Winning Space Race with Data Science

Ahmed Seif
15th June 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

- Dataset Info

  The SpaceX dataset compiles crucial data related to Falcon 9 rocket launches. It includes variables such as launch cost, rocket specifications, and the success rate of first stage landings, instrumental for gauging launch cost efficiency. This resource is particularly useful for alternate companies planning to compete with SpaceX's launch services, enabling them to strategize based on accurate, market-leading benchmarks.

- Project Objective

  Develop a predictive model using the SpaceX dataset to accurately forecast the success of the first stage landing of Falcon 9 rockets. This prediction will help in determining the cost of each launch, thereby providing valuable insights for alternate companies planning to competitively bid against SpaceX for rocket launches.

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - Data was collected using SpaceX API and web scraping from Wikipedia.
- Perform data wrangling
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models

Data Collection – SpaceX API

- We used the get request to the SpaceX API to collect data, clean the requested data and did some basic data wrangling and formatting.
- The link to the notebook is https://github.com/chuksoo/IBM-Data-Science-Capstone-SpaceX/blob/main/Data%20Collection%20API.ipynb.

1. Get request for rocket launch data using API

In [6]:
```
spacex_url="https://api.spacexdata.com/v4/launches/past"
```

In [7]:
```
response = requests.get(spacex_url)
```

2. Use json_normalize method to convert json result to dataframe

In [12]:
```
# Use json_normalize method to convert the json result into a dataframe

# decode response content as json
static_json_df = res.json()
```

In [13]:
```
# apply json_normalize
data = pd.json_normalize(static_json_df)
```

3. We then performed data cleaning and filling in the missing values

In [30]:
```
rows = data_falcon9['PayloadMass'].values.tolist()[0]

df_rows = pd.DataFrame(rows)
df_rows = df_rows.replace(np.nan, PayloadMass)

data_falcon9['PayloadMass'][0] = df_rows.values
data_falcon9
```

Data Collection - Scraping

- Using BeautifulSoup library in Python
- Collected data over Wikipedia
- Parsed html files with requests and html5lib to conduct data into tables
- Saved collected data to a Pandas Dataframe

Data Wrangling

- EDA (Exploratory Data Analysis) to:
- Identify dataset key features
- Clean missing values
- Get a detailed version of the dataset
- Determine training labels

Data Visualization EDA

Visualizations talk better than words, This is why conducting a visualized report is needed

Among the Visualization EDA, there was found that:

- Rocket missions has developed over the years 2013-2020
- Over the years, The payload masses has increased as well as the success rate
- The heliosynchronous orbit, had only one mission and it failed

EDA with SQL

SQL has always been a useful tool for Data Analysis, featuring JOIN and other data manipulation statements. Exploring the data with sql turned out to show:

- Successful missions over data was 35 missions
- Only F9 B5 Boosters could carry the 15,600 KG payloads
- Total Payloads sent to space was around 45,600 KG

Build an Interactive Map with Folium

Folium is an interactive map visualizer, Using folium, detected that:

- Launch sites are at least 50KM away from cities
- Launch sites are around 6 KM away from highways
- 1 KM away from railway stations and coastlines

Build a Dashboard with Plotly Dash

Plotly Dash is a mix of tools that use flask server to display interactive visualizations on the web, using them we assured that:

- Development over years that successful missions completed increased year by year
- Highest Success rate for a launch site was around 80% of missions launched from there
- Launch Sites are strongly correlated to payload masses

Predictive Analysis (Classification)

Using classification techniques, SVM, Decision Trees, Logistic Regression and KNN classifier:

