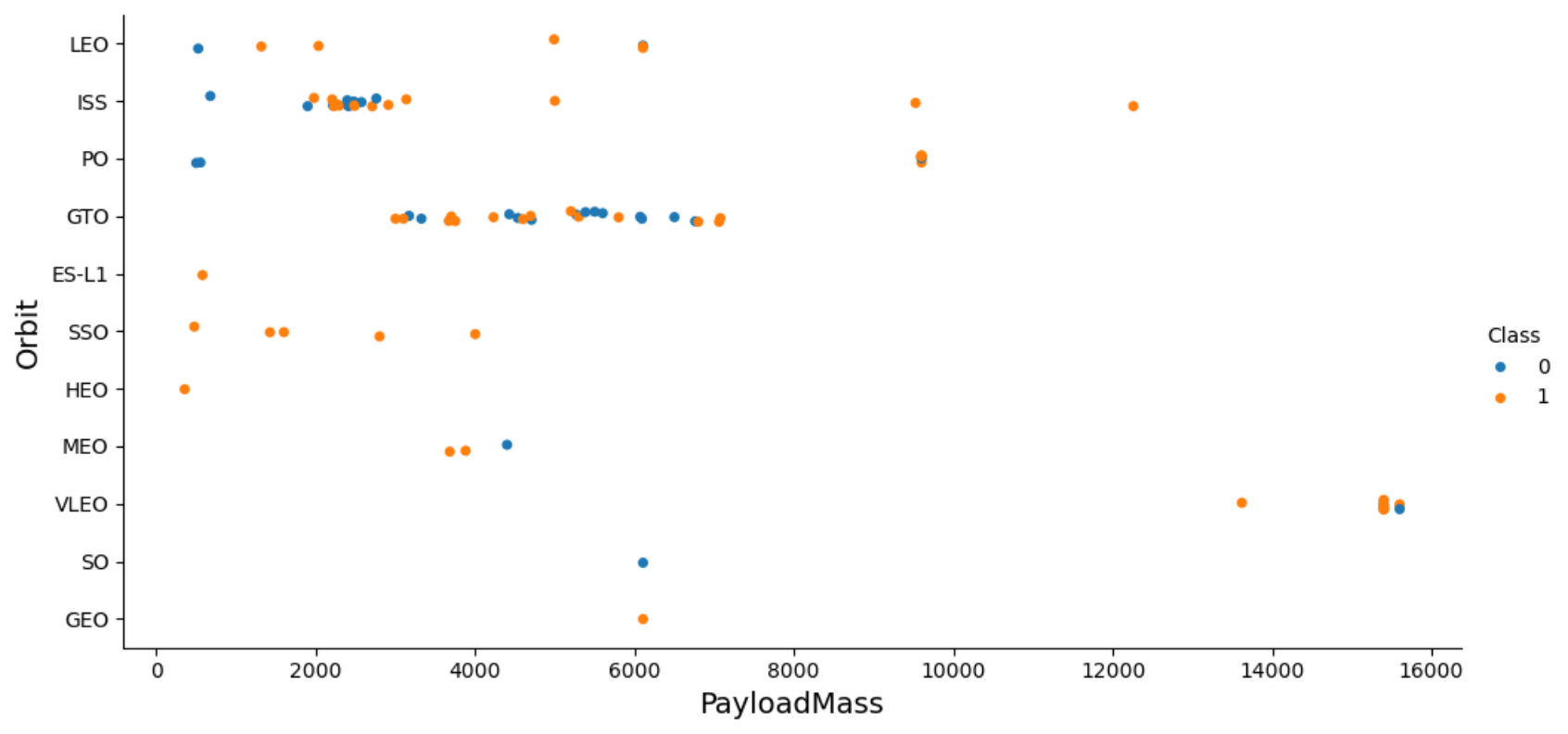
Flight Number vs. Orbit Type



Class = 1 : Successful
Class = 0 : Unsuccessful

- As expected successes rate increases with Flight Number
- ES-L1, SSO, HEO and GEO have only one or a few launches

