IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Skyler
Sep 2022

Outline

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

Executive Summary

- Data was collected from the open source SpaceX API and by web scraping Wikipedia's list on Falcon 9 and Falcon Heavy launches
- EDA and data wrangling was performed on the extracted data to clean and prepare it to train machine learning models. Data was normalized and split into train and test sets.
- We found that launches with lower weights had a higher chance of success and later launches had higher success rates.
- Multiple types of regression models were trained on the data (Logistic Regression, SVM, Decision Tree, KNN) and hyperparameters were tuned by using GridSearchCV
- All machine learning models performed similarly but Decision Tree model saw poor performance. All predicted successful launch outcomes very well, however the models suffered from false positives.

Introduction

- We would like to determine if the Falcon 9's first stage will land successfully through a data science approach
- Doing this successfully will allow us to determine the cost saving achieved from the success rate of reusing boosters, potentially allowing us to bid against SpaceX as a competitor for launching rockets

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - Data was collected through the SpaceX API, and web scraping a Wikipedia page
- Perform data wrangling
  - Data was processed using Pandas, taking care of Null values and one-hot encoding relevant data features
- 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 was split into train & test sets. Each classification model was trained and best parameters were found via GridSearchCV.

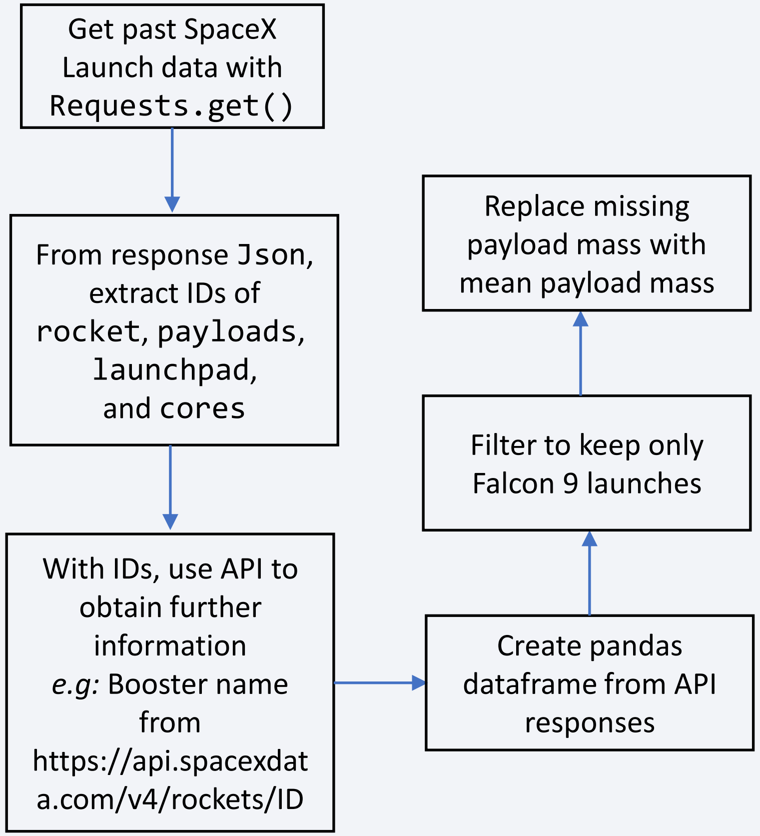
Data Collection

- Data was collected from 2 sources:
  - Open source SpaceX API – https://github.com/r-spacex/SpaceX-API
  - Web scraping launch data from a list of SpaceX launches on Wikipedia – https://en.wikipedia.org/wiki/List_of_Falcon_9_and_Falcon_Heavy_launches
- The next 2 slides will go into detail on collection of data from these 2 sources

Data Collection – SpaceX API

- Data was collected from the SpaceX API using the process as shown on the right →

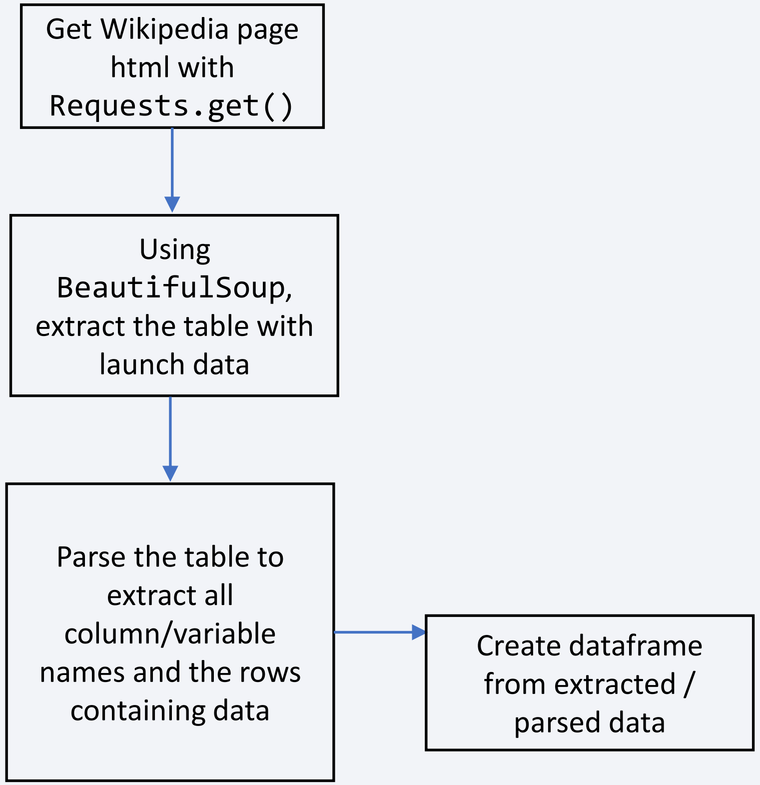
- https://github.com/skulu/IBM-Coursera-DS-Capstone-Project/blob/main/1%20Data%20Collection.ipynb



Data Collection - Scraping

- Data was scraped from Wikipedia using the process as shown on the right →

- https://github.com/skulu/IBM-Coursera-DS-Capstone-Project/blob/main/2%20Web%20Scraping.ipynb

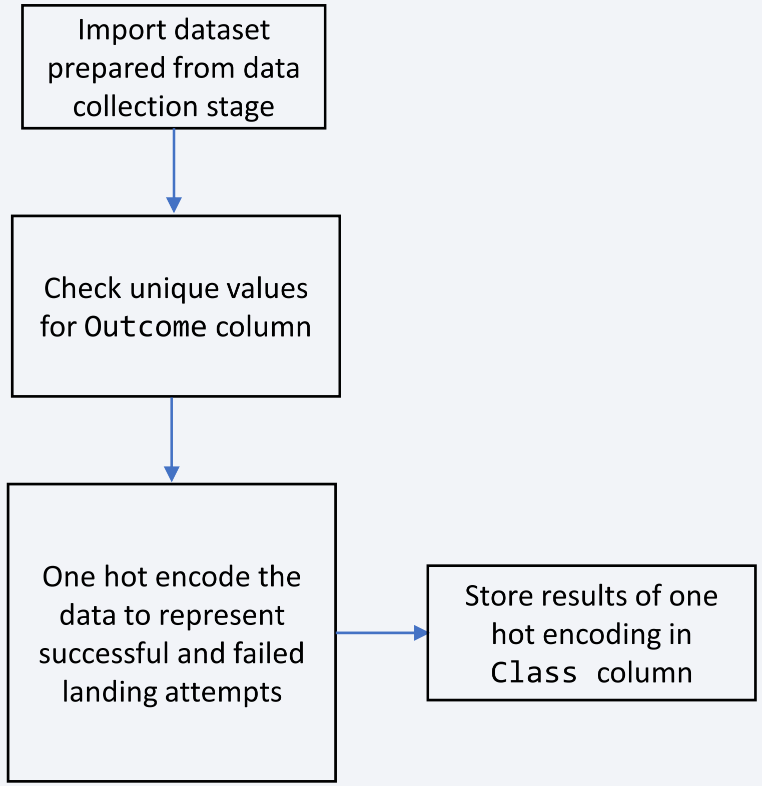


Data Wrangling

- During this phase, some initial data exploration was conducted:
  - Find out number of null values
  - Find out column data types
  - Check number of launches at each launch site
  - Check number of launches for each orbit type

