IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Yi Hyun Choi
May 23, 2023

Outline

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

Executive Summary

- Summary of methodologies
  - Collecting the Data
  - Data Wrangling
  - Exploratory Analysis using SQL
  - Exploratory Data Analysis with Visualization
  - Interactive Visual Analytics and Dashboard
  - Predictive Analysis
- Summary of all results

Introduction

- Project background and context

The commercial space companies are making space travel cheaper than before. spaceX advertises Falcom 9 rocket launches with $ 62M than other's $165M, because of reusing the first stage rocket. The first stage is the most expensive and large in the rocket launch.

- Problems you want to find answers

Determine the price of each launch from SpaceX launch data and predict SpaceX will reuse the first stage.

Section 1

Methodology

Methodology

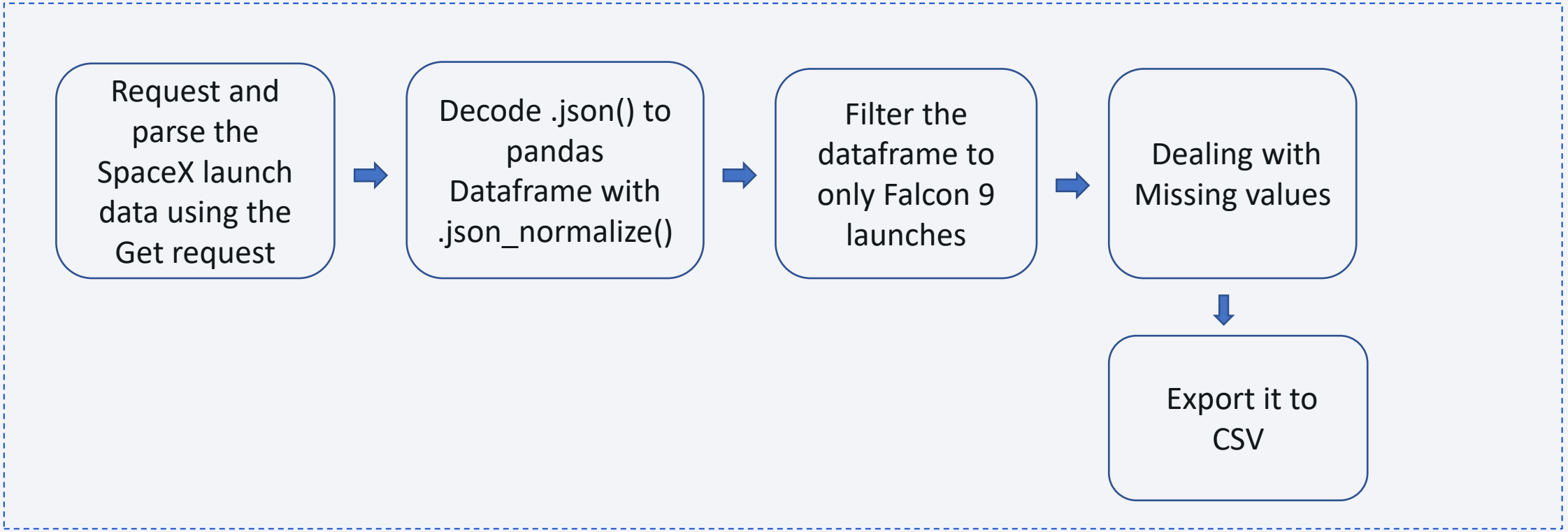
Executive Summary

- Data collection methodology:
  - SpaceX launch data from the SpaceX REST API
- Perform data wrangling
  - Using an API, Sampling Data, and Dealing with Nulls
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - Perform exploratory Data Analysis and determine Training Labels
  - Find best Hyperparameter for SVM, classification Trees and Logistic Regression

Data Collection

- Describe how data sets were collected.
  - Collecting SpaceX launch data from SPACEX REST API
  - URL: api.spacexdata.com/v4/
  - Convert json objects to a dataframe with json_normalize function

Data Collection – SpaceX API



- Add the GitHub URL of the completed SpaceX API calls notebook (must include completed code cell and outcome cell), as an external reference and peer-review purpose

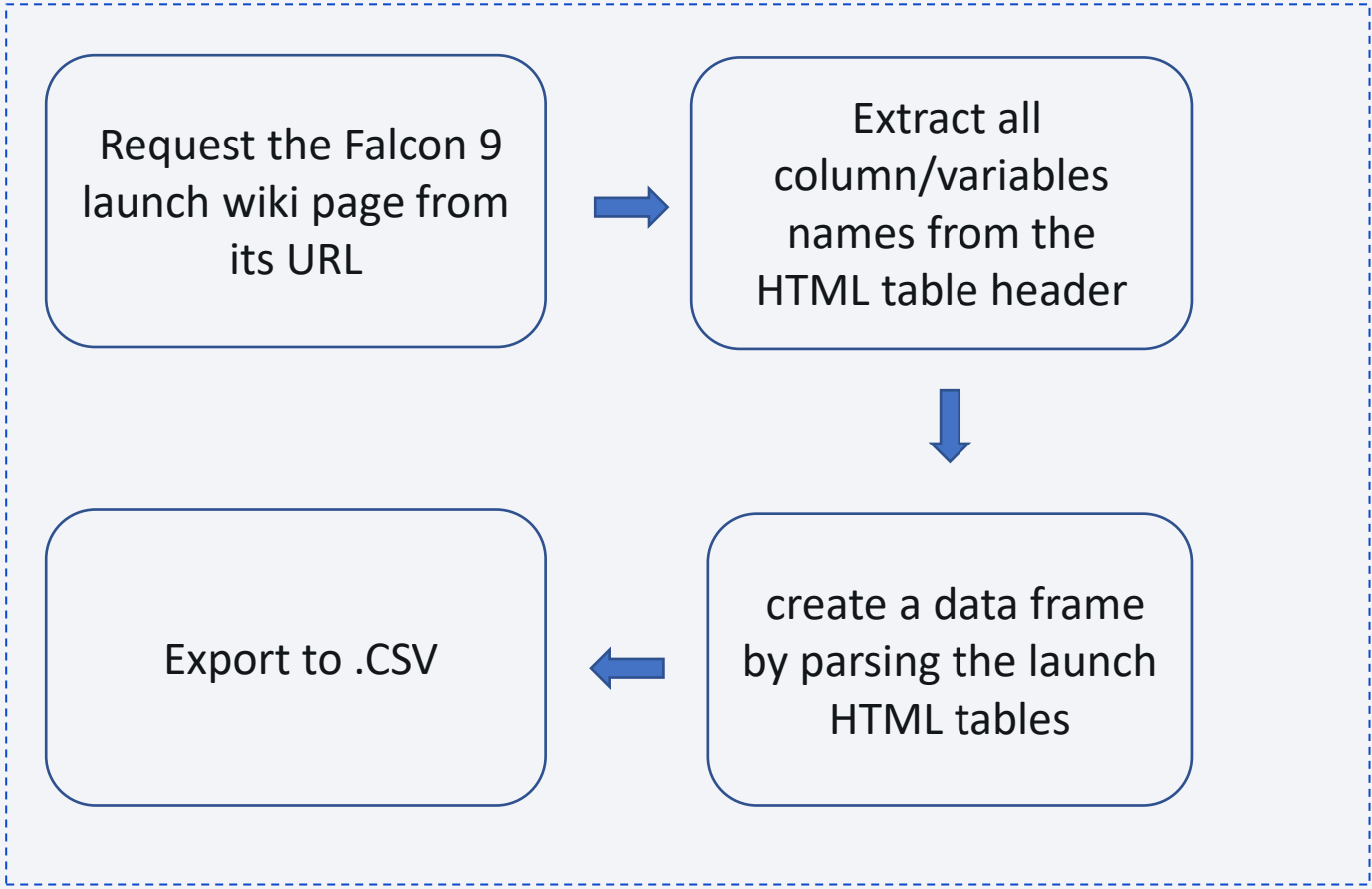
https://github.com/nageune1004/Capstone_Project/blob/main/jupyter-labs-spacex-data-collection-api.ipynb

Data Collection - Scraping

- Present your web scraping process using key phrases and flowcharts

- Add the GitHub URL of the completed web scraping notebook, as an external reference and peer-review purpose

https://github.com/nageune1004/Capstone_Project/blob/main/jupyter-labs-webscraping.ipynb



