Winning Space Race with Data Science

Joshua McCurry
31/07/2024

Outline

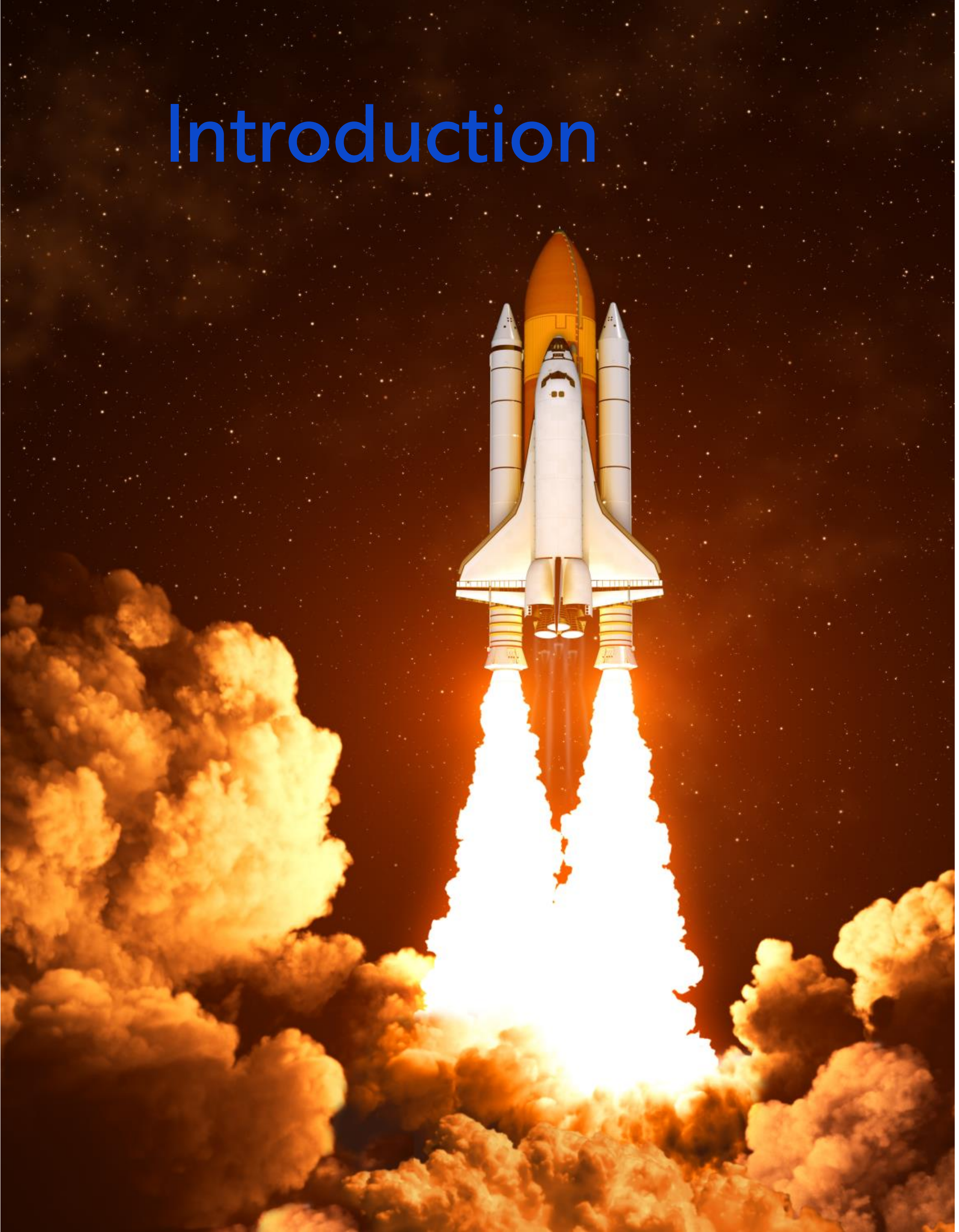
- Executive Summary
- Introduction
- Methodology
- Results
- Conclusion
- Appendix

Executive Summary

Data was extracted from multiple sources including scraping from a website and directly from a web API. Once the data was prepared, exploratory data analysis was performed to inspect patterns in the data. With these patterns in mind, various machine learning techniques were compared to determine which gave the most accurate predictions. We can then use these models to determine which factors in our control can improve the likelihood of a successful launch outcome.

This comparison of techniques shows that the Decision Tree Classifier is the method with the best prediction accuracy.

Introduction



Launching rockets into space is an expensive and risky venture, in order to guarantee the highest return on investment, we would need to find the method of launching the rockets with the best chance of success at the most affordable cost.

SpaceX offers a very competitive price, so we investigated the launch statistics for different locations, boosters and payloads to develop a predictive model to show the best launch options.

Section 1

Methodology

Methodology

Executive Summary

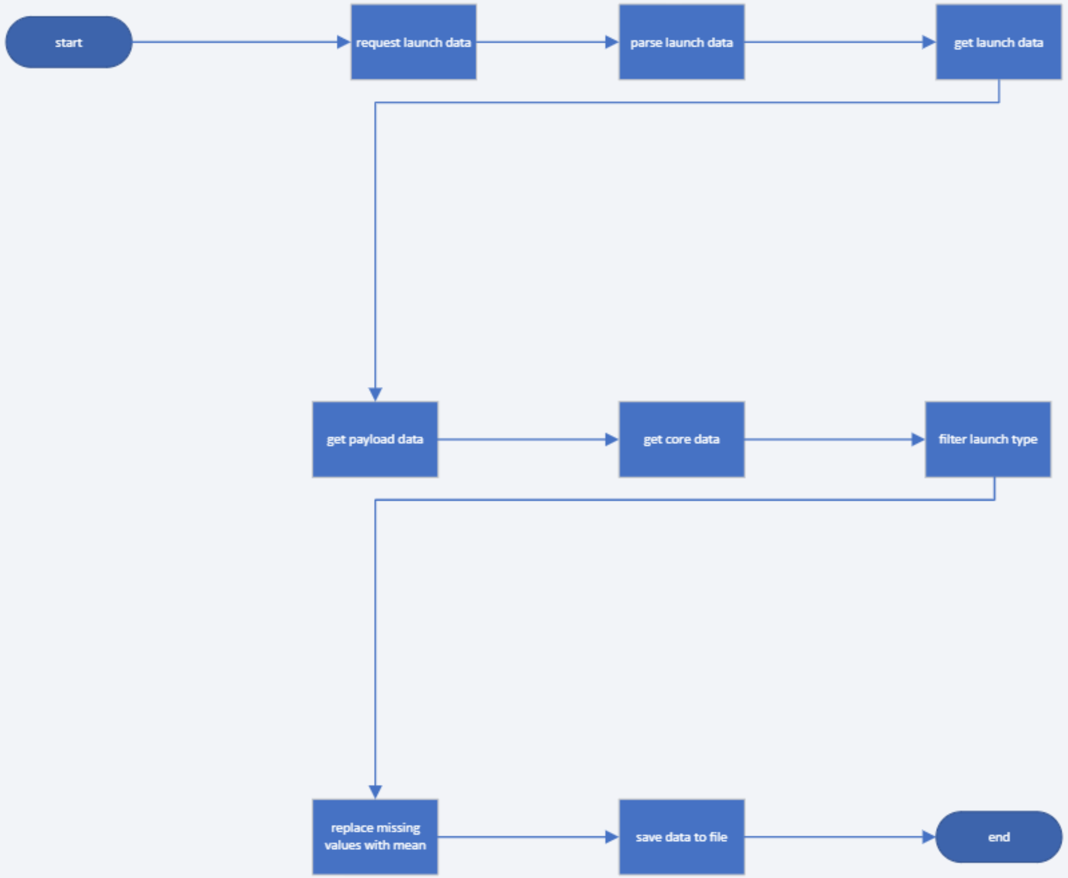
- Data collection methodology
- Perform data wrangling
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - How to build, tune, evaluate classification models

Data Collection – SpaceX API

Data was downloaded from the SpaceX API https://api.spacexdata.com/v4/launches/past

This data was converted to a dataframe using pandas.json_normalize on the response text. This data was then filtered to show only comparable launches, and then missing values were replaced with average values from the same column.

