IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

<Hashir Ali Khan>
<19th September 2021>

Outline

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

Executive Summary

- Summary of methodologies
- Summary of all results

Introduction

- Project background and context
- Problems you want to find answers

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - Requested SpaceX API to import the required Data Set
- Perform data wrangling
  - Replaced the missing values with mean payload value
- 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

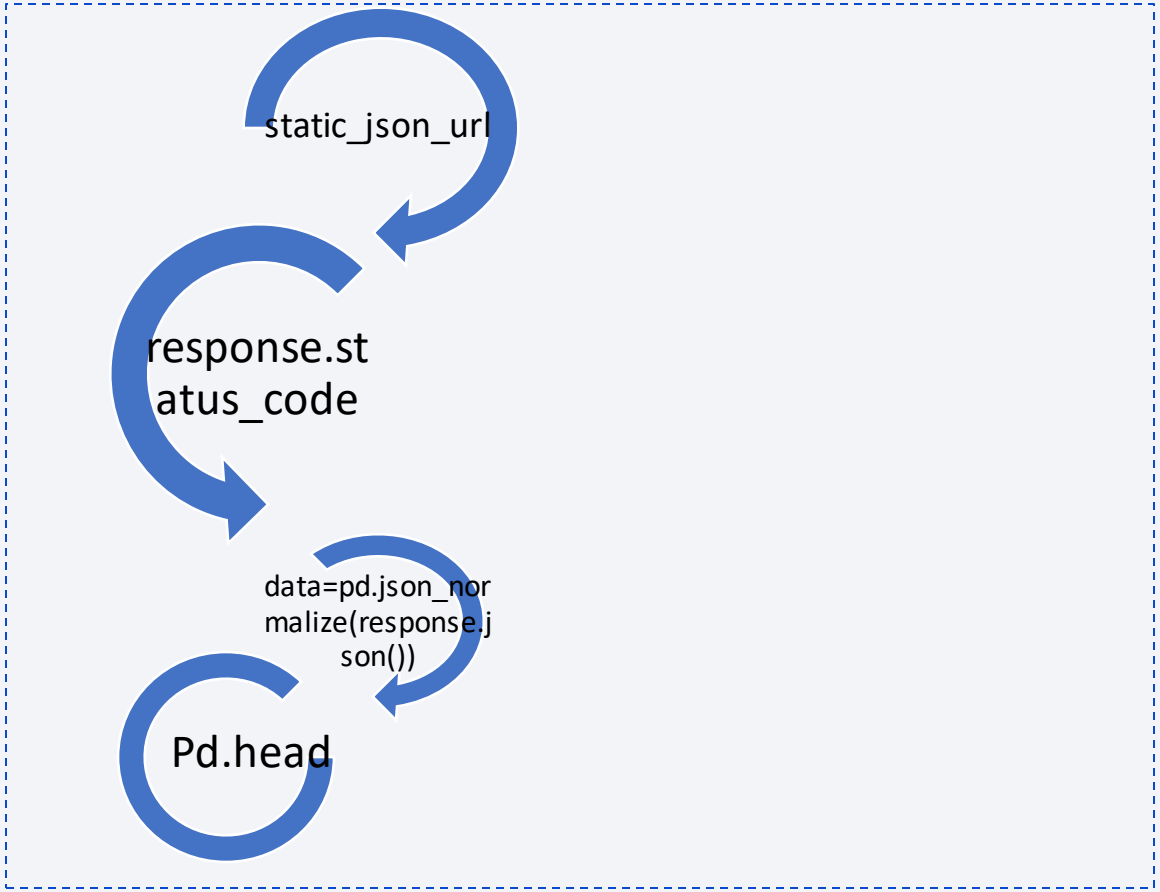
Data Collection

- Describe how data sets were collected.
- Data sets were collected using Spacex API and Json objects.
- Creating a Response object using spacex URL
- Converting JSON result into pandas data frame using the following code: **data=pd.json_normalize(response.json())**
- You need to present your data collection process use key phrases and flowcharts

Data Collection – SpaceX API

- Present your data collection with SpaceX REST calls using key phrases and flowcharts

- https://github.com/hashirkhan-786/Data-Science-/blob/master/Data%20Collection%20API%20Lab.ipynb



Data Collection - Scraping

- Present your web scraping process using key phrases and flowcharts

- Add the GitHub URL of the completed web scraping notebook

- https://github.com/hashirkhan-786/Data-Science-/blob/master/Web%20Scraping%20for%20Final%20Project.ipynb

- HTTP Get method to create a response object
- Creating beautifulSoup Object
- Finding all tables using beautiful soup

Data Wrangling

- Load Space X dataset, and identify the missing values using the following code, : df.isnull().sum()/df.count()*100

- https://github.com/hashirkhan-786/Data-Science-/blob/master/EDA%20Lab%20for%20Final%20Capstone.ipynb

EDA with Data Visualization

- Scatter plots, Bar charts and Line plots were plotted to find the Success rates of different orbits, launch sites and relationships of payloads on the success of the launch outcome
- https://github.com/hashirkhan-786/Data-Science-/blob/master/jupyter-labs-eda-dataviz%20hashir.ipynb

EDA with SQL

- Using bullet point format, summarize the SQL queries you performed
- %sql SELECT DISTINCT LAUNCH_SITE from SPACEXTBL
- %sql SELECT LAUNCH_SITE from SPACEXTBL WHERE LAUNCH_SITE like '%CCA%' LIMIT 5
- %sql SELECT SUM (PAYLOAD_MASS__KG_) FROM SPACEXTBL
- %sql SELECT AVG (PAYLOAD_MASS__KG_) FROM SPACEXTBL WHERE BOOSTER_VERSION = 'F9 v1.1‘
- %sql SELECT MIN(DATE) FROM SPACEXTBL WHERE LANDING__OUTCOME = 'Success (ground pad)'

CONTD EDA with SQL

- %sql SELECT BOOSTER_VERSION , 4000< PAYLOAD_MASS__KG_ <6000 FROM SPACEXTBL WHERE LANDING__OUTCOME = 'Success (drone ship)'
- %sql select sum(case when MISSION_OUTCOME like '%Success%' THEN 1 else 0 end) as SUCCESS , sum(case when MISSION_OUTCOME like '%Failure%' THEN 1 else 0 end) AS FAILURE from SPACEXTBL GROUP BY 0
- %sql SELECT COUNT(*), MISSION_OUTCOME FROM SPACEXTBL GROUP BY MISSION_OUTCOME HAVING (MISSION_OUTCOME LIKE '%Success%' OR MISSION_OUTCOME LIKE '%Failure%')
- %sql SELECT BOOSTER_VERSION FROM SPACEXTBL WHERE PAYLOAD_MASS__KG_ = (SELECT MAX(PAYLOAD_MASS__KG_) FROM SPACEXTBL)
- %sql SELECT YEAR(DATE)='2015', LANDING__OUTCOME,DATE, BOOSTER_VERSION, LAUNCH_SITE FROM SPACEXTBL WHERE LANDING__OUTCOME='Failure (drone ship)‘
- %sql SELECT COUNT(*), LANDING__OUTCOME FROM SPACEXTBL WHERE DATE BETWEEN '2010-06-04' AND '2017-03-20' GROUP BY LANDING__OUTCOME ORDER BY COUNT(LANDING__OUTCOME) DESC
- Add the GitHub URL of your completed EDA with SQL notebook, as an external reference and peer-review purpose
- https://github.com/hashirkhan-786/Data-Science-/blob/master/jupyter-labs-eda-sql-coursera-HASHIR.ipynb

Build an Interactive Map with Folium

- Featuregroup was created to add multiple launch sites to the map using the following code
- launch_sites = folium.map.FeatureGroup()
- Folium circular marker of all sites was added by fetching the lat and Long from the data

```
for lat, lng, in zip(launch_sites_df.Lat, launch_sites_df.Long):
launch_sites.add_child(folium.features.CircleMarker([lat,
lng],radius=5,color='#d35400',fill=True))
```

- These objects were added so we could mark the launch sites on the maps and find their locations
- https://github.com/hashirkhan-786/Data-Science-/blob/master/lab_jupyter_launch_site_location_hashir.ipynb