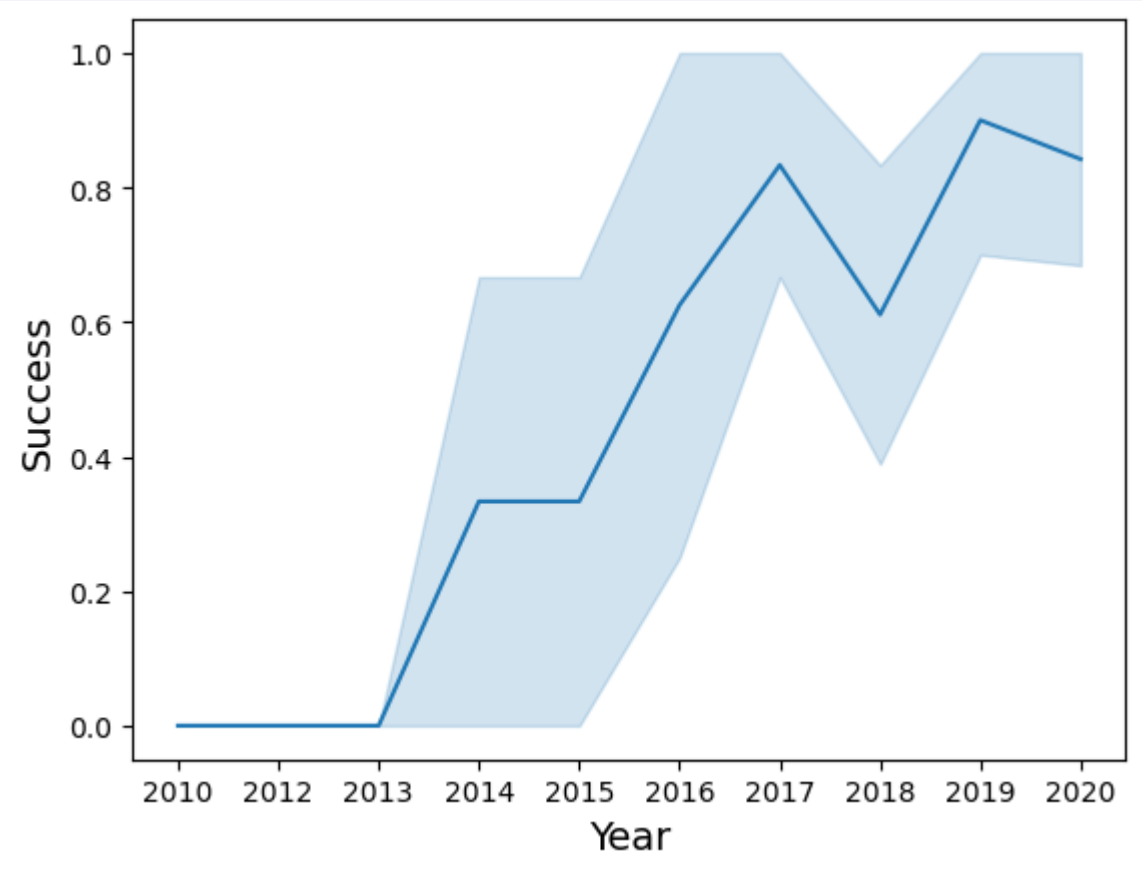
Payload vs. Orbit Type



Class = 1 : Successful
Class = 0 : Unsuccessful

- Orbit types with higher Payload seem to be more successful
- However for GTO there seems to be no correlation between Successes and Payload

Launch Success Yearly Trend



- The Successes rate is increasing since 2013

All Launch Site Names

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

```
%config SqlMagic.style = '_DEPRECATED_DEFAULT'
```

```
%sql select Distinct Launch_Site from SPACEXTBL;
```

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

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

- Distinct is the key word to select unique entries.

Launch Site Names Begin with 'KSC'

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

```
%sql select * FROM SPACEXTBL WHERE launch_site LIKE 'KSC%' LIMIT 5;
```

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

Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Orbit	Customer	Mission_Outcome	Landing_Outcome
2017-02-19	14:39:00	F9 FT B1031.1	KSC LC-39A	SpaceX CRS-10	2490	LEO (ISS)	NASA (CRS)	Success	Success (ground pad)
2017-03-16	6:00:00	F9 FT B1030	KSC LC-39A	EchoStar 23	5600	GTO	EchoStar	Success	No attempt
2017-03-30	22:27:00	F9 FT B1021.2	KSC LC-39A	SES-10	5300	GTO	SES	Success	Success (drone ship)
2017-05-01	11:15:00	F9 FT B1032.1	KSC LC-39A	NROL-76	5300	LEO	NRO	Success	Success (ground pad)
2017-05-15	23:21:00	F9 FT B1034	KSC LC-39A	Inmarsat-5 F4	6070	GTO	Inmarsat	Success	No attempt

- LIKE is the key word to surch for a given string
- LIMIT defines the numbers of output rows

Total Payload Mass

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

```
%sql select sum(PAYLOAD_MASS__KG_) from SPACEXTBL Where Customer = 'NASA (CRS)'
```

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

sum(PAYLOAD_MASS__KG_)
45596

- Sum() function is used to calculate the sum of the payload

Average Payload Mass by F9 v1.1

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

```
%sql select AVG(PAYLOAD_MASS__KG_) from SPACEXTBL Where Booster_Version = 'F9 v1.1'
```

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

AVG(PAYLOAD_MASS__KG_)
2928.4

- AVG() function is used to calculate the average weight of the F9 v1.1 Booster

First Successful Ground Landing Date

List the date where the succesful landing outcome in drone ship was acheived.

Hint:Use min function

```
%sql select min(Date) from SPACEXTBL Where Landing_Outcome = 'Success (drone ship)'
```

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

min(Date)
2016-04-08

- Min() function is used to find the earliest date with an successful landing outcome on a drone ship

Successful Drone Ship Landing with Payload between 4000 and 6000

List the names of the boosters which have success in ground pad and have payload mass greater than 4000 but less than 6000

```
%%sql  SELECT Booster_Version
FROM SPACEXTBL
WHERE Landing_Outcome = 'Success (ground pad)'
AND PAYLOAD_MASS__KG_ BETWEEN 4001 AND 5999;
```

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

Booster_Version
F9 FT B1032.1
F9 B4 B1040.1
F9 B4 B1043.1

- AND is used to combine logic statements
- BETWEEN is used to filter the weight

Total Number of Successful and Failure Mission Outcomes

List the total number of successful and failure mission outcomes

```
%sql select Count(Mission_Outcome) from SPACEXTBL Where Mission_Outcome = 'Success'
```