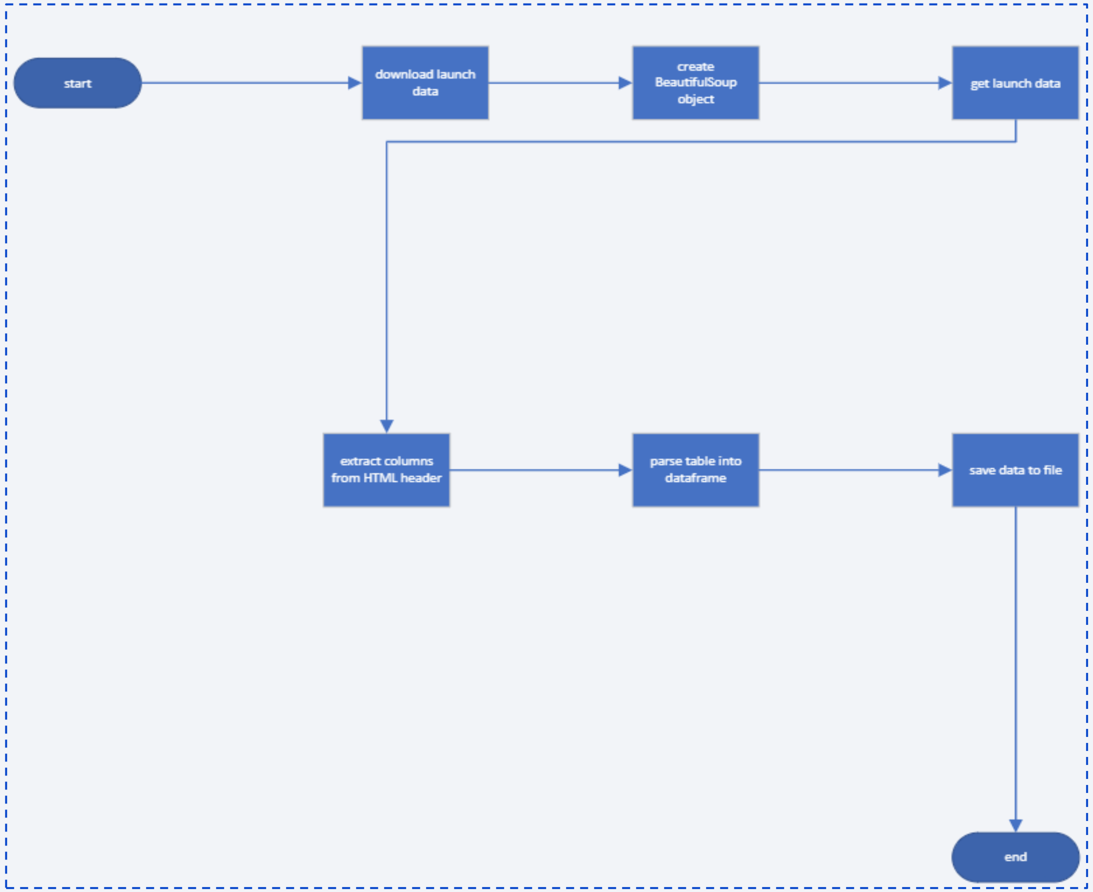
Available from https://github.com/justjoshin78/Applied-Data-Science-Capstone/blob/main/jupyter-labs-spacex-data-collection-api.ipynb



Data Collection - Scraping

The web page is downloaded and a BeautifulSoup object is created from it. A dataframe is then created by parsing the launch tables. This dataframe is then saved to file.

Available from https://github.com/justjoshin78/Applied-Data-Science-Capstone/blob/main/jupyter-labs-spacex-data-collection-api.ipynb

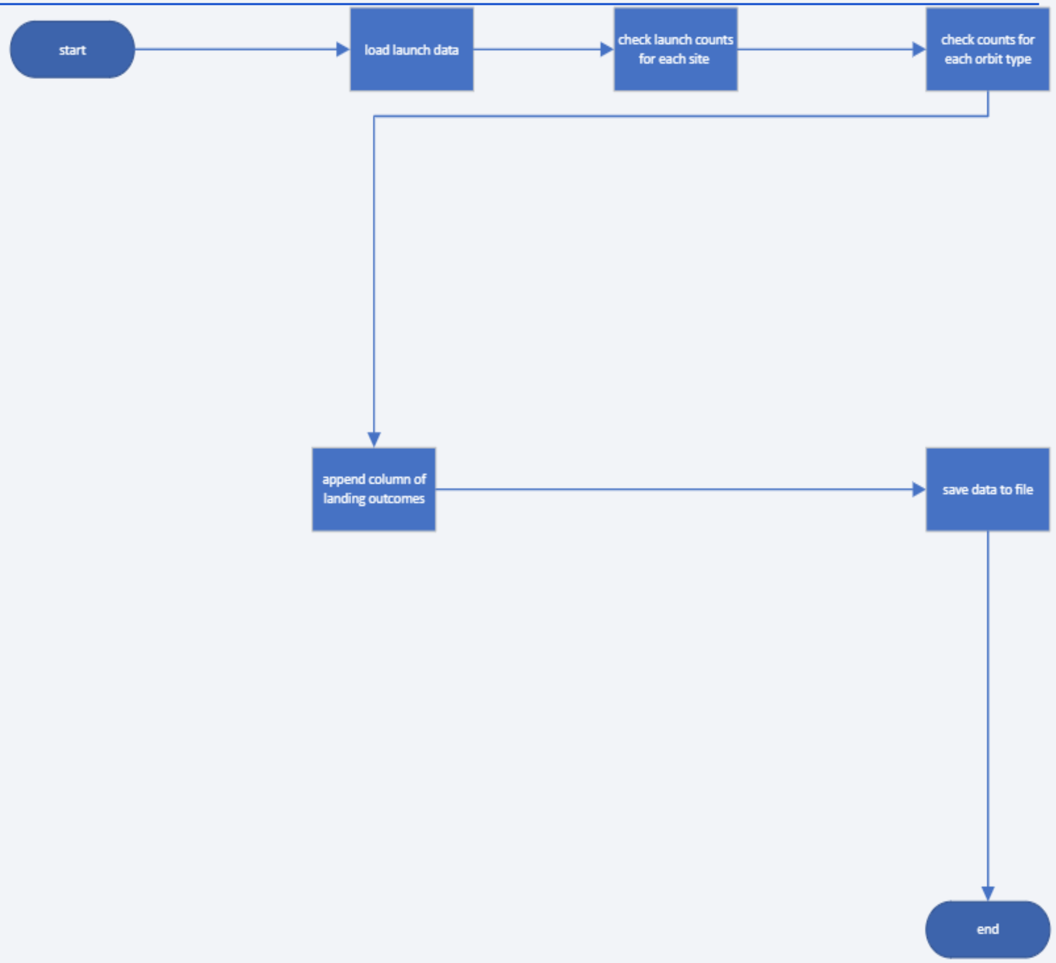


Data Wrangling

Initially basic statistics were generated about launch success rates for different locations and orbits. Then landing outcomes were categorized for all launches and a column containing this data was appended. This data was then saved to file.

Available from

https://github.com/justjoshin78/Applied-Data-Science-Capstone/blob/main/labs-jupyter-spacex-Data-wrangling.ipynb



EDA with Data Visualization

Scatter plots were used to investigate the relationship between different pairs of variable to determine which ones to focus on. A bar chart was used to examine the success rate of categorical variables. A line chart was used to see the trend in average success rate over time.

