Winning Space Race with Data Science

Duy Pham
June 2024

Outline

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

Executive Summary

- We have enough data to process and build insights
- We can predict correctly the success rate of landing the first stage

Introduction

- 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
- What is the probability that the first stage will land?
  - Then we can determine the cost of a launch
  - Our company wants to bid against SpaceX for a rocket launch

Section 1

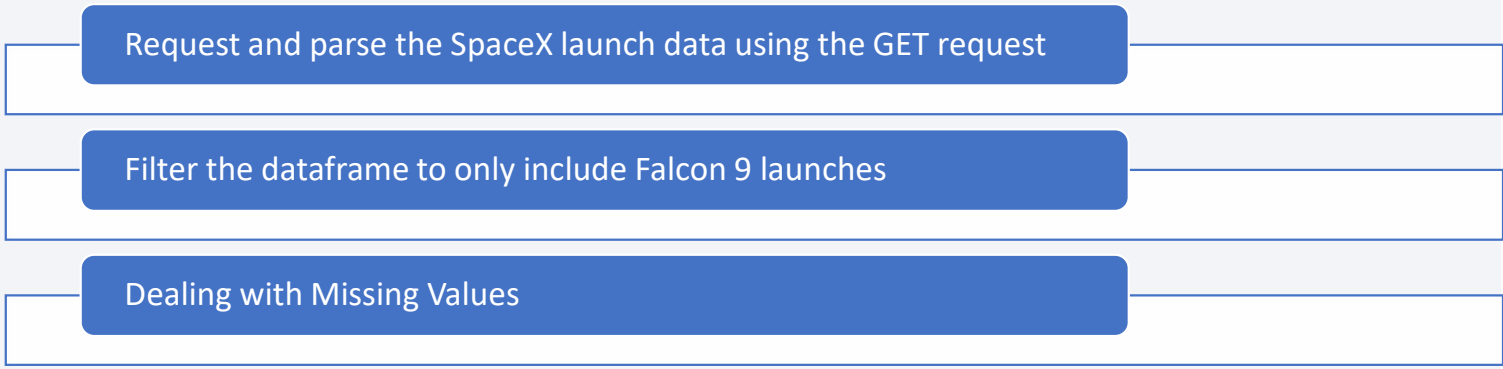
Methodology

Methodology

Executive Summary

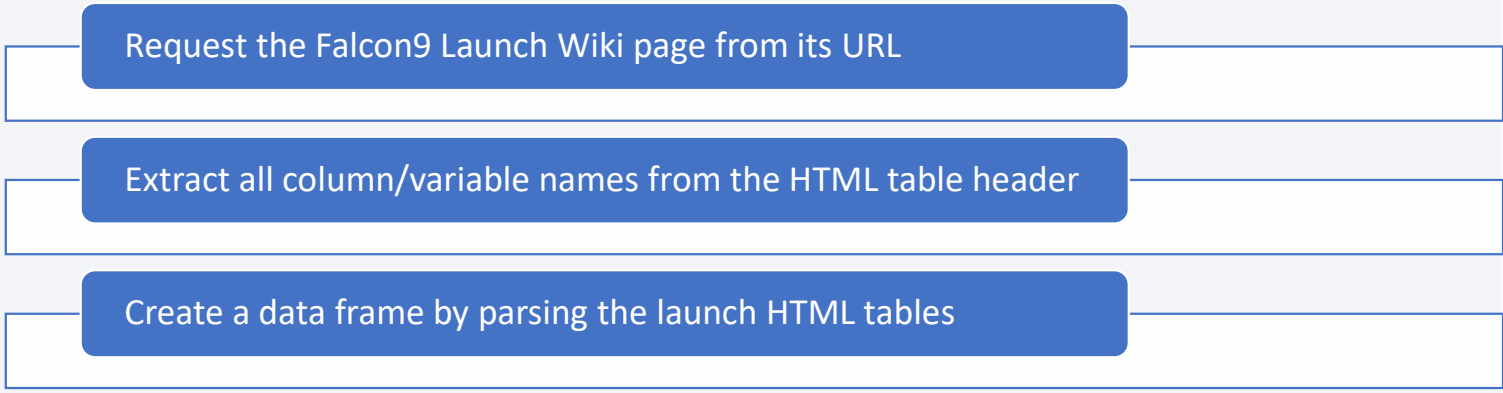
- Data collection methodology:
  - Request to the SpaceX API and scrape records from Wikipedia
- Perform data wrangling
  - Perform exploratory Data Analysis and determine Training Labels
- Perform exploratory data analysis (EDA) using visualisation and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - Find the best Hyperparameter for SVM, Classification Trees and Logistic Regression
  - Find the method that performs best using test data

Data Collection – SpaceX API



- SpaceX URL
  - https://api.spacexdata.com/v4/launches/past
- SpaceX API calls notebook
  - https://github.com/phamngocduy/data-science-capstone/blob/main/jupyter-labs-spacex-data-collection-api.ipynb