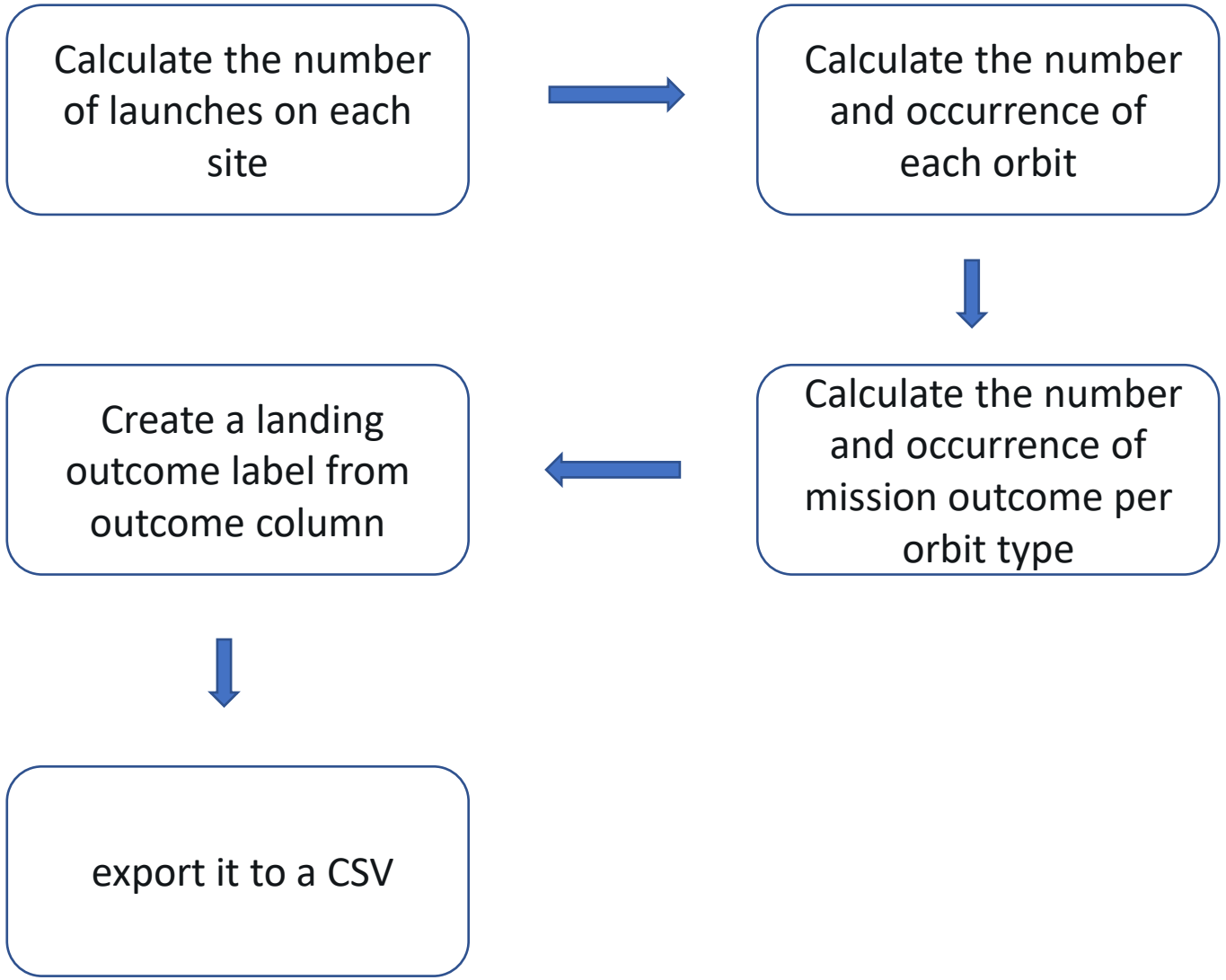
Data Wrangling

- Describe how data were processed
  - Outcome to be converted to classes Y (either 0 or 1). 0 is a bad outcome, 1 is good. The variable Y will represent the classification vaable that represent s the outcome of each launch

Data Wrangling

- You need to present your data wrangling process using key phrases and flowcharts
  - Calculate the number of launches
    - df['LaunchSite'].value_counts()
  - Calculate the number of Orbit
    - df['Orbit'].value_counts()
  - Calculate the number of Mission outcome
    - df['Outcome'].value_counts()
  - Create a landing outcome label
- Add the GitHub URL of your completed Data Wrangling notebook, as an external reference and peer-review purpose
  - https://github.com/nageune1004/Capstone_Project/blob/main/IBM-DS0321EN-SkillsNetwork_labs_module_1_L3_labs-jupyter-spacex-data_wrangling_jupyterlite.jupyterlite.ipynb

Data Wrangling



EDA with Data Visualization

- Summarize what charts were plotted and why you used those charts
  - To determine what attributes are correlated with successful landings, it will predict if the fist stage will successfully land.
  - Visualize the relationship between Flight Number and Payload Mass, Flight Number and Launch Site, Payload Mass and Launch Site, Success rate and each Orbit, Flight Number and Orbits, Payload Mass and Orbits, and yearly launch success variables how would affect the launch outcome.
- Add the GitHub URL of your completed EDA with data visualization notebook, as an external reference and peer-review purpose

  https://github.com/nageune1004/Capstone_Project/blob/main/IBM-DS0321EN-SkillsNetwork_labs_module_2_jupyter-labs-eda-dataviz.ipynb.jupyterlite.ipynb

EDA with SQL

- Using bullet point format, summarize the SQL queries you performed
  - Display the names of the unique launch sites
  - 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 versin F9 v1.1
  - List the date when the first successful landing outcome in 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 successful landing_outcomes

EDA with SQL

- Add the GitHub URL of your completed EDA with SQL notebook, as an external reference and peer-review purpose
  - https://github.com/nageune1004/Capstone_Project/blob/main/jupyter-labs-eda-sql-coursera_sqllite.ipynb

Build an Interactive Map with Folium

- Summarize what map objects such as markers, circles, lines, etc. you created and added to a folium map
  - Folium.marker to show the launch sites on the map, and success/failed launches for each site on the map with marker cluster. Folium.polyline to show distance between launch site and closet interesting point(such as railroad station).
- Explain why you added those objects
  - The launch success rate depend on many factors. It may depend on the location and proximities of a launch site. Finding an optimal location for a launch site.
- Add the GitHub URL of your completed interactive map with Folium map, as an external reference and peer-review purpose
  - https://github.com/nageune1004/Capstone_Project/blob/main/IBM-DS0321EN-SkillsNetwork_labs_module_3_lab_jupyter_launch_site_location.jupyterlite.ipynb

Build a Dashboard with Plotly Dash

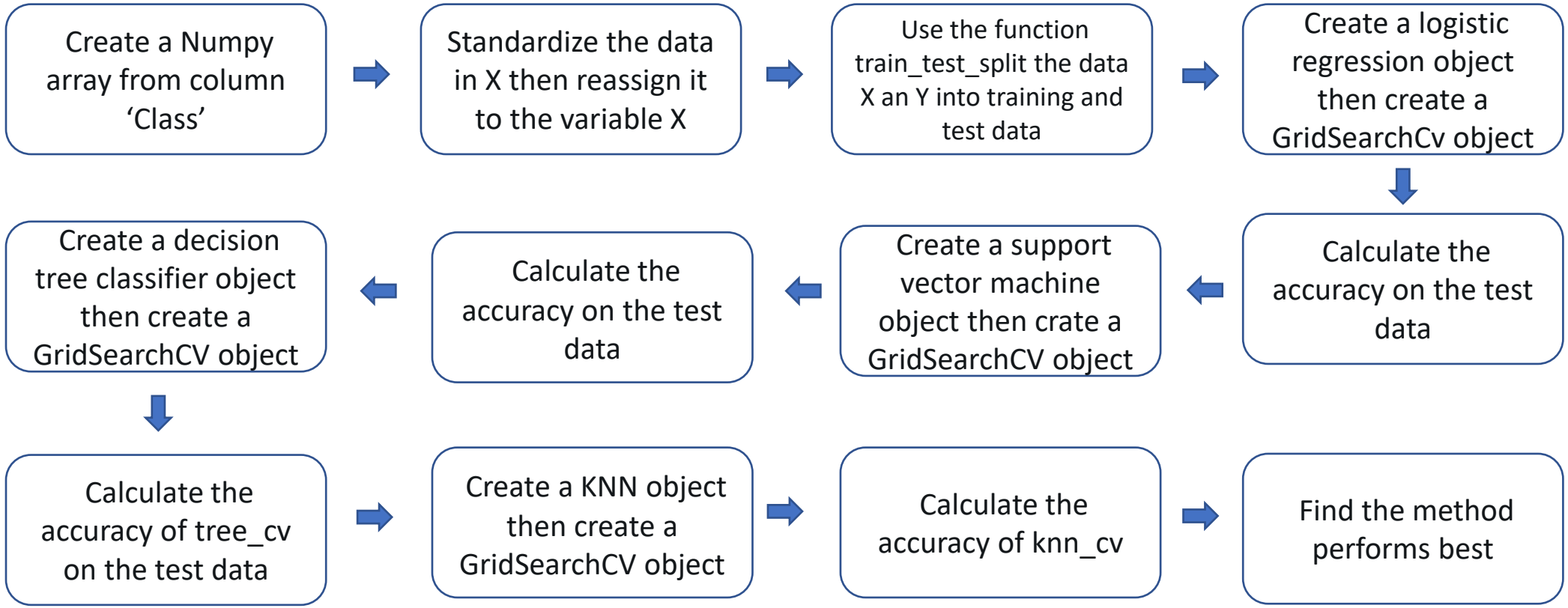
- Summarize what plots/graphs and interactions you have added to a dashboard
  - Add launch sites in drop-down menu with link a success-pie-char based on each launch site. Add a range slider to select payload and scatter plot base on success/failed payload mass chart.
- Explain why you added those plots and interactions
  - The Dashboard will present visual analysis in following questions:
    - Which site has the largest successful launches
    - Which site ahs the highest launch success rate
    - Which payload range has the highest launch success rate
    - Which payload range has the lowest launch success rate
    - Which F9 Booster version has the highest launch success rate
- Add the GitHub URL of your completed Plotly Dash lab, as an external reference and peer-review purpose
  - https://github.com/nageune1004/Capstone_Project/blob/main/spacex_dash_app.py

Predictive Analysis (Classification)

- Summarize how you built, evaluated, improved, and found the best performing classification model
  - Build a machine learning pipeline to predict if the first stage of the Falcon9 lands successfully. Include Preprocessing, standardize the data, and Train_test_split. Train the model and perform Grid Search to find the hyperparameter that allow a given algorithm to perform best. Using the best hyperparameter values, it will determine the model with the best accuracy using the training data.

