IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Laurent Klopfenstein
05/06/2024

Outline

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

Executive Summary

- Summary of methodologies
  - Data collection of SpaceX launch data
  - Data wrangling and exploratory data analysis (EDA)
  - Creation of an interactive dashboard to analyze launch records and of a map to visualize infrastructures in proximity to launch sites
  - Predictive analysis with the help of a machine learning model (classification)
- Summary of all results
  - Features largely contributing to the success of launches are :
    - Localisation of a launch site
    - Payload mass
    - Booster version
  - Machine learning model with an accuracy of 83,3% on test data

Introduction

- ## Project background and context
- The commercial space age is here, companies are making space travel affordable for everyone
- Several players are competing for this market but the most successful is SpaceX
- SpaceX's success comes from their lower/affordable prices which are achieved by reusing the first stage of their rockets, which drastically reduces their costs
- The new company Space Y would like to enter the market and compete with SpaceX,

- ## Problems and questions this project addresses
- How much do reusable launches cost ?
- What features help predict the outcome of a launch ?
- Can the outcome of a launch be predicted with a model ?

Section 1

Methodology

Methodology

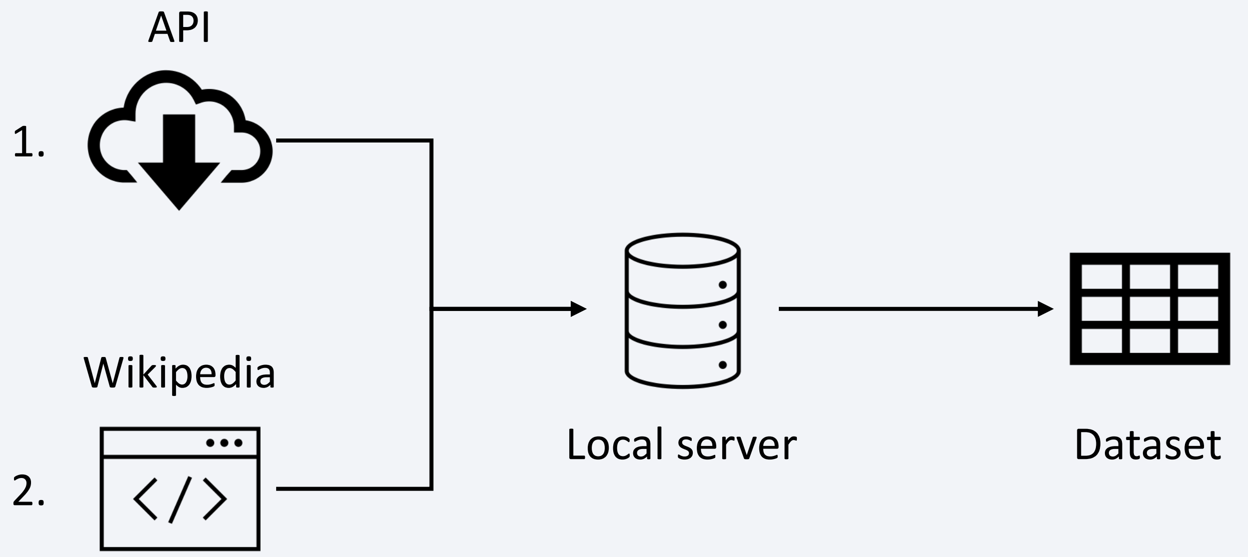
Executive Summary

- Data collection methodology:
  - Collection of relevant launch data via GET request to the SpaceX REST API and through web scraping of the “List of Falcon 9 and Falcon heavy launches” Wikipedia page
- Data wrangling
  - Dataset filtered on Falcon 9 launches and addition of a landing outcome feature
- Exploratory data analysis (EDA) using visualization and SQL
- Interactive visual analytics using Folium and Plotly Dash
- Predictive analysis using classification models
  - Performance evaluation of 4 models (Logistic regression, SVM, Decision Tree, KNN) on dataset to predict launch outcome

Data Collection

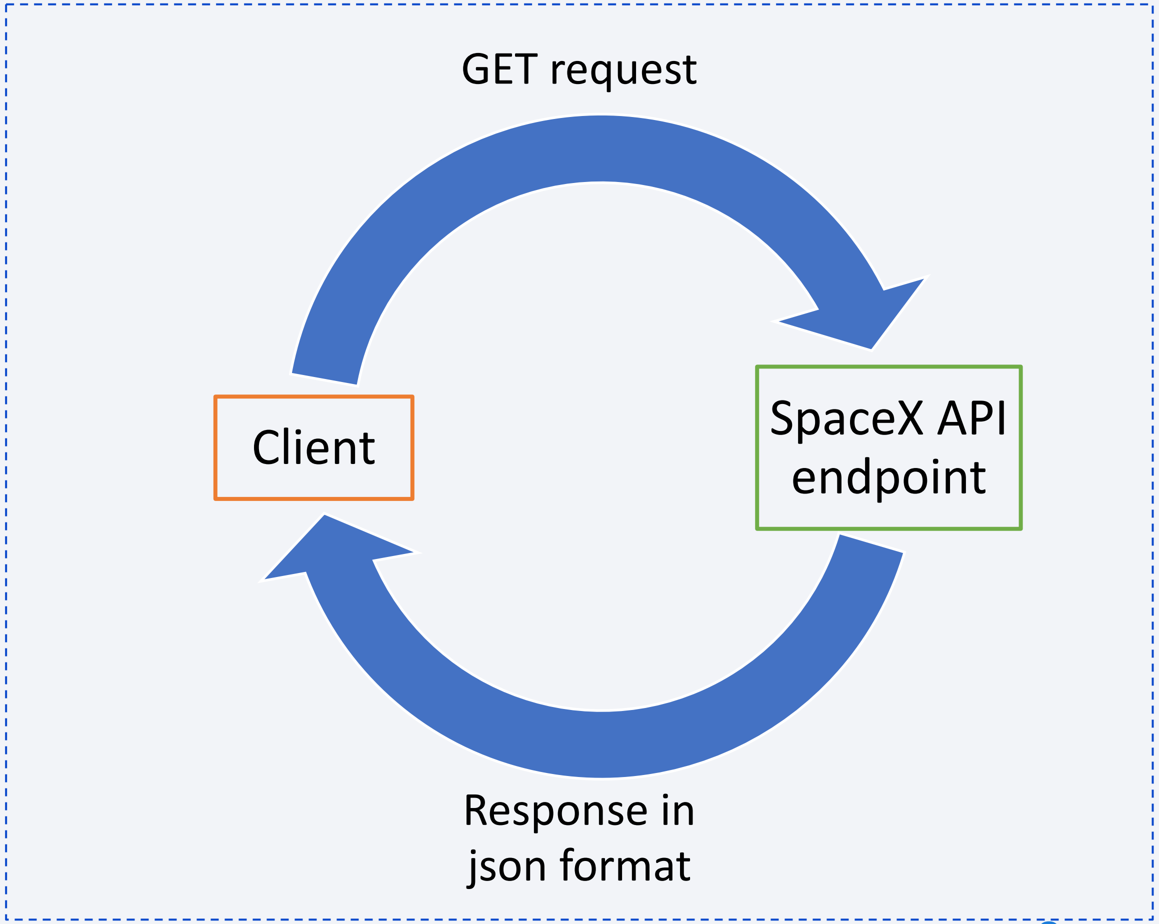
Datasets were collected through 2 sources :

1. SpaceX REST API calls
2. Web scraping the Falcon9 launches Wikipedia page



Data Collection – SpaceX API

- How the SpaceX API works :
  - We ask for information by sending a HTTP GET request from our client to the SpaceX API endpoint (usage of **requests** library)
  - The SpaceX API endpoint sends a response back in the format of a json file
- Because the information we need is scattered across several endpoints, to get a complete dataset, we must send a GET request to each endpoint individually. The endpoints are :
  - Past launches : https://api.spacexdata.com/v4/launches/past
  - Rockets : https://api.spacexdata.com/v4/rockets/
  - Launchpads : https://api.spacexdata.com/v4/launchpads/
  - Payloads : https://api.spacexdata.com/v4/payloads/
  - Cores : https://api.spacexdata.com/v4/cores/

