IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

DARSHAN BENNUR
August 16,2022

Outline

- **EXECUTIVE SUMMARY**
- **INTRODUCTION**
- **METHODOLOGY**
- **RESULTS**
- **CONCLUSION**
- **APPENDIX**

Executive Summary

- Summary of methodologies
  - Data Collection through API
  - Data Collection with Web Scraping
  - Data Wrangling
  - Exploratory Data Analysis with SQL
  - Exploratory Data Analysis with Data Visualization
  - Interactive Visual Analytics with Folium
  - Machine Learning Prediction
- Summary of all results
  - Exploratory Data Analysis result
  - Interactive analytics in screenshots
  - Predictive Analytics result

Introduction

- Project background and context

  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 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. This information can be used if an alternate company wants to bid against spaceX for a rocket launch. This 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 factors determine if the rocket will land successfully?
  - What is the effect of each relationship of rockets will land successfully?
  - What are the conditions needs to be in place to ensure a successful landing program.

Section
1

Methodology

Methodology

Executive Summary

- Data collection methodology - Data was collected using SpaceX API and web scraping from Wikipedia.
- Perform data wrangling - One-hot encoding was applied to categorical features
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification model - How to build, tune, evaluate classification models

Data Collection

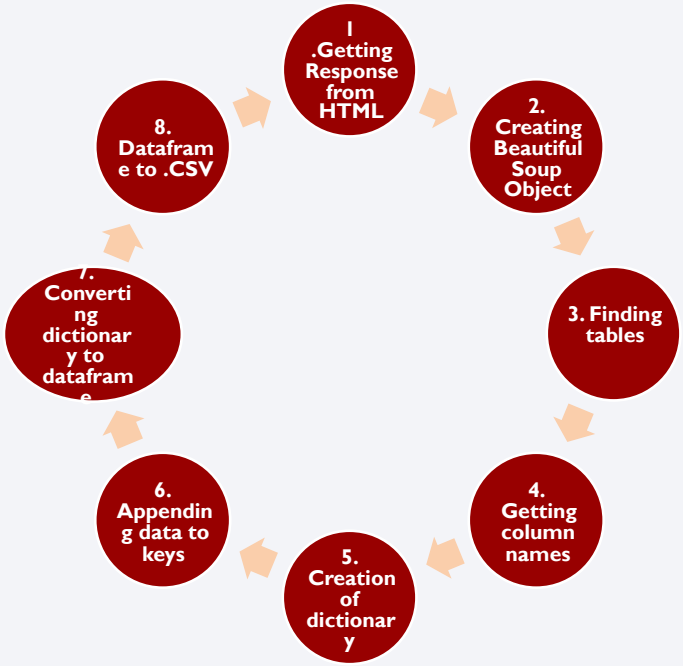
- The data was collected using various methods
  - Data collection was done using get request to the SpaceX API.
  - Next, we decoded the response content as a Json using .json() function call and turn it into a pandas dataframe using .json_normalize().
  - We then cleaned the data, checked for missing values and fill in missing values where necessary.
  - In addition, we performed web scraping from Wikipedia for Falcon 9 launch records with BeautifulSoup.
  - The objective was to extract the launch records as HTML table, parse the table and convert it to a pandas dataframe for future analysis.

Data Collection – SpaceX API

1. Getting Response from API
2. Converting response to a .json file
3. Apply Custom functions to clean data
4. Assign list to dictionary then dataframe
5. Filter dataframe and export as .csv file

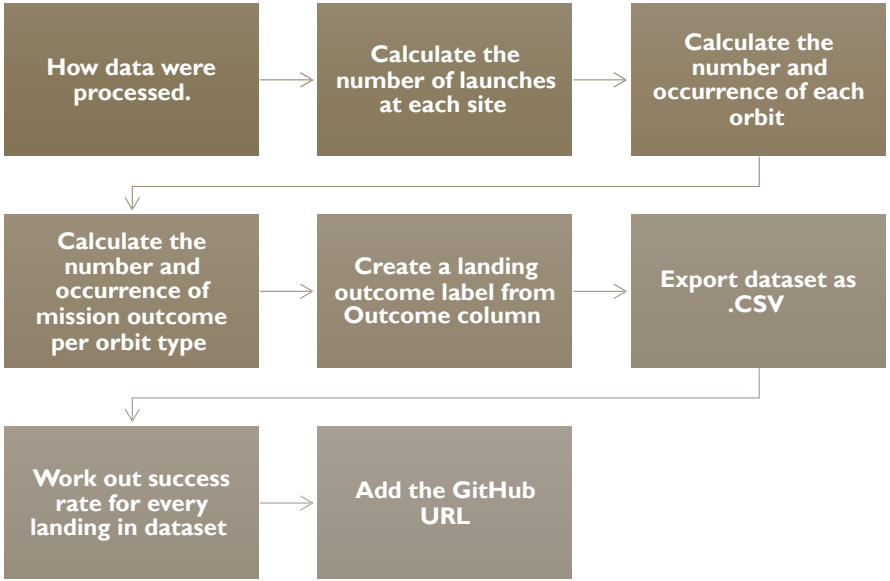
GitHub URL Link:
https://github.com/darshanbennur9/SpaceX_Capstone_Project/blob/e249f78c50933a3118a77e42d3a345b6617fb1e3/jupyter-labs-spacex-data-collection-api_DB.ipynb

Data Collection - Scraping



Github URL Link:
https://github.com/darshanbennur9/SpaceX_Capstone_Project/blob/e249f78c50933a3118a77e42d3a345b6617fb1e3/jupyter-labs-webscraping_DB.ipynb