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

Count(Mission_Outcome)
98

```
%sql select Count(Mission_Outcome) from SPACEXTBL Where Mission_Outcome != 'Success'
```

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

Count(Mission_Outcome)
3

- Count() function is used to count the total number of events

Boosters Carried Maximum Payload

List all the booster_versions that have carried the maximum payload mass. Use a subquery.

```
%sql select Distinct Booster_Version from SPACEXTBL Where PAYLOAD_MASS__KG_ = (select max(PAYLOAD_MASS__KG_) from SPACEXTBL);
```

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

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

- Subquery in the Where clause is used to select the max payload mass

2015 Launch Records

List the records which will display the month names, succesful landing_outcomes in ground pad ,booster versions, launch_site for the months in year 2017

Note: SQLLite does not support monthnames. So you need to use substr(Date,6,2) for month, substr(Date,9,2) for date, substr(Date,0,5),='2017' for year.

```
%sql select substr(Date,6,2) as Month ,Booster_Version, Launch_Site, Landing_Outcome  from SPACEXTBL Where Landing_Outcome = 'Success (ground pad)' and  substr(Date,
```

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

Month	Booster_Version	Launch_Site	Landing_Outcome
02	F9 FT B1031.1	KSC LC-39A	Success (ground pad)
05	F9 FT B1032.1	KSC LC-39A	Success (ground pad)
06	F9 FT B1035.1	KSC LC-39A	Success (ground pad)
08	F9 B4 B1039.1	KSC LC-39A	Success (ground pad)
09	F9 B4 B1040.1	KSC LC-39A	Success (ground pad)
12	F9 FT B1035.2	CCAFS SLC-40	Success (ground pad)

- Substr() function is used to select specific parts from the date entry

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

```
%%sql select Landing_Outcome, count(Landing_Outcome) as Landing_count from SPACEXTBL
    WHERE Date BETWEEN '2010-06-04' AND '2017-03-20' Group By Landing_Outcome ORDER BY Landing_count DESC;
```

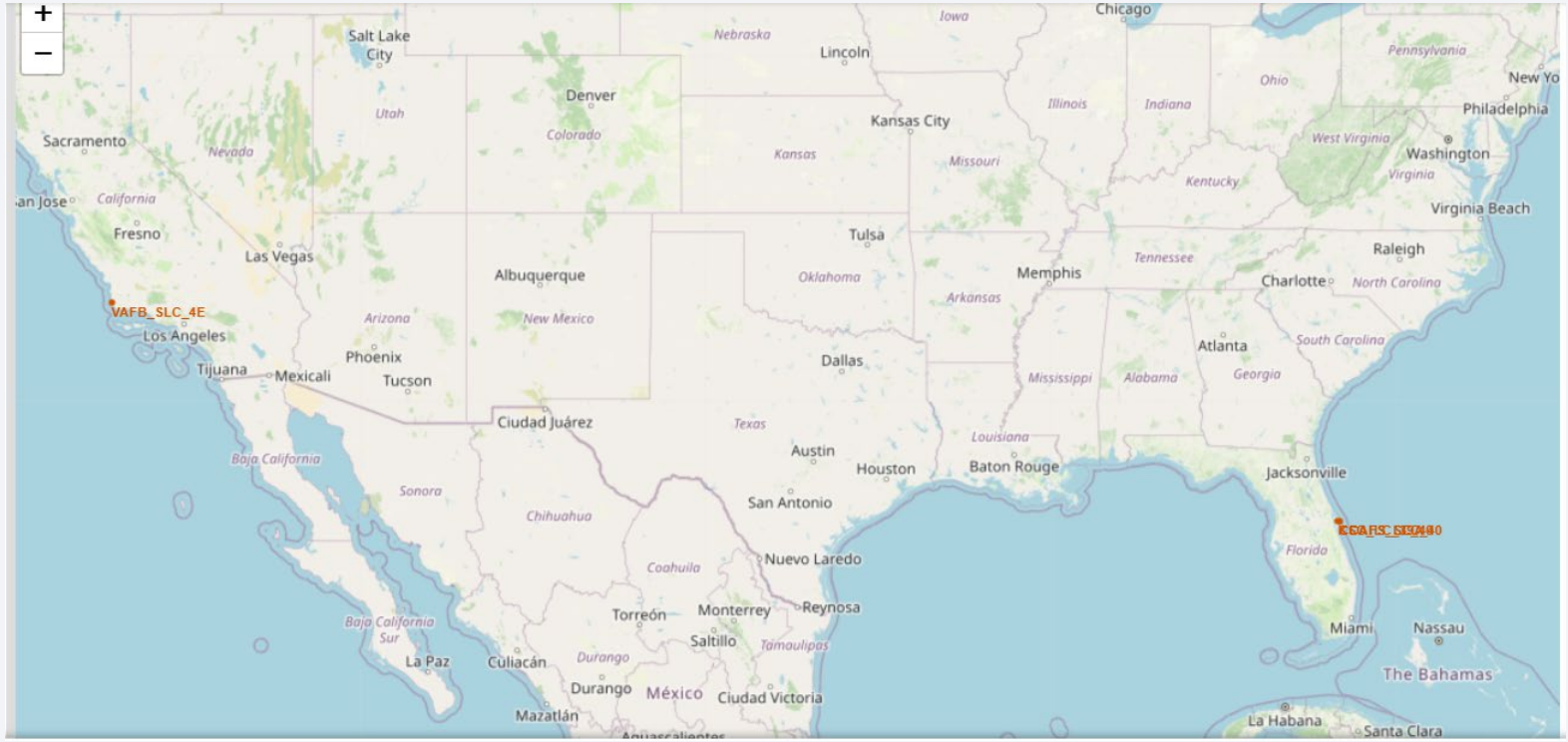
```
 * sqlite:///my_data1.db
Done.
```

Landing_Outcome	Landing_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

- Group By is used to grouped the data by landing outcome
- Order By is used to sort by landing count
- DESC is the keyword to display the table in descending order