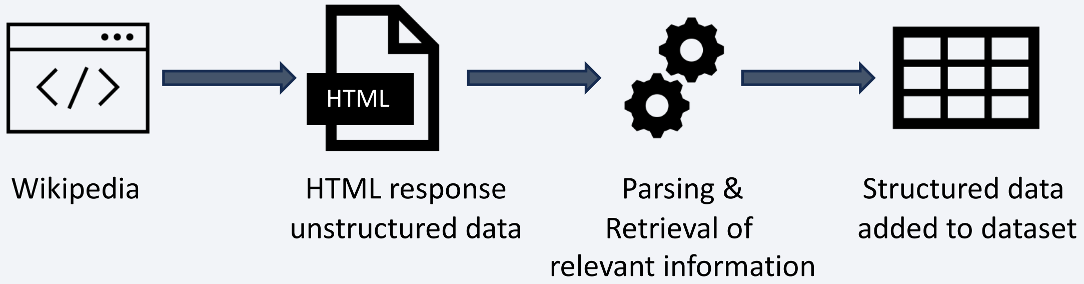


GitHub link :
[Jupyter notebook - SpaceX API calls]()

Data Collection - Scraping

- How the collection through scraping works :
  - We make a HTTP GET request to get the HTML page from the Wikipedia link we want to scrape as an HTTP response, with the **requests** library
  - With the response object we create a BeautifulSoup object which will allow us to parse the content of the response
  - We go through the different HTML elements of the Beautiful object to retrieve all relevant information which can be found in a Falcon 9 launches table and add these to our dataset
- Wikipedia link : https://en.wikipedia.org/w/index.php?title=List_of_Falcon_9_and_Falcon_Heavy_launches&oldid=1027686922

GitHub link :
Jupyter notebook - Web scraping



Data Wrangling

- Basic preliminary analysis of the content of the dataset
  - Missing values
  - Data type of the values in each column
  - Number of launches on each site
  - Number and occurrence of each orbit
  - Number and occurrence of mission outcome for each orbit
- Using the Outcome feature, creation of an outcome label ‘Class’ to distinguish between successful and failed landings of the first stage of a rocket

GitHub link :
[Jupyter notebook - Data Wrangling]

EDA with Data Visualization

To visualize the effect of the selected features on successful landing of the first stage, different charts have been drawn.
You'll find below the list of charts that've been used for analysis :

- Scatter plot of :
  - Relationship between Payload Mass in kg and Flight Number with landing outcome
  - Relationship between Launch Site and flight number with landing outcome
  - Relationship between Payload Mass in kg and Launch Site with landing outcome
  - Relationship between Flight Number and Orbit type with landing outcome
  - Relationship between Payload Mass in kg and Orbit type with landing outcome
- Other charts :
  - Bar chart of Landing outcome success rate per Orbit type
  - Line graph representing a yearly trend for Average Landing outcome success rate

GitHub link :
[Jupyter notebook - EDA with Data Visualization]

EDA with SQL

SQL performed queries :

- Display unique launch site names
- Display records for which the launch site starts with 'CCA'
- Display total payload mass carried by boosters launched by NASA (CRS)
- Display average payload mass carried by booster version F9 v1.1
- Display the date when the first successful landing outcome on ground pad was achieved :
- Display the names of the boosters which have had success in drone ship landing outcome and have a payload mass between 4000 kg and 6000 kg
- Display the total number of successful and failed mission outcomes
- Display the names of the booster_versions which have carried the maximum payload mass (15600 kg)
- Display the month, landing outcome, booster version and launch site for records with a landing outcome failure for drone ship in year 2015
- Display the count of landing outcomes between the date 2010-06-04 and 2017-03-20 ranked in descending order

GitHub link :
[Jupyter notebook - EDA with SQL]

Build an Interactive Map with Folium

To find geographical patterns about launch sites, an interactive Folium map has been built with the data of the launch sites. Several objects were added to the map to better understand the location choice :

- Markers of the location of the launch sites
- Markers of successful/failed launches clustered together for each site
- Distances between launch sites and their proximities – to help understand how the location of a launch sites is determined

GitHub link :
[Jupyter notebook - Folium interactive Map]

Build a Dashboard with Plotly Dash

An interactive dashboard has been built with Dash so that users can examine information about launch sites, launches and their associated payload, and most importantly the outcome of the landings.
The objective of the interactive components is to enable users to access and filter information based on their needs and to give them statistical data about launches and show how the chosen parameters influence the landing outcomes.

In the dashboard layout several components were implemented :

- A dropdown list for launch site selection, with a default setting of combined information (”ALL”) or upon selection, individual information for launch sites (”CCAFS SLC-40”, “KSC LC-39A”, …)
- A pie chart to show the total successful launches count for all sites. If a specific launch site was selected, the Success vs. Failed counts for the site are shown
- A range slider to select an interval of payload mass in kg
- A scatter chart to show the correlation between payload and launch success

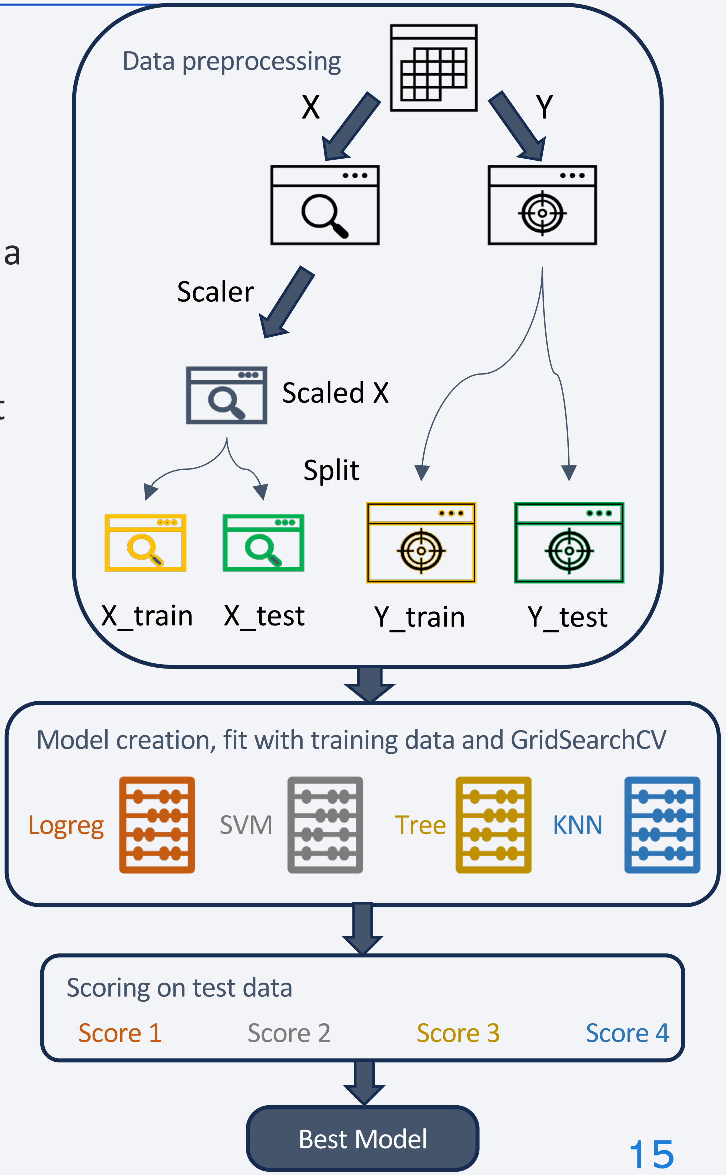
Furthermore 2 callback functions were added for automatic update of the 2 graphical components upon input modification :

- A callback function to render the pie chart based on the selection in the site dropdown
- A callback function to render the scatter chart based on the payload interval selected in the range slider

GitHub links :
[Jupyter notebook - Dash app run]
[Dash.py]

Predictive Analysis (Classification)

- Separation of the dataset into 2 variables, one for the features called X and one for the target called Y (outcome of the landing called 'Class').
  In the end we want the model to be able to predict Y with the input of X.
- We scale the features data using StandardScaler from sklearn.preprocessing library.
  This step ensures that the model is robust against the negative impact of extreme values (outliers) and a potentially skewed dataset on its accuracy.
- Split of the dataset into training and test (i.e. validation) set with a split of 80/20.
  This is to assess the ability of the model to generalize and accurately predict results of an unknown test set after having been trained on our training data.
- Testing of 4 different classification models to find the best performing one. This is done in 4 steps :
  - Creation of a model specific object
  - Create and fit a GridSearchCV object with the model and training data and test different parameter values to find the best ones
  - Calculate accuracy on the test data using the score method
  - Plot a confusion matrix
- Compare the results of all the models to select the best performing model



GitHub links :
[Jupyter notebook - Predictive Analysis]