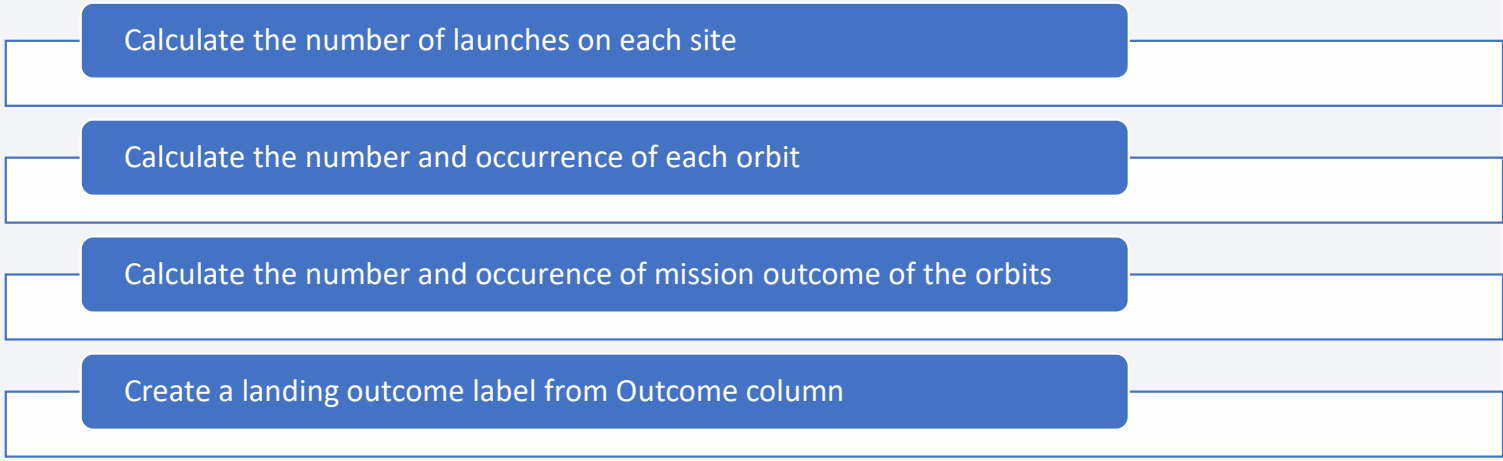
Data Collection - Scraping



- List of Falcon 9 and Falcon Heavy launches from Wikipage
  - https://en.wikipedia.org/w/index.php?title=List_of_Falcon_9_and_Falcon_Heavy_launches
- Web scraping notebook
  - https://github.com/phamngocduy/data-science-capstone/blob/main/jupyter-labs-webscraping.ipynb

Data Wrangling

- Data wrangling notebooks
  - https://github.com/phamngocduy/data-science-capstone/blob/main/labs-jupyter-spacex-Data%20wrangling.ipynb



EDA with SQL

- SQL queries
  - Display the names of the unique launch sites in the space mission
  - Display 5 records where launch sites begin with the string 'CCA'
  - 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 when the first successful landing outcome in the ground pad was achieved
  - List the names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000
  - List the total number of successful and failure mission outcomes
  - List the names of the booster_versions which have carried the maximum payload mass
  - List the records which will display the month names, failure landing_outcomes in drone ship, booster versions, launch_site for the months in year 2015
  - 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
- EDA with SQL notebook
  - https://github.com/phamngocduy/data-science-capstone/blob/main/jupyter-labs-eda-sql-coursera_sqllite.ipynb

Build an Interactive Map with Folium

- Map objects
  - Mark all launch sites on a map
  - Mark the success/failed launches for each site on the map
  - Calculate the distances between a launch site to its proximities
- For finding some geographical patterns about launch sites
- Interactive map with Folium map
  - https://github.com/phamngocduy/data-science-capstone/blob/main/lab_jupyter_launch_site_location.ipynb

Build a Dashboard with Plotly Dash

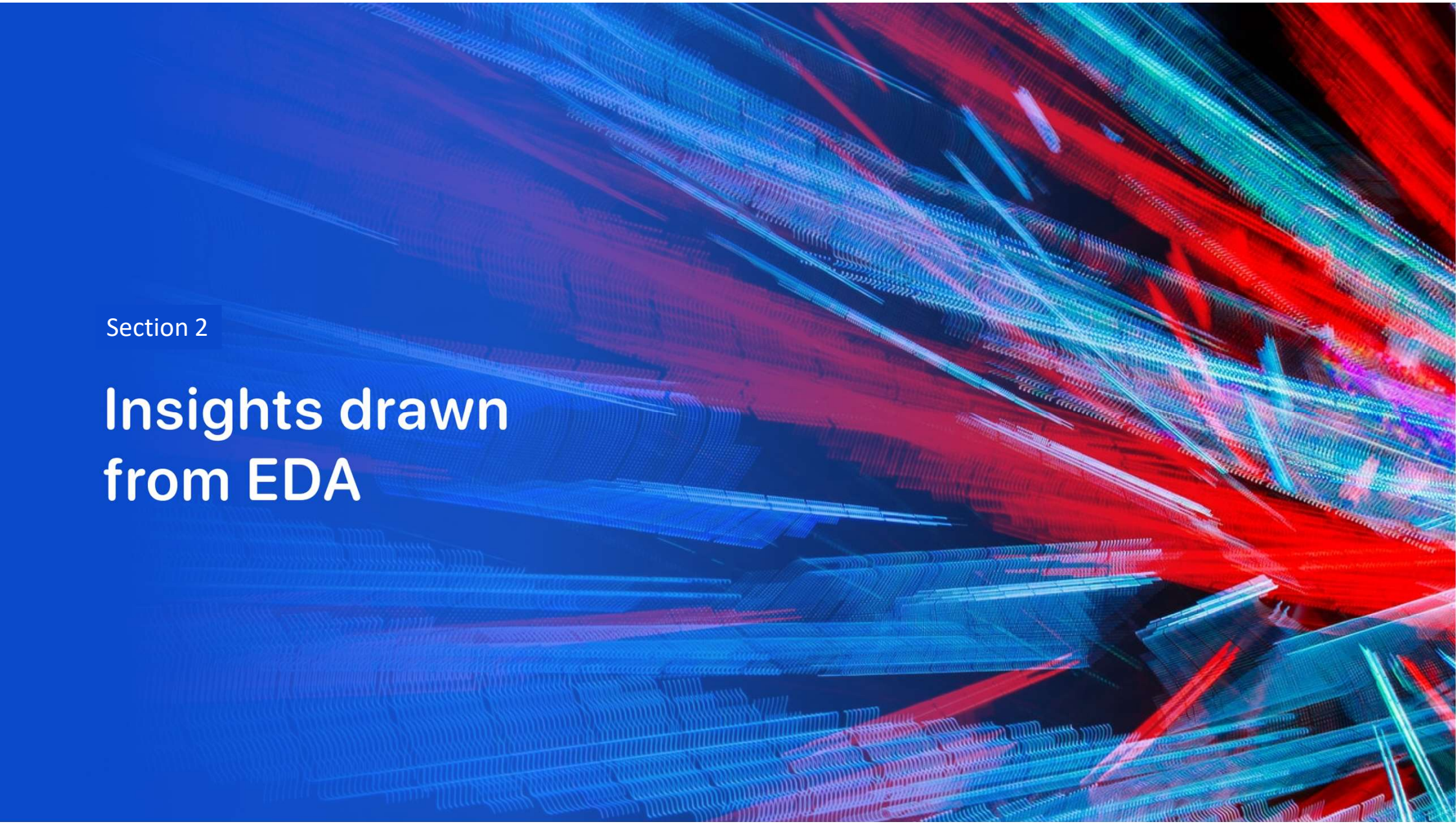
- Dashboard
  - Pie chart graph to show the success (class=1) count and failed (class=0) count for the selected site or all sites
  - Scatter chart to show correlations between Payload Mass (kg) and class for the selected site or all sites
- Insights
  - Which site has the highest launch success rate?
  - Which payload range(s) has the highest launch success rate?
- Plotly Dash lab
  - https://github.com/phamngocduy/data-science-capstone/blob/main/spacex_dash_app.py

