

Winning Space Race with Data Science

Assel Zhauletbayeva

February 26, 2024

Capstone Project



Outline

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

Executive Summary

- Summary of methodologies:

Data collection through API and webscraping

Exploratory Data Analysis with the help of Data Visualization

EDA with SQL

Interactive Map with Folium

Dashboard with Plotly Dash

Predictive Analysis

- Summary of all results:

Exploratory Data Analysis results

Interactive Maps and Dashboards

Predictive results

Introduction

- Project background and context:

Because the Falcon 9's first stage could be reused, SpaceX was able to reduce costs to a point where it completely changed the aerospace sector.

After launch, the first stage of the Falcon 9 can return to Earth and be repurposed for a different mission.

This capability changes the game for the business by drastically lowering the cost of space travel.

- Problems you want to find answers:

Using data gathered from prior launches, apply machine learning models to forecast whether the first stage of a Falcon 9 rocket will land successfully.

What are the main characteristics of successful landings?

Methodology



Methodology

Executive Summary

Data collection methodology:

- 1.SpaceX REST API calls
2. Web scraping Wikipedia's 'List of Falcon 9 and Falcon Heavy launches' using BeautifulSoup.

Data wrangling:

- 1.Dropping unnecessary values
2. One Hot Encoding for classification models

Exploratory data analysis (EDA) using visualization and SQL

Interactive visual analytics using Folium and Plotly Dash

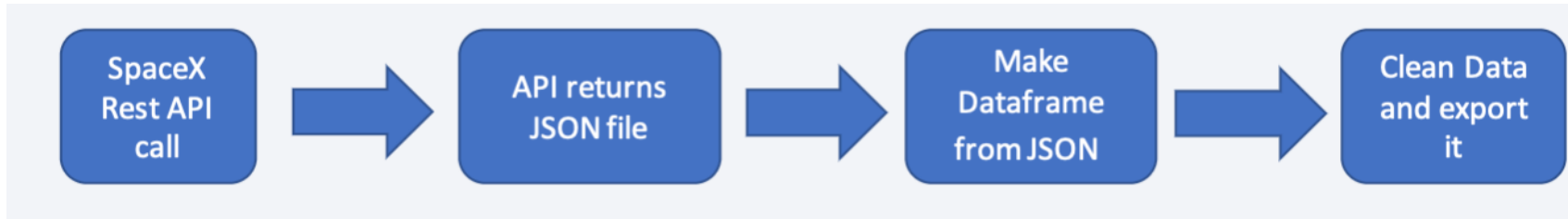
Predictive analysis using classification models

- How to build, tune, evaluate classification models

Data Collection

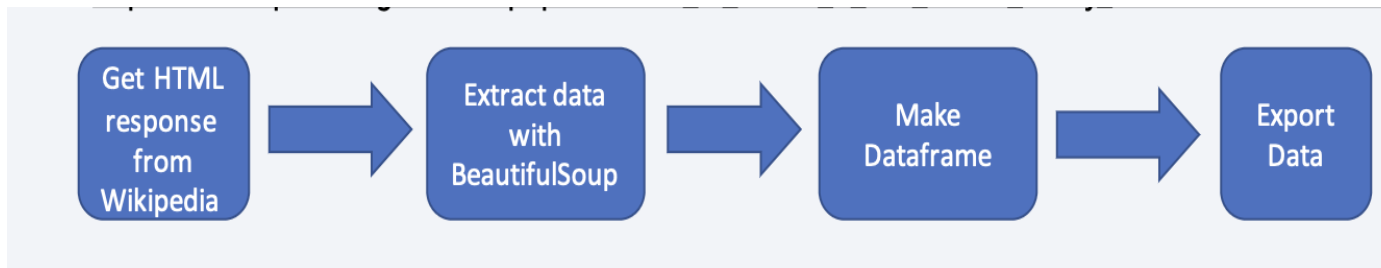
Data was collected through sources:

- SpaceX REST API calls



- Web scraping from Wikipedia:

URL is https://en.wikipedia.org/w/index.php?title=List_of_Falcon_9_and_Falcon_Heavy_launches&oldid=1027686922



Data Collection – SpaceX API

Steps were accomplished:

1. Getting Response from API
2. Convert Response to JSON File
3. Transform data
4. Create dictionary with data
5. Create dataframe
6. Filter dataframe to only Falcon 9
7. Export to file



- GitHub URL:
<https://github.com/asselkey/Capstone-Project-IBM-Assel/blob/main/jupyter-labs-spacex-data-collection-api.ipynb>

Data Collection - Scraping

Web scrap Falcon 9 launch records with BeautifulSoup:

We extracted a Falcon 9 launch records HTML table from Wikipedia
Then, parsed the table and converted it into a Pandas data frame.

Steps that were taken:

1. Getting Response from HTML
2. Create BeautifulSoup Object
3. Find all tables
4. Get column names
5. Create dictionary
6. Add data to keys
7. Create dataframe from dictionary
8. Export to file



GitHub URL:

<https://github.com/asselkey/Capstone-Project-IBM-Assel/blob/main/jupyter-labs-web scraping.ipynb>

Data Wrangling

Objective:

- We have transformed the string variables into categorical variables where 1 means the mission has been successful and 0 means the mission was a failure.

Steps that were taken:

1. Calculated the launches number for each site ---->

CCAFS	SLC	40	55
KSC	LC	39A	22
VAFB	SLC	4E	13

2. Calculated the number and occurrence of each orbit ---->

GTO	27
ISS	21
VLEO	14
PO	9
LEO	7
SSO	5
MEO	3
ES-L1	1
HEO	1
SO	1
GEO	1

True	ASDS	41
None	None	19
True	RTLS	14
False	ASDS	6
True	Ocean	5
False	Ocean	2
None	ASDS	2
False	RTLS	1

3. Calculate number and occurrence of mission outcome per orbit type ---->

4. Create landing outcome label from Outcome column

5. Export to file

GitHub URL:

<https://github.com/asselkey/Capstone-Project-IBM-Assel/blob/main/labs-jupyter-spacex-Data%20wrangling.ipynb>

EDA with Data Visualization

Plots were used for analysis:

Scatter Plots:

- Flight Number vs. Payload Mass
- Flight Number vs. Launch Site
- Payload vs. Launch Site
- Orbit vs. Flight Number
- Payload vs. Orbit Type
- Orbit vs. Payload Mass

Bar Chart:

- Success rate vs. Orbit

Line Graph:

- Success rate vs. Year

GitHub URL:

https://github.com/asselkey/Capstone-Project-IBM-Assel/blob/main/jupyter-labs-EDA_DataVisualization.ipynb

EDA with SQL

1. Display the names of unique launch sites in the space mission
2. Display 5 records where launch sites begin with string 'CCA'
3. Display total payload mass carried by boosters launched by NASA (CRS)
4. Display average payload mass carried by booster version F9 v1.1
5. List the date when the first successful landing outcome in ground pad achieved.
6. List names of boosters which have success in drone ship and have payload mass between 40 and 6000
7. List total number of successful and failure mission outcomes
8. List names of the booster_versions which have carried maximum payload mass.
9. List records which will display month names, failure landing_outcomes in drone ship ,booster versions, launch_site for the months in year 2015.
10. Rank the count of landing outcomes (such as Failure or Success) between the date 2010-06-04 and 2017-03-20.

