IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Bin Wang

January 2, 2024

Outline

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

Executive Summary

Summary of methodologies

- Data Collection with API and Web Scraping
- Transform Data with Data Wrangling
- Exploratory Data Analysis with Data Visualization - Exploratory Data Analysis with SQL
- Building a Dashboard with Plotly Dash
- Predictive analysis (Classification)

Summary of all results

- Data analysis results
- Data visuals, interactive dashboards
- Predictive analysis results

Introduction

Project background and context

- With the recent successes in private space travel, space industry is becoming more and more mainstream and accessible to general population. Cost of launch continues to remain a key barrier for new competitors to enter the space race.
- SpaceX advertises Falcon 9 rocket launches on its website, with a cost of 62 million dollars; other providers cost upward of 165 million dollars each, much of the savings is because SpaceX can reuse the first stage. Therefore, if we can determine the cost of a launch by determining if the first stage will land.
- Based on public information and data analysis models, we will predict if SpaceX will reuse the first stage.

Problems you want to find answers

- How do different variables, such as payload mass, launch site, number of flights, and orbits, affect the landing outcome (the success of the first stage landing).
- Correlations between launch sites and success rate.

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - API requests from SpaceX REST API.
  - Web Scraping data from “List of Falcon 9 and Falcon Heavy launches” provided by Wikipedia.
- Perform data wrangling
  - Generate training labels for supervised models by mapping mission outcomes to binary values (0 for unsuccessful, 1 for successful).
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - Identified the optimal classification algorithm (Logistic Regression, SVM, Decision Tree, & KNN) based on test data by Establishing a ‘class’ column, standardizing, transforming the data, and conducting a train/test split.

Data Collection

- Describe how data sets were collected.
  - API requests from SpaceX REST API.
  - Web Scraping data from “List of Falcon 9 and Falcon Heavy launches” provided by Wikipedia.

Data Collection – SpaceX API

GitHub URL:
https://github.com/JoeBloggs-hub/testrepo/blob/main/Week-1%20Data%20Collection%20with%20API.ipynb

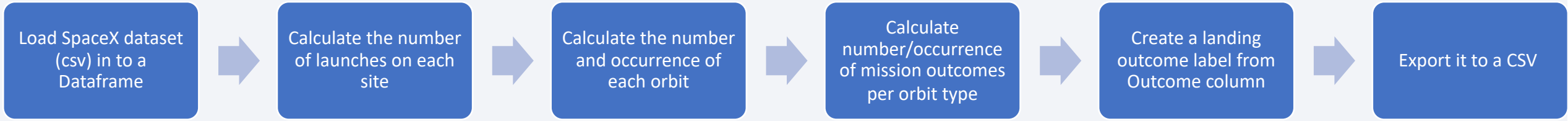
Data Collection - Scraping

GitHub URL:
https://github.com/JoeBloggs-hub/testrepo/blob/main/Week-1%20Data%20Collection%20with%20Web%20Scraping.ipynb

Data Wrangling

Convert outcomes into Training Labels with “1” means the booster successfully landed, while “0” means it was unsuccessful.

- True Ocean means the mission outcome was successfully landed to a specific region of the ocean.
- False Ocean means the mission outcome was unsuccessfully landed to a specific region of the ocean.
- RTLS means the mission outcome was successfully landed to a ground pad.
- False RTLS means the mission outcome was unsuccessfully landed to a ground pad.
- True ASDS means the mission outcome was successfully landed on a drone ship.
- False ASDS means the mission outcome was unsuccessfully landed on a drone ship.



GitHub URL:
https://github.com/JoeBloggs-hub/testrepo/blob/main/Week1%20Data%20wrangling.ipynb

EDA with Data Visualization

- Summarize what charts were plotted and why you used those charts

1. Scatter Plot: show the correlation between:

   Flight Number vs. Payload Mass, Flight Number vs. Launch Site, Payload Mass vs. Launch Site, Orbit Type vs. Success Rate, Flight Number vs. Orbit Type, Payload Mass vs Orbit Type

2. Bar Chat: compare the value, and show the relation between success rate of each orbit shape.

3. Line Chart: show the trend of the success of yearly average lunch.

GitHub URL:
https://github.com/JoeBloggs-hub/testrepo/blob/main/Week%202-eda-dataviz.ipynb.jupyterlite.ipynb

EDA with SQL

- Summarize the SQL queries you performed

1. Displaying the names of the unique launch sites in the space mission.
2. Displaying 5 records where launch sites begin with the string 'CCA'.
3. Displaying the total payload mass carried by boosters launched by NASA (CRS).
4. Displaying average payload mass carried by booster version F9 v1.1.
5. Listing the date when the first successful landing outcome in ground pad was achieved.
6. Listing the names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000.
7. Listing the total number of successful and failure mission outcomes.
8. Listing the names of the booster versions which have carried the maximum payload mass
9. Listing the failed landing outcomes in drone ship, their booster versions and launch site names for the months in year 2015.
10. Ranking 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:
https://github.com/JoeBloggs-hub/testrepo/blob/main/Week%202%20eda-sql-coursera_sqllite.ipynb

Build an Interactive Map with Folium

- Added ‘folium.circle’ and ‘folium.marker’ to highlight circle area with a text label over each launch site.
- Added a ‘MarkerCluster()’ to show launch success (green) and failure (red) markers for each launch site.
- Calculated distances between a launch site to its proximities.

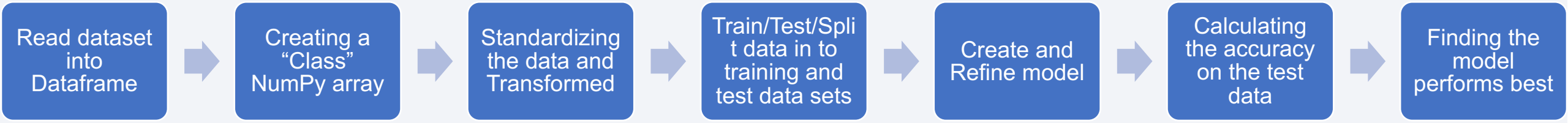
GitHub URL:
https://github.com/JoeBloggs-hub/testrepo/blob/main/Week%203%20launch_site_location.jupyterlite.ipynb

Build a Dashboard with Plotly Dash

- Launch Sites Dropdown List: Added a dropdown list to enable Launch Site selection.
- When a particular set is selected, Added a pie chart to show the total successful launches count for all sites and the Success vs. Failed counts for the site.
- Slider of Payload Mass Range: Added a slider to select Payload range.
- Added a scatter chart to show the correlation between Payload and Launch Success.