Predictive Analysis (Classification)

- You need present your model development process using key phrases and flowchart

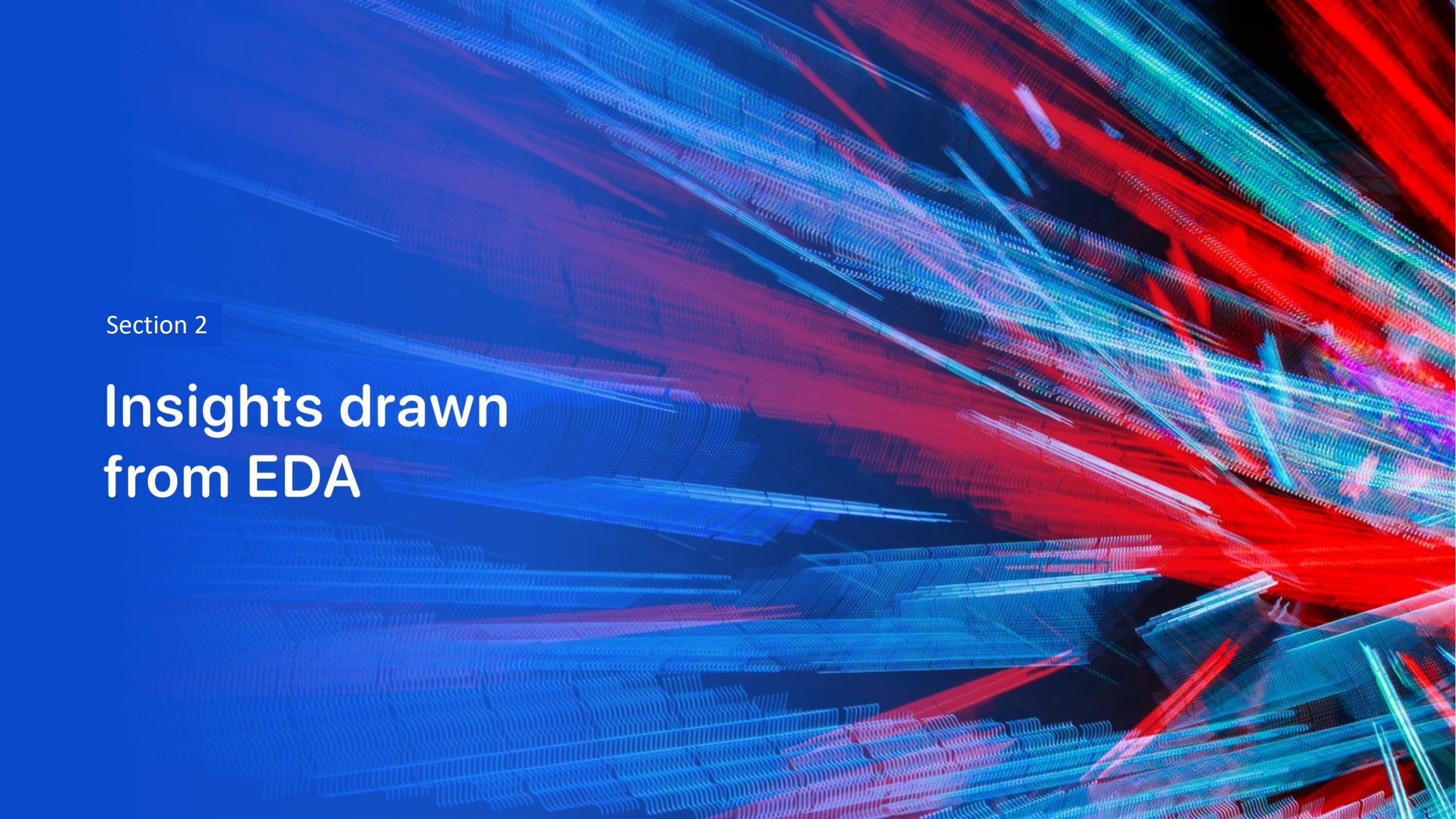


Predictive Analysis (Classification)

- Add the GitHub URL of your completed predictive analysis lab, as an external reference and peer-review purpose
  - https://github.com/nageune1004/Capstone_Project/blob/main/IBM-DS0321EN-SkillsNetwork_labs_module_4_SpaceX_Machine_Learning_Prediction_Part_5.jupyterlite.ipynb

Results

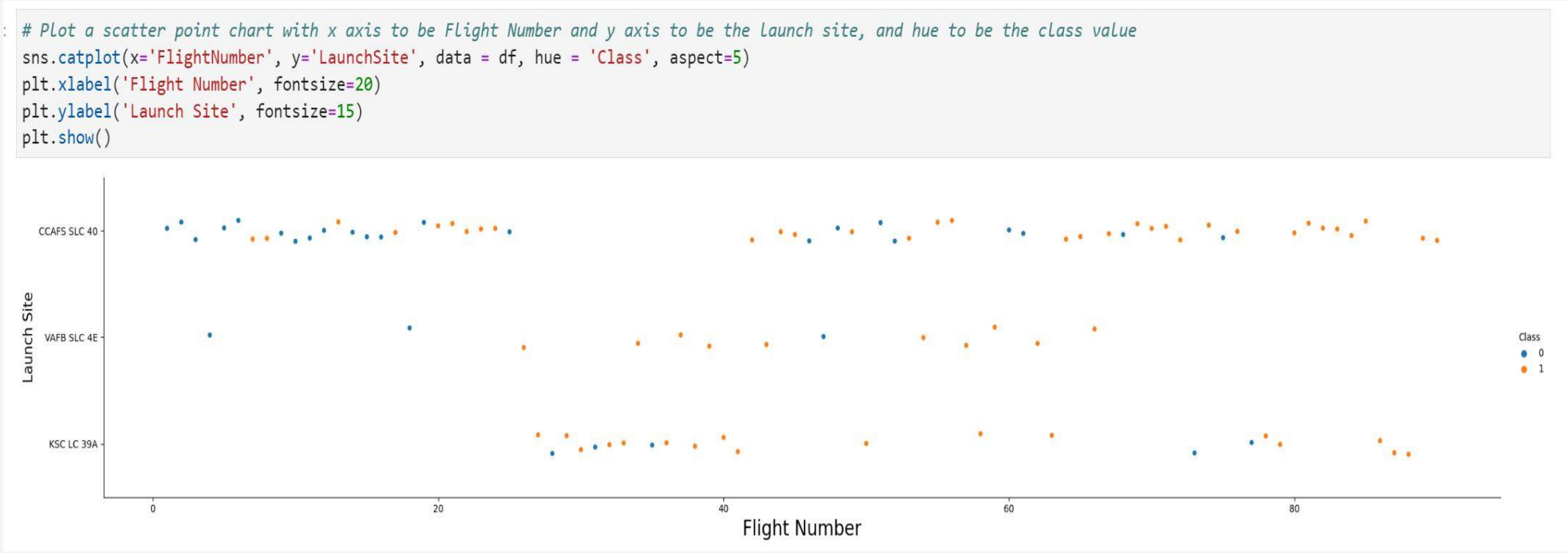
- Exploratory data analysis results
- Interactive analytics demo in screenshots
- Predictive analysis results


Section 2
Insights drawn from EDA

Flight Number vs. Launch Site

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

```
# Plot a scatter point chart with x axis to be Flight Number and y axis to be the launch site, and hue to be the class value
sns.catplot(x='FlightNumber', y='LaunchSite', data = df, hue = 'Class', aspect=5)
plt.xlabel('Flight Number', fontsize=20)
plt.ylabel('Launch Site', fontsize=15)
plt.show()
```



More flight test, success rate increased too.

