IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Aayush D. Gandhi
28-04-2024

Outline

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

Executive Summary

We here studied SpaceX Falcon 9 launch data in order to predict whether the rocket's first stage will land successfully.

Several Machine Learning classification algorithms were used, following the following steps:

1. Data Collection, Wrangling, and Formatting
2. Exploratory Data Analysis
3. Interactive Data Visualisation
4. Predictive Machine Learning

Main findings:

- Many features of the Falcon 9 Launch influence the outcome of the landing

Introduction

- Project background and context
  - Space industry is becoming more mainstream, thanks to private companies entering the market. Analysis of launch costs and parameters influencing mission success are vital to keep this market viable.
  - SpaceX developed a rocket with an unique re-usable first stage (RFS), cutting the costs of launches by more than 100 million USD when compared to competitors.
  - Optimising the success rate of RFS preservation after launches is essential to maintain this financial edge.
- Problems you want to find answers
  - Determine successful landing of the RFS
  - Assess the impact of various factors on landing outcome
  - Analyse correlations between launch sites and landing outcome

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - SpaceX API
  - Web Scrapping data from Wilipedia pages
- Perform data wrangling
  - Supervised models were trained after converting mission outcomes (0: Failure, 1: successful)
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - Created a column for 'class'
  - Standardised and transformed data
  - Plit dataset in train/test data
  - Determination of best classification algorithm using test data

Data Collection

- Data was collected using the SpaceX API and by web scraping the Wikipedia page listing all Falcon 9 (heavy) launches.

API approach

API request: request launch data using SpaceX API → **Read API response:** Convert .json file to a DataFrame → **Data Wrangling:** Extract relevant data to match requirements → **Convert to CSV:** Convert new dataframe to CSV for further processing

Web scraping approach

Perform HTML GET: request Wikipedia HTML page through HTML GET → **Create BeautifulSoup:** Use HTML response to generate a BeautifulSoup Object → **Data Extraction:** Collect relevant data and headers for DataFrame → **Convert to CSV :** Convert new dataframe to CSV for further processing

Data Collection – SpaceX API

SpaceX REST API

URLs of rockets, launchpads,payloads, Cores ans past launch

GET request

Requests.get(URL)

Receive response

Response.status_code=200

Convert .json to DF

pd.DataFrame([launch_dictl)

data=pd.json_normalize(response_json_file)
launch_dict = {'FlightNumber': list(data['flight_number'l)

Filtering Data

Filter data to include only Falcon 9 launches:

GitHub: [Data Collection using API]

Data Collection - Scraping

Wikipedia page

"List of Falcon 9 and Falcon Heavy launches"

Static_url = https://en.wikipedia.org/w/index.php?title=List_of_Falcon_9_and_Falcon_Heavy_launches&oldi=1027686922

GET request to obtain data:

html_data = requests.get(Static_url)

Create object from response with BeautifulSoup:

soup = BeautifulSoup(html_data.text, 'html.parser')

Find HTML table with Falcon 9 data:

html_tables =soup.find_all('table')

Iterate through <th> elements to extract column names

```
column_names = []
html_th=first_launch_table.find_all('th')
for i in html_th:
    name=extract_column_from_header(i)
    if name is not None and len(name)>0:
        column_names.append(name)
```

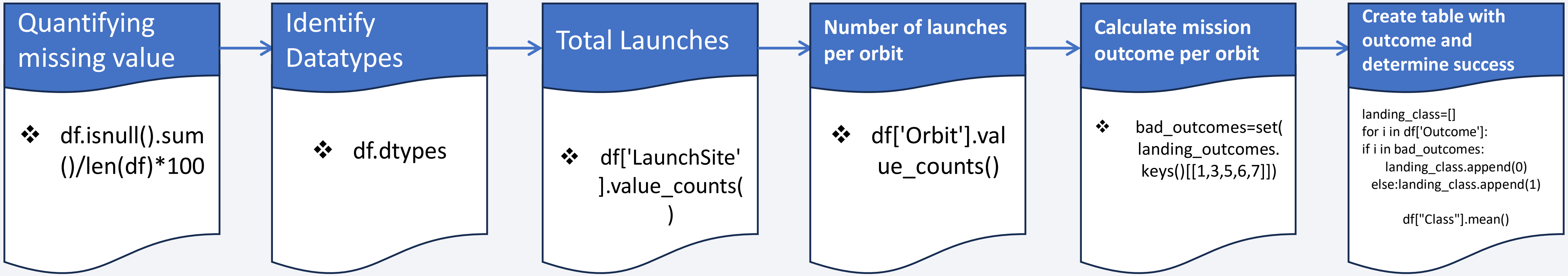
Iterate through <th> elements to extract column names

```
launch_dict= dict.fromkeys(column_names)

# Remove an irrelvant column
del launch_dict['Date and time ( )']

# Let's initial the launch_dict with each value to be an empty list
launch_dict['Flight No.'] = []
launch_dict['Launch site'] = []
launch_dict['Payload'] = []
launch_dict['Payload mass'] = []
launch_dict['Orbit'] = []
launch_dict['Customer'] = []
launch_dict['Launch outcome'] = []
# Added some new columns
launch_dict['Version Booster']=[]
launch_dict['Booster landing']=[]
launch_dict['Date']=[]
launch_dict['Time']=[]
```


GitHub : [Data Collection using web scrapping]

Data Wrangling

- Data was first assessed by identifying missing points, data type, and count
- Then, data was split per orbit type and mission outcome per orbit was calculated
- Data was subsequently put into a table, and success rate per orbit type was calculates



GitHub : Data Wrangling

EDA with Data Visualization

Plot types used:

- Scatter plots:
  - To identify relationships between two variables
    - ➢ Flight Number vs. Payload
    - ➢ Flight Number vs. Launch Site
    - ➢ Payload vs. Launch Site
    - ➢ Class vs. Orbit
    - ➢ Flight Number vs. Orbit
    - ➢ Payload vs. Orbit
- Bar charts:
  - To compare values between two groups, often used to compare a variable at a given point in time
    - ➢ Success Rate per Orbit
- Line charts:
  - To track changes over time
    - ➢ Success Rate over Time

GitHub : URL

EDA with SQL

- **Identify unique Launch Sites**:
  - *%sql SELECT DISTINCT "Launch_Site" FROM SPACEXTABLE;*
- **Retrieve 5 records with Launch Site name beginning with CCA:**
  - *%sql SELECT "Launch_Site" FROM SPACEXTABLE WHERE "Launch_Site" LIKE 'CCA%" LIMIT 5;*
- **Find total Payload Mass carried by NASA-launched boosters (CRS):**
  - *%sql SELECT customer, AS total_payload FROM SPACEXTABLE WHERE (CRS)" GROUP BY customer;*
- **Retrieve average Payload Mass carried by F9 VI .1 boosters :**
  - *%sql SELECT "Booster_version", AS avg_payload FROM SPACEXTABLE WHERE vl.l" GROUP BY "Booster_version";*
- **List date with first successful Landing Outcome:**
  - *%sql SELECT "Landing_Outcome", MIN("Date")AS min_date FROM SPACEXTABLE WHERE "Landing_Outcome"="Success (ground pad)" GROUP BY 'Landing_outcome";*
- **List Boosters with success in Drone Ship and a Payload Mass >4000 and <6000:**
  - *%sql SELECT DISTINCT "Booster_Version" FROM SPACEXTABLE WHERE "Landing_Outcome"="Success (drone ship)" AND (PAYLOAD_MASS KG_>4000 AND PAYLOAD_MASS KG_<6000);*
- **List total number of successful and failure Mission Outcomes:**
  - *%sql SELECT "Mission_outcome", AS FROM SPACEXTABLE GROUP BY*
- **Find names of Booster Versions which carried the max. Payload Mass (using a subquery):**
  - *%sql SELECT WHERE PAYLOAD_MASS*
- **Retrieve records, displaying Month Name, Landing Outcome, and Launch Site for missions in 2015:**
  - *%sql SELECT AS year, AS month, "Landing_outcome", "Booster_version", "Launch_Site" FROM SPACEXTABLE WHERE "Landing_outcome"='Failure (drone ship)" AND substr("Date",0,5)='2015*
- **Rank the count of Landing Outcomes between 2010-06-04 and 2017-03-20:**
  - *%sql SELECT AS count_landing_outcomes FROM SPACEXTABLE WHERE ("Landing_Outcome" = "Failure (drone ship)" OR "Landing_Outcome"=Success (ground pad) ") AND ("Date" BETWEEN "2010-06-04" AND "2017-03-20")*