GitHub URL:
https://github.com/JoeBloggs-hub/testrepo/blob/main/Week3%20spacex_dash_app.py

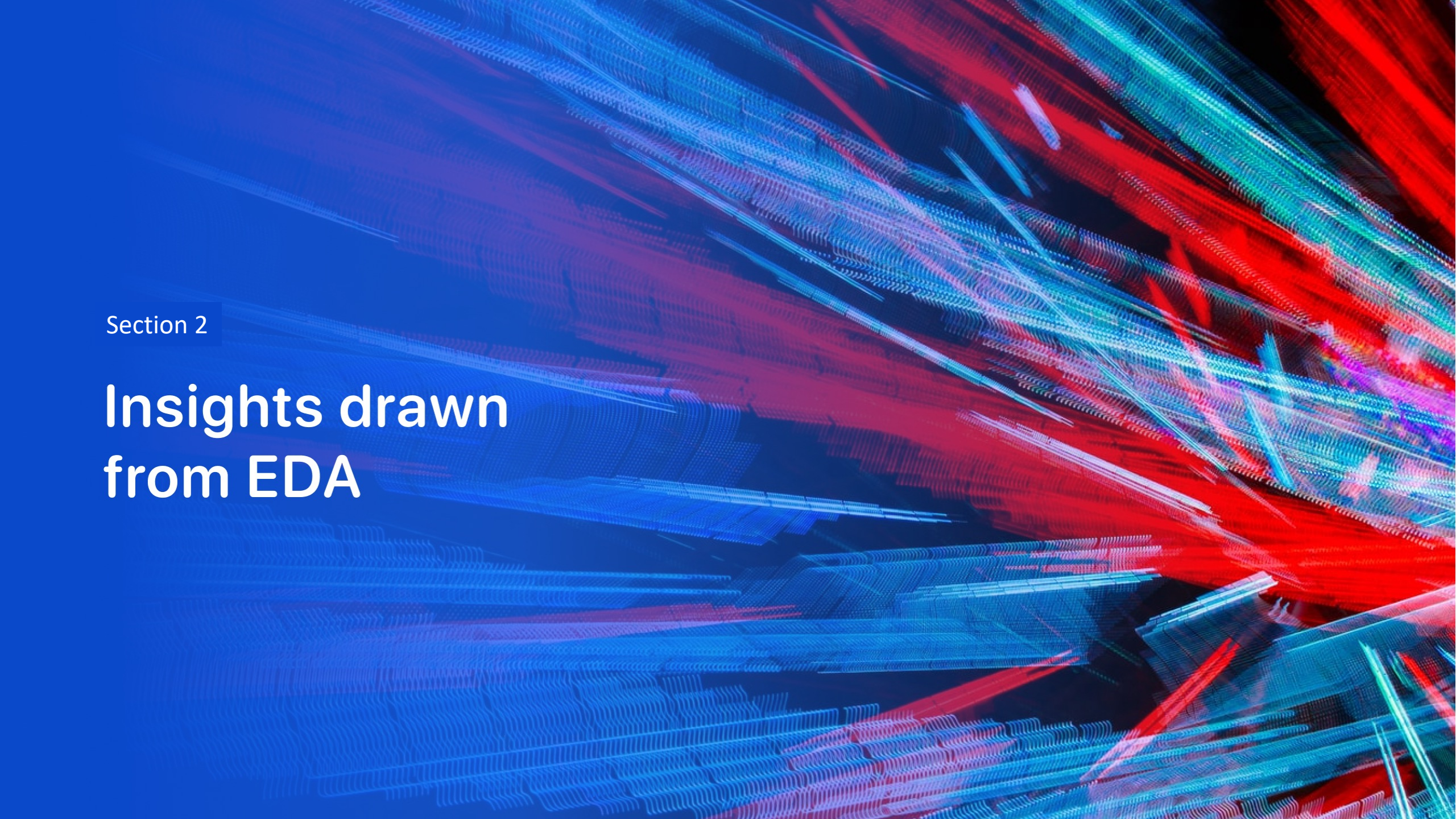
Predictive Analysis (Classification)



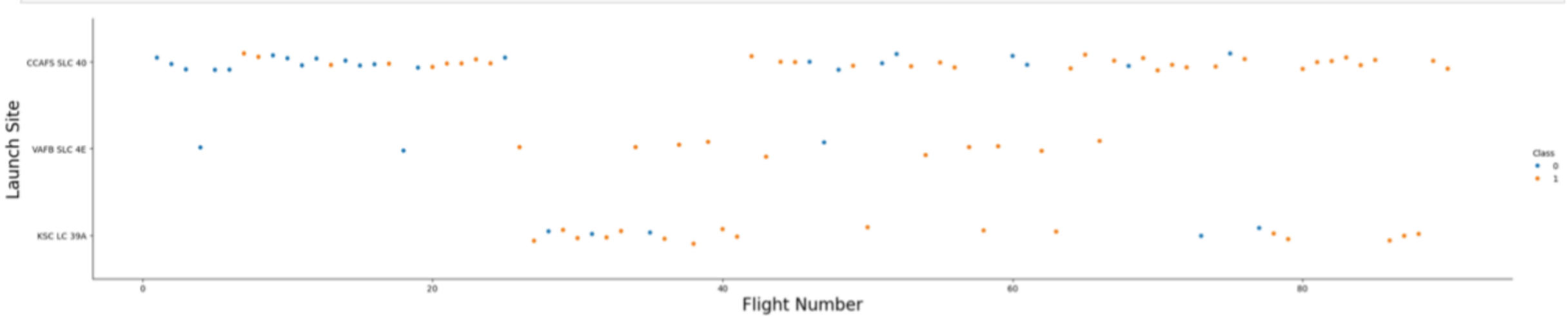
GitHub URL:
https://github.com/JoeBloggs-hub/testrepo/blob/main/Week%204%20Machine_Learning_Prediction_Part_5.jupyterlite.ipynb

Results

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


Section 2
Insights drawn from EDA

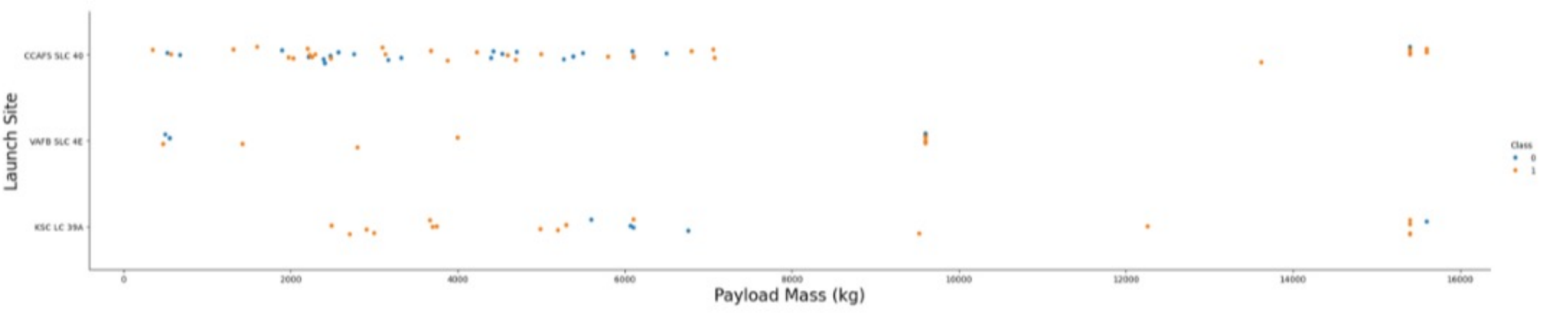
Flight Number vs. Launch Site



•Explanation:

1. As the number of flight increases, the success rate increases.
2. VAFB SLC 4E and KSC LC 39A have higher success rates.

