IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Bert Roelants
07 Feb 2025

Outline

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

Executive Summary

- Methodology:
  - Data Collection & Cleaning: Pull in data sets from the SpaceX database with an API request.
  - Exploratory Data Analysis: Analyze patterns and trends of rocket model, booster model, Payload and launch locations
  - Geospatial Analysis: Using Folium to Visualize geographical data for launch sides
  - Clustering and Machine learning: K-means clustering, decision tree algorithms and regression models
- Result summary:
  - Launch Site Matters: Some launch sites have higher success rates than others due to location and weather conditions.
  - Boosters Used: Reusable boosters improve landing success and reduce costs.
  - Payload Mass Impact: Heavier payloads decrease the probability of successful landings.
  - Orbit Type: Certain orbits (e.g., Low Earth Orbit) are more favourable for successful landings.

Introduction

- SpaceX has revolutionized the aerospace industry by pioneering reusable rockets, drastically reducing launch costs. The Falcon 9 booster can be reused multiple times, cutting expenses from hundreds of millions to as low as $67 million. SpaceX's rapid innovation forces competitors like Boeing and ULA to adapt. NASA now relies on SpaceX for key missions, including the Artemis lunar lander. By making space more accessible and affordable. SpaceY a new aerospace company wants to copy the SpaceX module, the analysis in this report is the first step in the understanding of Spacex's module and learn for faults SpaceX has made, so to build a competing company with the SpaceX module.
- Problems you want to find answers:
  - What is the effect of launch site and weather on the success rate.
  - Which models of rockets and boosters have had best success rate for SpaceX
  - Which orbit and payload have higher landing success rate.
  - With the data available form SpaceX can we predict the succes rate of there lauch and landings

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - Web scraping: SpaceX launch data was scraped from Wikipedia using BeautifulSoup
  - Public data sets: SpaceX APIs
- Perform data wrangling
  - Data Cleaning: Removing incomplete or duplicate data, standardizations
  - Data transformations: Creating new variables like “Cost Efficiency.”, conversion (mass, date)
  - Data merge: Weather and geospacial data
- 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

Public Data Sets

Web scraping

SpaceX API's

Wikipedia

Python packages:
- Requests
- Numpy
-Pandas

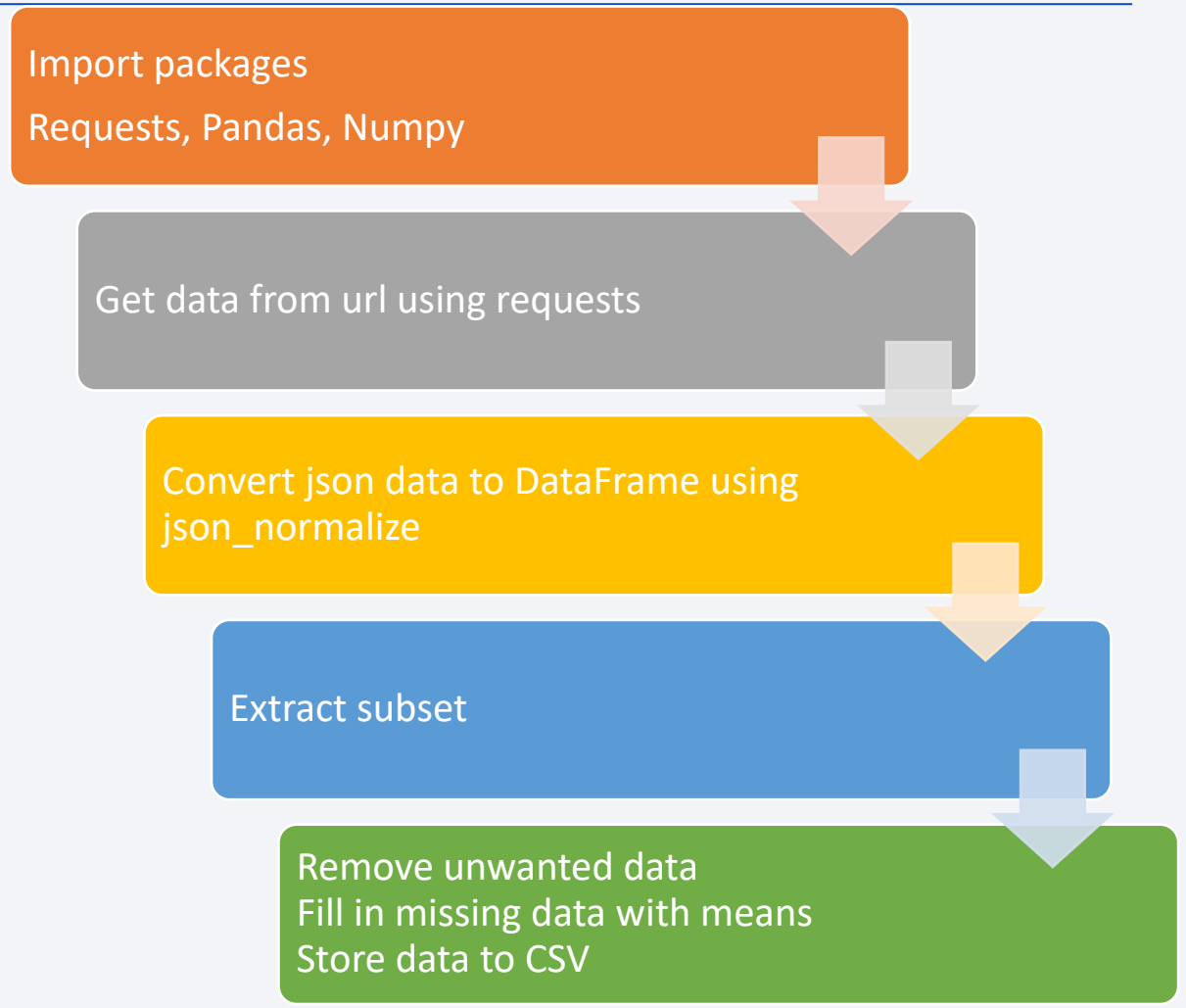
- URL to JSON
- Store in Data frame
- Save as CSV

Packages:
- Beautifull soup
-requests
-Pandas

- Beautiful soup on HTML response
-Extract columns form headers
- Parsing data to Data Frame

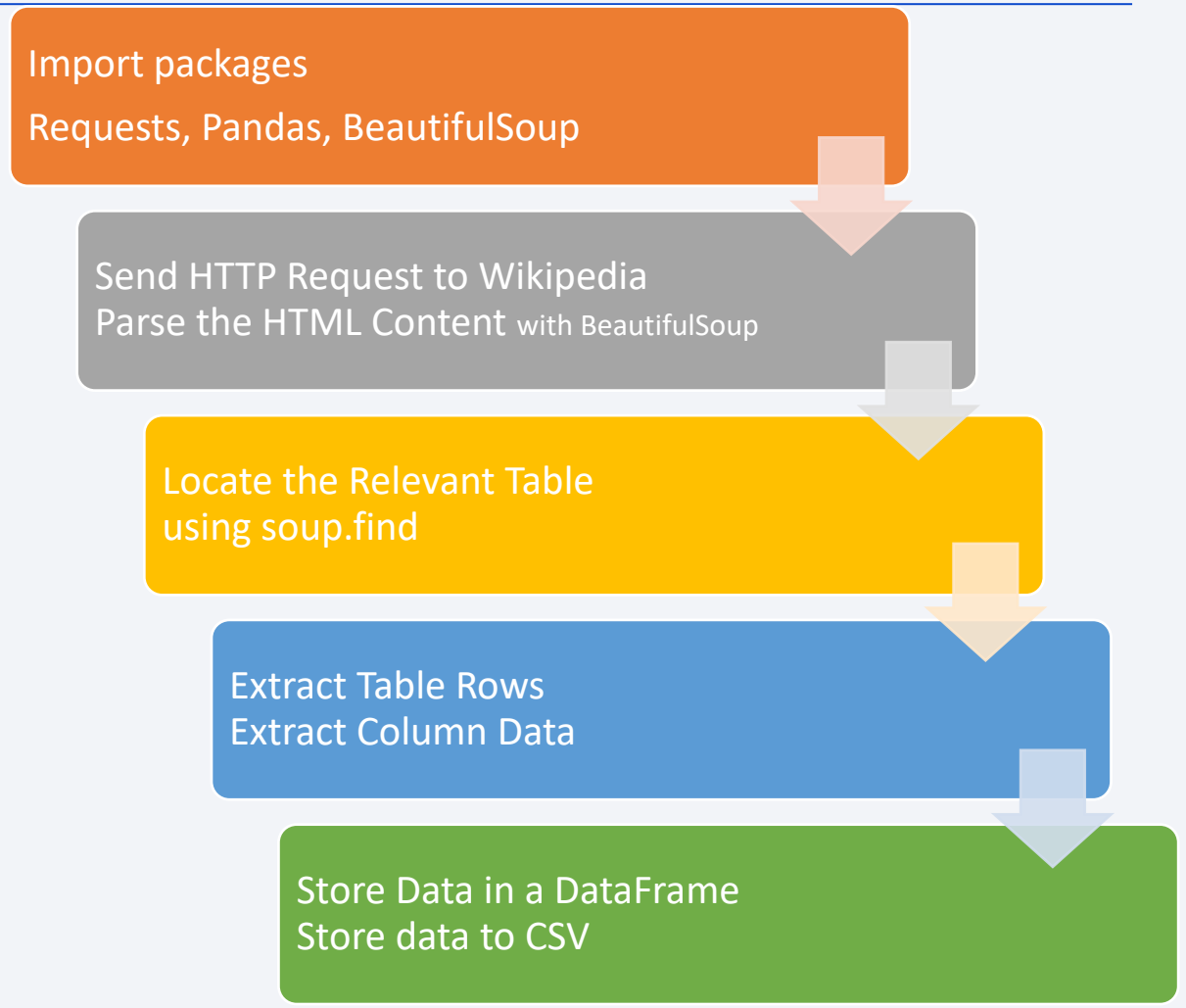
Data Collection – SpaceX API

https://github.com/roelantsbert/IBM-Applied-Data-Science--Capstone/blob/main/jupyter-labs-spacex-data-collection-api.ipynb