Section 3
Launch Sites
Proximities Analysis

Launch Site Locations

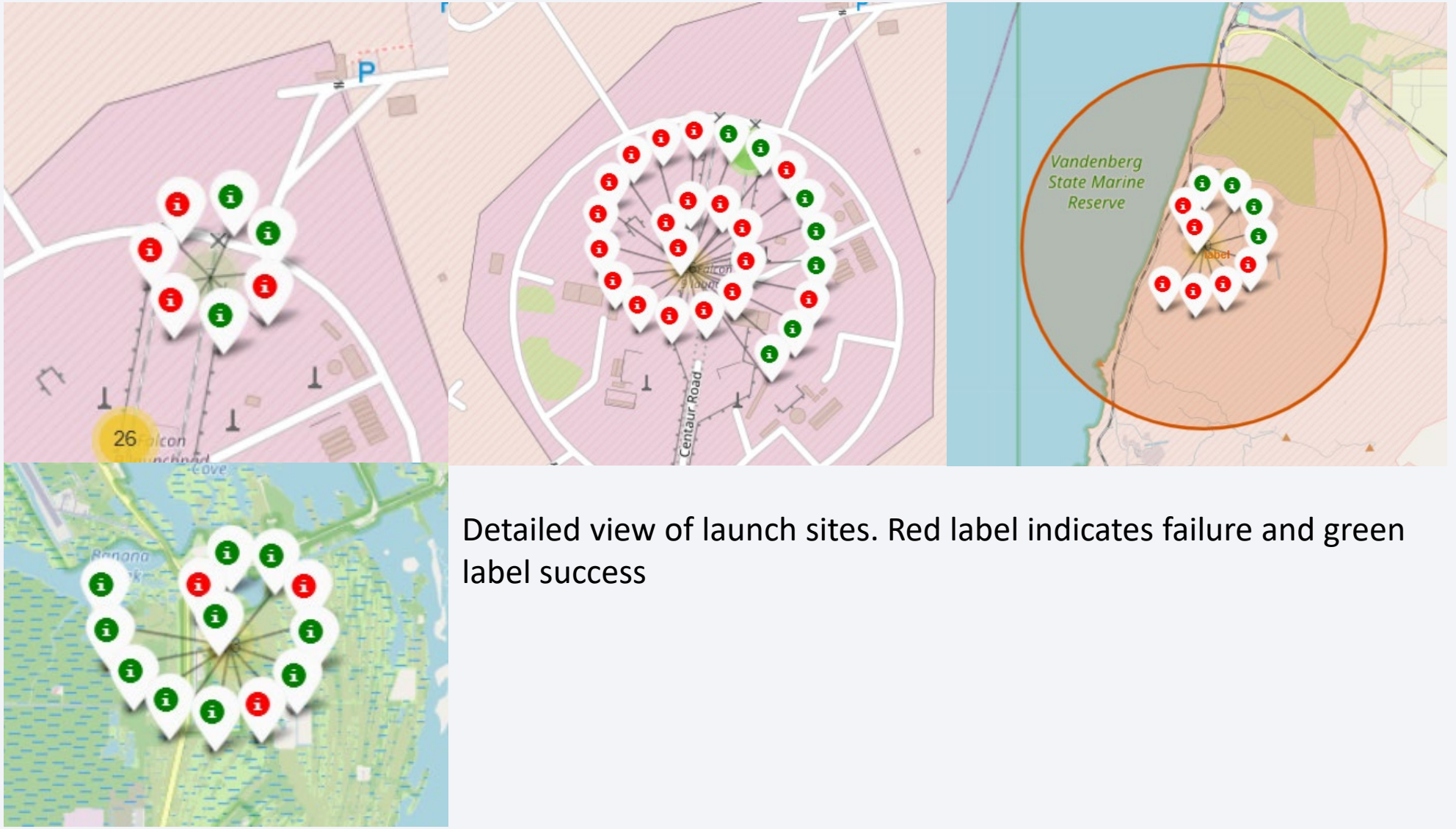


- All Launch Sites are in close proximity to the coast
- All Launch Sites are in restricted areas