	LANDING_OUTCOME	COUNT
0	No attempt	10
1	Failure (drone ship)	5
2	Success (drone ship)	5
3	Controlled (ocean)	3
4	Success (ground pad)	3
5	Failure (parachute)	2
6	Uncontrolled (ocean)	2
7	Precluded (drone ship)	1

GitHub URL:
https://github.com/asselkey/Capstone-Project-IBM-Assel/blob/main/jupyter-labs-eda-sql-coursera_sqlite.ipynb

Build an Interactive Map with Folium

Folium map object is a map centered on NASA Johnson Space Center at Houston, Texas.

map objects created and added to a folium map:

1. `Folium.Circle` was used to add a highlighted circle area with a text label on a specific coordinate.
2. `folium.Circle` and `folium.Marker` for each launch site on the site map in order to generate map with marked launch sites
3. Created markers for all launch records. If a launch was successful (`class=1`), then we use a `green marker` and if a launch was failed, we use `a red marker` (`class=0`). For this, `MarkerCluster` object was created and `folium.Marker` was added.
4. `folium.Marker` was created to show the distance between the coastline point and the launch site.
5. We draw a `PolyLine` between a launch site to the selected coastline point
6. We draw a line between a launch site to its closest city, railway, and highway.

These objects help to show all launch sites, their surroundings and the number of successful and unsuccessful landings as well as better understanding the data.

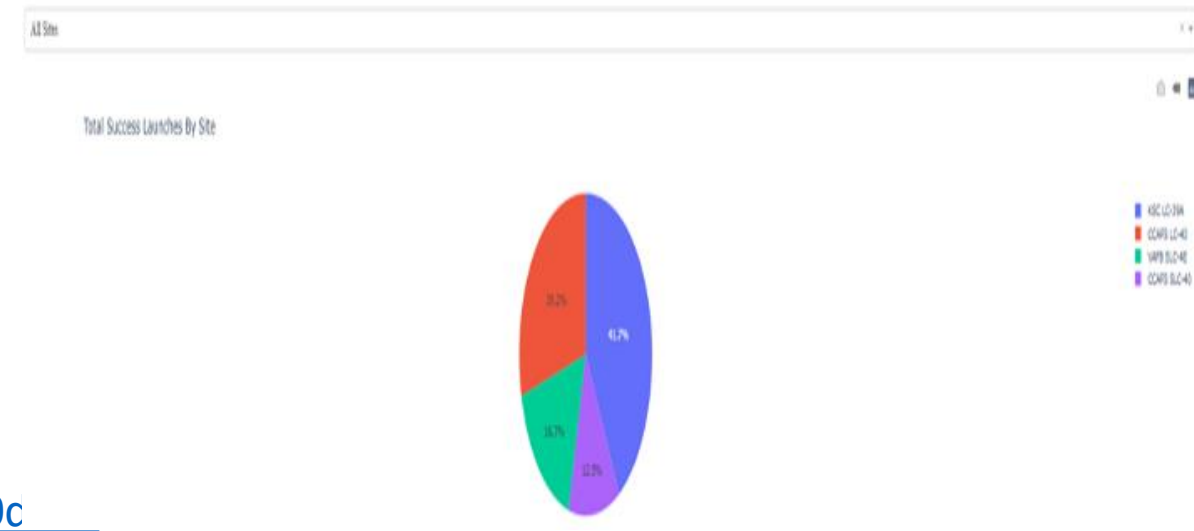
GitHub URL:

<https://github.com/asselkey/Capstone-Project-IBM-Assel/blob/main/labs-jupyter-folium.ipynb>

Build a Dashboard with Plotly Dash

plots/graphs and interactions added to a dashboard:

1. **Dropdown** - helps user to choose the launch sites
2. **Pie Chart** - show total success/failure for the chosen launch site
3. **Rangeslider** - helps to select a payload mass in a fixed range
4. **Scatter Plot** - shows the relationship between 2 variables



GitHub URL:

<https://github.com/asselkey/Capstone-Project-IBM-Assel/blob/main/dashboard%20application%20with%20plotly%20c>

Predictive Analysis (Classification)

Data preparation

- Normalize data
- Split data into training and test sets.

Model preparation

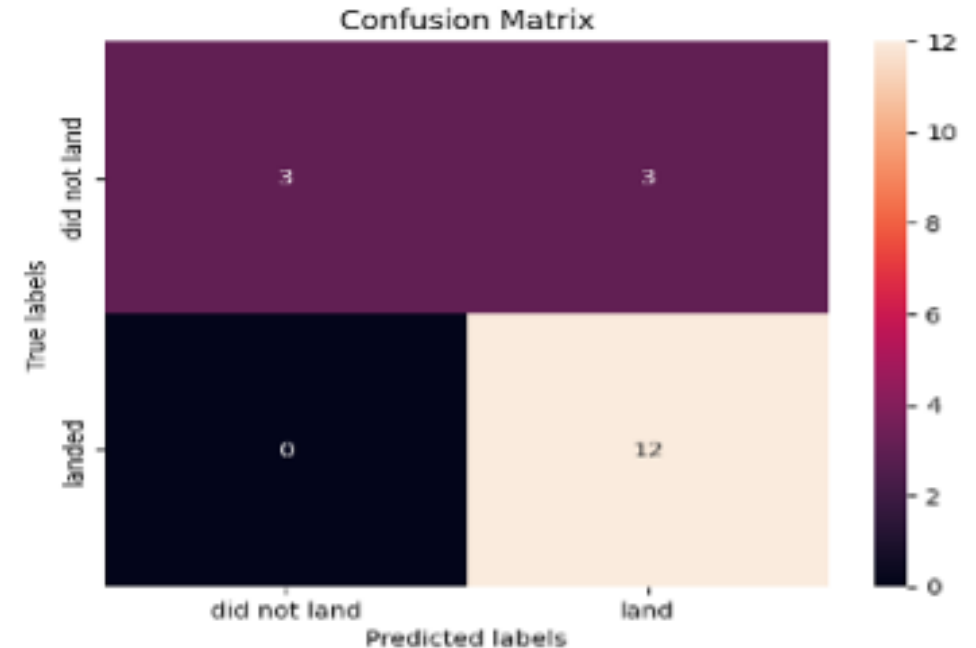
- Selection of machine learning algorithms
- Set parameters for each algorithm to GridSearchCV
- Training GridSearchModel models with training dataset

Model evaluation

- Get the best hyperparameters for each type of model
- Compute accuracy for each model with test dataset
- Plot Confusion Matrix

Model comparison

- Comparison of models by their accuracy
- The model with the best accuracy will be chosen (they all were equal)



GitHub URL:

<https://github.com/asselkey/Capstone-Project-IBM-Assel/blob/main/jupyter-lab-Machine%20Learning%20Prediction.ipynb>

Results

- Exploratory data analysis results

The site that has the largest successful launches is the KSC LC-39A, with 41.7% of total successful launches.