Available from https://github.com/justjoshin78/Applied-Data-Science-Capstone/blob/main/jupyter-labs-eda-dataviz.ipynb

EDA with SQL

- Clean up previous temporary table
  DROP TABLE IF EXISTS SPACEXTABLE
- Create a temporary table with null date rows removed
  create table SPACEXTABLE as select * from SPACEXTBL where Date is not null
- List all launch sites
  SELECT DISTINCT Launch_Site FROM SPACEXTABLE
- List 5 launch sites that start with CCA
  SELECT * FROM SPACEXTABLE WHERE Launch_Site LIKE 'CCA%' LIMIT 5
- Show total payload mass for customer NASA (CRS)
  SELECT SUM(PAYLOAD_MASS__KG_) FROM SPACEXTABLE WHERE Customer = 'NASA (CRS)’

EDA WITH SQL

- Show average payload mass for booster F9 v1.1
  SELECT AVG(PAYLOAD_MASS__KG_) FROM SPACEXTABLE WHERE Booster_Version = 'F9 v1.1'
- Show the earliest date of a successful landing on ground pad
  SELECT MIN(Date) FROM SPACEXTABLE WHERE Landing_Outcome = 'Success (ground pad)'
- Show the boosters that successfully landed on a drone ship with payload between 4000 and 6000kg
  SELECT Booster_Version FROM SPACEXTABLE WHERE Landing_Outcome = 'Success (drone ship)' AND PAYLOAD_MASS__KG_ BETWEEN 4000 AND 6000

EDA WITH SQL

- Show the total number of successful and failed launches
```
SELECT 'success', COUNT(*)
FROM SPACEXTABLE
WHERE Landing_Outcome LIKE 'Success%'
UNION SELECT 'failure', COUNT(*)
FROM SPACEXTABLE
WHERE Landing_Outcome LIKE 'Failure%'
```
- Show the booster/s used to launch the largest payload mass
```
SELECT Booster_Version
FROM SPACEXTABLE
WHERE PAYLOAD_MASS__KG_ IN
(
    SELECT MAX(PAYLOAD_MASS__KG_)
    FROM SPACEXTABLE
)
```
- Show the month, outcome, booster and site of the launch/s that failed to land on a drone ship in 2015
```
SELECT substr(Date, 6,2) AS Month, Landing_Outcome, Booster_Version, Launch_Site
FROM SPACEXTABLE
WHERE Landing_Outcome = 'Failure (drone ship)' AND substr(Date,0,5)='2015'
```

EDA WITH SQL

- Show all launches between 2010-06-04 AND 2017-03-20
  SELECT *
  FROM SPACEXTABLE
  WHERE Date BETWEEN '2010-06-04' AND '2017-03-20'

- Show the totals of all landing outcomes between 2010-06-04 AND 2017-03-20 in descending order
  SELECT Landing_Outcome, COUNT(*) as Count
  FROM SPACEXTABLE
  WHERE Date BETWEEN '2010-06-04' AND '2017-03-20'
  GROUP BY Landing_Outcome
  ORDER BY COUNT(*) DESC

Available from https://github.com/justjoshin78/Applied-Data-Science-Capstone/blob/main/jupyter-labs-eda-sql-coursera_sqllite.ipynb

Build an Interactive Map with Folium

- Circles and Markers are added at launch sites to denote the site with a popup to show details about launch outcomes. Lines were added to the map to show the distance between the sites and certain landmarks.

- Available here https://github.com/justjoshin78/Applied-Data-Science-Capstone/blob/main/lab_jupyter_launch_site_location.ipynb

Build a Dashboard with Plotly Dash

A pie chart and a scatter chart were added to the dash application, along with a dropdown and a rangeslider to control the parameters of these graphs. Callbacks were then added to pass the values from the input controls (dropdown/rangeslider) to the output controls (pie/scatter chart).

Available here https://github.com/justjoshin78/Applied-Data-Science-Capstone/blob/main/spacex_dash_app.py

Predictive Analysis (Classification)

- The data was partitioned into testing and training sets. Then various types of machine learning models are generated including logistic regression, support vector machines, decision trees and K nearest neighbour classifiers using the training data. The parameters of these models are then tuned for the best performance against the test set, and the performance of these models is compared.

Available here https://github.com/justjoshin78/Applied-Data-Science-Capstone/blob/main/SpaceX_Machine_Learning_Prediction_Part_5.jupyterlite.ipynb

PREDICTIVE ANALYSIS (CLASSIFICATION)

Results

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

Section 2

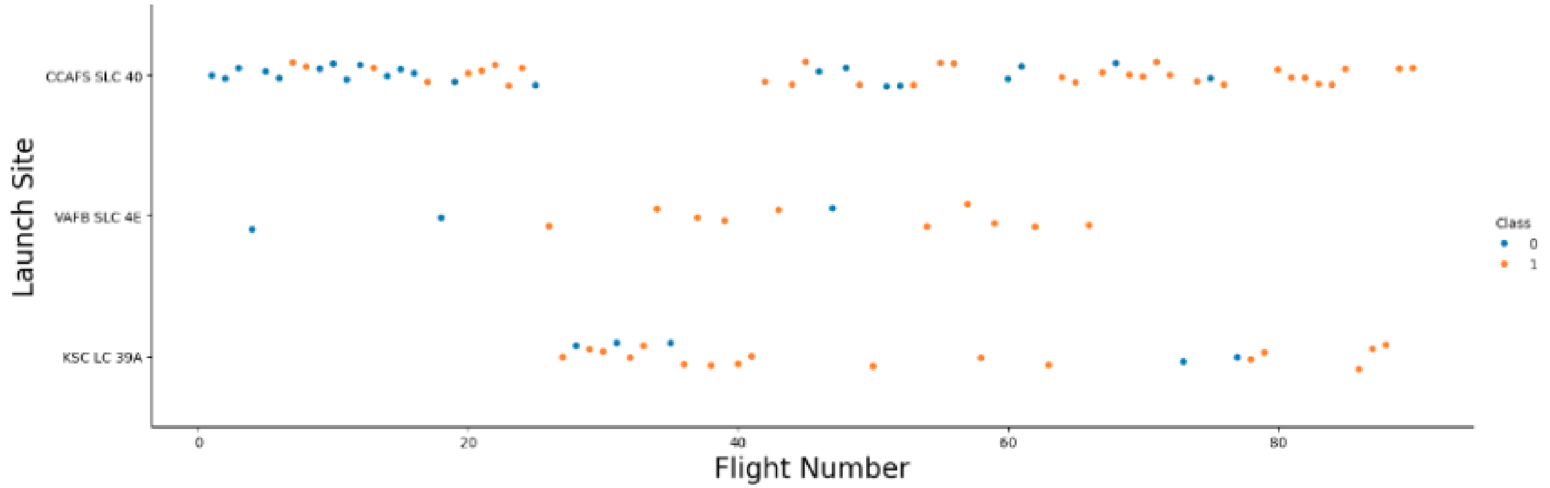
Insights drawn from EDA

Flight Number vs. Launch Site

- Scatter plot of Flight Number vs. Launch Site

In [7]:

```
# 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
sns.catplot(y="LaunchSite", x="FlightNumber", hue="Class", data=df, aspect = 3)
plt.xlabel("Flight Number",fontsize=20)
plt.ylabel("Launch Site",fontsize=20)
plt.show()
```



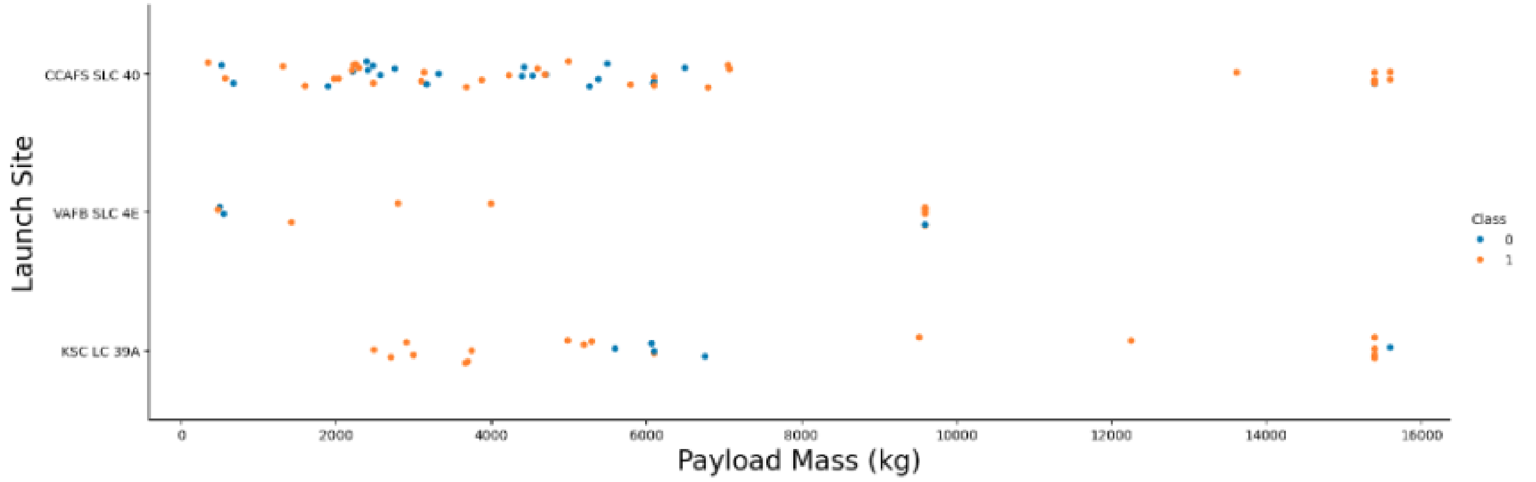
Most early flights were from the CCAFS SLC 40 launch site, with later flights being shared between the other sites. Site VAFB SLC 4E had a few early flights and some in the middle period, and site KSC LC 39A had flights in the middle and late periods.