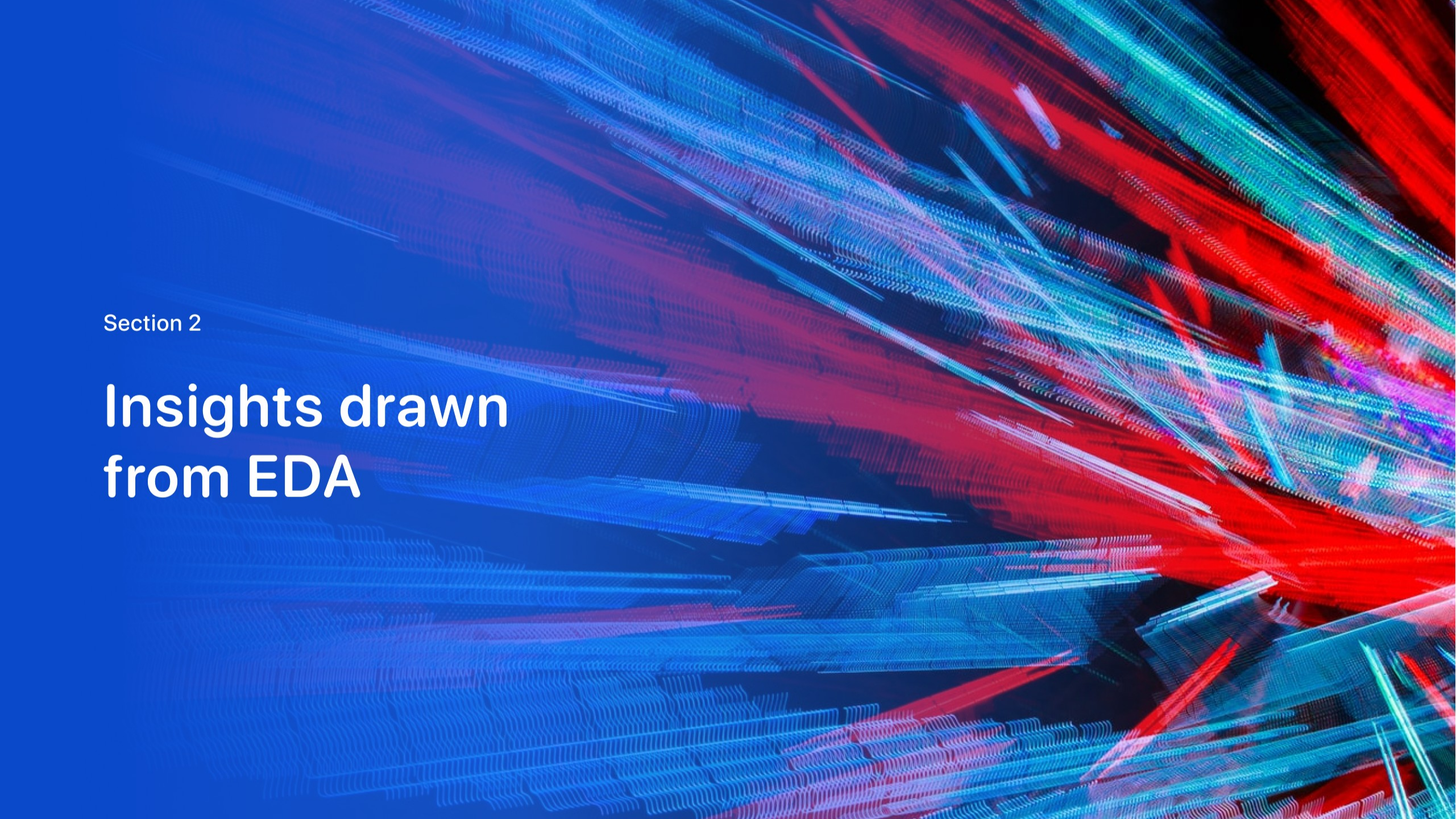
Actually no classifier showed a better accuracy than the other and all of them tended to cover around 84% accuracy for detection of success-to-failure of missions

