Winning Space Race with Data Science

Jonathan Wright
13th September 2023

Outline

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

Executive Summary

SpaceX competitor, SpaceY, has conducted a data science project investigating SpaceX's past launches in order to determine if a predictive model can be identified to help predict whether future launches will be successful, and thus, help predict how much these launches will cost.

Data on past launches has been collected via SpaceX's API and web scraped from Wikipedia, before being run through EDA using Data Visualization packages and SQL. Additional data visualization packages (Plotly, and Dash) were used to for further data visualizations.

Four classification models were investigated, which yielded similar predictive accuracies, with the Decision Tree Classifier yielding the best accuracy.

This suggests that it is possible to use prior launch data to develop a predictive model for future launches.

Introduction

SpaceX has accomplished the feat of re-using spacecraft by landing them, drastically reducing the cost of space travel. These launches are not fool-proof however, and do have a success rate based on a number of factors.

A competing rocket manufacturer wants to analyze SpaceX's past successful and failed launches to identify whether predictions can be made on the success of future launches without the use of rocket science, and thus, help predict how much future launches will cost.

- Can past launch factors be used to develop a predictive model for the success of future launches?
- What features will be required to model this?
- What factors lead to a lower launch success rate?

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - Data will be collected via open API and via web scraping Wikipedia
- Perform data wrangling
  - Data will be processed by using Pandas and NumPy Python packages
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - 4 Different classification models will be investigated and compared to identify the best performing model
    - Logistic Regression, SVM, KNN, and Decision Tree Classifier

Data Collection

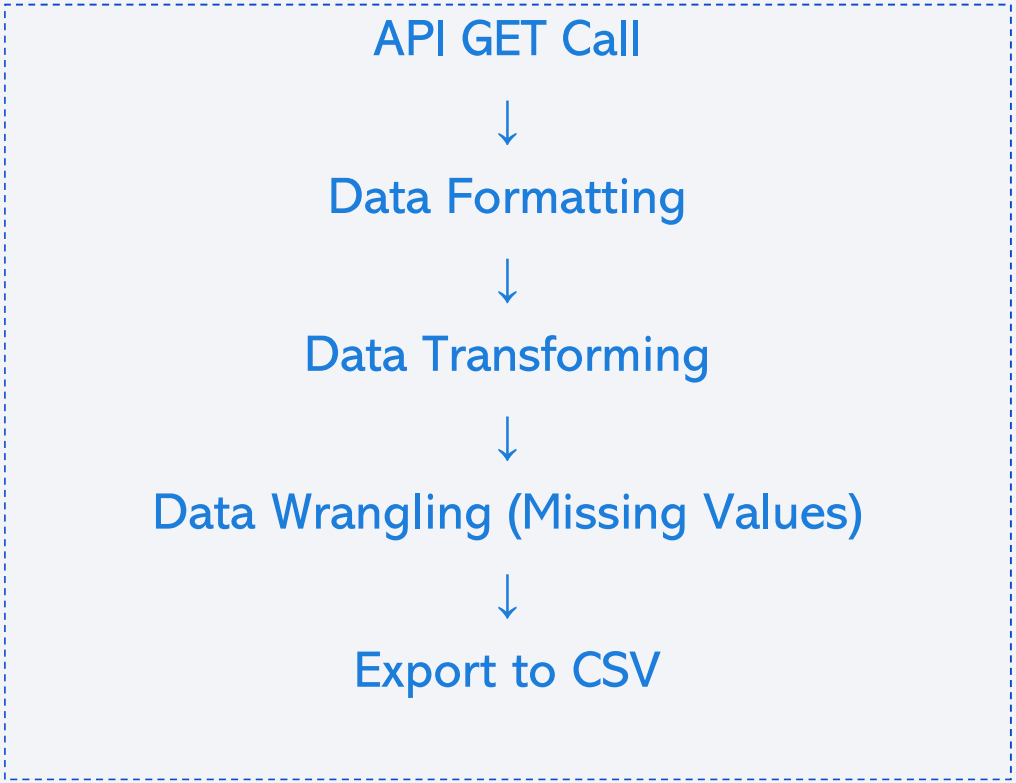
Data was collected from 2 key sources:

- SpaceX API URL
  - Collected via Python Requests package
- Web Scraped from Wikipedia
  - Collected via Python Requests package
  - Parsed using BeautifulSoup

Data Collection – SpaceX API

- The requests module in Python is used to get a JSON response from the SpaceX API URL, which is parsed to a Pandas DataFrame using the json_normalize() function.
- Formatting is then performed on the data (Subset of data before 13th November 2020, and only data with single cores and payloads).
- Pre-defined functions are then used to append key data to new lists, which are then used to build a new dataset called launch_dict.
- This new dataset is then filtered to focus on only Falcon 9 launches
- Data wrangling is then used to deal with missing values, which are replaced by the column means before the data is exported to a csv

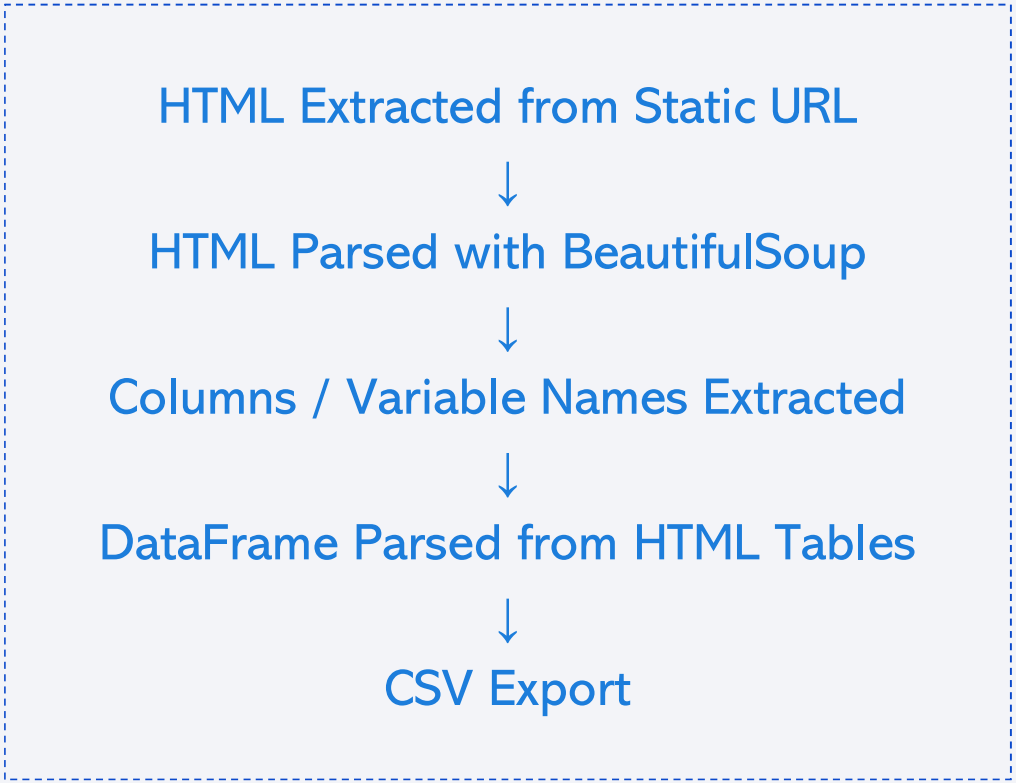
- https://github.com/jonwright13/ibm-applied-data-science-capstone/blob/main/Week%201%20-%20Lab%201%20-%20Collecting%20the%20data.ipynb



Data Collection - Scraping

- The response package in Python is used to retrieve the HTML text from a static URL, which is then parsed using the BeautifulSoup package.
- Columns and variables names are then extracted from the HTML table header.
- A Pandas DataFrame is then created by parsing the HTML tables before it is exported to a CSV.