Payload vs. Launch Site

- Scatter plot of 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
sns.catplot(y="LaunchSite", x="PayloadMass", hue="Class", data=df, aspect = 3)
plt.xlabel("Payload Mass (kg)",fontsize=20)
plt.ylabel("Launch Site",fontsize=20)
plt.show()
```



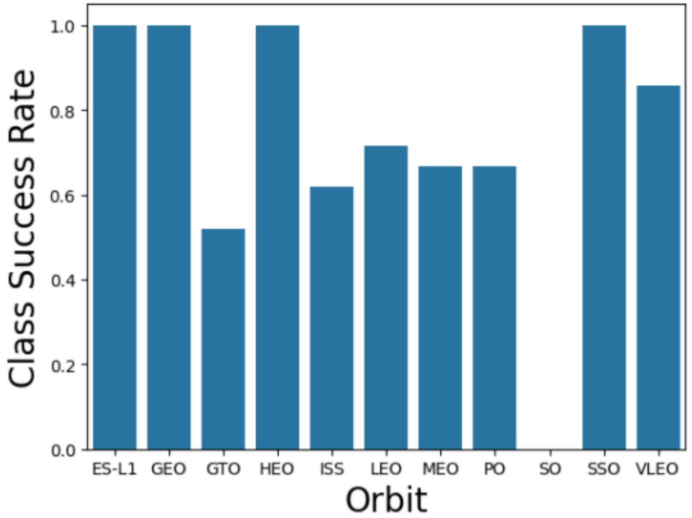
Examination of this graph shows that launches from CCAFS SLC 40 and KSC LC 39A have gone as high as nearly 16000kg, while launches from VAFB SLC 4E have a maximum launch mass so far of just under 10000kg.

Success Rate vs. Orbit Type

- Bar chart for the success rate of each orbit type

In [34]:

```
# HINT use groupby method on Orbit column and get the mean of Class column
df_orbit = pd.DataFrame(df.groupby('Orbit')['Class'].mean())
#print(df_orbit)
sns.barplot(x = 'Orbit', y = 'Class', data = df_orbit)
plt.xlabel("Orbit",fontsize=20)
plt.ylabel("Class Success Rate",fontsize=20)
plt.show()
```

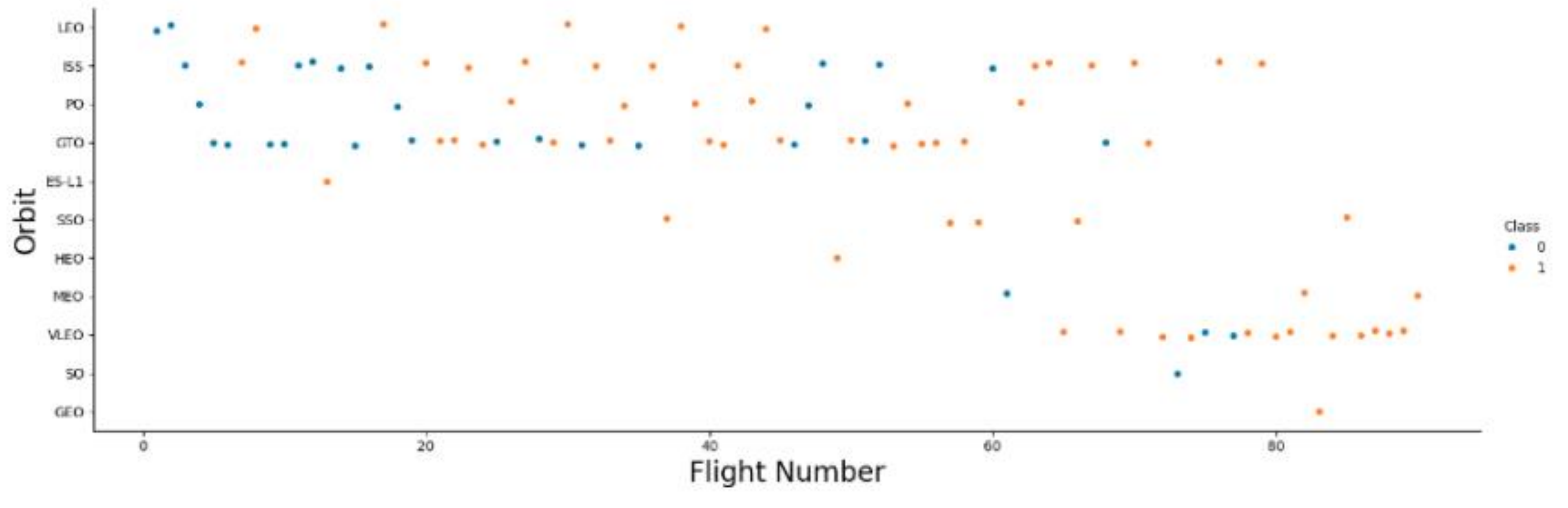


Analysis of this graph shows that certain orbital types of launches (ES-L1, GEO, HEO, SSO) have been completely successful, and one type of launch (SO) has no successful launches. The other orbital launch types have varying degrees of success.

Flight Number vs. Orbit Type

- Scatter plot of Flight number vs. Orbit type

```
# Plot a scatter point chart with x axis to be FlightNumber and y axis to be the Orbit, and hue to be the class v
sns.catplot(y="Orbit", x="FlightNumber", hue="Class", data=df, aspect = 3)
plt.xlabel("Flight Number",fontsize=20)
plt.ylabel("Orbit",fontsize=20)
plt.show()
```

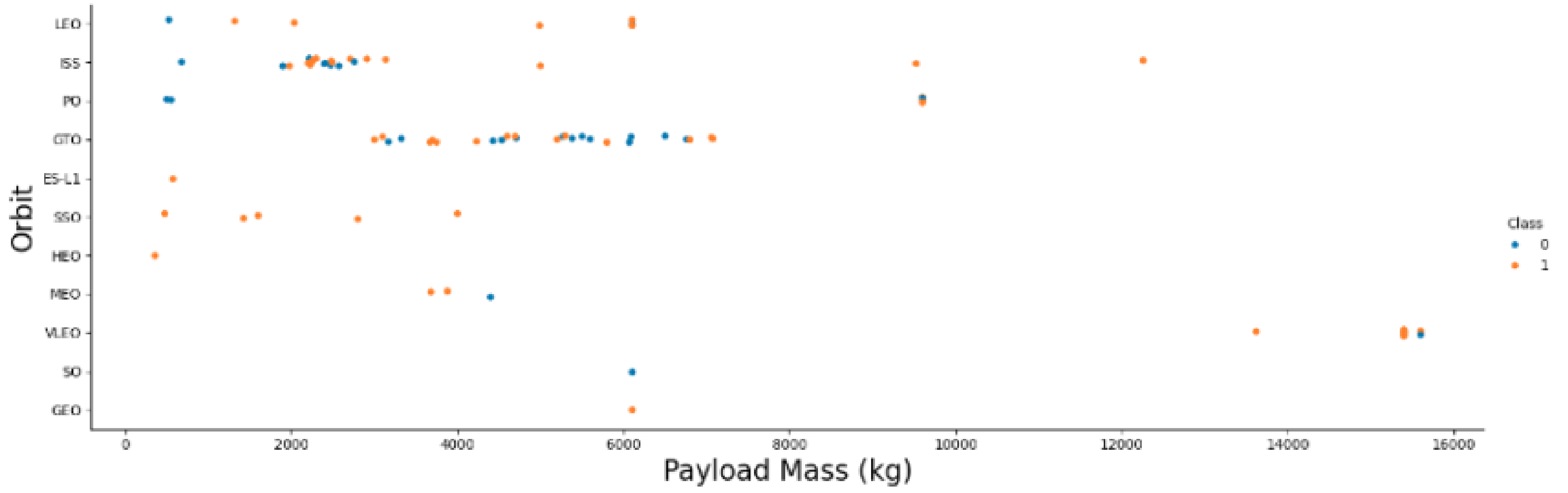


From visual inspection you can see that some orbital launch types were attempted earlier than others. You can also see that success rates tended to improve in each of these orbital launch types over time, with more failures early and more successes later.

Payload vs. Orbit Type

- Scatter plot of payload vs. orbit type