Payload vs. Launch Site



•Explanation:

1. The number of sample is not huge enough, cannot find the specific correlation between Payload Mass and Launch Site.
2. KSC LC 39A under 5500kg has a 100% success rate for payload mass.

Success Rate vs. Orbit Type

- Explanation:

1. Orbits with 100% success rate:

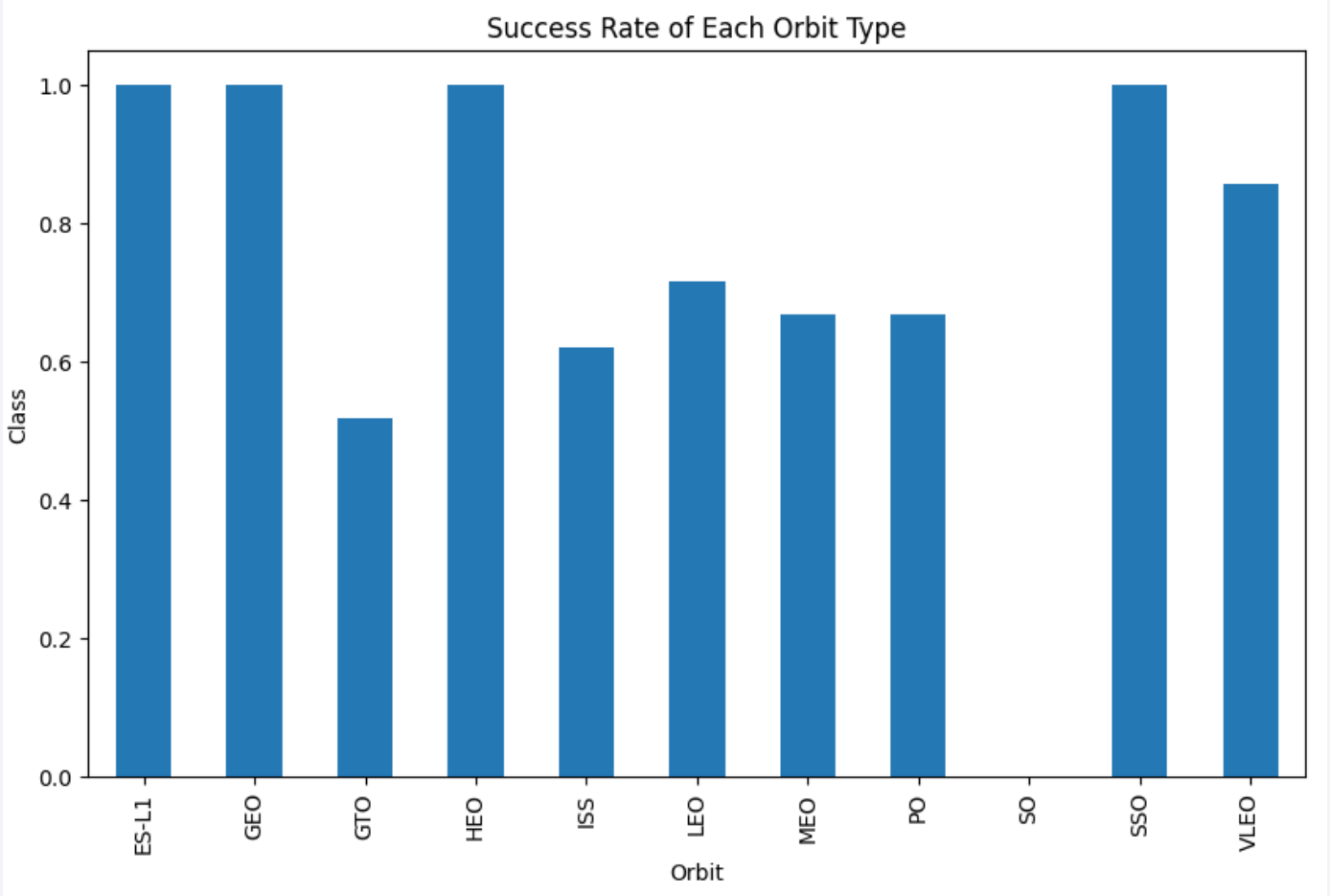
   ES-L1, GEO, HEO, SSO

2. Orbits with lowest success rate:

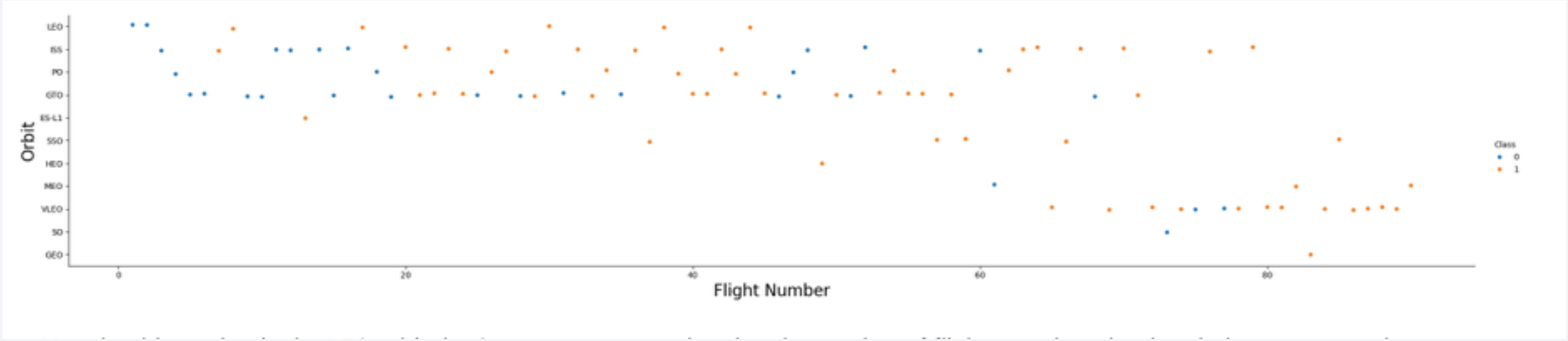
   GTO

3. Orbits with 0% success rate:

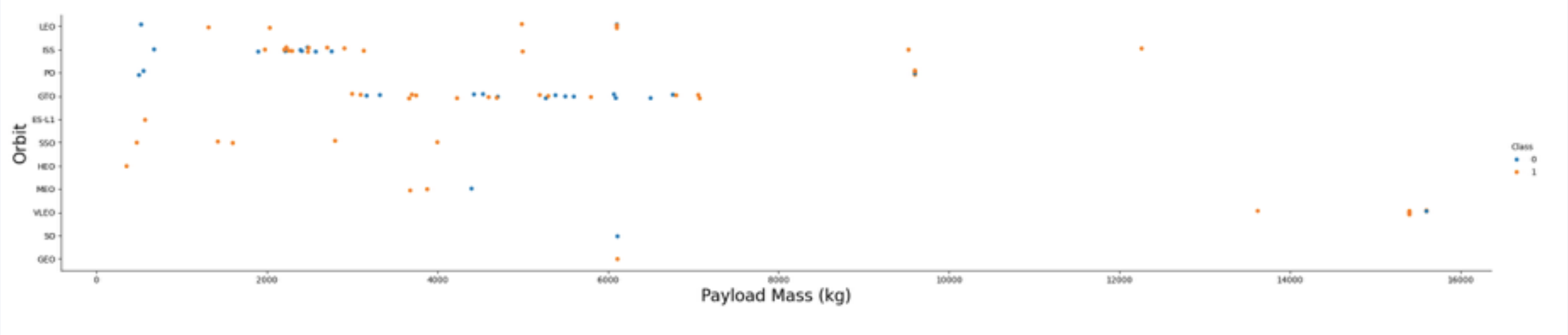
   SO



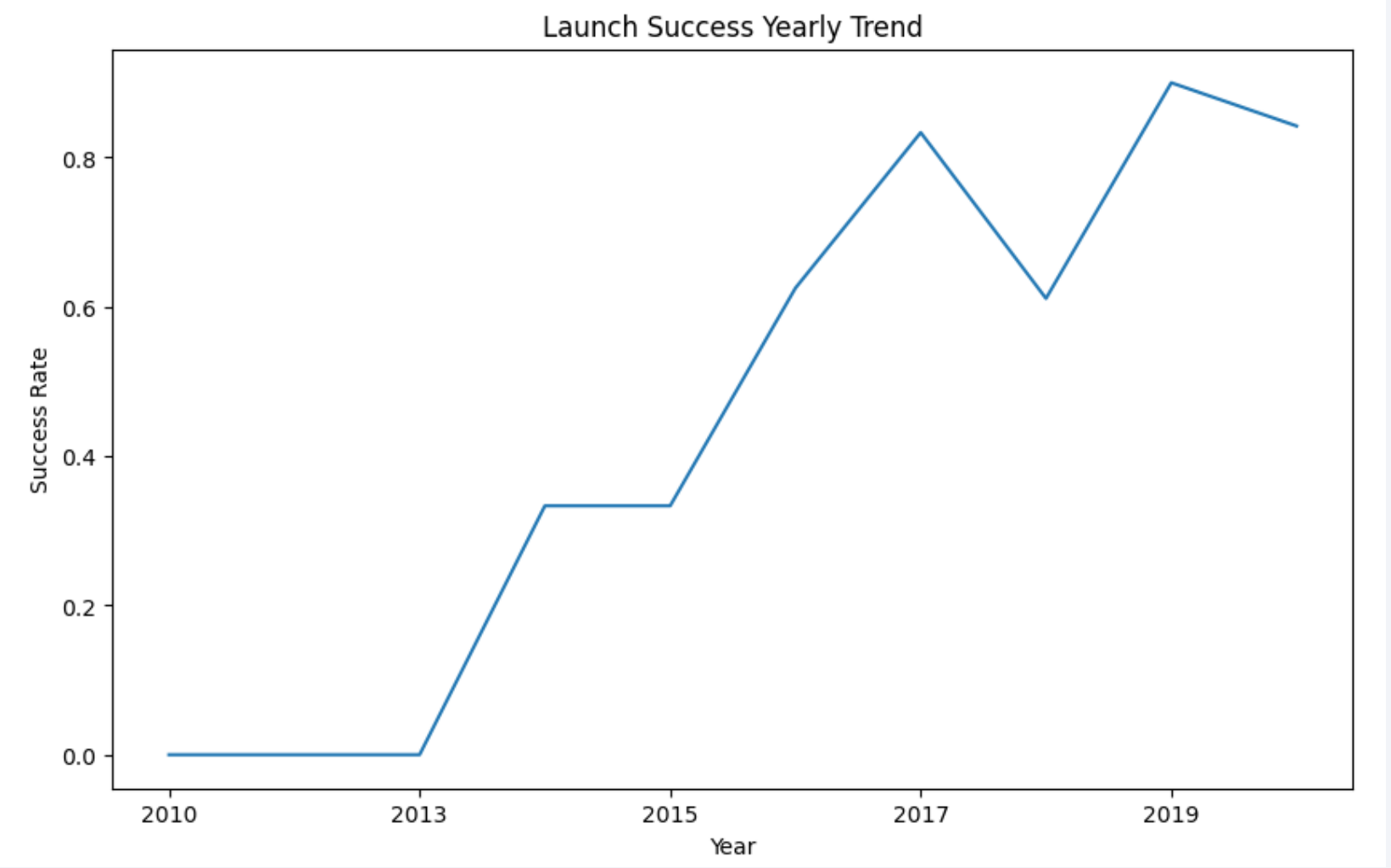
Flight Number vs. Orbit Type



Payload vs. Orbit Type



Launch Success Yearly Trend



All Launch Site Names

- Find the names of the unique launch sites
- Present your query result with a short explanation here

```
%%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

Launch Site Names Begin with 'CCA'

- Find 5 records where launch sites begin with `CCA`
- Present your query result with a short explanation here

```
%%sql

select * from spacextbl where Launch_Site LIKE 'CCA%' limit 5;
```

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

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

Total Payload Mass

- Calculate the total payload carried by boosters from NASA
- Present your query result with a short explanation here

```
%%sql

select sum(PAYLOAD_MASS__KG_) from spacextbl where Customer = 'NASA (CRS)'
```

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

sum(PAYLOAD_MASS__KG_)
45596

Average Payload Mass by F9 v1.1

- Calculate the average payload mass carried by booster version F9 v1.1
- Present your query result with a short explanation here

```
%%sql

select avg(PAYLOAD_MASS__KG_) from spacextbl where Booster_Version LIKE 'F9 v1.1';
```

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

avg(PAYLOAD_MASS__KG_)
2928.4

First Successful Ground Landing Date

- Find the dates of the first successful landing outcome on ground pad
- Present your query result with a short explanation here

```
%%sql

select min(Date) as first_sucess from spacextbl where Landing_Outcome = 'Success (ground pad)';
```

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

first_sucess
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

- Present your query result with a short explanation here