- https://github.com/jonwright13/ibm-applied-data-science-capstone/blob/main/Week%201%20-%20Lab%202%20-%20Webscraping%20from%20Wikipedia.ipynb



Data Wrangling

- Data is analyzed through several methods to:
  - Calculate the number of launches per site
  - Calculate the number of an occurrence of each orbit
  - Calculate the number of and occurrence of each mission outcome per orbit
- The categorical column ‘class’ is converted into numerical values to represent good and bad outcomes as 1 and 0 respectively.
- Data is exported to a CSV
- https://github.com/jonwright13/ibm-applied-data-science-capstone/blob/main/Week%201%20-%20Lab%203%20-%20Data%20Wrangling.ipynb

EDA with Data Visualization

List of Charts Plotted and Why:

- **Scatter Chart:**
  - Task 1: To visualize the relationship between Flight Number and launch site
  - Task 2: To visualize the relationship between payload and launch site
  - Task 4: To visualize the relationship between Flight Number and Orbit type
  - Task 5: To visualise the relationship between payload and orbit type
- **Bar Chart:**
  - Task 3: To visualize the relationship between success rate and orbit type
- **Line Chart:**
  - Task 6: To visualize the launch success yearly trend

- https://github.com/jonwright13/ibm-applied-data-science-capstone/blob/main/Week%202%20-%20Lab%202%20-%20EDA%20with%20Pandas%20%26%20Visualisations.ipynb

EDA with SQL

- The selection of SQL queries performed on the data are:

1. Creating a table in the database for the SpaceX data and populating records where date is not null
2. Displaying unique (Distinct) launch sites
3. Displaying only 5 records for a specific launch site
4. Display total (Sum) payload mass carried by a specific customer
5. Display the average payload mass by a specific booster version
6. List the date of the first successful landing outcome in ground pad
7. List the names of the boosters, which have seen success in the drone ship launch site, with a specific payload mass range
8. List the total number of successful and failed mission outcomes
9. List the booster versions, which have carried the max payload mass (Sub-Query)
10. List records contained failed landing outcomes at drone ship, along with specific columns and parsed dates
11. Rank the landing outcomes between specific dates in descending order

- https://github.com/jonwright13/ibm-applied-data-science-capstone/blob/main/Week%202%20-%20Lab%201%20-%20SQL%20Exploratory%20Data%20Analysis.ipynb

Build an Interactive Map with Folium

Created Map Objects:

- Circle is created with color #d35400 around NASA Johnson Space Center
- Marker is added to NASA Johnson Space Center with an icon displaying the name
- A series of circles and markers are then added to all the launch sites to mark them on the map clearly
- A marker cluster is used to mark the location of all the individual launches with a color denoting their success (Green) or failure (Red)
- PolyLines are drawn on the map to mark/calculate distance between launch sites and specific locations (Coastline, railway station, city, highway)

https://github.com/jonwright13/ibm-applied-data-science-capstone/blob/main/Week%203%20-%20Lab%201%20-%20Launch%20Sites%20Locations%20Analysis%20with%20Folium.ipynb

Build a Dashboard with Plotly Dash

Plots/Graphs and interactions added to dashboard:

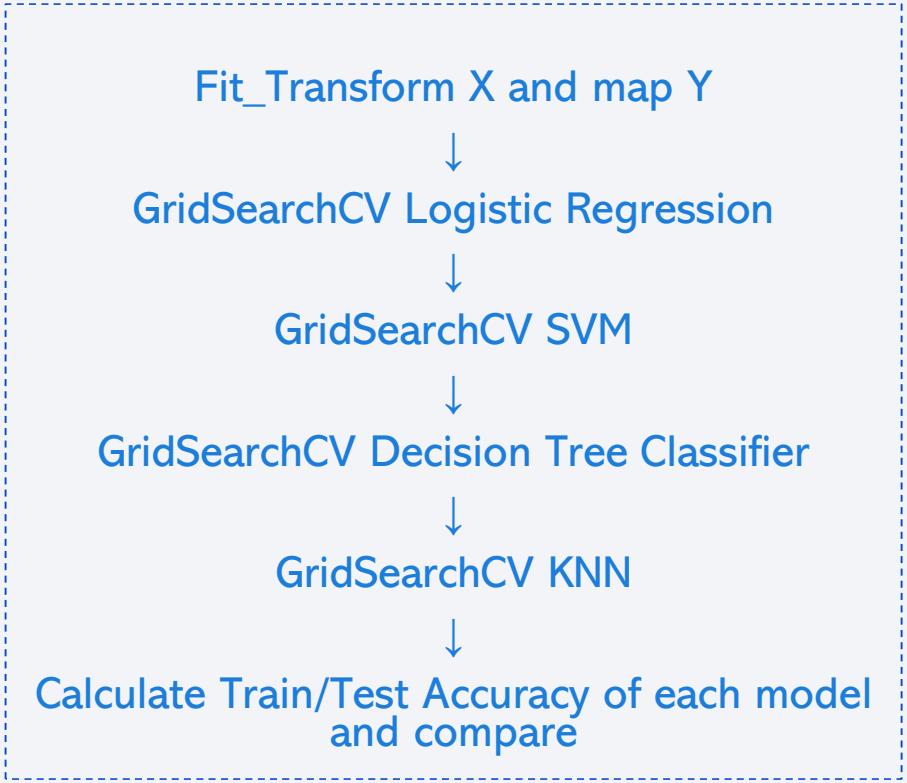
- **Pie Chart: -> To visualize categorical success/failure data**
  - Displays all launch sites and % of successful launches
  - Displays a specific launch site with 2 sections representing success/failure of launch
- **Scatter Chart: -> To visualize correlation of categorical success/failure data**
  - Displays the correlation between payload and success for all launch sites
  - Displays the correlation between payload and success for a specific launch site

- https://github.com/jonwright13/ibm-applied-data-science-capstone/blob/main/Week%203%20-%20Web%20App%20Dashboard%20with%20Dash.py

Predictive Analysis (Classification)