Markers of Success/ Failed Landings

Map shows cluster for every launch site

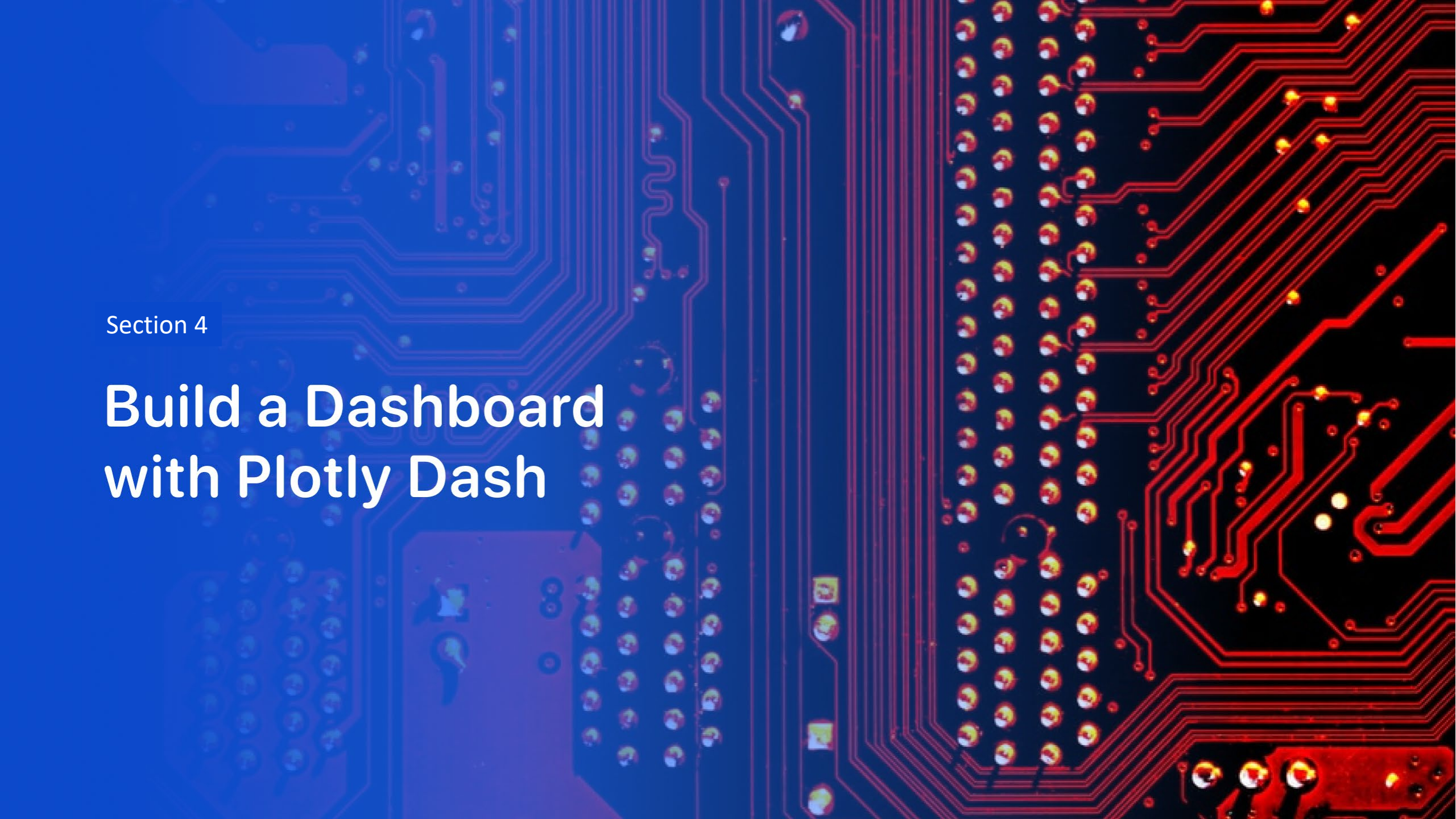
Markers of Success/ Failed Launches



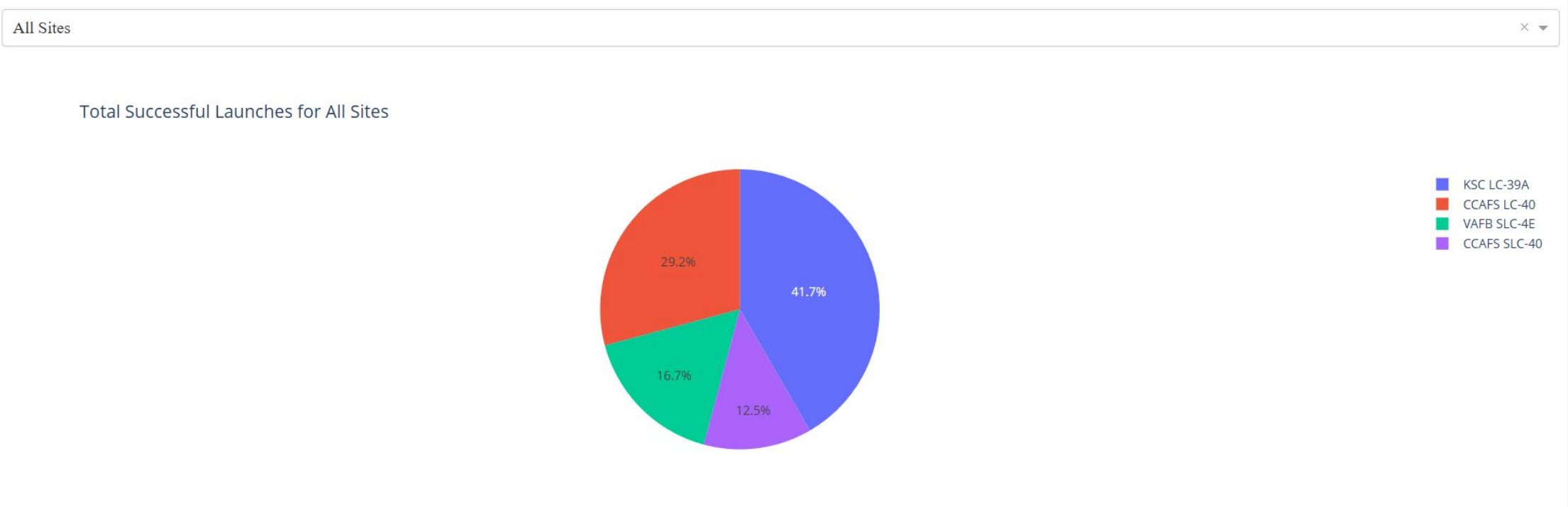
Detailed view of launch sites. Red label indicates failure and green label success