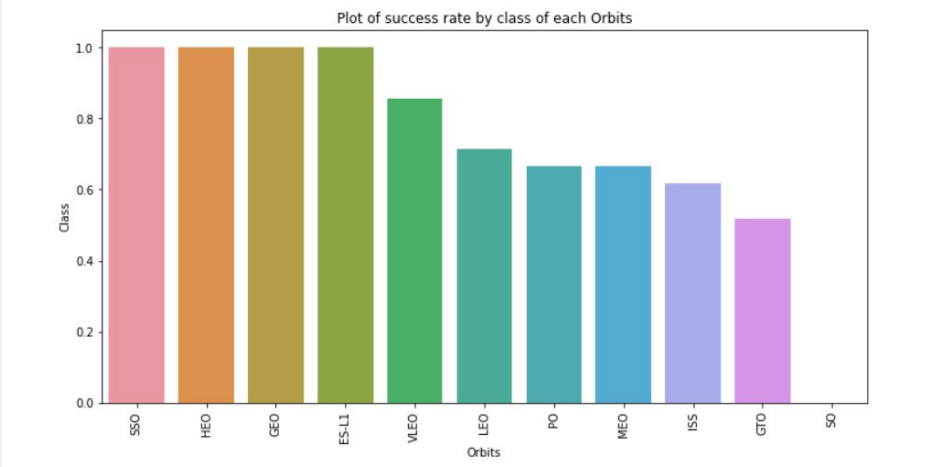
DATA WRANGLING

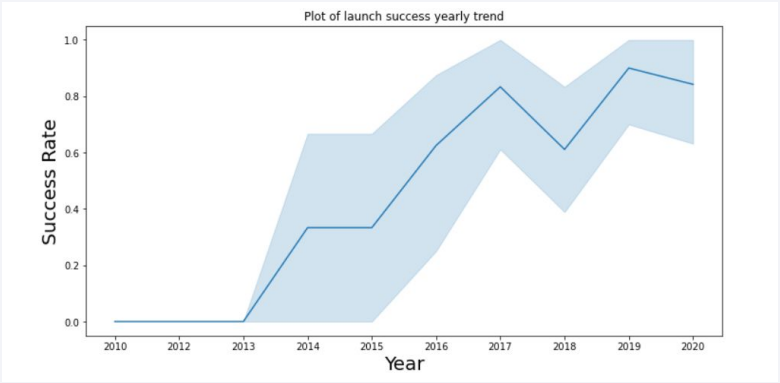


GitHub URL Link:
https://github.com/darshanbennur9/SpaceX_Capstone_Project/blob/e249f78c50933a3118a77e42d3a345b6617fb1e3/labs-jupyter-spacex-Data%20wrangling_DB.ipynb

EDA with Data Visualization

- We explored the data by visualizing the relationship between flight number and launch Site, payload and launch site, success rate of each orbit type, flight number and orbit type, the launch success yearly trend.





GitHub URL Link:
https://github.com/darshanbennur9/SpaceX_Capstone_Project/blob/e249f78c50933a3118a77e42d3a345b6617fb1e3/jupyter-labs-eda-dataviz_DB.ipynb

EDA WITH VISUALIZATION

SUMMARY OF THE SQL QUERIES PERFORMED

- The names of unique launch sites in the space mission.
- The total payload mass carried by boosters launched by NASA (CRS)
- The average payload mass carried by booster version F9 v1.1
- The total number of successful and failure mission outcomes
- The failed landing outcomes in drone ship, their booster version and launch site names.

GitHub URL Link:
https://github.com/darshanbennur9/SpaceX_Capstone_Project/blob/e249f78c50933a3118a77e42d3a345b6617fb1e3/jupyter-labs-eda-sql-coursera_sqllite_DB.ipynb

Build an Interactive Map with Folium

- We marked all launch sites, and added map objects such as markers, circles, lines to mark the success or failure of launches for each site on the folium map.
- We assigned the feature launch outcomes (failure or success) to class 0 and 1.i.e., 0 for failure, and 1 for success.
- Using the color-labeled marker clusters, we identified which launch sites have relatively high success rate.
- We calculated the distances between a launch site to its proximities. We answered some question for instance:
  - Are launch sites near railways, highways and coastlines.
  - Do launch sites keep certain distance away from cities.

GitHub URL Link:
https://github.com/darshanbennur9/SpaceX_Capstone_Project/blob/e249f78c50933a3118a77e42d3a345b6617fb1e3/lab_jupyter_launch_site_location_DB.ipynb

Build a Dashboard with Plotly Dash

- We built an interactive dashboard with Plotly dash
- We plotted pie charts showing the total launches by a certain sites
- We plotted scatter graph showing the relationship with Outcome and Payload Mass (Kg) for the different booster version.

GitHub URL Link:
https://github.com/darshanbennur9/SpaceX_Capstone_Project/blob/41e7b34bdfe3985bd91f64551daa9dcf4f6ae62a/lab_jupyter_launch_site_location_DB.ipynb

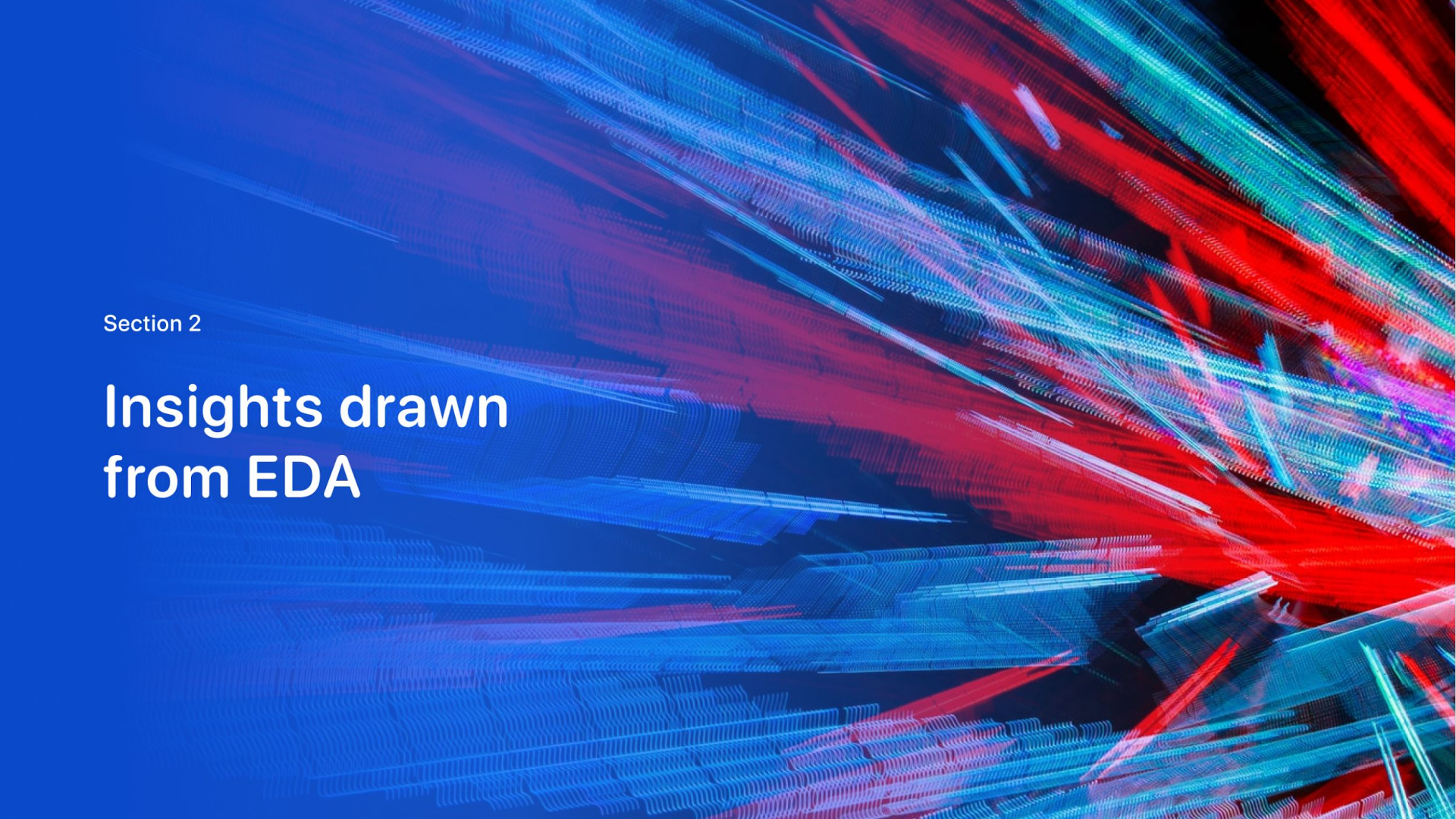
Predictive Analysis (Classification)