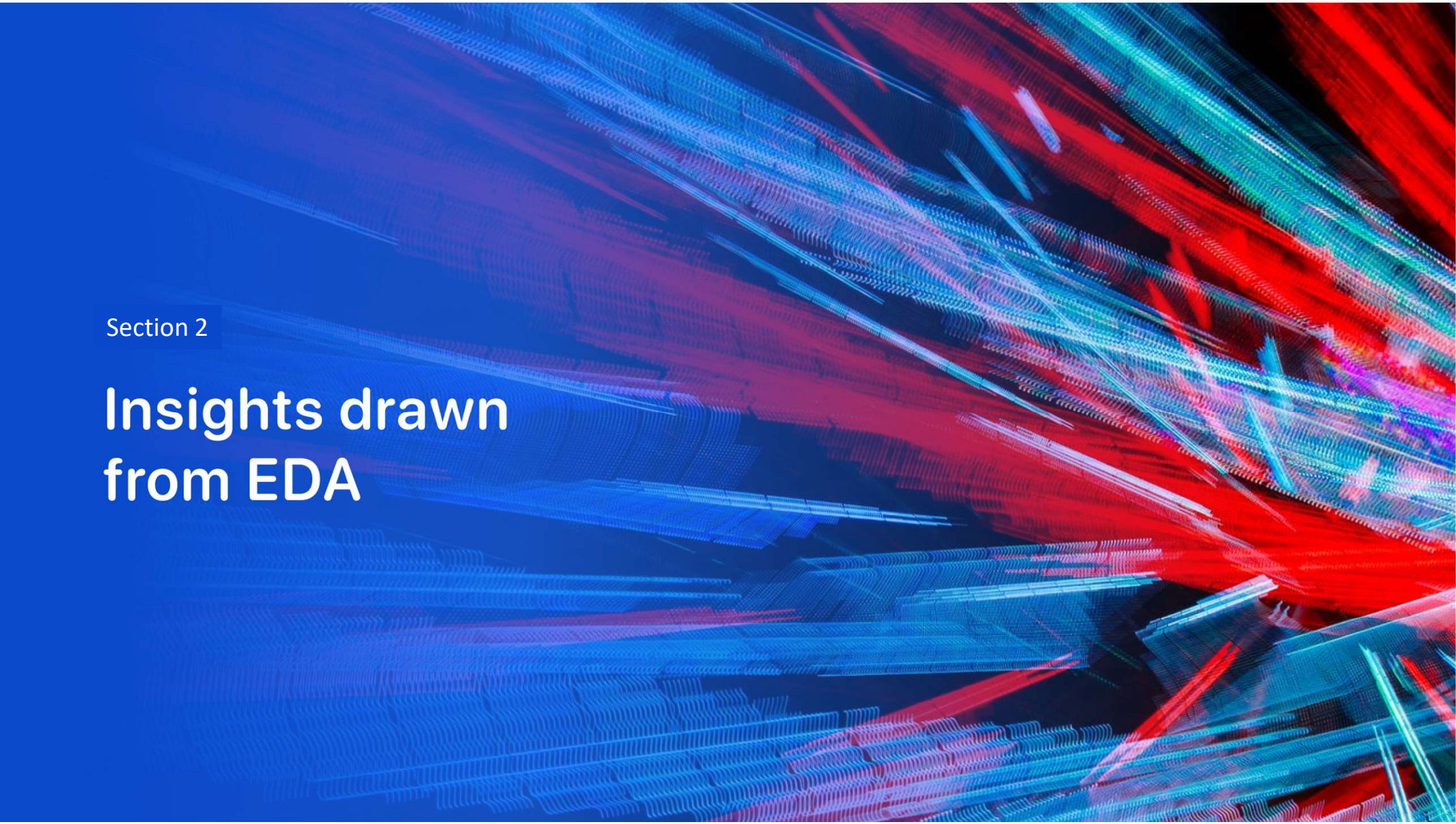
- ML Models were built by first mapping the class column to Y, and a fit_transformed features to X before being split into test/train sets
- GridSearchCV was used across several model types using pre-set parameters to train, predict, and identify tuned hyperparameters, accuracy, and a confusion matrix for each:
  - Logistic Regression, Support vector Machine (SVM), Decision Tree Classifier, and K-Nearest Neighbors Classifier
- All models performed practically the same using test data; however, the decision tree classifier had the best accuracy on the training data, at 87.7% while the others had slightly below this.

https://github.com/jonwright13/ibm-applied-data-science-capstone/blob/main/Week%204%20-%20Lab%201%20-%20Building%20The%20Models.ipynb



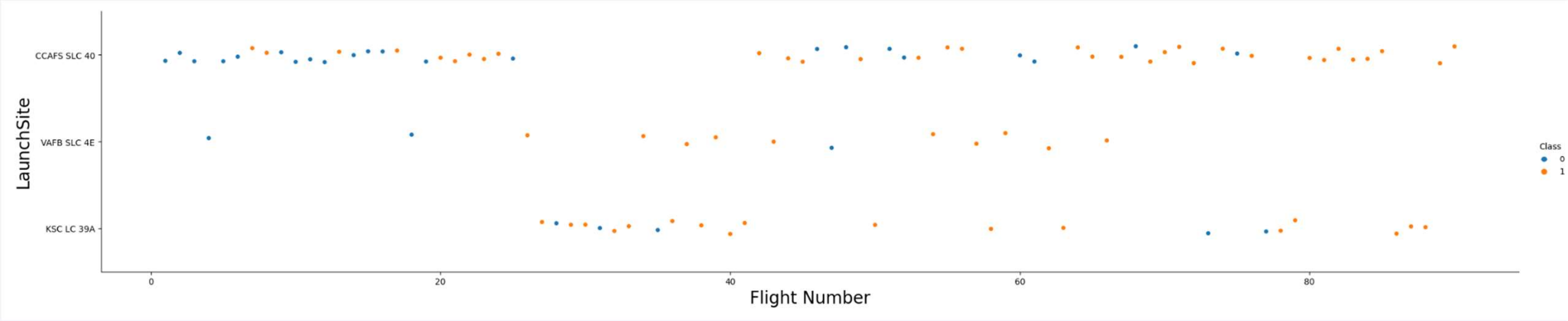
Results

- Launch success has been increasing steadily year on year since start in 2013 regardless of other factors
- Launch site KSC LC-39A has the highest number of successful launches compared to all other sites (Nearly 50 of all successful launches came from here), while site CCAFS SLC-40 has the lowest number of successful launches
  - Launch site KSC LC-39A has a 77% success rate, indicating a very high chance of future launches here being successful
- Orbit SSO has a 100% success rate but limited launches, while orbit GTO has a higher number of launches but does see some failures with repeated failures after a single occurrence.
  - LEO Orbit shows the best repeated success with no failures after the first 2 launches
- All 4 machine learning models discussed revealed similar test accuracies (83.33%), whilst the decision tree classifier presented a slightly higher training accuracy (87%)
  - This suggests the Tree classifier should be used to predict whether future launches are successful based on launch site, booster version, orbit, and payload


Section 2
Insights drawn
from EDA

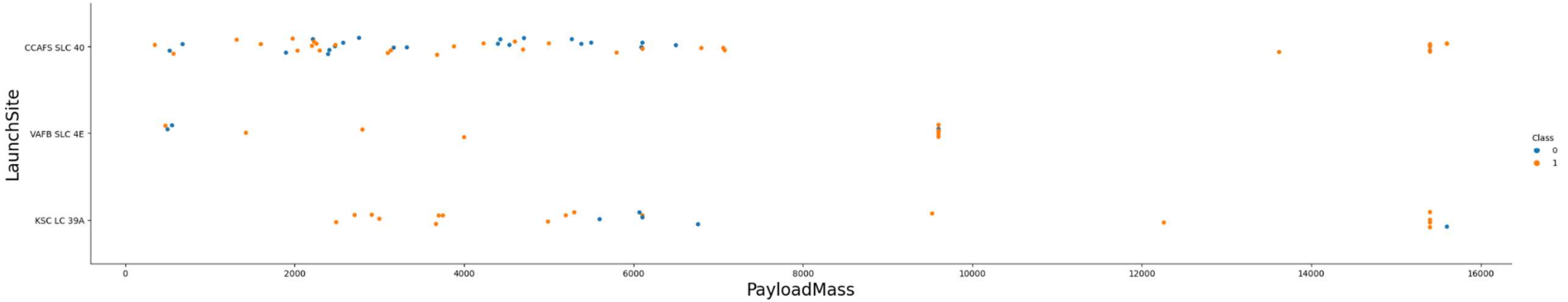
Flight Number vs. Launch Site

- Show a scatter plot of Flight Number vs. Launch Site
- Scatter plot shows successive / failed launches (Denoted by class color) for all flight numbers
- There is a lower number of failed launches (Blue) compared to Successful launches (Orange).
- A single failed launch at CCAFS SLC-40 usually follows or precedes another failed launch at this site (Not isolated), while failures at KSC LC 39A are generally moreso