Predictive Analysis (Classification)

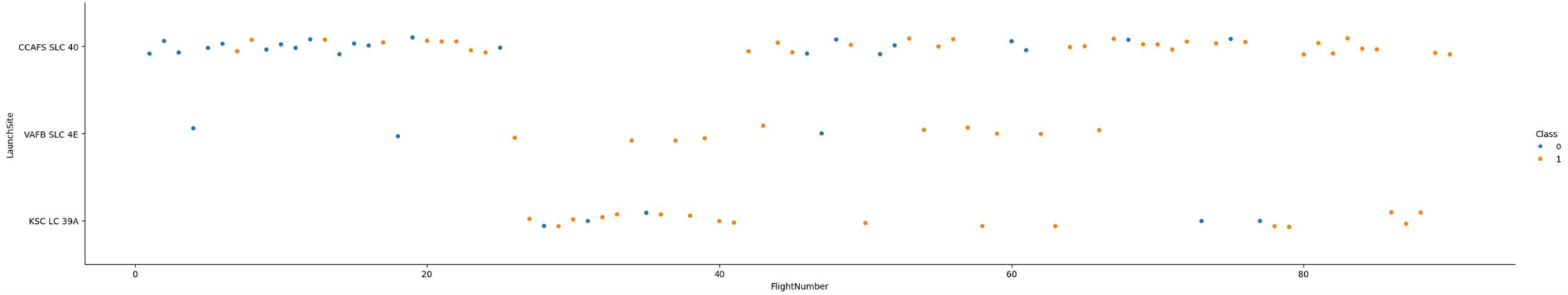
- Perform exploratory Data Analysis and determine Training Labels
  - Create a column for the class
  - Standardise the data
  - Split into training data and test data
- Find best Hyperparameter for SVM, Classification Trees and Logistic Regression
  - Use GridSearchCV to find the best parameters
  - Find the method that performs best using test data
- Predictive analysis lab
  - https://github.com/phamngocduy/data-science-capstone/blob/main/SpaceX_Machine%20Learning%20Prediction_Part_5.ipynb