Results

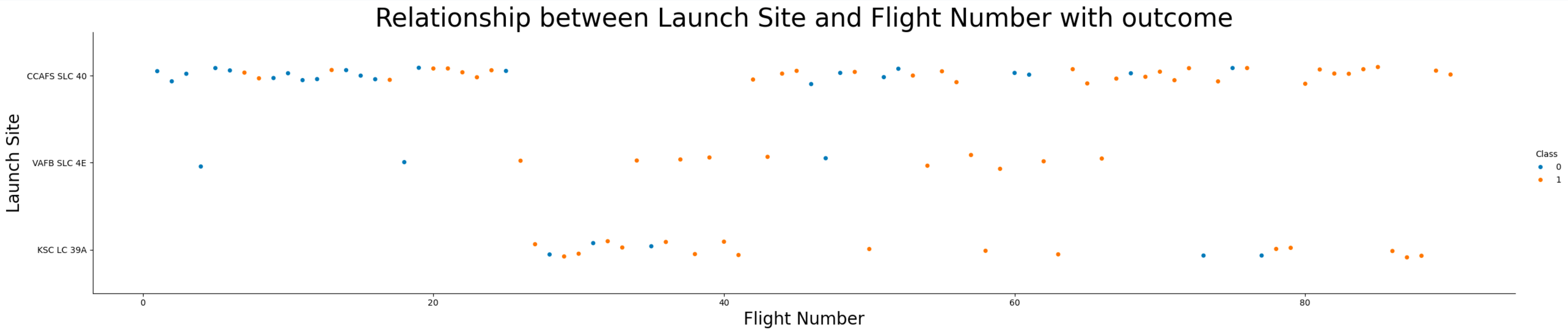
- Exploratory data analysis results
- Interactive analytics demo in screenshots
- Predictive analysis results

Section 2

Insights drawn from EDA

Flight Number vs. Launch Site

To visualize the effect of the selected features on successful landing, different scatter plots have been drawn.
First, we tried to see if there was a relationship between the choice launch site and the outcome of each flight :

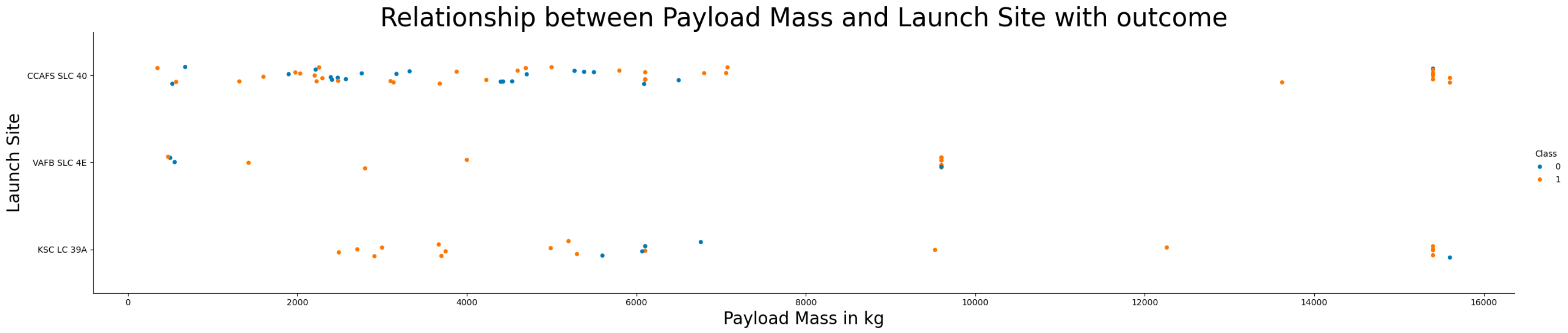


Observations :

- Site CCAFS SLC-40 was being used almost exclusively in the 25 first launches, with little outcome success
- CCAFS SLC-40 stopped being used between flight 25 and 40 and the other launch sites, VAFB SLC-4^{E} & KSC LC-39A, showed promising outcome successes
- From flight 40 and onwards CCAFS SLC-40 was predominantly used with an improved outcome success compared to the initial 25 flights, which can mean that either problems related to the launch site were fixed, or that technology in general was improved which resulted in better chances for landing success

Payload vs. Launch Site

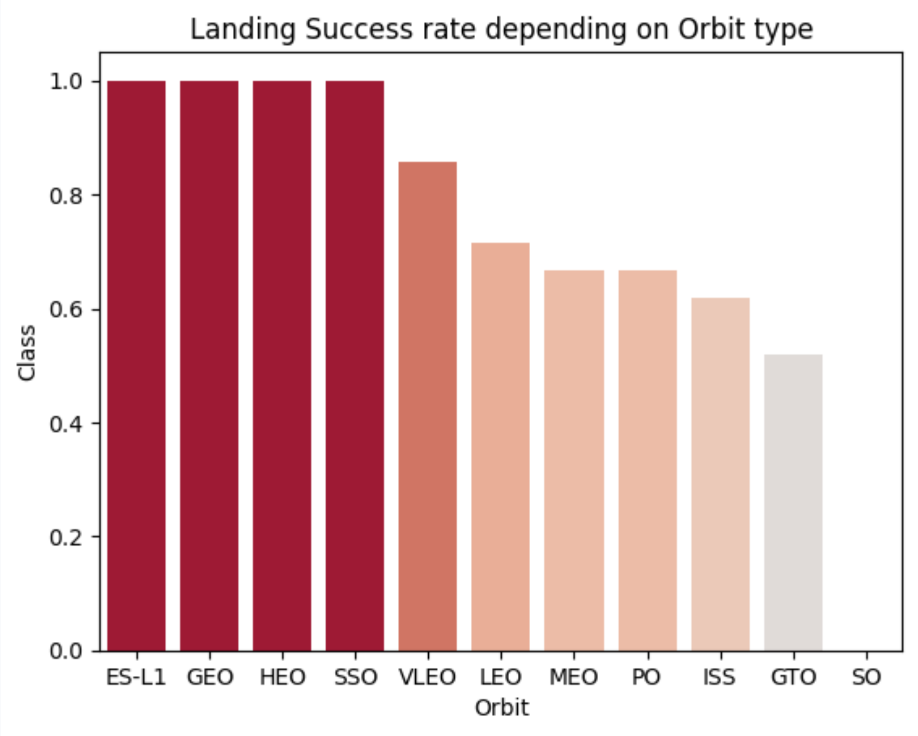
Payload mass and launch site seem to be correlated, but another feature might explain why some heavy launches are made exclusively at some launch sites, which points towards a differentiating feature between launch sites. Either VAFB SLC-40 doesn't meet some requirements, for instance specific infrastructure needs or another feature influences this choice, for instance the type of Orbit.



If you observe the Payload mass vs. Launch Site chart you will find that for the VAFB-SLC launch site there are fewer launches and no rockets launched for heavy payload mass (greater than 10000).

Success Rate vs. Orbit Type

Another factor seems to come into play when explaining landing success, which is Orbit type as we can see in the bar chart below :



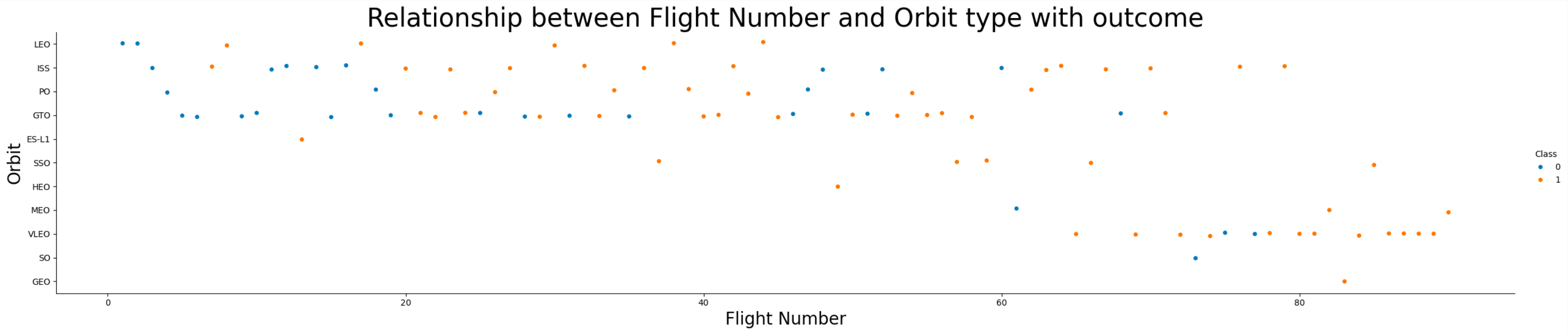
Observations :

- 4 orbit types seem to have a perfect 100% success rate, which shows that some orbit types carry less/more risk for landing outcome
- VLEO orbit type shows a high success rate around 85% success rate
- 4 orbit types compete in the same range of success rate around 60-70%
- Finally, GTO's success rate is the lowest at about 53%

We can infer that different orbit types carry different outcome success because they also have different starting requirements. For instance certain orbit types might require a specific launch site to be used, or a certain amount of payload to be used, which can augment the risk of landing failure for the first stage.