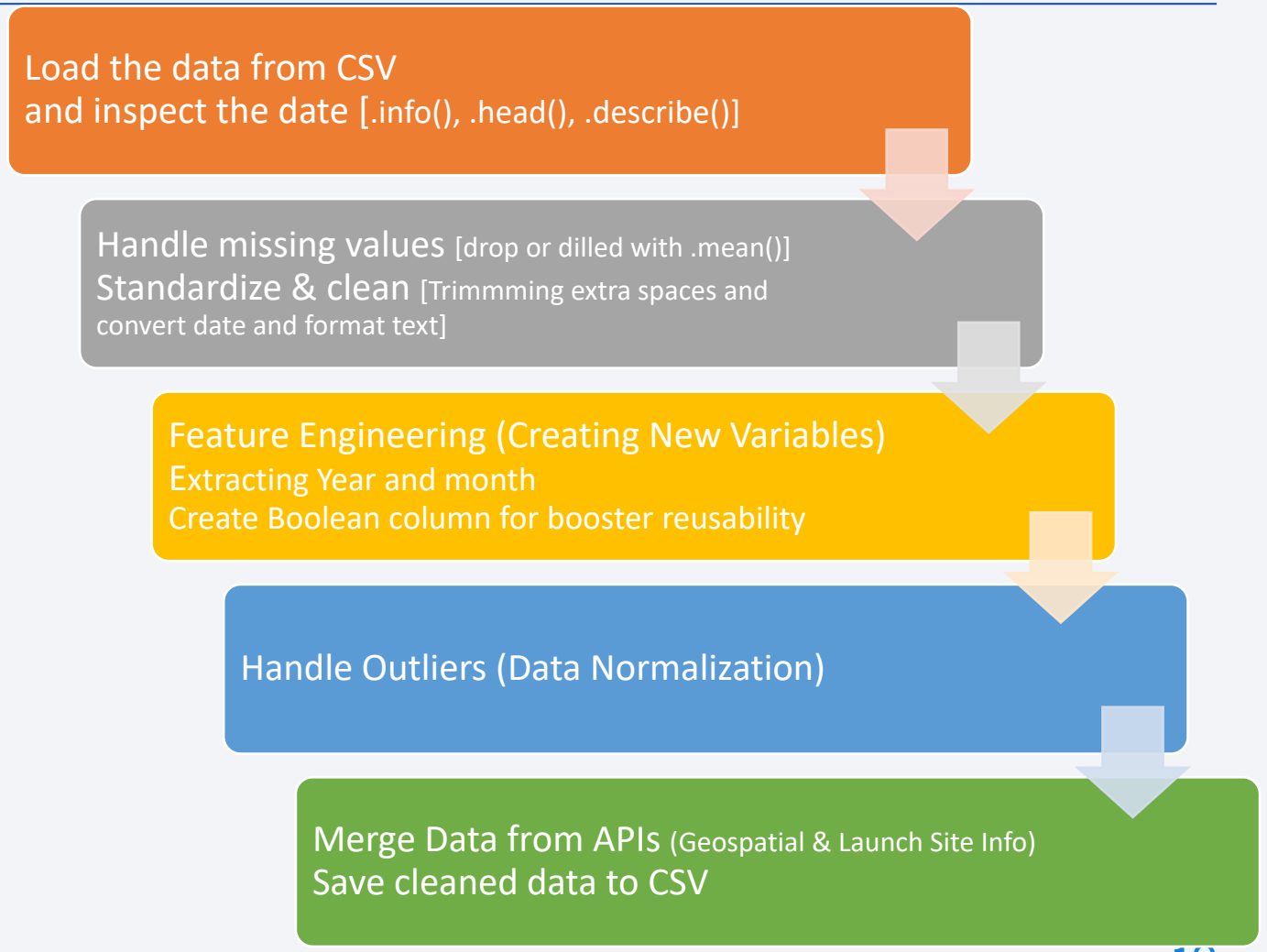
Data Collection - Scraping

https://github.com/roelantsbert/IBM-Applied-Data-Science--Capstone/blob/main/jupyter-labs-webscraping.ipynb



Data Wrangling

https://github.com/roelantsbert/IBM-Applied-Data-Science--Capstone/blob/main/labs-jupyter-spacex-Data%20wrangling.ipynb



EDA with Data Visualization

- Scatter Plots:
  - success rate plotted on (Payload Mass vs Flight number, Launch Site vs Flight number, Launch site vs Payload Mass, Orbit vs Flight Number & Orbit vs Payload) This to id the factors that influence the success rate)
- Line chart:
  - Success Rate vs Time, to determine how success rate increased over time
- Bar chart
  - Success rate vs the orbit types, visualise relationship between success rate and the orbit
- https://github.com/roelantsbert/IBM-Applied-Data-Science--Capstone/blob/main/EDA%20with%20visualization.ipynb

EDA with SQL

- Filter Successful Landings
- Count Total Launches per Launch Site
- Calculate Landing Success Rate per Launch Site
- Find the Average Payload Mass for Successful Landings
- Find Launches with the Heaviest Payloads
- Find the Most Frequent Orbit Types Used
- Compare Launch Frequency Over Time
- Find the Success Rate of Reusable Boosters
- Calculate Cost Efficiency Based on Reusability

https://github.com/roelantsbert/IBM-Applied-Data-Science--Capstone/blob/main/jupyter-labs-eda-sql-coursera_sqllite.ipynb

Build an Interactive Map with Folium

Map Objects Added

1. Markers (folium.Marker)
   - Placed at each SpaceX launch site using their latitude and longitude.
   - Used to visually identify where launches occurred.

2. Circle Markers (folium.CircleMarker)
   - Added around each launch site to highlight its location more clearly.
   - The radius was adjusted based on launch frequency to indicate site activity.

3. Lines (folium.PolyLine)
   - Used to connect launch sites to target landing locations.
   - Helped visualize booster landing patterns and distances.

4. Popups & Tooltips
   - Included site names, success rates, and additional information in popups.
   - Allowed users to interact with the map and gain insights.

- https://github.com/roelantsbert/IBM-Applied-Data-Science--Capstone/blob/main/lab_jupyter_launch_site_location.ipynb