- We loaded the data using numpy and pandas, transformed the data, split our data into training and testing.
- We built different machine learning models and tune different hyperparameters using GridSearchCV.
- We used accuracy as the metric for our model, improved the model using feature engineering and algorithm tuning.
- We found the best performing classification model.

GitHub URL Link:
https://github.com/darshanbennur9/SpaceX_Capstone_Project/blob/e249f78c50933a3118a77e42d3a345b6617fb1e3/SpaceX_Machine%20Learning%20Prediction_Part_5_DB.ipynb

Results

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

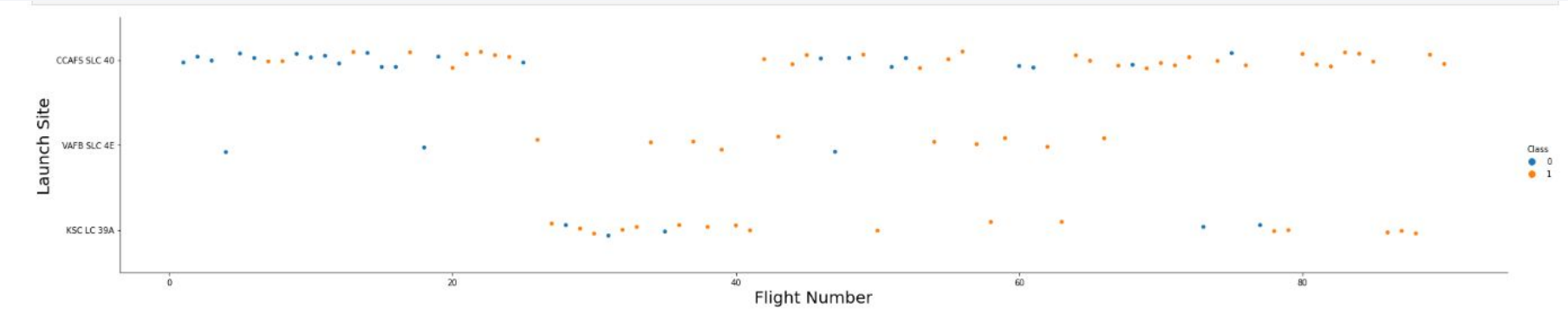


Section 2

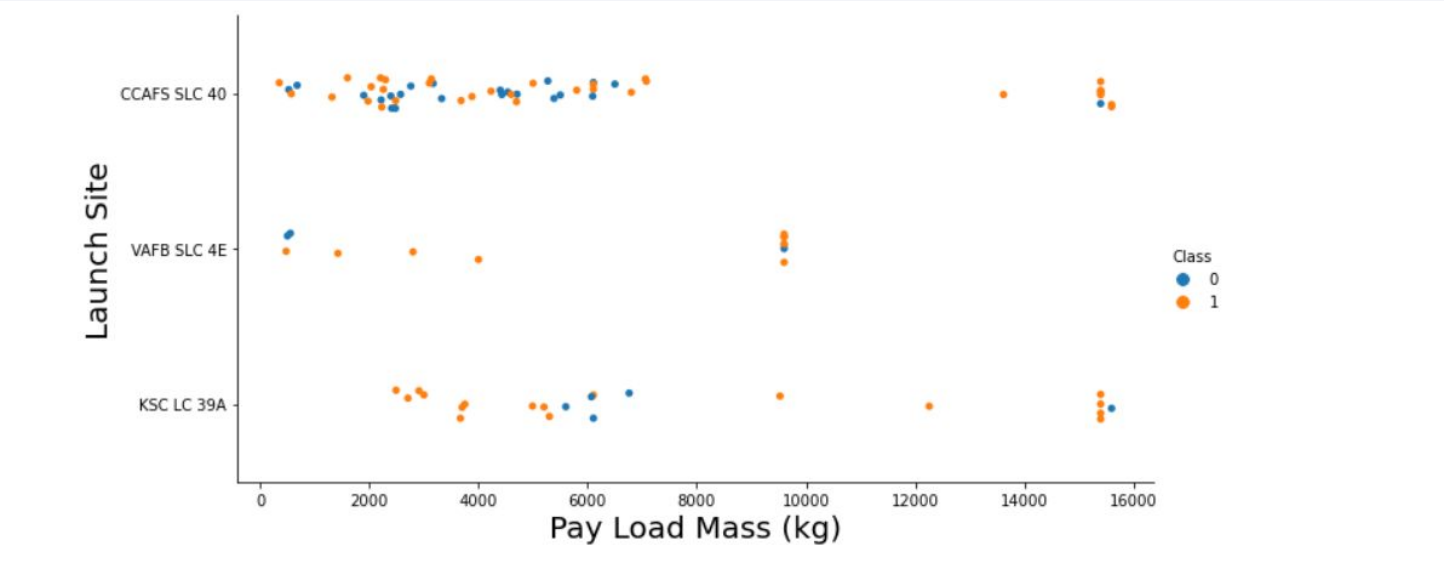
Insights drawn from EDA

Flight Number vs. Launch Site

- From the plot, we found that the larger the flight amount at a launch site, the higher the success rate at a launch site.