Flight Number vs. Orbit Type

Watching the relationship between flight number and orbit type unveils a trend over time (especially more recently) towards Very low Earth orbit (VLEO orbit type) launches.

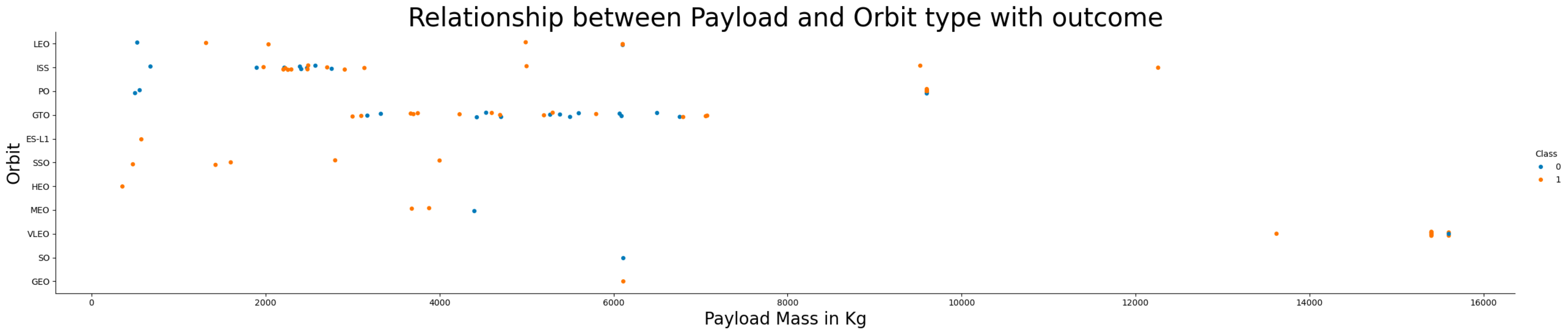


Observations :

- For the LEO orbit type the success appears to improve with the number of flights
- On the other hand, quantity seems to have improved the success for the GTO orbit, but the outcome of a successful landing remains uncertain as is shown in the results of the flight before last of this orbit type

Payload vs. Orbit Type

The scatter chart for the relationship between payload and orbit type shows evidence for a positive correlation between the 2 features; each orbit type requires a certain amounts of payload and the higher the orbit, the bigger the payload.

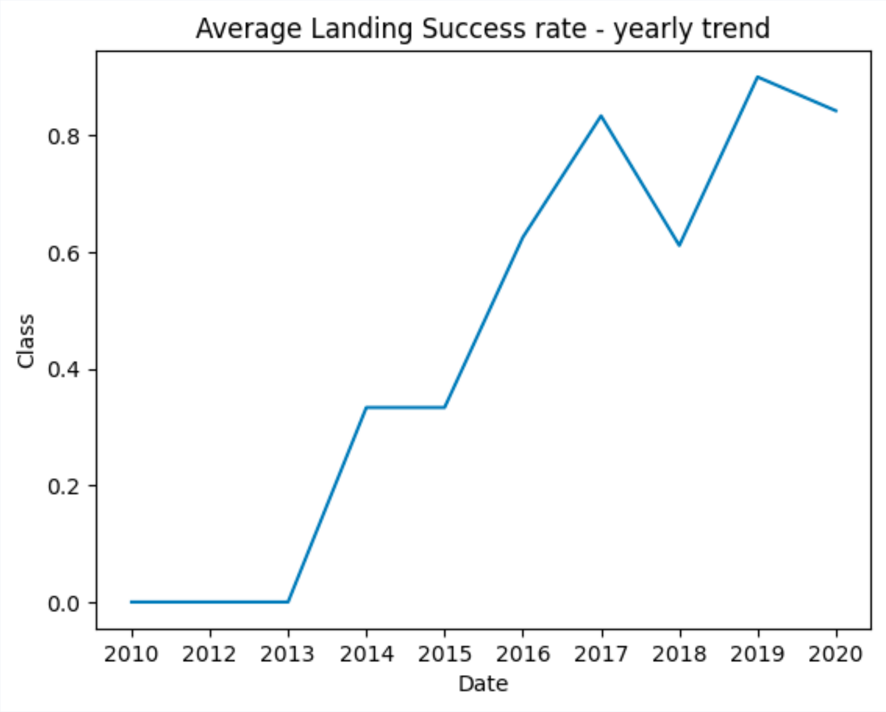


<u>Observations :</u>

- With heavy payloads the successful landing or positive landing rate are higher for Polar, LEO and ISS.
- For GTO we cannot distinguish this clearly, as both positive landing rate and negative landing (unsuccessful missions) are both equally displayed for the same payload mass

Launch Success Yearly Trend

With time (i.e. increasing flight number) success seems to have been drastically improved, starting from 2013.



Observations :

- Over the first 3 years, no landing of the first stage was successful
- After 2013 landing success rate has drastically been improved and has overall followed a positive/upward trend
- A drop in success rate from ≈ 80% to ≈60% can be seen right around the time where launch site CCAFS LC-40 was employed again
- Landing success reached an all-time high in 2019

All Launch Site Names

SQL script :

```
SELECT DISTINCT Launch_Site FROM SPACEXTBL
```

Output :

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

Query explanation :

- `SELECT DISTINCT Launch_Site FROM SPACEXTBL` **:** Select all **DISTINCT (=unique)** values in the **column** called **Launch_Site** from the table **SPACEXTBL** (**FROM SPACEXTBL**)

Launch Site information :

- **Cape Canaveral Space Launch Complex 40** (**CCAFS SLC-40**), sometimes pronounced **Slick Forty** and previously **Cape Canaveral Launch Complex 40** (**CCAFS LC-40**) is a launch pad for rockets located at the north end of Cape Canaveral Space Force Station, Florida
- **Vandenberg Force Base Space Launch Complex 4** (**VAFB SLC-4**) is a launch and landing site at Vandenberg Space Force Base, California, U.S. It has two pads, both of which are used by SpaceX for Falcon 9, one for launch operations, and the other as **Landing Zone 4** (**LZ-4**) for SpaceX landings.
- **Kennedy Space Center Launch Complex 39A** (**KSC LC-39A**) is the first of Launch Complex 39's three launch pads, located at NASA's Kennedy Space Center in Merritt Island, Florida. The pad, along with Launch Complex 39B, was first constructed in the 1960s to accommodate the Saturn V launch vehicle, and has been used to support NASA crewed space flightmissions, including the historic Apollo 11 moon landing and the Space Shuttle.

(Source : en.wikipedia.org)

Launch Site Names Begin with 'CCA'

SQL script :

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

Output :

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

Query explanation :

- `SELECT * FROM SPACEXTBL` : select all columns and associated values from the table SPACEXTBL (**FROM SPACEXTBL**)
- Condition `WHERE "launch_site" LIKE "CCA% »` : Records must verify the condition that the values in column « **launch_site** » start with « **CCA** » and can end with anything else « **%** »
- `LIMIT 5` : Display only 5 records from the selection

Total Payload Mass

SQL script :

```
SELECT SUM(PAYLOAD_MASS__KG_) AS "Total Payload" FROM SPACEXTBL WHERE "Customer" =
"NASA (CRS)"
```

Output :

Total Payload
45596

Query explanation :

- `SELECT SUM(PAYLOAD_MASS__KG_) AS "Total Payload" FROM SPACEXTBL` : Select the sum of the values in the column called **PAYLOAD_MASS_KG** (**SUM(PAYLOAD_MASS_KG**) from the table **SPACEXTBL** (**FROM SPACEXTBL**) and name the resulting column « **Total Payload** » (**AS « Total Payload »**)
- Condition `WHERE "Customer" = "NASA (CRS) »` : records must verify the condition of having the value « **NASA (CRS)** » in the **Customer column**

Average Payload Mass by F9 v1.1

SQL script :

```
SELECT AVG(PAYLOAD_MASS__KG_) AS "Average Payload" FROM SPACEXTBL WHERE
"booster_version" = "F9 v1.1"
```

Output :

Average Payload
2928.4

Query explanation :