Build a Dashboard with Plotly Dash

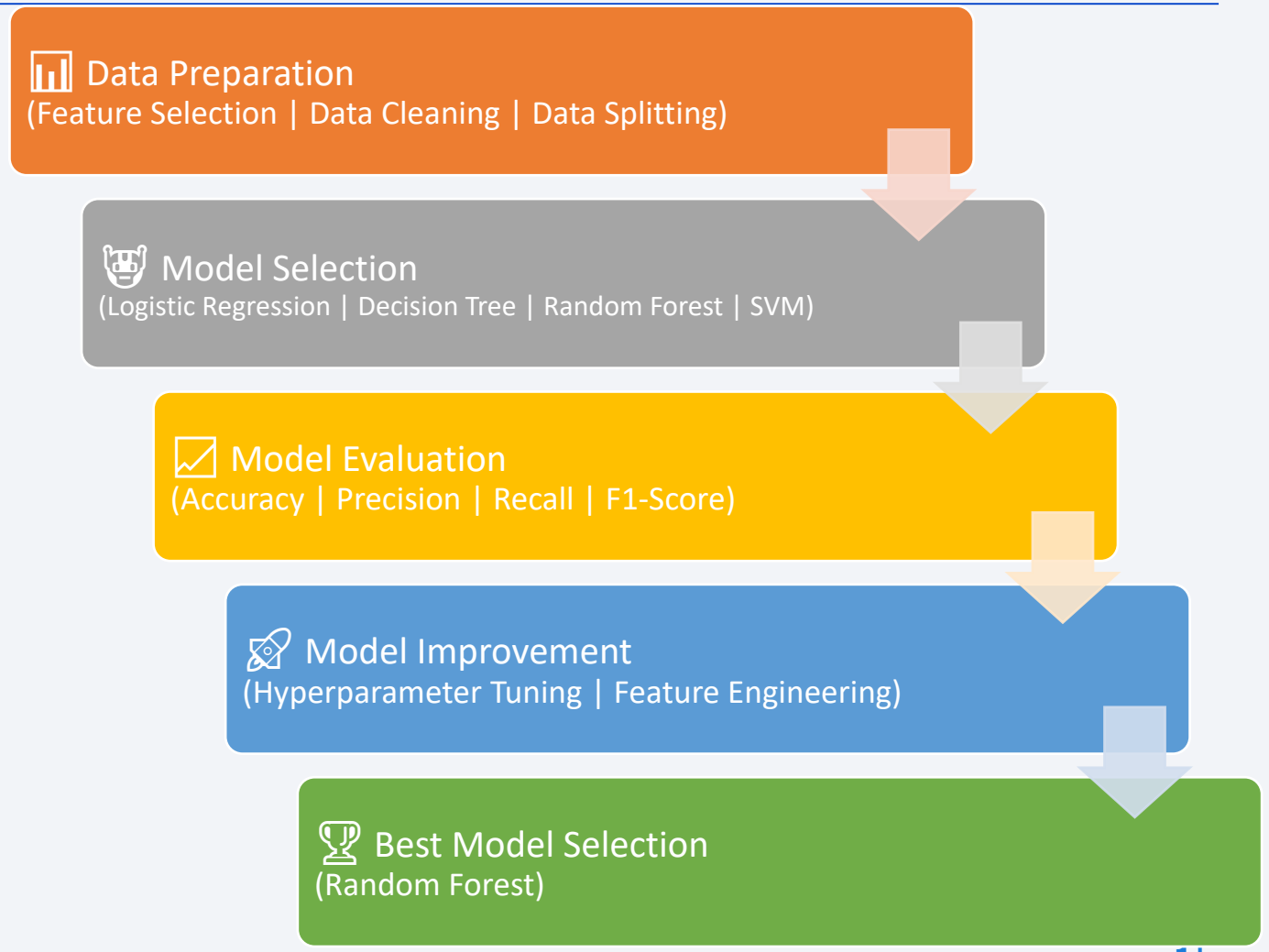
Plots/Graphs Added

1. Pie Chart
   - Placed at each SpaceX launch site using their latitude and longitude.
   - Used to visually identify where launches occurred.
2. Scatter Plot
   - Displays payload mass vs. landing success.
   - Helps analyze how payload weight impacts landing success.
3. Bar Chart
   - Represents the number of launches per launch site.
   - Identifies the most active launch locations.
4. Line Chart
   - Shows launch trends over time.
   - Helps track SpaceX's launch frequency growth.

- https://github.com/roelantsbert/IBM-Applied-Data-Science--Capstone/blob/main/spacex_dash_app%20(1).py

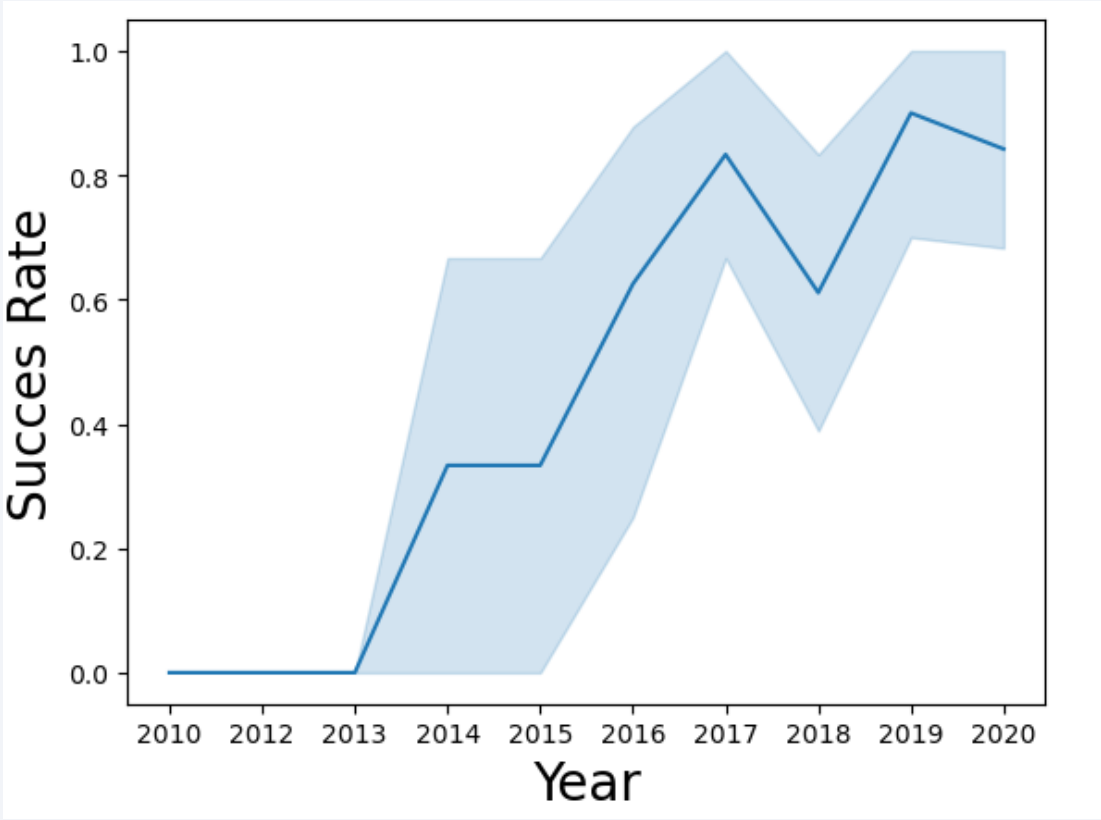
Predictive Analysis (Classification)

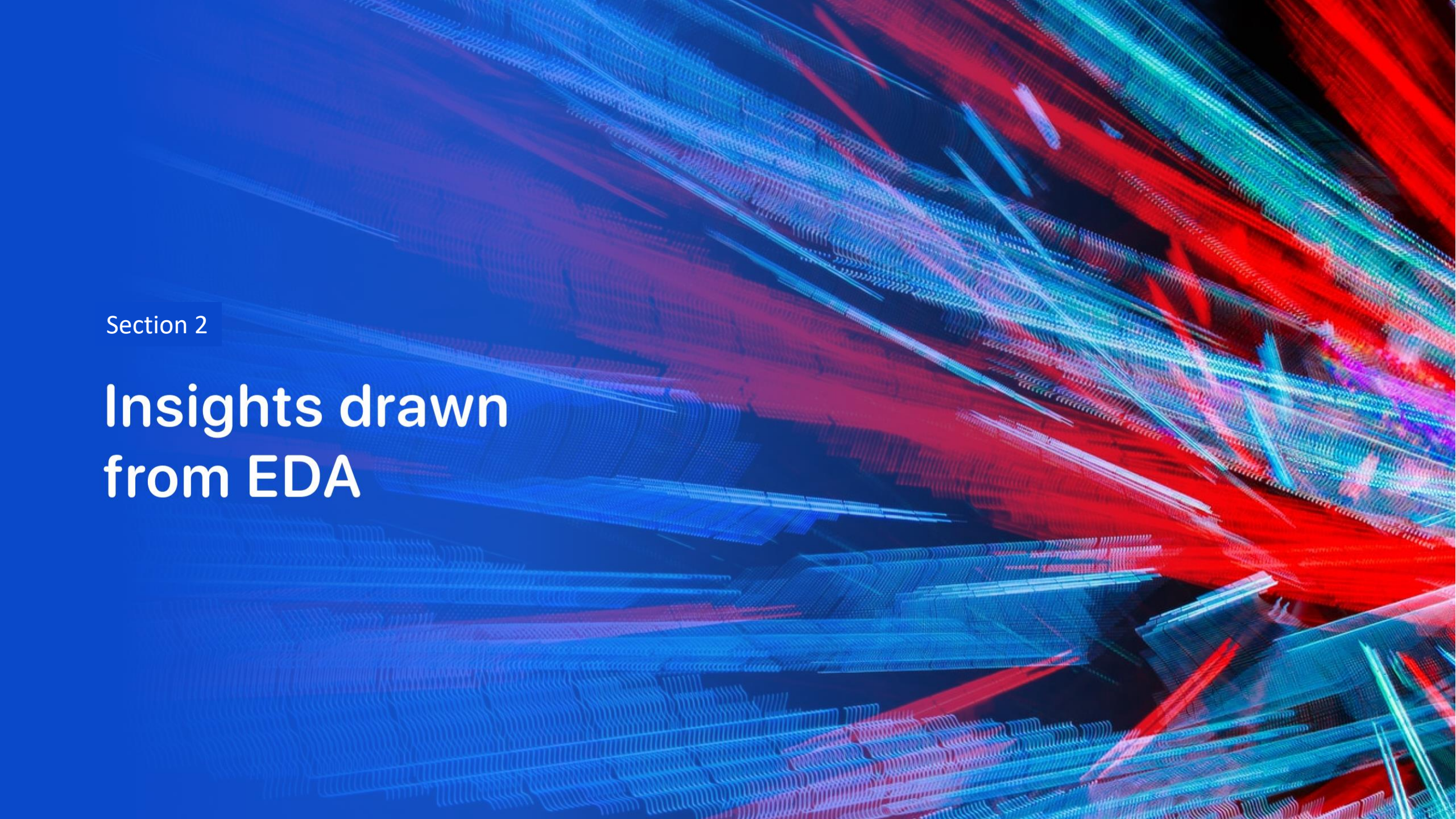
https://github.com/roelantsbert/IBM-Applied-Data-Science--Capstone/blob/main/SpaceX_Machine%20Learning%20Prediction_Part_5%20(1).ipynb