- The landing outcomes column was encoded to 1 and 0 to prepare for data visualization and predictive analytics

- https://github.com/skulu/IBM-Coursera-DS-Capstone-Project/blob/main/3%20Data%20Wrangling.ipynb



EDA with Data Visualization

- Scatter plots, bar charts and a line chart were used.
- The scatter plots had points color coded to indicate landing outcomes to explore how `FlightNumber, PayloadMass, LaunchSite, Orbit` affected success probability.
- A Bar chart was used to check the effect on success rate based on different orbits
- Finally a line chart was used to visualize the average yearly probability of success as the years went by
- https://github.com/skulu/IBM-Coursera-DS-Capstone-Project/blob/main/5%20jupyter-labs-eda-dataviz.ipynb

EDA with SQL

- SQL queries were used to find out certain statistics such as:
  - Total payload mass launched by SpaceX for NASA (CRS)
  - Average payload mass carried by F9 V1.1
  - Total successful and failed mission outcomes
  - Boosters that carried the maximum payload mass
  - Finding out count of successful landing_outcomes between 04-06-2010 and 20-03-2017
- https://github.com/skulu/IBM-Coursera-DS-Capstone-Project/blob/main/4%20eda-sql-coursera.ipynb

Build an Interactive Map with Folium

- Circles were added to indicate the launch site locations, with a marker used with text to label the sites
- A MarkerCluster object was then used to mark the launch outcomes at each launch site
- To indicate distance to the coast, a PolyLine was drawn and the distance labelled on the map with a marker object
- The objects above were added to the map to visualize the location of the launch sites, the outcomes of the launch attempts at the sites and proximity to the coast.
- https://github.com/skulu/IBM-Coursera-DS-Capstone-Project/blob/main/6%20lab_jupyter_launch_site_location.ipynb

Build a Dashboard with Plotly Dash

- A drop down was added for users to filter the pie chart and the scatter chart in the dashboard by launch sites
- A range slider was added for users to filter the scatter chart by payload mass range
- The pie chart shows the proportion of successful launches by site when “All Sites” is selected in the drop down. When a specific site is selected, it changes to show the breakdown of success and failures for that site.
- The scatter plot shows the correlation between payload mass and success for the selected site in the drop down box.
- The dashboard is a good way to explore the relationship between site, payload mass and success / failure of launches.
- https://github.com/skulu/IBM-Coursera-DS-Capstone-Project/blob/main/7%20spacex_dash_app.py

Predictive Analysis (Classification)

- Data processed from previous steps were loaded and standardized to 0 mean and unit variance
- The features matrix X and target vector Y was obtained from the data and split into train test sets
- Logistic regression, support vector machine, decision tree, k nearest neighbours models were trained and parameters were optimsed with GridSearchCV
- Based on the score and confusion matrices, all models performed similarly except for the decision tree model which performed worse than the rest.
- https://github.com/skulu/IBM-Coursera-DS-Capstone-Project/blob/main/8%20SpaceX_Machine%20Learning%20Prediction_Part_5.ipynb

Results

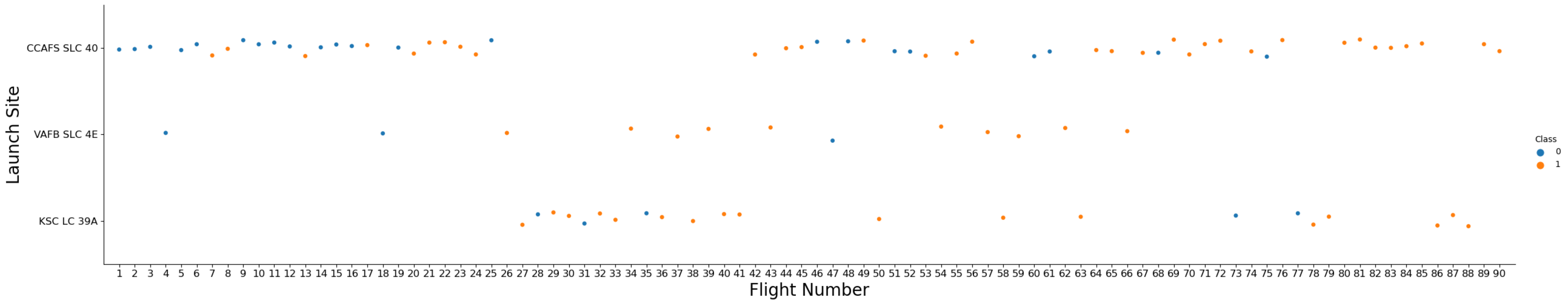
- Exploratory data analysis results
- Interactive analytics demo in screenshots
- Predictive analysis results

Section 2

Insights drawn from EDA

Flight Number vs. Launch Site

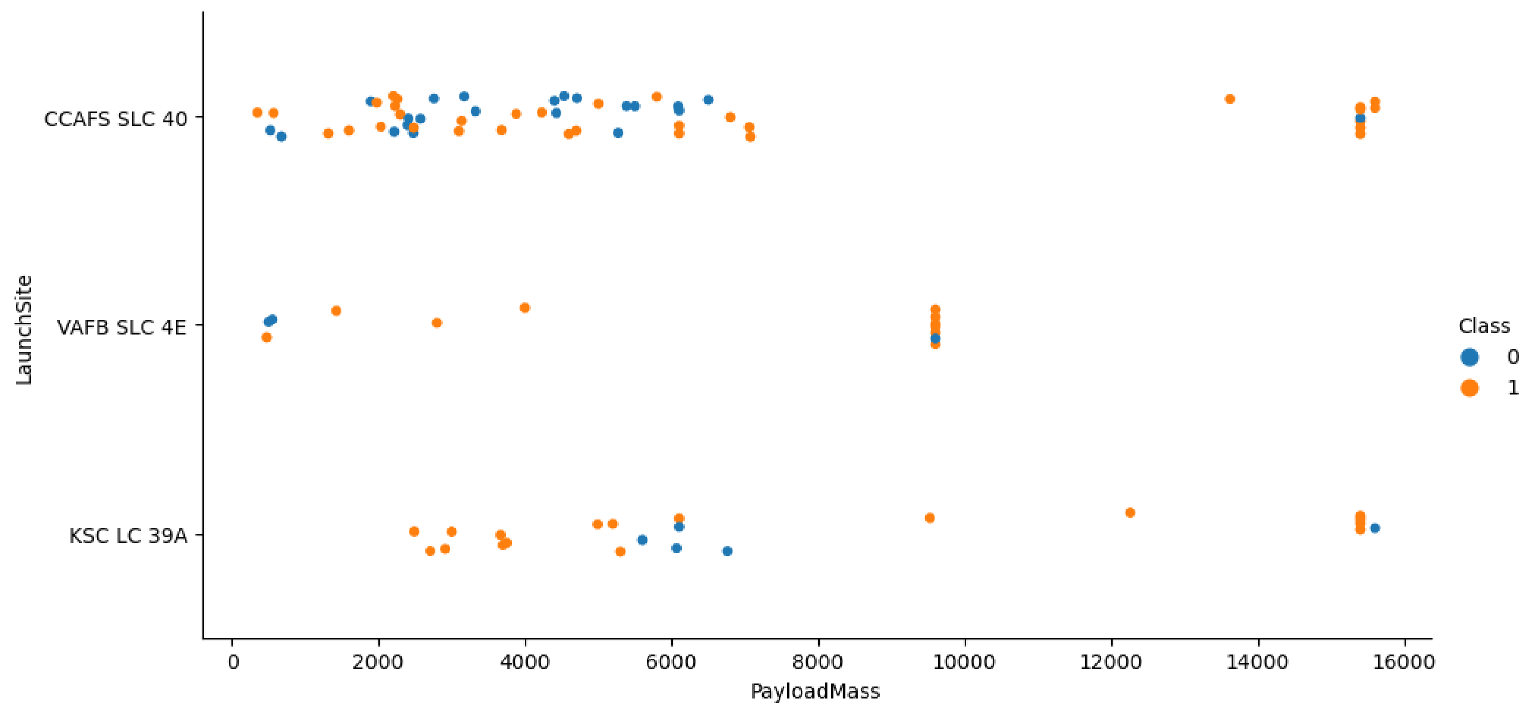
- Scatter plot of Flight Number vs. Launch Site