- `SELECT AVG(PAYLOAD_MASS__KG_) AS "Average Payload" FROM SPACEXTBL` : Select the average of the values in the column called **PAYLOAD_MASS_KG** (**AVG(PAYLOAD_MASS_KG**) from the table **SPACEXTBL** (**FROM SPACEXTBL**) and name the resulting column « **Average Payload** » (**AS « Average Payload »**)
- Condition `WHERE "booster_version" = "F9 v1.1"` : records must verify the condition of having the value « **F9 v1.1**» in the « **booster_version** » **column**

First Successful Ground Landing Date

SQL script :

```
SELECT MIN("Date") AS 'First successful landing' FROM SPACEXTBL WHERE
"landing_outcome" = 'Success'
```

Output :

First successful landing
2018-07-22

Query explanation :

- `SELECT MIN("Date") AS 'First successful landing' FROM SPACEXTBL` : Select the smallest = earliest value of the values in the column called **Date** (**MIN(« Date »)**) from the table **SPACEXTBL** (**FROM SPACEXTBL**) and name the resulting column « **First successful landing** » (**AS « First successful landing»**)
- Condition `WHERE "landing_outcome" = 'Success'` : records must verify the condition of having the value «**Success**» in the « **landing outcome» column**

Successful Drone Ship Landing with Payload between 4000 and 6000

SQL script :

```
SELECT DISTINCT "booster_version" FROM SPACEXTBL WHERE ("payload_mass__kg_" BETWEEN
4000 AND 5999) AND ("landing_outcome" LIKE 'Success (drone ship)')
```

Output :

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

Query explanation :

- `SELECT DISTINCT "booster_version" FROM SPACEXTBL` : Select all **distinct** (**=unique**) values in the column called « **booster_version** » (**DISTINCT « booster_version »**) from the table **SPACEXTBL** (**FROM SPACEXTBL**)
- Condition 1 `WHERE ("payload_mass__kg_" BETWEEN 4000 AND 5999)` : records must verify the condition of having a value **between 4000 and 6000** in the « **payload_mass_kg** » **column**
- Condition 2 `AND ("landing_outcome" LIKE 'Success (drone ship)')`: records must verify the condition of having the value « **Success (drone ship)** » in the « **landing_outcome** » **column**

Total Number of Successful and Failure Mission Outcomes

SQL script :

```
SELECT CASE WHEN "mission_outcome" LIKE 'Success%' THEN 'Success' ELSE 'Failure' END
AS 'Mission Outcome Category', COUNT(*) AS 'Count' FROM SPACEXTBL GROUP BY "Mission
Outcome Category" ORDER BY 'Count' DESC
```

Output :

Mission Outcome Category	Count
Success	100
Failure	1

Query explanation :

- `SELECT CASE WHEN "mission_outcome" LIKE 'Success%' THEN 'Success' ELSE 'Failure' END AS 'Mission Outcome Category' AS 'Mission Outcome Category'` : Select values from column « **mission_outcome** » and modify them based on 2 cases. If values start with « **Success%** » then modify them into « **Success** » (**CASE…. LIKE 'Success%'**), for anything else modify them into « **Failure** » (**ELSE 'Failure' END**); from the table **SPACEXTBL** (**FROM SPACEXTBL**) and call the resulting column '**Mission Outcome Category**' (**AS 'Mission Outcome Category'**)
- And `COUNT(*) AS 'Count' FROM SPACEXTBL` : add a count of all above created values, either « Success » or « Failure » in a column called « **Count** »
- `GROUP BY "Mission Outcome Category" ORDER BY 'Count' DESC` : group records by « Mission Outcome category » (**GROUP BY « Mission Outcome Category »**) and order them by value in column « **Count** » in descending order (**ORDER BY 'Count' DESC**)

Boosters Carried Maximum Payload

SQL script :

```
SELECT Booster_Version FROM SPACEXTBL WHERE PAYLOAD_MASS__KG_ = (SELECT
MAX(PAYLOAD_MASS__KG_) FROM SPACEXTBL)
```

Output :

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

Query explanation :

- `SELECT Booster_Version FROM SPACEXTBL` : Select values from **column Booster_Version** from the table **SPACEXTBL** (**FROM SPACEXTBL**)
- Condition with a nested select `WHERE PAYLOAD_MASS__KG_ = (SELECT MAX(PAYLOAD_MASS__KG_) FROM SPACEXTBL)` : condition that the selected values of booster_version must have a payload_mass_kg value equal (**WHERE PAYLOAD_MASS_KG =**) to the selection of the highest in the column PAYLOAD_MASS_KG from the SPACEXTBL (**SELECT MAX(PAYLOAD_MASS_KG) FROM SPACEXTBL**)

2015 Launch Records

SQL script :

```
SELECT substr(Date, 6,2) AS 'Month', "landing_outcome", "booster_version", "launch_site" FROM SPACEXTBL WHERE (substr(Date,0,5)='2015') AND ("landing_outcome" Like 'Failure%')
```

Output :

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

Query explanation :

- `SELECT substr(Date, 6,2) AS 'Month', "landing_outcome", "booster_version", "launch_site" FROM SPACEXTBL` : Select from the values in **column Date** the characters starting at position 6 and stopping after 2 values (**SELECT substr(Date, 6, 2) AS 'Month'**) and call the resulting column Month. This will effectively extract the month of the Date values.
- `"landing_outcome", "booster_version", "launch_site" FROM SPACEXTBL` : also select the columns « **landing_outcome** », « **booster_version** », « **launch_site** » from the table **SPACEXTBL** (**FROM SPACEXTBL**)
- Condition 1 `WHERE (substr(Date,0,5)='2015')` : condition that the values at position 0 and going to position 5 are equal to « **2015** »
- Condition 2 : `AND ("landing_outcome" Like 'Failure%')` : condition that the values in column « **landing_outcome** » at minimum start with the word Failure (**LIKE « Failure% »**)

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

SQL script :

```
SELECT "landing_outcome", COUNT(*) AS 'Count' FROM SPACEXTBL WHERE "Date" BETWEEN
DATE('2010-06-04') AND DATE('2017-03-20') GROUP BY "landing_outcome" ORDER BY Count
DESC
```

Output :

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

Query explanation :

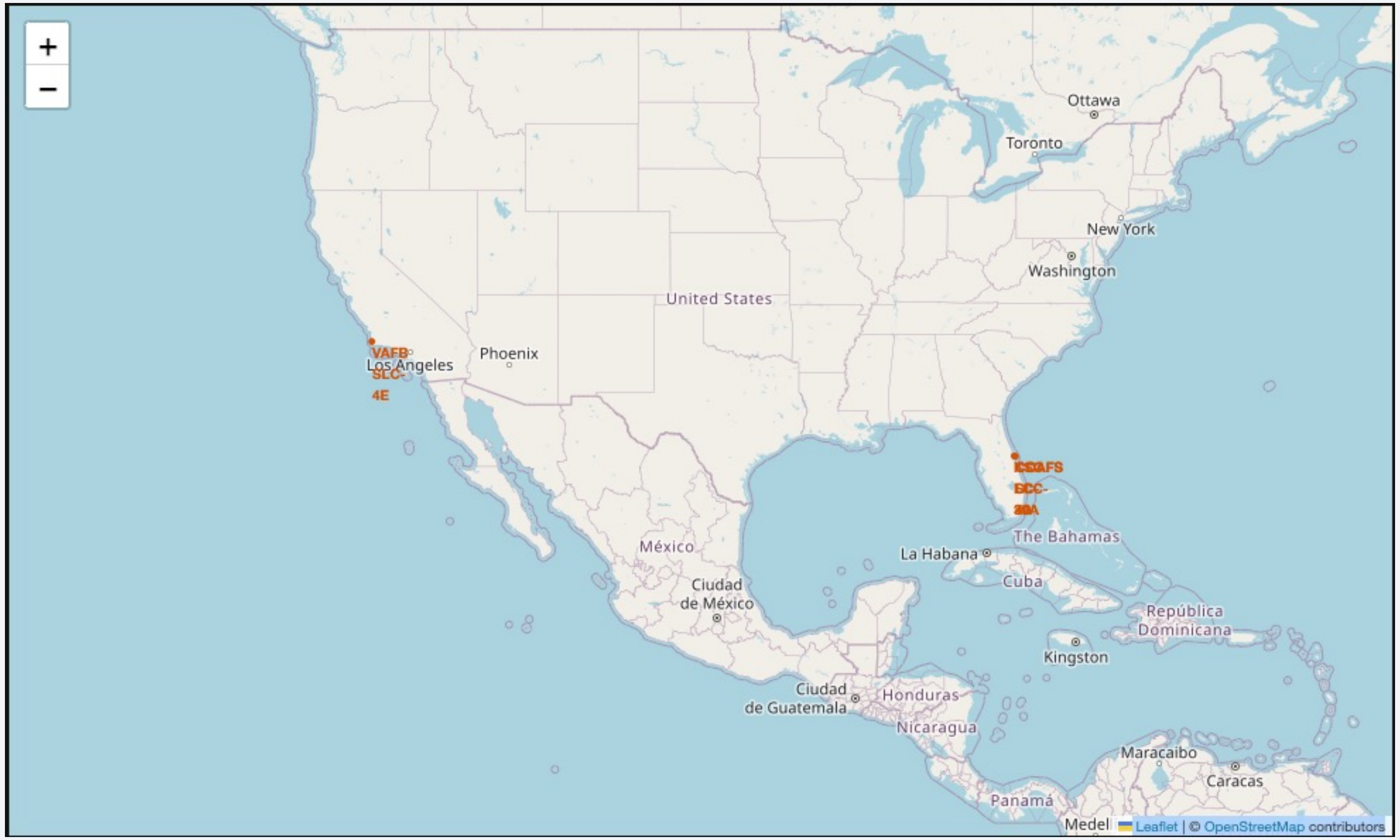
- `SELECT "landing_outcome", COUNT(*) AS 'Count' FROM SPACEXTBL` : Select the **column « landing_outcome »** (SELECT « landing_outcome ») and create a column called **Count** of all records (COUNT(*) AS 'Count') from the table **SPACEXTBL** (**FROM SPACEXTBL**)
- Condition `WHERE "Date" BETWEEN DATE('2010-06-04') AND DATE('2017-03-20')` : condition that the values in the column « **Date** » are in the date interval from **2010-06-04** to **2017-03-20**
- `GROUP BY "landing_outcome »` : group the records by the value that they have in **column « landing_outcome »**
- `ORDER BY Count DESC` : order the values by **descending order** in the **column Count**