Results

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

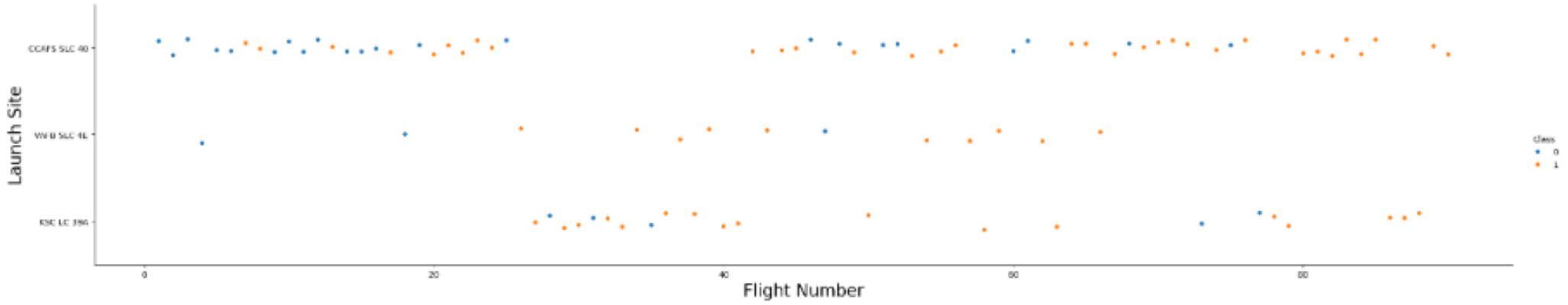



Section 2
Insights drawn from EDA

Flight Number vs. Launch Site

In [5]:

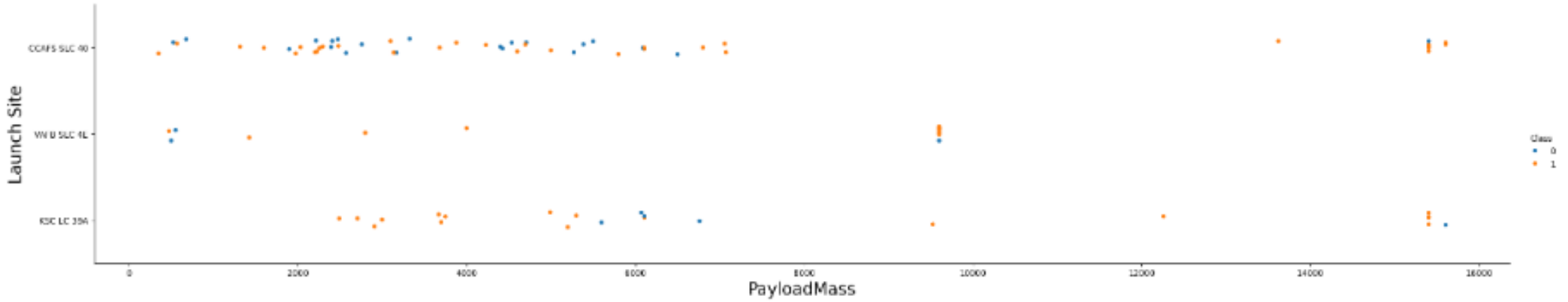
```
# Plot a scatter point chart with x axis to be Flight Number and y axis to be the launch site, and hue to be the class valu
sns.catplot(y="LaunchSite", x="FlightNumber", hue="Class", data=df, aspect = 5)
plt.xlabel('Flight Number', fontsize=20)
plt.ylabel('Launch Site', fontsize=20)
plt.show()
```



Payload vs. Launch Site

In [8]:

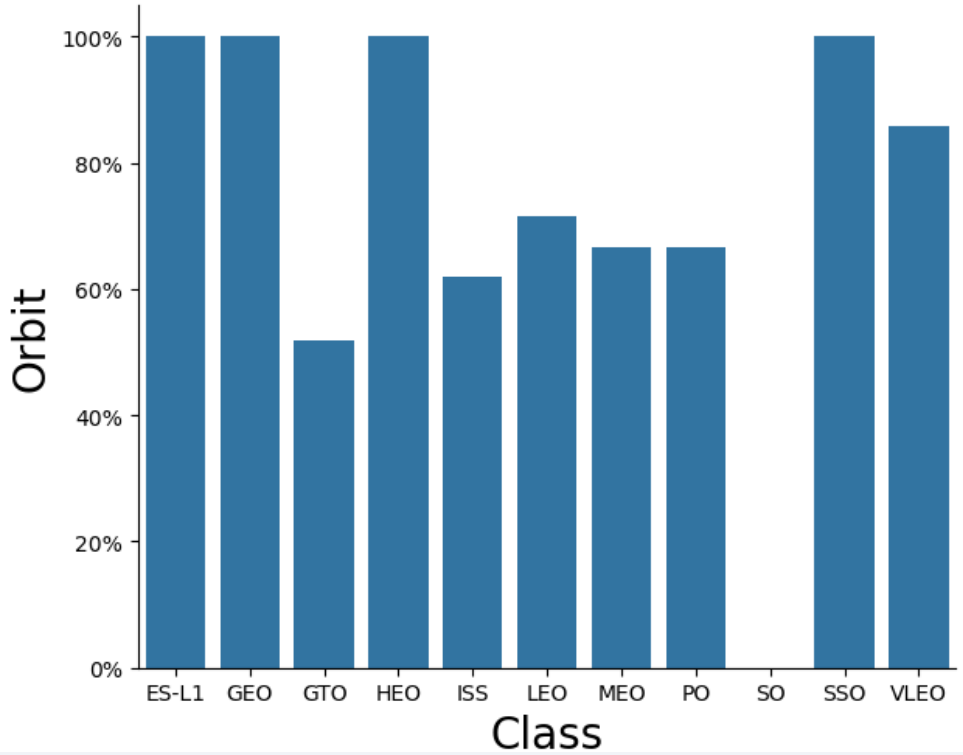
```
# Plot a scatter point chart with x axis to be Pay Load Mass (kg) and y axis to be the launch site, and hue to be the class
sns.catplot(y="LaunchSite", x="PayloadMass", hue="Class", data=df, aspect = 5)
plt.xlabel('PayloadMass', fontsize=20)
plt.ylabel('Launch Site', fontsize=20)
plt.show()
```



Success Rate vs. Orbit Type

In [18]:

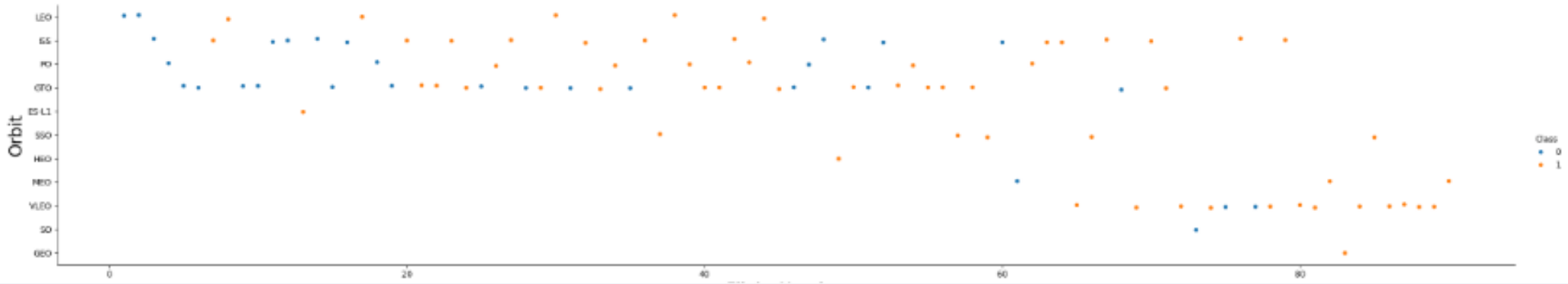
```
# HINT use groupby method on Orbit column and get the mean of Class column
from matplotlib.ticker import PercentFormatter
plot = sns.catplot(y="Class", x="Orbit",  data=df.groupby('Orbit')['Class'].mean().reset_index(), kind = 'bar', aspect=1.2)
plt.xlabel('Class', fontsize=20)
plt.ylabel('Orbit', fontsize=20)
for ax in plot.axes.flat:
    ax.yaxis.set_major_formatter(PercentFormatter(1))
plt.show()
```



Flight Number vs. Orbit Type

In [19]:

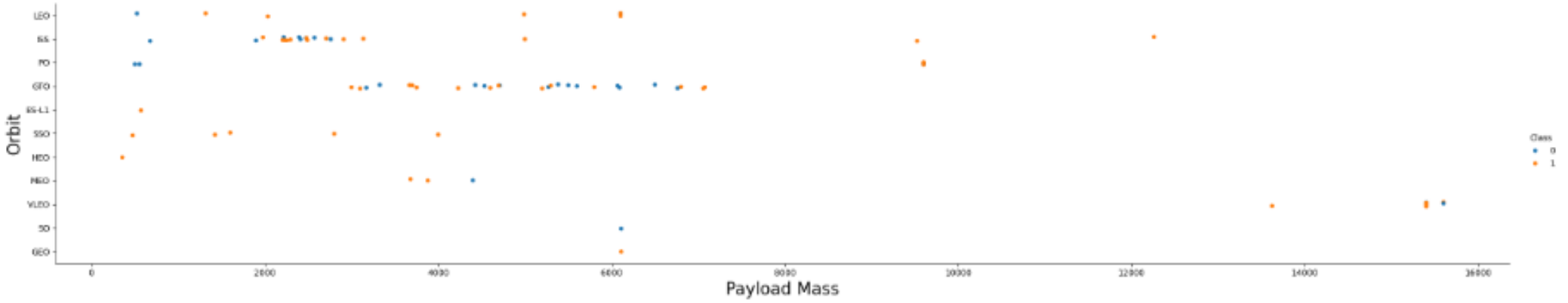
```
# Plot a scatter point chart with x axis to be FlightNumber and y axis to be the Orbit, and hue to be the class value
sns.catplot(y = 'Orbit', x = 'FlightNumber', hue = 'Class', data = df, aspect = 5)
plt.xlabel('Flight Number', fontsize = 20)
plt.ylabel('Orbit', fontsize = 20)
plt.show()
```