Payload vs. Launch Site

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

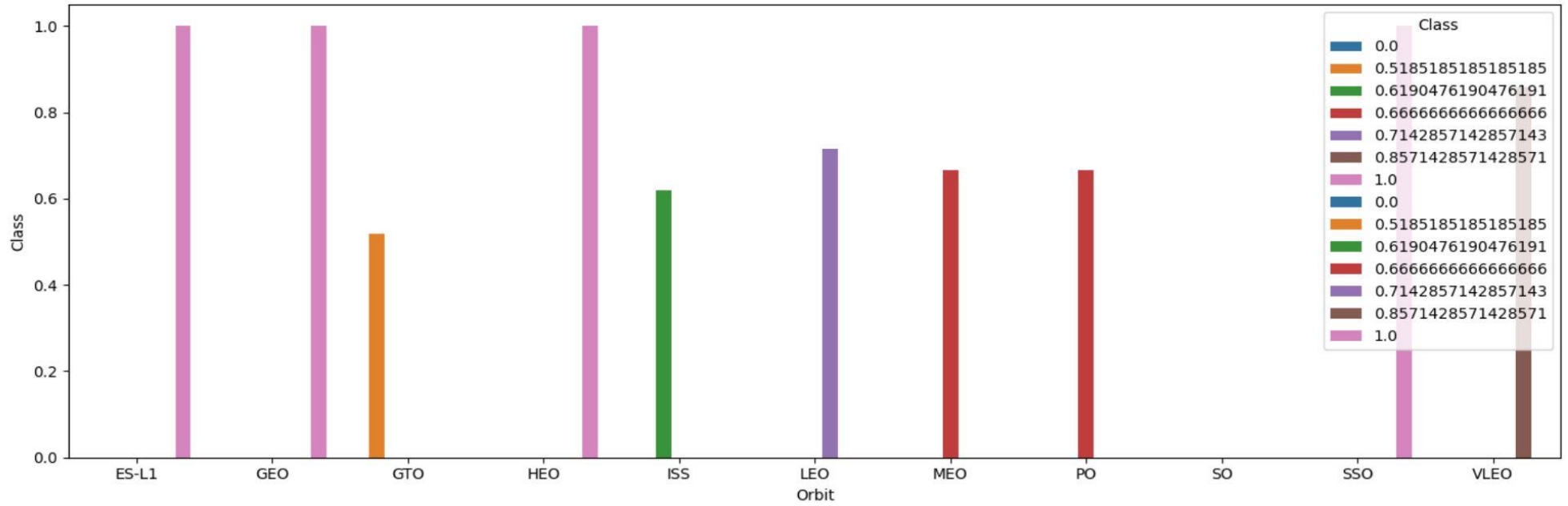
```
# Plot a scatter point chart with x axis to be Pay Load Mass (kg) and y axis to be the launch site, and hue to be the class value
sns.catplot(y='LaunchSite', x='PayloadMass', data = df, hue='Class', aspect=3)
plt.xlabel('Pay Load Mass(kg)', fontsize=15)
plt.ylabel('Launch Site')
plt.show()
```

Between 8000 and 16000 pay load mass has better success rate

Success Rate vs. Orbit Type

- Show a bar chart for the success rate of each orbit type

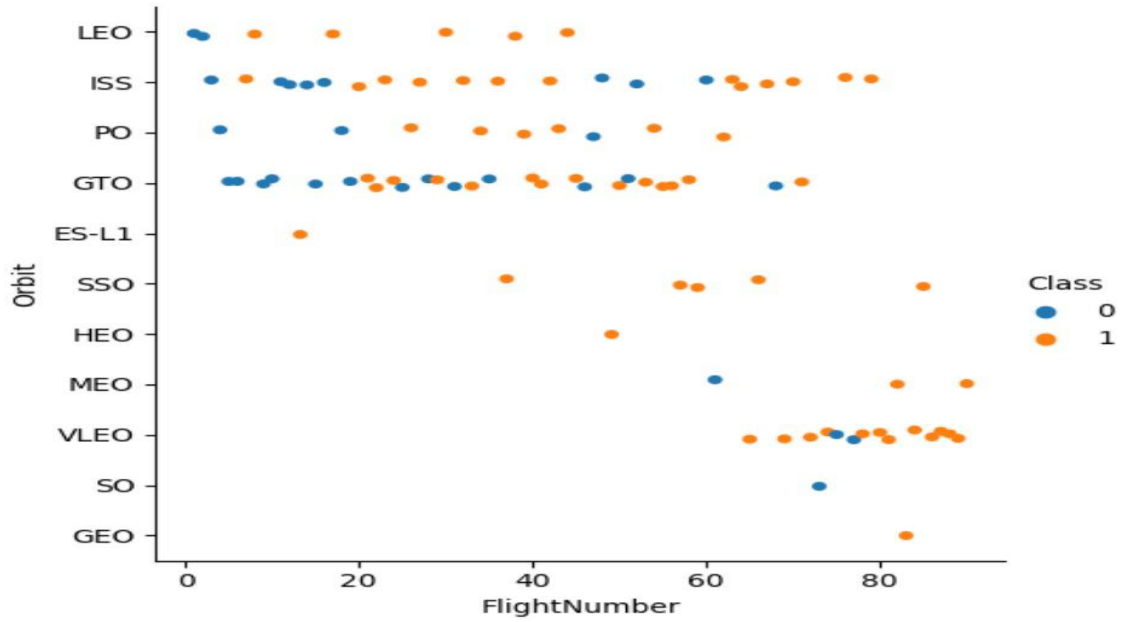
```
sns.barplot(y='Class', x='Orbit', data=orbit_mean, hue='Class')
plt.show()
```



Flight Number vs. Orbit Type

- Show a scatter point of Flight number vs. Orbit type
- Show the screenshot of the scatter plot with explanations

```
# Plot a scatter point chart with x axis to be FlightNumber and y axis to be the Orbit, and hue to be the class value
sns.catplot(x='FlightNumber', y='Orbit', data=df, hue='Class')
plt.show()
```

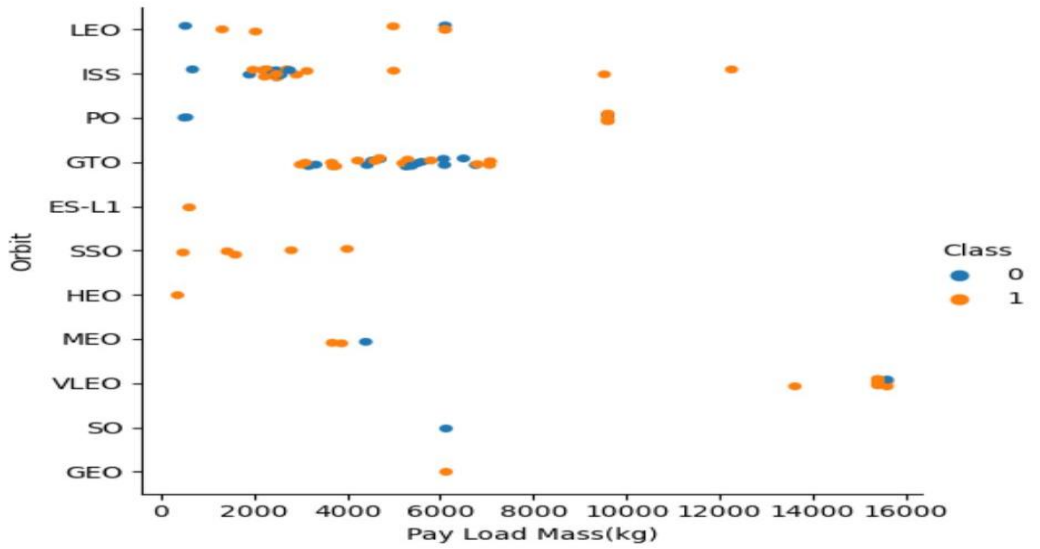


LEO orbit the success rate related to the number of flights, but GTO has no relationship.

Payload vs. Orbit Type

- Show a scatter point of payload vs. orbit type
- Show the screenshot of the scatter plot with explanations

```
# Plot a scatter point chart with x axis to be Payload and y axis to be the Orbit, and hue to be the class value
sns.catplot(x='PayloadMass', y='Orbit', hue = 'Class', data = df)
plt.xlabel('Pay Load Mass(kg)')
plt.show()
```