Section 3

Launch Sites Proximities Analysis

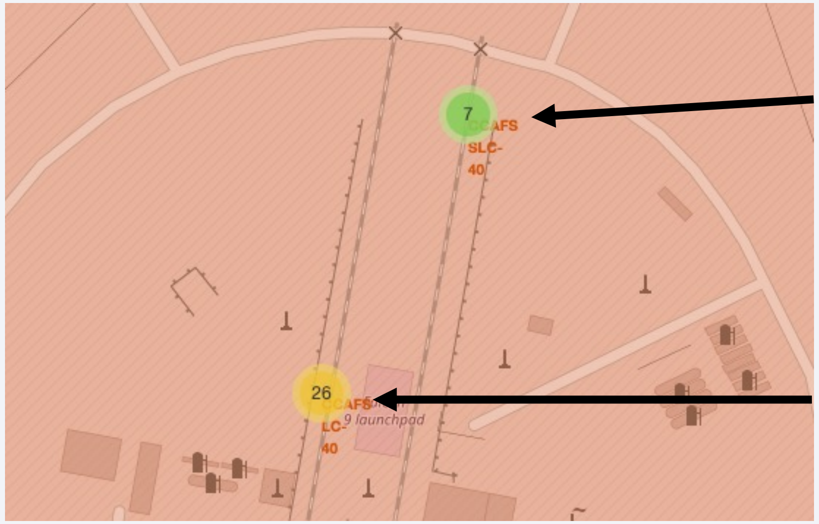
SpaceX Launch Sites Location

- SpaceX's uses 3 launch sites on the USA East coast and 1 on the West coast.
- Because of the zoom of the screenshot, it is difficult to distinguish the 3 launch sites **CCAFS LC-40, CCAFS SLC-40** and **KSC LC-39A** in **Florida** as they are only a few kilometers apart.
- The fourth launch site, **VAFB SC-4E** is located in **California**.



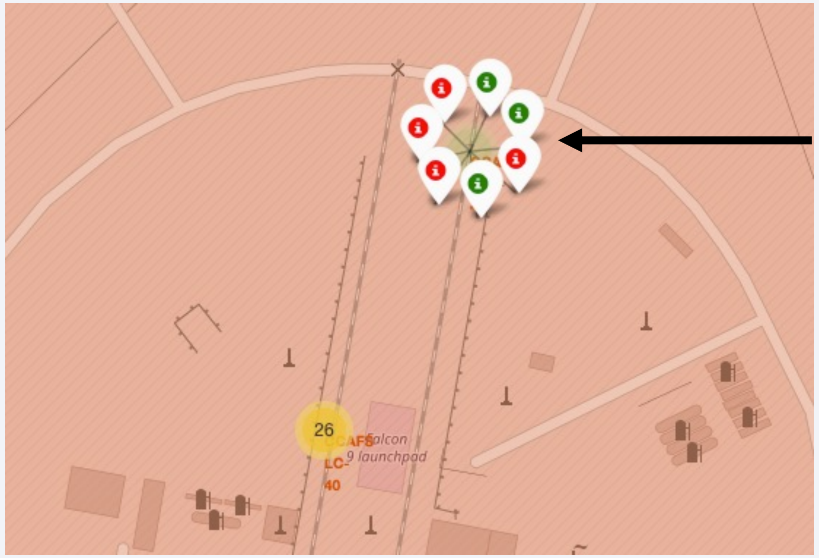
Color labeled Launch Outcome Markers

- Color labeled markers have been added for each Launch Site
- Markers for landing outcome success have been labeled green and for failure red
- As outcome markers all share common locations (i.e., launch site locations to which they belong), they have voluntarily been grouped into marker clusters and only display when users click on the cluster



Marker clusters for launch site CCAFS SLC-40 (top)

and CCAFS LC-40 (bottom)

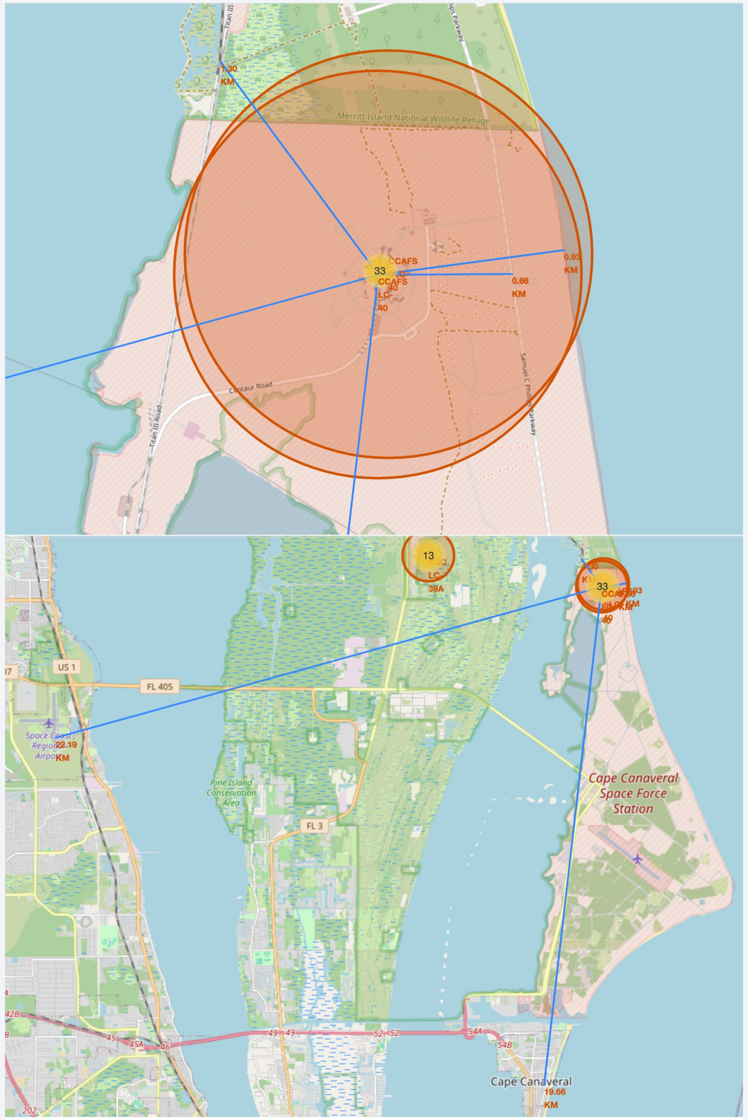


Color labeled markers for landing outcomes of launches from launch site CCAFS SLC-40