GitHub : URL

Build an Interactive Map with Folium

Folium is a Python library allowing creation of interactive maps

- We created a Folium map containing:
    - Circles to highlight Launch Sites
    - Markers to indicate Mission Outcome
    - Indicator of Mouse Pointer position
- Additionally, we used lines and markers to calculate the distance of Launch Sites and:
    - Railways
    - Highways
    - Coastlines
    - Cities

GitHub : Folium Map

Build a Dashboard with Plotly Dash

We built a Plotly Dash dashboard to visualise data in a real-time, interactive manner

Visualisations used:

- Pie chart to visualise contributions
    - Total Mission Success Counts per Launch Site
    - Identify Launch Site with highest Success Ratio
- Scatter plots to visualise relationships between variables:
    - Mission Outcome vs Payload Mass

GitHub : Dashboard

Predictive Analysis (Classification)

Predictive analysis was used to predict the Mission Outcome

Dependent Variable:

- Class (Mission Outcome; 1= success, 0 = fail)

Excluded from Independent Variables:

- Date, Outcome, Booster Version, Longitude, Latitude

One-hot encoding of categorical Independent Variables:

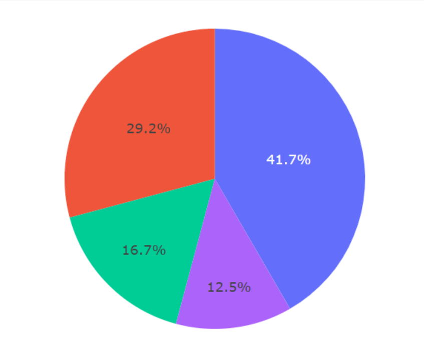
- Orbit, Launch Site, Landing Pad, Serial. Grid Fins, Reused, Legs

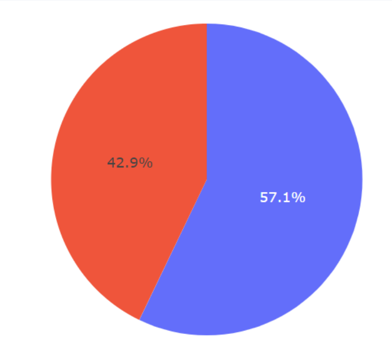
- Data was standardised, split into train and test data, and various classification methods were assessed.
- Optimal Hyperparameters per Model and the best performing Model were identified

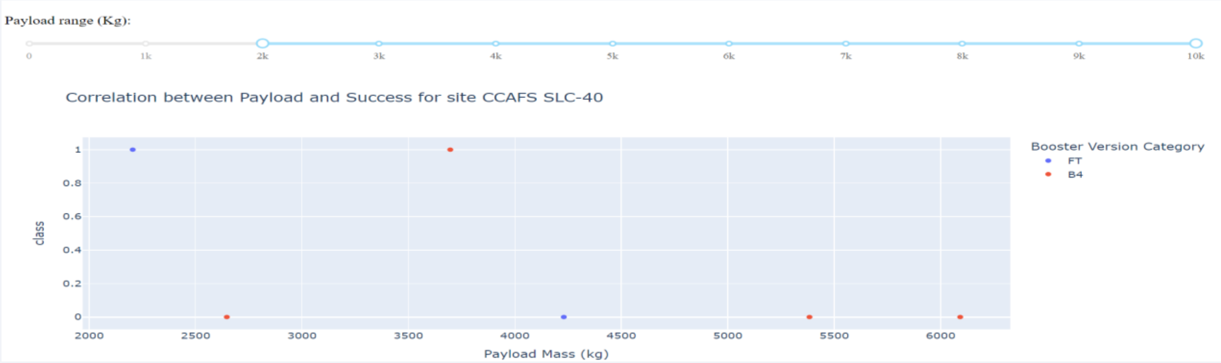
GitHub : URL

Results

- Exploratory data analysis results
  - Relevant parameters influencing Mission Outcome were identified
- Interactive analytics demo in screenshots







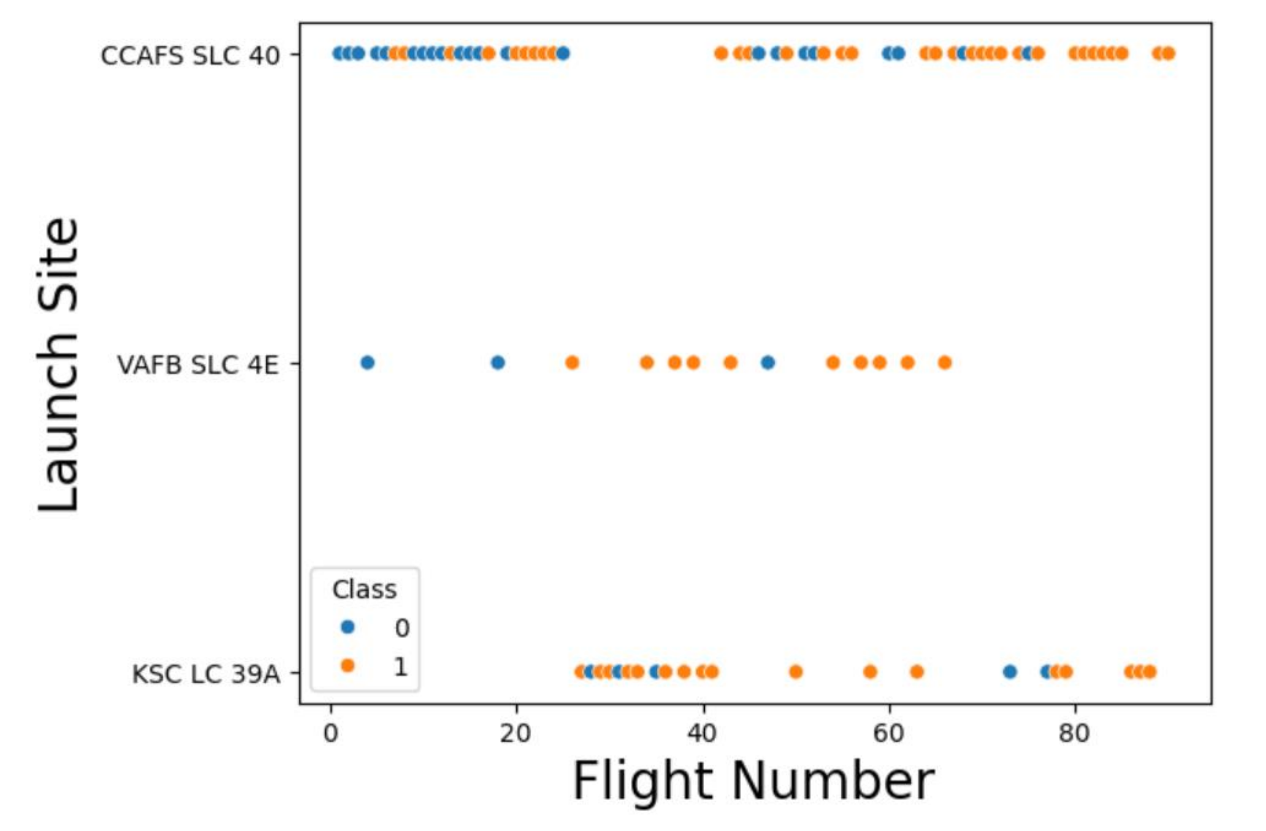
- Predictive analysis results

	Model Name	Score
0	Logistic Regression	0.833333
1	SVM	0.833333
2	Decision Tree	0.722222
3	K Nearest Neighboors	0.833333