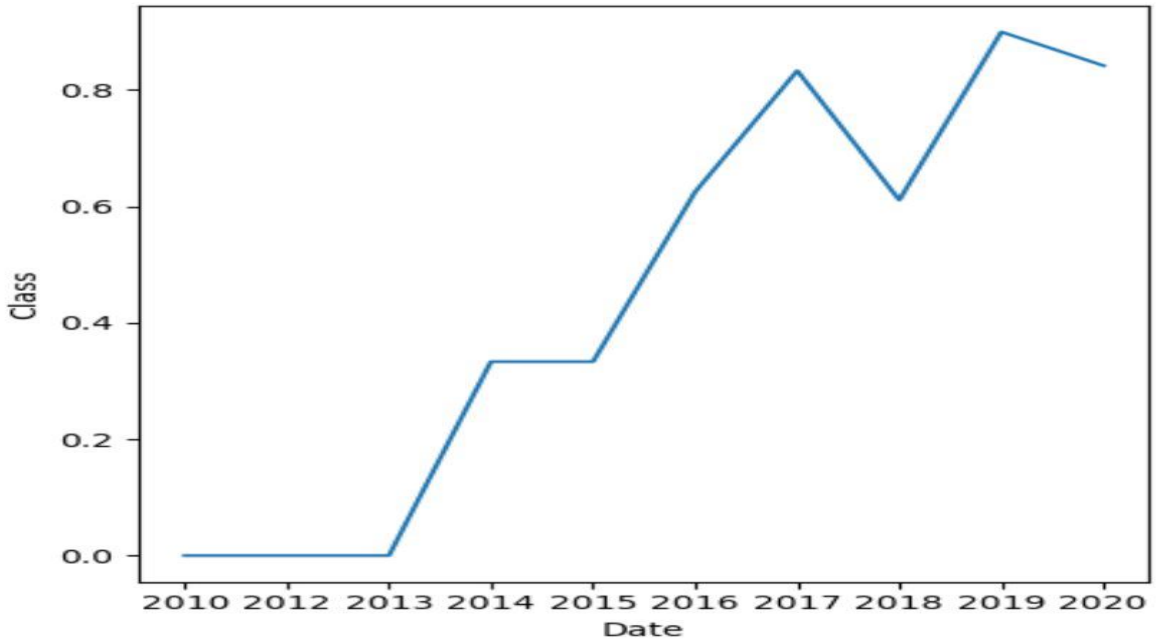


With heavy payloads the successful landing or positive landing rate are more for Polar, LEO and ISS

Launch Success Yearly Trend

- Show a line chart of yearly average success rate

```
# Plot a line chart with x axis to be the extracted year and y axis to be the success rate
yearly = df.groupby('Date').mean()
yearly.reset_index(inplace = True)
sns.lineplot(x='Date', y='Class', data = yearly)
plt.show()
```



The success rate is increased since 2013

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;
 * ibm_db_sa://[redacted]ludb
   sqlite:///my_data1.db
Done.
```

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

Launch Site Names Begin with 'CCA'

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

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

```
%sql select * from spacextbl where (launch_site like 'CCA%') limit 5;
```

```
 * ibm_db_sa://tnl72347:***[redacted]databases.appdomain.cloud:30875/bludb
   sqlite:///my_data1.db
Done.
```

DATE	time__utc_	booster_version	launch_site	payload	payload_mass__kg_	orbit	customer	mission_outcome	landing__outcome
2010-04-06	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success	Failure (parachute)
2010-08-12	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)
2012-08-10	00:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500	LEO (ISS)	NASA (CRS)	Success	No attempt
2013-01-03	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677	LEO (ISS)	NASA (CRS)	Success	No attempt
2013-03-12	22:41:00	F9 v1.1	CCAFS LC-40	SES-8	3170	GTO	SES	Success	No attempt

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_) as total_payload_mass from spacextbl where customer = 'NASA (CRS)' ;
```

```
 * ibm_db_sa://tnl72347:***[redacted]databases.appdomain.cloud:30875/bludb
   sqlite:///my_data1.db
Done.
```

total_payload_mass
22007

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_) as average_payload_mass from spacextbl where booster_version = 'F9 v1.1';
```

```
 * ibm_db_sa://tnl72347:***[redacted]databases.appdomain.cloud:30875/bludb
   sqlite:///my_data1.db
Done.
```

average_payload_mass
3676

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) from spacextbl where landing__outcome = 'Success (ground pad)';
```

```
 * ibm_db_sa://tnl72347:***[redacted]databases.appdomain.cloud:30875/bludb
   sqlite:///my_data1.db
Done.
```

1
2017-01-05

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 (landing__outcome = 'Success (drone ship)') and (4000 < payload_mass__kg_ < 6000);
```

```
 * ibm_db_sa://tnl72347:***[redacted]tabases.appdomain.cloud:30875/bludb
   sqlite:///my_data1.db
Done.
```

booster_version
F9 FT B1021.1
F9 B5 B1046.1

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(*) as Total_number from spacextbl group by mission_outcome;
```

```
 * ibm_db_sa://tnl72347:***@[redacted]databases.appdomain.cloud:30875/bludb
   sqlite:///my_data1.db
Done.
```

mission_outcome	total_number
Success	44
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 from spacextbl where payload_mass__kg_ = (select max(payload_mass__kg_) from spacextbl);
```

```
 * ibm_db_sa://tnl72347:***[redacted]databases.appdomain.cloud:30875/bludb
   sqlite:///my_data1.db
Done.
```

booster_version
F9 B5 B1048.4
F9 B5 B1049.4
F9 B5 B1049.5
F9 B5 B1060.2
F9 B5 B1058.3

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 substr(date, 6,2) as Month, landing__outcome, booster_version, launch_site from spacextbl where landing__outcome = 'Failure (drone ship)' AND substr(Da
```

```
 * ibm_db_sa://tnl72347:***[redacted]databases.appdomain.cloud:30875/bludb
   sqlite:///my_data1.db
Done.
```

MONTH	landing__outcome	booster_version	launch_site
10	Failure (drone ship)	F9 v1.1 B1012	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
- Present your query result with a short explanation here

```
select count(*)AS total_success_landing, landing__outcome from spacextbl where (Date between '2010-06-04' and '2017-03-20') and landing__outcome like 'Success%' gro
```

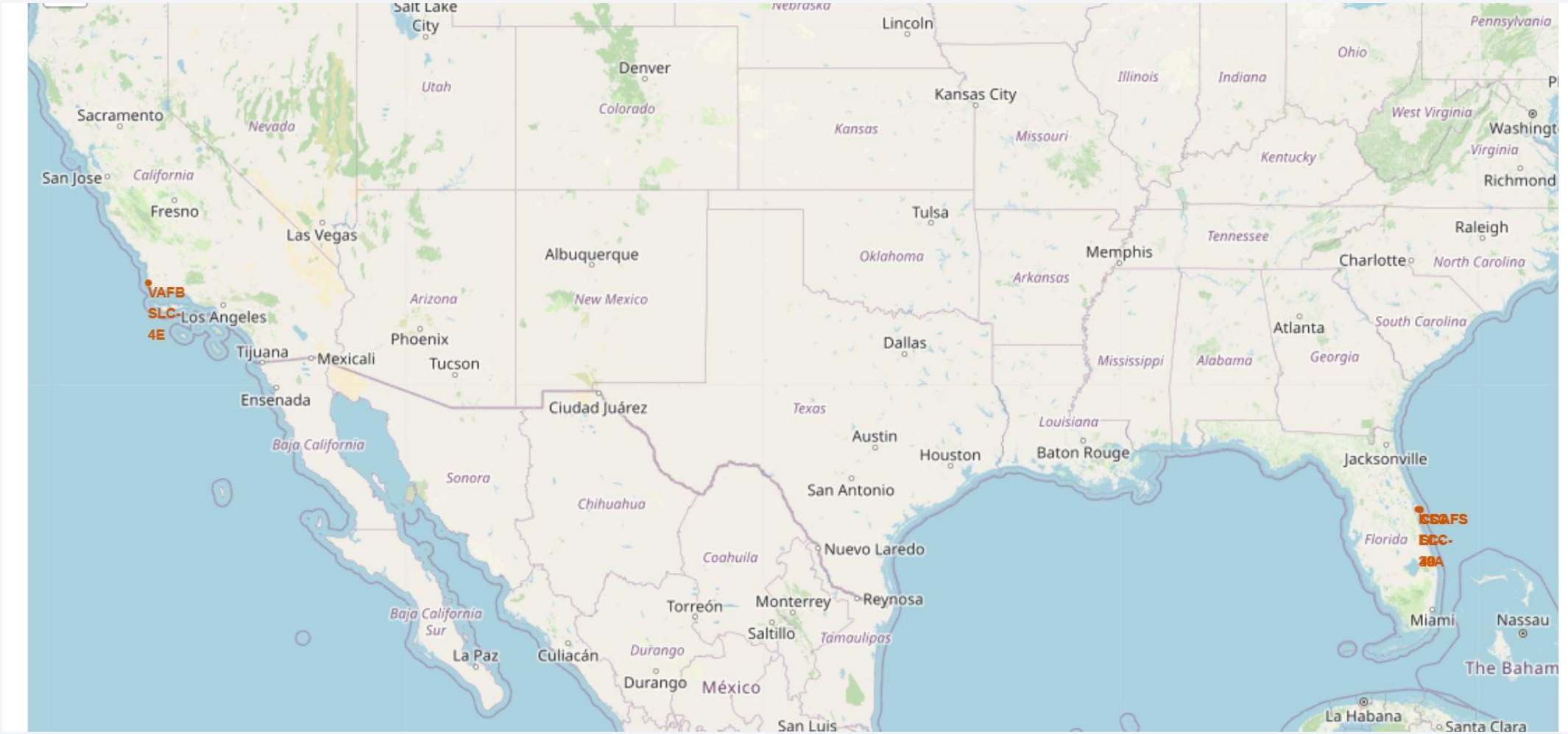
```
 * ibm_db_sa://tnl72347:***@[redacted]databases.appdomain.cloud:30875/bludb
   sqlite:///my_data1.db
Done.
```

total_success_landing	landing__outcome
2	Success (drone ship)
2	Success (ground pad)