- As the flight number increased, we can see that the probability of success increased. Probability of success did not seem to be affected by launch site choice after flight 27.

- CCAFS SLC 40 launch site was used almost exclusively at the start followed by a period between launches 27 and 42 where KSC LC 39A was primarily used

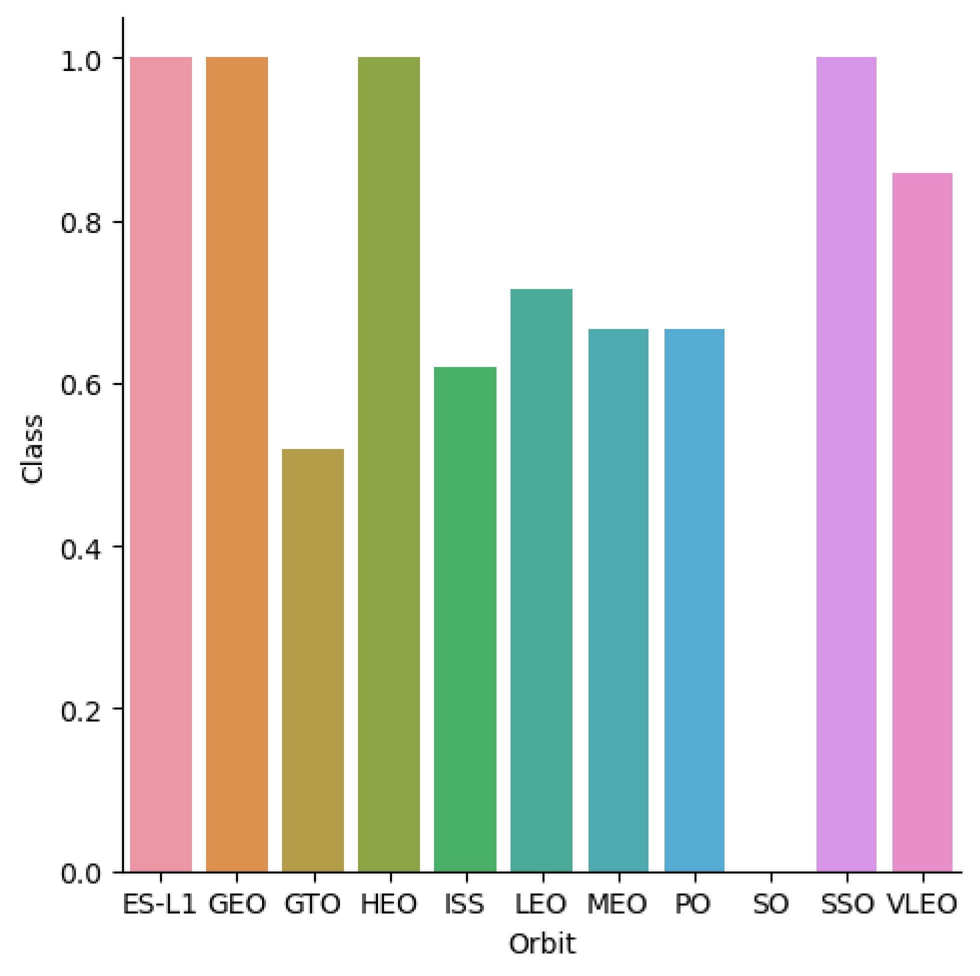
Payload vs. Launch Site



- Most payloads were below 8,000 kg. For VAFB SLC 4E, no payloads above 10,000kg was launched.
- No clear trend for probability of success vs payload mass or launch site