```
# 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(y="Orbit", x="PayloadMass", hue="Class", data=df, aspect = 3)
plt.xlabel("Payload Mass (kg)",fontsize=20)
plt.ylabel("Orbit",fontsize=20)
plt.show()
```

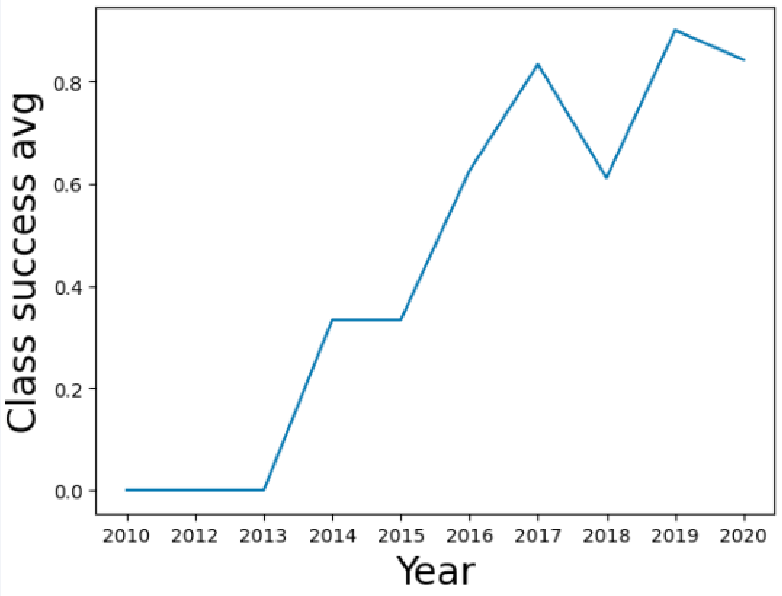


Certain orbital launch types are associated with lower payloads while others are associated with larger payloads. A couple of the orbital launch types are associated with a wide range of payload sizes.

Launch Success Yearly Trend

- 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
df_year = pd.DataFrame(df.groupby('Year')['Class'].mean())
sns.lineplot(x = 'Year', y = 'Class', data = df_year)
plt.xlabel("Year",fontsize=20)
plt.ylabel("Class success avg",fontsize=20)
plt.show()
```



Launch success rates have increased greatly over time, from zero at the beginning to over eighty percent successful launches in recent years.

All Launch Site Names

Query : SELECT DISTINCT Launch_Site FROM SPACEXTABLE

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

There are 4 different launch sites that SpaceX has used.

Launch Site Names Begin with 'CCA'

Query : SELECT * FROM SPACEXTABLE
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

The query selects 5 rows from the table that have a launch site beginning with CCA.

Total Payload Mass

Query : SELECT SUM(PAYLOAD_MASS__KG_) FROM SPACEXTABLE
WHERE Customer = 'NASA (CRS)'

SUM(PAYLOAD_MASS__KG_)
45596

This query returns the sum of the payloads for all launches with the customer “NASA (CRS)”

Average Payload Mass by F9 v1.1

Query : SELECT AVG(PAYLOAD_MASS__KG_) FROM SPACEXTABLE
WHERE Booster_Version = 'F9 v1.1'

AVG(PAYLOAD_MASS__KG_)
2928.4

The query returns the average payload for all launches with the booster "F9 v1.1"

First Successful Ground Landing Date

Query : SELECT MIN(Date) FROM SPACEXTABLE
WHERE Landing_Outcome = 'Success (ground pad)'

MIN(Date)
2015-12-22

The query returns the minimum date for a launch where the outcome is "Success (ground pad".

Successful Drone Ship Landing with Payload between 4000 and 6000

Query :
```
SELECT Booster_Version FROM SPACEXTABLE
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

The query returns all the boosters from launches with a successful landing on a drone ship with payload between 4000 and 6000kg.

Total Number of Successful and Failure Mission Outcomes

Query : SELECT 'success', COUNT(*) FROM SPACEXTABLE
WHERE Landing_Outcome LIKE 'Success%'
UNION
SELECT 'failure', COUNT(*) FROM SPACEXTABLE
WHERE Landing_Outcome LIKE 'Failure%'

'success'	COUNT(*)
failure	10
success	61

This query combines the results of two other queries which calculate the counts of successes and failures separately.

Boosters Carried Maximum Payload