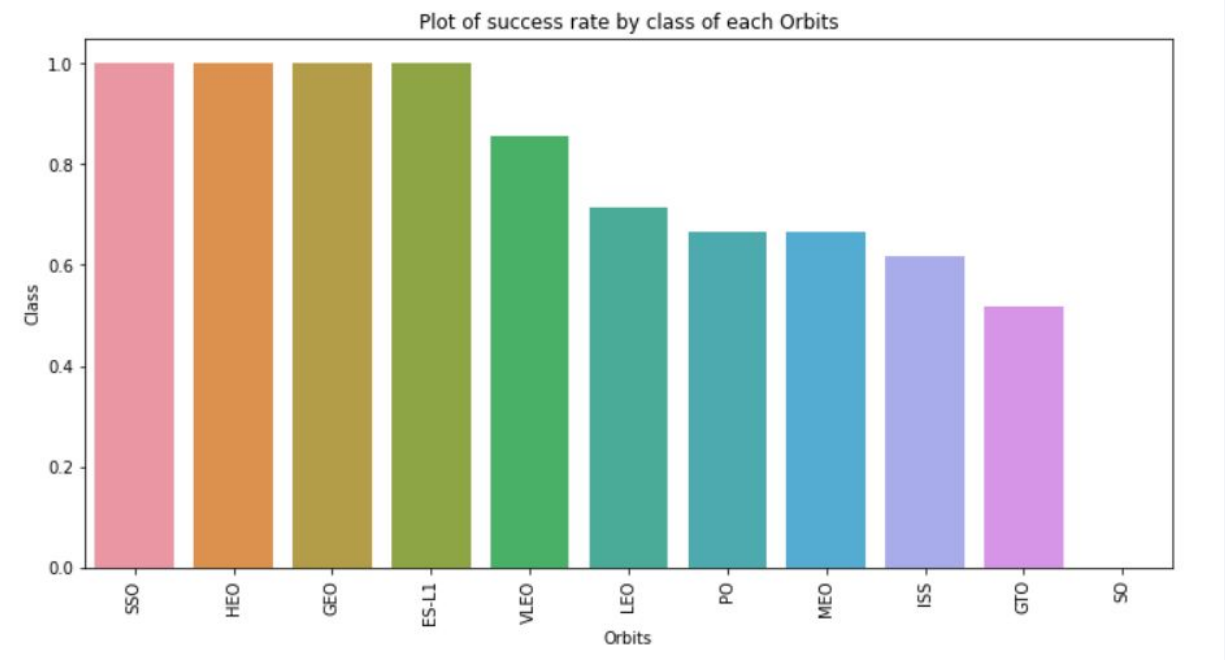


PAYLOAD vs. Launch Site



The greater the payload mass for launch site CCAFS SLC 40 the higher the success rate for the rocket.

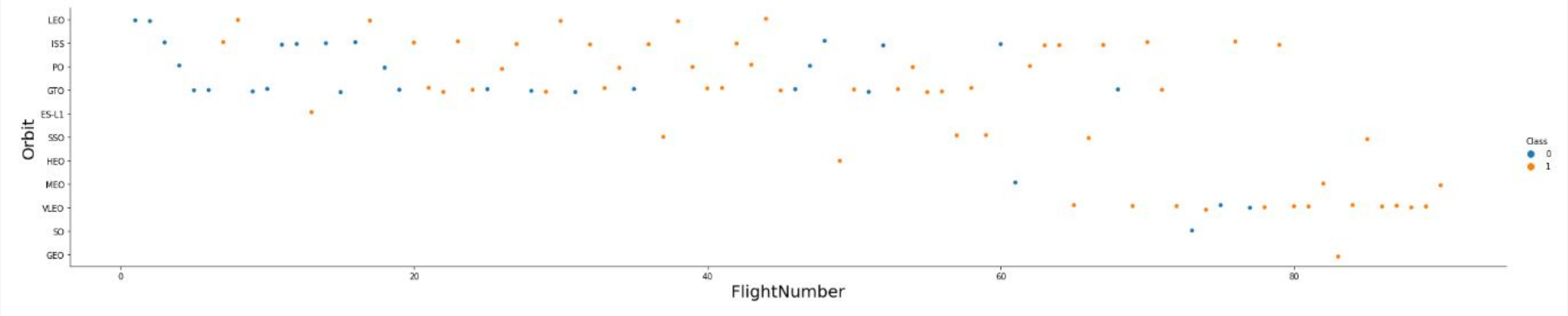
Success Rate vs. Orbit



- From the plot, we can see that ES-L1, GEO, HEO, SSO, VLEO had the most success rate.

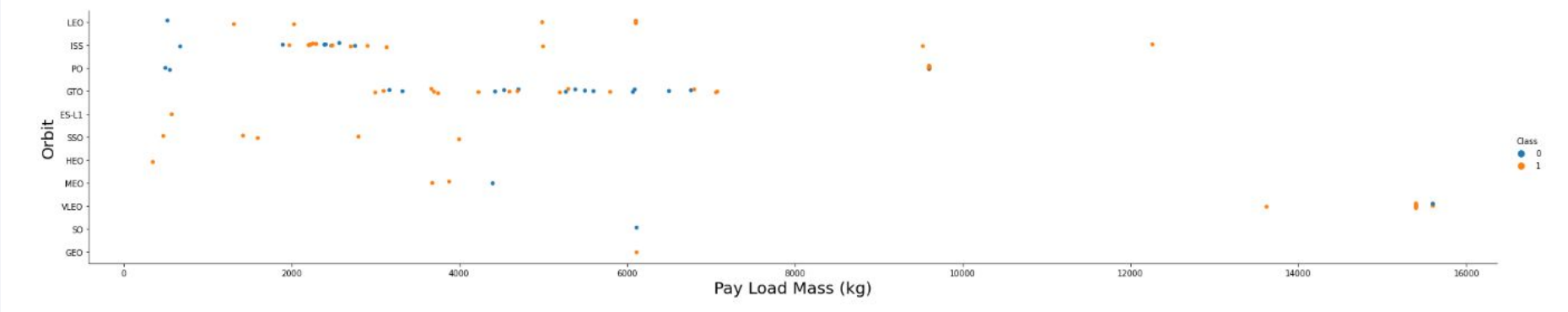
Flight Number vs. Orbit Type

- The plot below shows the Flight Number vs. Orbit type. We observe that in the LEO orbit, success is related to the number of flights whereas in the GTO orbit, there is no relationship between flight number and the orbit.