CCAFS SLC-40 location and proximities

- CCAFS SLC-40 is located on the coast facing the Atlantic ocean at a distance of 0.9 km from water
- The nearest highway is at a distance of 0.66 km
- The nearest train railway is at a distance of 1.3 km
- The nearest town is Cape Canaveral at a distance of 19.66 km
- The nearest airport is at Space Coast Region airport at a distance of 22.19 km

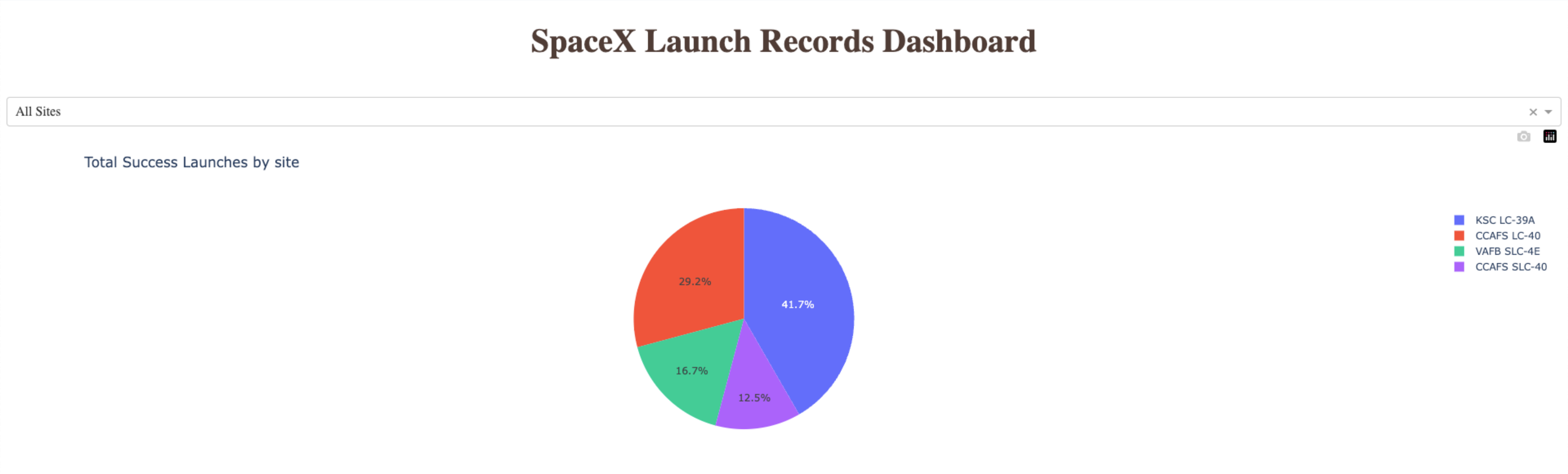
We can deduce that launch sites need to be close to coast lines , with close access routes granted to launch sites (most likely for equipment supply) and that launch sites must be far enough from cities/airports to not disturb or endanger civils.



Section 4

Build a Dashboard with Plotly Dash

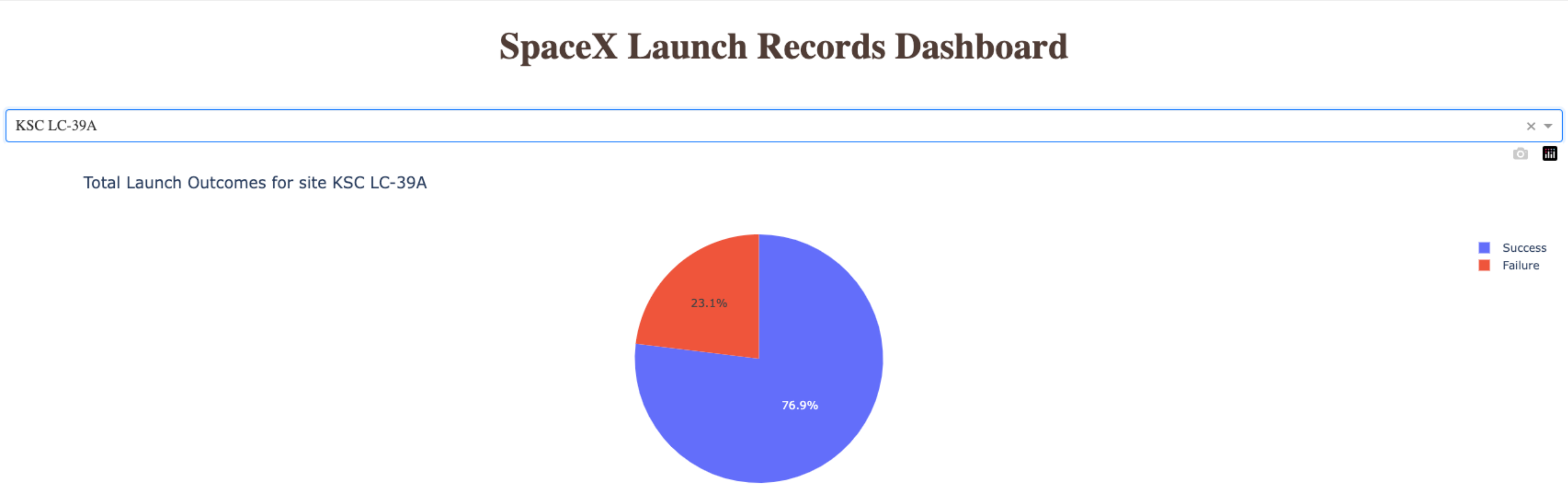
SpaceX Launch Records – Total successful launches



As displayed in the pie chart above, out of 24 successful launches :

- KSC LC-39A has the biggest share of success with 41.7% of them (10 successful launches)
- CCAFS LC-40 has a third of the success (7 successful launches)
- VAFB SLC-4E and CCAFS SLC-40 both have similar share of 16.7% and 12.5%

SpaceX Launch Records – KSC LC-39A launch outcomes

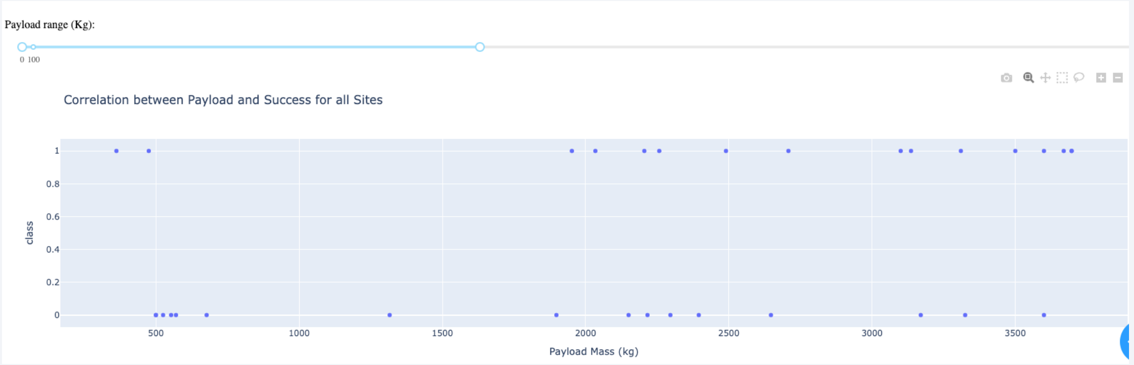


Explain the important elements and findings on the screenshot

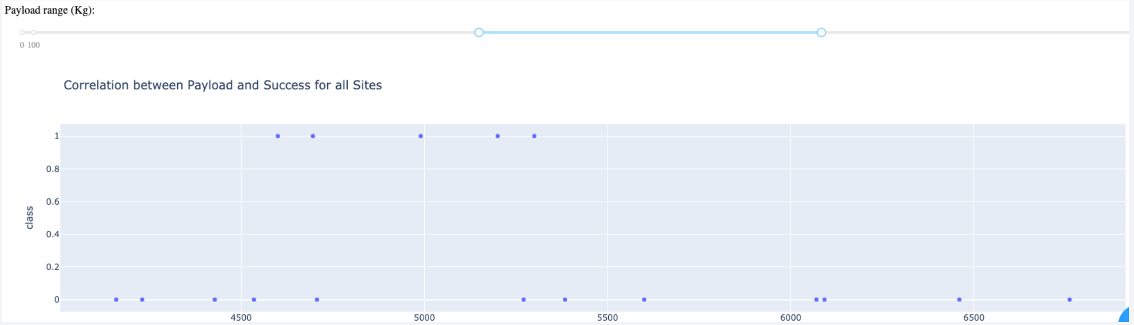
- The success rate of launch outcomes for launch site KSC LC-39A is of 76.9% meaning that only 1 in 4 launches has a failed outcome.
- The total launch number for this launch site is of 13, meaning 10 successes and 3 failures.