Build a Dashboard with Plotly Dash

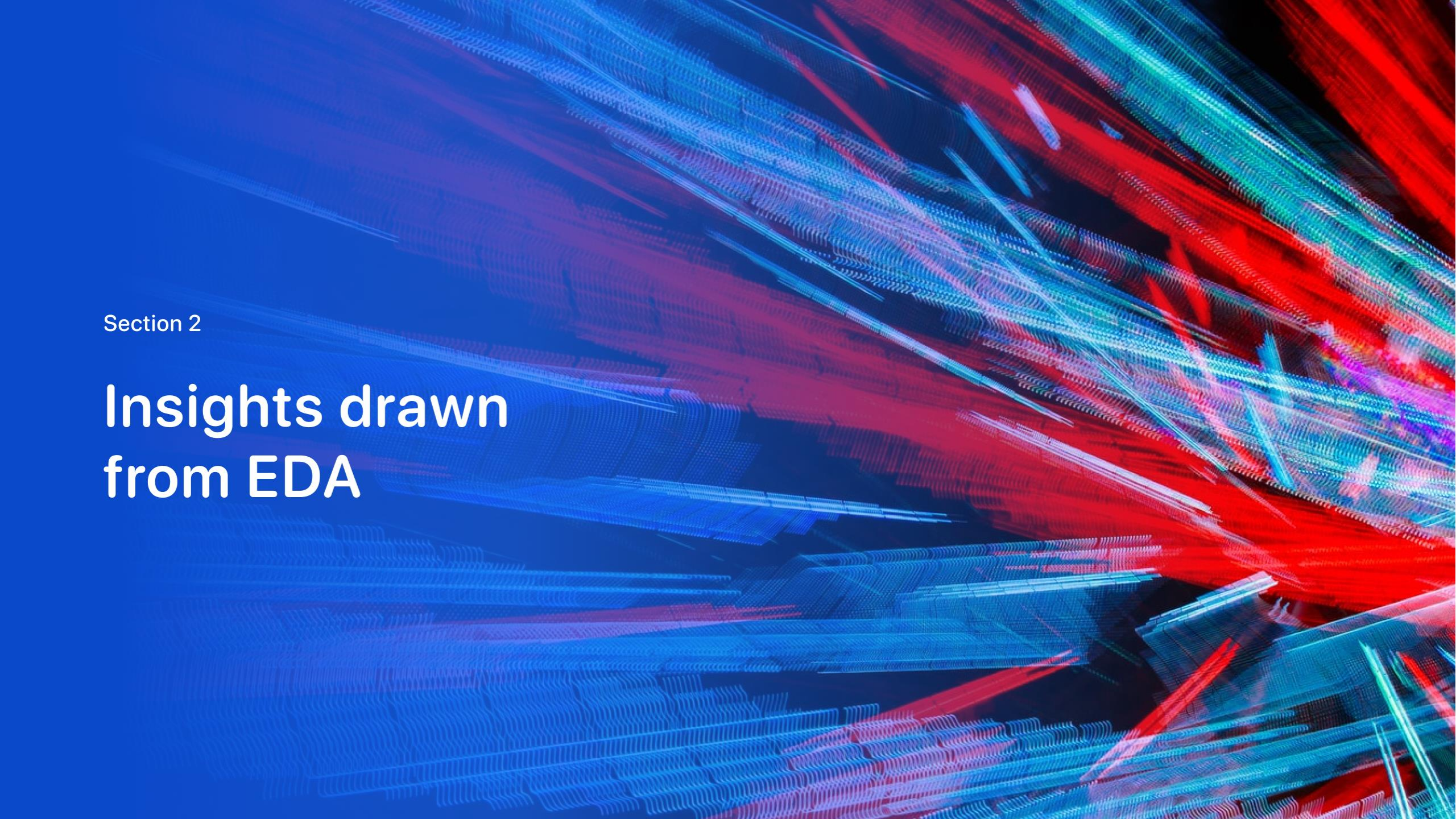
- In formulating the Dashboard scatter plot along with Pie charts were created.
- These plots were created so we could find the affect of payload on the outcome of the launch
- https://github.com/hashirkhan-786/Data-Science-/commit/50f57fc84ea6abbb1552ae49958cfae15cc1a4f5

Predictive Analysis (Classification)

- DataSet of Spacex was obtained from previous analysis and then it was standardized using standardscaler furthermore train and test Samples were created.
- 1) Logistic Regression
- 2) Support Vector Machine Object
- 3) Decision Tree Classifier
- 4) Confusion Matrix of All models and their accuracies
- Add the GitHub URL of your completed predictive analysis lab, as an external reference and peer-review purpose