Payload vs. Orbit Type

In [21]:

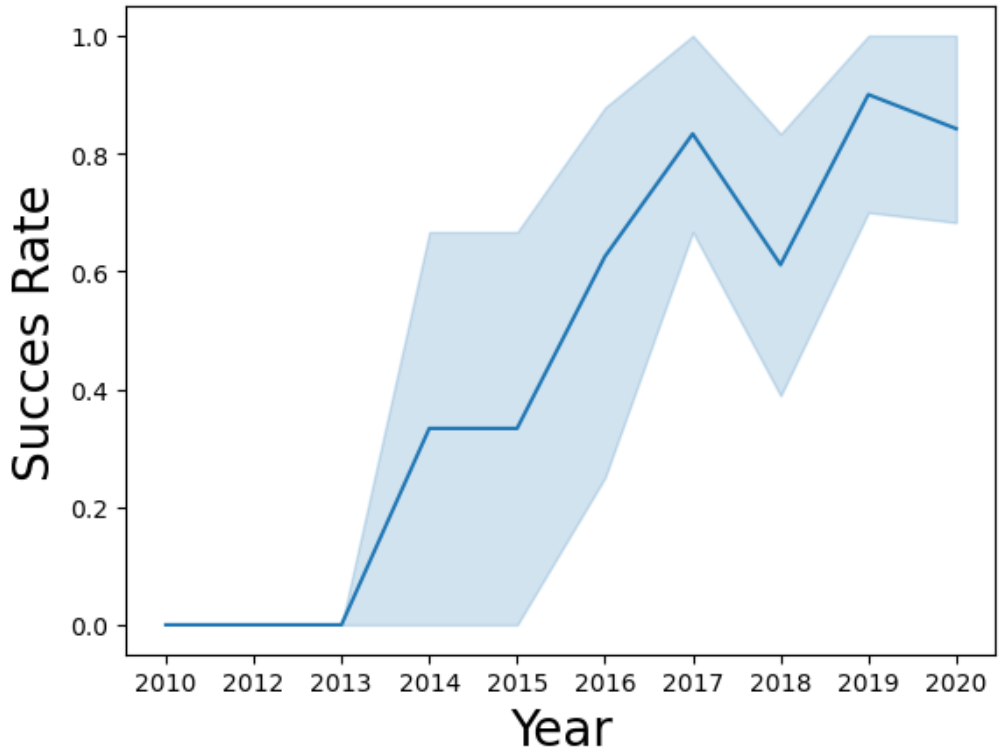
```
# Plot a scatter point chart with x axis to be Payload Mass and y axis to be the Orbit, and hue to be the class value
sns.catplot(y = 'Orbit', x = 'PayloadMass', hue = 'Class', data = df, aspect = 5)
plt.xlabel('Payload Mass', fontsize = 20)
plt.ylabel('Orbit', fontsize = 20)
plt.show()
```



Launch Success Yearly Trend

In [42]:

```
# Plot a line chart with x axis to be the extracted year and y axis to be the success rate
sns.lineplot(y = 'Class', x = 'Date', data = df)
plt.xlabel('Year', fontsize = 20)
plt.ylabel('Succes Rate', fontsize = 20)
plt.show()
```



All Launch Site Names

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

```
%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'

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

In [12]:
```
sql SELECT * FROM SPACEXTBL WHERE LAUNCH_SITE LIKE 'CCA%' LIMIT 5;
```

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

Out[12]:

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

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

In [13]:
```
sql SELECT SUM(PAYLOAD_MASS__KG_) AS TOTAL_PAYLOAD FROM SPACEXTBL WHERE PAYLOAD  LIKE '%CRS%';
```

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

Out[13]:

TOTAL_PAYLOAD
111268

Average Payload Mass by F9 v1.1

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

In [14]:
```
sql SELECT AVG(PAYLOAD_MASS__KG_) AS AVG_PAYLOAD FROM SPACEXTBL WHERE BOOSTER_VERSION = 'F9 v1.1';
```

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

Out[14]:

AVG_PAYLOAD
2928.4

First Successful Ground Landing Date

Task 5

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

Hint:Use min function

In [18]:
```
sql SELECT MIN(DATE) AS FIRST_SUCCESS_GP FROM SPACEXTBL WHERE Landing_Outcome = 'Success (ground pad)';
```

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

Out[18]:

FIRST_SUCCESS_GP
2015-12-22

Successful Drone Ship Landing with Payload between 4000 and 6000

List the names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000

In [19]:
```
sql SELECT DISTINCT BOOSTER_VERSION FROM SPACEXTBL WHERE PAYLOAD_MASS__KG_ BETWEEN 4000 AND 6000 AND Landing_Outcome = 'Suc
```

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

Out[19]:

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

Total Number of Successful and Failure Mission Outcomes

Task 7

List the total number of successful and failure mission outcomes

In [17]:
```
sql SELECT MISSION_OUTCOME, COUNT(*) AS QTY FROM SPACEXTBL GROUP BY MISSION_OUTCOME ORDER BY MISSION_OUTCOME;
```

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

Out[17]:

Mission_Outcome	QTY
Failure (in flight)	1
Success	98
Success	1
Success (payload status unclear)	1

Boosters Carried Maximum Payload

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

In [21]:
```
sql SELECT DISTINCT BOOSTER_VERSION FROM SPACEXTBL WHERE PAYLOAD_MASS__KG_ = (SELECT MAX(PAYLOAD_MASS__KG_) FROM SPACEXTBL)
```

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

Out[21]:

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

2015 Launch Records

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.

Note: SQLLite does not support monthnames. So you need to use substr(Date, 6,2) as month to get the months and substr(Date,0,5)='2015' for year.

In [29]:
```
sql SELECT select substr(Date, 6,2) as month, substr(Date,0,5) as year , date, booster_version, launch_site, landing__outcor
```

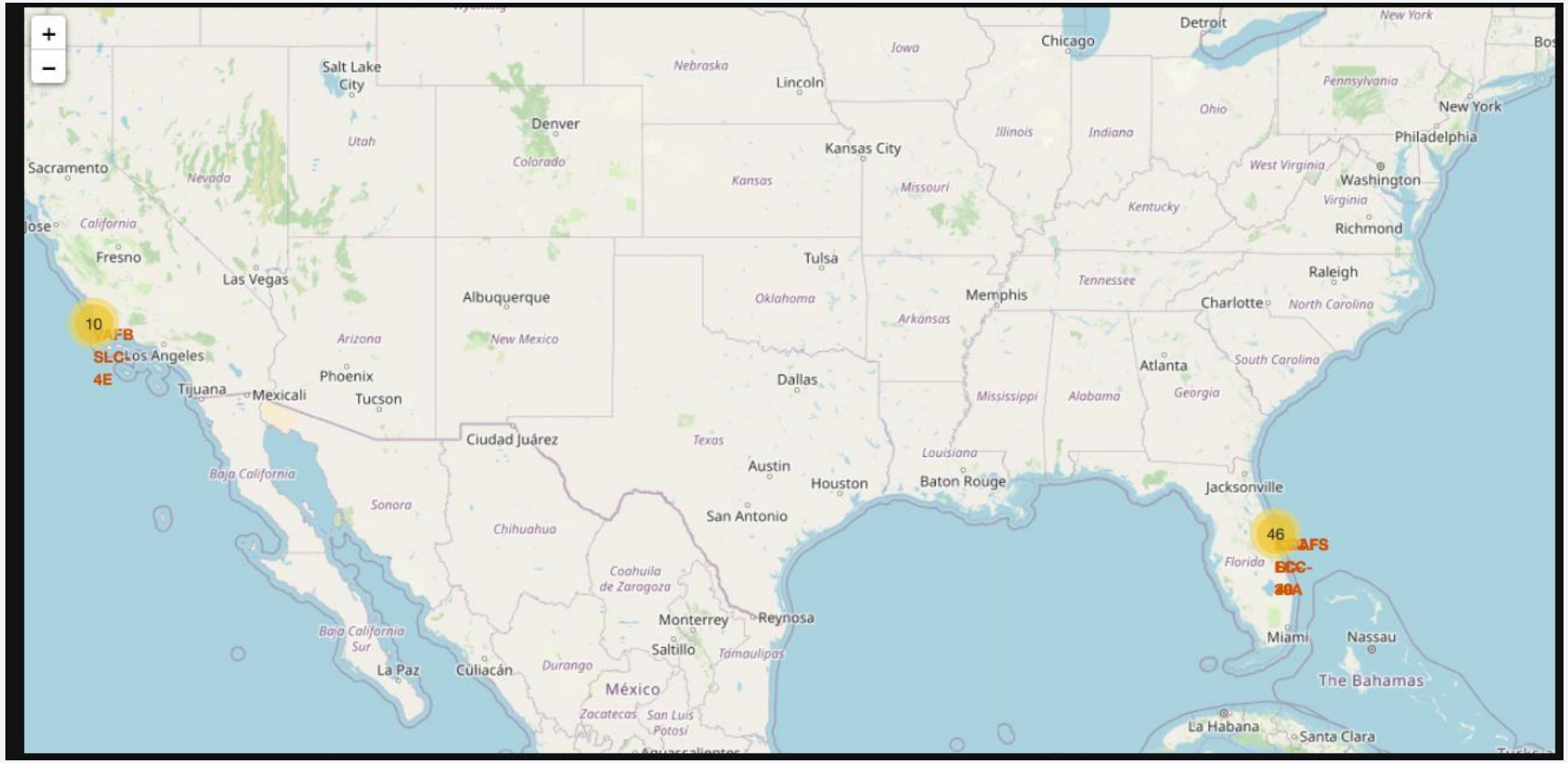
```
 * sqlite:///my_data1.db
(sqlite3.OperationalError) near "select": syntax error
[SQL: SELECT select substr(Date, 6,2) as month, substr(Date,0,5) as year , date, booster_version, launch_site, landing__outco
me from SPACEXTBL where landing__outcome = 'Failure (drone ship)' and substr(Date,0,5)='2015';]
(Background on this error at: https://sqlalche.me/e/20/e3q8)
```

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