```
Query : SELECT Booster_Version
        FROM SPACEXTABLE
        WHERE PAYLOAD_MASS__KG_ IN
        (
                SELECT MAX(PAYLOAD_MASS__KG_)
                FROM SPACEXTABLE
        )
```

This query uses a subselect to get the maximum payload, which is used to find all of the boosters used to launch loads of that size.

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

2015 Launch Records

Query : SELECT substr(Date, 6,2) AS Month, Landing_Outcome,
Booster_Version, Launch_Site
FROM SPACEXTABLE
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

The query shows information for launches in 2015 that failed to land on a drone ship.

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

Query : SELECT Landing_Outcome, COUNT(*) as Count
FROM SPACEXTABLE
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

This query lists the landing outcomes along with their count in the date range specified in descending order.

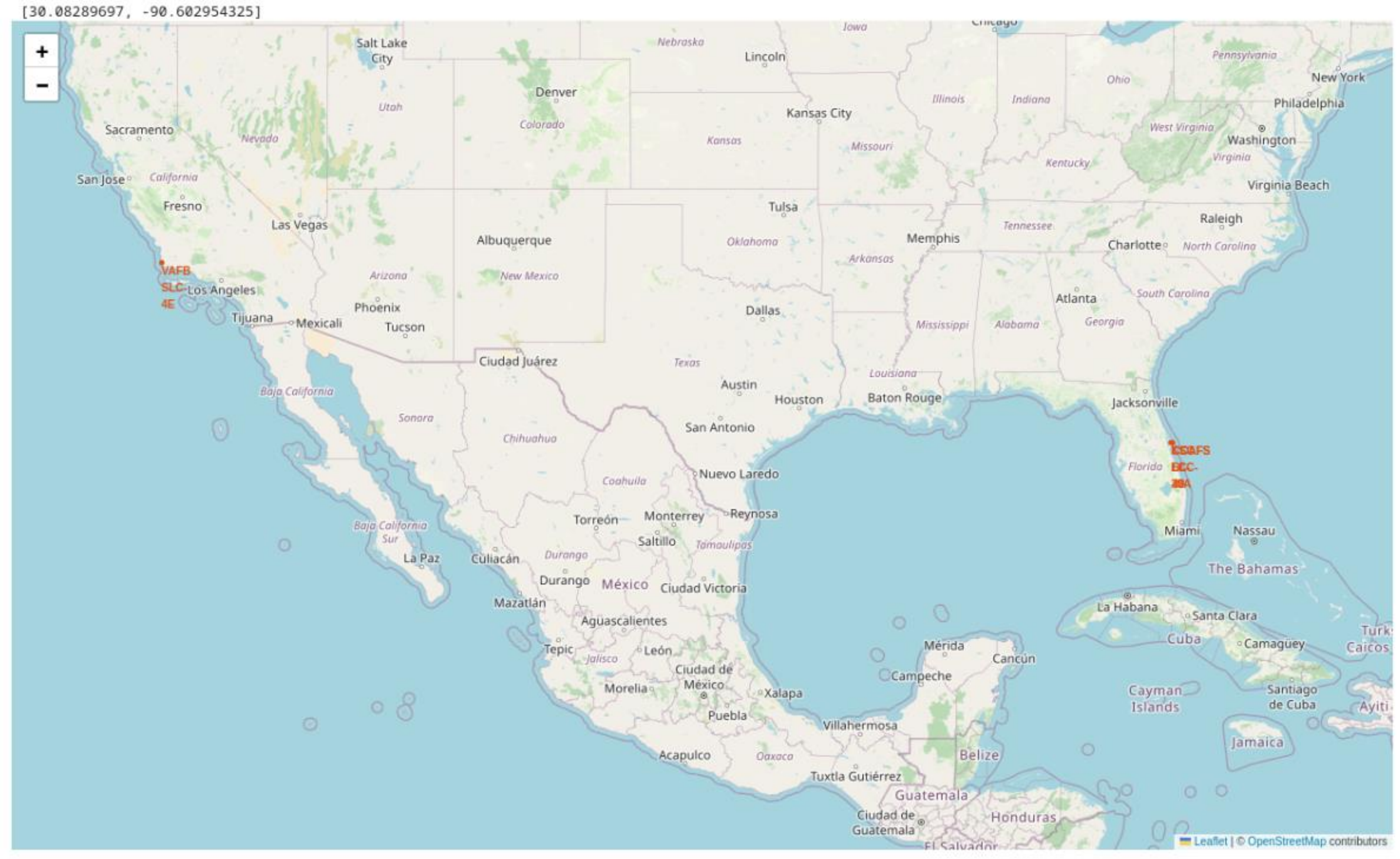
Section 3

Launch Sites Proximities Analysis

Launch site analysis with Folium

There are four launch locations used by SpaceX in the data.

VAFB SLC-4E (California)
KSC LC-39A (Florida)
CCAFS SLC-40 (Florida)
CCAFS LC-40 (Florida)

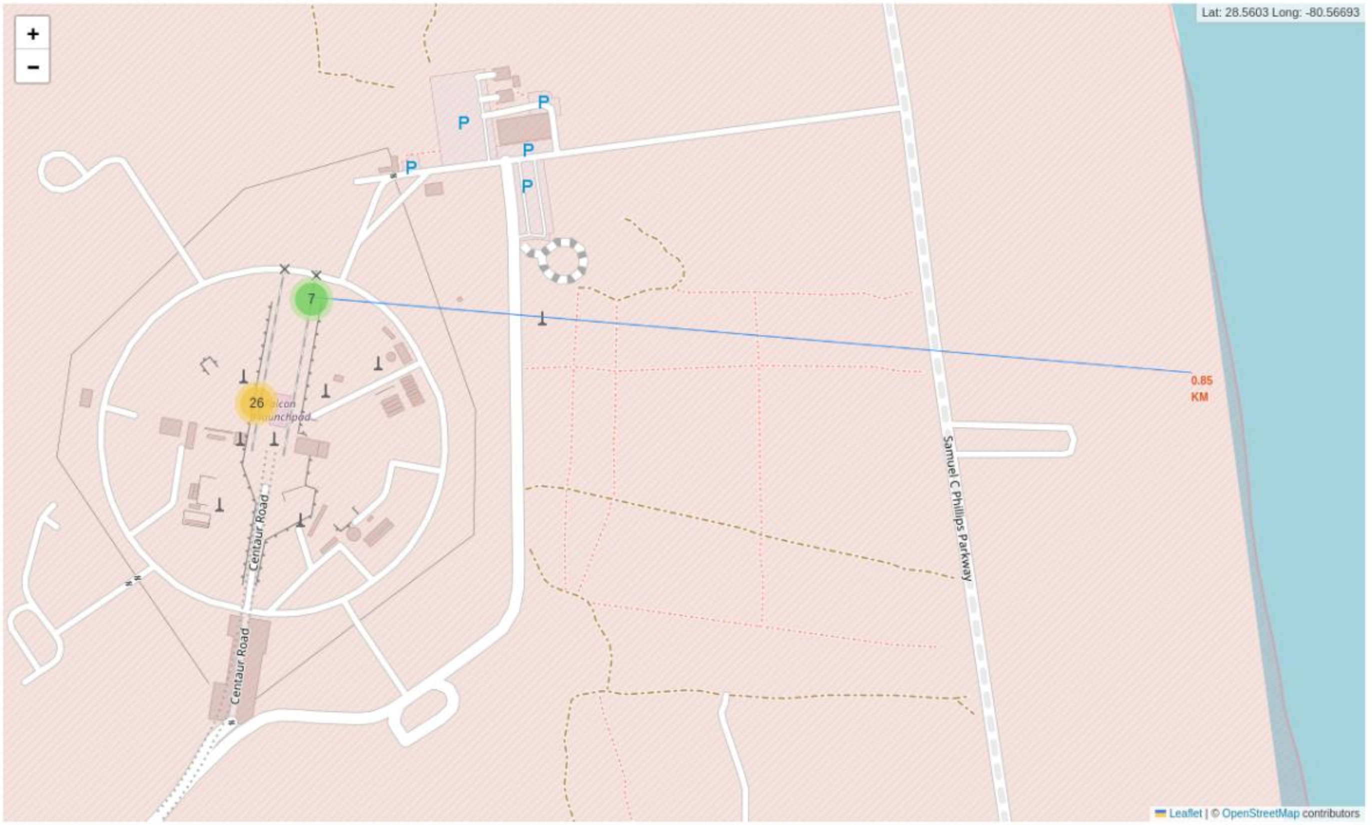


Colour coded launch success marking with Folium

The coloured marker clusters on the map show the success or failure of launches from those locations. Green markers are used to designate a successful launch while red show unsuccessful launches.

Launch site proximities to landmarks using Folium