Results

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

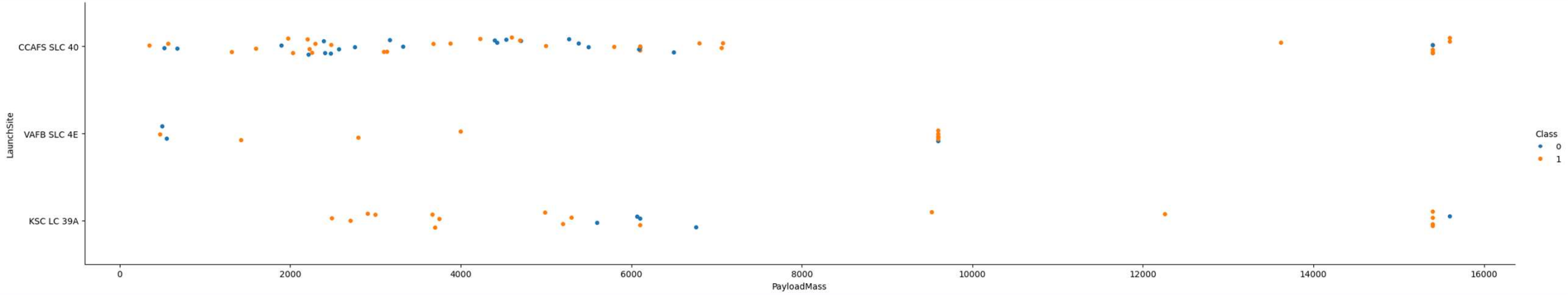

Section 2
Insights drawn
from EDA

Flight Number vs. Launch Site



- VAFB SLC 4E has more success than other sites

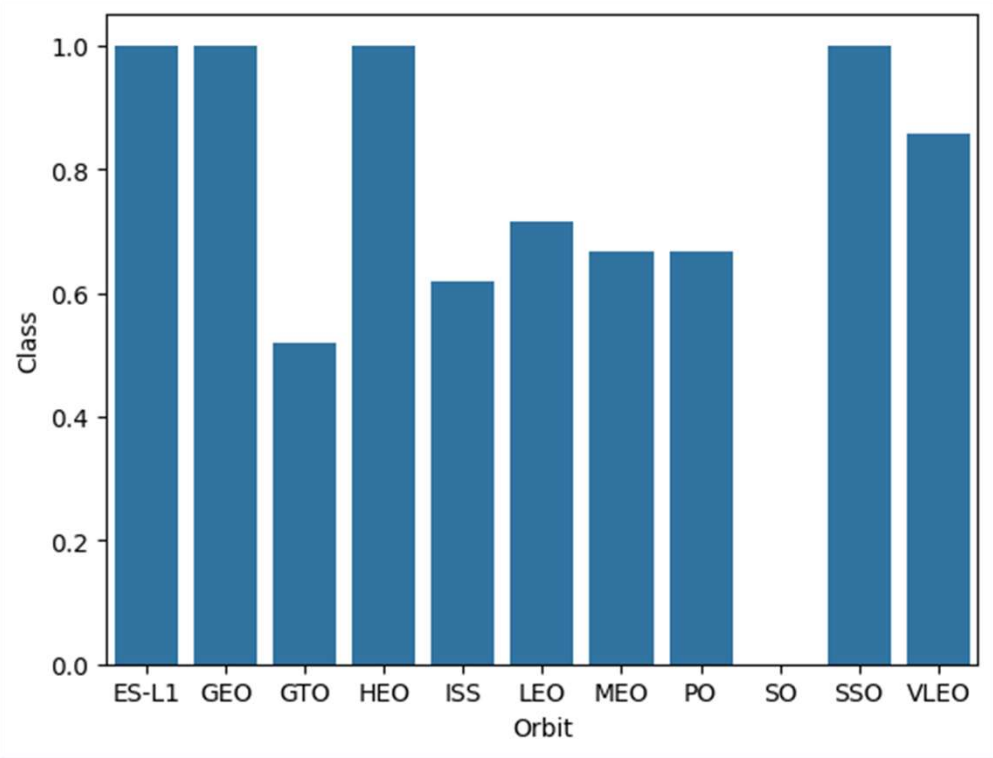
Payload vs. Launch Site



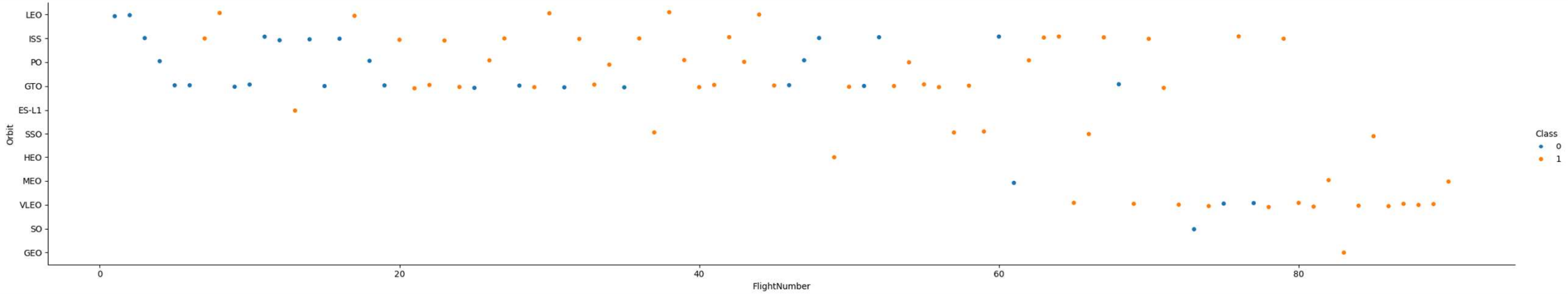
- CCAFS SLC 40 has less payload compared to other sites
- Higher payloads have more success rates

Success Rate vs. Orbit Type

- ES-L1, GEO, HEO, and SSO have a success rate of 1

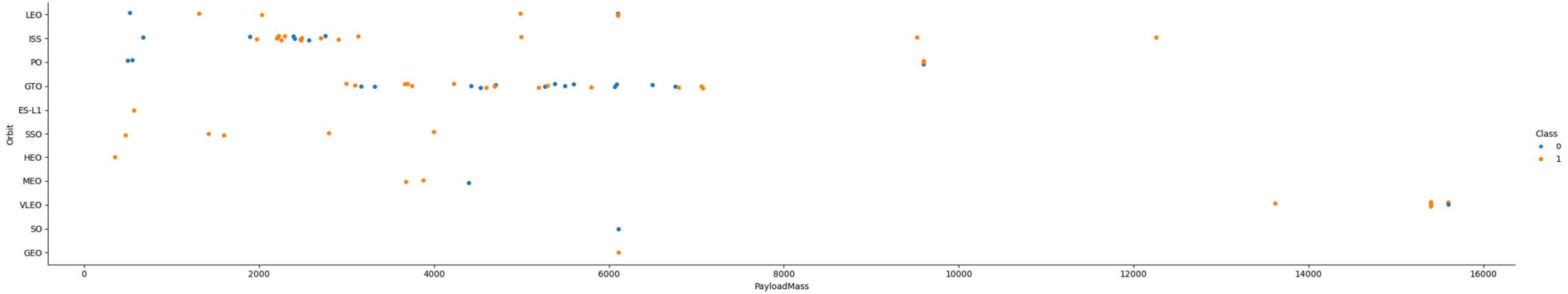


Flight Number vs. Orbit Type



- MEO, VLEO, SO, and GEO have flights recently

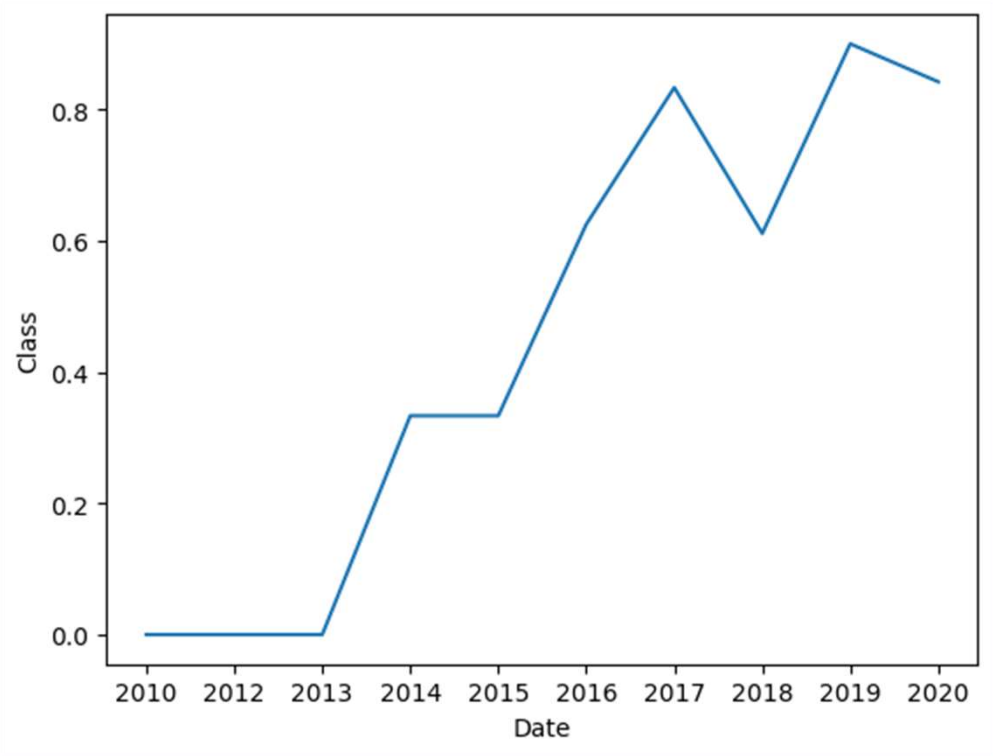
Payload vs. Orbit Type



- VLEO has more payload compared to others

Launch Success Yearly Trend

- Success rate increases over the years, except 2018



All Launch Site Names

- Find the names of the unique launch sites
- There are 4 Sites

```
%sql select distinct Launch_Site from SPACEXTABLE
 * 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`
- Most of Landing Outcome are Failure

```
%sql select * from SPACEXTABLE 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	Landing_Outcome
2010-06-04	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success	Failure (parachute)
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 (parachute)
2012-05-22	7:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success	No attempt
2012-10-08	0:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500	LEO (ISS)	NASA (CRS)	Success	No attempt
2013-03-01	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677	LEO (ISS)	NASA (CRS)	Success	No attempt

Total Payload Mass

- Calculate the total payload carried by boosters from NASA
- Total is 107 tons

```
%sql select sum(PAYLOAD_MASS__KG_) from SPACEXTABLE where Customer like '%NASA%'
```