Section 2

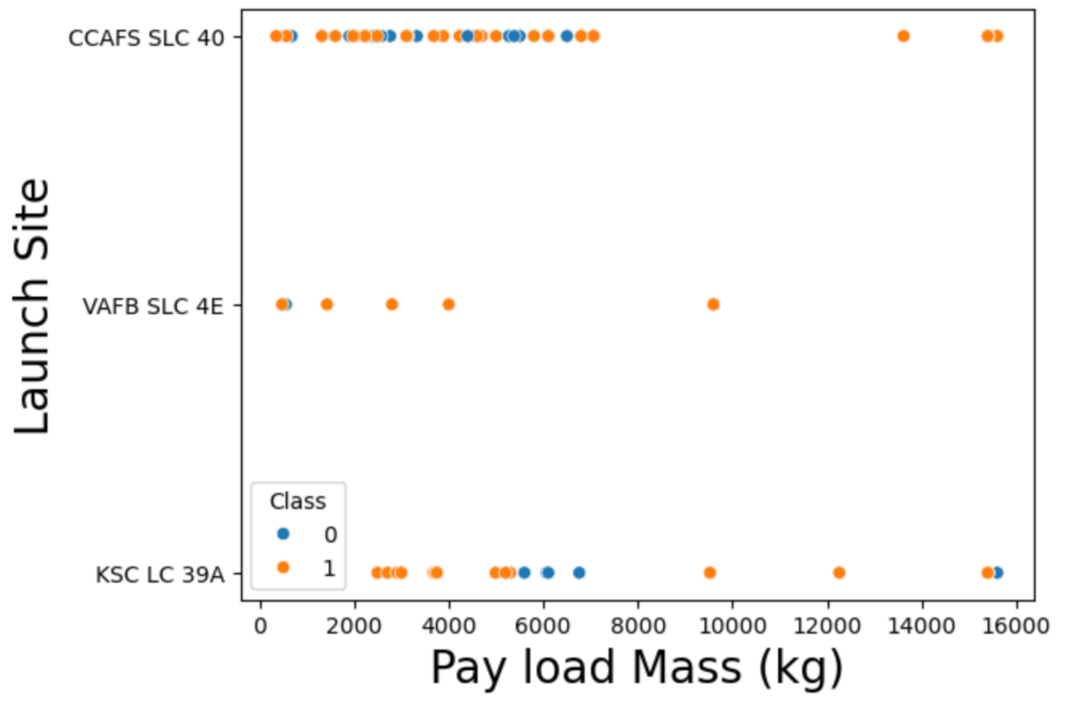
Insights drawn from EDA

Flight Number vs. Launch Site



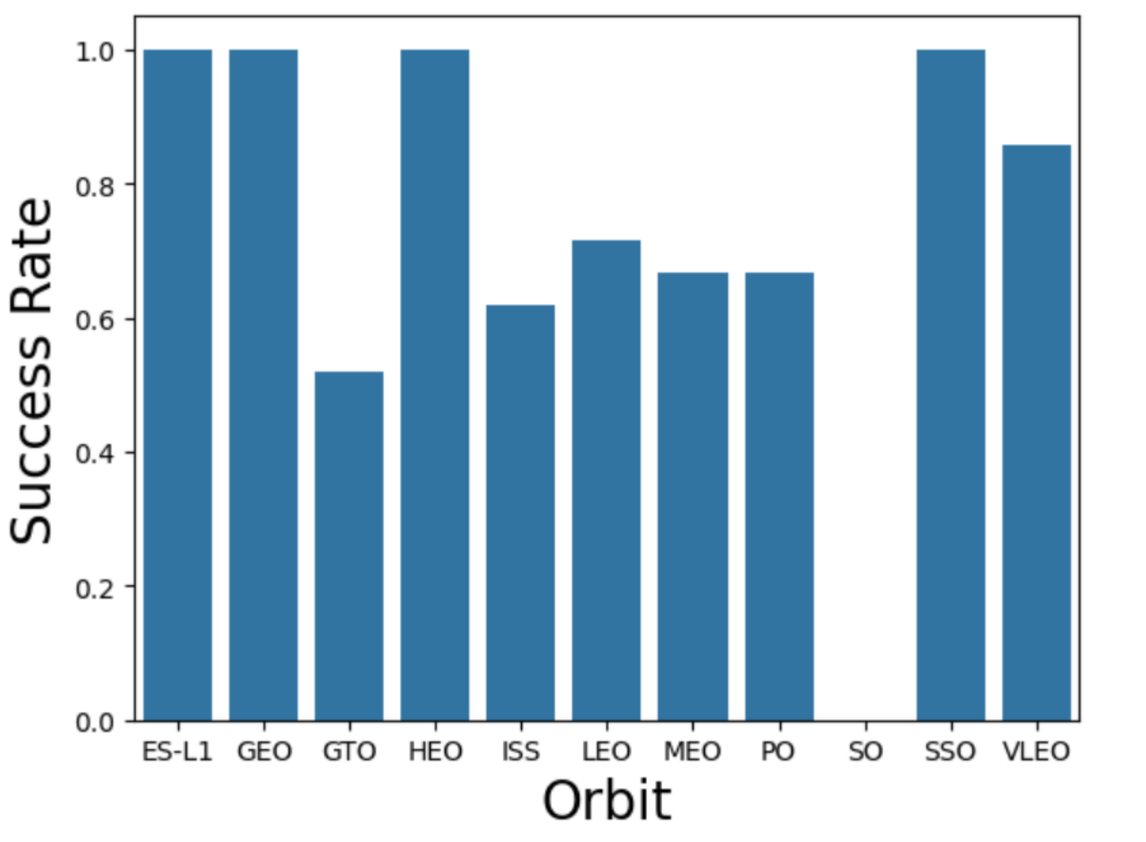
- Most initial missions were unsuccessful (colored blue)
- As flight number increases, the mission is more likely to be successful (colored orange)

