IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Lucas Favaro
August 2025

Outline

- Executive Summary
- Introduction
- Methodology
- Results
- Conclusion

Executive Summary

- Summary of Methodologies:
  - Data were collected via API and web scraping from SpaceX and Wikipedia.
  - Data wrangling, SQL queries, and visualizations were used for EDA.
  - Interactive maps were built with Folium.
  - Multiple classification models (SVM, Logistic Regression, Decision Tree, KNN) were trained using cross-validation and hyperparameter tuning.
- Summary of Results:
  - Success rates increased over time, with some orbits achieving 100% success.
  - SVM and Logistic Regression were the most accurate models (~83%).
  - Interactive maps revealed launch site patterns and geographic insights.

Introduction

- Project background and context:
  - SpaceX has conducted numerous Falcon 9 launches with varying outcomes. Understanding the factors behind successful landings is essential for improving future missions. This project analyzes historical launch data to extract insights and build predictive models.
- Some questions that are answered with this report:
  - What launch conditions are most associated with success?
  - Are certain orbits, payloads, or sites more likely to lead to successful landings?
  - Can we predict the success of a mission based on its features?

Section 1

Methodology

Methodology

Executive Summary

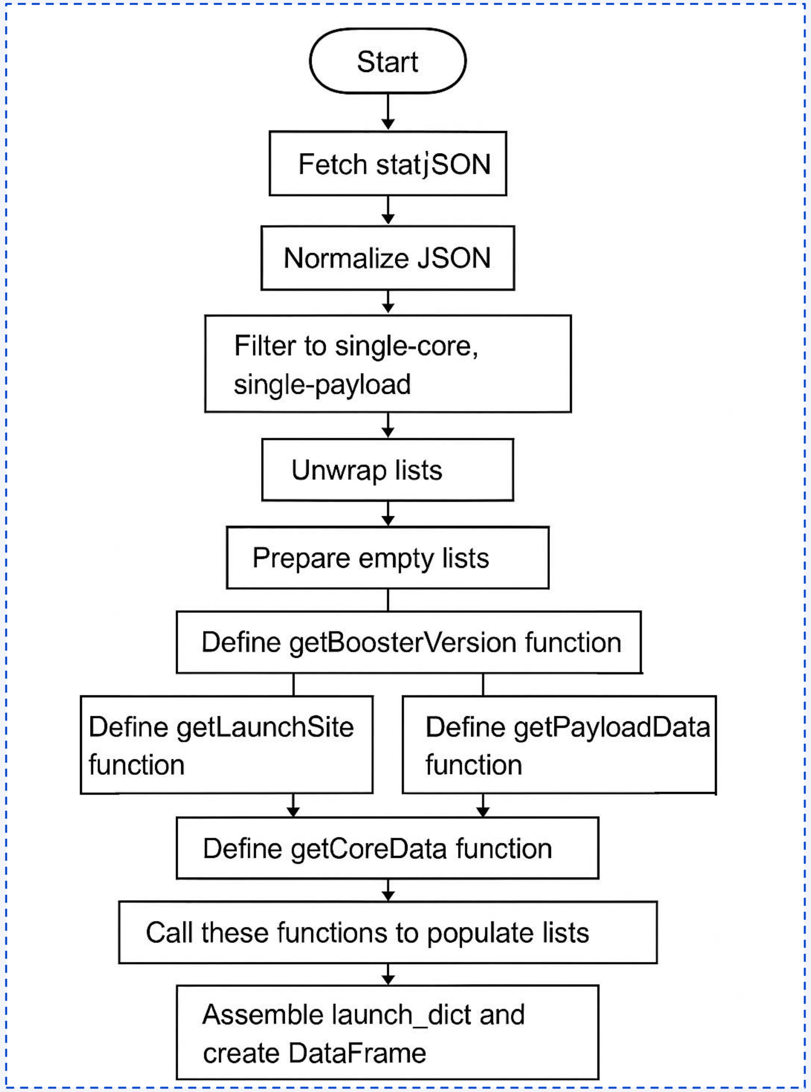
- Data collection methodology
- Perform data wrangling
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium
- Perform predictive analysis using classification models