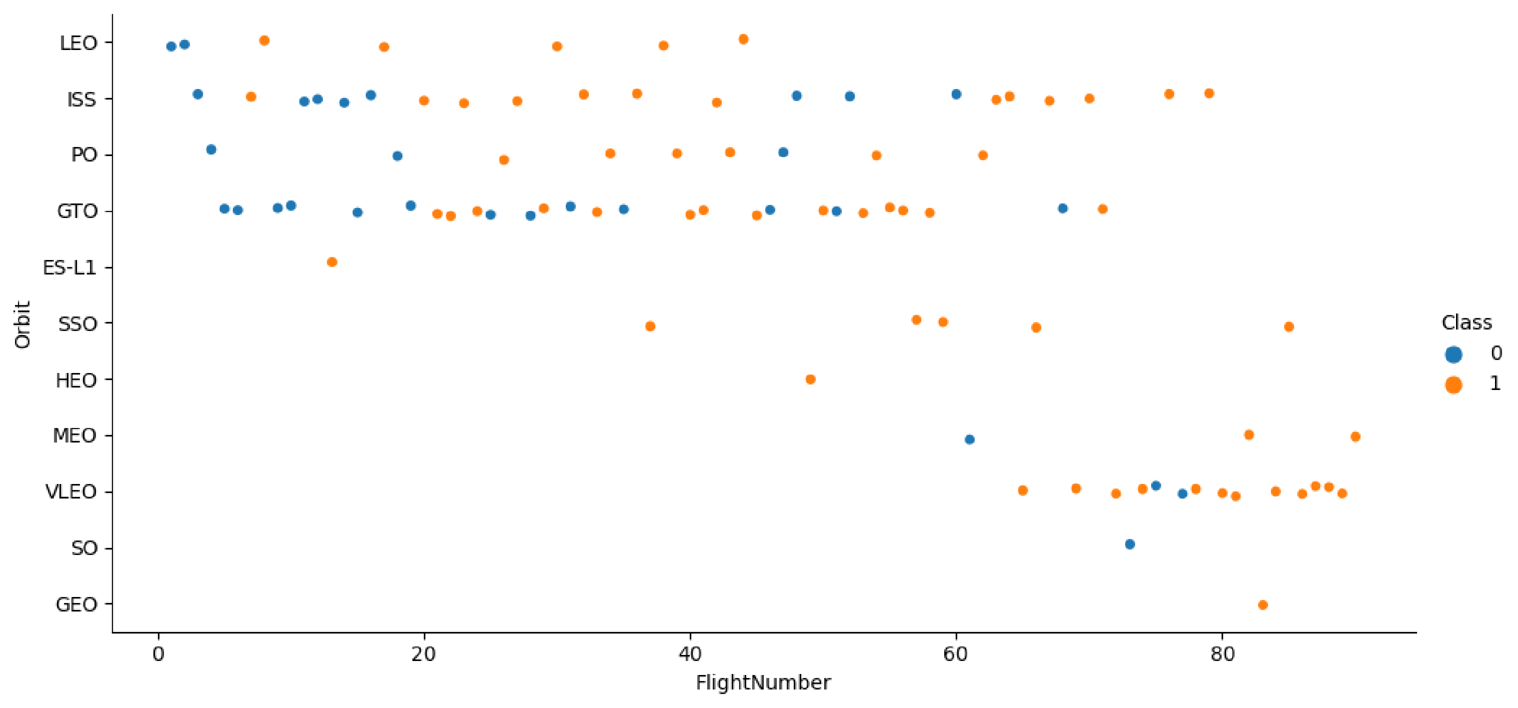
Success Rate vs. Orbit Type

- **GEO, SO, HEO, ES-L1, MEO, SSO** had low sample sizes (5 or less launches), so it is difficult to draw conclusions.
- **GTO**, or geosynchronous orbit had 27 launches and a success rate of ~0.52. It had the lowest success rate after SO. This may be because it is a high Earth orbit that takes more fuel to get to, leaving insufficient fuel for a successful landing.



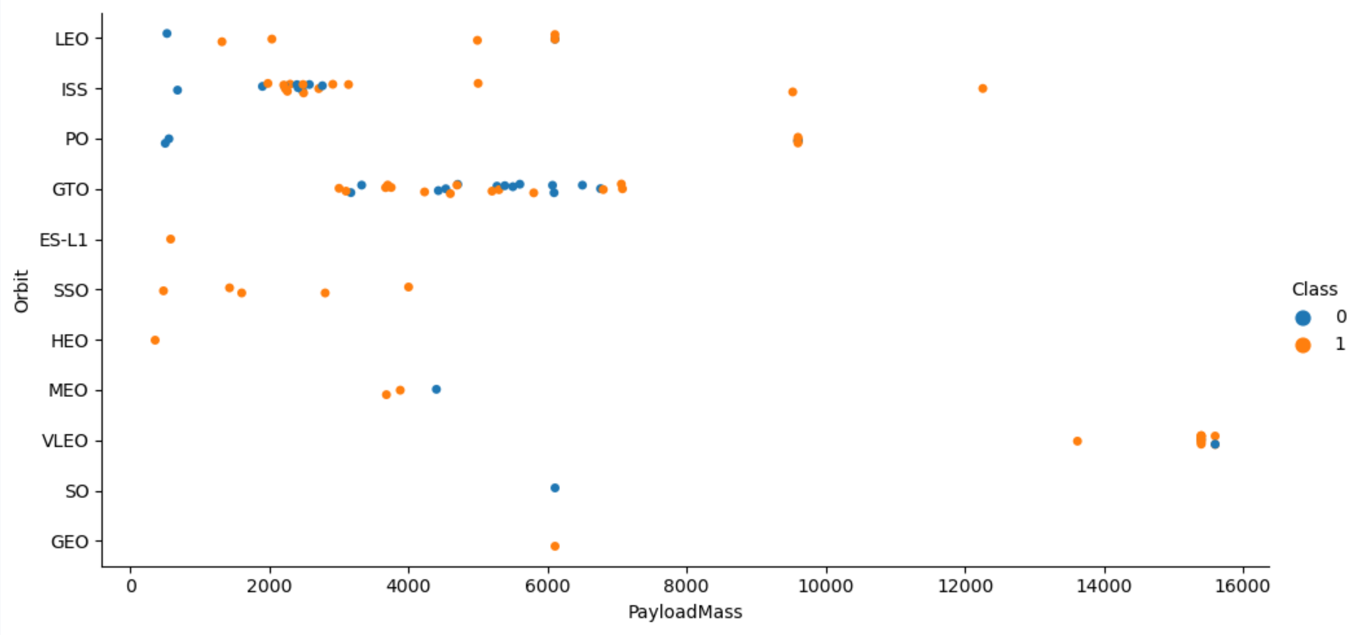
Flight Number vs. Orbit Type

- For LEO orbit, the Success appears related to the number of flights; on the other hand, there seems to be no relationship between flight number when in GTO orbit.



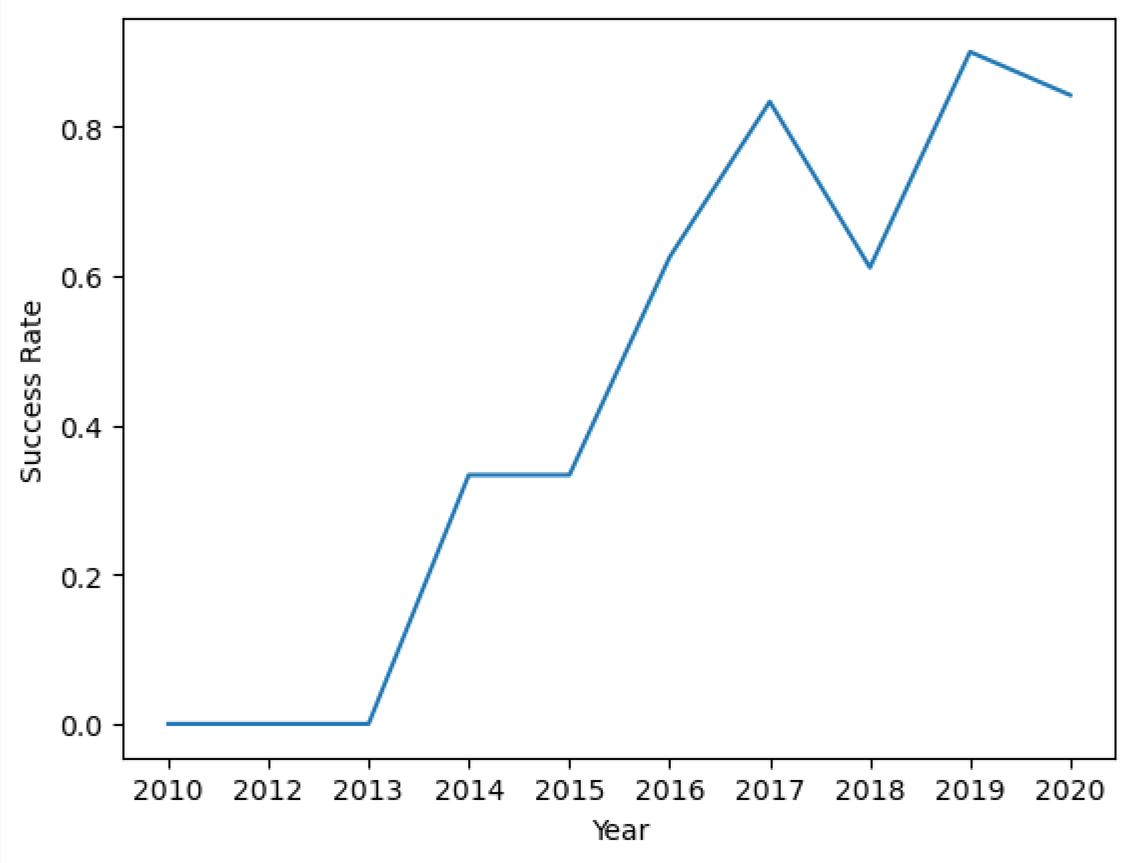
Payload vs. Orbit Type

- VLEO had the heaviest payloads, likely because it is the lowest orbit and hence there is capacity to carry heavier loads to this orbit
- With heavy payloads the successful landing or positive landing rate are more for Polar, LEO and ISS.