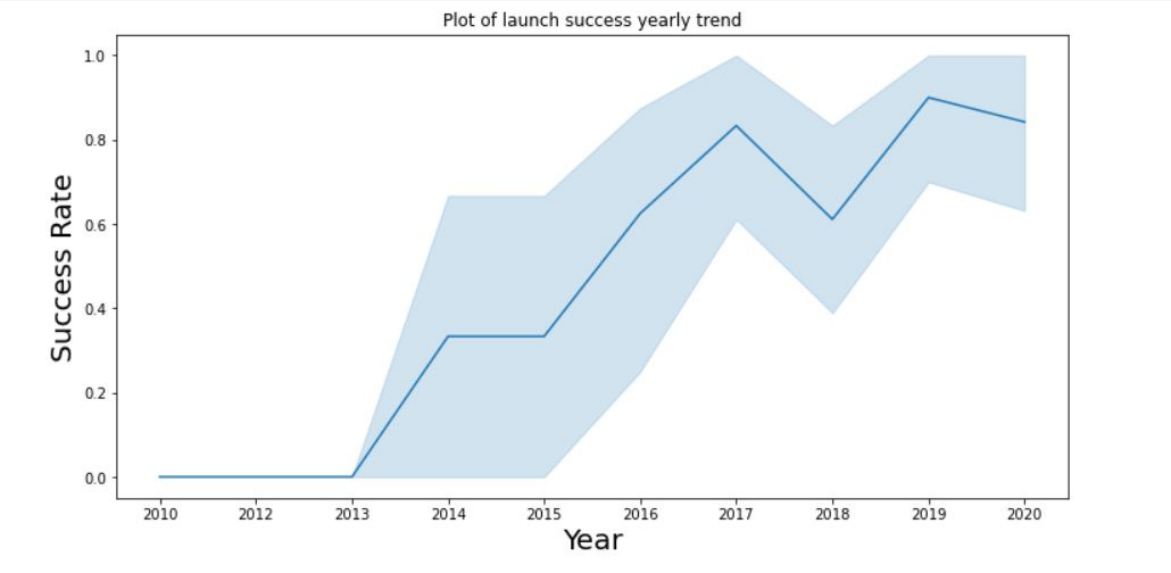


Payload vs. Orbit Type

- We can observe that with heavy payloads, the successful landing are more for PO, LEO and ISS orbits.



Launch Success Yearly Trend



- From the plot, we can observe that success rate since 2013 kept on increasing till 2020.

EDA WITH SQL

ALL Launch Site Names

SQL Query

```
%sql SELECT DISTINCT(Launch_Site) FROM SPACEXTBL;
```

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

- We used the keyword **DISTINCT** to show only unique launch sites from the SpaceX data.

Launch Site Names Begin with 'CCA'

SQL Query

```
%sql SELECT * FROM SPACEXTBL WHERE Launch_Site LIKE 'CCA%' LIMIT 5;
```

SQL query to display 5 records where launch sites begin with `CCA`

Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Orbit	Customer	Mission_Outcome	Landing _Outcome
04-06-2010	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success	Failure (parachute)
08-12-2010	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)
22-05-2012	07:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success	No attempt
08-10-2012	00:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500	LEO (ISS)	NASA (CRS)	Success	No attempt
01-03-2013	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677	LEO (ISS)	NASA (CRS)	Success	No attempt

Total Payload Mass

SQL Query

```
%sql SELECT sum(PAYLOAD_MASS__KG_) FROM SPACEXTBL WHERE Customer == 'NASA (CRS)';
```

sum(PAYLOAD_MASS__KG_)
45596

- Using SQL Query calculated the total payload carried by boosters from NASA as 45596

Average Payload Mass by F9 v1.1

SQL Query

```
%sql SELECT AVG(PAYLOAD_MASS__KG_) FROM SPACEXTBL WHERE Booster_Version LIKE '%F9 v1.1';
```

AVG(PAYLOAD_MASS__KG_)
2928.4

- Using SQL Query calculated the average payload mass carried by booster version F9 v1.1 as 2928.4

First Successful Ground Landing Date

SQL Query

```
%sql SELECT Date FROM SPACEXTBL WHERE "Landing _Outcome" LIKE 'Success (ground pad)' ORDER BY Date DESC LIMIT 1 ;
```

Date
22-12-2015

- Using SQL Query calculated the dates of the first successful landing outcome on ground pad was 22nd December 2015.

Successful Drone Ship Landing with Payload mass between 4000 and 6000

SQL Query

```
%sql SELECT Booster_Version FROM SPACEXTBL WHERE "Landing _Outcome" LIKE 'Success (drone ship)' \
AND PAYLOAD_MASS__KG_ > 4000 AND PAYLOAD_MASS__KG_<6000;
```

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

- Using SQL Query found out the booster version which have successfully landed on drone ship with payload mass between 4000 and 6000.

Total Number of Successful and Failure Mission Outcomes

SQL Query

```
%sql SELECT COUNT(Mission_Outcome) as Successful FROM SPACEXTBL WHERE Mission_Outcome LIKE 'Success%';
```

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

Successful
100

```
%sql SELECT COUNT(Mission_Outcome)as FAILURE FROM SPACEXTBL WHERE Mission_Outcome LIKE 'FAILURE%';
```

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

FAILURE
1

- Using SQL Query calculated the total number of successful mission outcome as 100 and failure outcome as 1.

Boosters Carried Maximum Payload

SQL Query

```
%sql SELECT Booster_Version,PAYLOAD_MASS__KG_ FROM SPACEXTBL WHERE PAYLOAD_MASS__KG_ = (SELECT MAX(PAYLOAD_MASS__KG_) FROM SPACEXTBL) ORDER BY Booster
```

Booster_Version	PAYLOAD_MASS__KG_
F9 B5 B1048.4	15600
F9 B5 B1048.5	15600
F9 B5 B1049.4	15600
F9 B5 B1049.5	15600
F9 B5 B1049.7	15600
F9 B5 B1051.3	15600
F9 B5 B1051.4	15600
F9 B5 B1051.6	15600
F9 B5 B1056.4	15600
F9 B5 B1058.3	15600
F9 B5 B1060.2	15600
F9 B5 B1060.3	15600

- Using SQL Query found the booster that have carried the maximum payload.

2015 Launch Records

SQL Query

```
%sql SELECT Booster_Version, Launch_Site, "Landing _Outcome", substr(Date,4,2) as Month FROM SPACEXTBL WHERE "Landing _Outcome" LIKE 'Failure (drone s
```

Booster_Version	Launch_Site	Landing _Outcome	Month
F9 v1.1 B1012	CCAFS LC-40	Failure (drone ship)	01
F9 v1.1 B1015	CCAFS LC-40	Failure (drone ship)	04

- Using SQL Query found the month, failure landing outcomes in drone ship, booster versions, launch_site for the year 2015.