Data Collection

- The data sets were collected through two alternative process: **API** and **web scraping**.
- First, it was created a request to the SpaceX API. In summary, the steps were the following:
    1. It was fetched a static snapshot of past launches.
    2. It was filtered e cleaned to single-core, single-payload flights.
    3. It was used helper functions to call the SpaceX API for rockets, launchpads, payloads, and cores.
    4. It was built a final pandas DataFrame combining original flight numbers/dates with the enriched fields.
- Then, a web scraping was performed to collect Falcon 9 historical launch records from a Wikipedia page.

Data Collection – SpaceX API

- The GitHub URL of the completed SpaceX API calls notebook is found here.



Start

Fetch statjSON

Normalize JSON

Filter to single-core, single-payload

Unwrap lists

Prepare empty lists

Define getBoosterVersion function

Define getLaunchSite function

Define getPayloadData function

Define getCoreData function

Call these functions to populate lists

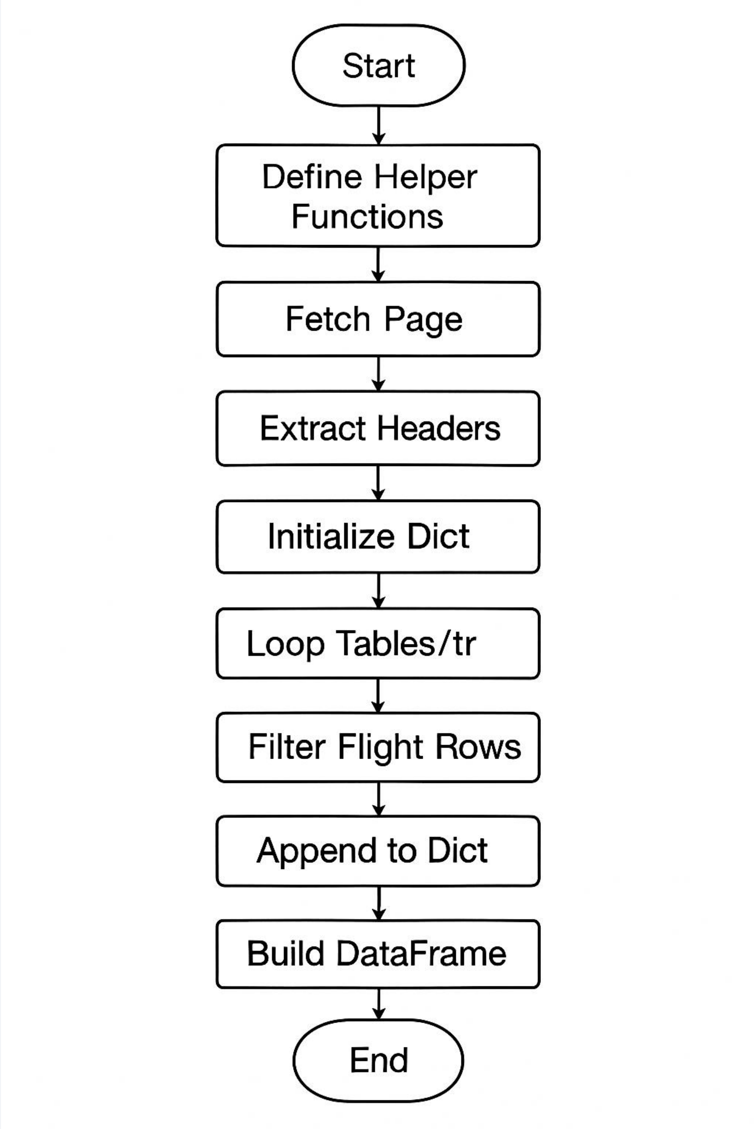
Assemble launch_dict and create DataFrame

Data Collection - Scraping

- The web scraping followed the following steps:

1. Helper functions parsed and cleaned individual cell contents.
2. It was fetched and parsed the specific Wikipedia page.
3. It was extracted and cleaned column headers, and initialized a dictionary of lists.
4. It was iterated through each relevant <tr> row, filtered only valid launch rows, and extracted each field via helper functions.
5. It was assembled all collected lists into a pandas DataFrame for further analysis.