Launch Success Yearly Trend

- We can see launch success rate improving over the years
- This is likely due to the continuous improvement in booster reliability as SpaceX learned from previous launches



All Launch Site Names

- SQL Query: `SELECT DISTINCT LAUNCH_SITE FROM SPACEXTBL`

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

- Distinct launch sites were selected from the table

Launch Site Names Begin with 'CCA'

- ```
  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
04-06-2010	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success	Failure (parachute)
08-12-2010	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)
22-05-2012	07:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success	No attempt
08-10-2012	00:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500	LEO (ISS)	NASA (CRS)	Success	No attempt
01-03-2013	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677	LEO (ISS)	NASA (CRS)	Success	No attempt

- 5 records were returned where launch site names began with CCA. % is a wildcard character.
  ```

Total Payload Mass

[9]:
```
%%sql
SELECT SUM(PAYLOAD_MASS__KG_) FROM SPACEXTBL
WHERE CUSTOMER LIKE 'NASA (CRS)'
```

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

[9]:

SUM(PAYLOAD_MASS__KG_)
45596

- The total payload mass was summed up after filtering for NASA (CRS) as the customer

Average Payload Mass by F9 v1.1

[10]:

```
%%sql
SELECT AVG(PAYLOAD_MASS__KG_) FROM SPACEXTBL
WHERE BOOSTER_VERSION LIKE 'F9 v1.1%'
```

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

[10]:

AVG(PAYLOAD_MASS__KG_)
2534.6666666666665

- The payload mass was averaged after filtering for booster version F9 v1.1

First Successful Ground Landing Date

```
[12]:  %%sql
       SELECT MIN(DATE) FROM SPACEXTBL
       WHERE "Landing _Outcome" LIKE 'Success%'

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

[12]:

MIN(DATE)
01-05-2017

- Used the min function on the date filed after filtering for a successful landing outcome

Successful Drone Ship Landing with Payload between 4000 and 6000

[13]:
```
%%sql
SELECT Booster_Version FROM SPACEXTBL
WHERE "Landing _Outcome" LIKE 'Success (drone ship)' AND PAYLOAD_MASS__KG_ > 4000 AND PAYLOAD_MASS__KG_ < 6000
```

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

[13]:

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

- Appropriate filters for payload weight and successful landings on drone ships was applied and the booster_version column containing the names of the booster was returned.

Total Number of Successful and Failure Mission Outcomes

[15]:
```
%%sql
SELECT COUNT(Mission_Outcome) FROM SPACEXTBL
WHERE MISSION_OUTCOME LIKE 'Success%'
```

* sqlite:///my_data1.db
Done.

[15]:

COUNT(Mission_Outcome)
100

[16]:
```
%%sql
SELECT COUNT(Mission_Outcome) FROM SPACEXTBL
WHERE MISSION_OUTCOME LIKE 'Failure%'
```

* sqlite:///my_data1.db
Done.

[16]:

COUNT(Mission_Outcome)
1

- Filtered and counted the number of mission outcomes for both success and failure

Boosters Carried Maximum Payload

```
[17]: %%sql
SELECT Booster_Version FROM SPACEXTBL
WHERE PAYLOAD_MASS__KG_ IN
(SELECT MAX(PAYLOAD_MASS__KG_) FROM SPACEXTBL)

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

[17]:

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

- A subquery selecting for maximum payload mass was used.