Payload vs. Launch Site



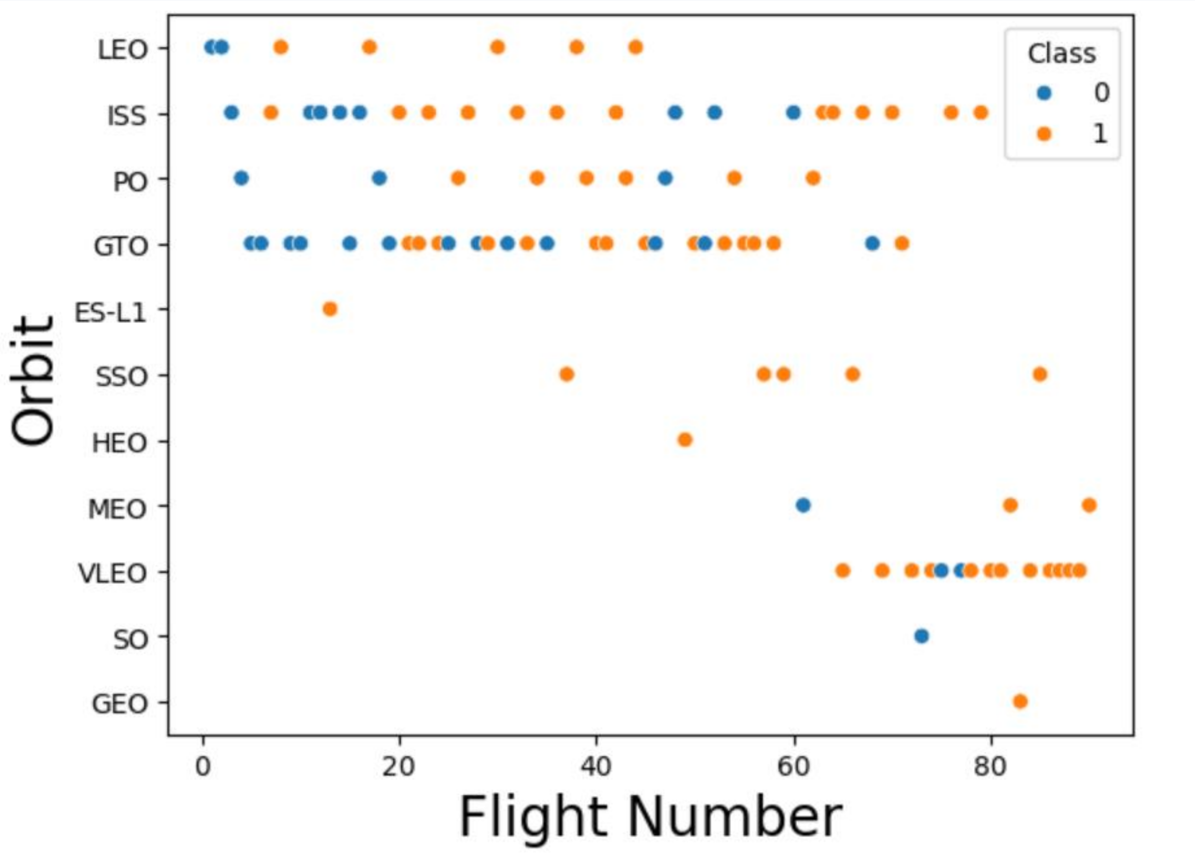
- At Launch Site 'CCAFS-SLC' (Caraveral Center), results for lighter Payloads (<8,000 kg) are mixed, but heavier Payloads are generally successful
- At Launch Site 'VAFB-SLC' (Vandenberg), the maximum Payload launched was 10,000 kg, and Missions were mostly successful
- At Launch Site 'KSC-LC' (Kennedy Space Center), most Missions are successful, apart from Missions with a Payload of around 6,000 kg

Success Rate vs. Orbit Type



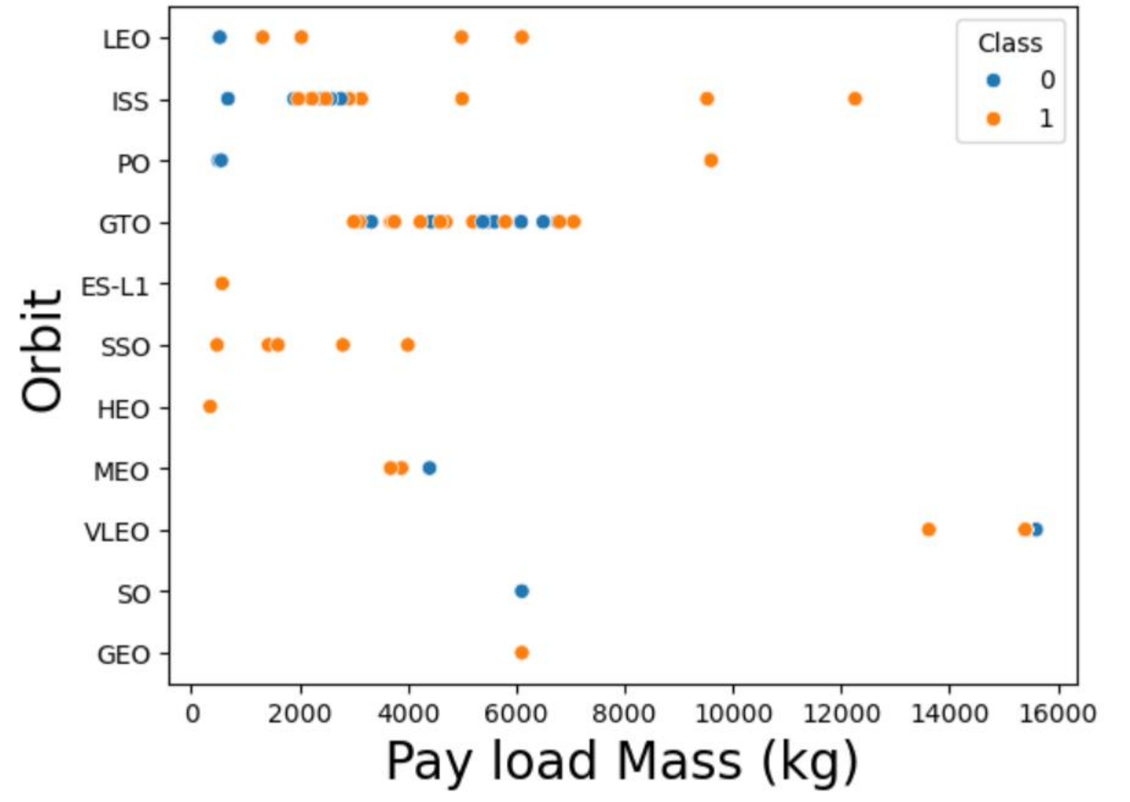
- Orbits ES-LI ,GEO, HEO, and SSO have a perfect success rate of 1
- VLEO also has a good success rate
- GTO, ISS, LEO, MEO, and PO have a mixed success rate
- All missions to orbit SO failed

Flight Number vs. Orbit Type



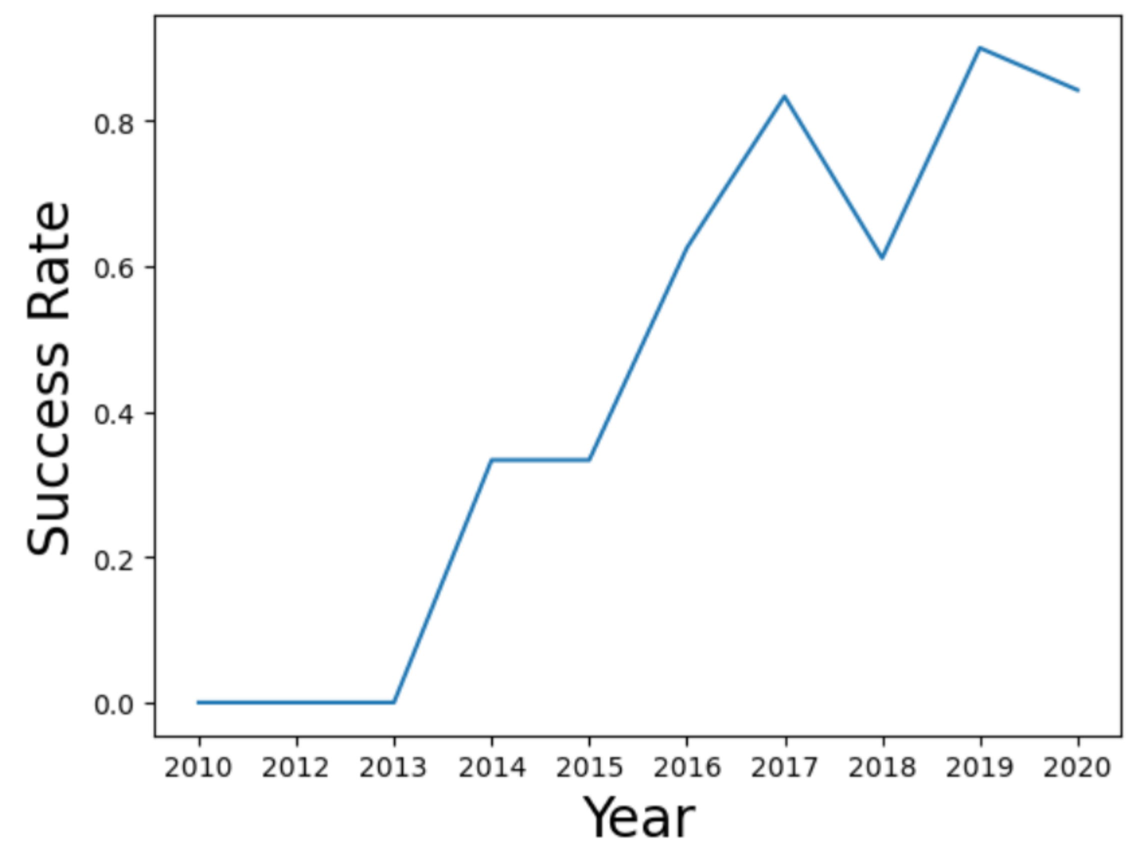
- Overall, flights with a low Flight Number are more often unsuccessful (colored blue)
- After two failed Missions, all following Missions to LEO were successful (colored orange)
- ISS displays a cluster of failed Missions around Flight Number 50
- For GTO, there is no clear relationship between Flight Number and Mission Outcome
- The 0% Success Rate for SO is due to only one Mission being attempted