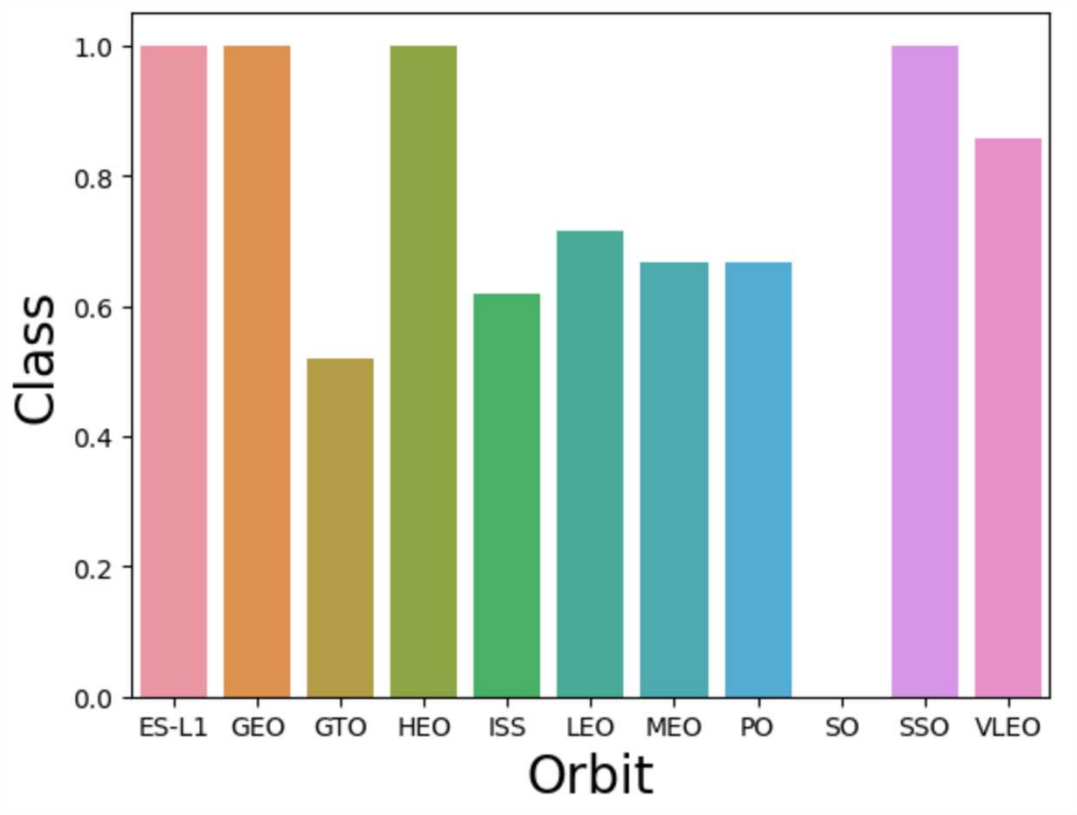
Payload vs. Launch Site

- Show a scatter plot of Payload vs. Launch Site
- Scatter plot shows that launch site VAFB SLC 4E does not accommodate payloads above 10,000kg



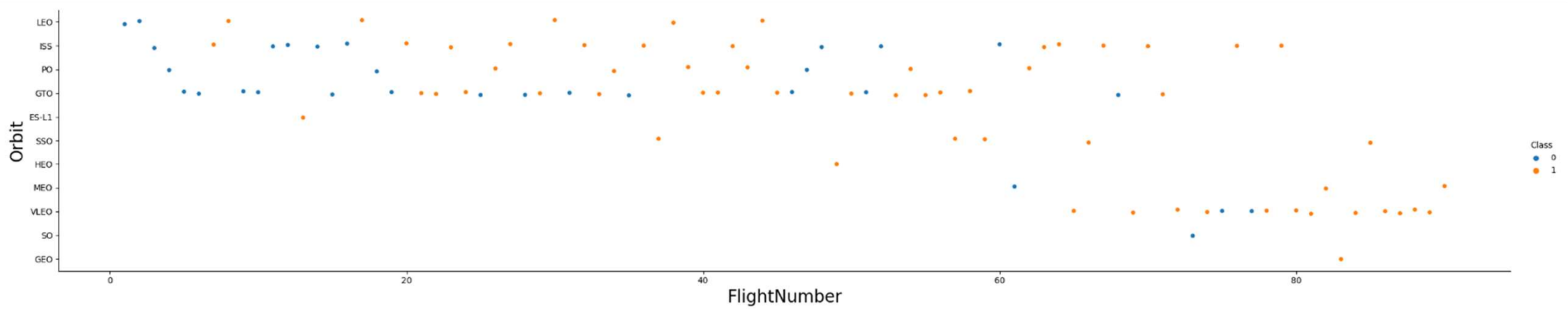
Success Rate vs. Orbit Type

- Show a bar chart for the success rate of each orbit type
- Bar chart shows that some orbits had a high success rate compared to others



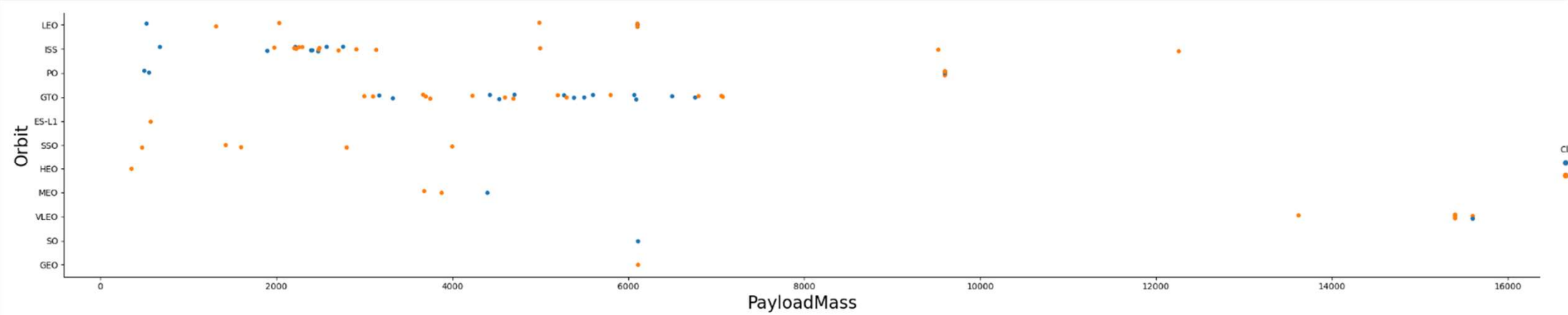
Flight Number vs. Orbit Type

- Show a scatter point of Flight number vs. Orbit type
- Scatter plot shows orbit LEO took 2 failed launches before being successful every time afterwards, while other orbits had varying successes and failures
- Orbit SSO showed all successes but with limited frequency