Distance from Launch Site to Proximities

- Roads, Railways an Coastlines are in close proximity while cities are further away
- Roads an Railways are needed for Supply.
- Close to the shore is for safety reasons as well as a large distance to cities or dense populated areas.

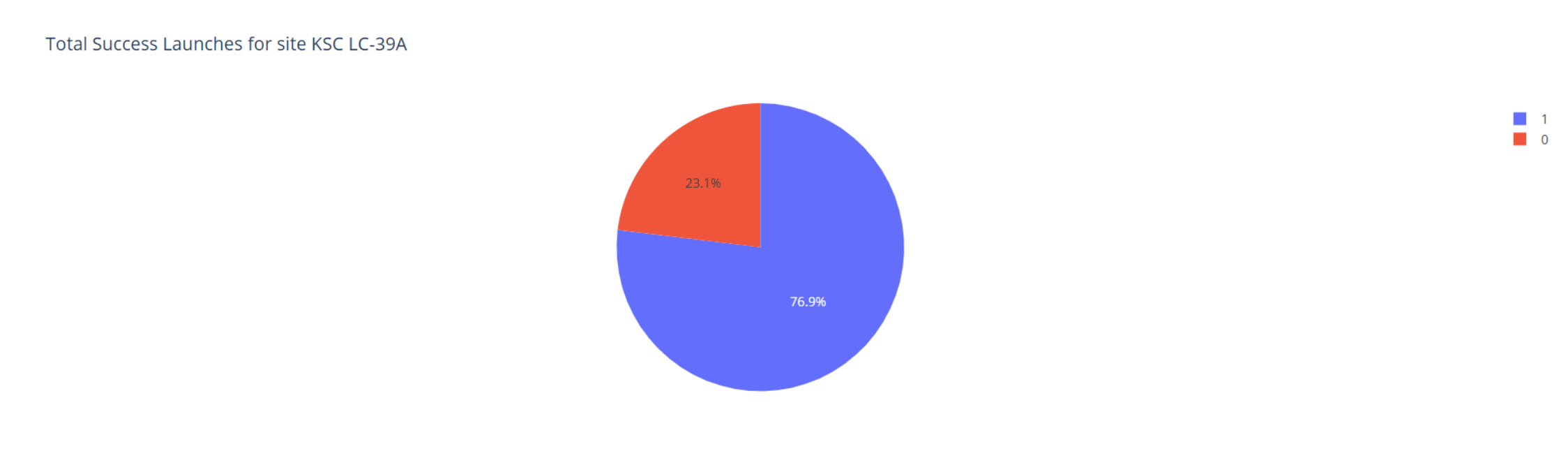

Section 4
Build a Dashboard with Plotly Dash

Total successful launches for All Sites



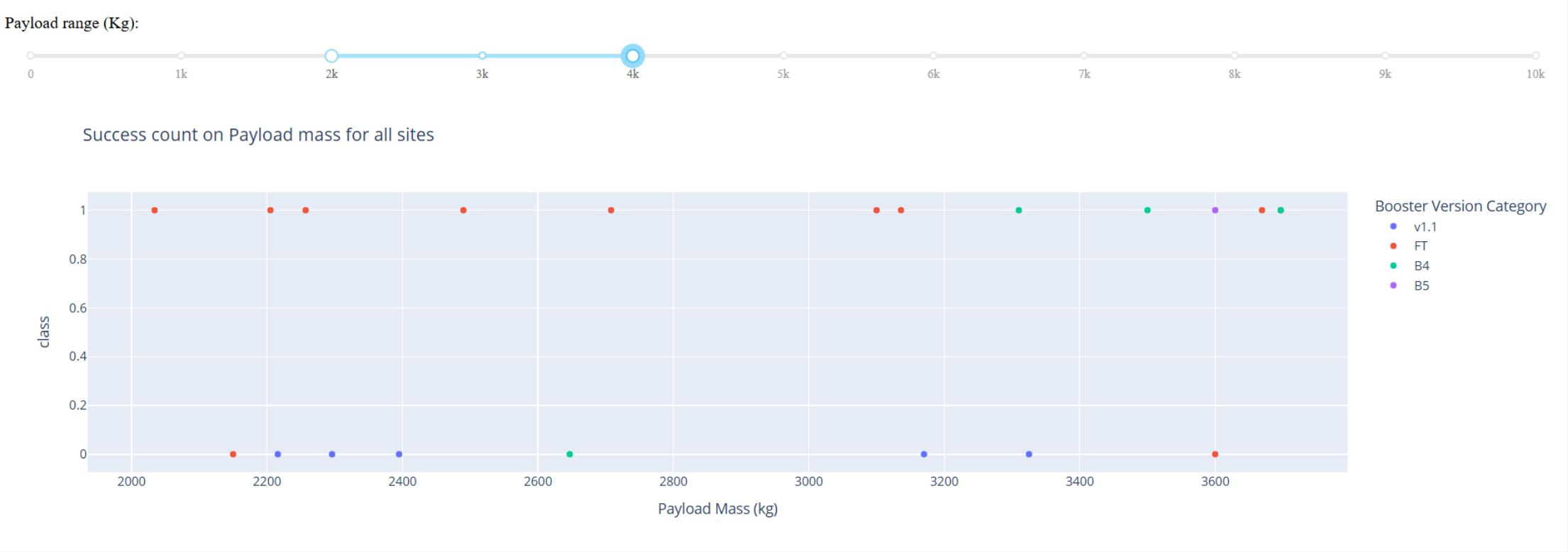
- KSC LC-39A has the highest number of total Successful Launches