The site that has the highest launch success rate is also the KSC LC-39A, with 76.9% of success rate.

The highest launch success rate has a payload between 3,000 Kg and 4,000 Kg.

The FT Booster version appears to have the highest launch success rate.

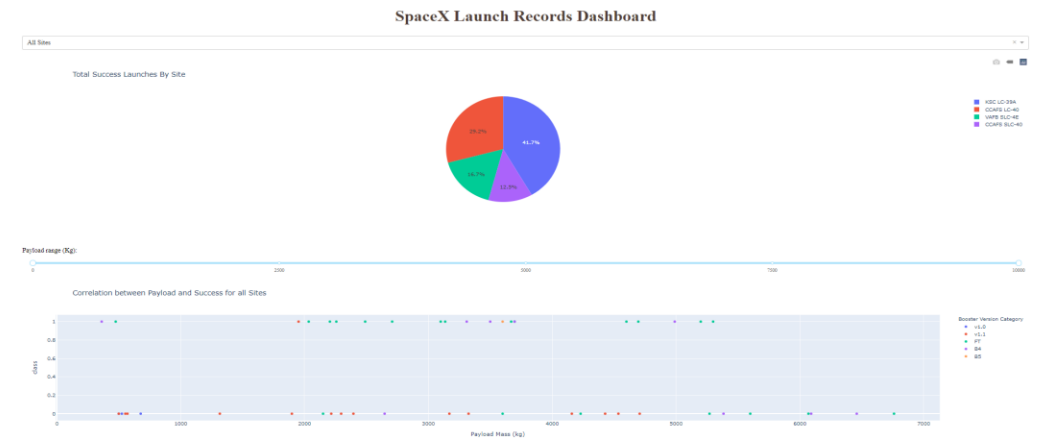
- Interactive analytics demo in screenshots:

- Predictive analysis results:

Logistic Regression, SVM, Decision Tree, and KNN models all achieved an accuracy of 83.33% on the test data.

All models faced a challenge with false positives in their confusion matrices.

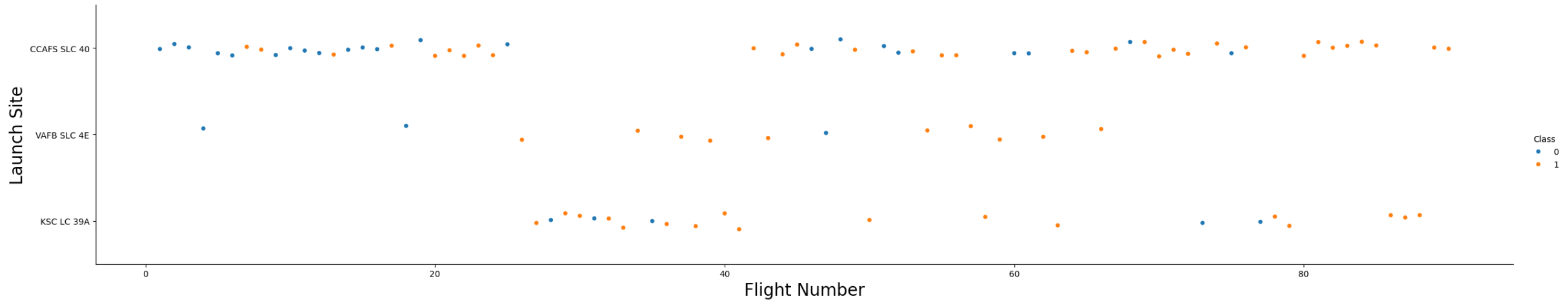
Further analysis and fine-tuning are required to mitigate false positives and enhance overall model performance.



Insights drawn
from EDA:



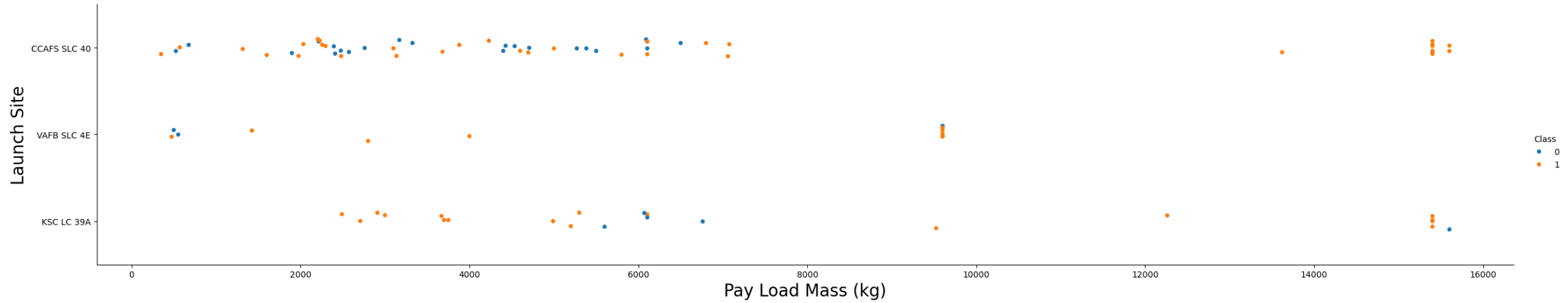
Flight Number vs. Launch Site



Explanation:

- No strong relationship between flight number and success rate. Success appears consistent across various flight numbers.
- no strong correlation between flight number and success rate. Success rates vary across flight numbers.
- Success rates appear consistent, regardless of flight number.