2015 Launch Records

```
[18]: %%sql
SELECT substr(Date,4,2) AS MONTH, substr(Date,7,4) AS YEAR, "Landing _Outcome", BOOSTER_VERSION, LAUNCH_SITE FROM SPACEXTBL
WHERE substr(Date,7,4) = '2015' AND "Landing _Outcome" LIKE 'Failure (drone ship)'

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

[18]:

MONTH	YEAR	Landing _Outcome	Booster_Version	Launch_Site
01	2015	Failure (drone ship)	F9 v1.1 B1012	CCAFS LC-40
04	2015	Failure (drone ship)	F9 v1.1 B1015	CCAFS LC-40

- Selected for failed landings on drone ship in the year 2015 and displayed the month, booster version, launch site

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

[26]:
```
%%sql
SELECT "Landing _Outcome", COUNT("Landing _Outcome") AS Count FROM SPACEXTBL
WHERE Date > '04-06-2010' and Date < '20-03-2017'
GROUP BY "Landing _Outcome"
ORDER BY Count DESC
```

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

[26]:

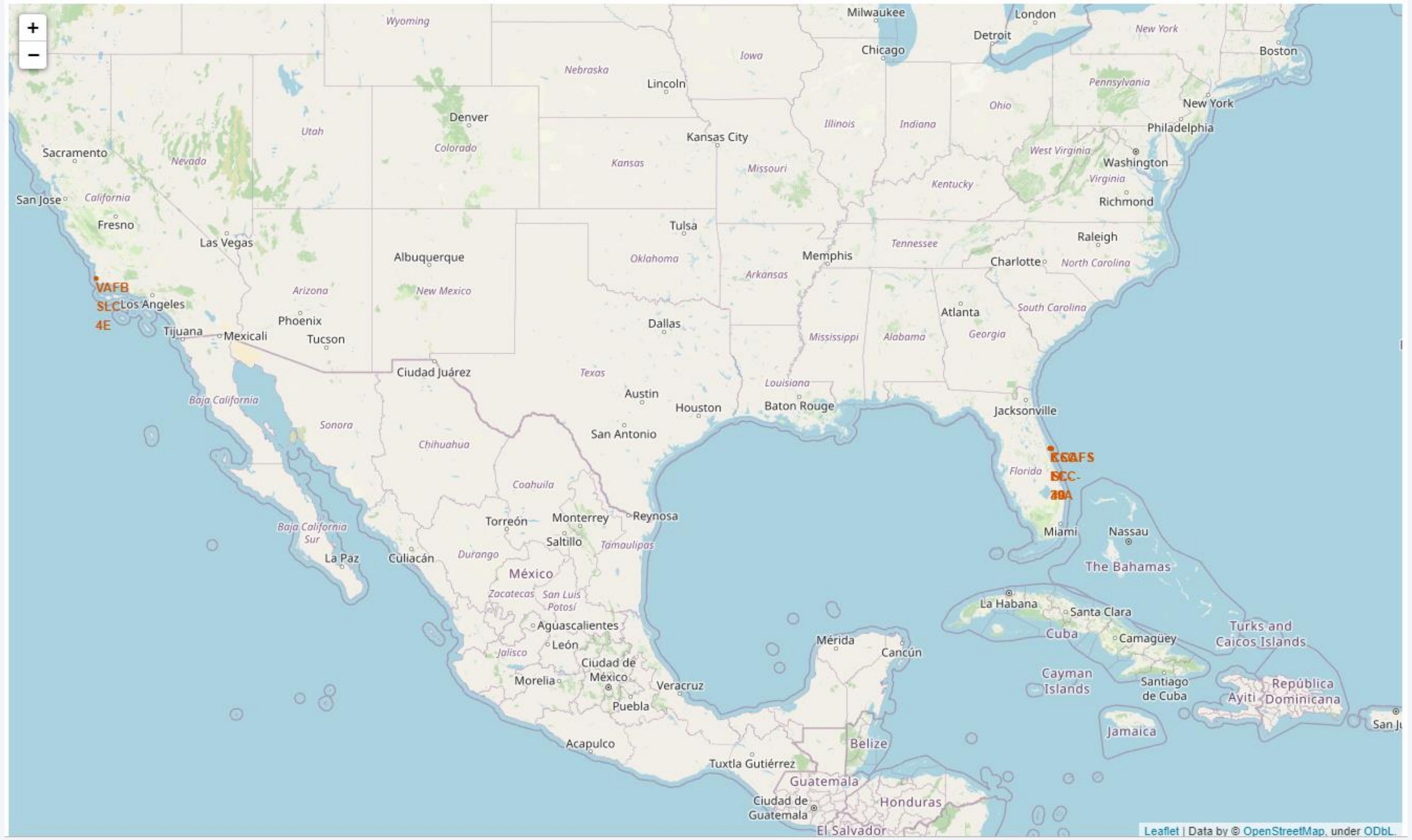
Landing _Outcome	Count
Success	20
No attempt	10
Success (drone ship)	8
Success (ground pad)	6
Failure (drone ship)	4
Failure	3
Controlled (ocean)	3
No attempt	1
Failure (parachute)	1

- Ranked the outcomes by descending order with the most common landing outcome first

Section 3