```
 * sqlite:///my_data1.db
(sqlite3.OperationalError) near "select": syntax error
[SQL: SELECT select substr(Date, 6,2) as month, substr(Date,0,5) as year , date, booster_version, launch_site, landing__outco
me from SPACEXTBL where landing__outcome = 'Failure (drone ship)' and substr(Date,0,5)='2015';]
(Background on this error at: https://sqlalche.me/e/20/e3q8)
```

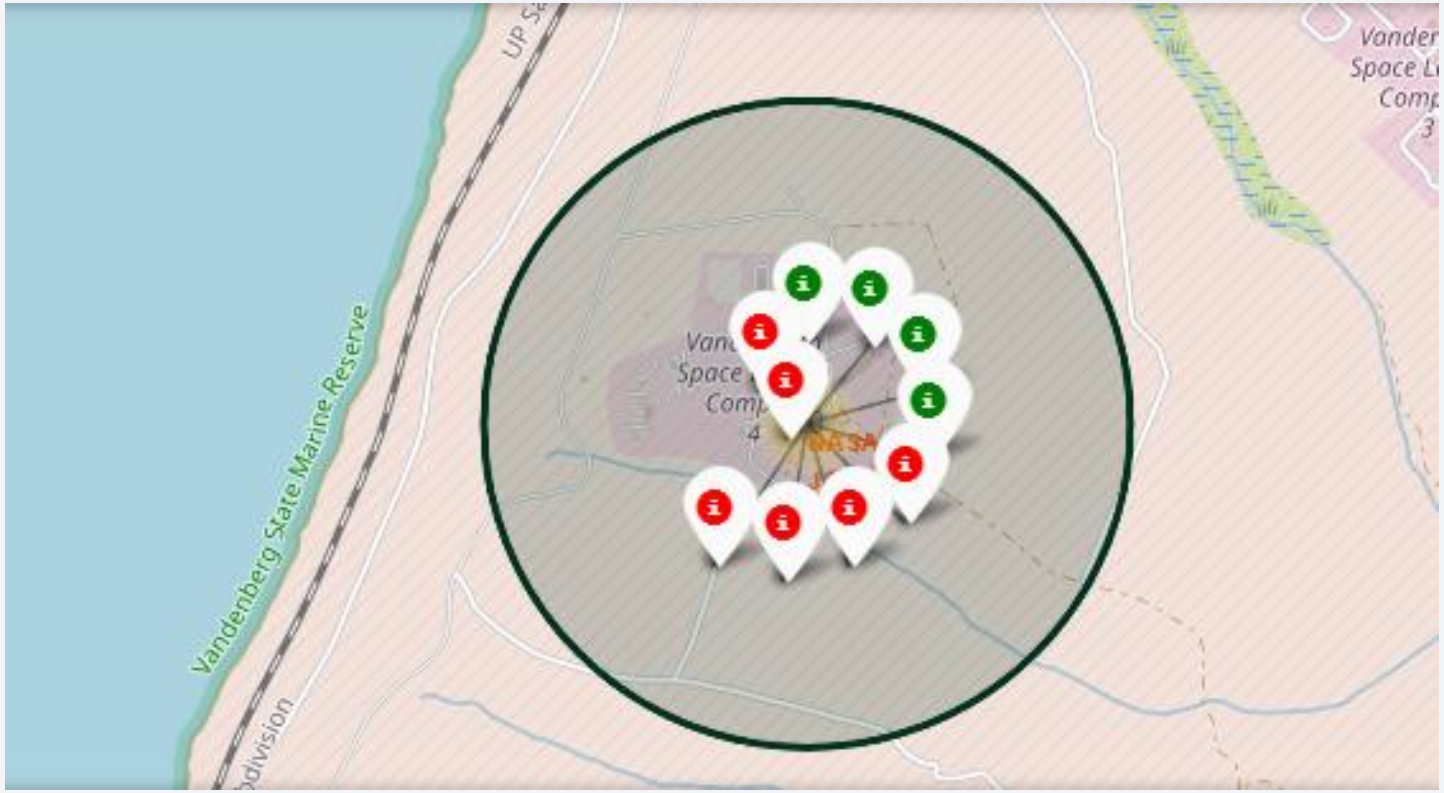

Section 3
Launch Sites
Proximities Analysis

Launch Site Locations



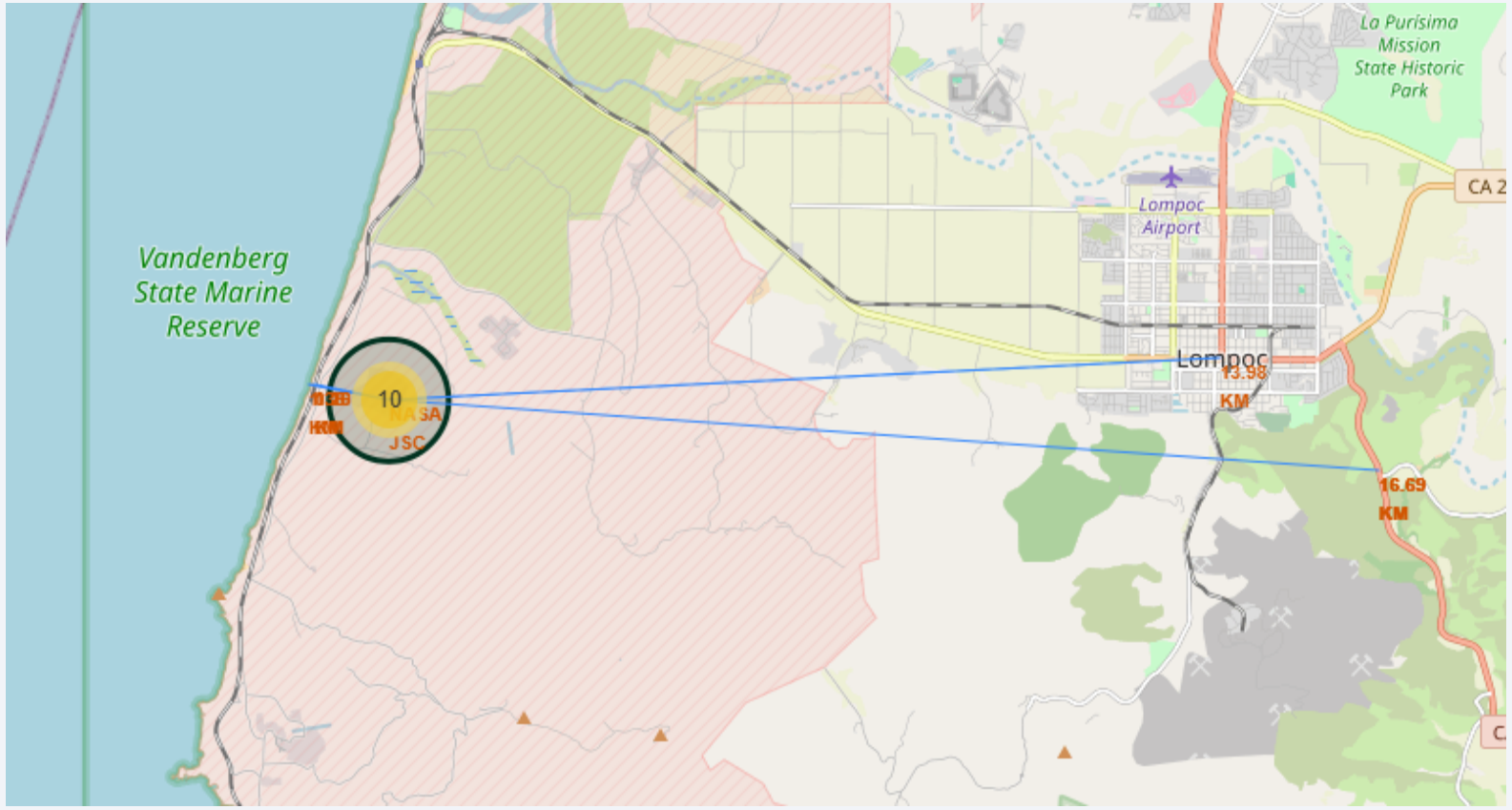
- Landing sides show by the red dots
  - 1 Landing site on the east coast
  - 1 landing site on the west coast

NASA JSC Launch site

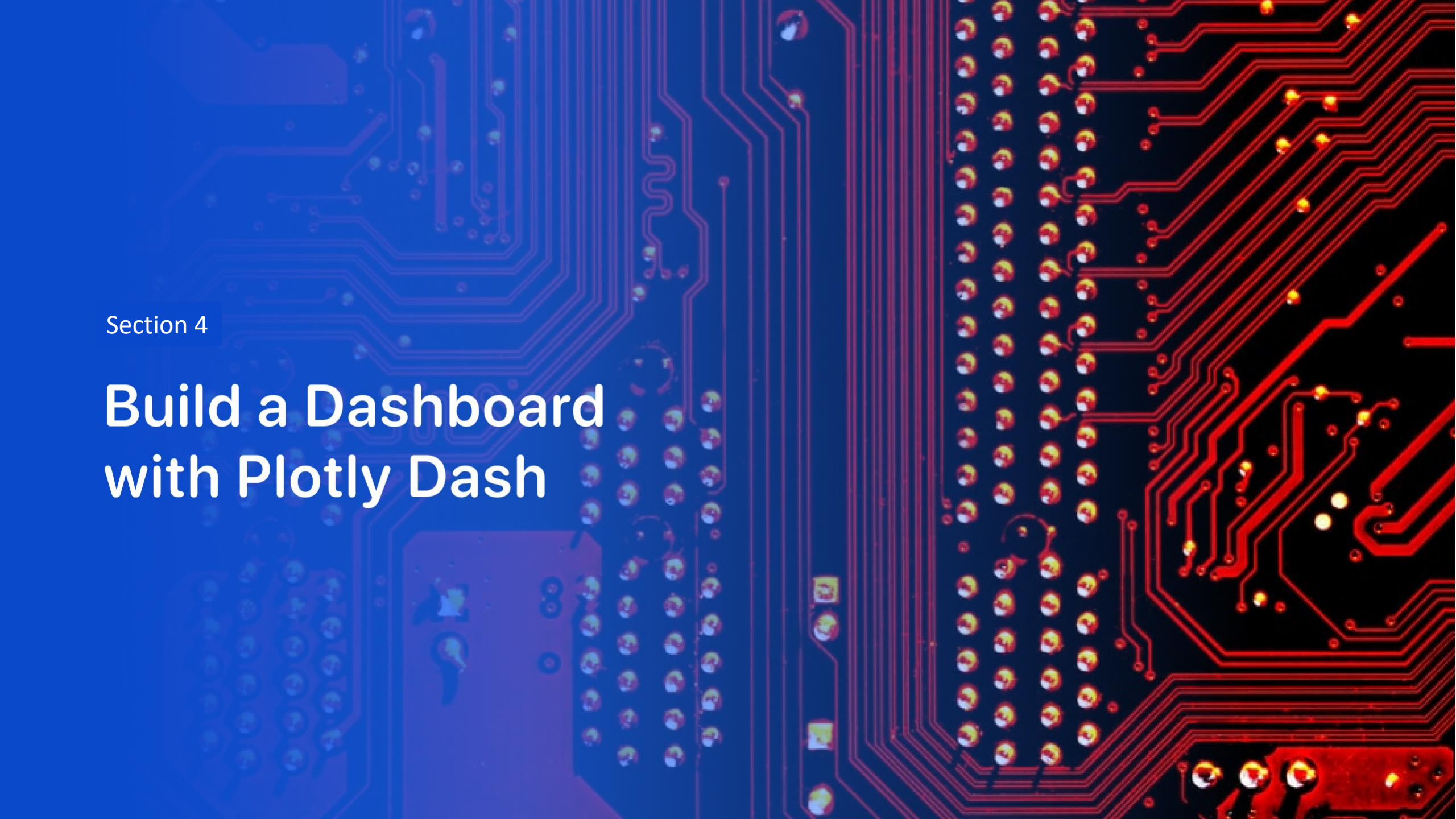


- Launch site NSA JSC
  - Each launch visualized with a pin, Colours of the pins represent the succes of the launch

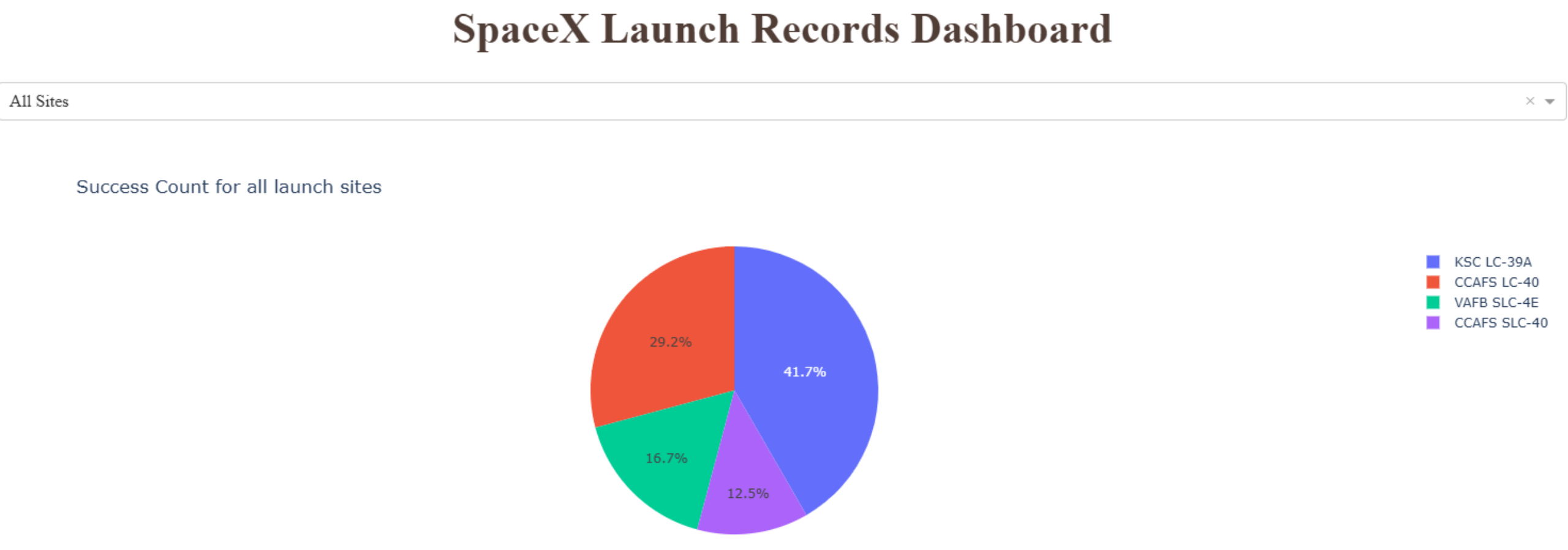
NASA JSC Launch Site Proximity Analysis



- Visualization of important infrastructures near the NASA JSC launch site
  - Coast Line: 1.36 km
  - Railway: 1.16 km