Payload vs. Orbit Type



- Heavy Payloads (>8 000 kg) were only deployed to orbits ISS, PO, and VLEO.
- Missions with a heavy Payload were generally successful (coloured orange)
- Missions with a very light Payload (<IOOO kg) are likely to be unsuccessful (coloured blue)
  - This is especially true for LEO, ISS, and PO
  - ES-LI , SSO, and HEO appear more suitable for very light Payloads
- For orbit GTO, there is no clear relationship between Payload and Mission Outcome

Launch Success Yearly Trend



- All Missions until 2013 were unsuccessful
- Mission success rate was the same in 2013 and 2015
- After 201 5, Mission success rate started increasing, with the exemption of 2018

All Launch Site Names

```
%sql SELECT DISTINCT(Launch_Site) from SPACEXTABLE
 * sqlite:///my_data1.db
Done.
```

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

"DISTINCT" only returns unique values"

Four Unique Launch Sites:

- CCAFS LC: Cape Canaveral Launch Complex
- VAFB SLC: Vandeberg space Force Base
- KSC LC: Kennedy Space Centre
- CCAFS SLC: Cape Canaveral Space Launch Complex

Launch Site Names Begin with 'CCA'

```
%sql SELECT Launch_Site FROM SPACEXTABLE WHERE Launch_Site like "CCA%"
```

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

Launch_Site
CCAFS LC-40
CCAFS LC-40
CCAFS LC-40
CCAFS LC-40
CCAFS LC-40

Explanation:
WHERE "Launch_Site" LIKE "CCA %" — to return Launch Sites starting with 'CCA’
LIMIT 5 — to return on/y five records

All returned values are CCAFS LC: Cape Canaveral Launch Complex

Total Payload Mass

```
%sql SELECT SUM(PAYLOAD_MASS__KG_) from SPACEXTABLE where Customer= "NASA (CRS)"
```

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

SUM(PAYLOAD_MASS__KG_)
45596

Full query:

- %sql SELECT SUM(PAYLOAD_MASS__KG_) from SPACEXTABLE where Customer= "NASA (CRS)"

Explanation:

- SUM(PAYLOAD_MASS KG_) to extract total payload
- where Customer= "NASA (CRS)" to return NASA records

Total Payload mass carried by boosters launched by NASA was 45,496 kg

Average Payload Mass by F9 v1.1

```
%sql SELECT AVG(PAYLOAD_MASS__KG_) FROM SPACEXTABLE WHERE Booster_Version = "F9 v1.1"
```

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

AVG(PAYLOAD_MASS__KG_)
2928.4

Full query:

- %sql SELECT AVG(PAYLOAD_MASS__KG_) FROM SPACEXTABLE WHERE Booster_Version = “F9 v1.1”

Explanation:

- AVG(PAYLOAD_MASS KG_) to extract average payload
- where Booster_Version = “F9 v1.1” to extract records with version “F9 v1.1”

Average Payload mass carried by F9 v 1.1 boosters was 2,928.4 kg

First Successful Ground Landing Date

```
%sql SELECT MIN(Date) FROM SPACEXTABLE WHERE Landing_Outcome ="Success (ground pad)"
```

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

MIN(Date)
2015-12-22

Full query:

- %sql SELECT MIN(Date) FROM SPACEXTABLE WHERE Landing_Outcome ="Success (ground pad)"

Explanation:

- MIN(Date) to extract first Date
- where Landing_Outcome ="Success (ground pad)" to return only records with successful ground landing

The first successful Ground Landing was on 2015-12-22 (22 December 2015)

Successful Drone Ship Landing with Payload between 4000 and 6000

```
%sql SELECT Booster_Version FROM SPACEXTABLE WHERE Landing_Outcome ="Success (drone ship)" and PAYLOAD_MA
```

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

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

Full query:

- %sql SELECT Booster_Version FROM SPACEXTABLE WHERE Landing_Outcome ="Success (drone ship)" and PAYLOAD_MASS__KG_ between 4000 and 6000

Explanation:

- SELECT Booster_Version – to return Booster Versions
- WHERE Landing_Outcome ="Success (drone ship)" and PAYLOAD_MASS__KG_ between 4000 and 6000 – to return only records with successful Drone Ship landing and specified Payload Mass

Booster versions F9 FT-B1022, -B1026, -B1021.2, and -B1031.2 fulfilled the specified requirements

Total Number of Successful and Failure Mission Outcomes

```
%sql SELECT "Mission_Outcome", COUNT(*) AS count_misssion_outcome FROM SPACEXTABLE GROUP BY TRIM("Mission_Outcome
```

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

Mission_Outcome	count_misssion_outcome
Failure (in flight)	1
Success	99
Success (payload status unclear)	1

Full query:
%sql SELECT “Mission_Outcome” , COUNT(*) AS count_mission_outcome FROM SPACEXTABLE GROUP BY TRIM("Mission_ Outcome ");

Explanation:
SELECT “Mission_Outcome”, COUNT(*) AS count_misssion_outcome — to return count of Missions
GROUP BY ("Mission_ Outcome"); — to group results by Mission Outcome

In total, 1 Mission failed, 1 was successful but Payload status was unclear, and 99 Missions were successful

Boosters Carried Maximum Payload

```
%sql SELECT Booster_Version from SPACEXTABLE where PAYLOAD_MASS__KG_== (SELECT MAX(PAYLOAD_MASS__KG_) from
```

```
 * 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

Full query:
%sql SELECT Booster_Version from SPACEXTABLE where PAYLOAD_MASS__KG_== (SELECT MAX(PAYLOAD_MASS__KG_) from SPACEXTABLE)

Explanation:
- SELECT "Booster Version",— to return count of Booster Version
- where PAYLOAD_MASS__KG_== (SELECT MAX(PAYLOAD_MASS__KG_) from SPACEXTABLE)
— to return on/y records with maximum Payload mass

The Booster versions listed in the table all carried the Maximum Payload Mass

2015 Launch Records