Results

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

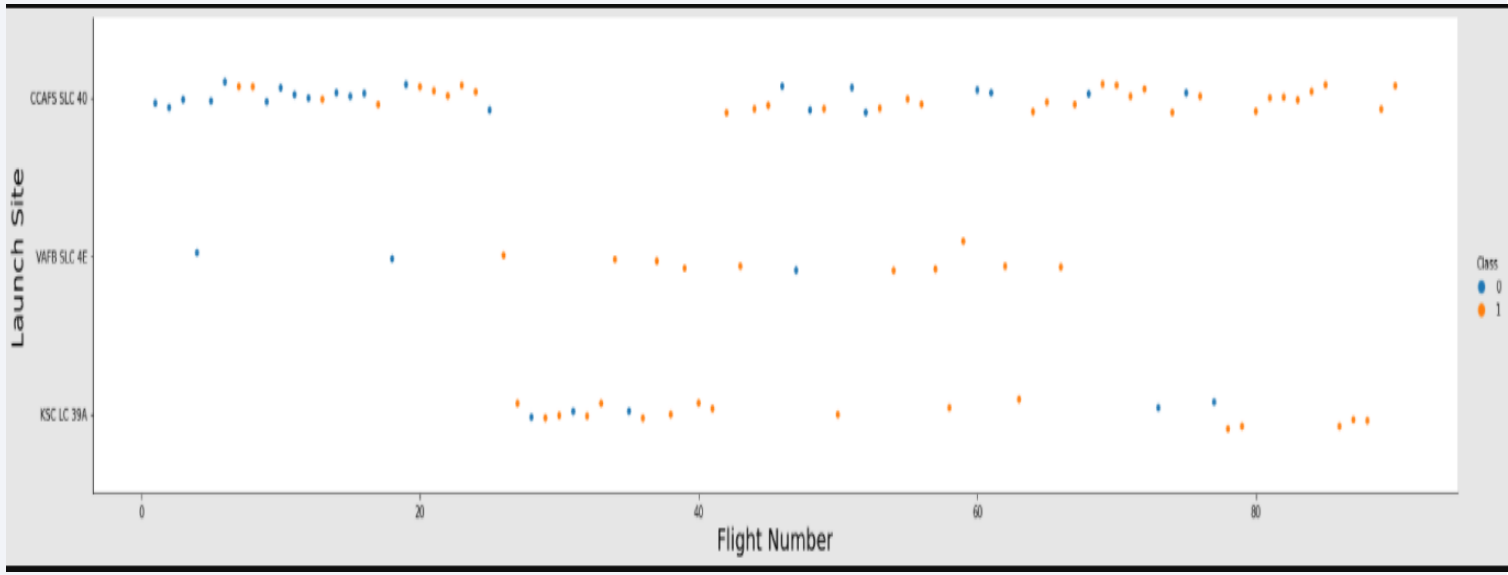


Section 2

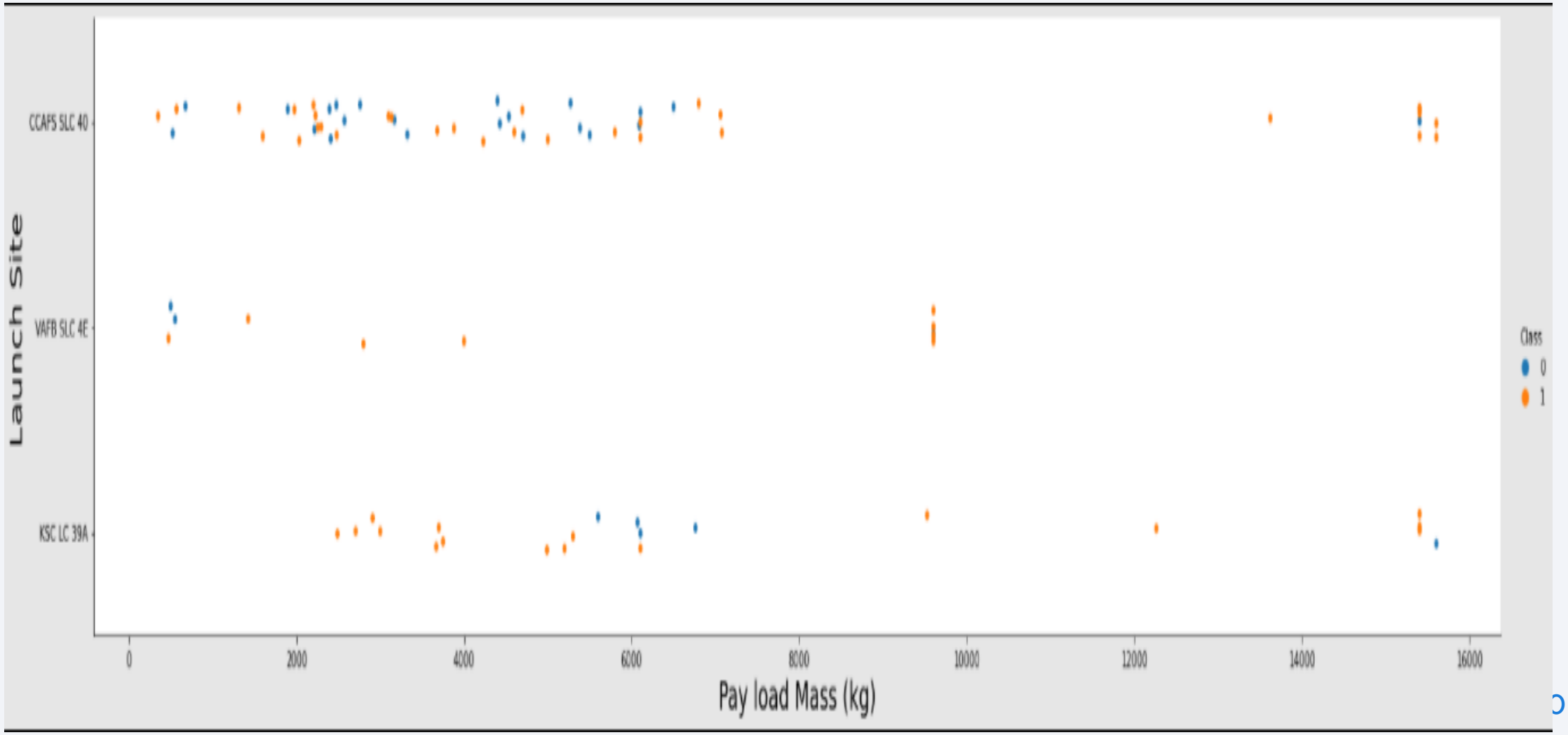
Insights drawn from EDA

Flight Number vs. Launch Site

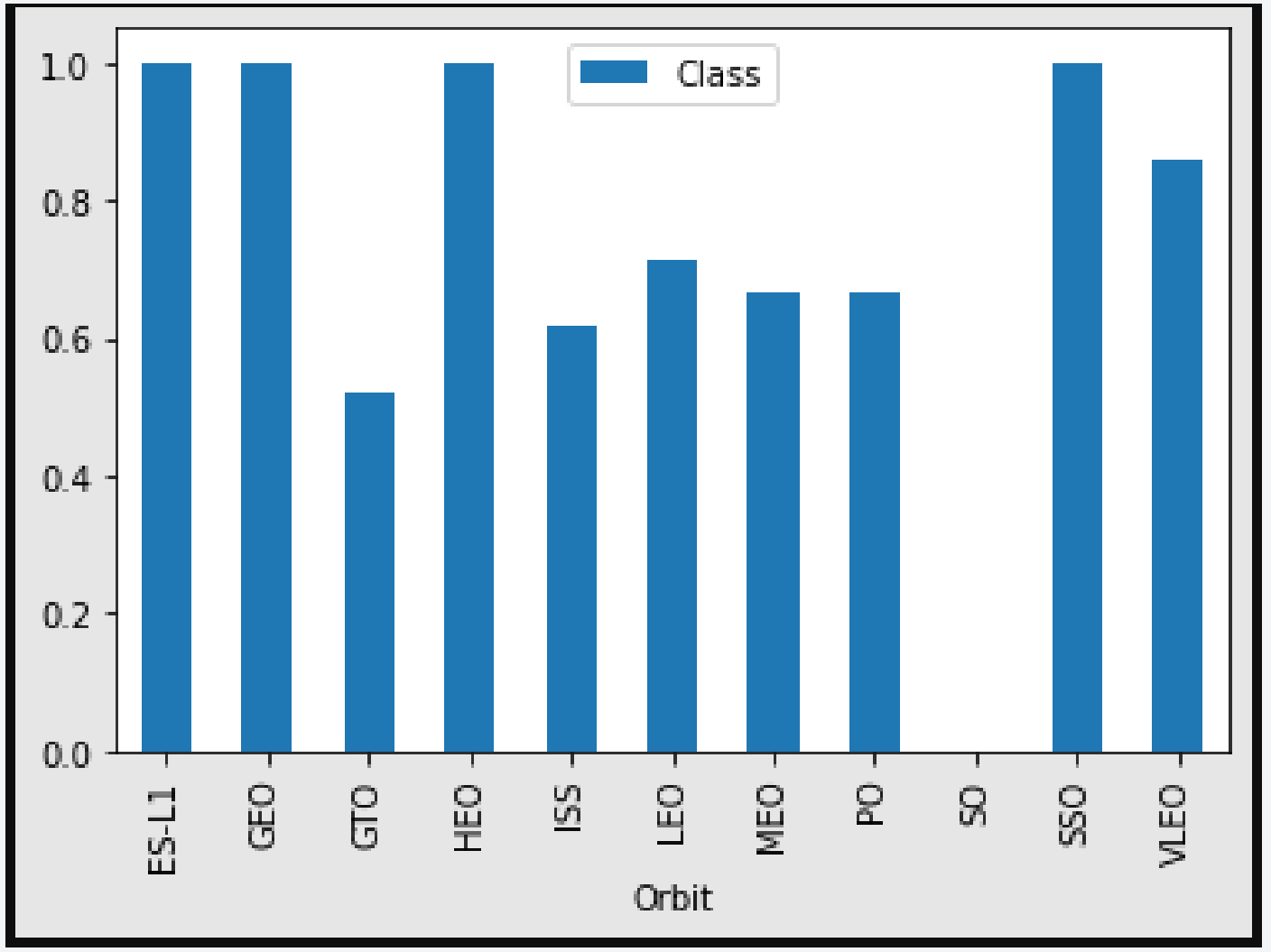
- Show a scatter plot of Flight Number vs. Launch Site
- Show the screenshot of the scatter plot with explanations