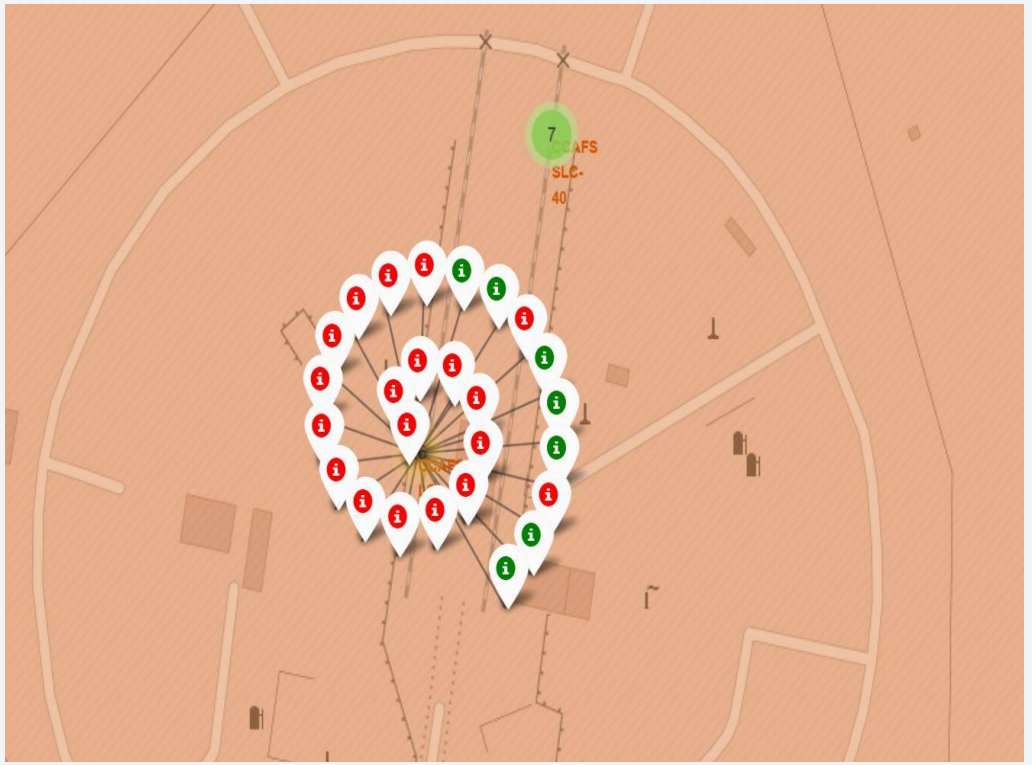
Section 3

Launch Sites Proximities Analysis

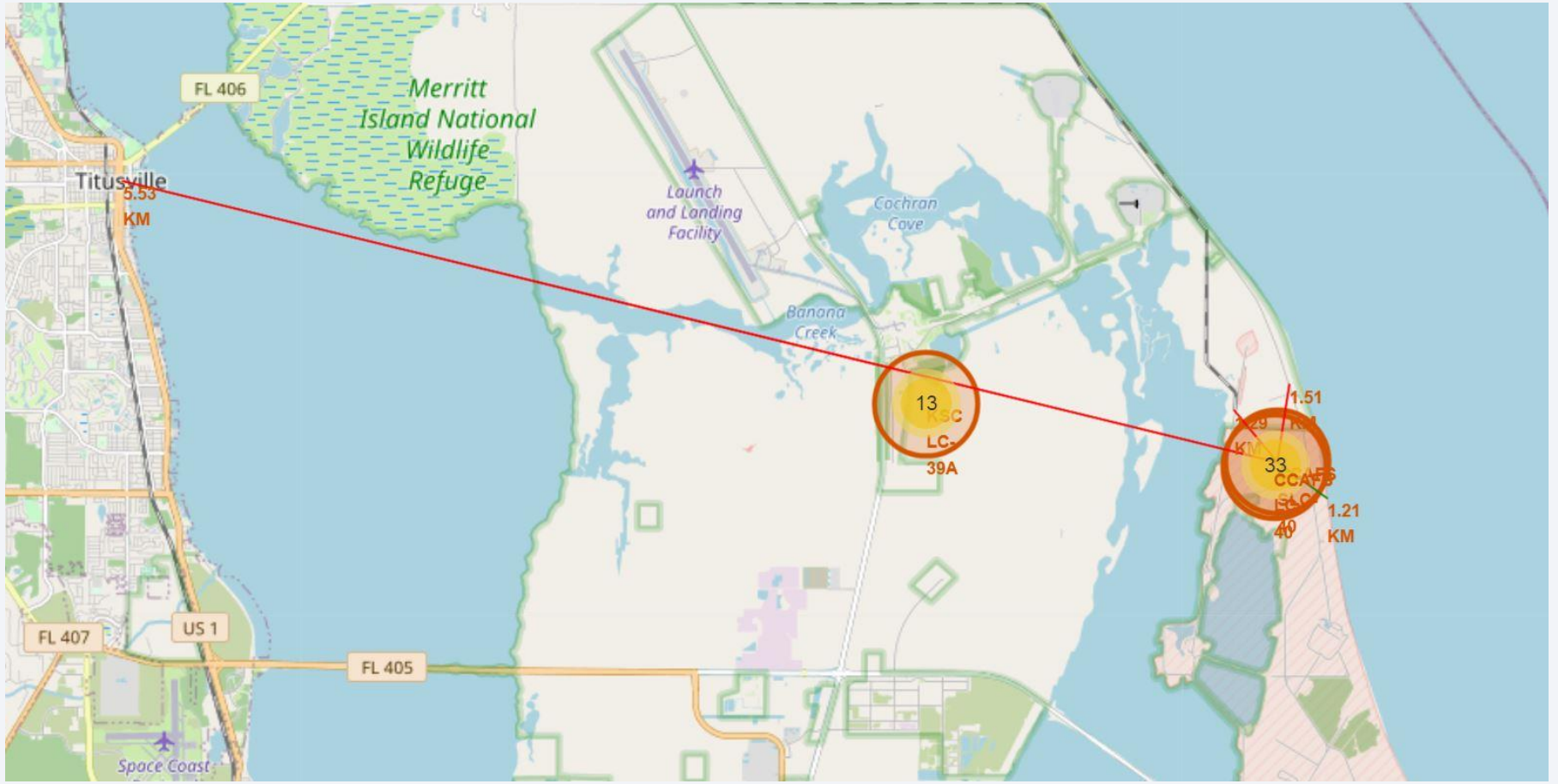
Launch sites on a Map

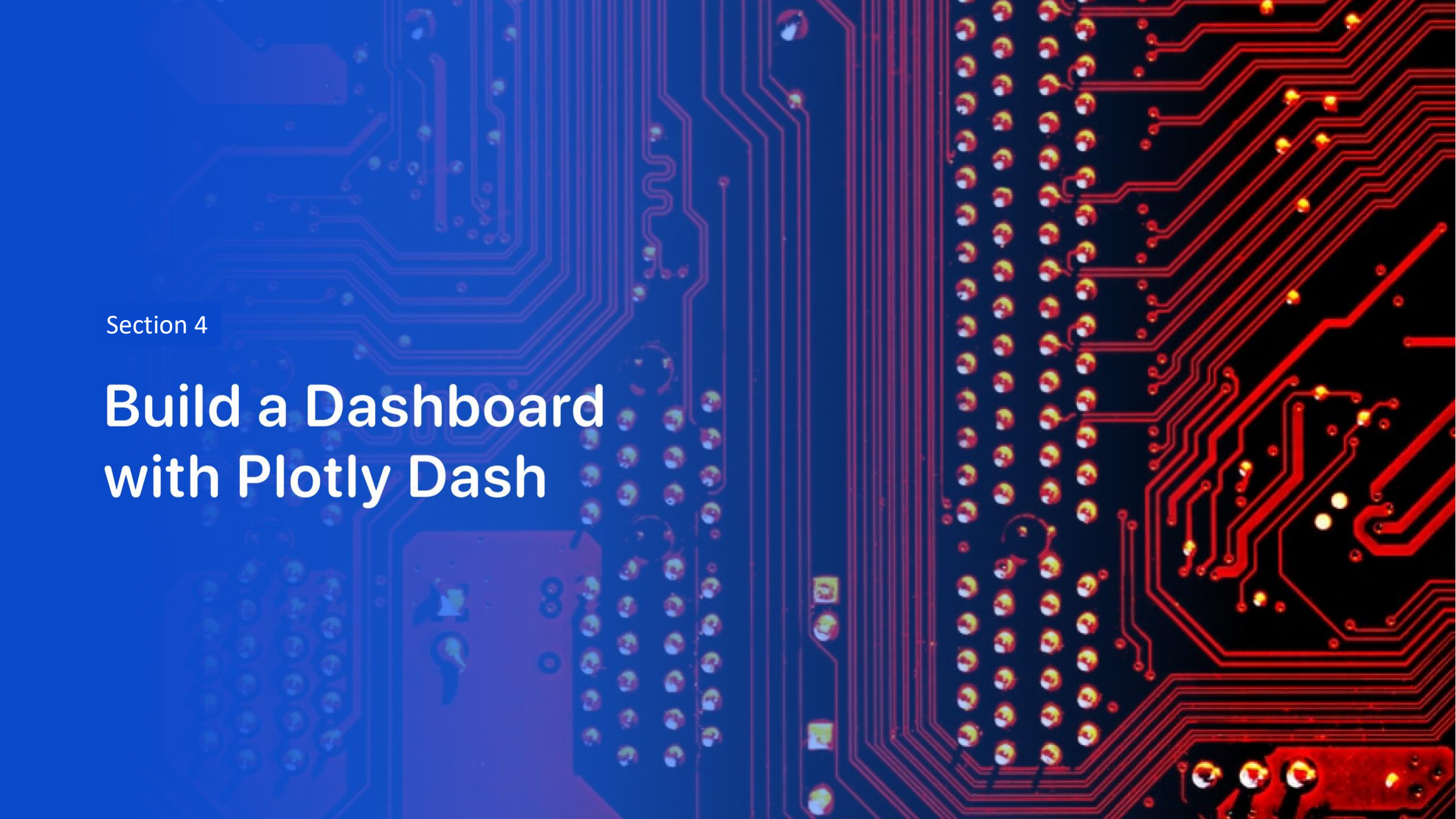


The Success/Failed launches on a launch site



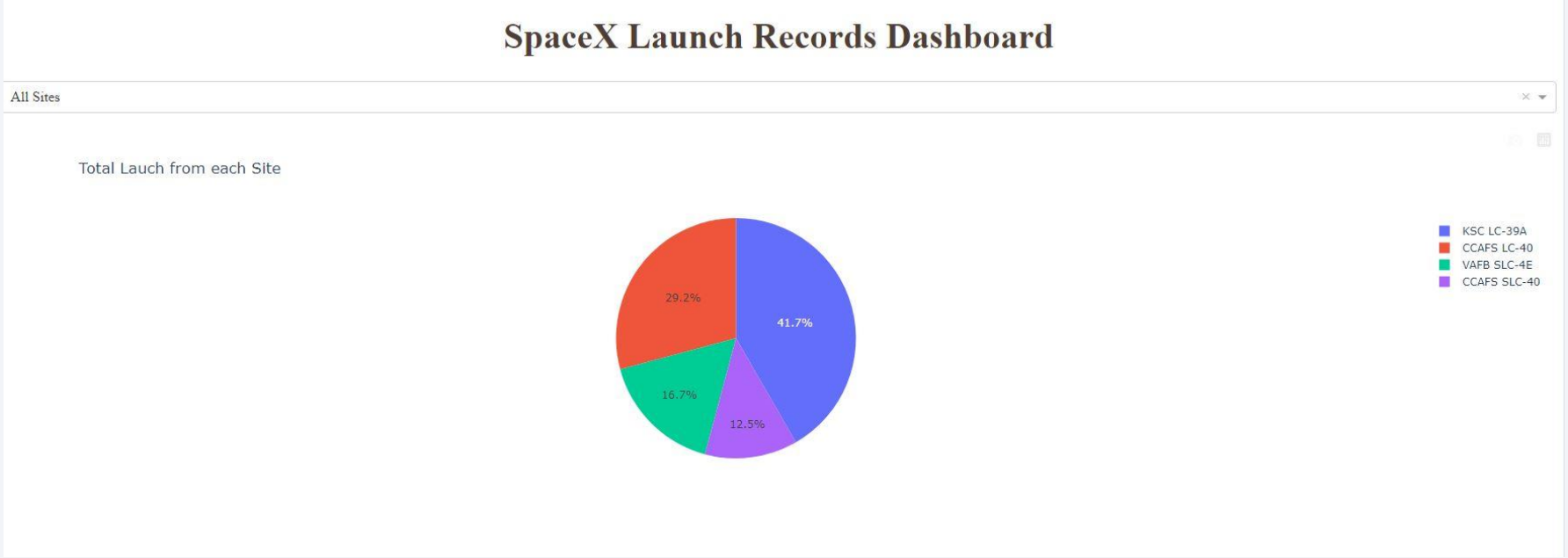
Distance between a launch site and proximities




Section 4
Build a Dashboard with Plotly Dash