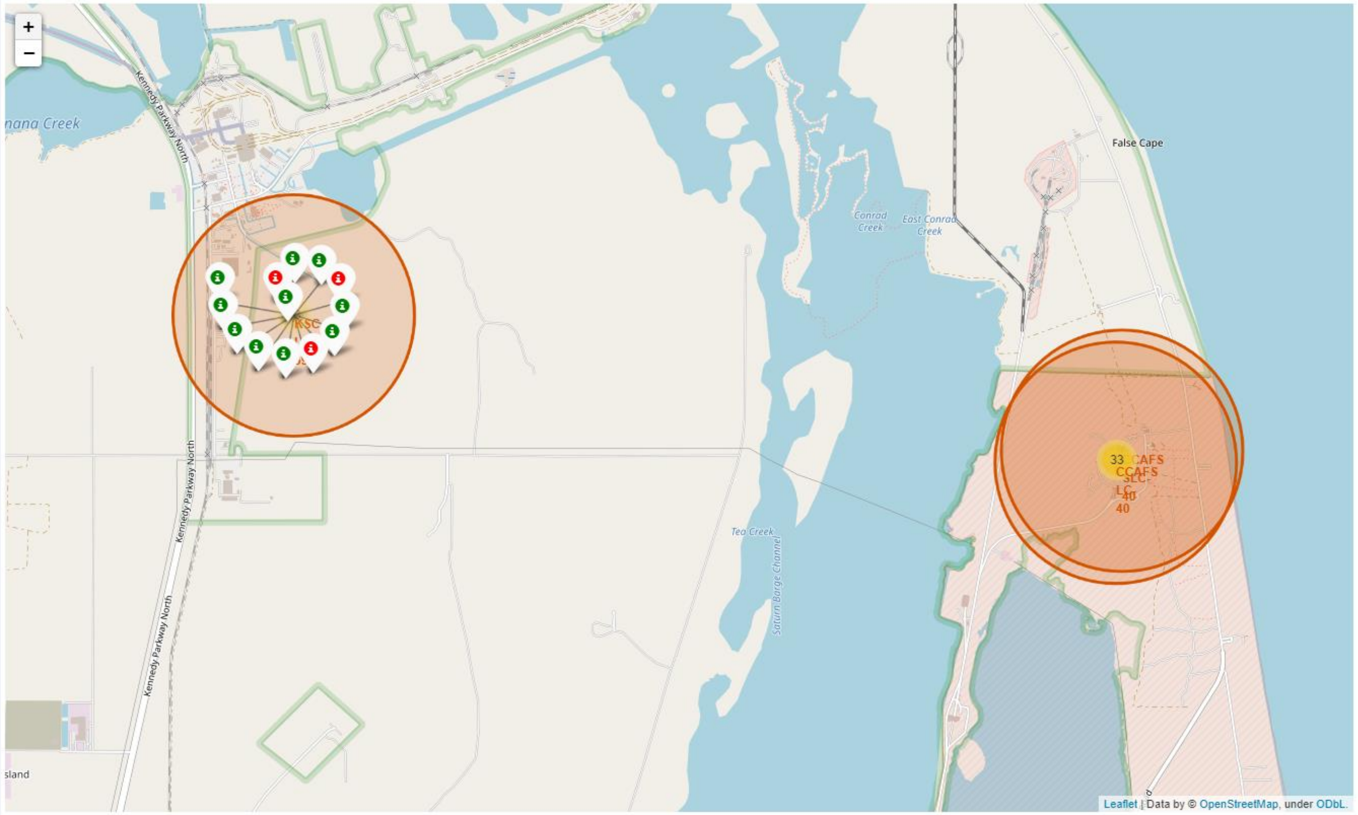
Launch Sites Proximities Analysis

Launch Site Locations



- The launch sites are close to the equator and the sea.
- One site is located on the west coast in California while two sites are located on the east coast in Florida

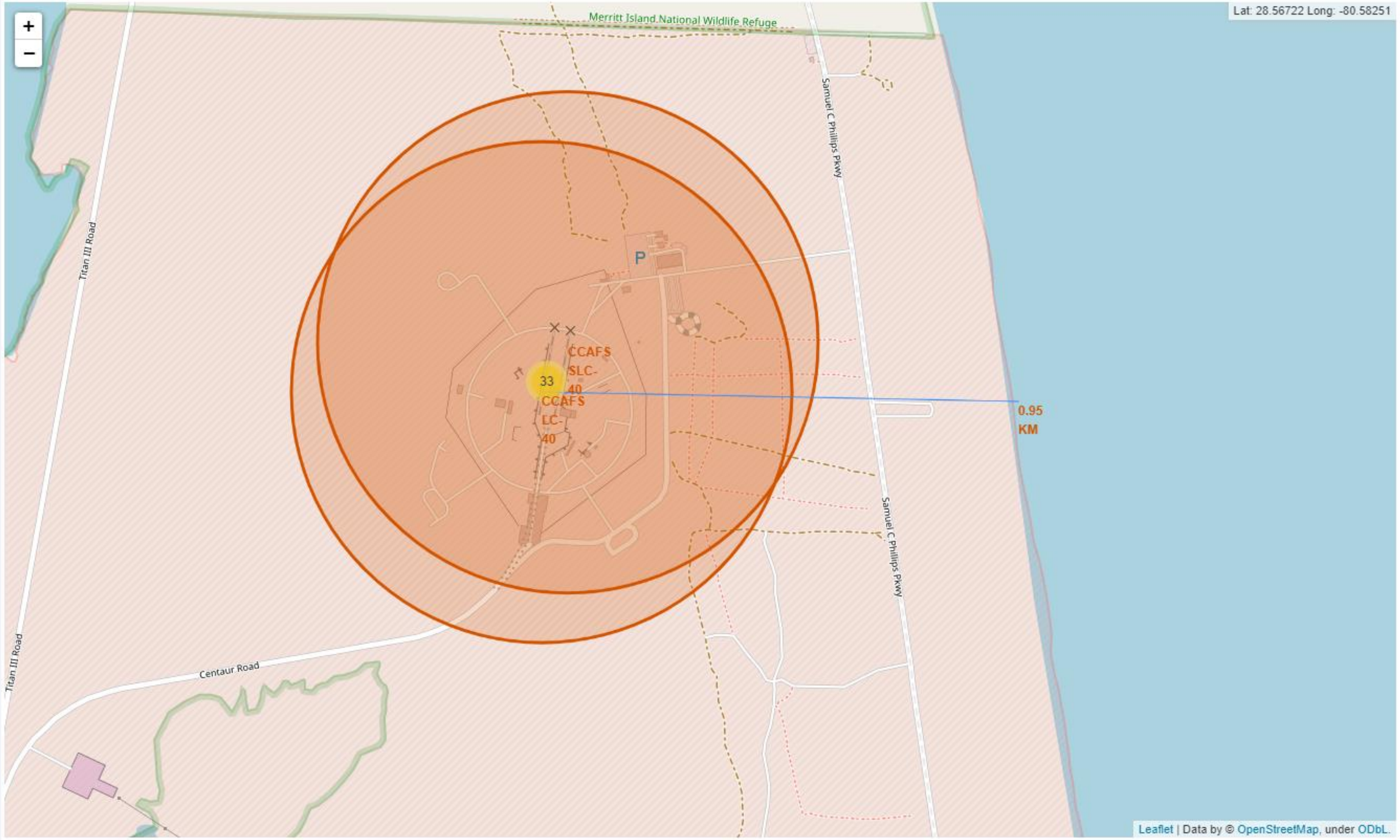
Success / Failed launches per site



- Zooming in to the Florida launch sites, we can see the marker cluster labelled in green for successful outcomes and red for failure outcomes.

- Before we click in, the marker clusters appear as a number in a colored circle as shown on the right

Distance from CCAFS to Coastline

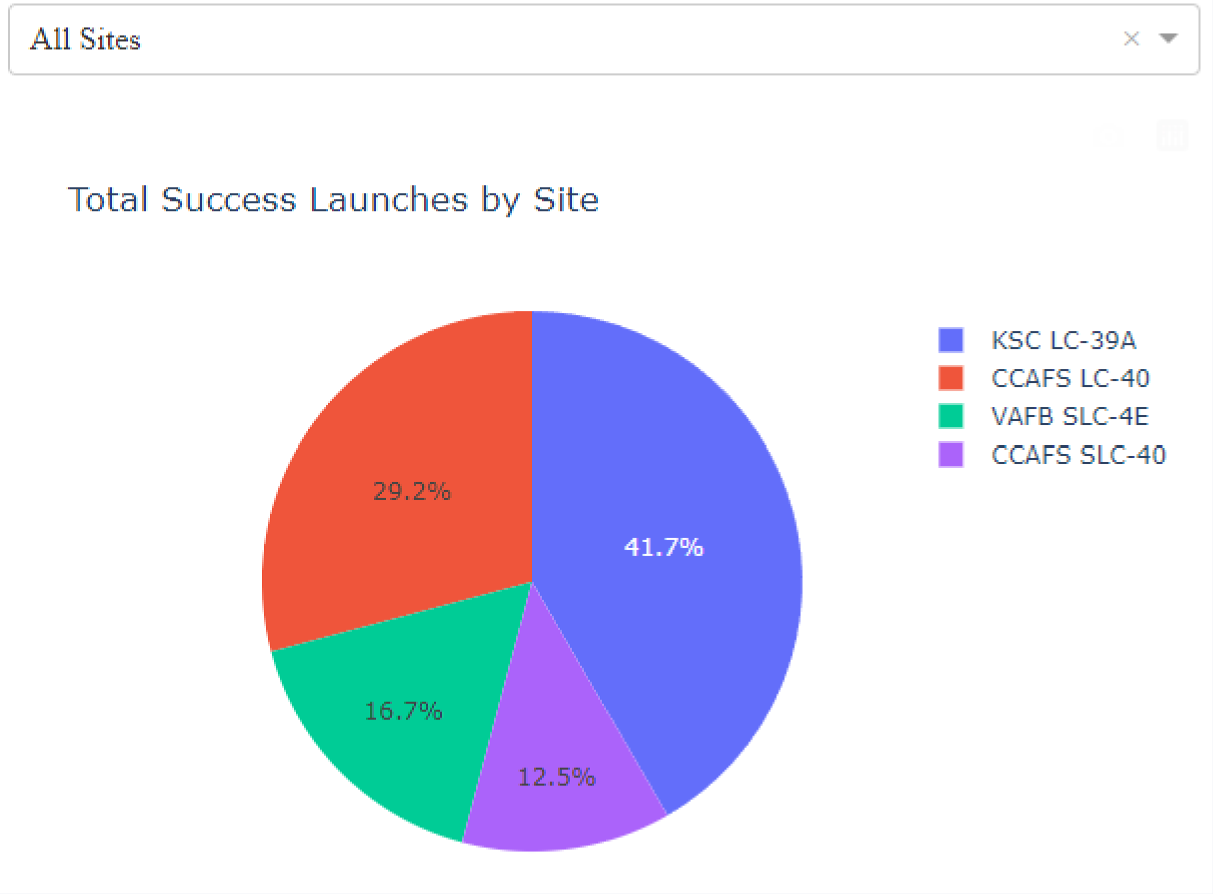


- The distance from CCAFS launch sites to the coastline was marked out on the map

Section 4

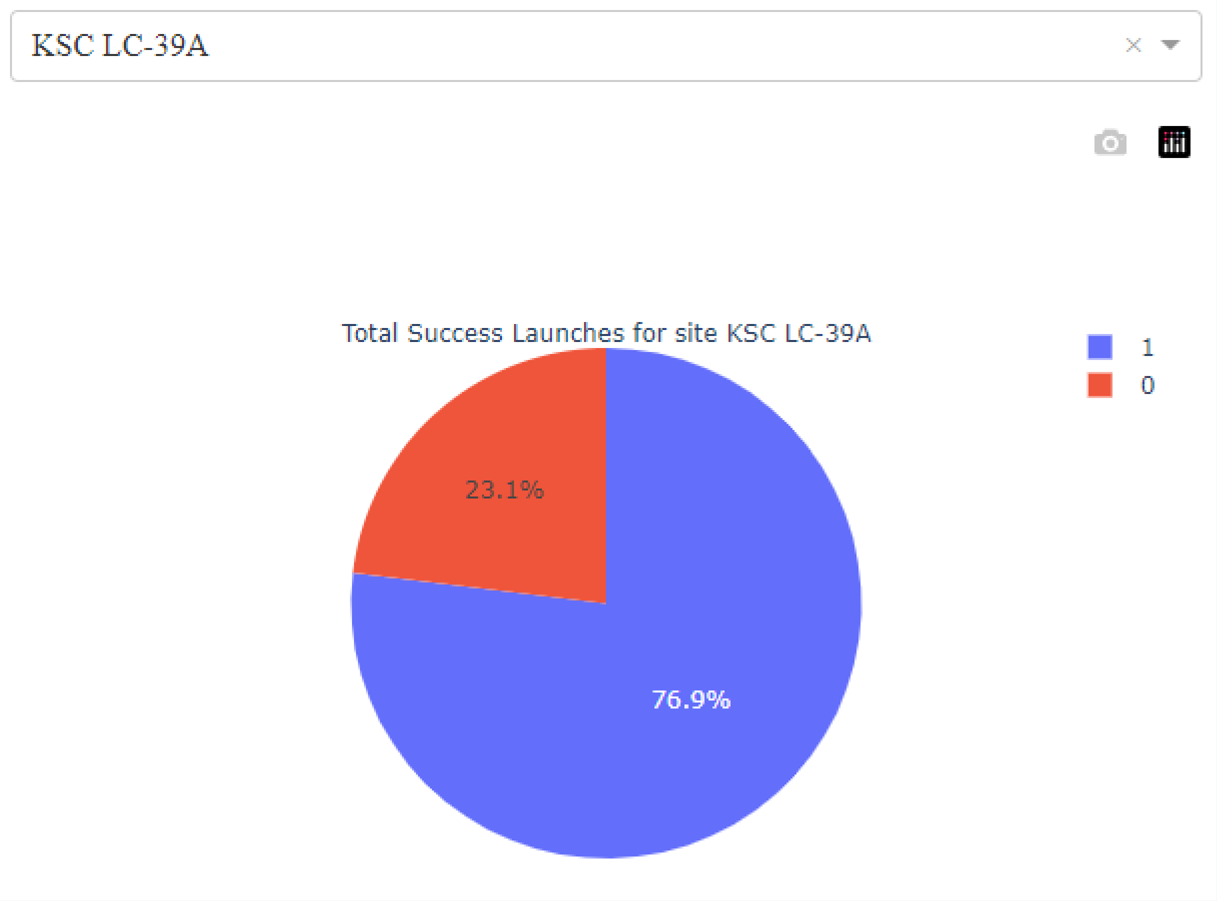
Build a Dashboard with Plotly Dash

Total Success Launches by Site



- The drop down allows the user to select whether to show the outcomes of launches by sites or to show it for all sites.
- The pie chart will update accordingly.
- Here it shows the proportion of total successful launches across all sites

Launch Site with highest launch success ratio



- Launch Site KSC LC-39A has the highest launch success ratio

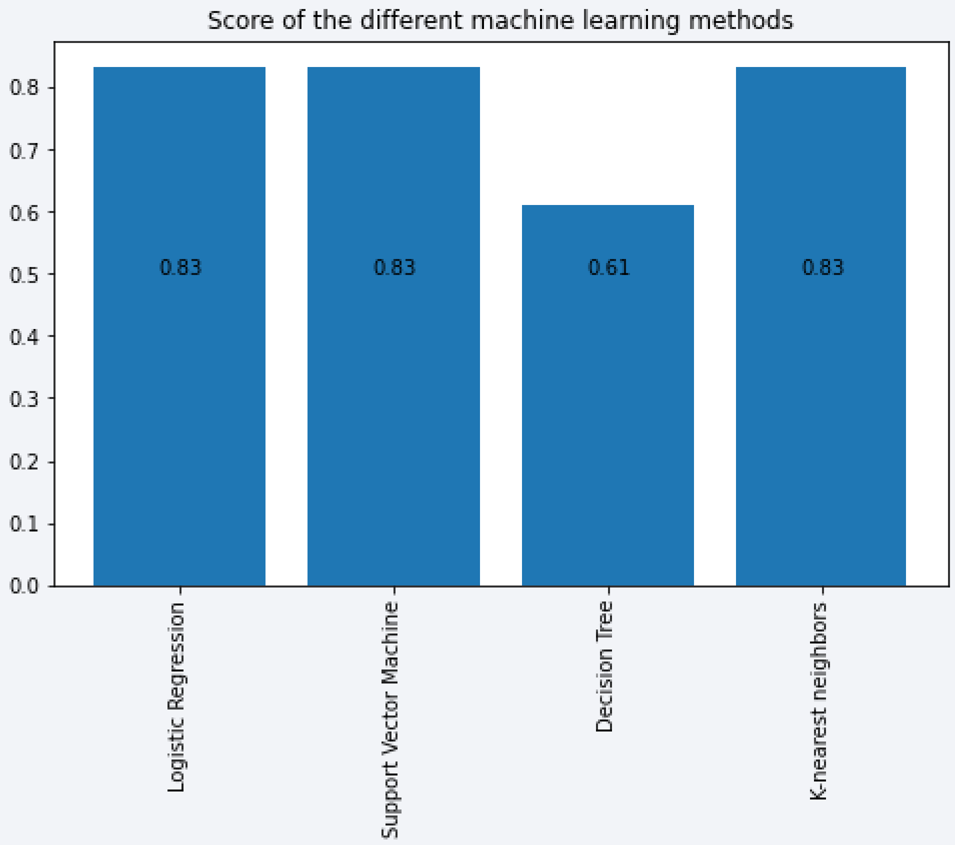