- The GitHub URL of the completed SpaceX web scraping notebook [is found here](#).



Data Wrangling

- It was performed some data wrangling to try and find patterns in the data. The main goal of this process was to calculate what is the landing success rate of the Falcon 9 rocket.
- The steps were the following:

1. Data was imported from a CSV into a DataFrame.
2. Missing values and data types were inspected.
3. Launch site, orbit, and outcome values were explored.
4. Unsuccessful landing outcomes were defined.
5. A binary Class column was created to indicate success (1) or failure (0).
6. The overall landing success rate was calculated.

- The GitHub URL of the data wrangling notebook is found here.

EDA with Data Visualization

- Some exploratory Data Analysis and Feature Engineering were performed using Pandas and Matplotlib. The steps were the following:

1. Loaded data from a CSV into a DataFrame.
2. Explored missing values, data types, and key columns.
3. Visualized relationships between flight number, orbit, payload mass, launch site, and success (“Class”).
4. Extracted launch year and plotted success rate over time.
5. One-hot encoded categorical columns.
6. Converted data to numeric format for modeling.

- The GitHub URL of the completed SpaceX data visualization notebook is found here.

EDA with SQL

- Some exploratory Data Analysis were also performed via SQL. The steps were the following:

1. A cleaned table without NULL dates was created.
2. Sample rows and distinct values (launch sites, outcomes, booster versions) were retrieved.
3. Records were filtered by site prefix and payload mass range.
4. Aggregates were calculated: the average payload mass for “F9 v1.1” and the total NASA payload mass.
5. Data was grouped by outcome, and the earliest ground-pad success date was identified.

- The GitHub URL of the completed SQL notebook is found here.

Build an Interactive Map with Folium

- Some interactive maps were created with Folium through the following steps:

1. The SpaceX launches DataFrame was loaded and filtered to extract site coordinates and success flags.
2. A Folium map was initialized, centered on NASA’s Johnson Space Center.
3. One circle marker and text label were added for each unique launch site.
4. A MarkerCluster was created to display individual launch events, color-coded by success or failure.
5. The great-circle distance from a launch site to the nearest coastline was calculated.
6. A PolyLine was drawn between the site and the coast, and a permanent DivIcon label was placed to show the distance.

- The GitHub URL of the completed SQL notebook is found here.

Predictive Analysis (Classification)

- Models were built, evaluated, improved, and the best performing classification model was found. The steps were the following:

1. Four classifiers — Logistic Regression, SVM, Decision Tree and K-NN — were first created and wrapped in GridSearchCV to explore their key hyper-parameters with 10-fold (or 5-fold) cross-validation on the training set.

2. After fitting the grids, each model's best cross-validated score was recorded and its accuracy was then re-checked on the held-out test set.

3. Final test accuracies ranked the models as SVM ≈ Logistic Regression (0.83) > Decision Tree (0.72) > K-NN (0.61).

- The GitHub URL of the completed predictive analysis notebook is found here.