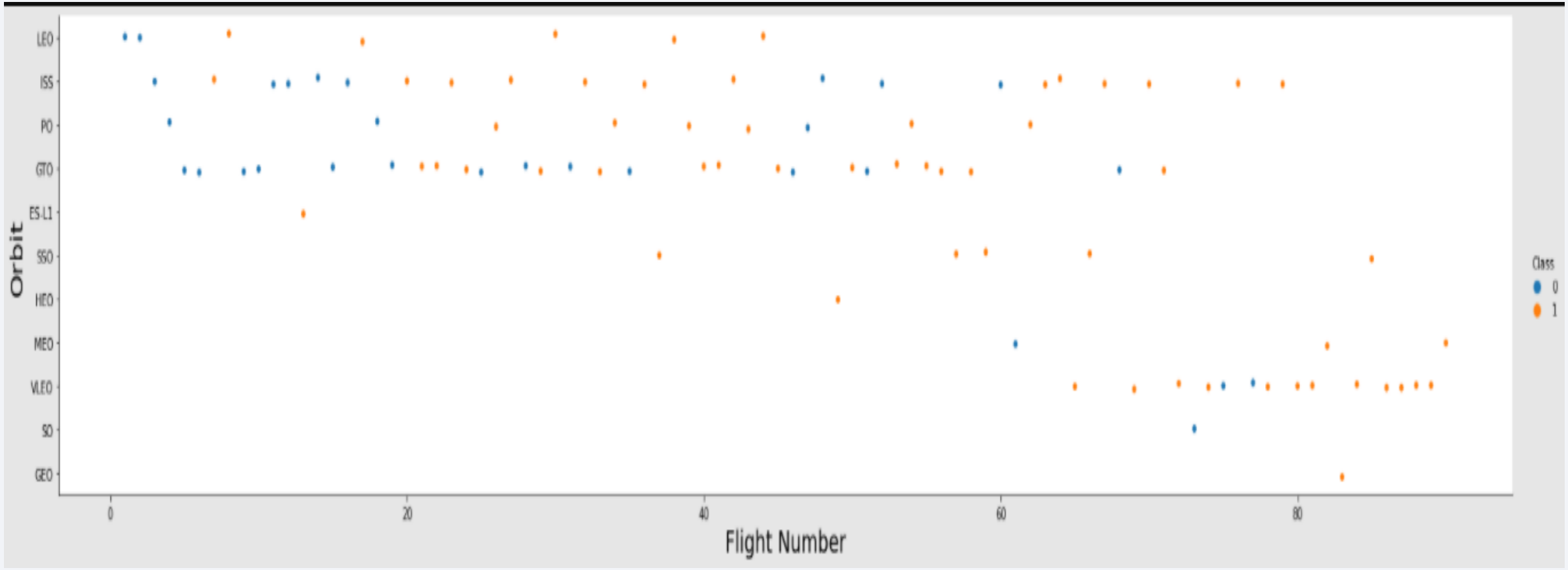
Payload vs. Launch Site



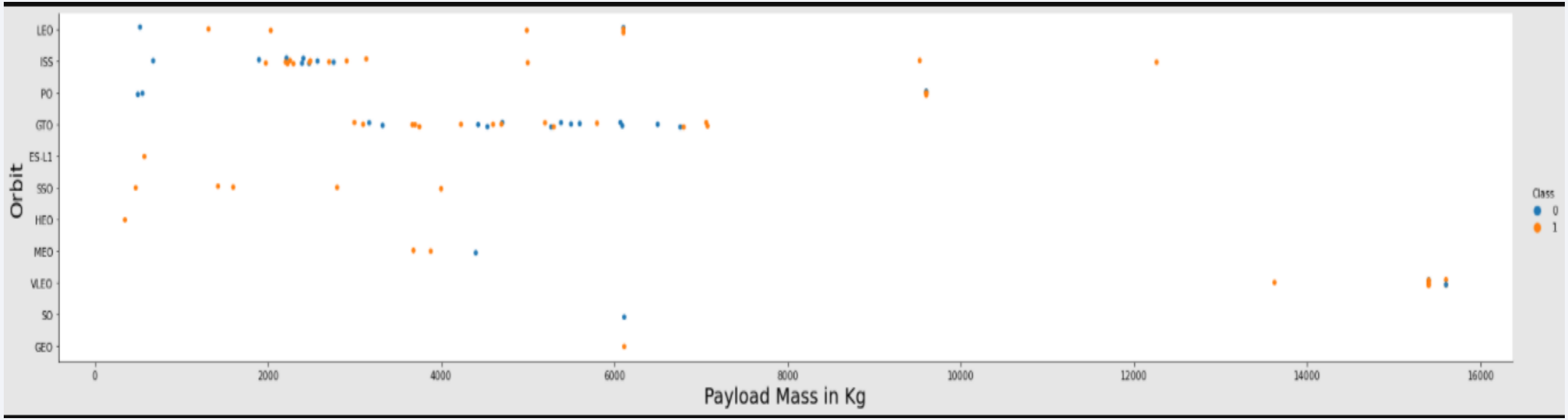
Success Rate vs. Orbit Type



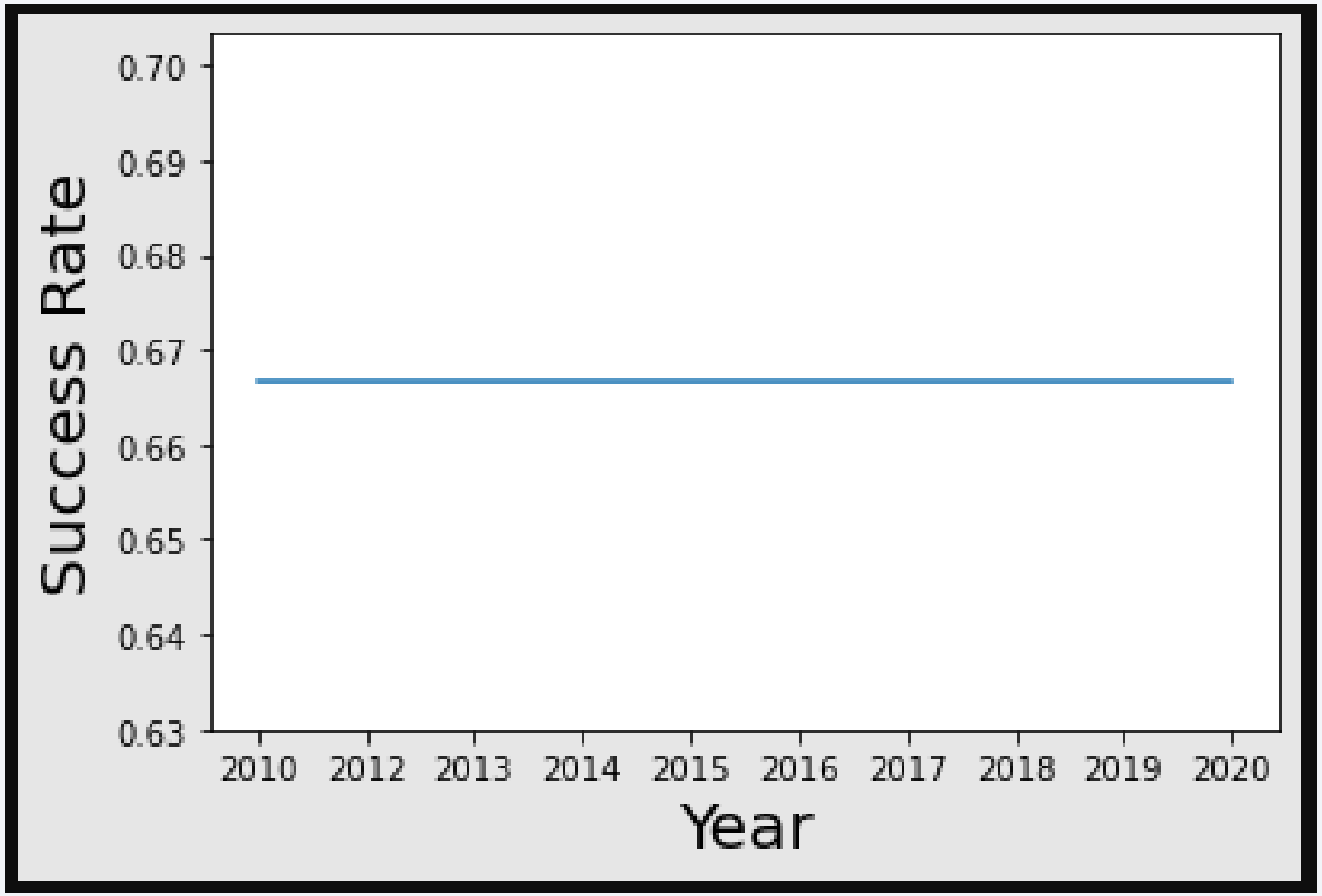
Flight Number vs. Orbit Type



Payload vs. Orbit Type



Launch Success Yearly Trend



All Launch Site Names

Task 1

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

[4]:
```
%sql SELECT DISTINCT LAUNCH_SITE from SPACEXTBL
```

```
 * ibm_db_sa://pxc73125:***@dashdb-txn-sbox-yp-dal09-11.services.dal.bluemix.net:50000/BLUDB
Done.
```

t[4]:

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

Launch Site Names Begin with 'CCA'

Task 2

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

In [5]:
```
%sql SELECT LAUNCH_SITE from SPACEXTBL WHERE LAUNCH_SITE like '%CCA%' LIMIT 5
```

```
 * ibm_db_sa://pxc73125:***@dashdb-txn-sbox-yp-dal09-11.services.dal.bluemix.net:50000/BLUDB
Done.
```

Out[5]:

launch_site
CCAFS LC-40
CCAFS LC-40
CCAFS LC-40
CCAFS LC-40
CCAFS LC-40

Total Payload Mass

Task 3

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

In [6]:
```
%sql SELECT SUM (PAYLOAD_MASS__KG_) FROM SPACEXTBL
```

```
 * ibm_db_sa://pxc73125:***@dashdb-txn-sbox-yp-dal09-11.services.dal.bluemix.net:50000/BLUDB
Done.
```

Out[6]:

1
619967

Average Payload Mass by F9 v1.1

Task 4

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

In [7]:
```
%sql SELECT AVG (PAYLOAD_MASS__KG_) FROM SPACEXTBL WHERE BOOSTER_VERSION = 'F9 v1.1'
```

```
 * ibm_db_sa://pxc73125:***@dashdb-txn-sbox-yp-dal09-11.services.dal.bluemix.net:50000/BLUDB
Done.
```

Out[7]:

1
2928.400000

First Successful Ground Landing Date

Task 5

List the date when the first successful landing outcome in ground pad was acheived.

Hint:Use min function

In [8]:
```
%sql SELECT MIN( DATE) FROM SPACEXTBL WHERE LANDING__OUTCOME = 'Success (ground pad)'
```

```
 * ibm_db_sa://pxc73125:***@dashdb-txn-sbox-yp-dal09-11.services.dal.bluemix.net:50000/BLUDB
Done.
```

Out[8]:

1
2015-12-22

Successful Drone Ship Landing with Payload between 4000 and 6000

```
In [9]: %sql SELECT BOOSTER_VERSION , 4000< PAYLOAD_MASS__KG_ <6000 FROM SPACEXTBL WHERE LANDING__OUTCOME = 'Success (drone ship)'
 * ibm_db_sa://pxc73125:***@dashdb-txn-sbox-yp-dal09-11.services.dal.bluemix.net:50000/BLUDB
Done.
```

Out[9]:

booster_version	2
F9 FT B1021.1	1
F9 FT B1022	0
F9 FT B1023.1	1
F9 FT B1026	0
F9 FT B1029.1	0
F9 FT B1021.2	0
F9 FT B1029.2	1
F9 FT B1036.1	0
F9 FT B1038.1	1
F9 B4 B1041.1	0
F9 FT B1031.2	0
F9 B4 B1042.1	1
F9 B4 B1045.1	1
F9 B5 B1046.1	1

Total Number of Successful and Failure Mission Outcomes

Task 7

List the total number of successful and failure mission outcomes

In [106]:
```
%sql select sum(case when MISSION_OUTCOME like '%Success%' THEN 1 else 0 end) as SUCCESS , sum(case when MISSION_OUTCOME like '%Failure%' THEN 1 else 0 end) AS FAILURE from SPACEXTBL GROUP BY 0
```

```
 * ibm_db_sa://pxc73125:***@dashdb-txn-sbox-yp-dal09-11.services.dal.bluemix.net:50000/BLUDB
Done.
```