Results

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

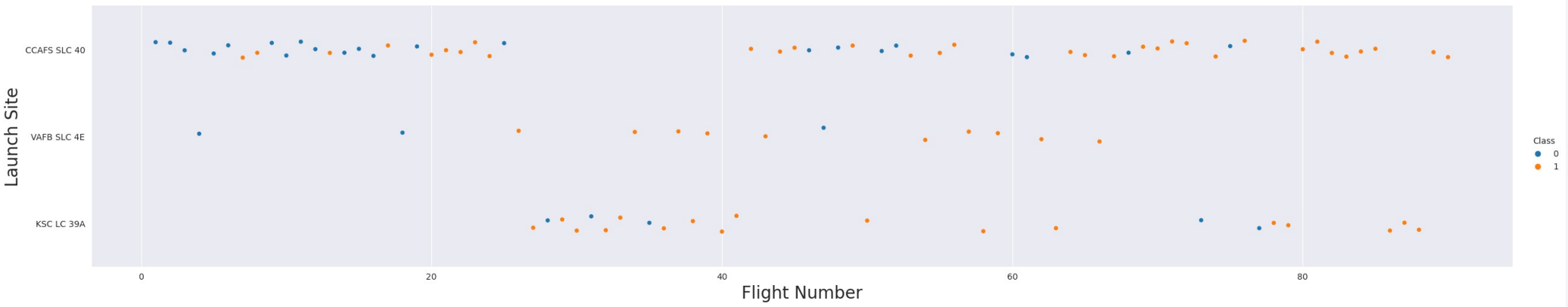
Section 2

Insights drawn from EDA



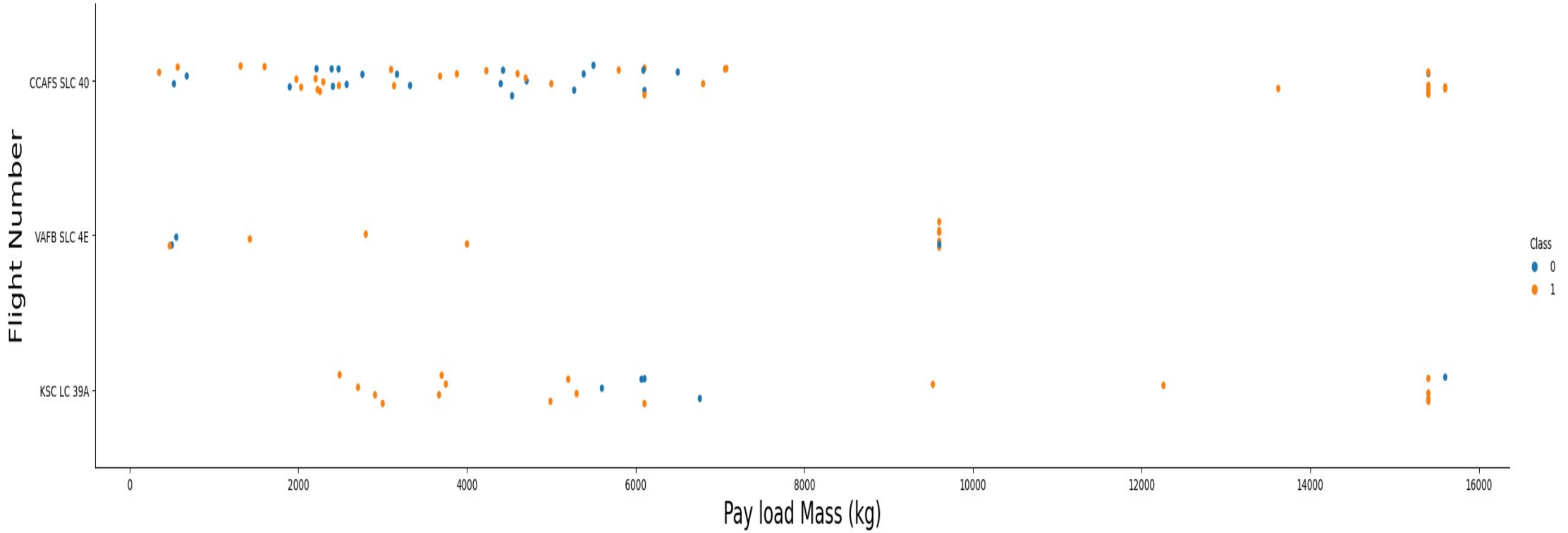
Flight Number vs. Launch Site

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



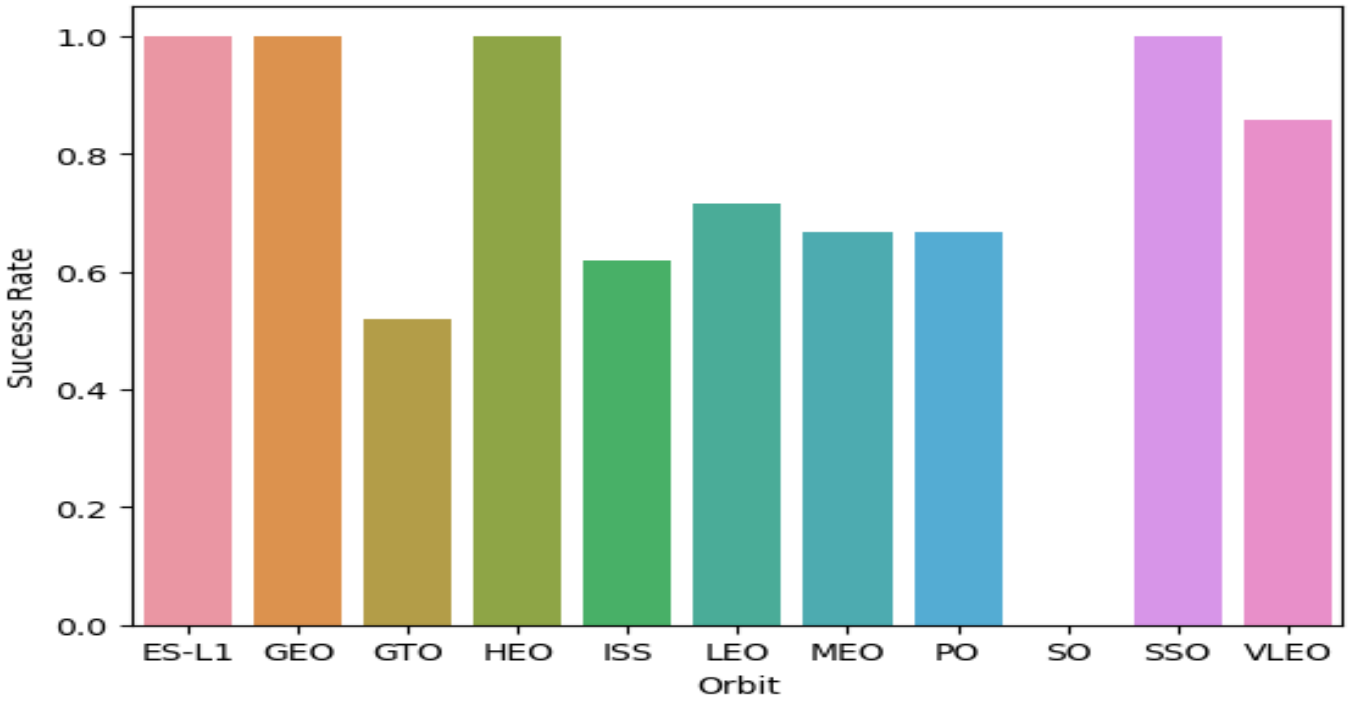
Payload vs. Launch Site

The higher the payload mass the more successful missions are