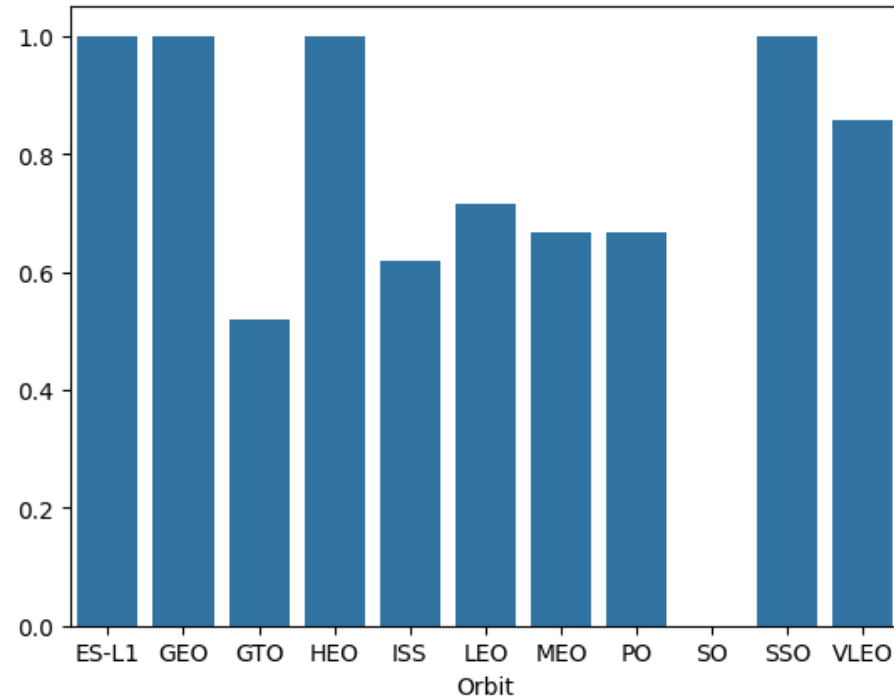
Payload vs. Launch Site



Explanation:

- No rockets with a hefty payload mass of more than 10,000 were launched.
- rockets fired with varying payload masses.
- Payload handling and capacity are impacted by the launch site.

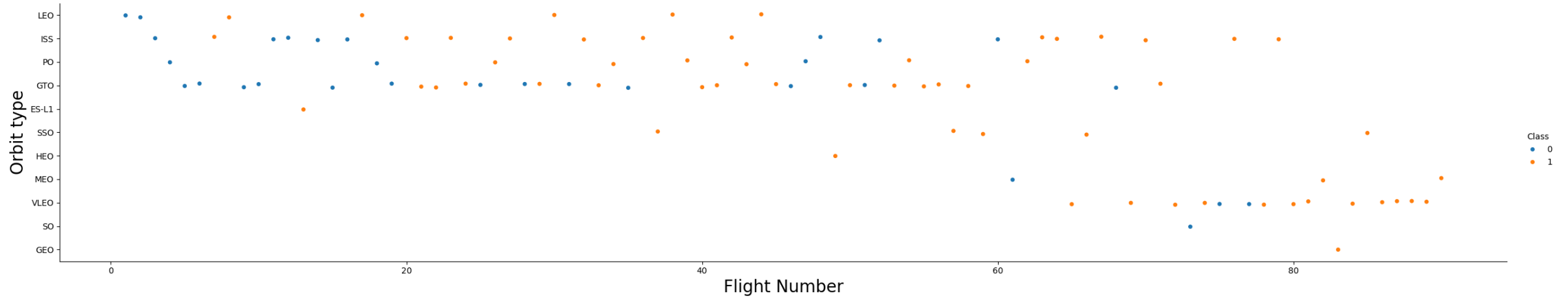
Success Rate vs. Orbit Type



Explanation:

- SSO, HEO, GEO, ES-L1 Orbits: 100% success rates, indicating high reliability.
- Other Orbit Types: Varying success rates, suggesting mission complexities.

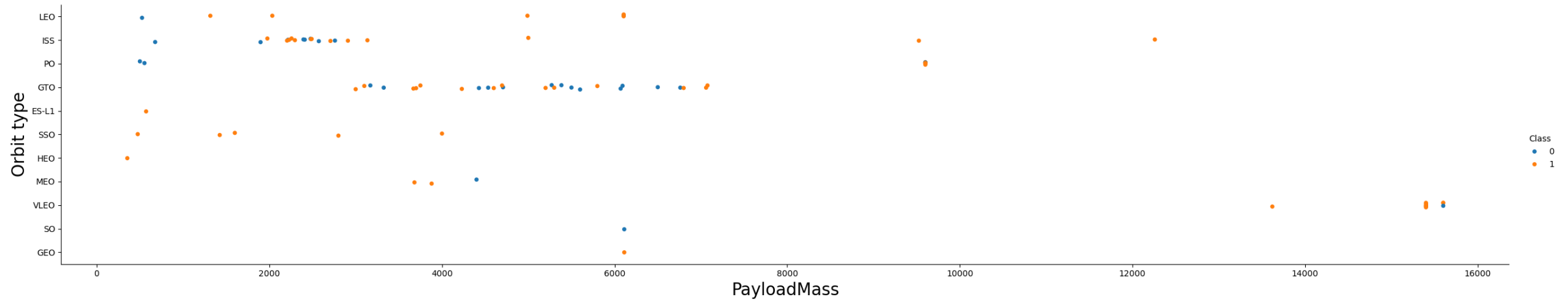
Flight Number vs. Orbit Type



Explanation:

- LEO Orbit: Success appears correlated with the number of flights.
- GTO Orbit: No clear relationship between flight number and success rate.