The map adjacent shows the distance to the coastline of the launch site. This is important as any failed launches should fall into water instead of falling over land. Launch sites are also located close to railways so heavy equipment can be delivered, along with a reasonable distance from population centres so that any failed launches land away from them.

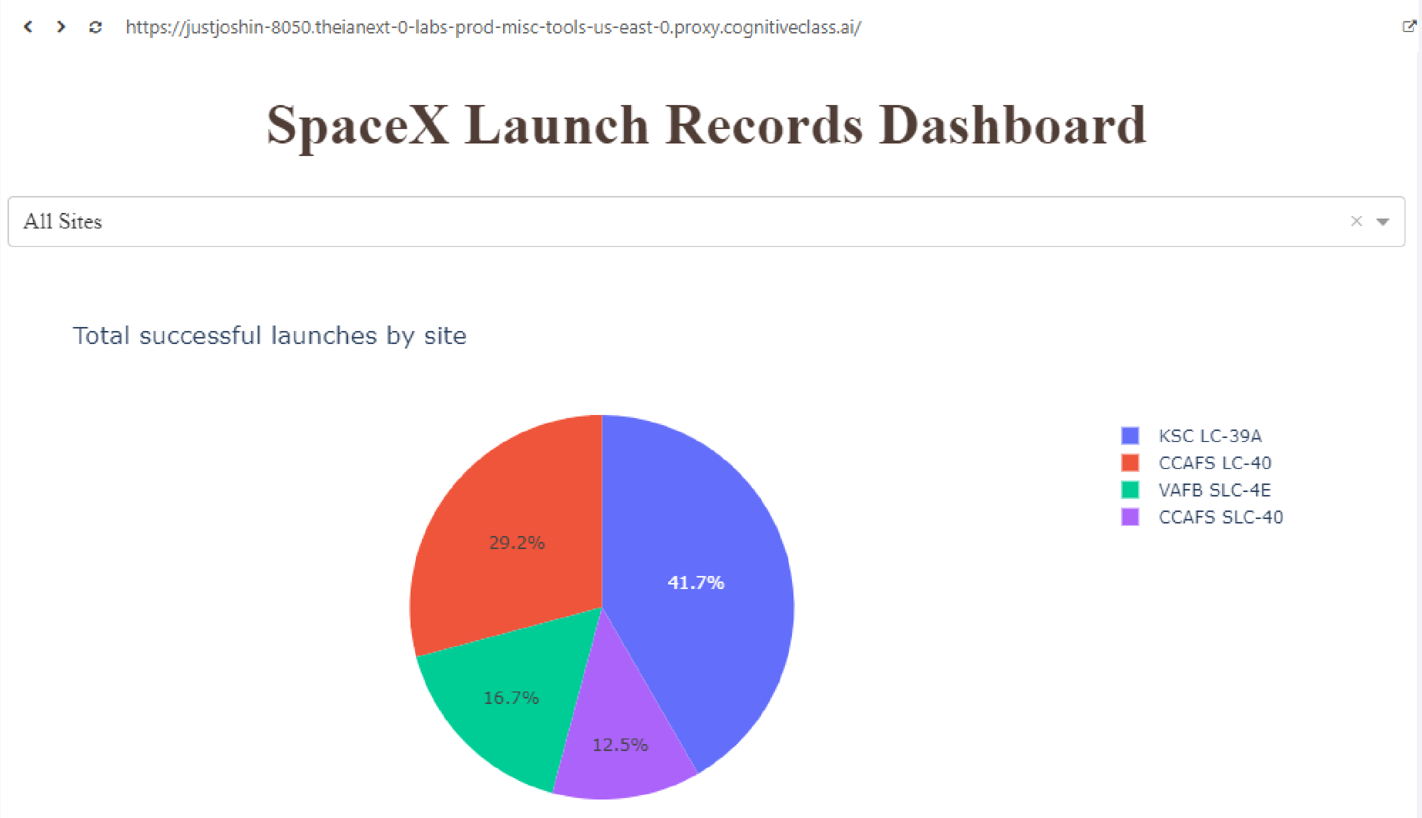


Section 4

Build a Dashboard with Plotly Dash

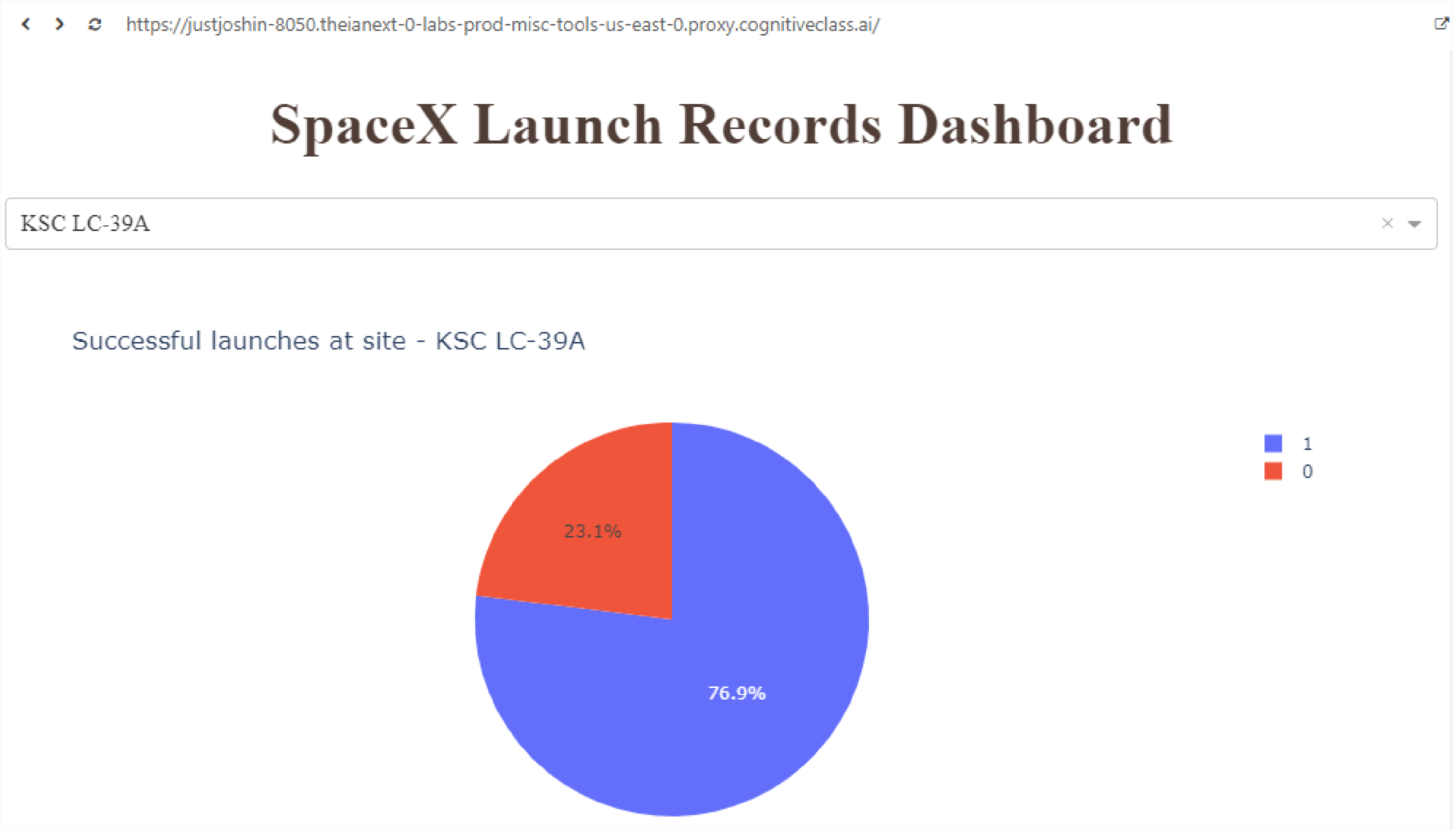
Dash application pie chart of successful launches

The dropdown allows users to select either an individual site or “all sites”. This will filter the data displayed on the graphs. When all sites are selected, the pie chart will compare the number of successful launches at all sites.



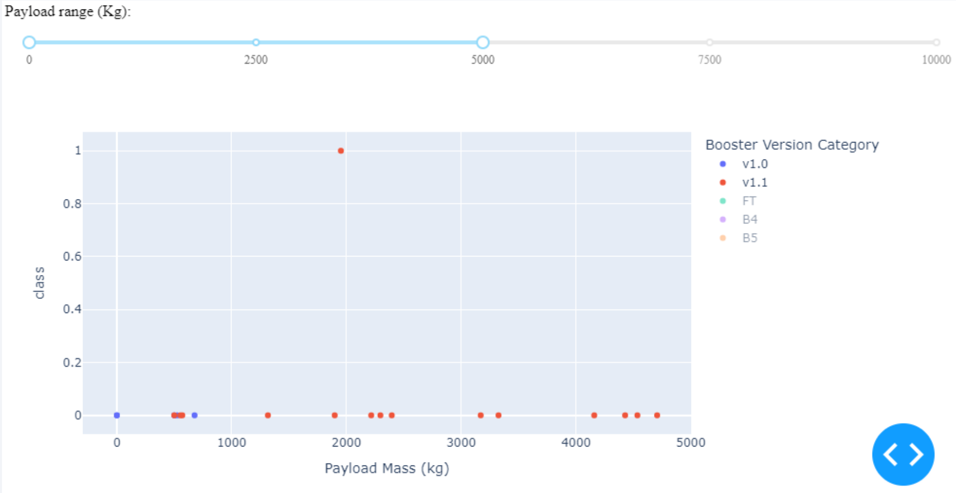
Dash app pie chart of most successful launches

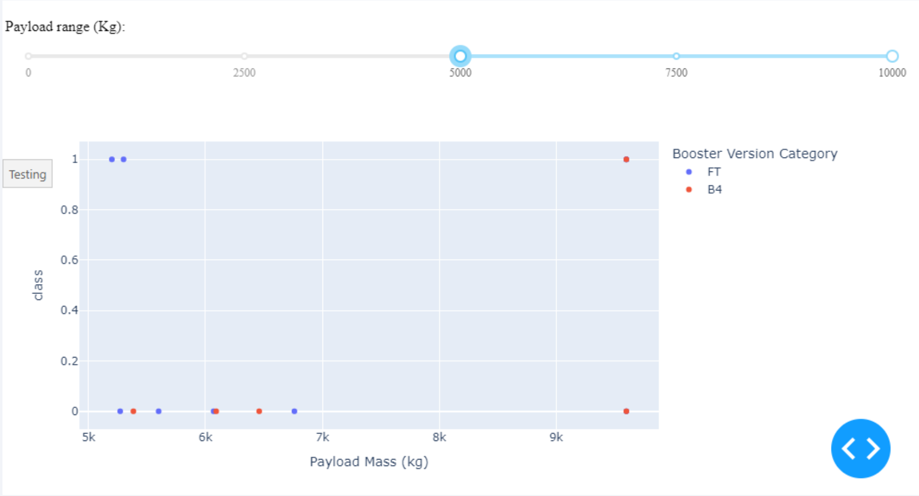
- If an individual site is selected, the pie chart will compare successful and unsuccessful launches at that site. The site with the highest rate of successful launches is KSC LC-39A.



Payload vs launch outcome for all sites

Earlier booster versions (v1.0/v1.1) only launched lower payloads, and had a much lower success rate than later versions. Comparing the later versions at different payload ranges shows much higher success rates at lower payloads suggesting it may be more difficult to launch larger masses.





Section 5

Predictive Analysis (Classification)

Classification Accuracy

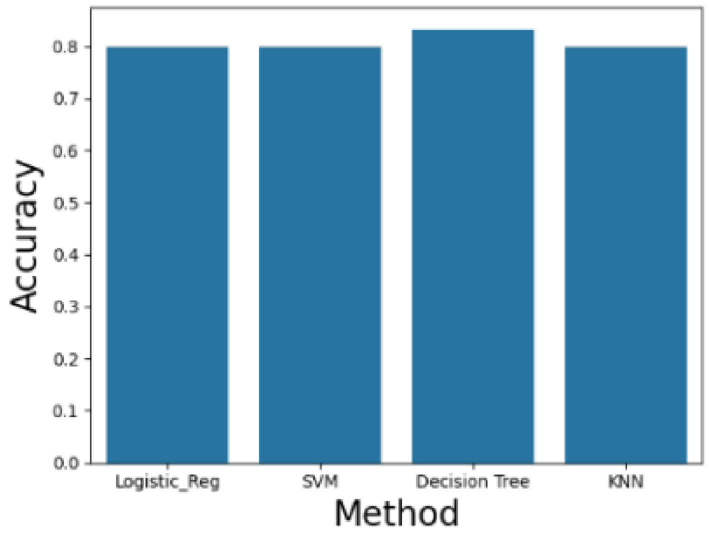
```
KNN_accuracy=knn_cv.score(X_test, Y_test)
Decision_tree_accuracy=tree_cv.score(X_test, Y_test)
SVM_accuracy=svm_cv.score(X_test, Y_test)
Logistic_Regression=logreg_cv.score(X_test, Y_test)