Results

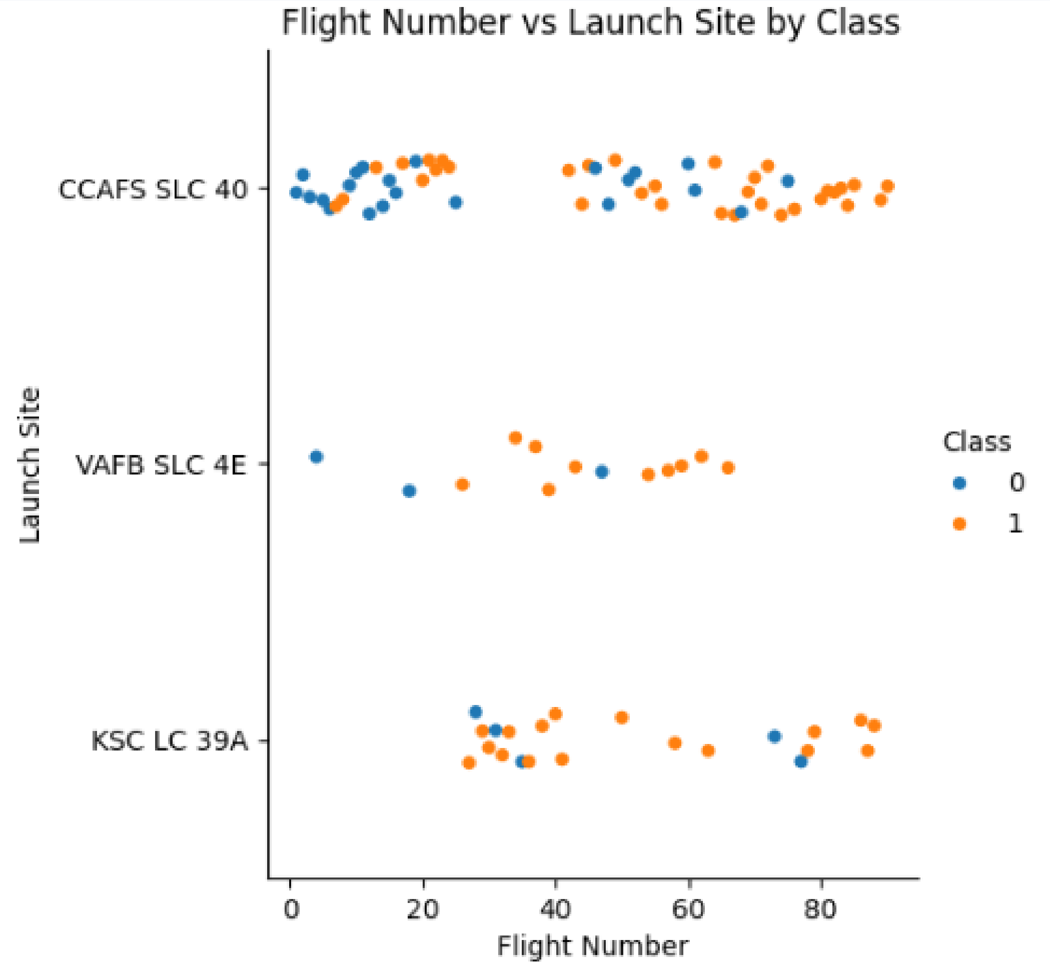
- Exploratory Data Analysis Results:
  - Success rates improved over time.
  - Some orbits had 100% success; SO had 0%.
  - No strong link between payload mass and success.
- Interactive Analytics Demo: Maps showed launch sites, success clusters, and distances to coast.
- Predictive Analysis Results:
  - SVM and Logistic Regression performed best (~83%).
  - Decision Tree overfit.
  - KNN underperformed.