Launch success counts for sites

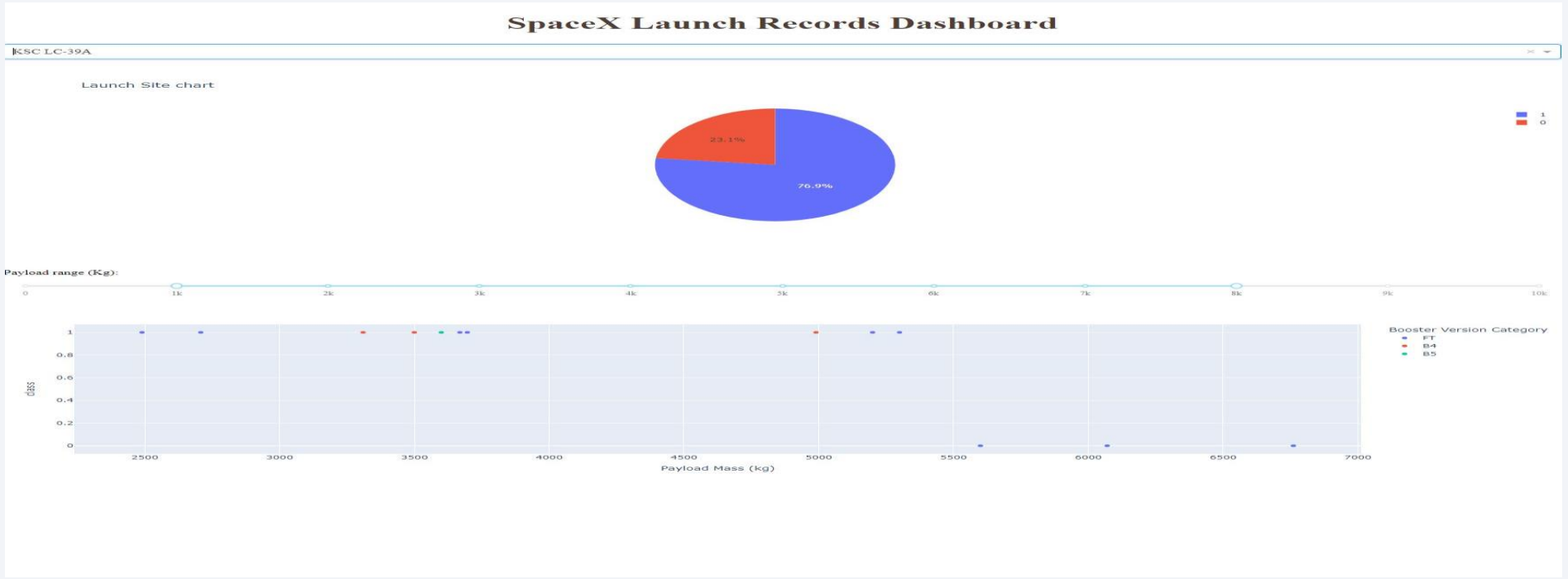
- Show the screenshot of launch success count for all sites, in a piechart



The Lauch site KSC LC-39A is the most launch rate (41.7%)

Highest Launch Success Ratio

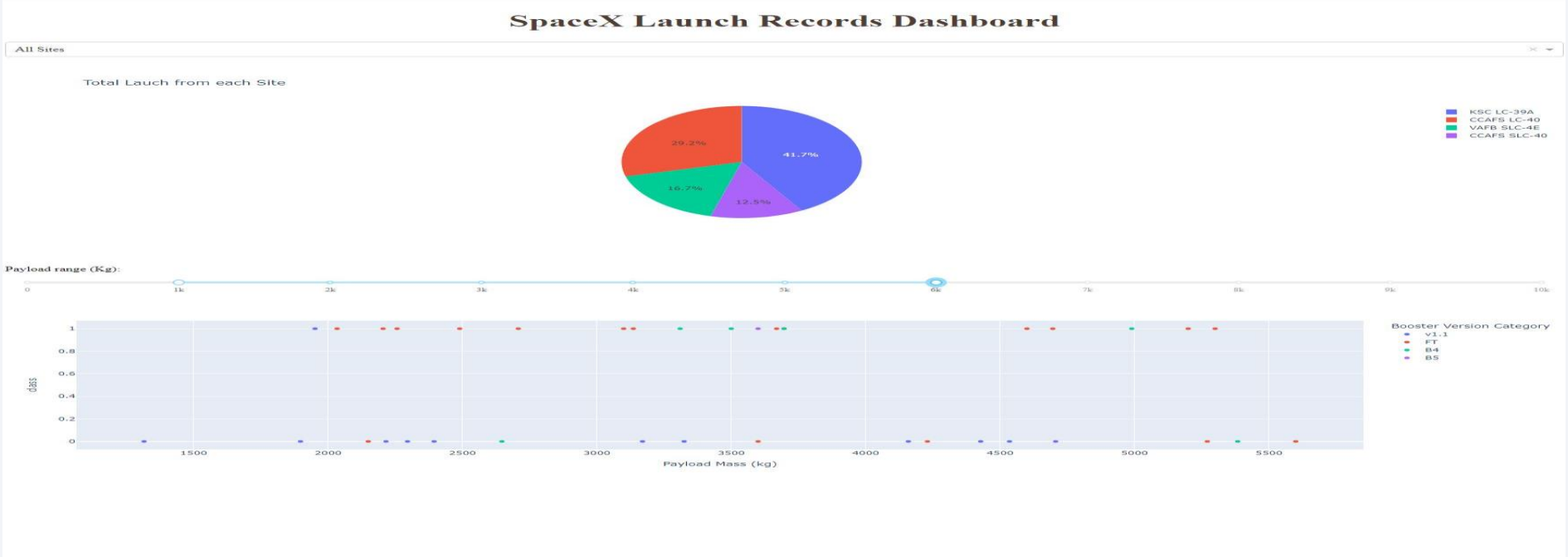
- Show the screenshot of the piechart for the launch site with highest launch success ratio