Report = pd.DataFrame({'Method' : ['Logistic_Reg', 'SVM', 'Decision Tree', 'KNN'],
                       'Accuracy' : [Logistic_Regression, SVM_accuracy, Decision_tree_accuracy, KNN_accuracy]})
Report
```

✓ Last executed at 2024-08-02 16:00:39 in 10ms

	Method	Accuracy
0	Logistic_Reg	0.800000
1	SVM	0.800000
2	Decision Tree	0.833333
3	KNN	0.800000

```
sns.barplot(x = 'Method', y = 'Accuracy', data = Report)
plt.xlabel("Method",fontsize=20)
plt.ylabel("Accuracy",fontsize=20)
plt.show()
```

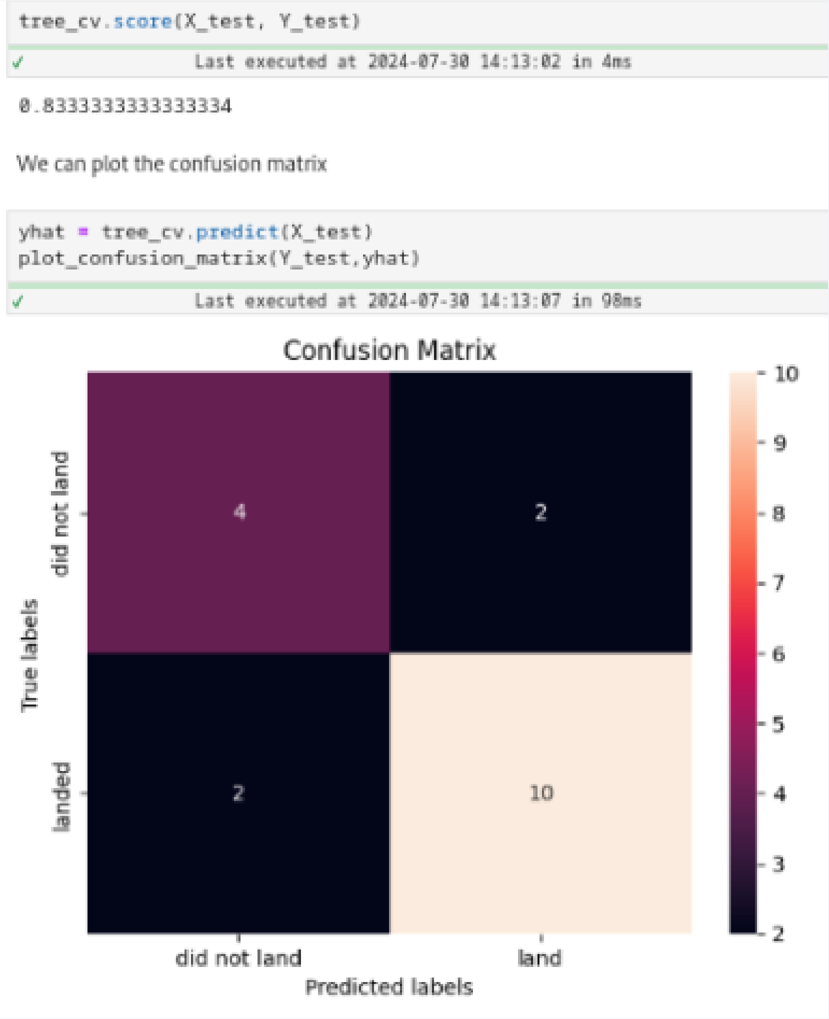
✓ Last executed at 2024-08-02 16:00:13 in 71ms



The method with the highest rate of prediction in the test data is the decision tree classifier with a rate of ~83%. The other classifiers are slightly less accurate at 80%.

Confusion Matrix

The confusion matrix for the decision tree shown right shows the breakdown of the model accuracy with true positive (bottom right), false positive (top right), true negative (top left) and false negative (bottom left) values shown in summary.



Conclusions

- Initial launch success rates were low but have increased over time, with launch success rates sitting at 100% after flight number 80.
- The lower successful launch rates at higher payloads suggest that it may be beneficial to have more launches at lower weights than to accept the risk of failure associated with higher payload mass. More research is warranted into the trade-offs of payload mass vs number of launches.
- Certain orbits (GEO, HEO, SSO and ES-L1) have the best success rates and if suitable, then they should be preferred.
- The decision tree classifier is the predictive model with the best accuracy of those considered.

Appendix

Related links

https://github.com/justjoshin78/Applied-Data-Science-Capstone/blob/main/dataset_part_1.csv

https://github.com/justjoshin78/Applied-Data-Science-Capstone/blob/main/dataset_part_2.csv

https://github.com/justjoshin78/Applied-Data-Science-Capstone/blob/main/dataset_part_3.csv

https://github.com/justjoshin78/Applied-Data-Science-Capstone/blob/main/spacex_launch_dash.csv

https://github.com/justjoshin78/Applied-Data-Science-Capstone/blob/main/spacex_launch_geo.csv

https://github.com/justjoshin78/Applied-Data-Science-Capstone/blob/main/spacex_web_scraped.csv

https://github.com/justjoshin78/Applied-Data-Science-Capstone/blob/main/spacex_dash_app.py

Thank you!