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

sum(PAYLOAD_MASS__KG_)
107010

Average Payload Mass by F9 v1.1

- Calculate the average payload mass carried by booster version F9 v1.1
- About 2.5 tons

```
%sql select avg(PAYLOAD_MASS__KG_) from SPACEXTABLE where Booster_Version like 'F9 v1.1%'
```

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

avg(PAYLOAD_MASS__KG_)
2534.6666666666665

First Successful Ground Landing Date

- Find the dates of the first successful landing outcome on the ground pad
- Before 2015, all are Failures

```
%sql select min(Date) from SPACEXTABLE where Landing_Outcome = 'Success (ground pad)'
```

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

min(Date)
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
- There are 4 versions

```
%sql select Booster_Version from SPACEXTABLE where Landing_Outcome='Success (drone ship)' and PAYLOAD_MASS__KG_>4000 and PAYLOAD_MASS__KG_<6000
```

```
 * 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
- The success rate is really good

```
%sql select Landing_Outcome, count(*) from SPACEXTABLE group by Landing_Outcome
```

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

Landing_Outcome	count(*)
Controlled (ocean)	5
Failure	3
Failure (drone ship)	5
Failure (parachute)	2
No attempt	21
No attempt	1
Precluded (drone ship)	1
Success	38
Success (drone ship)	14
Success (ground pad)	9
Uncontrolled (ocean)	2

Boosters Carried Maximum Payload

```
%sql select Booster_Version from SPACEXTABLE where PAYLOAD_MASS__KG_ = (select max(PAYLOAD_MASS__KG_) from SPACEXTABLE)
```

```
 * 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

- List the names of the booster which have carried the maximum payload mass
- F9 B5 version can carry large payload

2015 Launch Records

- List the failed landing_outcomes in drone ship, their booster versions, and launch site names for in year 2015
- There are 2 versions got Failures

```
%sql select substr(Date,0,5) as Month, Booster_Version, Launch_Site from SPACEXTABLE where substr(Date,0,5)='2015' and Landing_Outcome='Failure (drone ship)'
```

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

Month	Booster_Version	Launch_Site
2015	F9 v1.1 B1012	CCAFS LC-40
2015	F9 v1.1 B1015	CCAFS LC-40

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
- 30% is not attempt for landing

```
%sql select *, count(*) from SPACEXTABLE where Date between '2010-06-04' and '2017-03-20' group by Landing_Outcome order by count(*)
```

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

Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Orbit	Customer	Mission_Outcome	Landing_Outcome	count(*)
2015-06-28	14:21:00	F9 v1.1 B1018	CCAFS LC-40	SpaceX CRS-7	1952	LEO (ISS)	NASA (CRS)	Failure (in flight)	Precluded (drone ship)	1
2010-06-04	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success	Failure (parachute)	2
2013-09-29	16:00:00	F9 v1.1 B1003	VAFB SLC-4E	CASSIOPE	500	Polar LEO	MDA	Success	Uncontrolled (ocean)	2
2014-04-18	19:25:00	F9 v1.1	CCAFS LC-40	SpaceX CRS-3	2296	LEO (ISS)	NASA (CRS)	Success	Controlled (ocean)	3
2015-12-22	1:29:00	F9 FT B1019	CCAFS LC-40	OG2 Mission 2 11 Orbcomm-OG2 satellites	2034	LEO	Orbcomm	Success	Success (ground pad)	3
2015-01-10	9:47:00	F9 v1.1 B1012	CCAFS LC-40	SpaceX CRS-5	2395	LEO (ISS)	NASA (CRS)	Success	Failure (drone ship)	5
2016-04-08	20:43:00	F9 FT B1021.1	CCAFS LC-40	SpaceX CRS-8	3136	LEO (ISS)	NASA (CRS)	Success	Success (drone ship)	5
2012-05-22	7:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success	No attempt	10


Section 3
Launch Sites
Proximities Analysis

Positions of launch sites

- Launch sites in proximity to the Equator line
- Launch sites in very close proximity to the coast

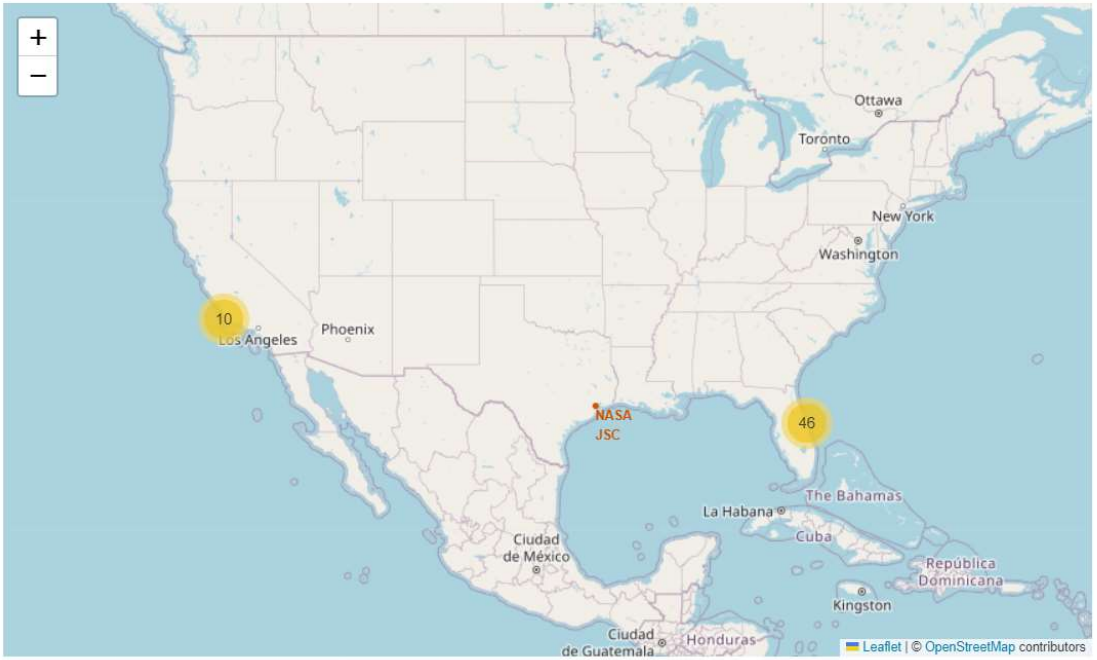
```
# Initial the map
site_map = folium.Map(location=nasa_coordinate, zoom_start=5)
# For each launch site, add a Circle object based on its coordinate (Lat, Long) values. In addition, add Launch site name
for row in launch_sites_df.itertuples():
    marker = folium.map.Marker([row.Lat,row.Long], icon=DivIcon(icon_size=(20,20),icon_anchor=(0,0),
        html='<div style="font-size: 12; color:#d35400;"><b>%s</b></div>' % row._1, ))
    site_map.add_child(marker)
site_map
```