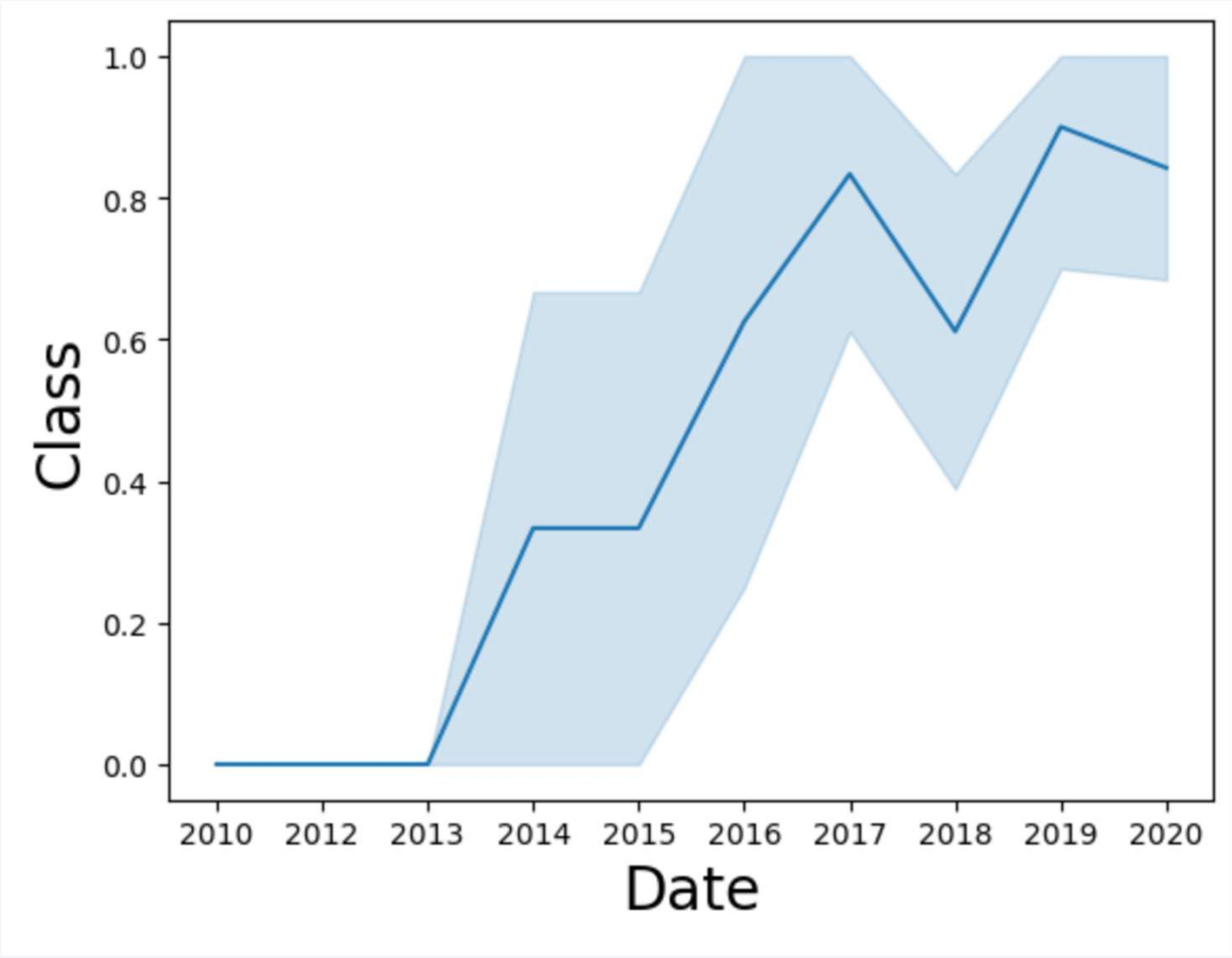
Payload vs. Orbit Type

- Show a scatter point of payload vs. orbit type
- Scatter chart shows a high frequency of low payloads across most orbits, while higher payloads are less frequent and only used in specific orbits



Launch Success Yearly Trend

- Show a line chart of yearly average success rate
- Line chart shows that successful launches increase year on year up till 2020



All Launch Site Names

Find the names of the unique launch sites

- SQL SELECT statement used to return distinct launch site names from table

```
%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'

Find 5 records where launch sites begin with `CCA`

- SQL SELECT statement used to return all columns from table WHERE the string “%CCA%” occurs in the launch site name, limited to the first 5 records

```
%sql SELECT * FROM SPACEXTBL WHERE Launch_Site LIKE "CCA%" LIMIT 5
```

```
* 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)
2012-05-22	07:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success	No attempt
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

Total Payload Mass

Calculate the total payload carried by boosters from NASA

- SQL SELECT statement used to sum up the payload columns with a specific cutomer (NASA)

```
%sql SELECT SUM(PAYLOAD_MASS__KG_) as Total_Payload_Mass_kg FROM SPACEXTBL WHERE Customer = "NASA (CRS)"
```

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

Total_Payload_Mass_kg
45596

Average Payload Mass by F9 v1.1

Calculate the average payload mass carried by booster version F9 v1.1

- SQL SELECT statement used to identify the average payload mass, achieved using a WHERE predicate

```
%sql SELECT AVG(PAYLOAD_MASS__KG_) as AVERAGE_PAYLOAD FROM SPACEXTBL WHERE Booster_Version LIKE "F9 v1.1%"
```

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

AVERAGE_PAYLOAD
2534.6666666666665

First Successful Ground Landing Date

Find the dates of the first successful landing outcome on ground pad

- SQL SELECT statement used to find the min date with a WHERE predicate and ordered by date

```
%sql SELECT Min(Date) as First_Successful_Landing FROM SPACEXTBL WHERE Landing_Outcome = "Success (ground pad)" ORDER BY Date
```

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

First_Successful_Landing
2015-12-22

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

- SQL SELECT query displays only the distinct booster versions which had a specific payload and landing outcome, achieved using a WHERE predicate

```
%sql SELECT DISTINCT(Booster_Version) FROM SPACEXTBL WHERE (PAYLOAD_MASS__KG_ BETWEEN 4000 AND 6000) AND (Landing_Outcome = "Success (drone ship)")
```

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

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

Total Number of Successful and Failure Mission Outcomes

Calculate the total number of successful and failure mission outcomes

- SQL SELECT statement counts the number of failed or successful mission outcomes, achieved using a WHERE predicate

```
%sql SELECT COUNT(*) FROM SPACEXTBL WHERE (Mission_Outcome = "Success") OR (Mission_Outcome = "Failure")
```

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

COUNT(*)
98

Boosters Carried Maximum Payload

List the names of the booster which have carried the maximum payload mass

- SQL SELECT statement used to return only the distinct booster versions which had a max payload mass
- Sub-query used to identify the max payload mass

```
%sql SELECT DISTINCT(Booster_Version) FROM SPACEXTBL WHERE PAYLOAD_MASS__KG_ = (SELECT MAX(PAYLOAD_MASS__KG_) FROM SPACEXTBL)
```

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

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

List the failed landing_outcomes in drone ship, their booster versions, and launch site names for in year 2015

- SQL SELECT query converts date values into year and month in separate columns
- Booster version, launch site, and landing outcome are the parsed into a table, orderd by month, where the year is 2015 with a specific landing outcome

```
%%sql
SELECT Date, SUBSTR(Date,1,4) AS Year, SUBSTR(Date, 6, 2) as Month, Booster_Version, Launch_Site, Landing_Outcome
FROM SPACEXTBL
WHERE (SUBSTR(Date,1,4)='2015') AND (Landing_Outcome = "Failure (drone ship)")
ORDER BY Month
```

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

Date	Year	Month	Booster_Version	Launch_Site	Landing_Outcome
2015-04-14	2015	04	F9 v1.1 B1015	CCAFS LC-40	Failure (drone ship)
2015-10-01	2015	10	F9 v1.1 B1012	CCAFS LC-40	Failure (drone ship)

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

- SQL SELECT query that presents 2 columns showing all the distinct landing outcomes and their rank in descending order
- Where clause used to only specify landing outcomes between specific dates

```
%%sql
SELECT Landing_Outcome, RANK() OVER (ORDER BY COUNT(Landing_Outcome)) AS RANK
FROM SPACEXTBL
WHERE Date BETWEEN "2010-06-04" AND "2017-03-20"
GROUP BY Landing_Outcome
ORDER BY RANK DESC
```

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

Landing_Outcome	RANK
No attempt	8
Failure (drone ship)	5
Success (drone ship)	5
Success (ground pad)	5
Controlled (ocean)	4
Uncontrolled (ocean)	3
Failure (parachute)	1
Precluded (drone ship)	1