  - City: 13.98 km
  - Highway: 16.69 km
- The location of the infrastructures is important to know to mitigate risk or arrange transport parts to launch site


Section 4
Build a Dashboard with Plotly Dash

Dashboard Launch Records

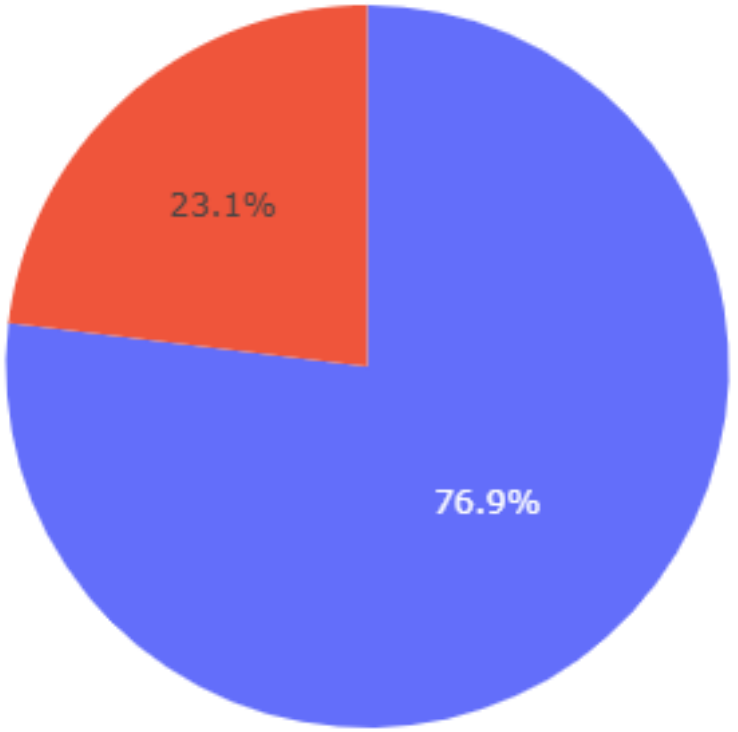


- KSC LC-39A launch site has been the most used Launch site for SpaceX

Dashboard Launch Records

Total Success Launches for KSC LC-39A

- CCAFS LC-40 has a success rate of 73.1%
- VAFB SLC-4E has a success rate of 60.0%
- KSC LC-39A has a success rate of 76.9%
  - Has the highest success rate off all sites
- CCAFS SLC-40 has a success rate of 57.1%



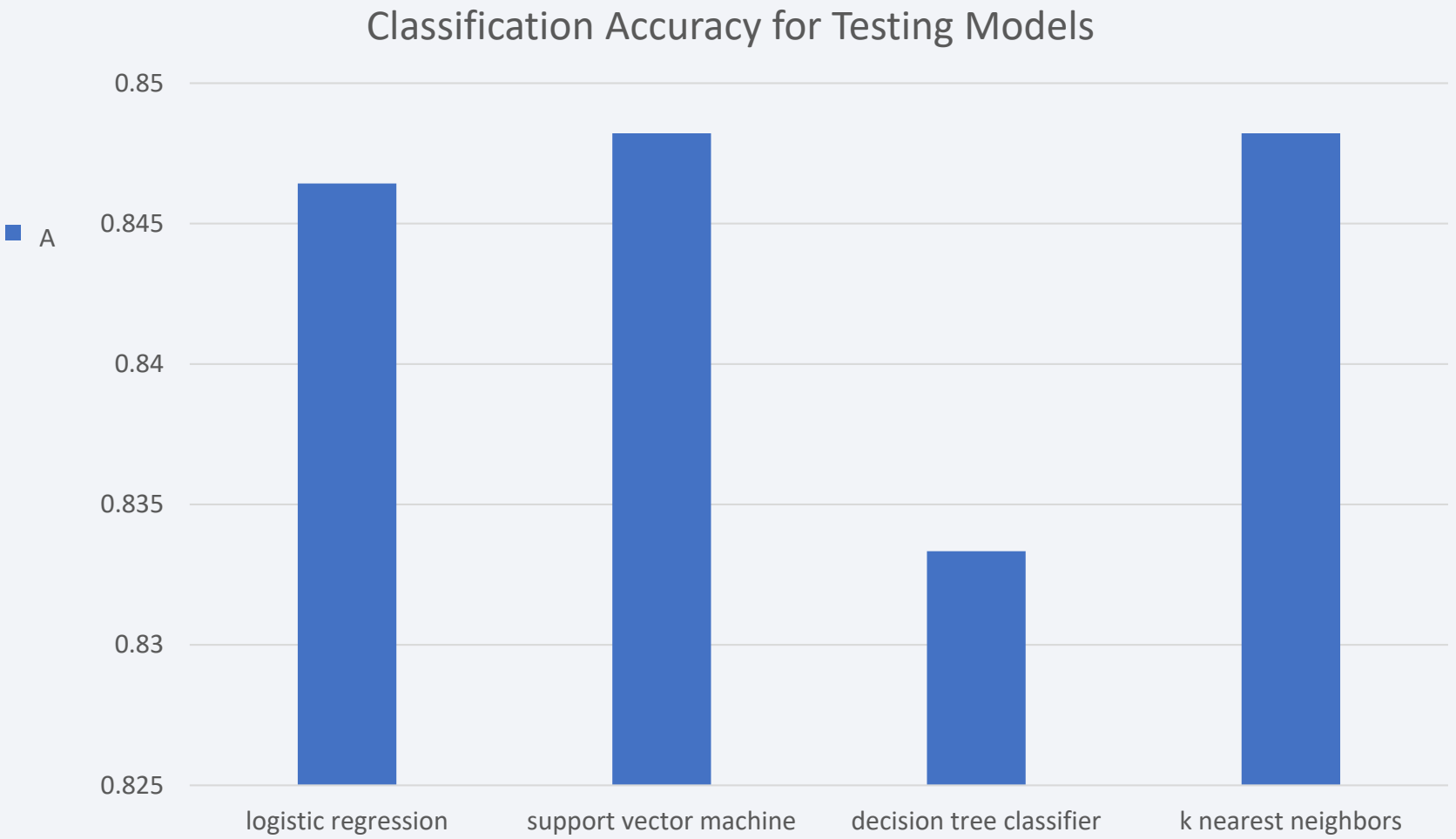
Dashboard Launch Records

- Successful launches are mor likely between 2500 and 5500. Although more analysis is needed as in previous analysis is show that the payload in early launches was lower this could show why the low payloads have such a low succes rate.

Section 5

Predictive Analysis (Classification)

Classification Accuracy