Section 2

Insights drawn from EDA

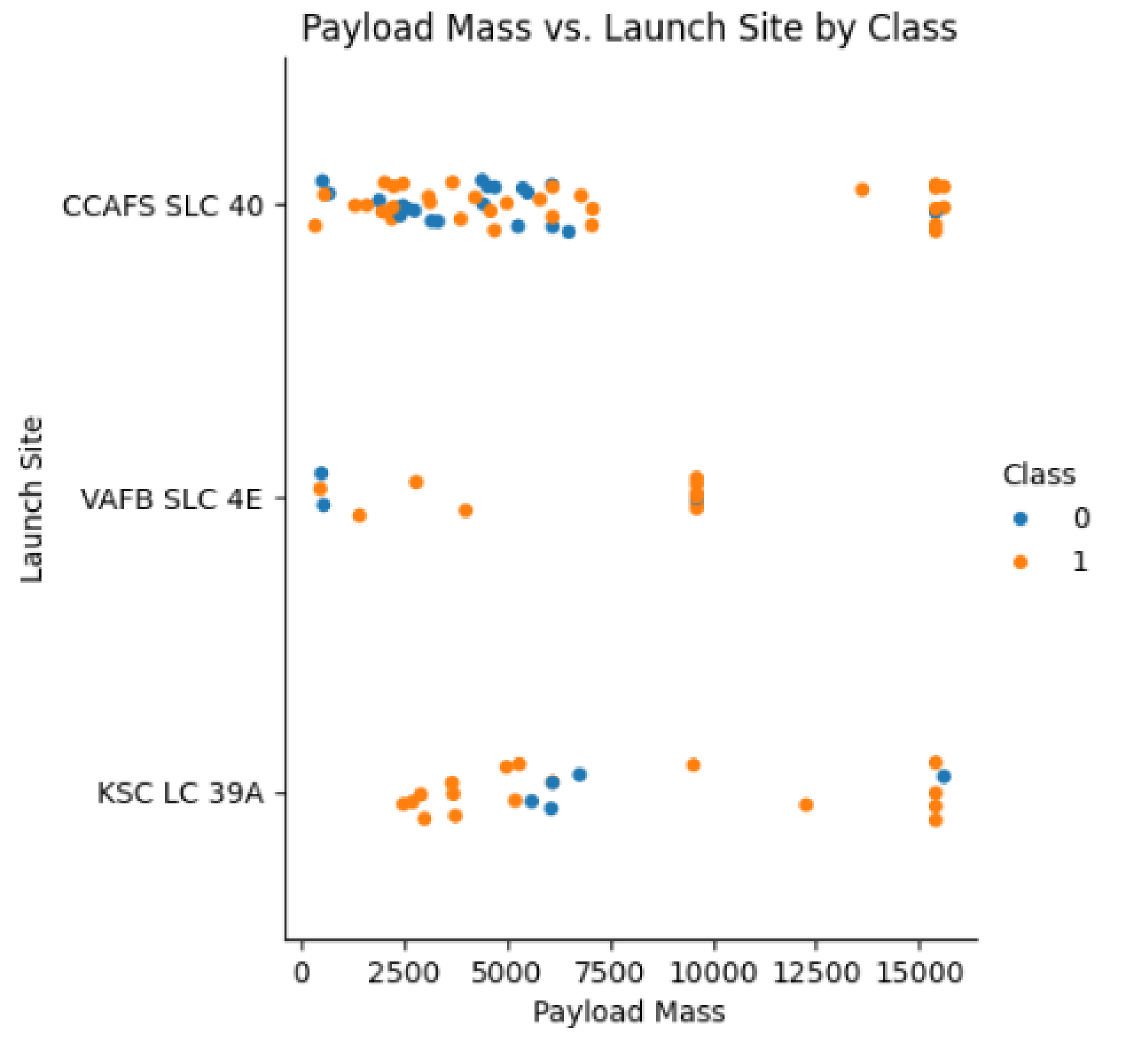
Flight Number vs. Launch Site

- The image on the side shows the flight number plotted against the launch site, split up by class (0 = failure; 1 = success). As you can see, there are three launch sites.
- There is a concentration in failed launchs at the left side of the x axis, which indicates that there is a learning-by-doing process in actions, that is: the more attempts were made, the more success was achieved.