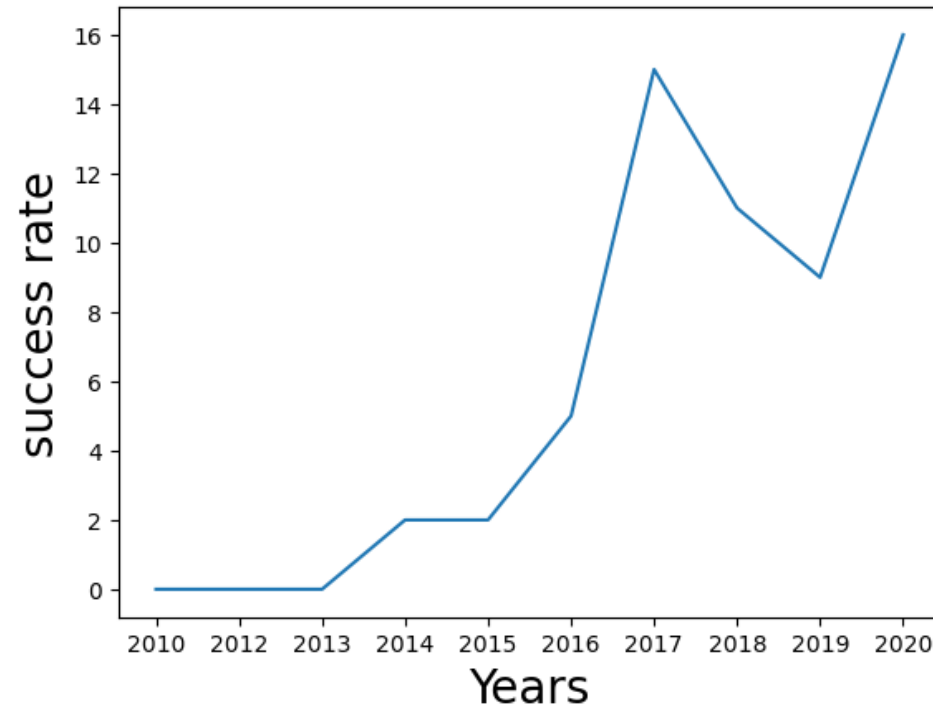
Payload vs. Orbit Type



Explanation:

- PO, LEO, ISS Orbits: Higher payload mass corresponds to a more successful landing.
- GTO Orbit: Payload mass does not clearly affect landing success.

Launch Success Yearly Trend



Explanation:

- 2013 to 2020: Consistent increase in success rates.
- SpaceX's continual improvement in launch reliability.

All Launch Site Names

```
%sql select distinct Launch_Site from SPACEXTBL
```



	LAUNCH_SITE
0	CCAFS LC-40
1	CCAFS SLC-40
2	KSC LC-39A
3	VAFB SLC-4E

Explanation: DISTINCT command helps to remove duplicates of launch site names.

Launch Site Names Begin with 'CCA'

```
%sql select * from SPACEXTBL where Launch_Site LIKE 'CCA%' limit 5
```



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

Explanation: WHERE and LIKE command help to show only launch sites that contain "CCA". LIMIT 5 helps to limit the dataset to 5 rows.

Total Payload Mass

```
%sql select SUM(PAYLOAD_MASS__KG_) from SPACEXTBL where Customer='NASA (CRS)'
```



SUM(PAYLOAD_MASS__KG_)
45596

Explanation: SUM function helps to return the sum of all Payload Mass data where the customer is NASA (CRS).

Average Payload Mass by F9 v1.1

```
%sql select AVG(PAYLOAD_MASS__KG_) from SPACEXTBL where Booster_Version = 'F9 v1.1'
```



AVG(PAYLOAD_MASS__KG_)
2928.4

Explanation: AVG function helps to return the average value of the payload masses where the Booster Version contains 'F9 v1.1' value.

First Successful Ground Landing Date

```
%sql select MIN(Date) from SPACEXTBL where Landing_Outcome = 'Success (ground pad)'
```

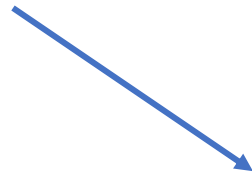


MIN(Date)
2015-12-22

Explanation: MIN(Date) helps to return the first date when the Landing Outcome was successful. The output: 2015, 12, 22.

Successful Drone Ship Landing with Payload between 4000 and 6000

```
%sql select distinct Booster_Version from SPACEXTBL where Landing_Outcome = 'Success (drone ship)' and  
PAYLOAD_MASS__KG_ between 4000 and 6000;
```

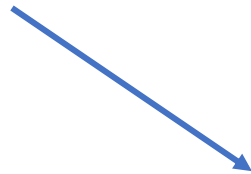


Booster_Version
F9 FT B1022
F9 FT B1026
F9 FT B1021.2
F9 FT B1031.2

Explanation: The query returns the list of Booster Version where the Landing Outcome was successful and Payload mass was between 4000 and 6000 kg.

Total Number of Successful and Failure Mission Outcomes

```
%sql select substr(Mission_Outcome,1,7) as Mission_Outcome, count(*) from SPACEXTBL group by 1
```



Mission_Outcome	count(*)
Failure	1
Success	100

Explanation: The query returns the total number of successful and failure mission outcomes showing the result of 1 Failure and 100 Success count.

Boosters Carried Maximum Payload

```
%sql select distinct Booster_Version from SPACEXTBL where  
PAYLOAD_MASS__KG_ = (select MAX(PAYLOAD_MASS__KG_) from  
SPACEXTBL)
```



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

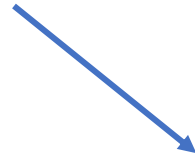
F9 B5 B1060.3

F9 B5 B1049.7

Explanation: The result shows the list with the names of booster_versions which have carried the maximum payload mass with the use of subquery.

2015 Launch Records

```
%sql select substr(Date, 6,2) as Month, Landing_Outcome, Booster_Version, Launch_Site from SPACEXTBL where  
Landing_Outcome = 'Failure (drone ship)' and substr(Date,0,5)='2015'
```




Month	Landing_Outcome	Booster_Version	Launch_Site
01	Failure (drone ship)	F9 v1.1 B1012	CCAFS LC-40
04	Failure (drone ship)	F9 v1.1 B1015	CCAFS LC-40

Explanation: The result shows the list of the records which display the month names, failure landing_outcomes in drone ship ,booster versions, launch_site for the months in year 2015.