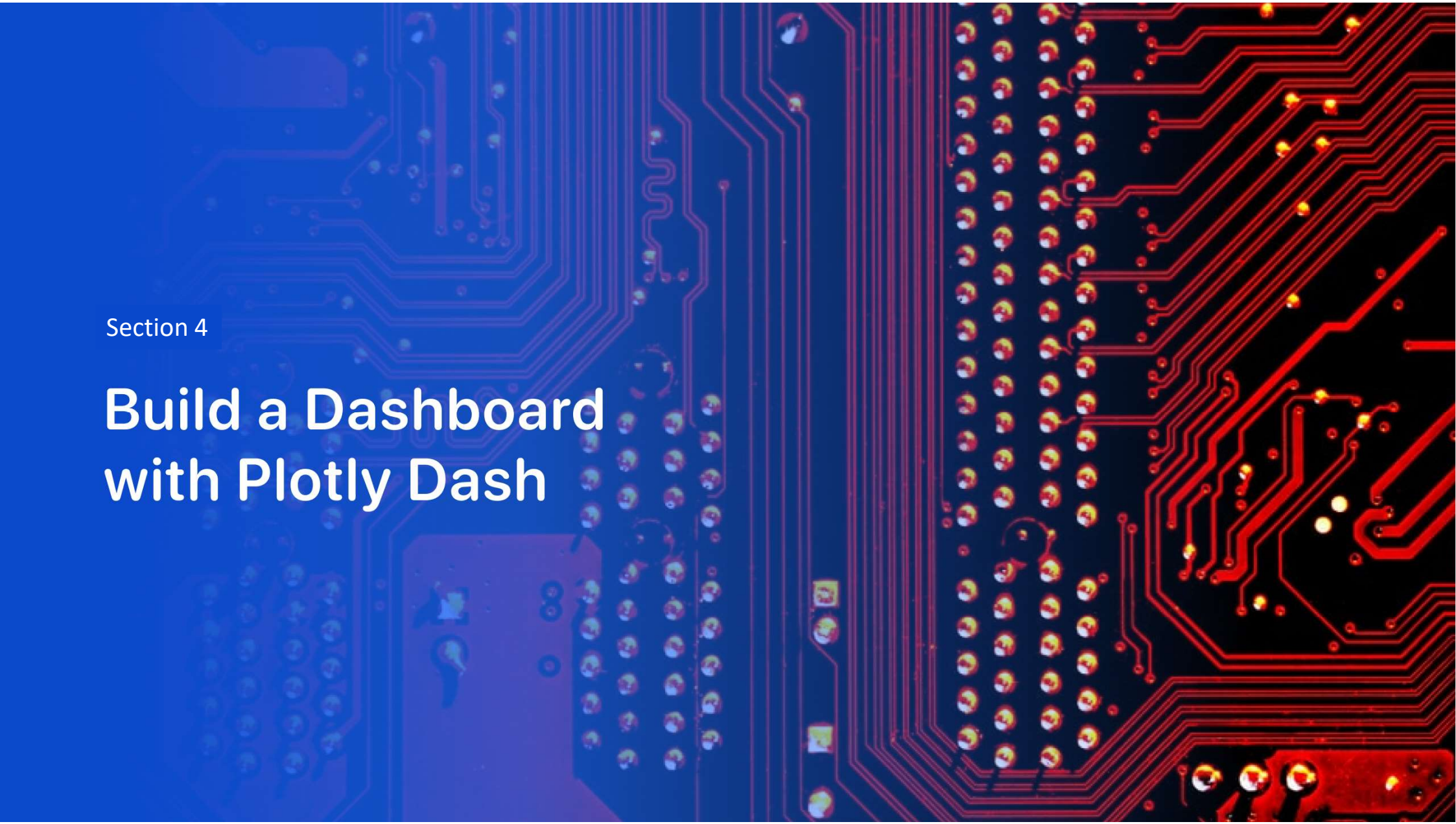
The generated map with marked launch sites should look similar to the following:

Launch records

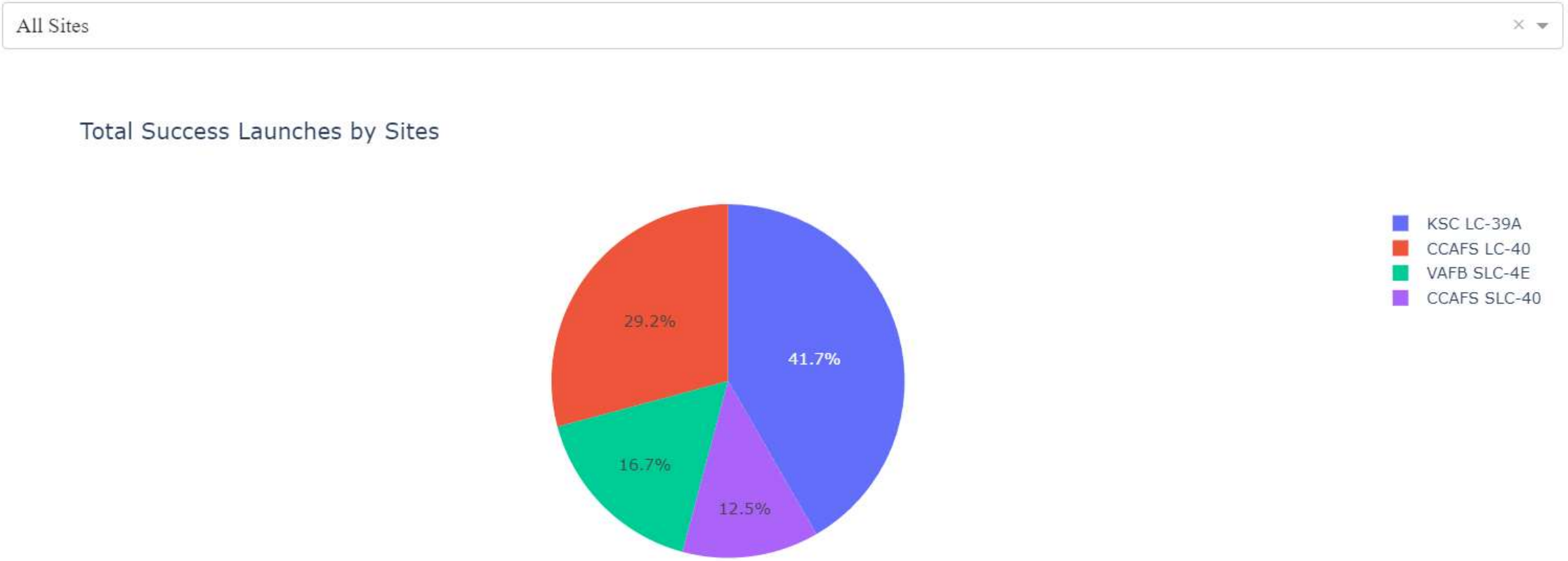
- More launches on East side compared to West side

```
# Add marker_cluster to current site_map
site_map.add_child(marker_cluster)

# for each row in spacex_df data frame
# create a Marker object with its coordinate
# and customize the Marker's icon property to indicate if this launch was successed or failed,
# e.g., icon=folium.Icon(color='white', icon_color=row['marker_color']
for index, record in spacex_df.iterrows():
    # TODO: Create and add a Marker cluster to the site map
    marker = folium.Marker([record.Lat, record.Long],
        popup = str(record.index), icon=folium.Icon(color=record.marker_color))
    marker_cluster.add_child(marker)

site_map
```