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

SQL Query

```
%sql SELECT "Landing _Outcome" ,COUNT("Landing _Outcome") as Count FROM SPACEXTBL WHERE "Landing _Outcome" LIKE '%Success%' AND Date BETWEEN '04-06-20
```

Landing _Outcome	Count
Success	20
Success (drone ship)	8
Success (ground pad)	6

- Using SQL Query found the total successful landing outcomes between 04-06-2010 and 20-03-2017

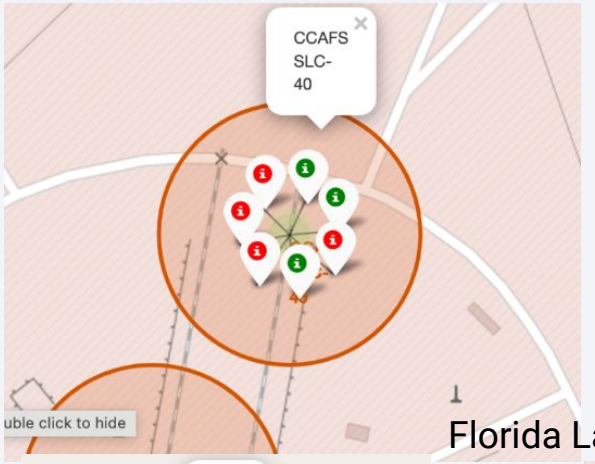
Section 4

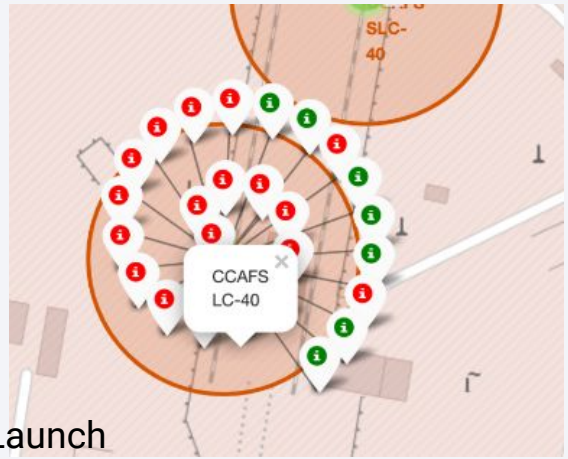
Launch Sites Proximities Analysis

All launch sites global map markers

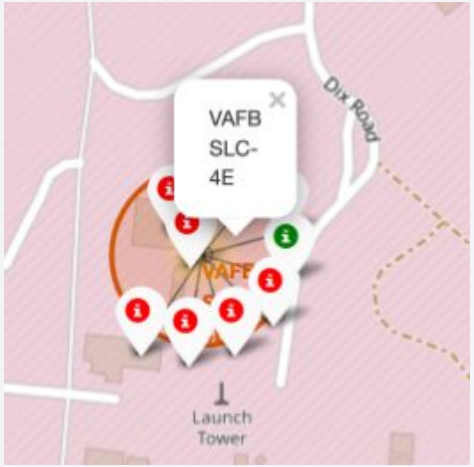
SpaceX launch sites are in United States of America coasts. California and Florida