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

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



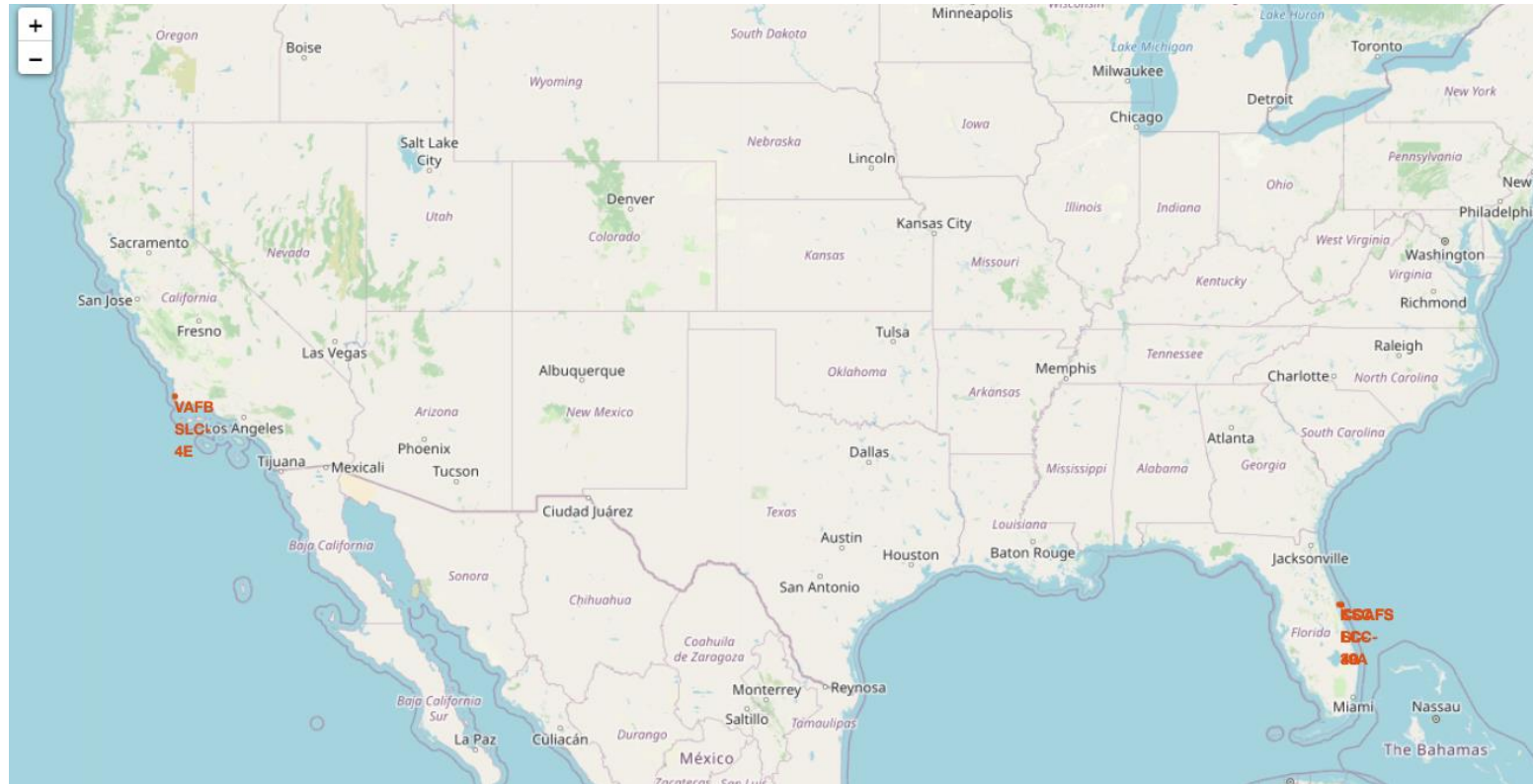
Landing_Outcome	Count
No attempt	10
Success (drone ship)	5
Failure (drone ship)	5
Success (ground pad)	3
Controlled (ocean)	3
Uncontrolled (ocean)	2
Failure (parachute)	2
Precluded (drone ship)	1

Explanation: The result shows the rank of 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.

Launch Sites Proximities Analysis

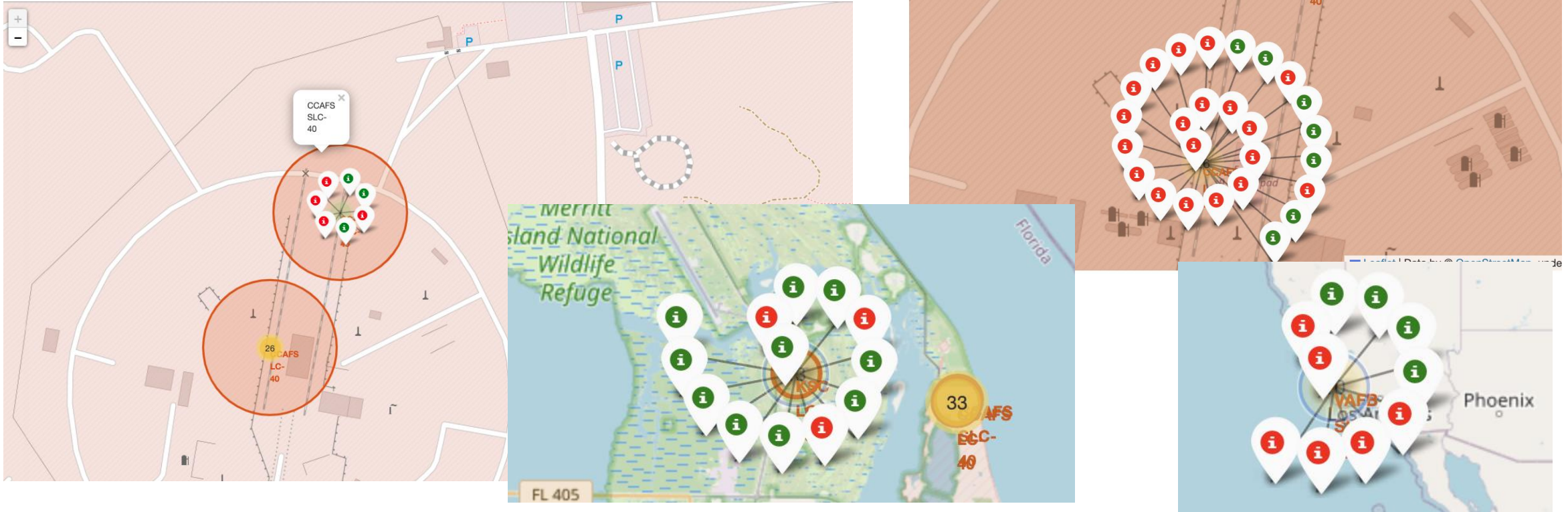


Folium Map With Marked Launch Sites



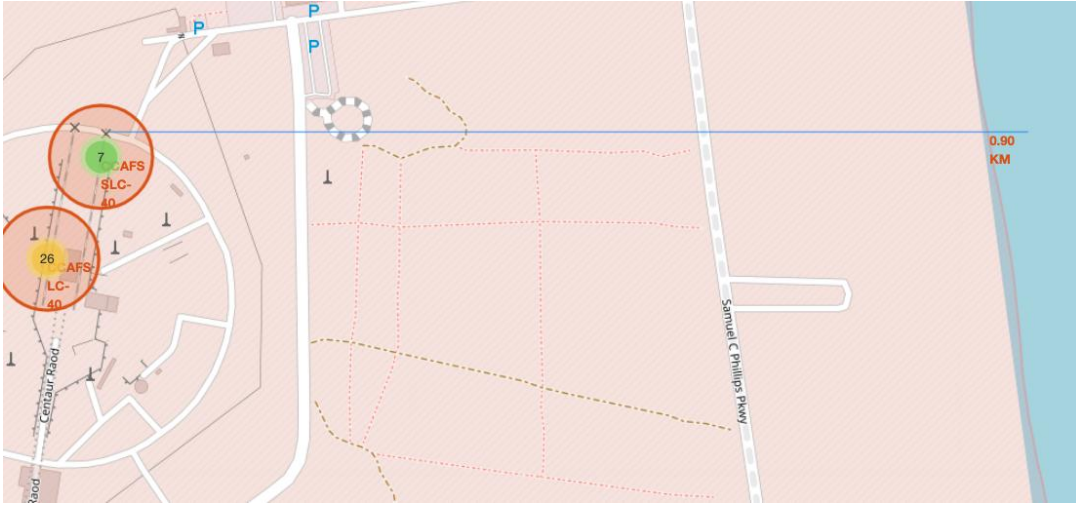
Explanation: It can be seen that SpaceX launch sites are located on the coastal area of the US, and are in proximity to the Equator line.