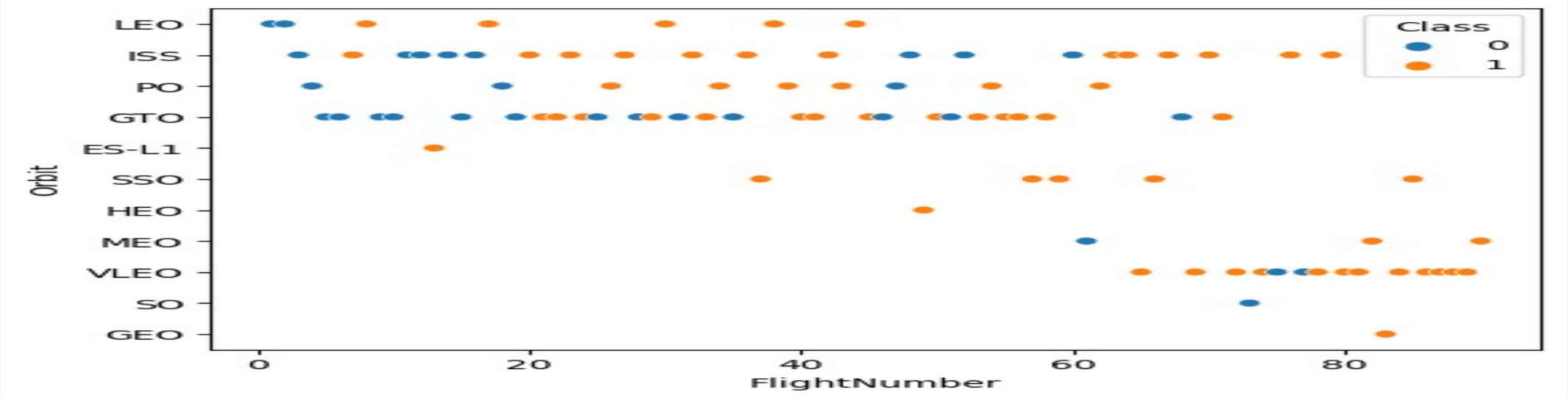
Success Rate vs. Orbit Type

- SO had No successful attempts
- ES-L1, GEO, HEO, SSO orbits are very successful



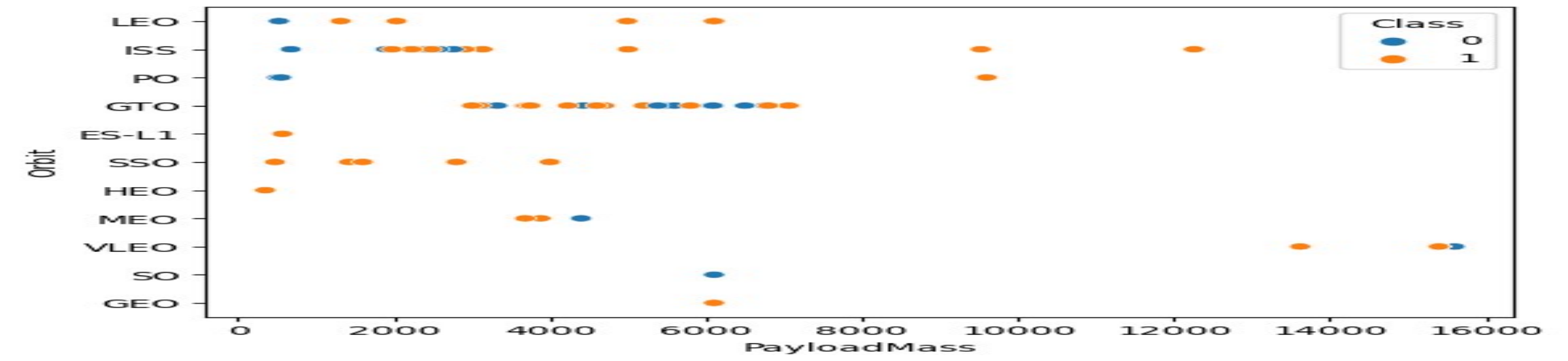
Flight Number vs. Orbit Type

- VLEO, had been the most recent orbit and it appears to be very sucessful
- ISS as well has good overall succession rates