Launch site with highest launch success ratio



- KSC LC-39A has the highest launch success ratio

Payload vs. Launch outcome

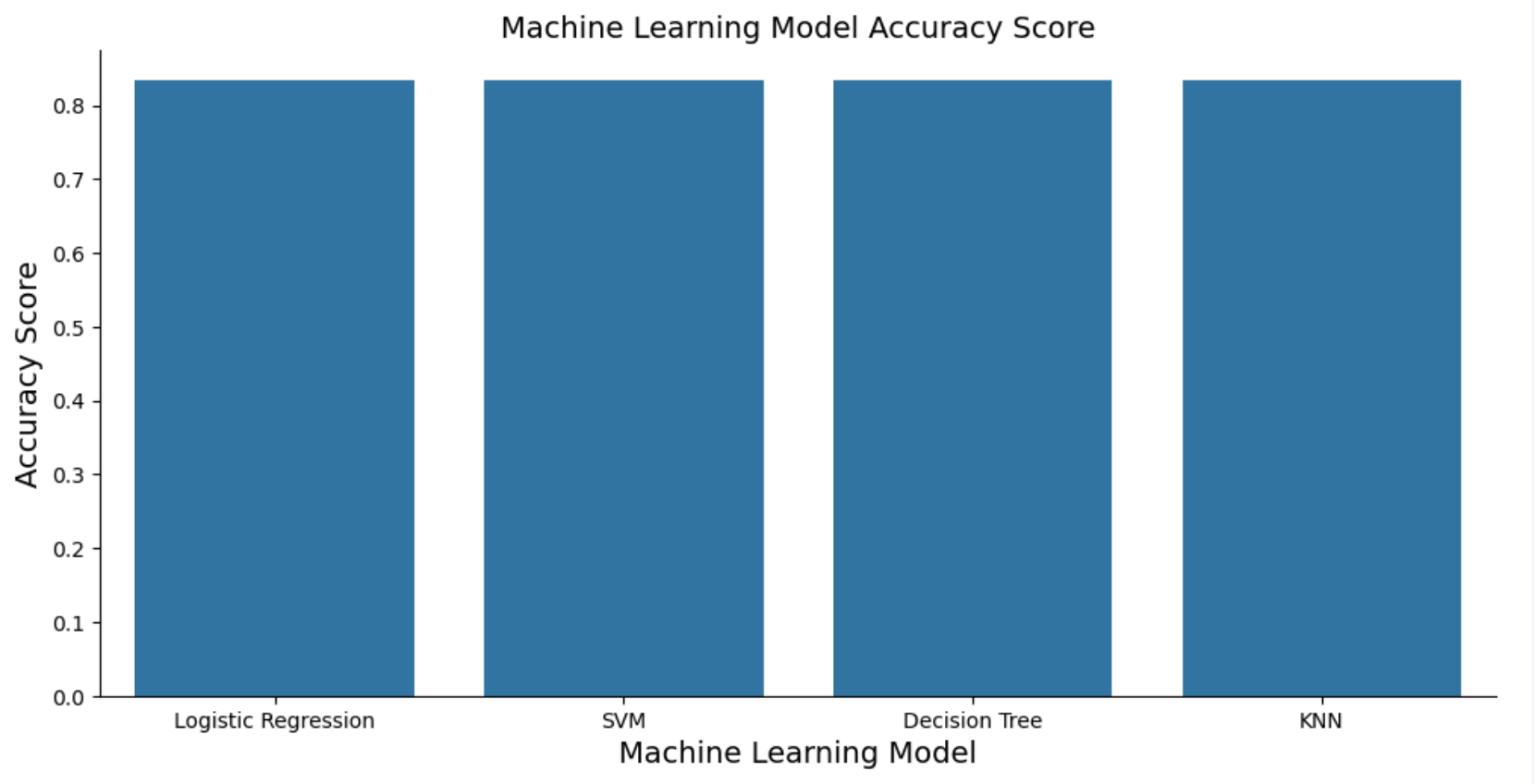


- The Range between 2000 kg and 4000 kg Payload has the highest success rate

Section 5

Predictive Analysis (Classification)