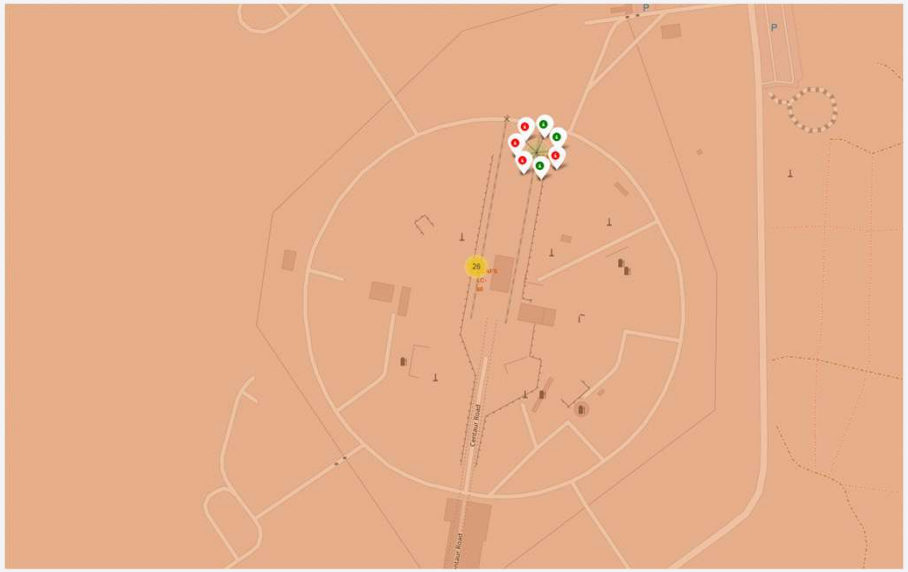

Section 3
Launch Sites
Proximities Analysis

Folium Map of All SpaceX Launch Sites

- Zoomed out Folium Map showing the entirety of the USA with orange markers denoting the launch sites in Florida and California

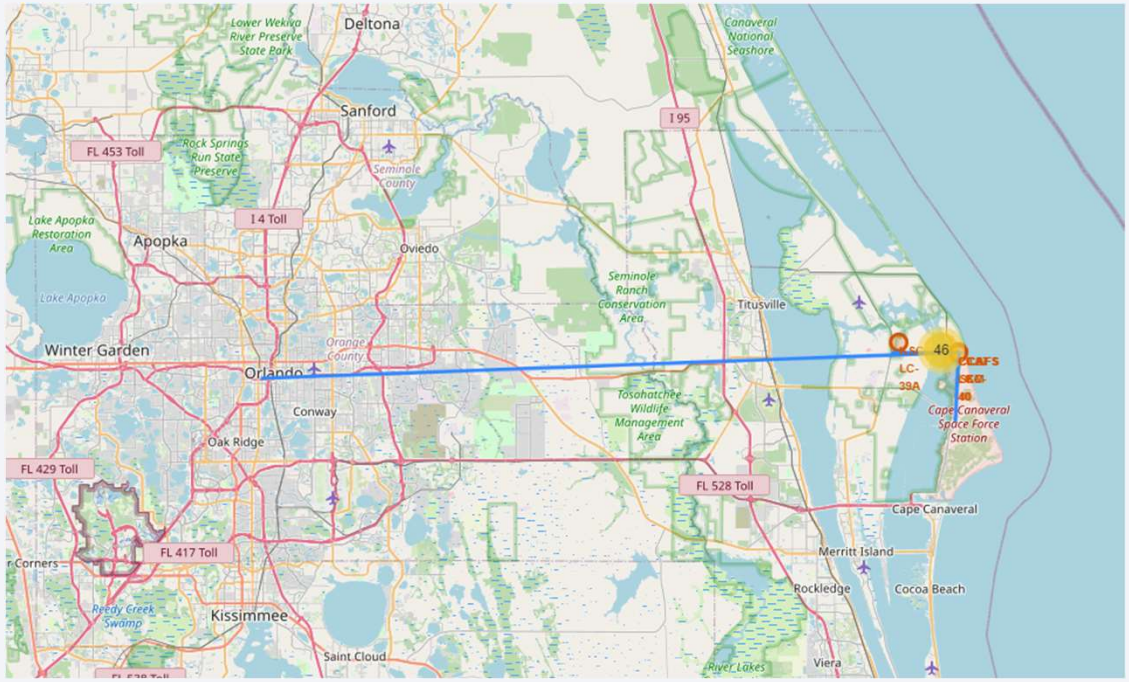
Folium Map of Success/Failed Launches at Site CCAFS SLC-40

- Zoomed in Folium map showing launch site CCAFS SLC-40 with individual markers (Marker Cluster) showing each launch with a color denoting the class (Success or failure in green / red respectively)