```
%%sql

select Booster_Version from spacextbl where (PAYLOAD_MASS__KG_> 4000 and PAYLOAD_MASS__KG_ < 6000)
and (Landing_Outcome = 'Success (drone ship)');
```

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

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

Total Number of Successful and Failure Mission Outcomes

- Calculate the total number of successful and failure mission outcomes
- Present your query result with a short explanation here

```
%%sql

select Mission_Outcome, count(Mission_Outcome) as counts from spacextbl group by Mission_Outcome;
```

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

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

Boosters Carried Maximum Payload

- List the names of the booster which have carried the maximum payload mass
- Present your query result with a short explanation here

```
%%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.
```

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

- Present your query result with a short explanation here

```
%%sql

select Booster_Version, Launch_Site, Landing_Outcome from spacextbl where Landing_Outcome = 'Failure (drone ship)' and year(date)
```

```
 * sqlite:///my_data1.db
(sqlite3.OperationalError) no such function: year
[SQL: select Booster_Version, Launch_Site, Landing_Outcome from spacextbl where Landing_Outcome = 'Failure (drone ship)' and year
(date) = '2015']
(Background on this error at: http://sqlalche.me/e/e3q8)
```

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
- Present your query result with a short explanation here

```
%%sql

select Landing_Outcome, count(*) as LandingCounts from spacextbl where Date between '2010-06-04' and '2017-03-20'
group by Landing_Outcome
order by count(*) desc;
```

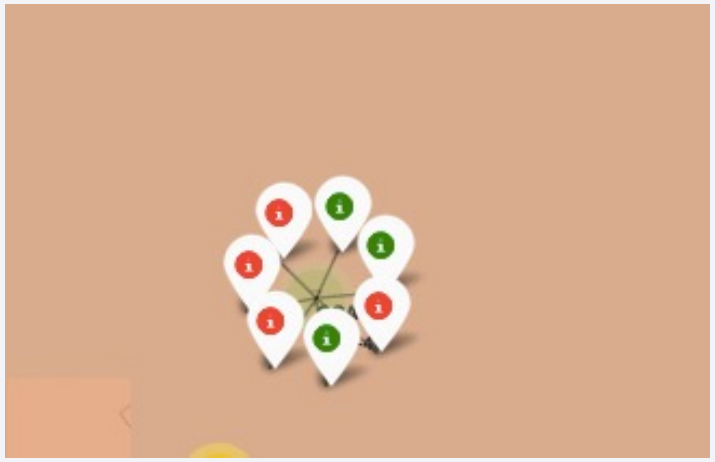
```
 * sqlite:///my_data1.db
Done.
```

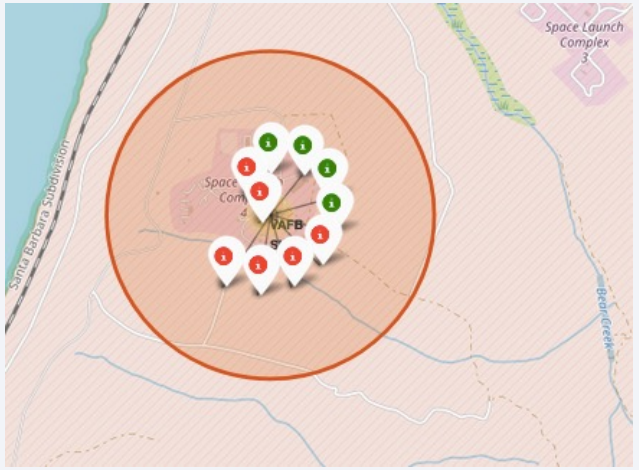