Folium Map Color Labeled Markers



Explanation: There are two markers in each launch site which represent: green color – successful landing outcome while red color – unsuccessful. From this, KSC LC-39A has a higher success rate than others.

Folium Map Proximity



Explanation: According to these maps, all 3 railway, highway and coastline are located in a close proximity to the launch sites.

Launch sites are strategically located near coastlines for safety, avoiding overflight of populated areas, and enabling water landing in case of an abort.

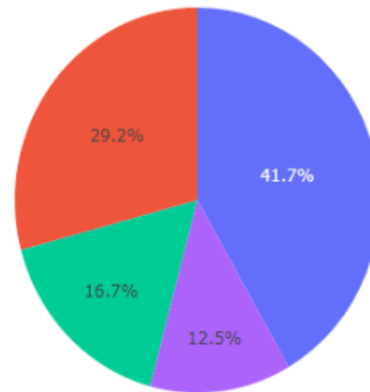
Launch sites are intentionally kept at a distance from cities to reduce the risk to densely populated areas, resulting in remote locations with limited infrastructure.

Build a Dashboard with Plotly Dash



Total Success Launches by Site

Total Success Launches by Site

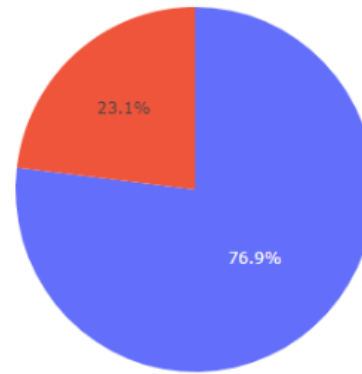


■ KSC LC-39A
■ CCAFS LC-40
■ VAFB SLC-4E
■ CCAFS SLC-40

Explanation: Here is the pie chart that represents launch sites and their success rates in percentage. It can be noted that KSC LC-39A has a higher success rate than other launch sites.

Total Success Launches for KSC LC-39A

Total Success Launches for Site KSC LC-39A



1
0

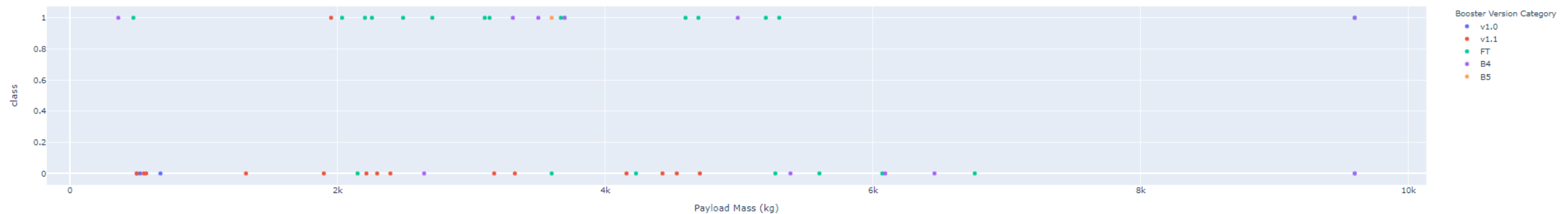
Explanation: The site that has the largest successful launches is the KSC LC-39A, with 41.7% of total successful launches and it has 76.9% of success rate.

Correlation between Payload and Success for all Sites

Payload range (Kg):



Correlation between Payload and Success for all Sites



Explanation: The highest launch success rate has a payload between 3,000 Kg and 4,000 Kg. The lowest launch success rate has a payload between 6,000 Kg and 8,000 Kg. The FT Booster version appears to have the highest launch success rate.

Predictive Analysis (Classification)