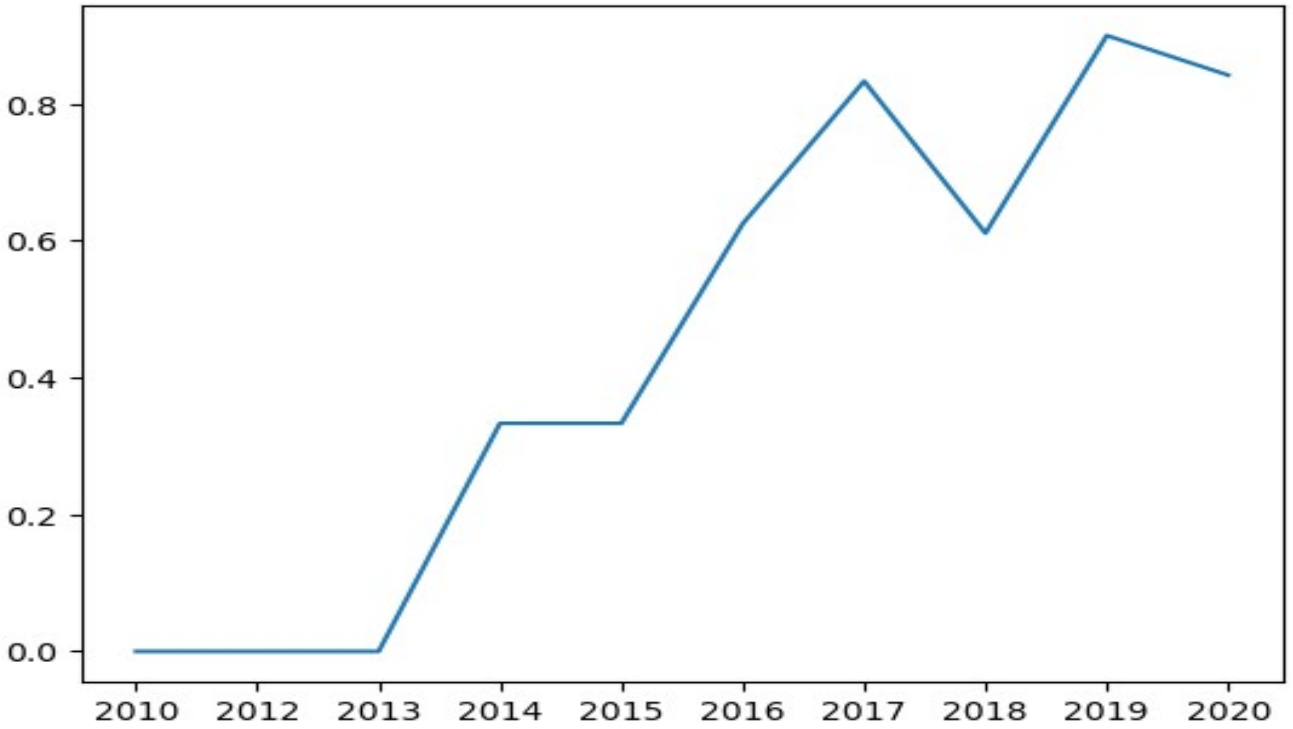
Payload vs. Orbit Type

- SSO has been used for lightweighted payloads
- At heavier loads PO, ISS, VLEO are very successful as well



Launch Success Yearly Trend

- Development since 2013 till 2020 Affected higher success rates



All Launch Site Names

- Using SQL, We can see that there are only 4 Launch Sites that SpaceX rockets are launched from
  - CCAFS LC-40
  - CCAFS SLC-40
  - VAFB SLC-4E
  - KSC LC-39A

```
1  %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'

```
In 9  1  %sql select * from spacextbl where launch_site like 'CCA%' limit 5
```

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

Out 9

Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Orbit	Customer	Mission_Outcome	Landing
06/04/2010	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0.0	LEO	SpaceX	Success	Failure (
12/08/2010	15:43:00	F9 v1.0 B0004	CCAFS LC-40	Dragon demo flight C1, two CubeSats, barrel of Brouere cheese	0.0	LEO (ISS)	NASA (COTS) NRO	Success	Failure (
22/05/2012	7:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525.0	LEO (ISS)	NASA (COTS)	Success	No atten
10/08/2012	0:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500.0	LEO (ISS)	NASA (CRS)	Success	No atten

Top First 5 launches was successful missions on CCA* Launch Sites

Total Payload Mass

NASA’s Total payloads were around 45600 KG

```
%sql select sum(PAYLOAD_MASS__KG_) from spacextbl where customer == 'NASA (CRS)'
```

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

sum(PAYLOAD_MASS__KG_)
45596.0

Average Payload Mass by F9 v1.1

Around 2928 KG

```
%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

First Successful Ground Landing Date

January 2020

```
%sql select min(spacextbl.Date) from spacextbl where landing_outcome like 'Success%'
```

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

min(spacextbl.Date)
01/07/2020

Successful Drone Ship Landing with Payload between 4000 and 6000

```
%sql select distinct(Booster_Version) from spacextbl where payload_mass__kg_ between 4000 and 6000 and landing_outcome ==
 'Failure (drone ship)'
```

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

Booster_Version
F9 FT B1020

Total Number of Successful and Failure Mission Outcomes

```
%sql select count(*),mission_outcome from spacextbl group by mission_outcome
```

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

count(*)	Mission_Outcome
898	None
1	Failure (in flight)
98	Success
1	Success
1	Success (payload status unclear)

Around 100 Missions Success and only 1 Failure

Boosters Carried Maximum Payload

```
In 40  1  %sql select Booster_Version from spacextbl where payload_mass__kg_ == (select max(payload_mass__kg_) from spacextbl)

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

Out 40

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

2015 Launch Records

April and October been failure landing months for 2015

```
In 44  1  %sql select substr(Date,4,2), landing_outcome, booster_version, launch_site from spacextbl where substr(Date,7,4)='2015' and
            landing_outcome=='Failure (drone ship)'

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

Out 44

substr(Date,4,2)	Landing_Outcome	Booster_Version	Launch_Site
10	Failure (drone ship)	F9 v1.1 B1012	CCAFS LC-40
04	Failure (drone ship)	F9 v1.1 B1015	CCAFS LC-40