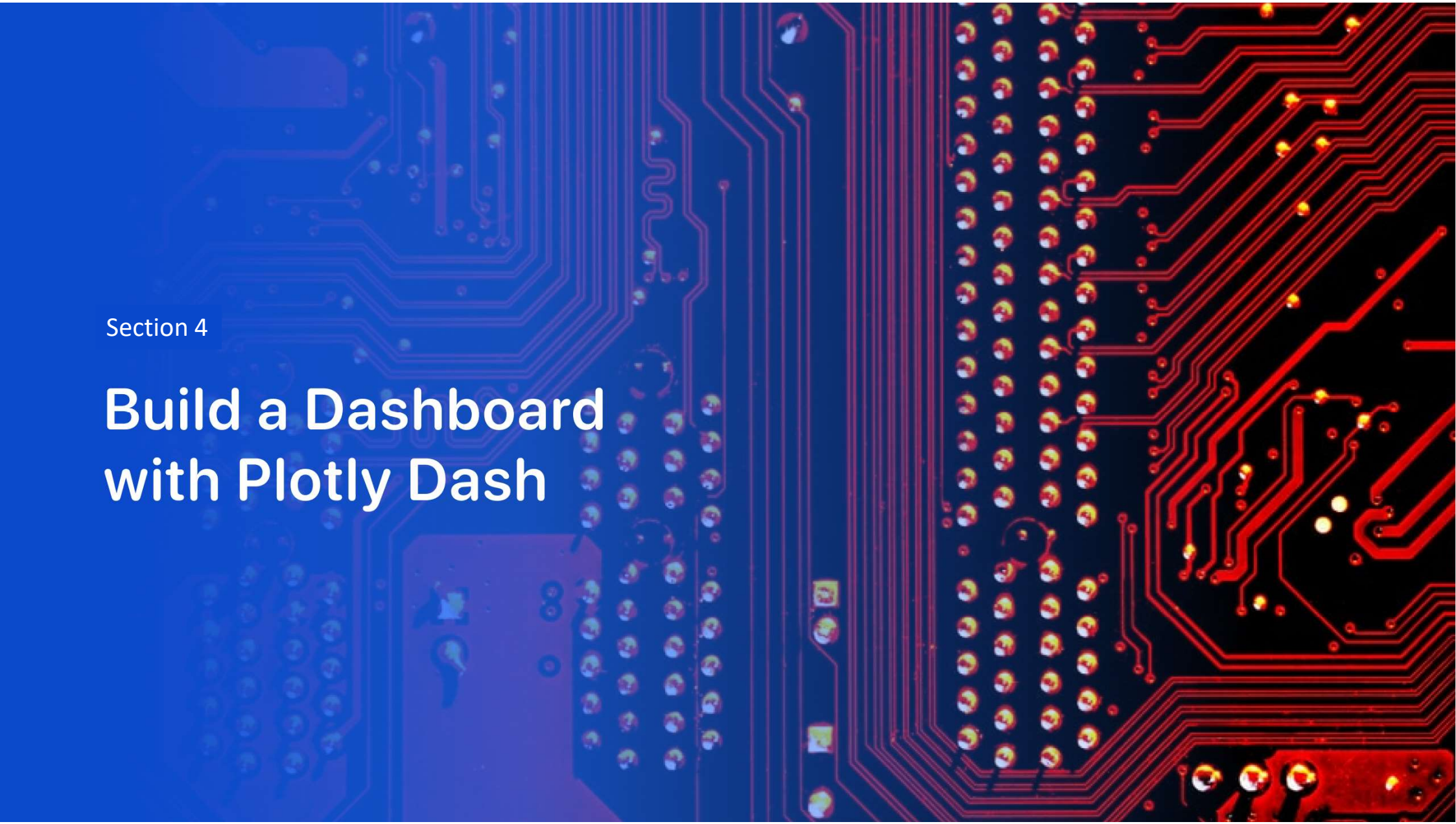


Folium Map of Launch Site CCAFS SLC-40 & Proximity to Infrastructure

- Folium map to right showing proximity to launch site CCAFS SLC-40 from key infrastructure denoted with a blue PolyLine (City of Orlando, Railway Station, Highway, and Coastline)
- Numerical distances shown below

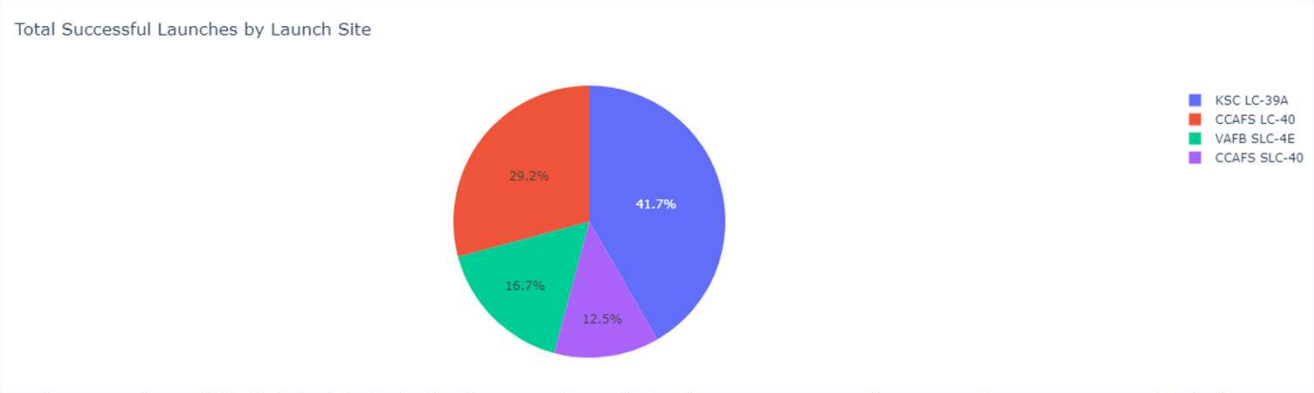



```
Distance between launch site and nearest Coastline:          0.87km
Distance between launch site and nearest Railway Station:    7.54km
Distance between launch site and nearest City Center:        79.72km
Distance between launch site and nearest Highway:            7.81km
```


Section 4
Build a Dashboard with Plotly Dash

Dashboard Launch Success Count Pie Chart

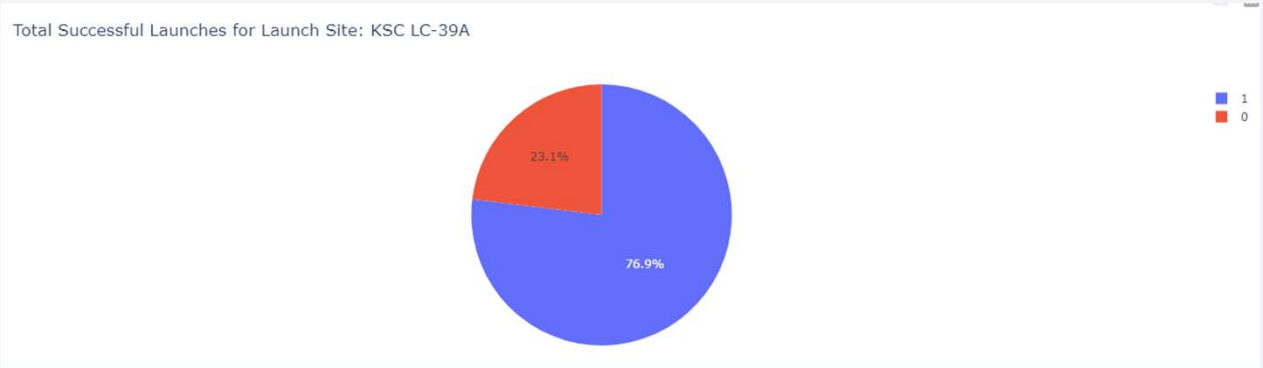
- Show the screenshot of launch success count for all sites, in a piechart



- Chart shows that site KSC LC-39A has the highest number of successful launches with nearly 50% of all successful launches coming from this site compared to the others, with CCAFS LC-40 coming in second.
- Site CCAFS SLC-40 has the lowest number of successful launches

Dashboard Pie Chart of Site KSC LC-39A

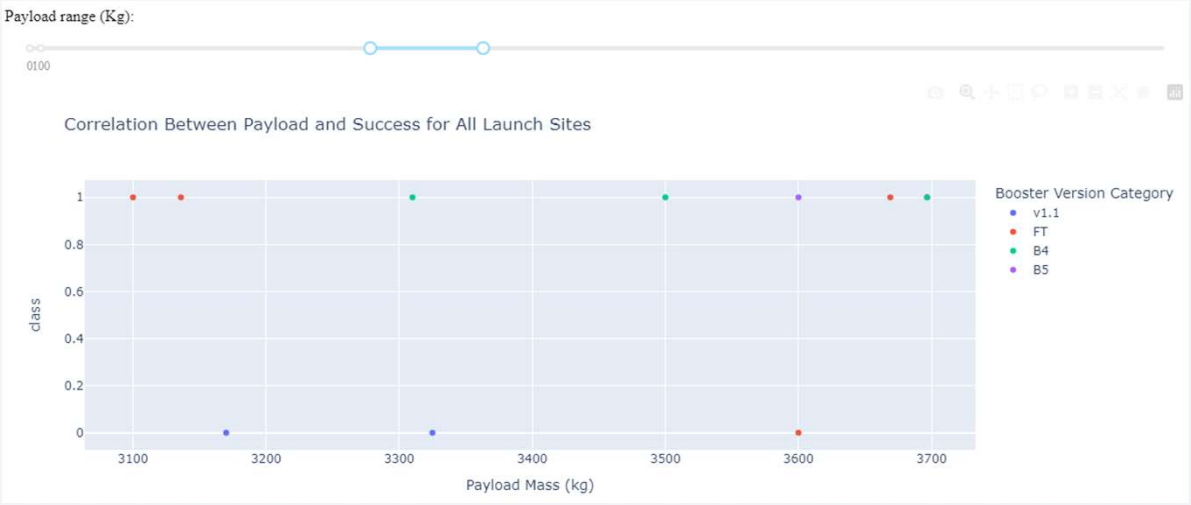
- Show the screenshot of the piechart for the launch site with highest launch success ratio



- Site KSC LC-39A showed the highest number of successful launches compared to all other sites, and when selected, shows that over ¾ of launches there were successful