Markers showing launch sites with color labels

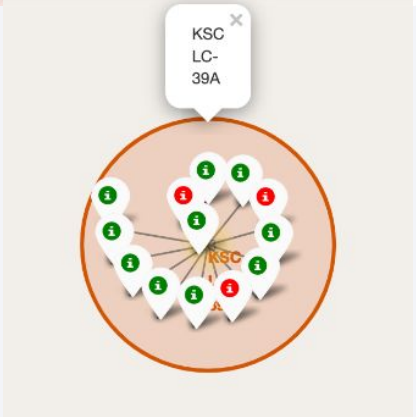




Florida Launch Sites

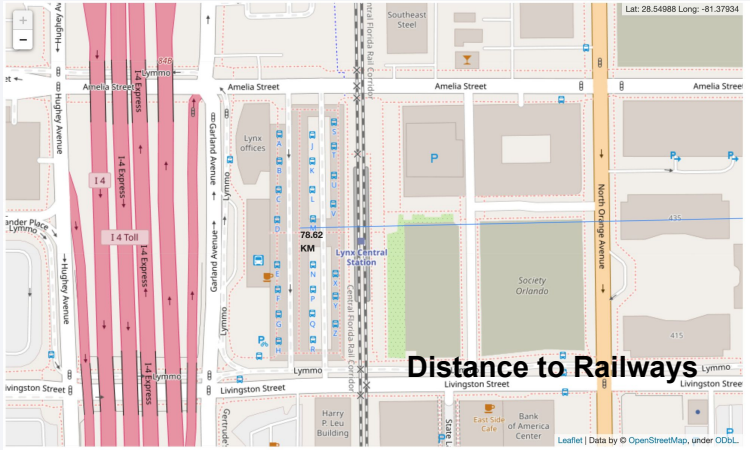


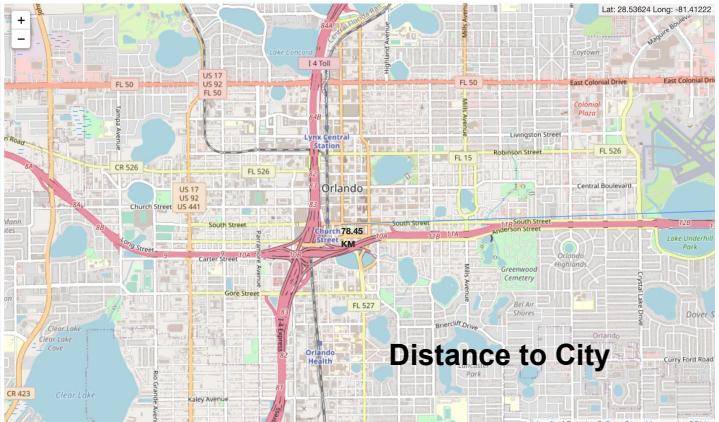
California Launch Sites



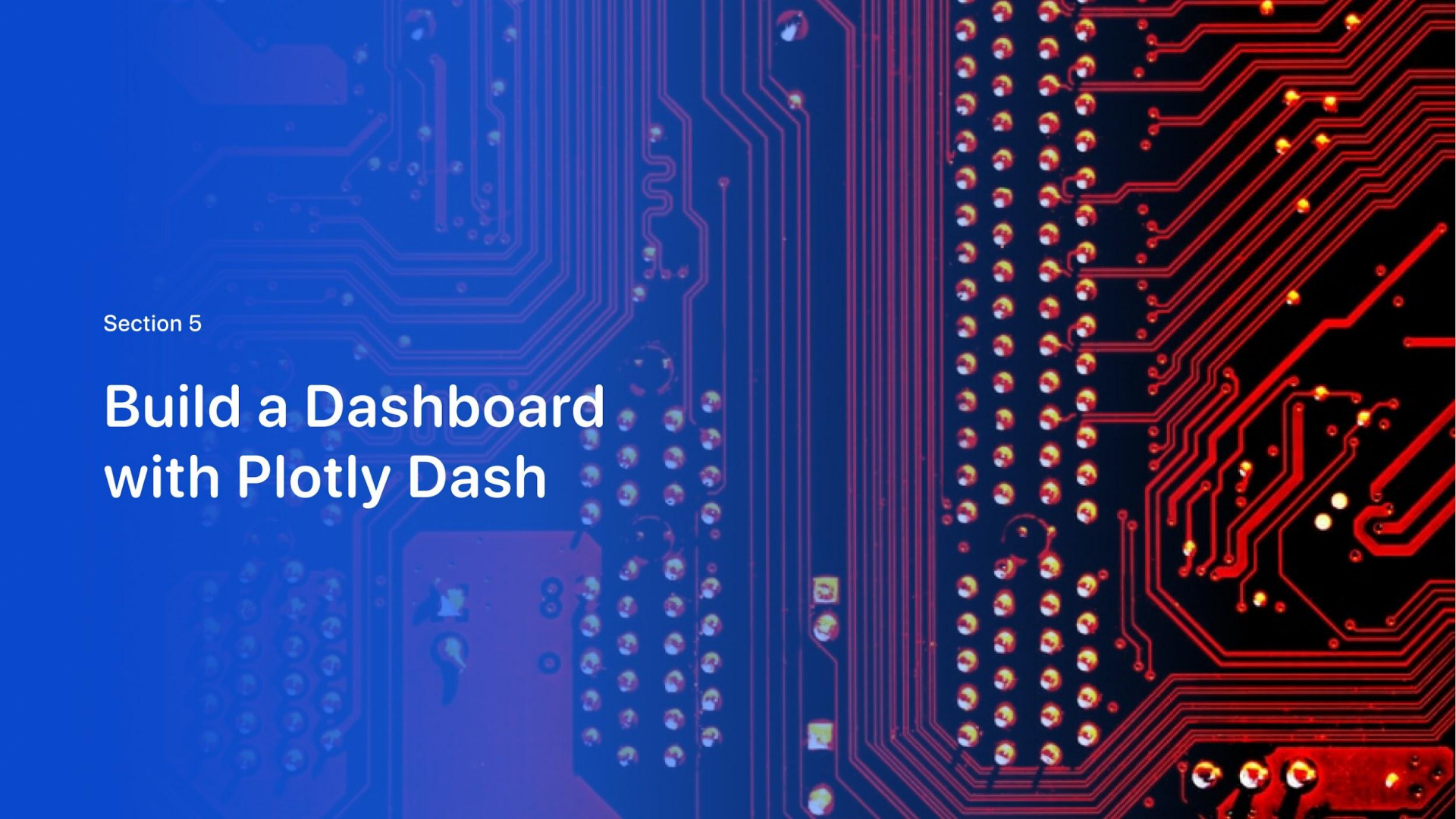
Green - Successful Launch
Red - Failure

Launch Site distance to landmarks





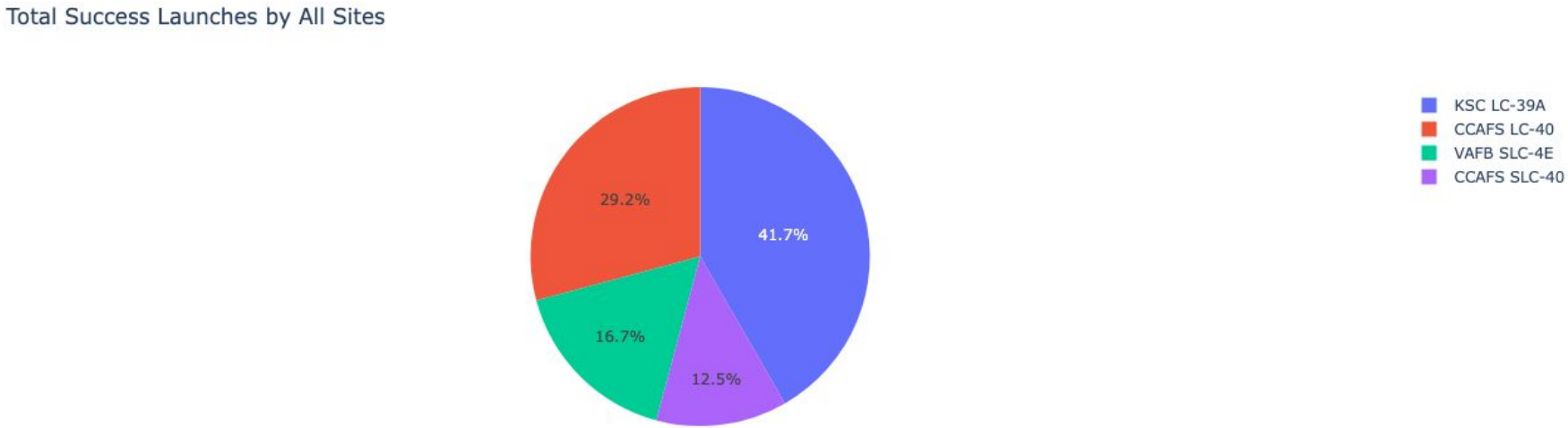
- Are launch sites in close proximity to railways? - No
- Are launch sites in close proximity to highways? - No
- Are launch sites in close proximity to coastline? - Yes
- Do launch sites keep certain distance away from cities? - Yes