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

Applying Count, Where, Between, Like, Group By and Order By Statments to view successful landings between June 2010 and March 2017

```
In 54  1  %sql SELECT landing_outcome, COUNT(*) AS count FROM spacextbl WHERE date between '04-06-2010' and '20-03-2017' AND
          landing_outcome like 'Success%' GROUP BY landing_outcome ORDER BY count DESC;

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

Out 54

Landing_Outcome	count
Success	20
Success (drone ship)	8
Success (ground pad)	7

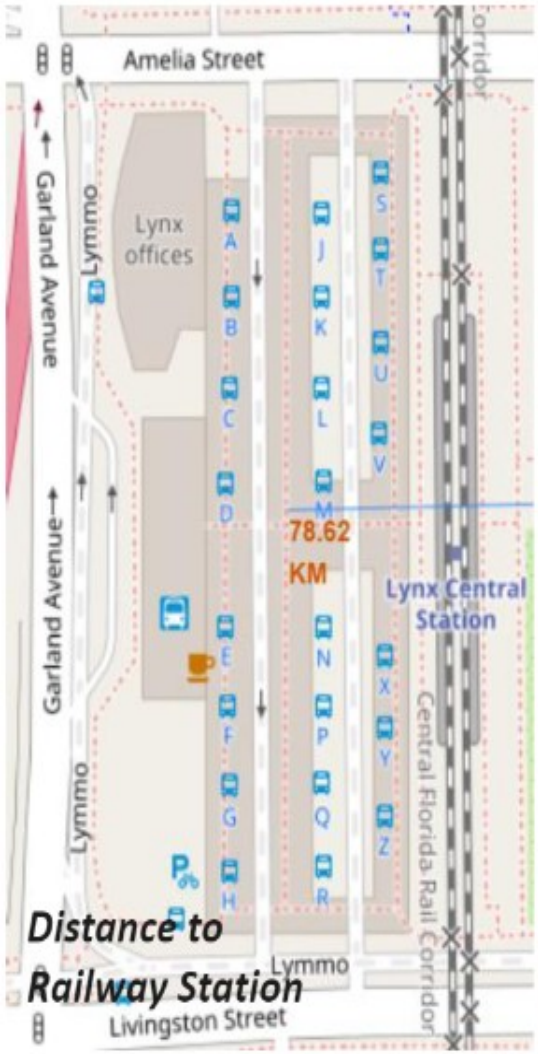
Section 4

Launch Sites Proximities Analysis

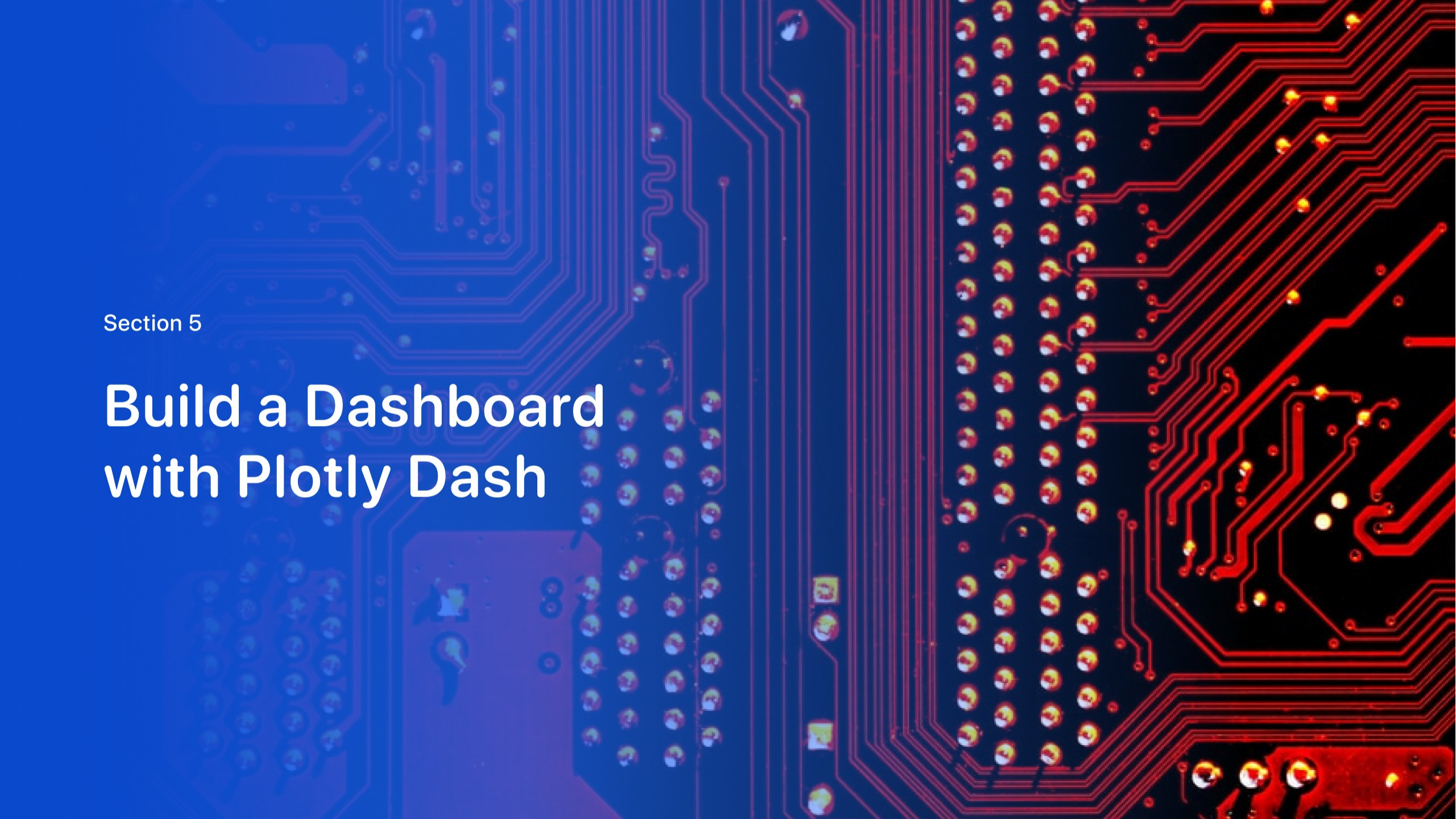
All launch sites global map markers

Markers showing launch sites with color labels

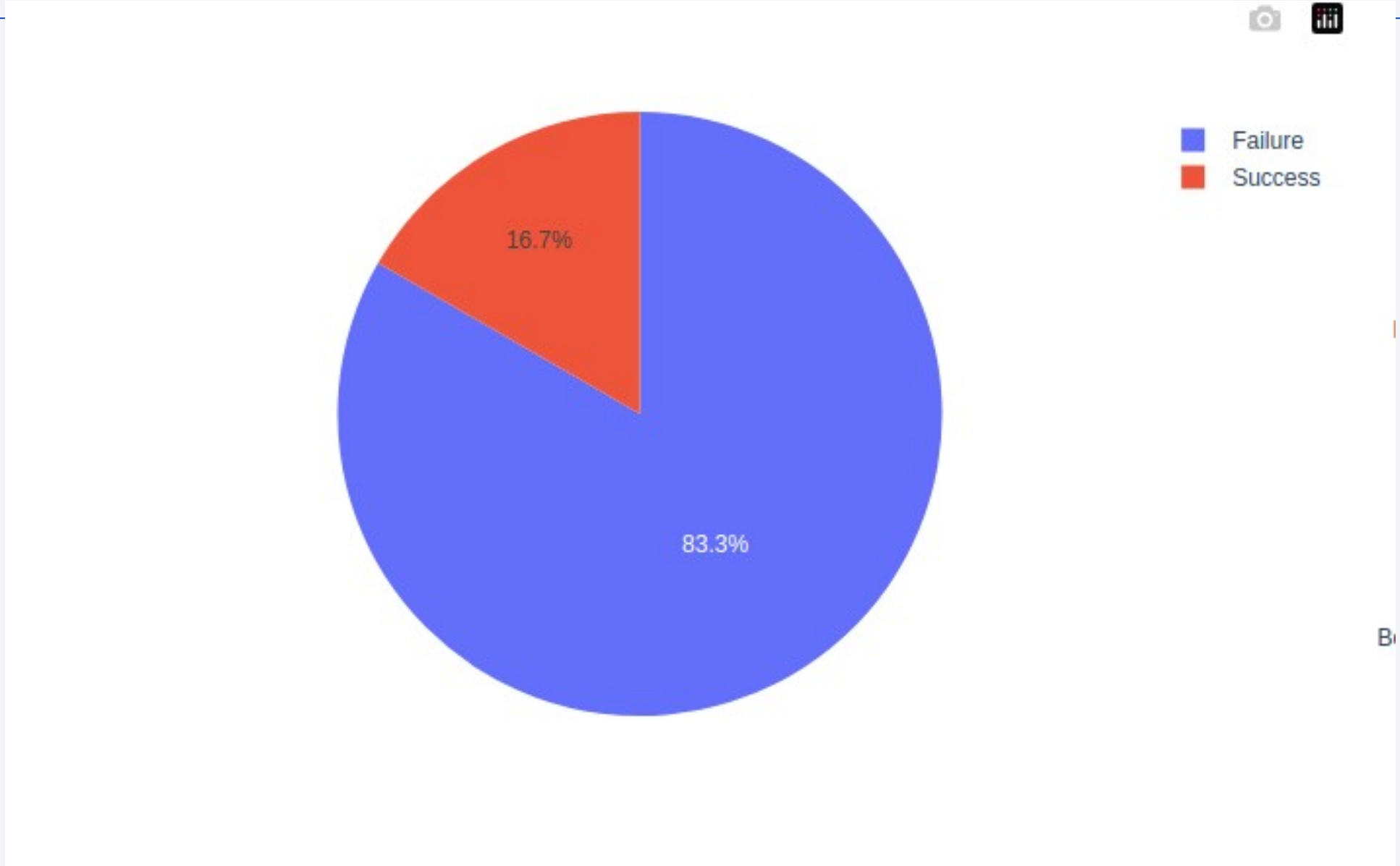
Launch Site distance to landmarks



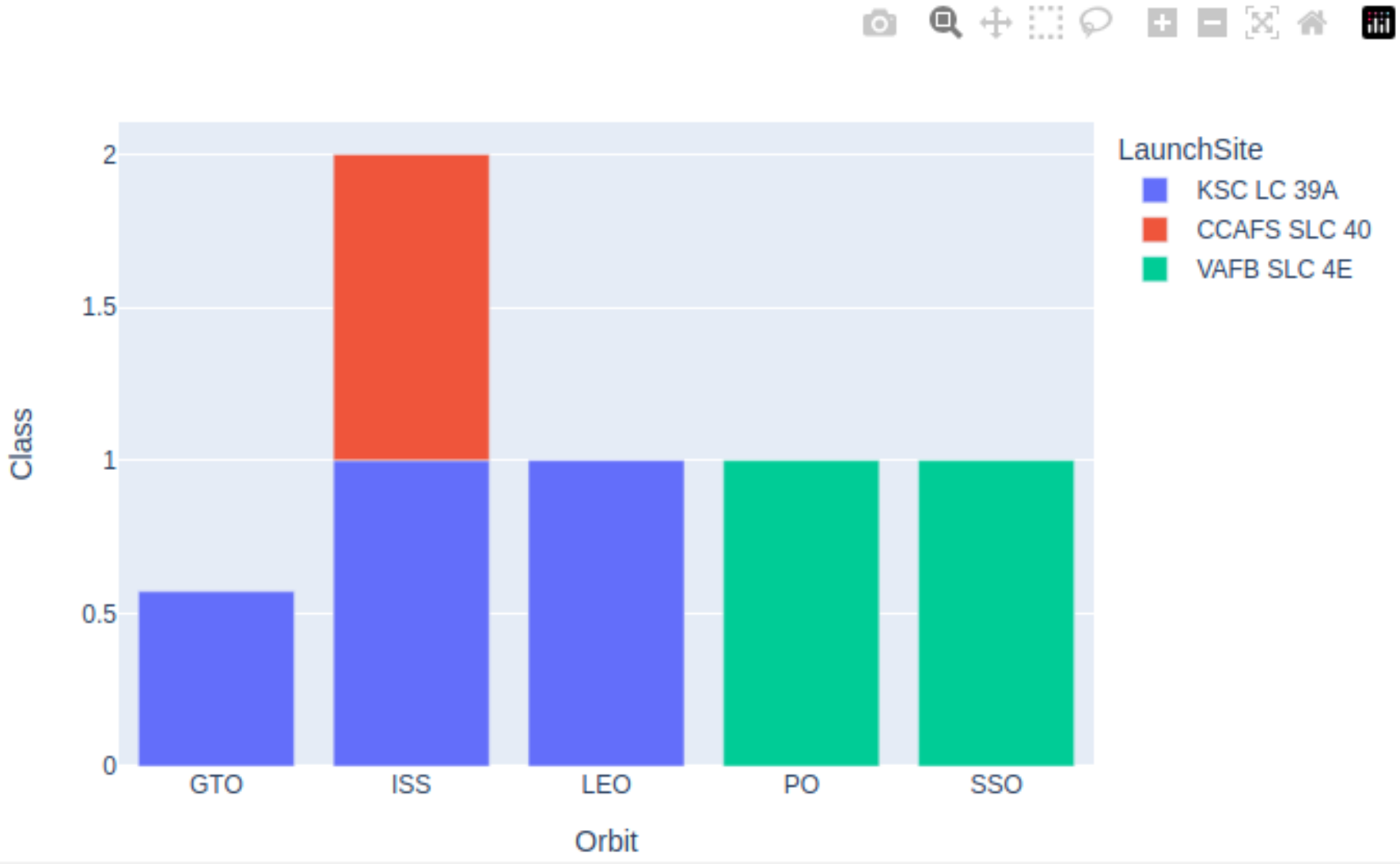