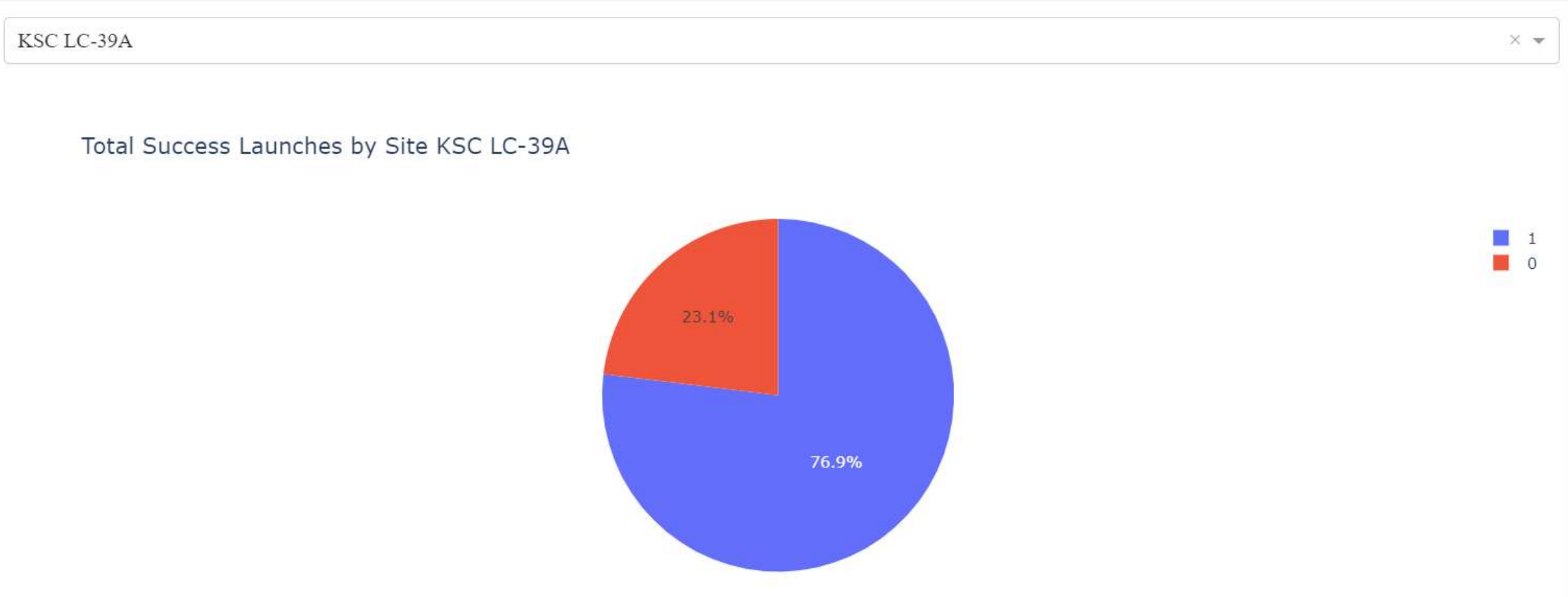

Section 4
Build a Dashboard with Plotly Dash

Total success launches by all sites



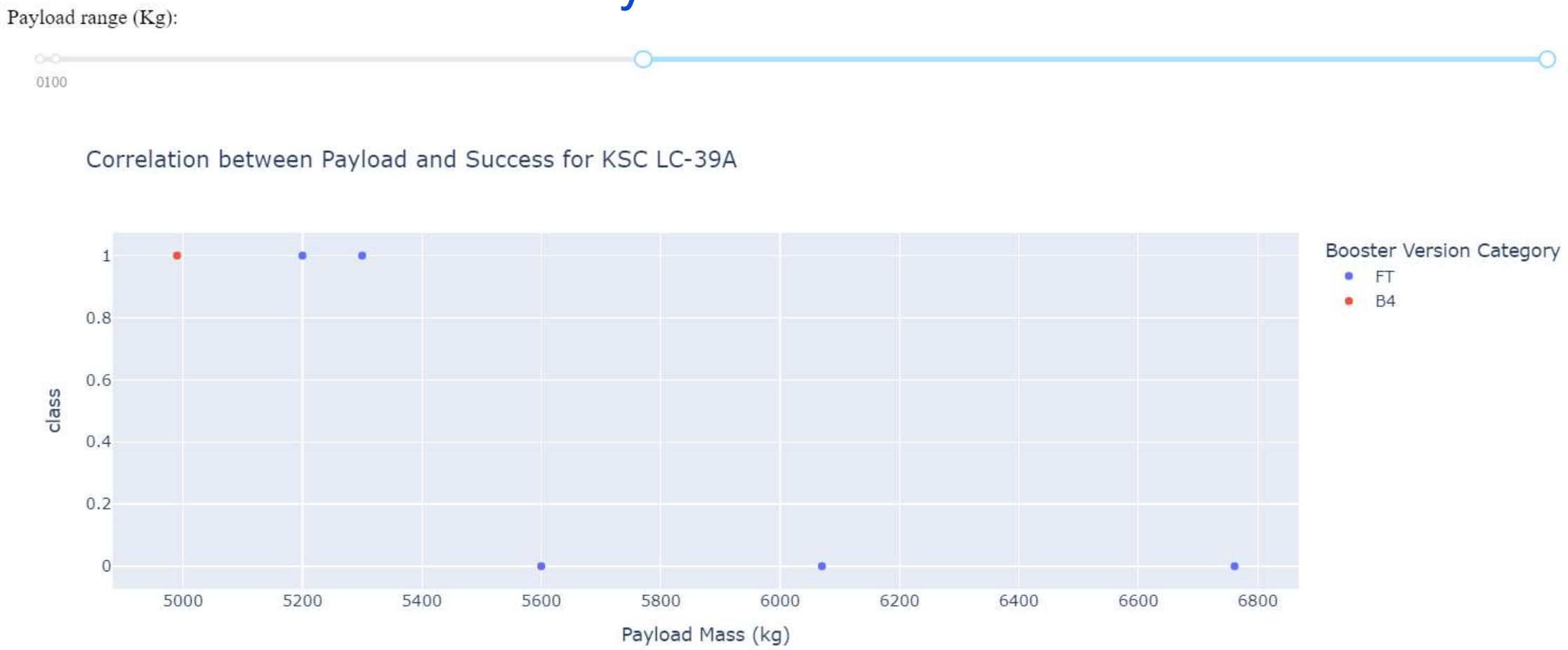
- KSC LC-39A has the most success rate

Total success launches by KSC LC-39A



- The site has 80% of success rate

Correlation between Payload and Success for KSC LC-39A



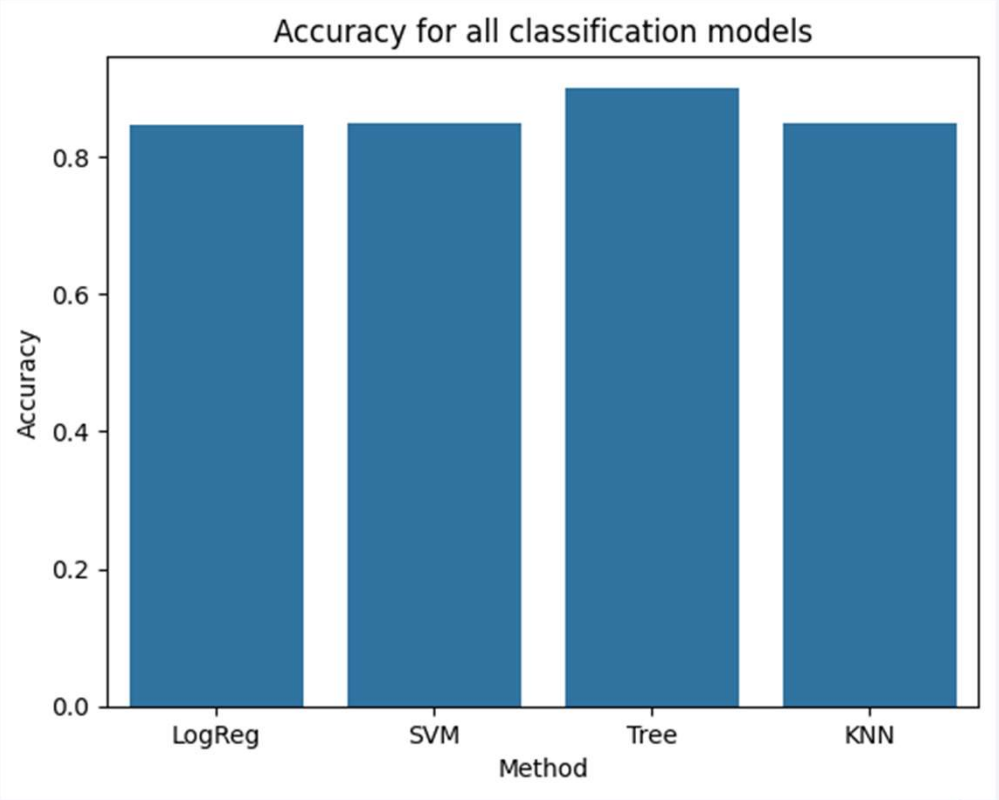
- Large payload causes almost failures