Payload vs Launch Outcome

- Payload range was filtered to be between 2,000 kg and 8,000 kg
- In this payload range, the FT booster has the highest rate of success.
- We can see that beyond 5,500 kg of payload mass, there were no successful launches

Section 5

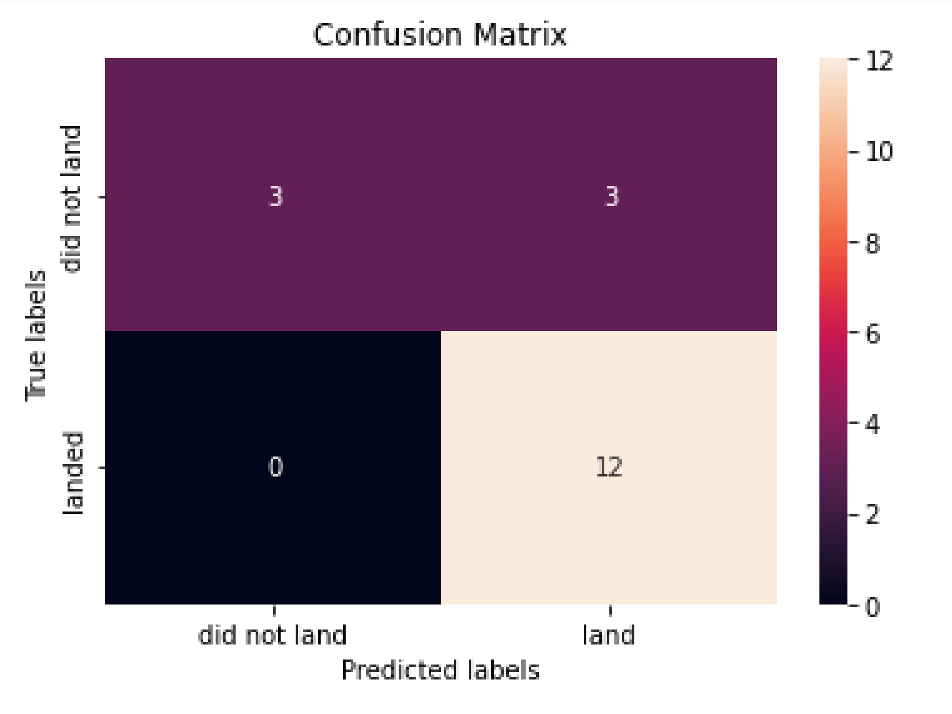
Predictive Analysis (Classification)

Classification Accuracy



- All models had the same accuracy on the test set except for Decision Tree which had a lower accuracy of 0.61.

Confusion Matrix



- The best performing models could predict all successful landings in the test set. Recall = 1

- However, the models have trouble predicting failed landings with Recall = 0.5.

Conclusions

- Chances of a successful landing goes up for later launches by SpaceX, likely due to improvements in booster reliability.
- Success rates for landings are above 0.8 in the last 2 years covered by the data set.
- We found that launch sites are located close to coast lines, likely to ensure safety in case something went wrong and the rocket failed.
- Machine learning models on this test set were able to predict landings with good recall but had a high rate of false positives predicting half of failed landings as successes.

Thank you!