- Are launch sites in close proximity to railways? No
- Are launch sites in close proximity to highways? No
- Are launch sites in close proximity to coastline? Yes
- Do launch sites keep certain distance away from cities? Yes


Section 5
Build a Dashboard with Plotly Dash

Failure and Success Total rates for 2017



Success Rates for Orbits with Launch sites Base in 2017



Section 6

Predictive Analysis (Classification)

Classification Accuracy

- Logistic Regression has scored the best cross validation score with 81%

```
In 31
from sklearn.model_selection import cross_val_score
models = [('Logistic Regression', logreg_cv),
          ('K Nearest Neighbors', knn_cv),
          ('Decision Tree', tree_cv),
          ('Support Vector Machine', svm_cv)]
for name, model in models:
    cv_score = cross_val_score(model, X, y, cv=10)
    print(f'{name} Cross-Validation Score: {cv_score.mean()}')
```

```
Logistic Regression Cross-Validation Score: 0.8111111111111111
K Nearest Neighbors Cross-Validation Score: 0.7999999999999999
Decision Tree Cross-Validation Score: 0.7999999999999999
Support Vector Machine Cross-Validation Score: 0.7999999999999999
```

Confusion Matrix

- The confusion matrix for the Logistic Regression shows that the classifier can distinguish between the different classes. The major problem is the false positives .i.e., unsuccessful landing marked as successful landing by the classifier.

Conclusions

We can conclude that:

- Development since 2013 to 2020 had great impact on landing rockets
- Orbits more far to the earth tend to show better success rates
- Logistic Regression can be the best classifier at 10 cross validations for the current dataset
- There’s a very strong correlation between launch sites and payload masses

Thank you!