Dashboard Correlation Between Payload & Success

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



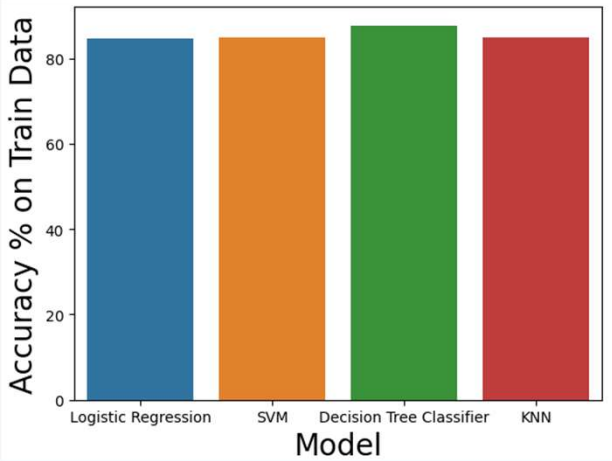
- Scatter plot shows booster versions, success/failures, and payloads between 3000kgand 4000kg.
- In this range, booster category had the best successful launches within this payload range, with B4 coming in second

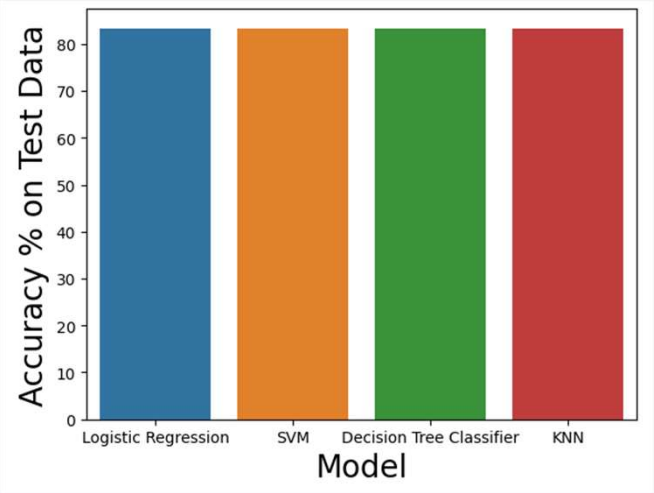
Section 5

Predictive Analysis (Classification)

Classification Accuracy

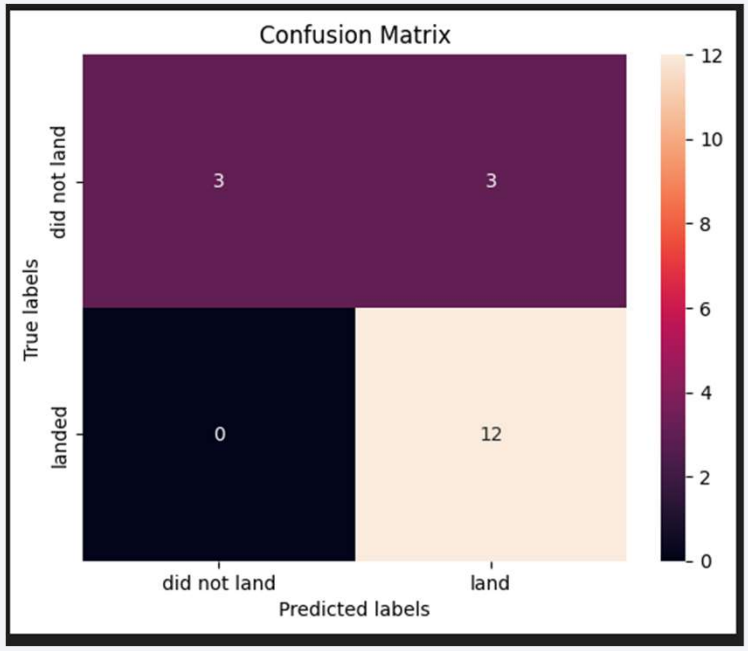
- Visualize the built model accuracy for all built classification models, in a bar chart
  - Accuracy of all 4 models displayed on right for both test and train data in bar charts
- Find which model has the highest classification accuracy
  - All 4 models perform the same with test data, but the decision tree classifier has a slightly higher accuracy on training data than other models





Confusion Matrix

- Confusion Matrix for the Decision Tree Classifier Model
- All models performed similarly with test data, but only the Tree model showed improvements compared with other models for the training data

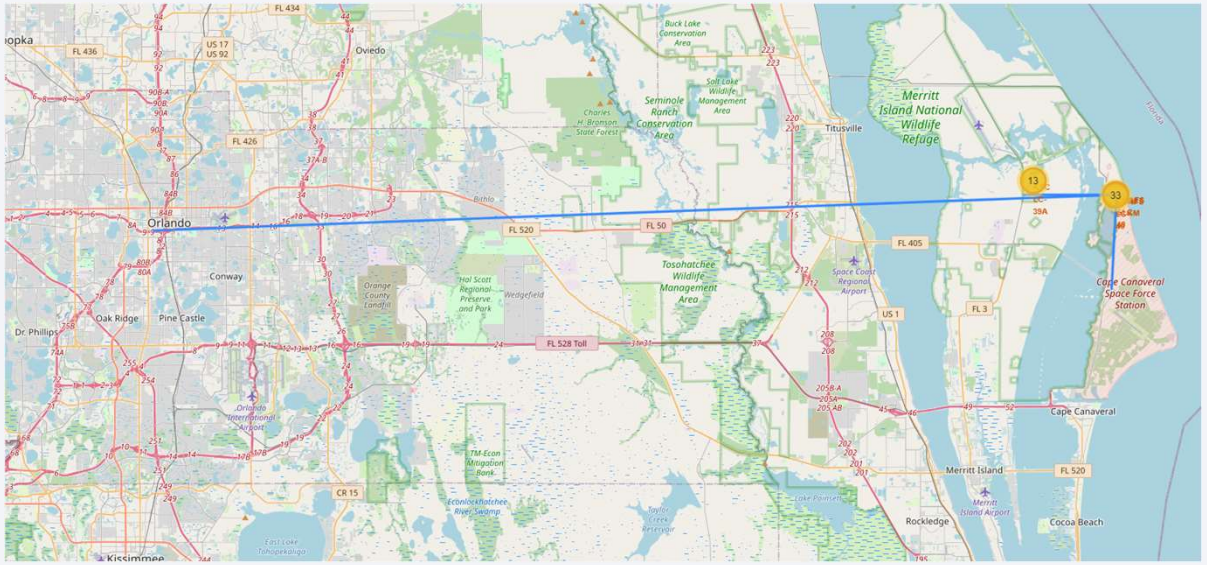


Conclusions

- Launch success has been increasing steadily year on year since start in 2013 regardless of other factors
- Launch site KSC LC-39A has the highest number of successful launches compared to all other sites (Nearly 50 of all successful launches came from here), while site CCAFS SLC-40 has the lowest number of successful launches
  - Launch site KSC LC-39A has a 77% success rate, indicating a very high chance of future launches here being successful
- Orbit SSO has a 100% success rate but limited launches, while orbit GTO has a higher number of launches but does see some failures with repeated failures after a single occurrence.
  - LEO Orbit shows the best repeated success with no failures after the first 2 launches
- All 4 machine learning models discussed revealed similar test accuracies (83.33%), whilst the decision tree classifier presented a slightly higher training accuracy (87%)
  - This suggests the Tree classifier should be used to predict whether future launches are successful based on launch site, booster version, orbit, and payload

Appendix

- Launch sites KSC LC-39A and CCAFS SLC-40 are both very close together but have distinctly different success rates, possibly due to proximity to coastline
- Wind flow from the sea could be a factor explaining the differences, as KSC LC-39A is more in-land and shielded, whilst CCAFS SLC-40 is less than 1km from the coast



Thank you!