Out[106]:

success	failure
100	1

Boosters Carried Maximum Payload

List the names of the booster_versions which have carried the maximum payload mass. Use a subquery

In [120]:
```
%sql SELECT BOOSTER_VERSION FROM SPACEXTBL WHERE PAYLOAD_MASS__KG_ = (SELECT MAX(PAYLOAD_MASS__KG_) FROM SPACEXTBL)
```

```
 * ibm_db_sa://pxc73125:***@dashdb-txn-sbox-yp-dal09-11.services.dal.bluemix.net:50000/BLUDB
Done.
```

Out[120]:

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

List the failed landing_outcomes in drone ship, their booster versions, and launch site names for in year 2015

In [145]:
```
%sql SELECT YEAR(DATE)='2015', LANDING__OUTCOME,DATE, BOOSTER_VERSION, LAUNCH_SITE FROM SPACEXTBL WHERE LANDING__OUTCOME='Failure (drone ship)'
```

```
 * ibm_db_sa://pxc73125:***@dashdb-txn-sbox-yp-dal09-11.services.dal.bluemix.net:50000/BLUDB
Done.
```

Out[145]:

1	landing__outcome	DATE	booster_version	launch_site
1	Failure (drone ship)	2015-01-10	F9 v1.1 B1012	CCAFS LC-40
1	Failure (drone ship)	2015-04-14	F9 v1.1 B1015	CCAFS LC-40
0	Failure (drone ship)	2016-01-17	F9 v1.1 B1017	VAFB SLC-4E
0	Failure (drone ship)	2016-03-04	F9 FT B1020	CCAFS LC-40
0	Failure (drone ship)	2016-06-15	F9 FT B1024	CCAFS LC-40

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

In [166]:
```
%sql SELECT COUNT(*), LANDING__OUTCOME FROM SPACEXTBL WHERE DATE BETWEEN '2010-06-04' AND '2017-03-20' GROUP BY LANDING__OUTCOME O
RDER BY COUNT(LANDING__OUTCOME) DESC
```

```
 * ibm_db_sa://pxc73125:***@dashdb-txn-sbox-yp-dal09-11.services.dal.bluemix.net:50000/BLUDB
Done.
```

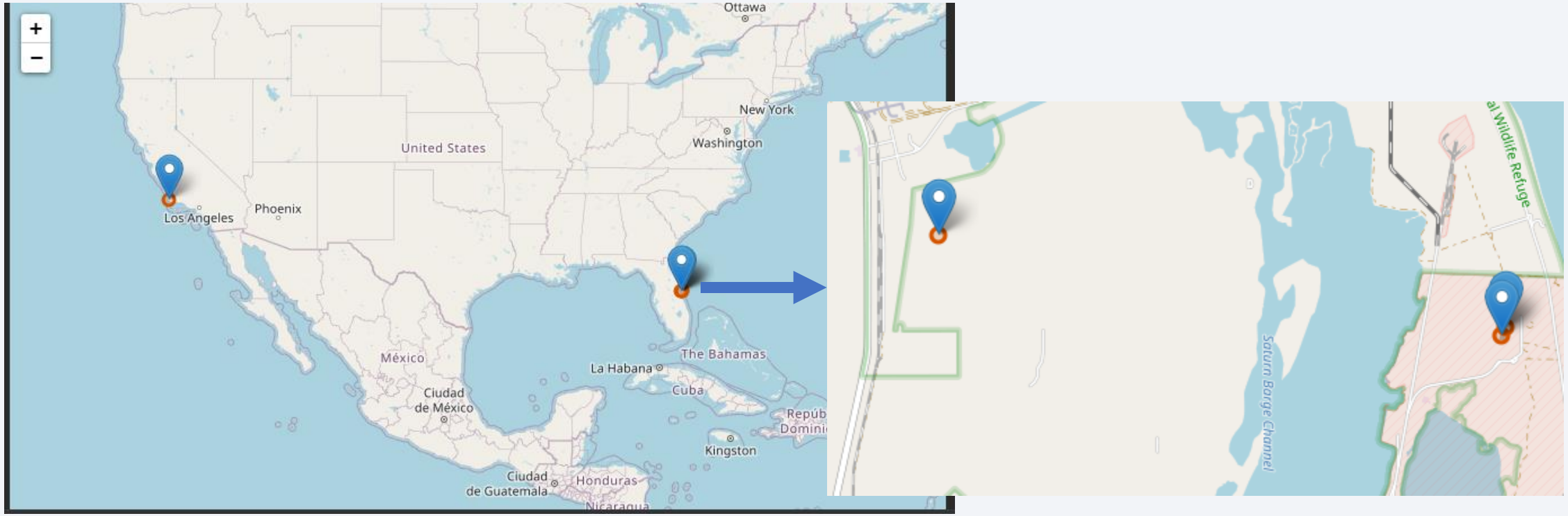
Out[166]:

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

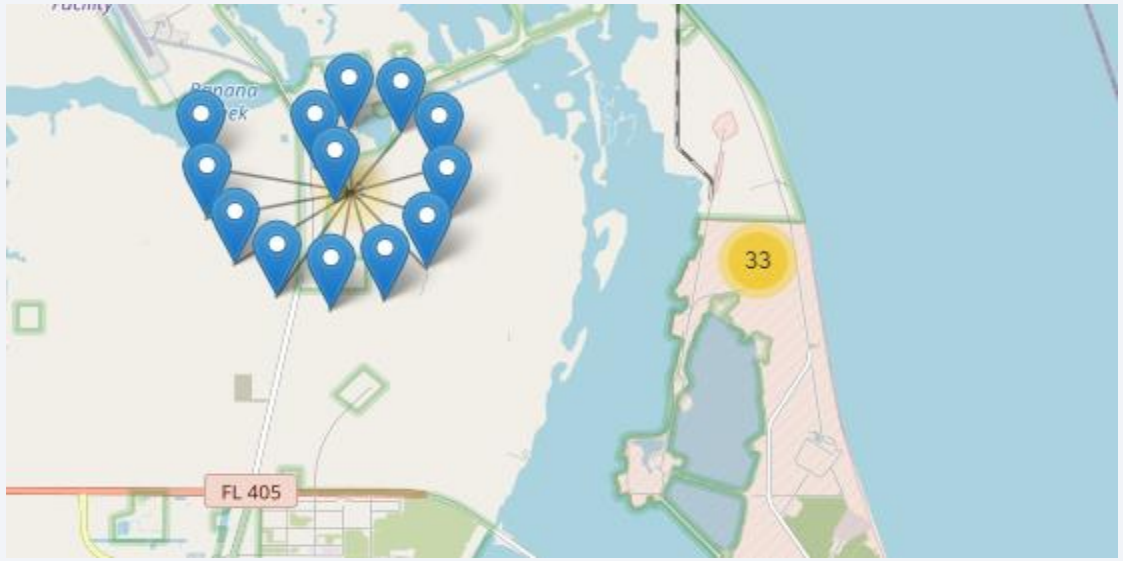
Section 4

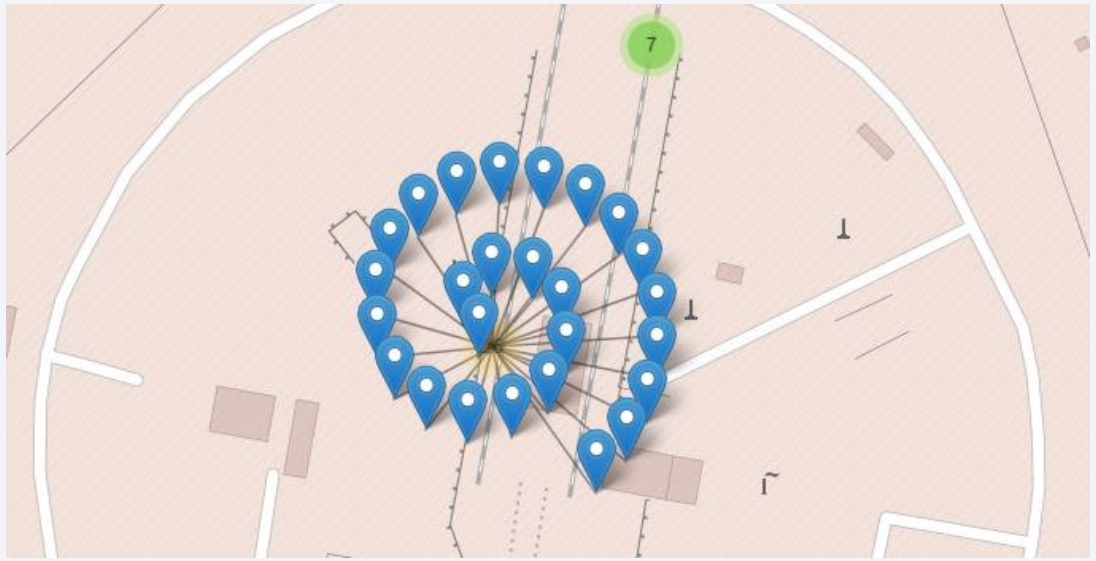
Launch Sites Proximities Analysis

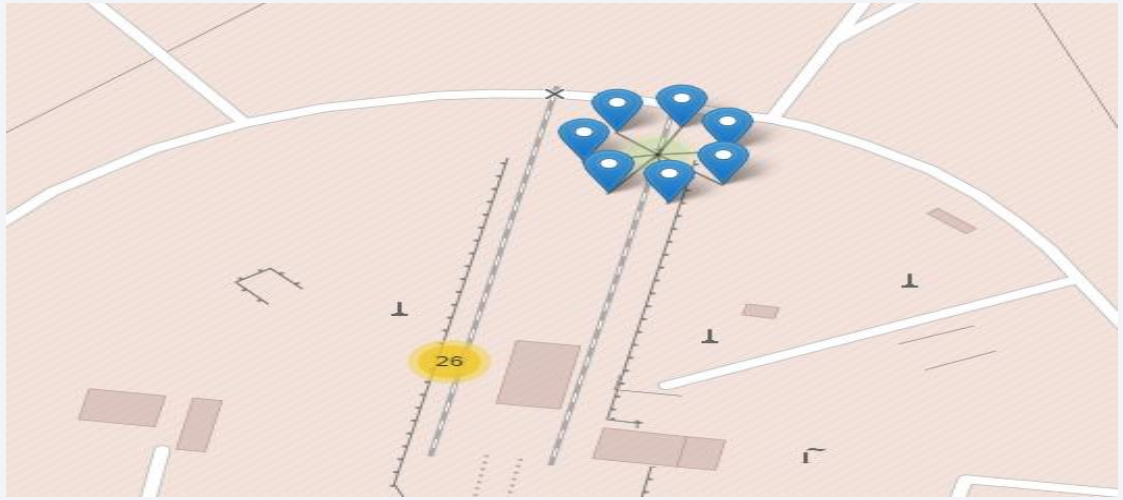
<SpaceX Launch Sites>



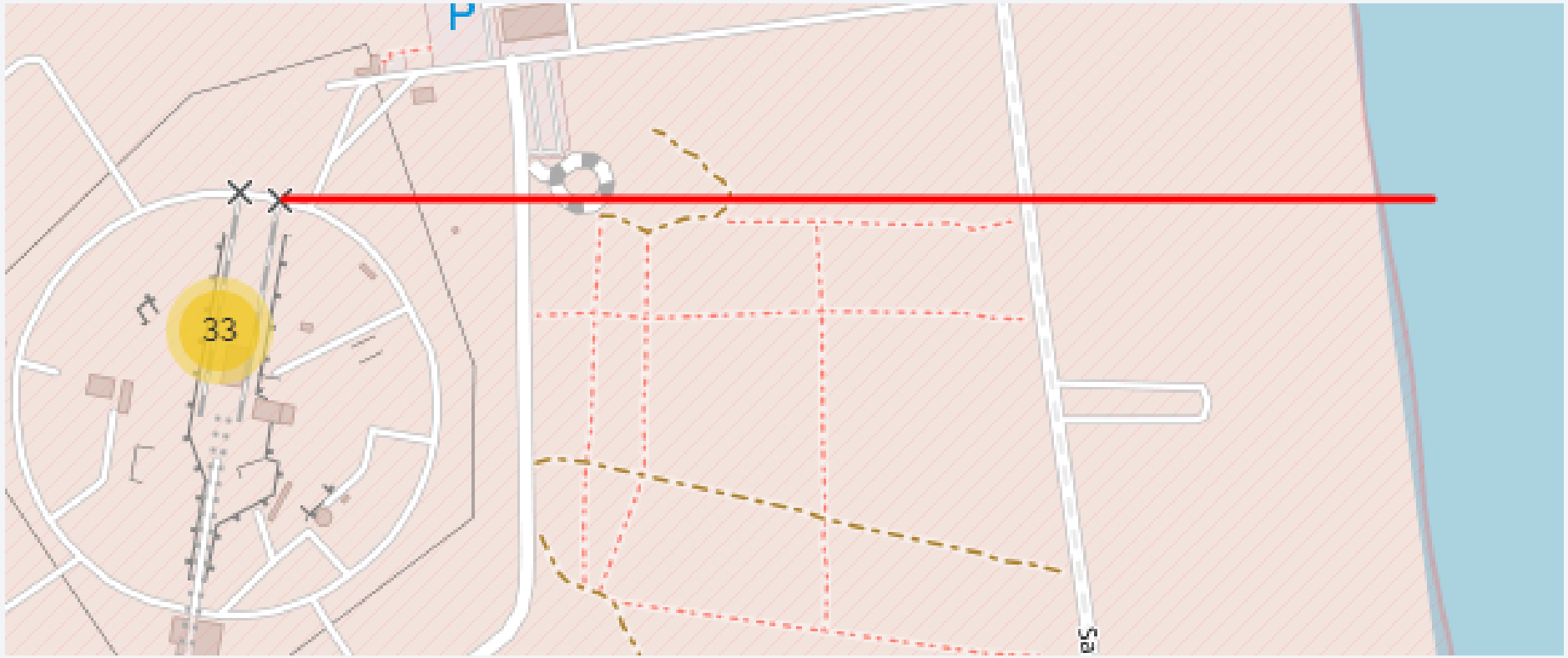
<Successful and Unsuccessful LaunchSites>