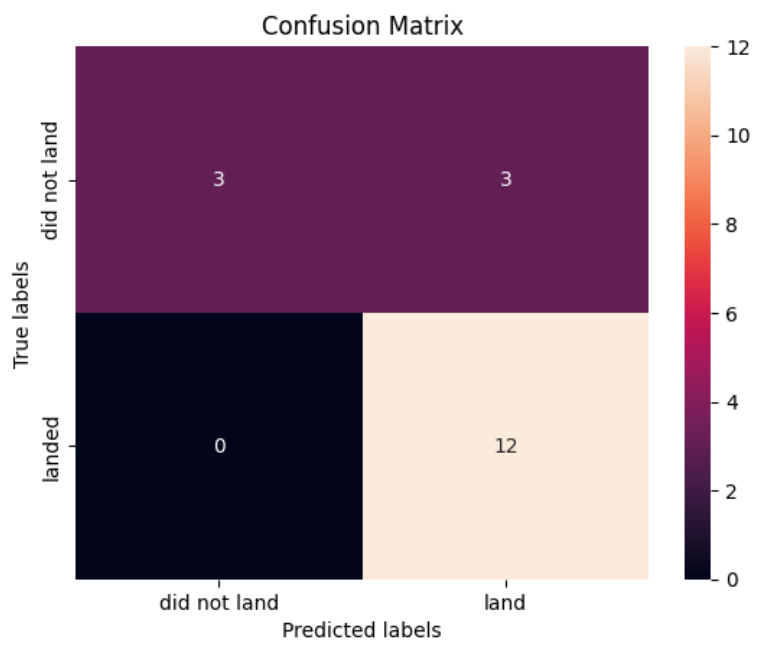


Support Vector Machine And K nearest neighbours have the best accuracy

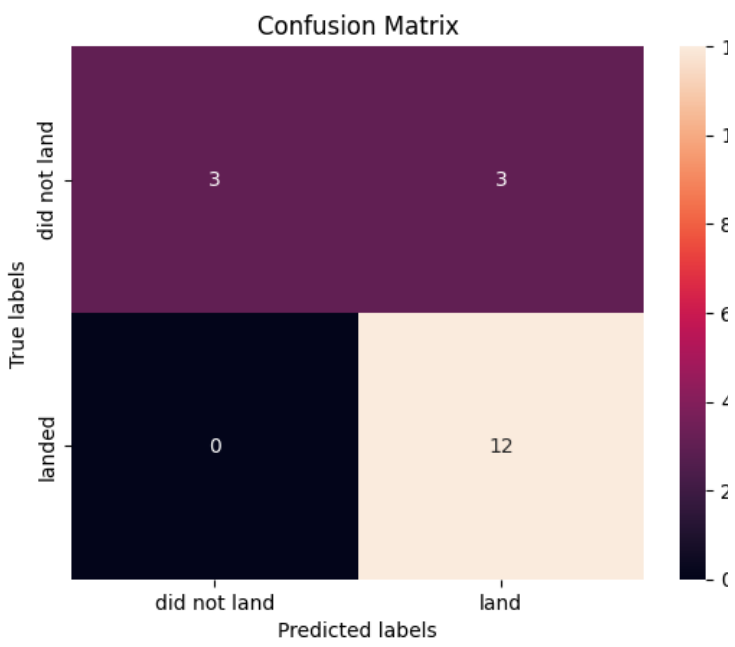
Confusion Matrix

- All model show similar outcomes

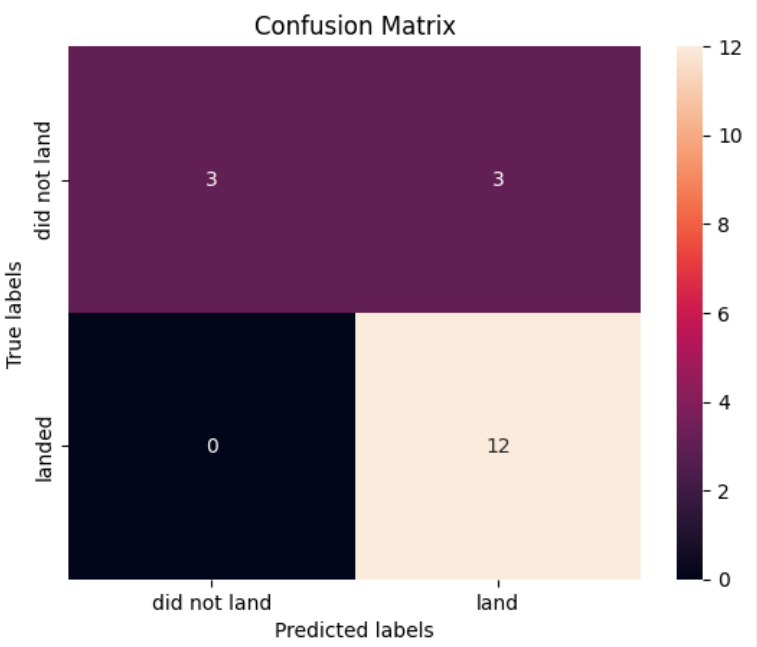
Logistic regression



Decision Tree Classifier



K Nearest Neighbour



Conclusions

- SpaceX's Launch Success is Influenced by Multiple Factors
  - The analysis revealed that payload mass, launch site, and booster type significantly impact landing success rates.
- Certain Launch Sites Have Higher Success Rates
  - Some locations, such as CCAFS SLC-40 and KSC LC-39A, had higher successful landing rates compared to others.
- Higher Payloads Reduce Landing Success
  - The classification model showed that heavier payloads often resulted in lower landing success rates.
- Both K Nearest Neighbored and Support Vector Machine as best prediction models

Appendix

- Data Analysis code stored on GitHub:
  - https://github.com/roelantsbert/IBM-Applied-Data-Science--Capstone/tree/main

Thank you!