Section 5

Predictive Analysis (Classification)

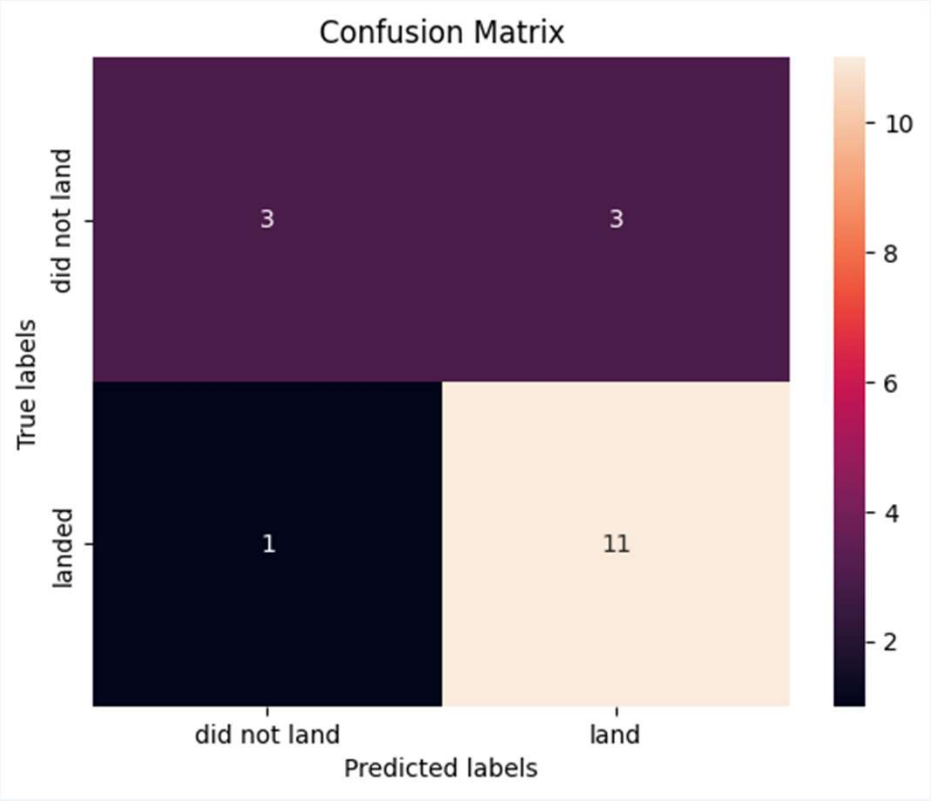
Classification Accuracy

- Decision tree classifier has the highest accuracy result



Confusion Matrix

- Has the most landed on land predicted correctly



Conclusions

- Based on prediction with high confidence (90%), that we can achieve 40% of the first stage land successfully.

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!