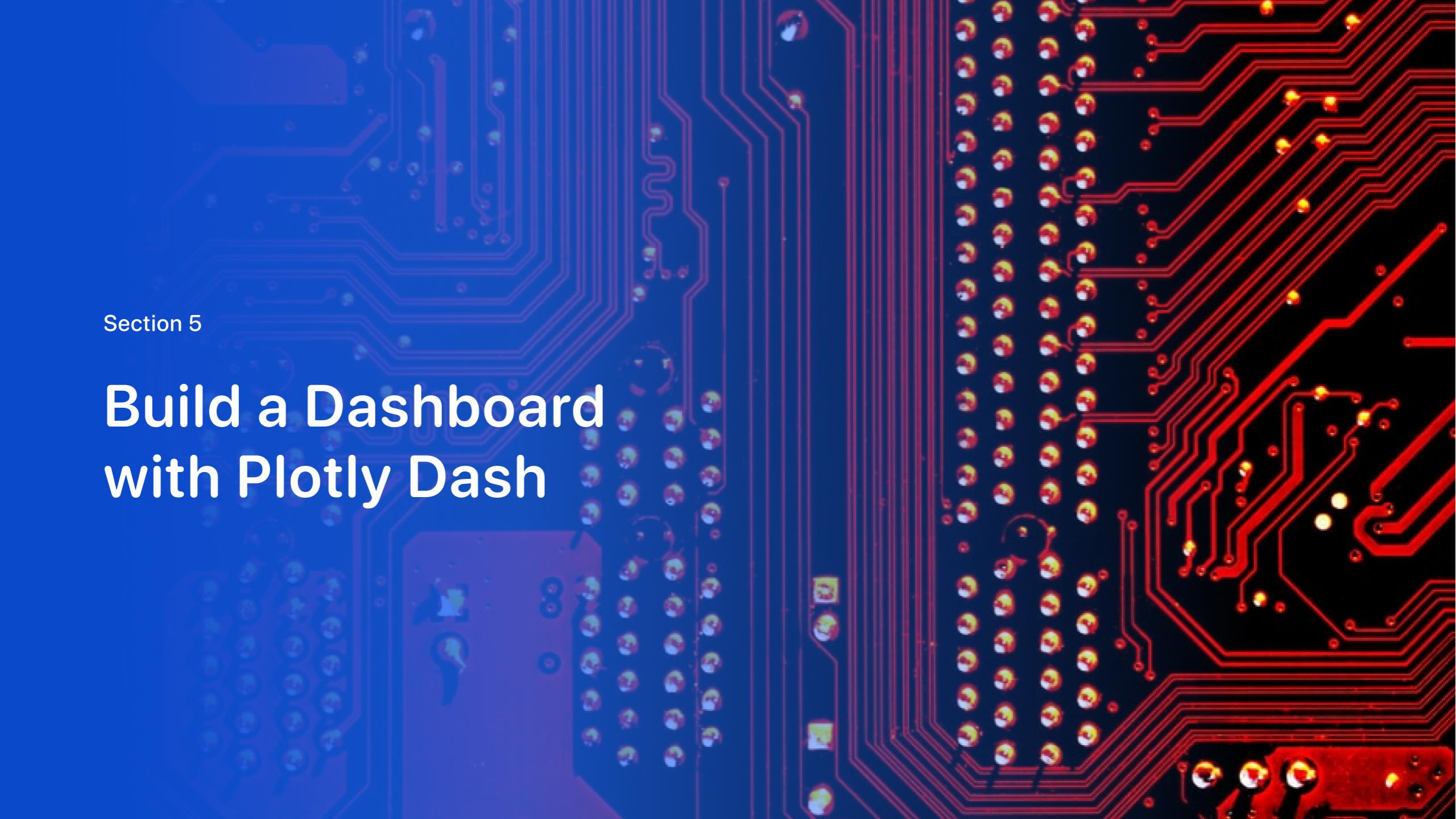






<Nearest LaunchSite to the Railway>

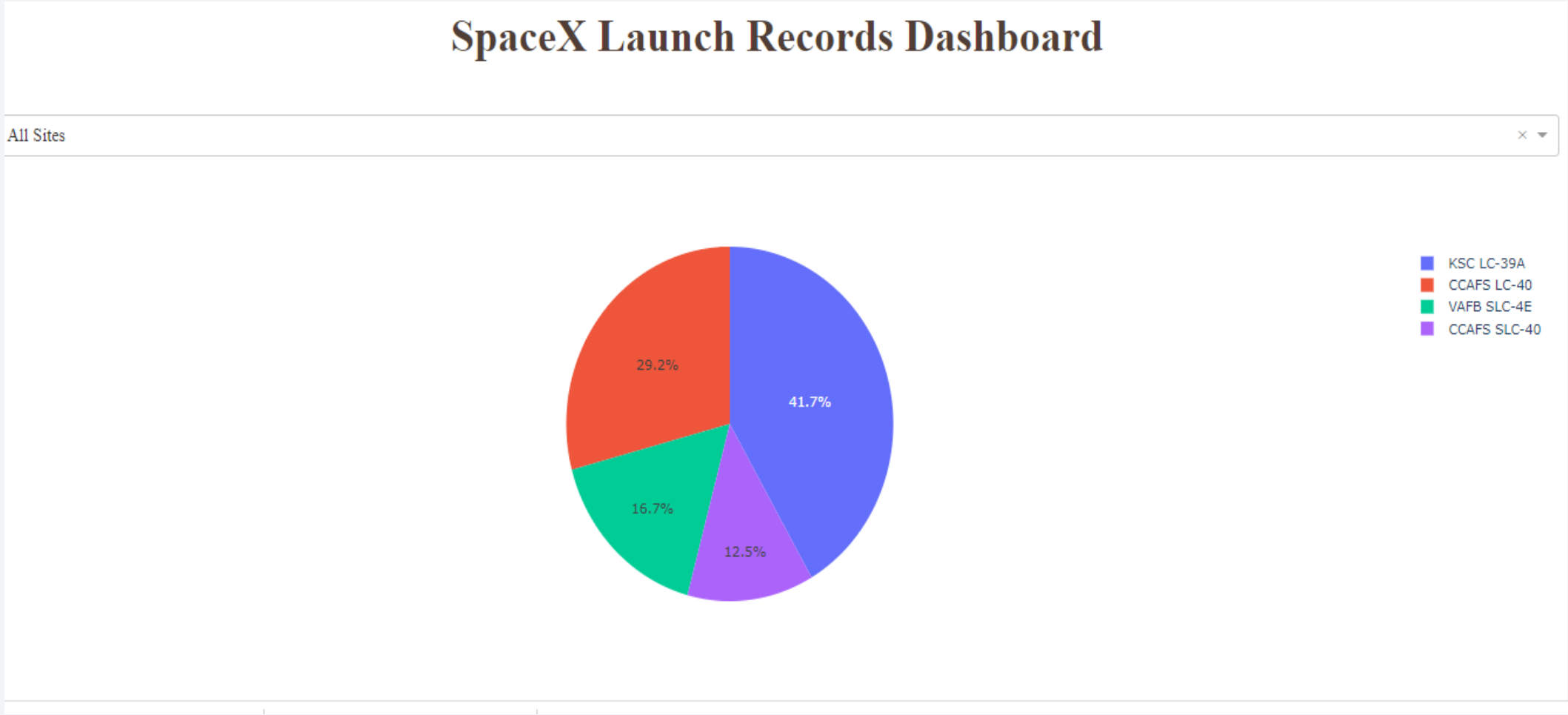




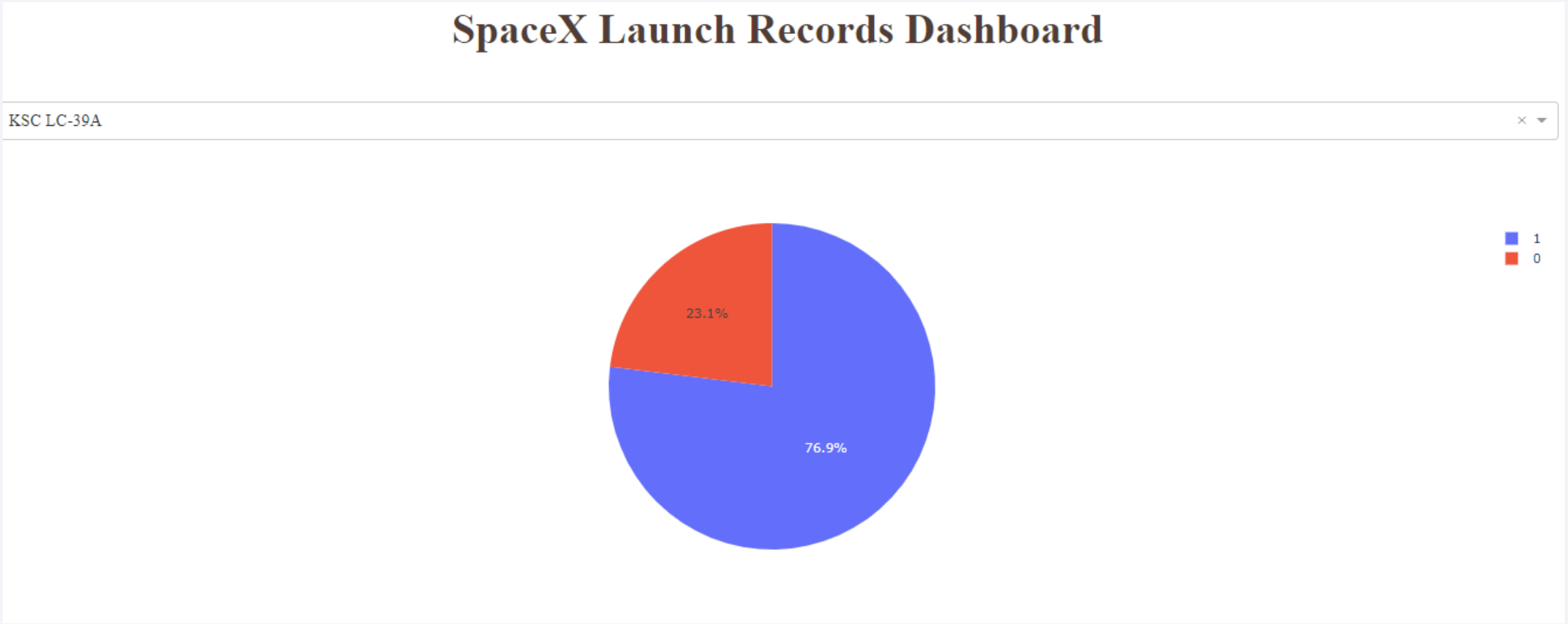
Section 5

Build a Dashboard with Plotly Dash

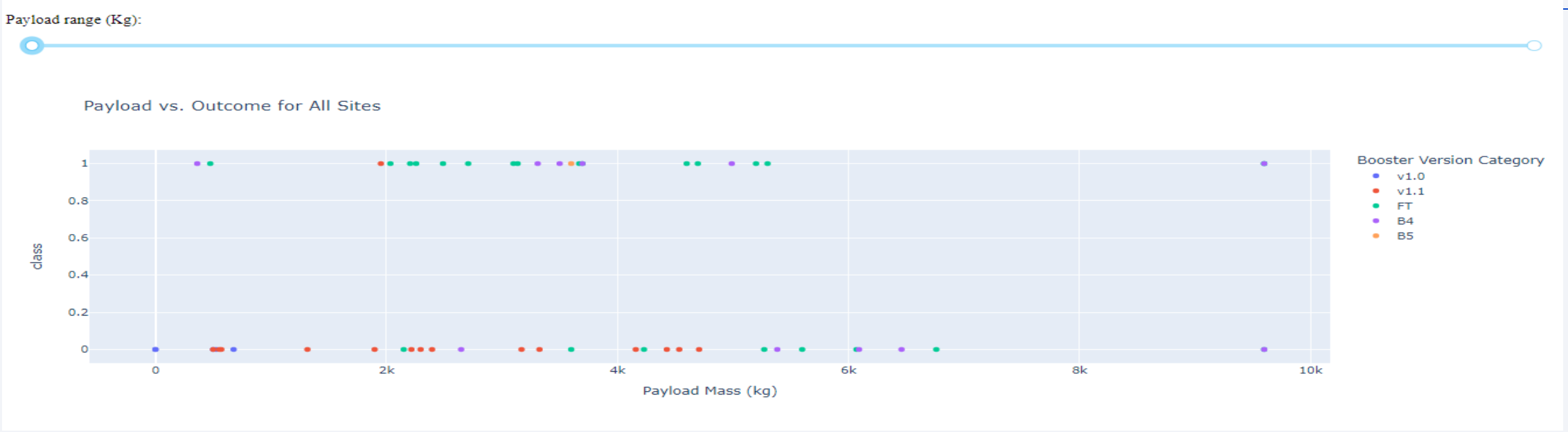
<All Launch Sites>



<KSC-LC 39A With Highest Launch Success>



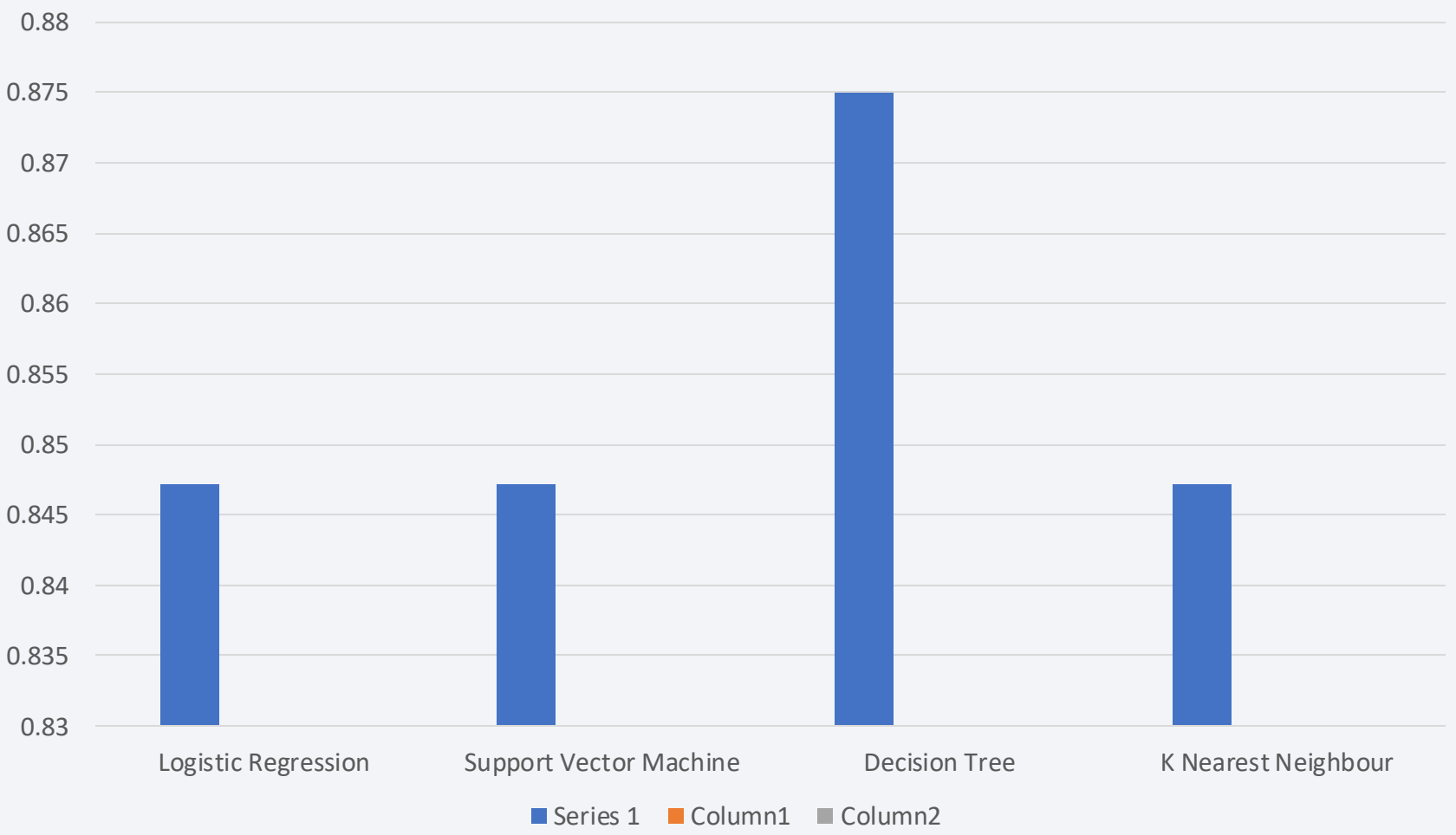
<Different Payloads and Outcomes>



Section 6

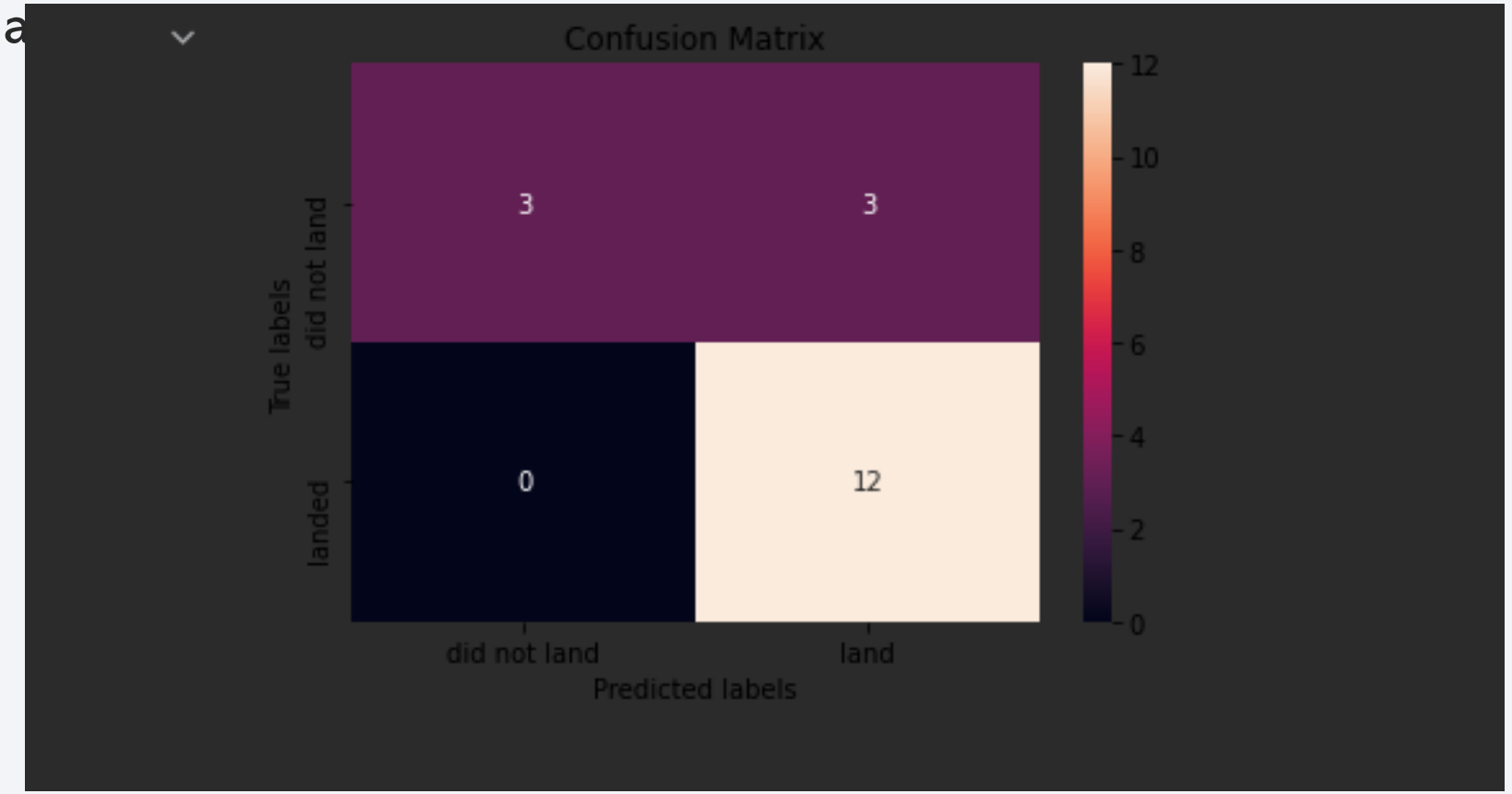
Predictive Analysis (Classification)

Classification Accuracy



Confusion Matrix

- Show the confusion matrix of the best performing model with an expla



Conclusions

- Complete Analysis of SpaceX Rockets Launch Sites was carried out
- Different Launch Sites with Different Payloads were compared with eachother in order to find the payload version with Max number of successes
- Outcomes of different launch sites have been compared to find the best launch site
- Different orbits against payloads were compared to find the most suited orbit for the launch of the rocket

Appendix

- https://github.com/hashirkhan-786/Data-Science-/tree/master

- This includes all Python Files and Codes used during the completion of this project

Thank you!