Section 5

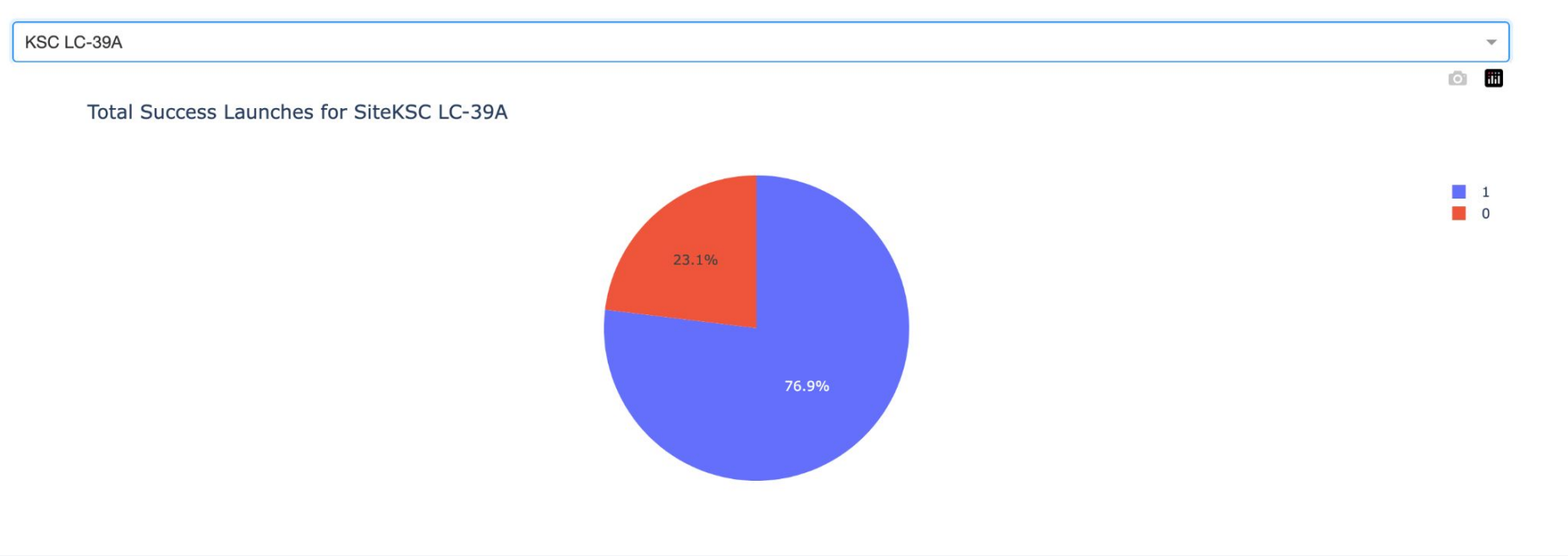
Build a Dashboard with Plotly Dash

Pie chart showing the success percentage achieved by each launch site



From the pie chart we can observe that KSC LC - 39 A has the most successful launch from all the sites.

Pie chart showing the Launch site with the highest launch success ratio



From the pie chart we can observe that KSC LC - 39 A achieved 76.9 % success rate.

Scatter plot of Payload vs Launch Outcome for all sites, with different payload selected in the range slider

Low weighted payload 0kg – 4000kg

Heigh weighted payload 4000kg – 10000kg

From the graph we can observe that success rate is more for low payloads than higher payloads.

Section 6

Predictive Analysis (Classification)

Classification Accuracy

`Algorithm	Accuracy	Accuracy on Test Data
Logistic Regression	0.8464	0.8333
SVM	0.8482	0.833
KN	0.8482	0.833
Decision Tree	0.875	0.833

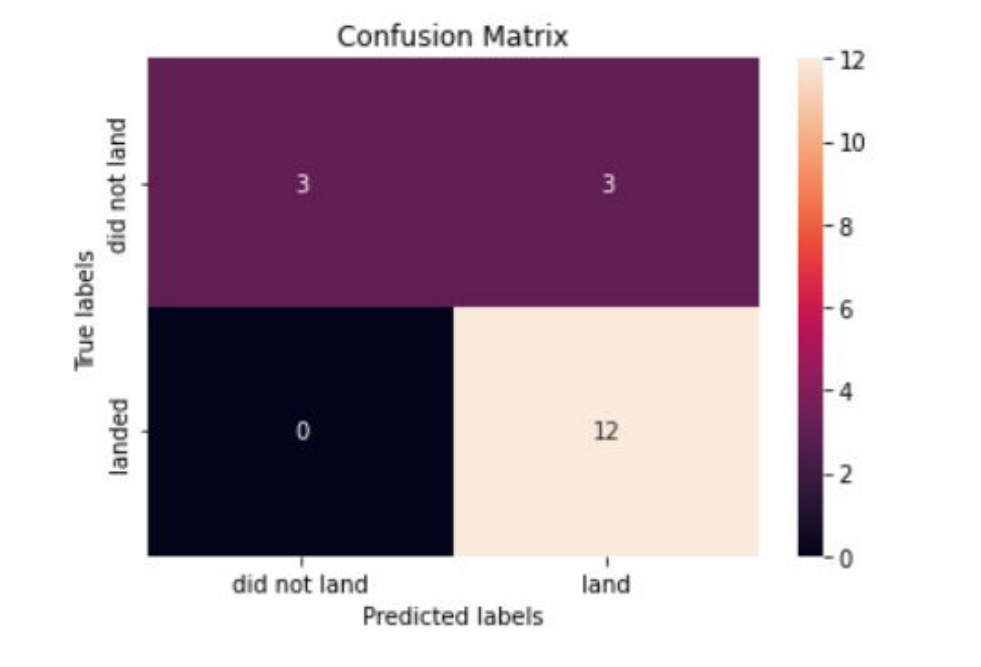
```
models = {'KNeighbors':knn_cv.best_score_,
          'DecisionTree':tree_cv.best_score_,
          'LogisticRegression':logreg_cv.best_score_,
          'SupportVector': svm_cv.best_score_}

bestalgorithm = max(models, key=models.get)
print('Best model is', bestalgorithm,'with a score of', models[bestalgorithm])
if bestalgorithm == 'DecisionTree':
    print('Best params is :', tree_cv.best_params_)
if bestalgorithm == 'KNeighbors':
    print('Best params is :', knn_cv.best_params_)
if bestalgorithm == 'LogisticRegression':
    print('Best params is :', logreg_cv.best_params_)
if bestalgorithm == 'SupportVector':
    print('Best params is :', svm_cv.best_params_)
```

```
Best model is DecisionTree with a score of 0.875
Best params is : {'criterion': 'gini', 'max_depth': 8, 'max_features': 'auto', 'min_samples_leaf': 1, 'min_samples_split': 5, 'splitter': 'random'}
```

- The decision tree classifier is the model with the highest classification accuracy

Confusion Matrix



All models used have the same confusion matrix.

Conclusions

We can conclude that:

- The Decision tree classifier is the best machine learning algorithm for this task.
- Launch success rate started to increase in 2013 till 2020.
- The larger the flight amount at a launch site, the greater the success rate at a launch site.
- Orbits ES-L1, GEO, HEO, SSO, VLEO had the most success rate.
- KSC LC-39A had the most successful launches of any sites.

Thank you!