Landing_Outcome	LandingCounts
No attempt	10
Success (drone ship)	5
Failure (drone ship)	5
Success (ground pad)	3
Controlled (ocean)	3
Uncontrolled (ocean)	2
Failure (parachute)	2
Precluded (drone ship)	1

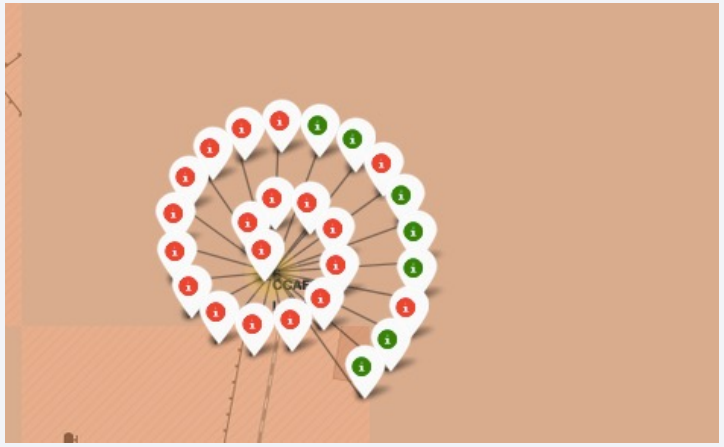

Section 3
Launch Sites
Proximities Analysis

SpaceX Falcon9 - Launch Sites Map

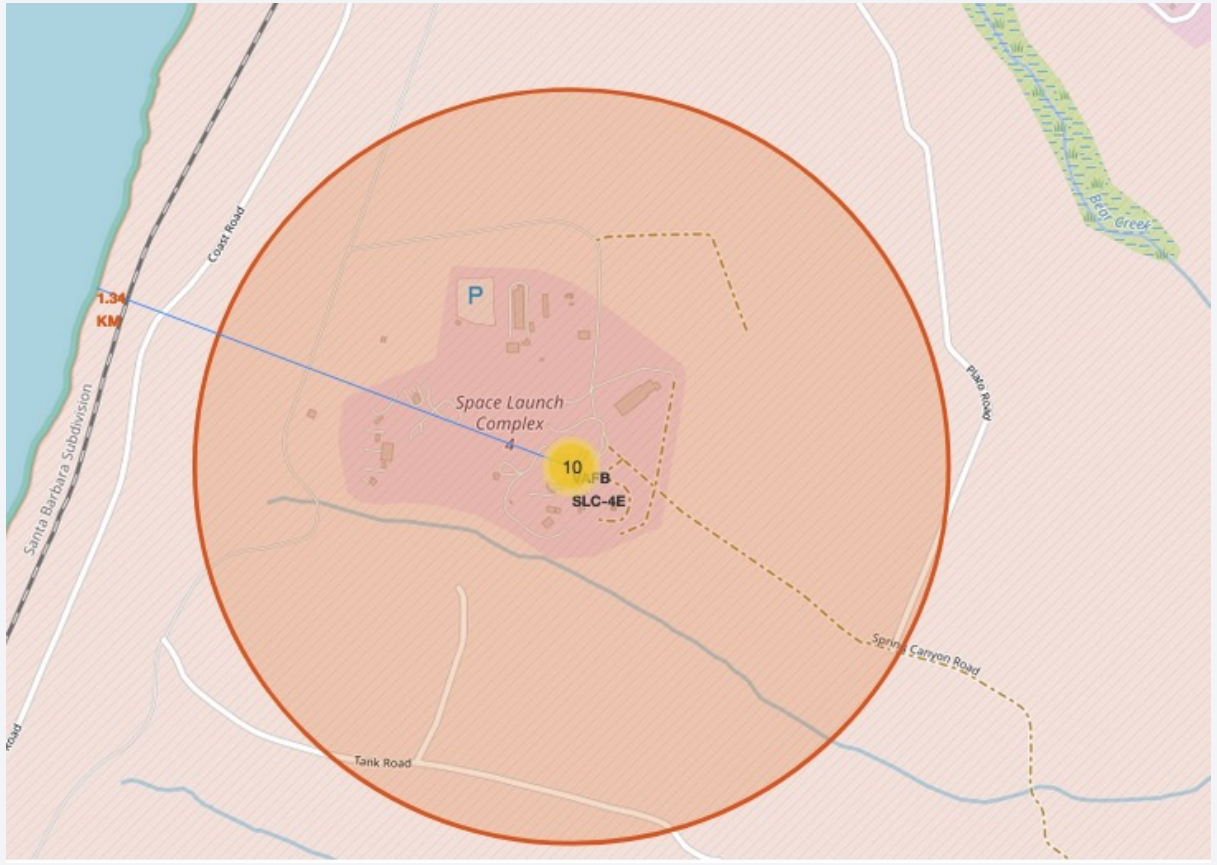
SpaceX Falon9 – Color-labeled Launch Records

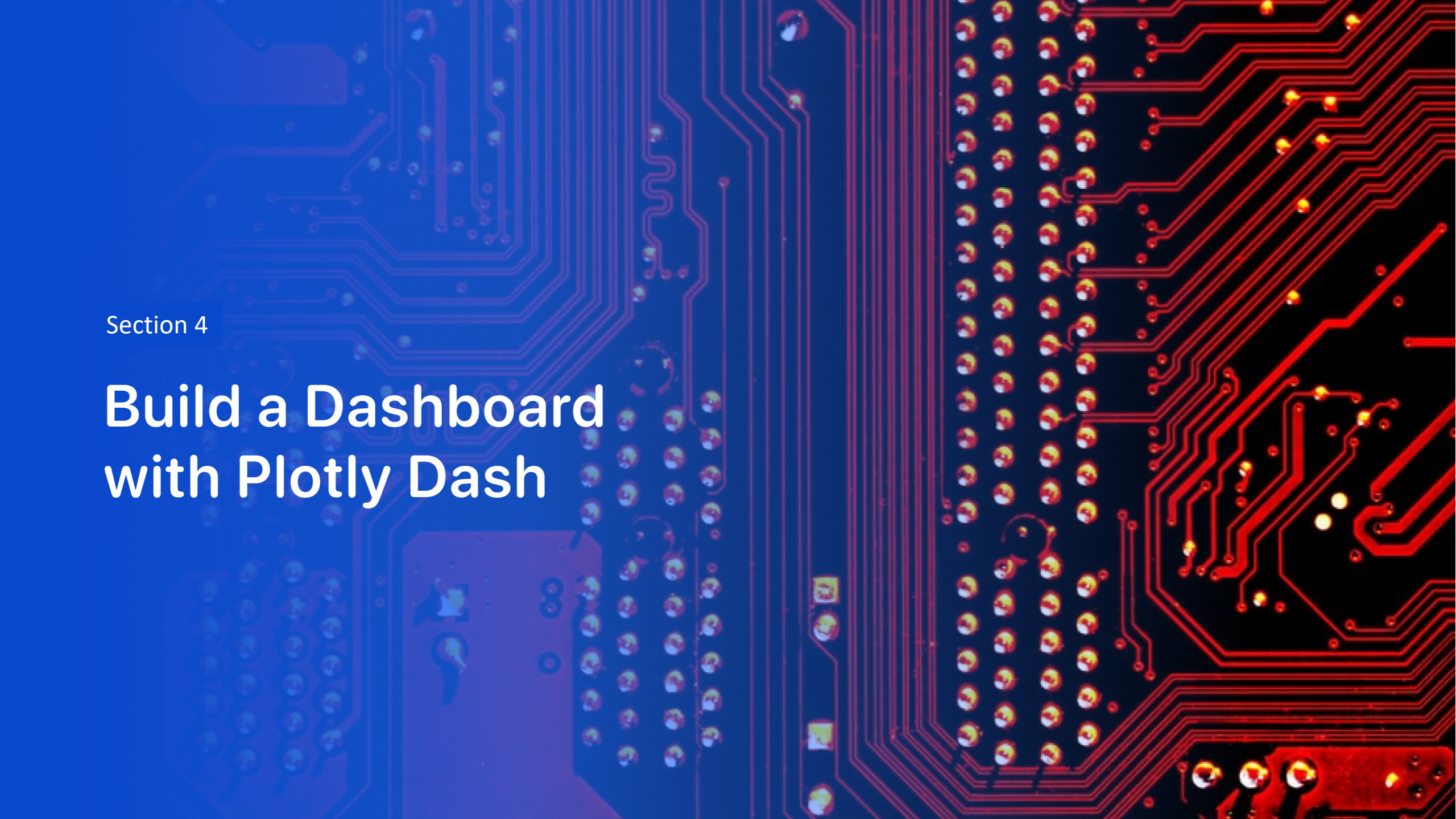




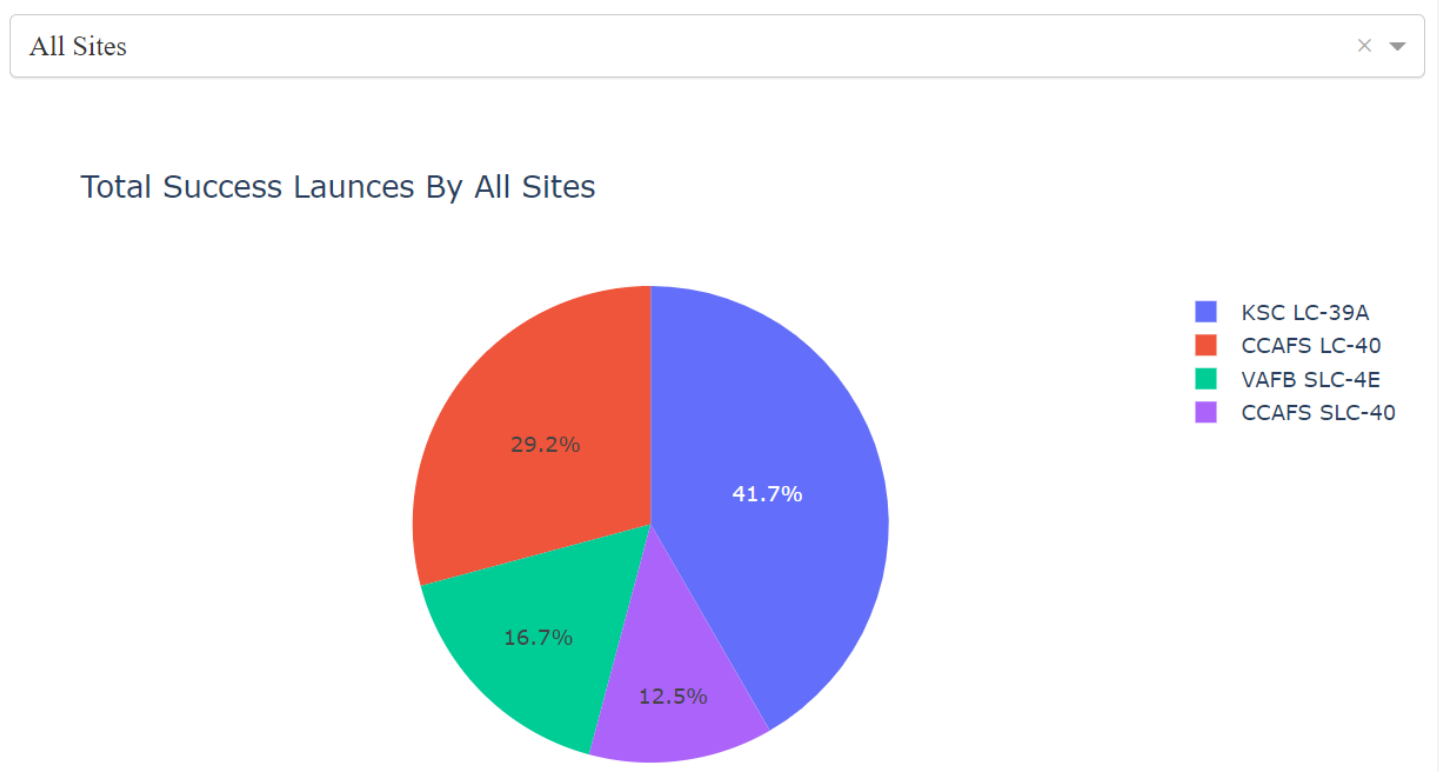


SpaceX Falcon9 – Launch Site to proximity Distance Map

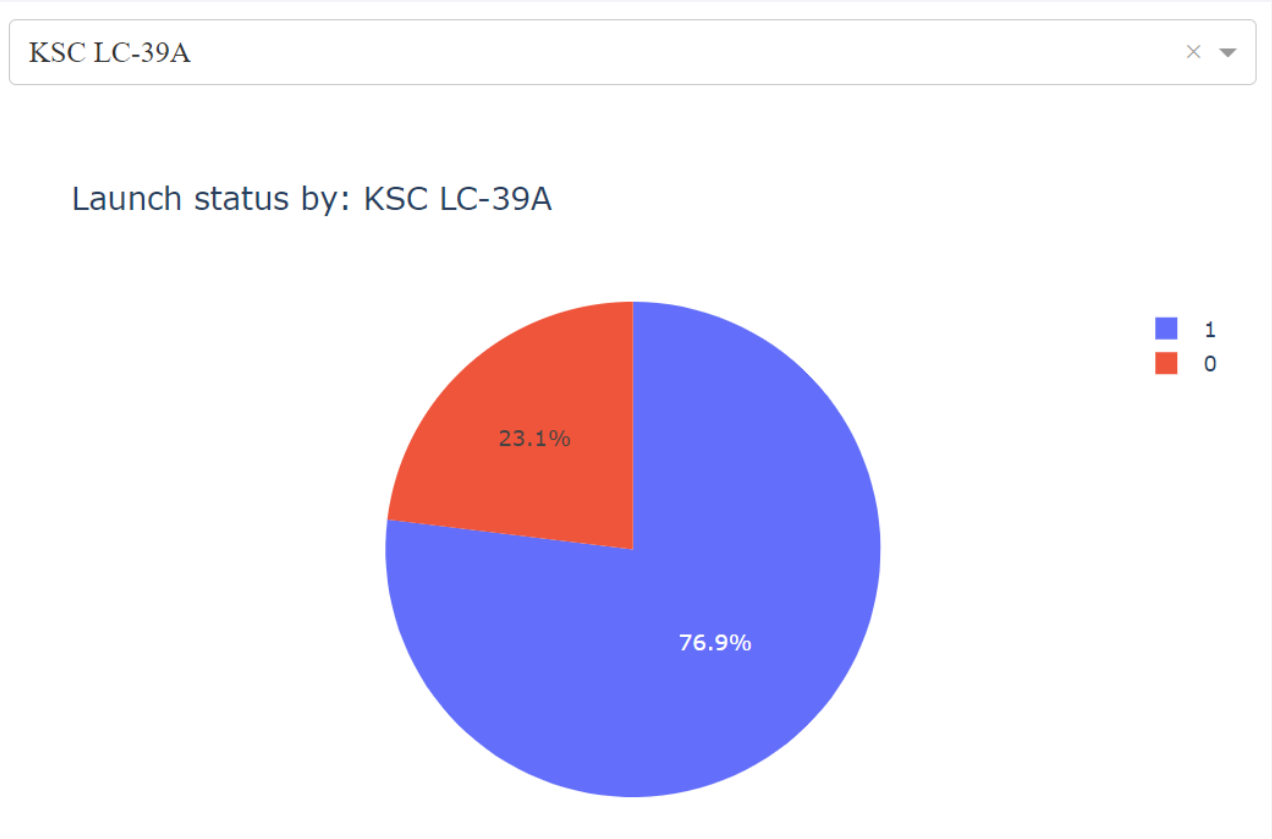



Section 4
Build a Dashboard with Plotly Dash

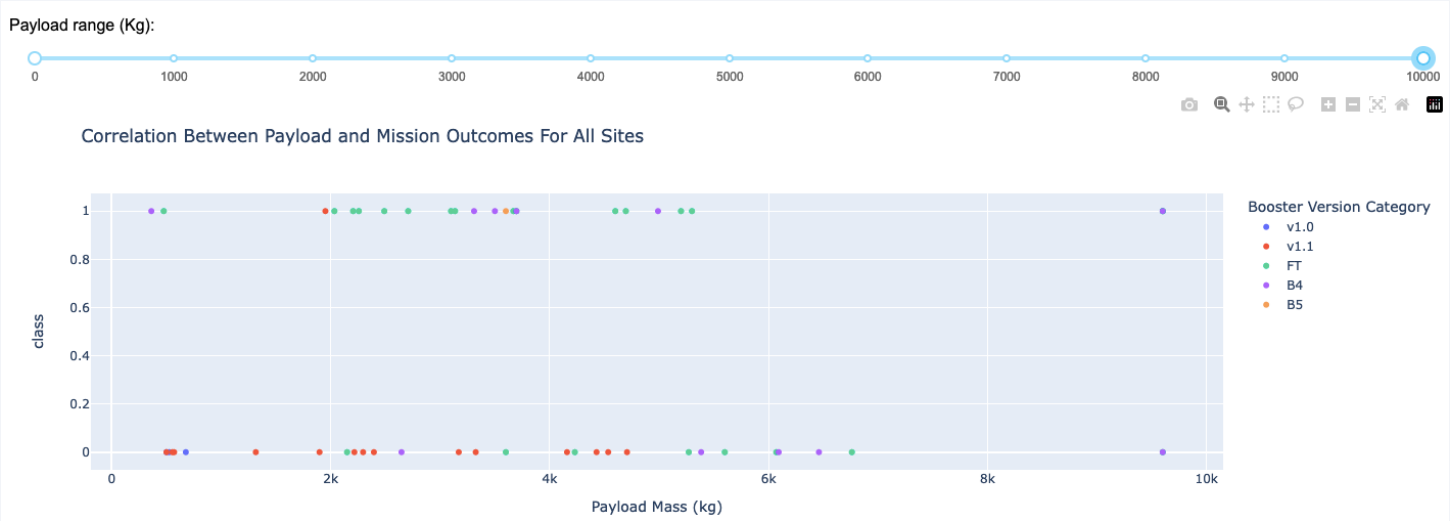
Launch Success Counts For All Sites

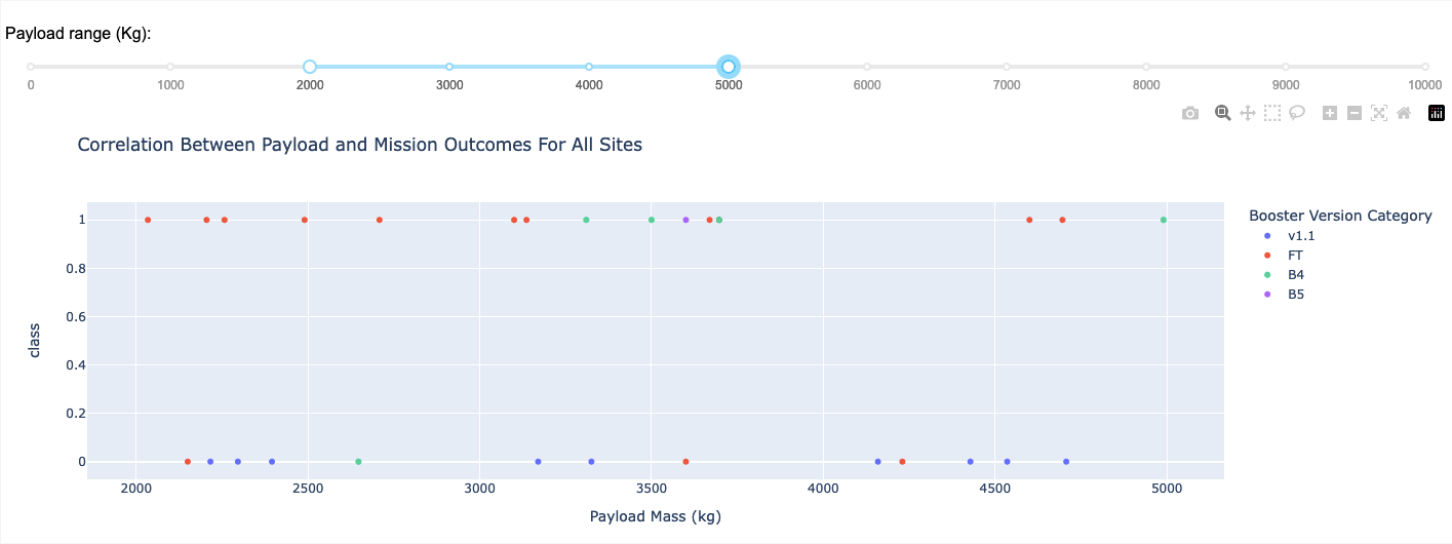


Launch Site with Highest Launch Success Ratio



Payload Mass vs. Launch Outcome for all sites





Section 5

Predictive Analysis (Classification)

Classification Accuracy