SpaceX Launch Records – Correlation between Payload and Success for all sites

- Payload Mass < 4000 kg
  - Launches with a payload mass between 500 and 2000 kg seem to be completely unsuccessful
  - While launches with a payload mass between 2000 and 4000 kg seem to have a 50% chance of being successful



- 4000 < Payload Mass < 7000 kg
  - Launches with a payload mass between 4000 and 4500 kg and 5400 and 7000kg seem to be completely failing
  - While launches between 4500 and 5400 kg payload seem to be successful



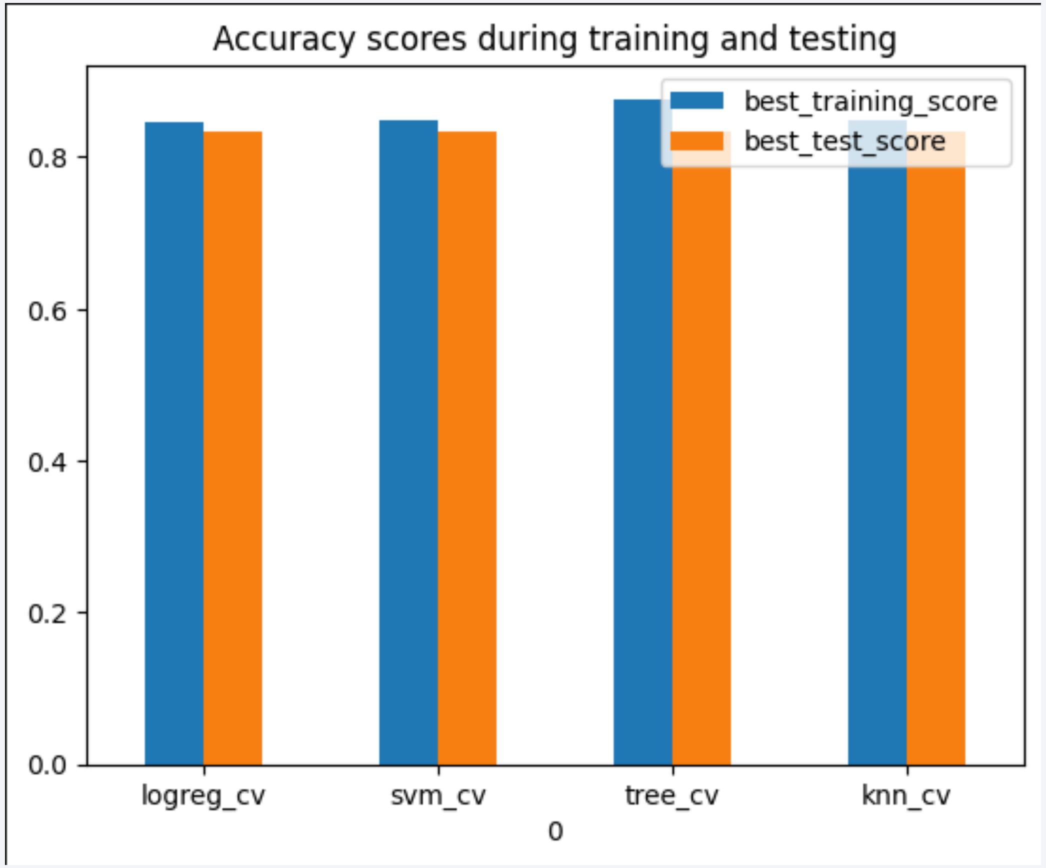
- Payload Mass > 7000 kg
  - We have data on only 2 launches with a payload over 7000kg in the dataset that we display
  - Since one was successful and the other one failed, we cannot really determine if this payload is riskier, as it currently has a 50/50 ratio

Section 5

Predictive Analysis (Classification)

Classification Accuracy

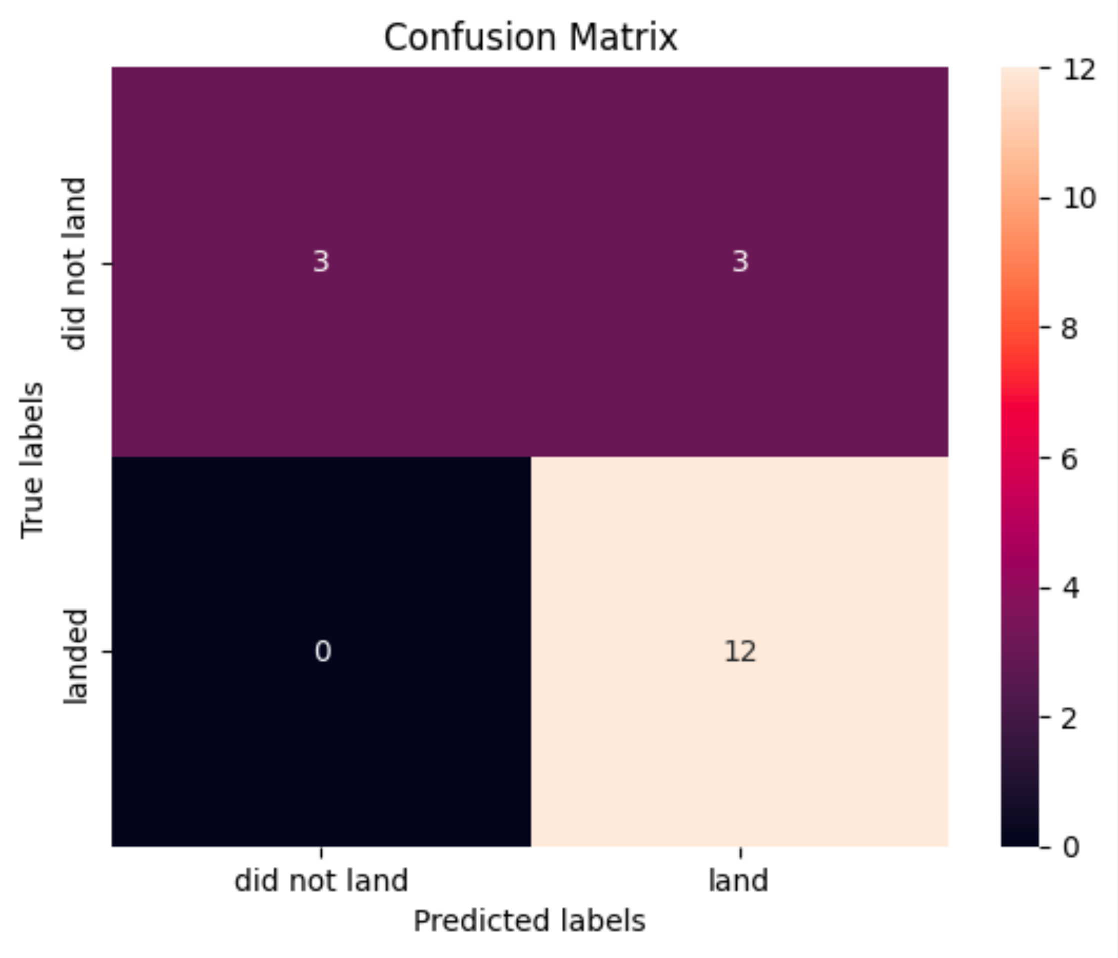
- The accuracy during testing for all 4 models is the same with a score of 83.3%
- This a consequence of having a small dataset, the testing data consists of only 16 records and the optimized models seem to fair all equally on the test sample.



Confusion Matrix

We can see that in the displayed confusion matrix for the Tree_cv model we find :

- 12 TP
- 3 TN
- 3 FP
- 0 FN



Conclusions

- Landing outcome for SpaceX's launches is related to several features, but the main ones are, launch site location, payload mass in kg, orbit type and booster version
- Over time the success rate of these launches has drastically improved which has played a big role in SpaceX's commercial success as it reduced their cost function
- SpaceX has been renting and using several launch sites, with different displays of success rate, KSC LC-39A being the most successful one
- Overall launch sites share common attributes, being located on the coast and close to supply infrastructures like train railways and highway routes, but with enough distance to cities and airports that they wouldn't disturb or endanger civilians
- The model training and testing has been done on a very small dataset which makes it difficult for any of the tested models to prevail, however the choice was settled on Tree_cv which had the best training_score
- With an accuracy of 83% on the test data, Tree_cv should be able to predict with high accuracy if launches will result in landing success of stage 1 for the rockets of company SpaceY

Appendix

- Include any relevant assets like Python code snippets, SQL queries, charts, Notebook outputs, or data sets that you may have created during this project

Thank you!