Classification Accuracy



- Accuracy is the same for all models
- All Models have the same Confusion Matrix

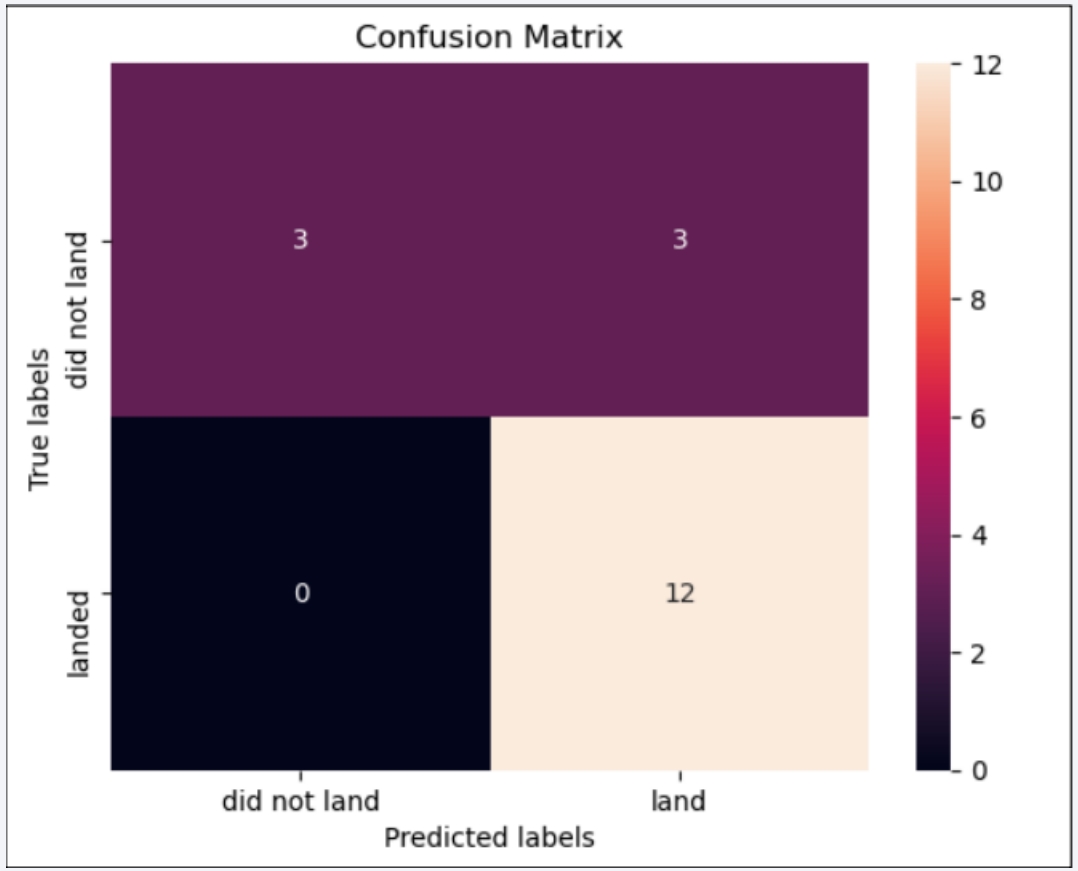
Confusion Matrix

Confusion matrix can be read as follows:

True Negative	False Positive
False Negative	True Positive

Prediction Breakdown:

- 12 True Positives and 3 True Negatives
- 3 False Positives and 0 False Negatives



Conclusions

- SpaceX's record for Falcon 9 first stage landing outcomes has improved
- Trend is towards a higher success rate of positive landing outcome
- All trained models performed equally well and are capable of predicting landing outcomes with a accuracy of 83%.
  - Decision Tree is slower as it needs more time to process the data
  - Recommendation use KNN or Log. Regression as those models are easier to interpret as an SVM.

Appendix

Data collection SpaceX API: https://github.com/JohannesHofmann81/IBM_Data_Science_Capstone/blob/main/jupyter-labs-spacex-data-collection-api.ipynb

Data collection Web scraping:

https://github.com/JohannesHofmann81/IBM_Data_Science_Capstone/blob/main/jupyter-labs-webscraping.ipynb

Data wrangling:

https://github.com/JohannesHofmann81/IBM_Data_Science_Capstone/blob/main/labs-jupyter-spacex-data%20wrangling_jupyterlite.ipynb

EDA with visualization:

https://github.com/JohannesHofmann81/IBM_Data_Science_Capstone/blob/main/edadataviz.ipynb

EDA with SQL:

https://github.com/JohannesHofmann81/IBM_Data_Science_Capstone/blob/main/jupyter-labs-eda-sql-edx_sqllite.ipynb

Interactive Folium Map:

https://github.com/JohannesHofmann81/IBM_Data_Science_Capstone/blob/main/lab_jupyter_launch_site_location%20(2).ipynb

Appendix

Interactive Dashboard:

https://github.com/JohannesHofmann81/IBM_Data_Science_Capstone/blob/main/spacex-dash-app.py

Machine Learning Prediction:

https://github.com/JohannesHofmann81/IBM_Data_Science_Capstone/blob/main/SpaceX_Machine%20Learning%20Prediction_Part_5.ipynb

Thank you!