```
%sql SELECT substr(Date, 6,2) as month, Landing_Outcome , Booster_Version, Launch_Site from SPACEXTABLE
```

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

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

Full query:
%sql 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’
Explanation:
- SELECT substr(Date, 6,2) as month, Landing_Outcome , Booster_Version, Launch_Site – returns month, Landing_Outcome , Booster_Version, Launch_Site
- WHERE Landing_Outcome ="Failure (drone ship)" and substr(Date,0,5)='2015’ – returns only values where landing failed on a Drone Ship and Launch Year was 2015

Two Missions where the Landing Site was a Drone Ship failed in 2015, one in January and the other in April

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

```
%sql SELECT Landing_Outcome,count(Landing_Outcome) from SPACEXTABLE where Date between '2010-06-04' and
```

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

Landing_Outcome	count(Landing_Outcome)
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

Full query:
%sql SELECT Landing_Outcome,count(Landing_Outcome) from SPACEXTABLE where Date between '2010-06-04' and '2017-03-20' group by Landing_Outcome order by count(Landing_Outcome) desc

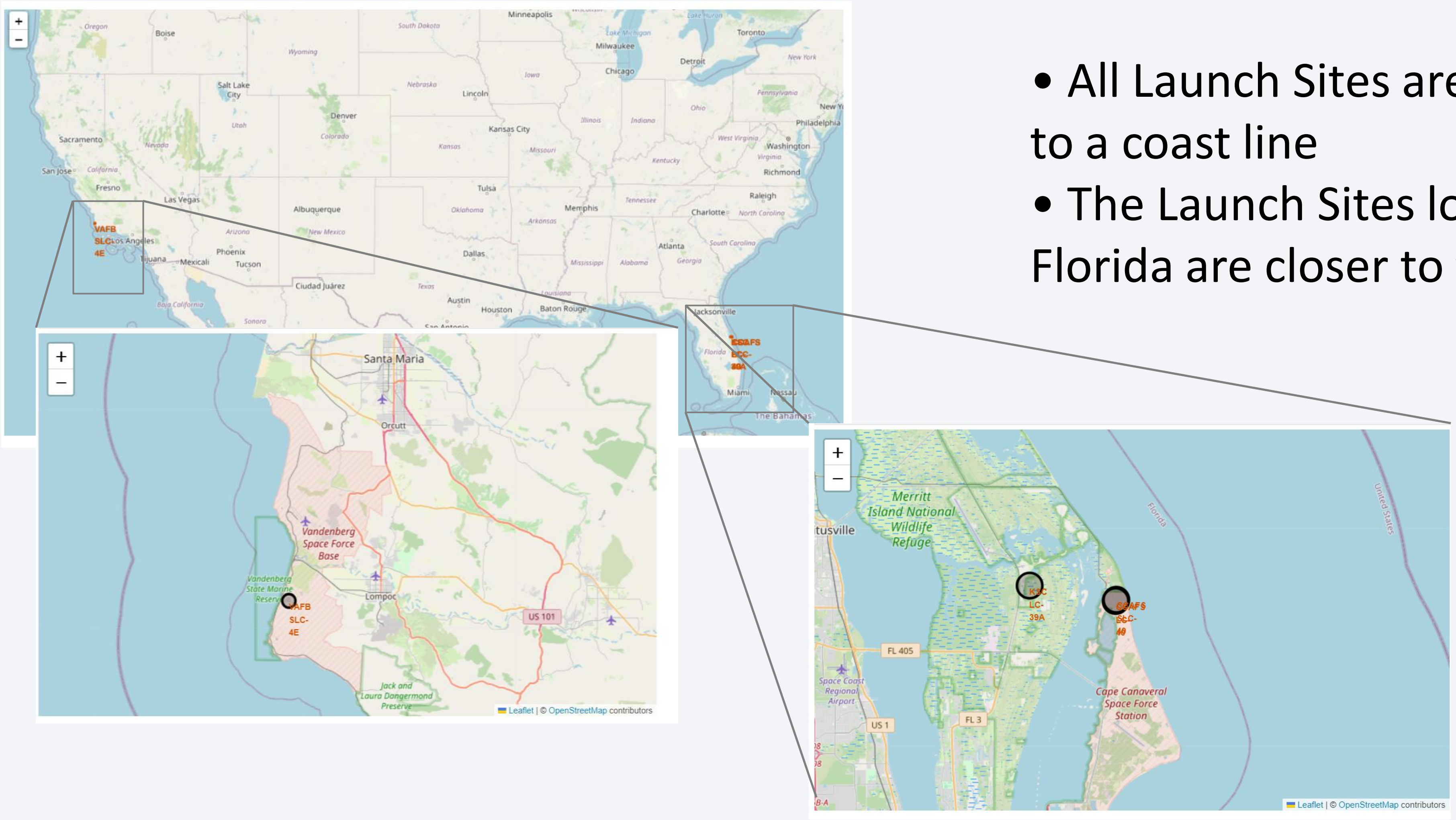
Explanation:
SELECT Landing_Outcome,count(Landing_Outcome) - to return count and Landing Outcome
where Date between '2010-06-04' and '2017-03-20' - to return only records within set data limit
group by Landing_Outcome order by count(Landing_Outcome) desc - to group by Outcome, and rank

Landing Outcome ranking was as listed to the right

Section 3

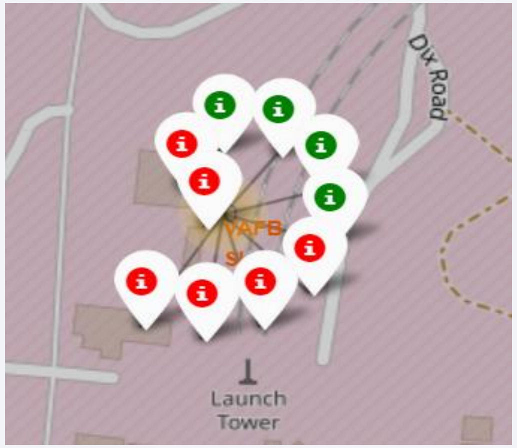
Launch Sites Proximities Analysis

Launch Sites Overview