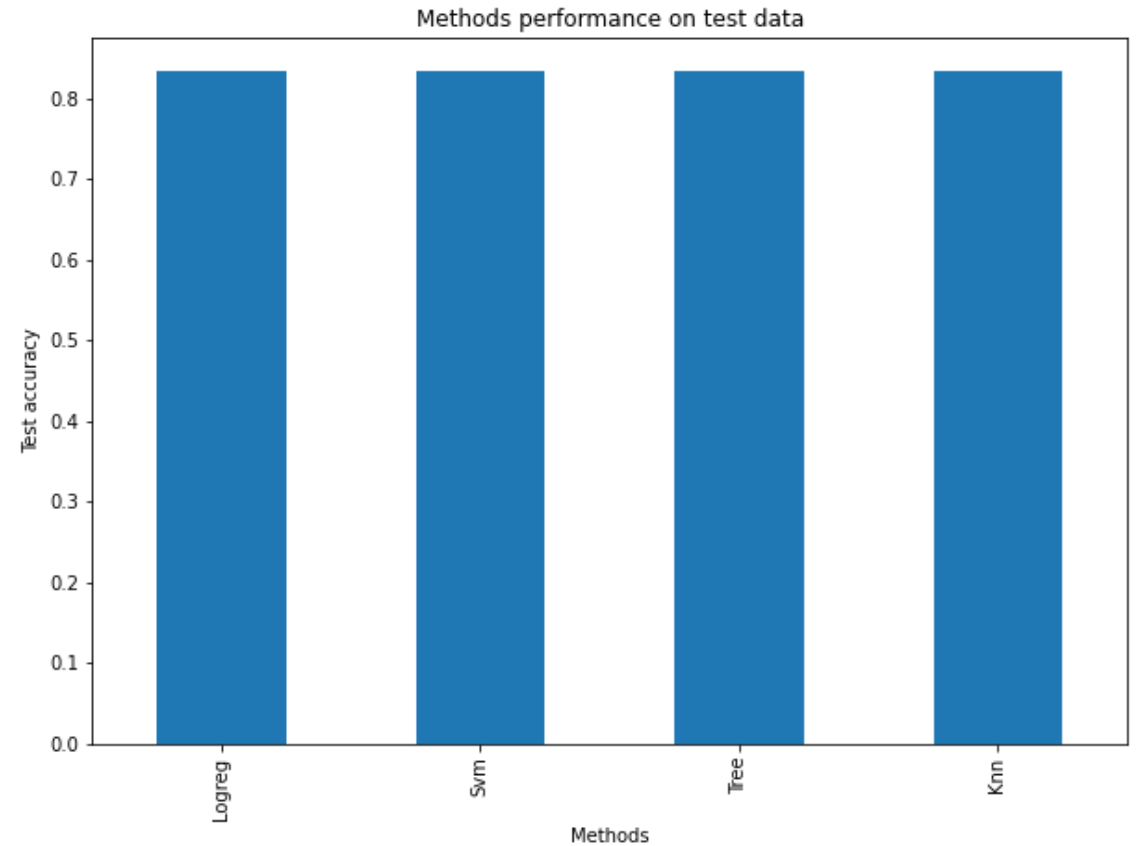
Classification Accuracy

LR Accuracy: 83.33%

SVM Accuracy: 83.33%

Decision Tree Accuracy: 83.33%

KNN Accuracy: 83.33%

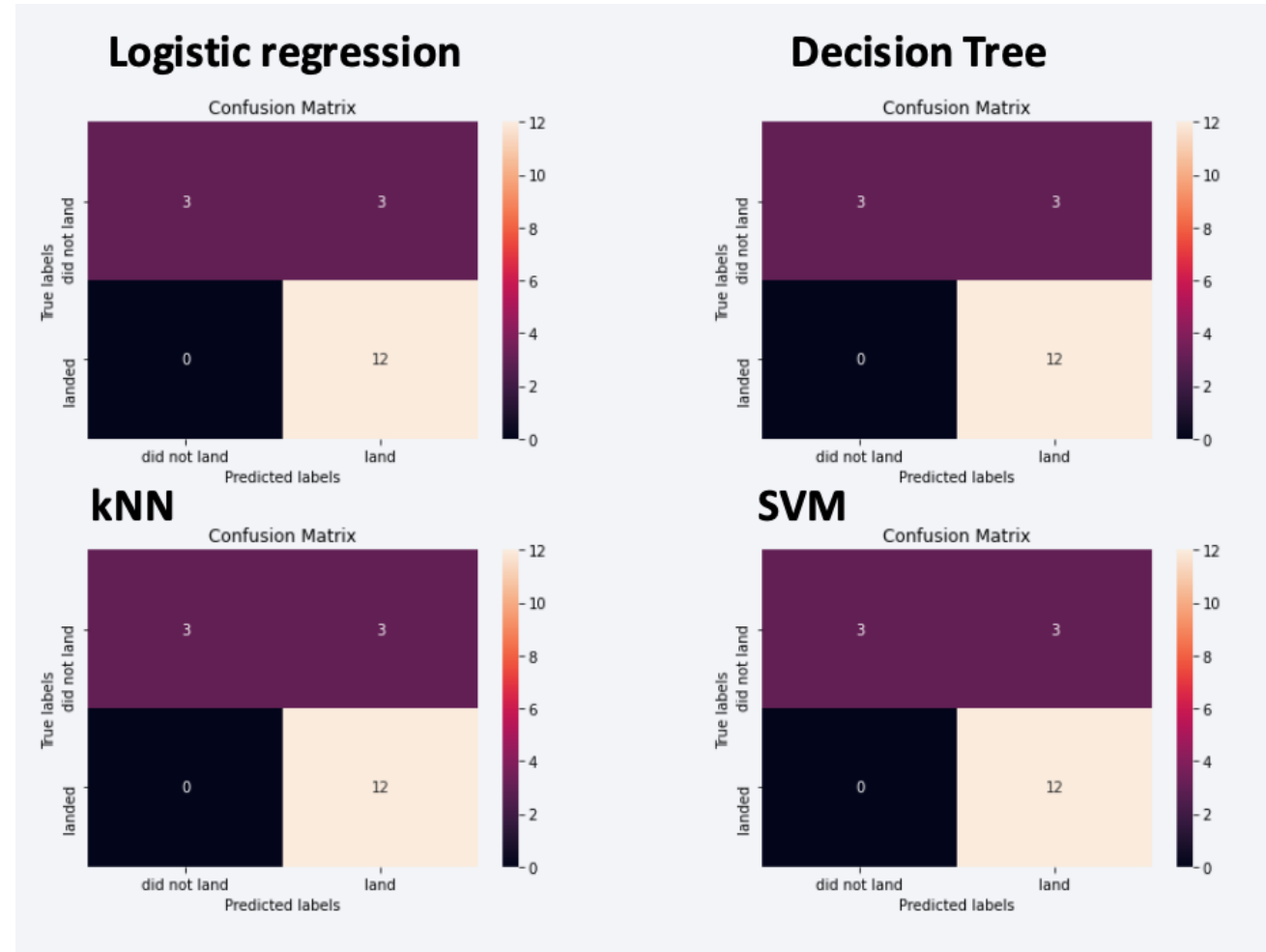


Logistic Regression, SVM, Decision Tree, and KNN models
all achieved an accuracy of 83.33% on the test data.

Confusion Matrix

Since the classification accuracy showed the equal accuracy rates in all 4 models, the confusion matrix also showed the same results.

All models faced a challenge with false positives in their confusion matrices.



Conclusions

- A mission's success can be attributed to a number of variables, including the **launch site, the orbit, and most notably, the quantity of prior launches**. In fact, we might presume that knowledge gained in the interim between launches is what made it possible to turn a failed launch into a successful one.
- The most successful **orbits** are ES-L1, GEO, HEO, SSO, and GEO.
- Depending on the orbits, the **payloads mass** may be a factor to consider when determining if a mission is successful. Certain orbits demand a large or small payload mass. However, low-weighted payloads outperform heavy-weighted payloads in most cases.
- Based on available data, it is not possible to explain why some launch sites perform better than others (**KS CLC-39A is the best launch site**). We might gather pertinent data on the atmosphere or other topics to find a solution to this issue.
- We select the **Decision Tree Algorithm** as the best model for this dataset, even in cases where the test accuracy of every model is the same. Because the Decision Tree Algorithm offers a higher train accuracy, we went with it.

Appendix

```
import requests
import pandas as pd
import numpy as np
import datetime

spacex_url = https://api.spacexdata.com/v4/launches/past
response = requests.get(spacex_url)
data = pd.json_normalize(response.json())
data.head(2)
```

```
%sql select distinct Launch_Site from SPACEXTBL
```

```
%sql select AVG(PAYLOAD_MASS__KG_) from SPACEXTBL
where Booster_Version = 'F9 v1.1'
```

Thank you!