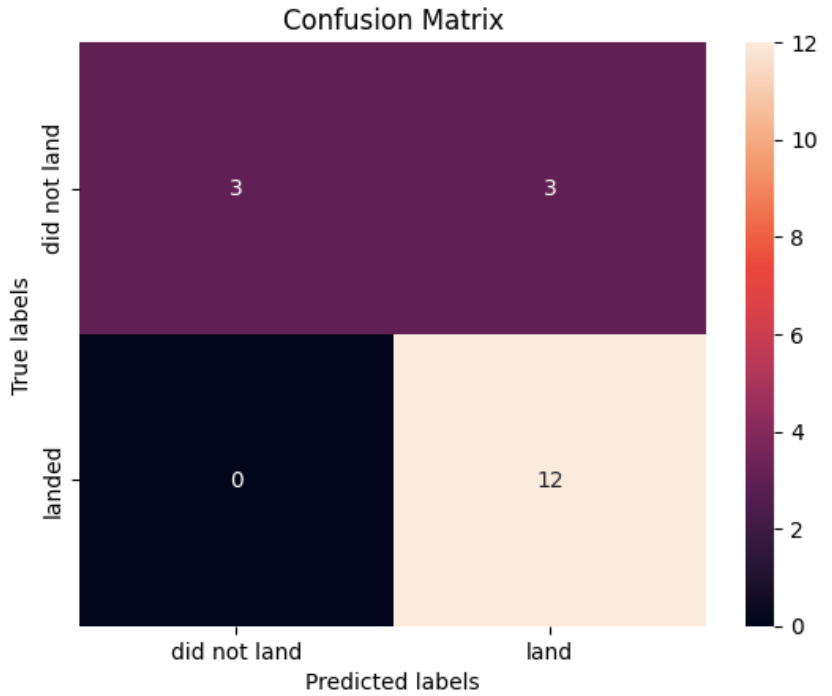
- Visualize the built model accuracy for all built classification models, in a bar chart

- Find which model has the highest classification accuracy

Confusion Matrix

lite.ipynb ets look at the confusion matrix:

```
[17]: yhat=logreg_cv.predict(X_test)
      plot_confusion_matrix(Y_test,yhat)
```



Conclusions

- Point 1: The Decision Tree model stands out as the optimal algorithm for this dataset.
- Point 2: As the number of flights increases, the initial stage is more likely to land successfully.
- Point 3: Launch Site 'KSC LC-39A' boasts the highest success rate, while Launch Site 'CCAFS SLC-40' has the lowest. Orbits ES-L1, GEO, HEO, and SSO exhibit the highest success rates, with GTO having the lowest.

Appendix

- Include any relevant assets like Python code snippets, SQL queries, charts, Notebook outputs, or data sets that you may have created during this project

Thank you!