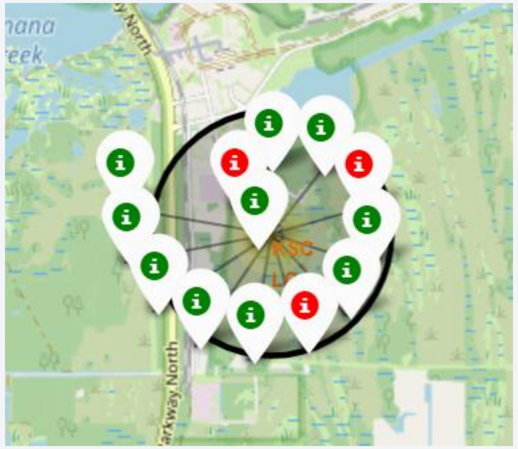


- All Launch Sites are located close to a coast line
- The Launch Sites located in Florida are closer to the equator

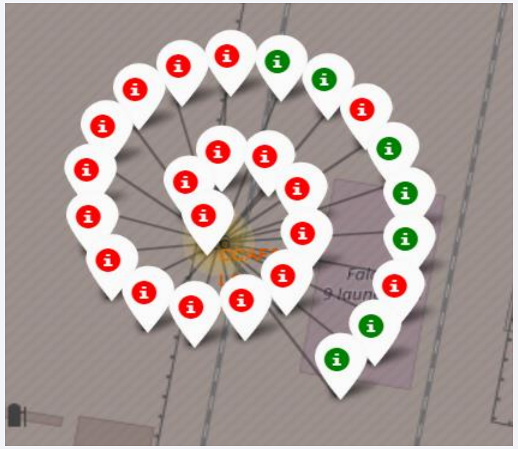
Launch Sites – Close-up and Mission Outcomes



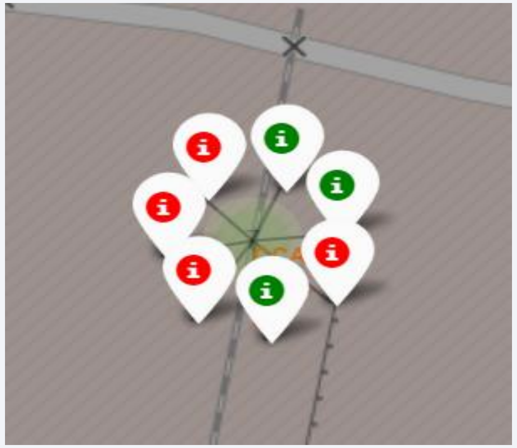
**Vandenberg
Space Force Base**



**Kennedy
Space Center**



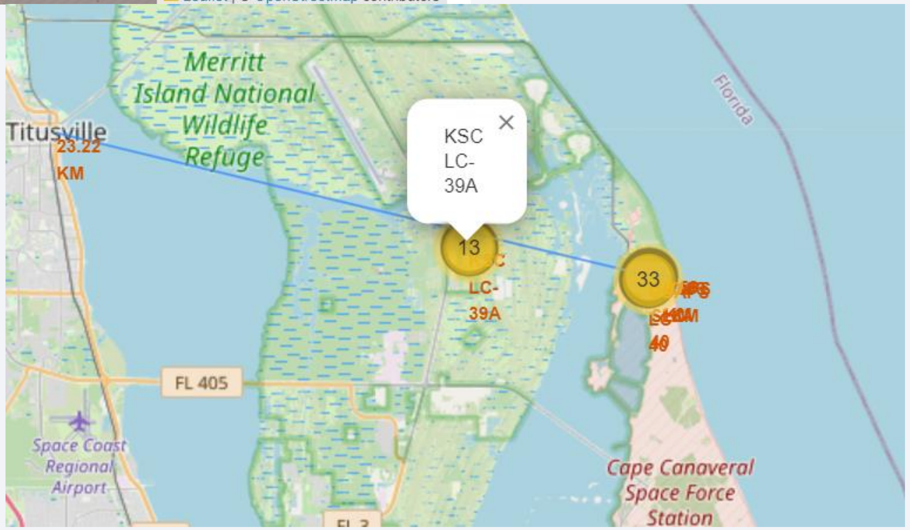
**Cape Canaveral
Space Launch Center**



**Cape Canaveral
Launch Complex**

- Launch Sites are indicated by the black circle
- Successful Missions are indicated by a green marker
- Failed Missions are indicated by a red marker
- Kennedy Space Center has the highest Mission Success Rate

Distance between Launch Sites and Point of Interest



- The distance between the CCAFS SLC several points of interest are indicated by the blue line and distance indicator

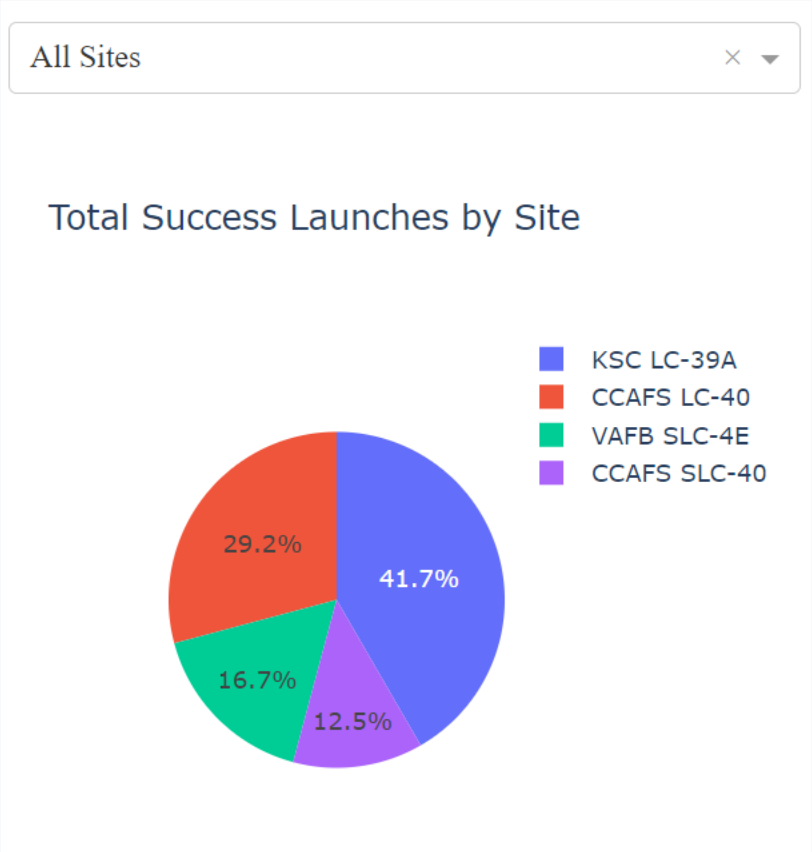
Distances:

- Upper screenshot:
  - Highway: 0.59 km
  - Coast line: 0.86 km
  - Railway: 1.22 km
- Lower screenshot:
  - City (Titusville): 23.22 km

Section 4

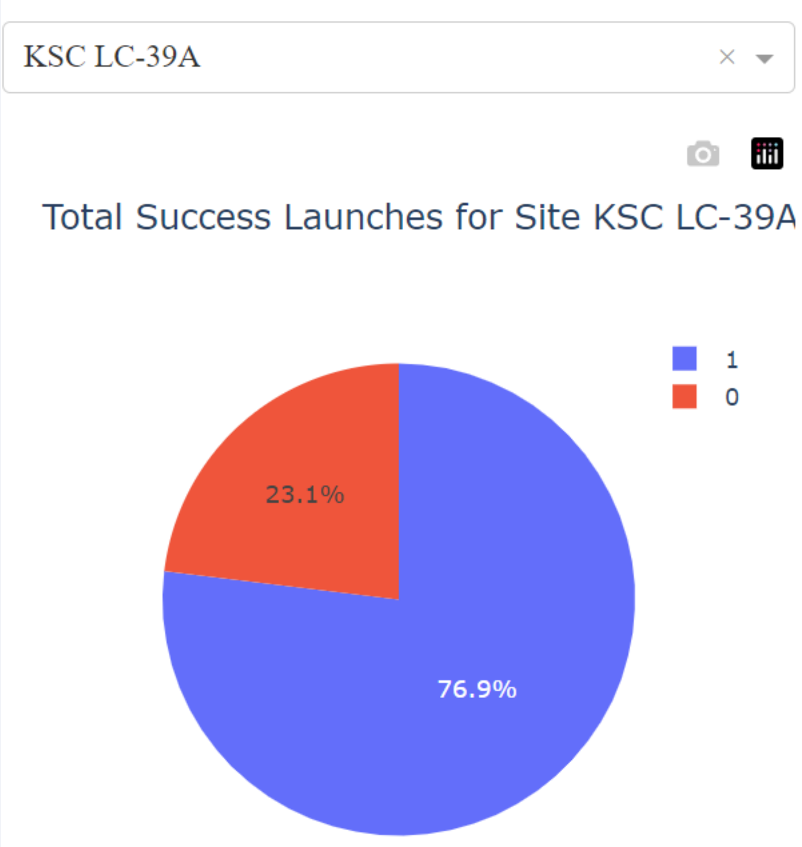
Build a Dashboard with Plotly Dash

Launch success count for all sites



- Kennedy Space Center (KSC LC, blue) contributes most to Mission Success, with 41.7% of the successful launches being performed there
- Cape Canaveral Space Launch Center (CCAFS SLC, purple) contributed the least to Mission Success, with 12.5% of the successful launches originating from there

Launch Success: rate at the most successful Launch Site



- Kennedy Space Center (KSC LC, blue) is the most successful Launch Site
- 76.9% of the Launches was successful here (blue, Class 1)
- 23.1% of the Launches failed (red, Class O)