KSC LC-39A is the most successful launch site(76.9%)

Payload vs. Launch Outcome

- Show screenshots of Payload vs. Launch Outcome scatter plot for all sites, with different payload selected in the range slider



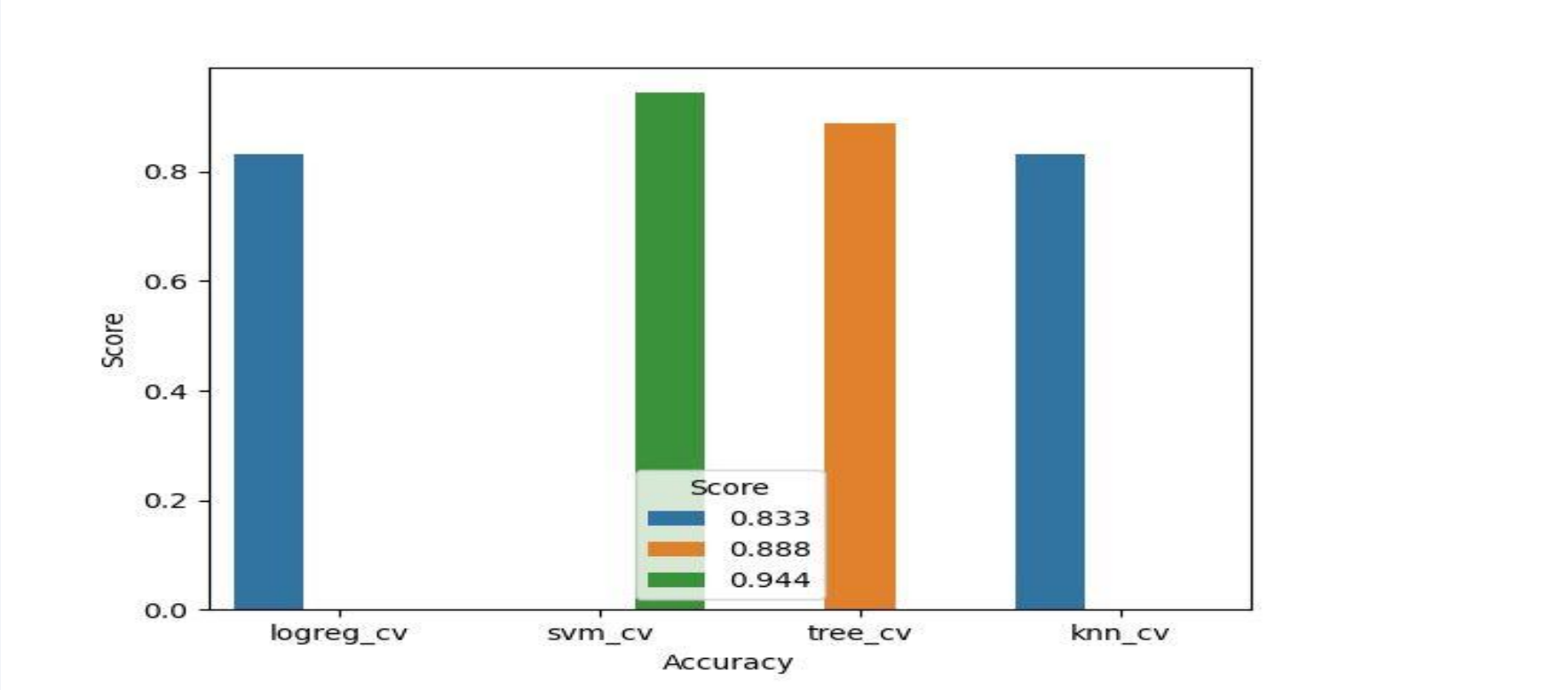
- The Launch KSC LC-39A with Payload Mass range between 2500 and 5500 Kg booster version FT, B4 and B5 are the most successful(100%)
- FT is the highest successful launch Booster version and V1.1 is the lowest launch success rate.

Section 5

Predictive Analysis (Classification)

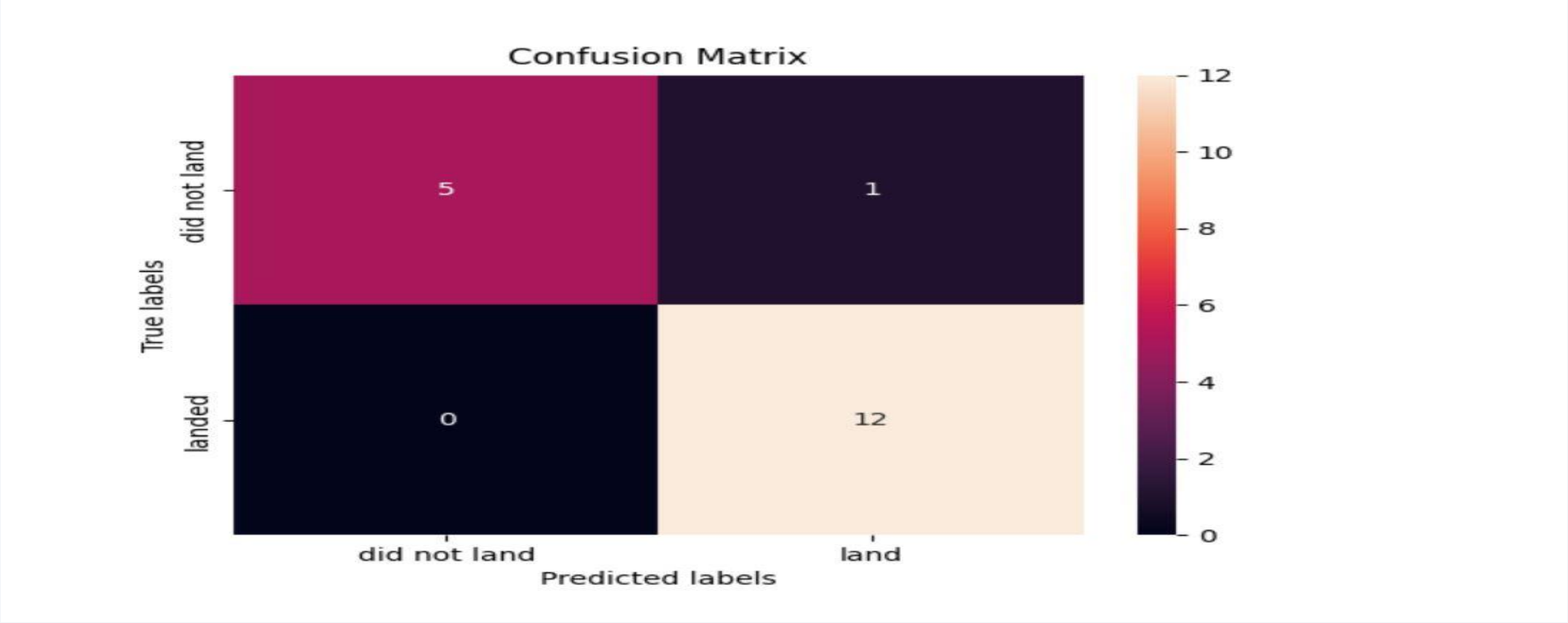
Classification Accuracy

- Visualize the built model accuracy for all built classification models, in a bar chart
- Find which model has the highest classification accuracy



Confusion Matrix

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



Conclusions

- The Location of Launch site with near proximities such ocean, railroad, and highway are important factors of the success launch.
- Payload Mass is another important factors of success
- Booster version can change the success rate

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
  - SpaceX dataset https://api.spacexdata.com/v4/launches/past
  - List of Falcon 9 and Falcon heavy Launches "https://en.wikipedia.org/w/index.php?title=List_of_Falcon_9_and_Falcon_Heavy_launches&oldid=1027686922"

Thank you!