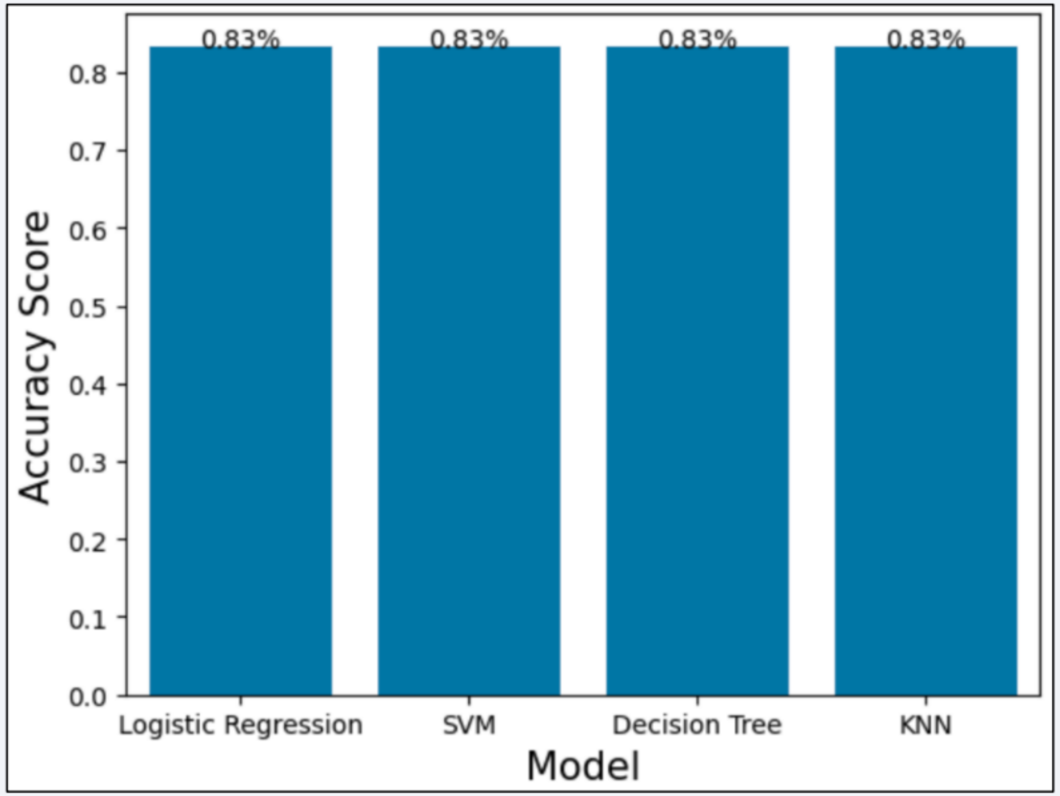
Effect of Payload Mass on Launch Outcome

- Optimal Payload Mass appears to range from 2,000 to 6,000 kg
- Booster version FT appears to have the highest success rate
- Booster v 1.1 is generally unsuccessful, even in the optimal Payload Mass range

Section 5

Predictive Analysis (Classification)

Classification Accuracy

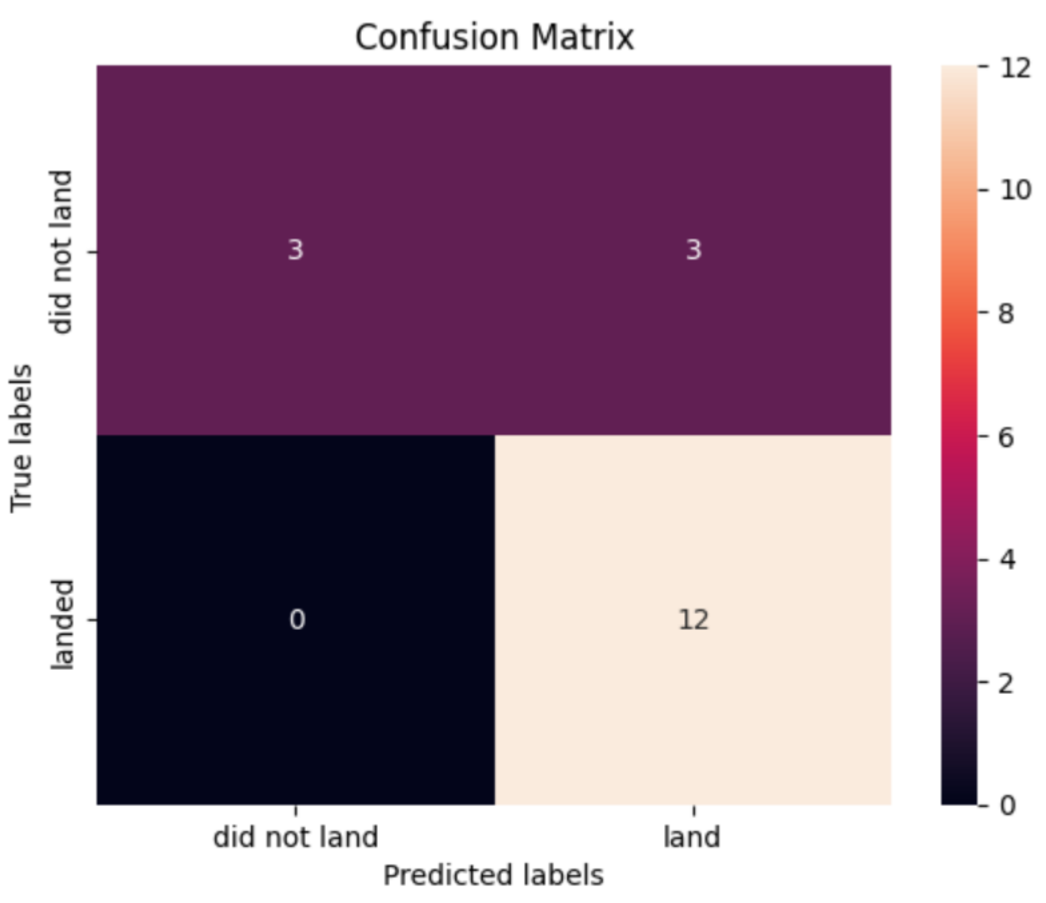


Four Predictive Models were Developed and tested:

- Logistic Regression
- Support Vector Machines
- Decision Tree
- k-Nearest Neighbours

With a Classification Accuracy of 0.833333, all predictive models performed equally well

Confusion Matrix



- All models performed equally well, and generated the same Confusion Matrix

- The Confusion Matrix of k-Nearest Neighbours is shown here

- The models work well in predicting a positive outcome, but not for the negative outcome

Conclusions

- To maintain the financial edge of SpaceX, it is vital that the Falcon 9's Reusable First Stage (RFS) lands successfully, in order to be re-used

- We here use Data Science to optimise Mission Success Rate, collecting historical data utilising the SpaceX API and web scraping, applying data wrangling, analysing data, and generating predictive models

- With time and increasing Flight Number, Mission Success Rate increased hugely, highlighting the importance of gaining experience

- The optimal Payload Mass appears to range from 2K to 6K kg, although a clear correlation could not be detected

- Kennedy Space Center is the most successful base, with a success rate of over 75%

- Prediction of Mission Failure proved challenging. Supplementing the models with additional parameters might be useful to address this issue

Thank you!