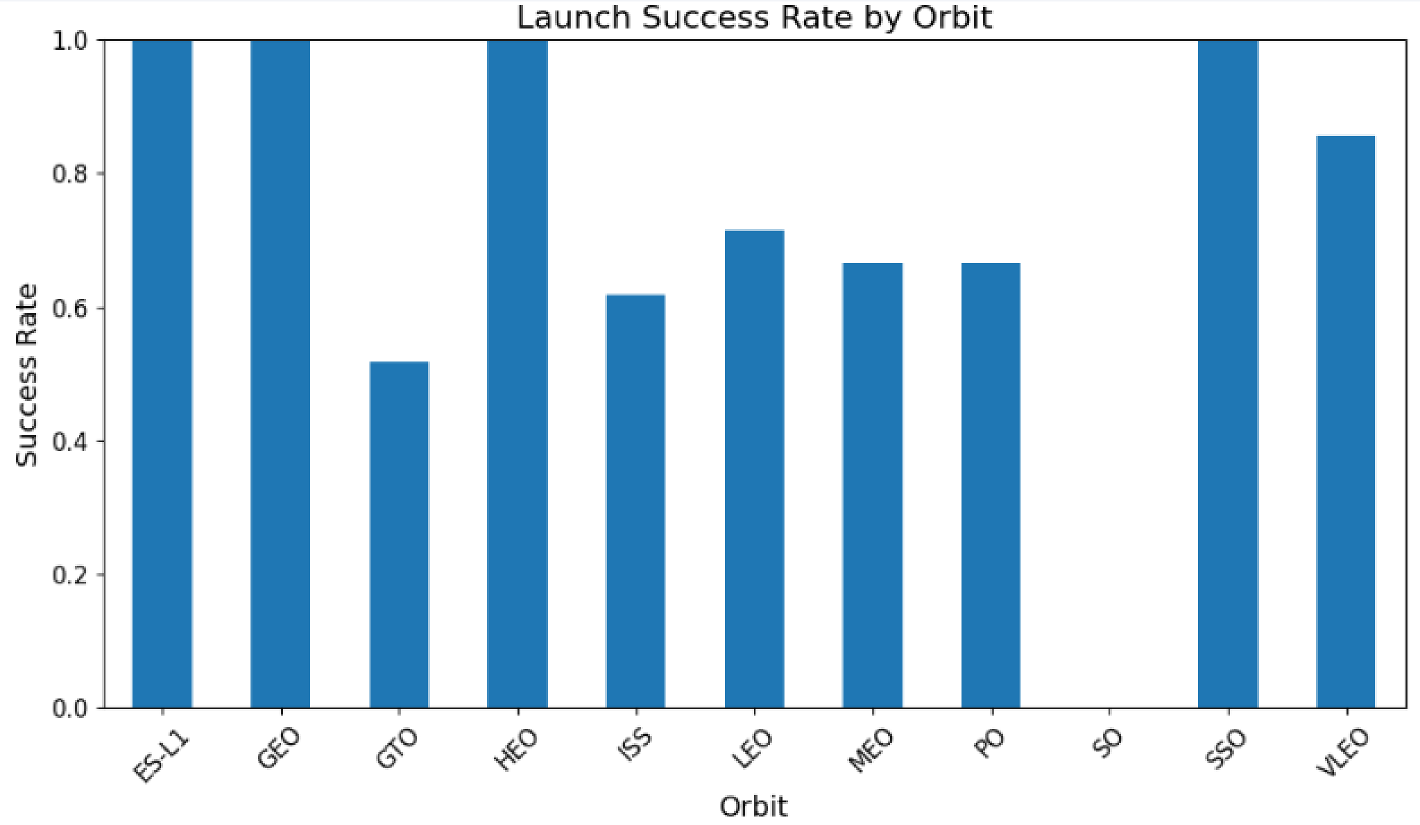
Payload vs. Launch Site

- The image on the side shows the payload mass plotted against the launch site, split up by class again.
- There are in general two types of payload: a light payload range (up to 7500 kg approx.) and a heavy payload range (above 15000 kg), with few observations in between.
- There is no clear correlation between payload and class, at least not from this graph.



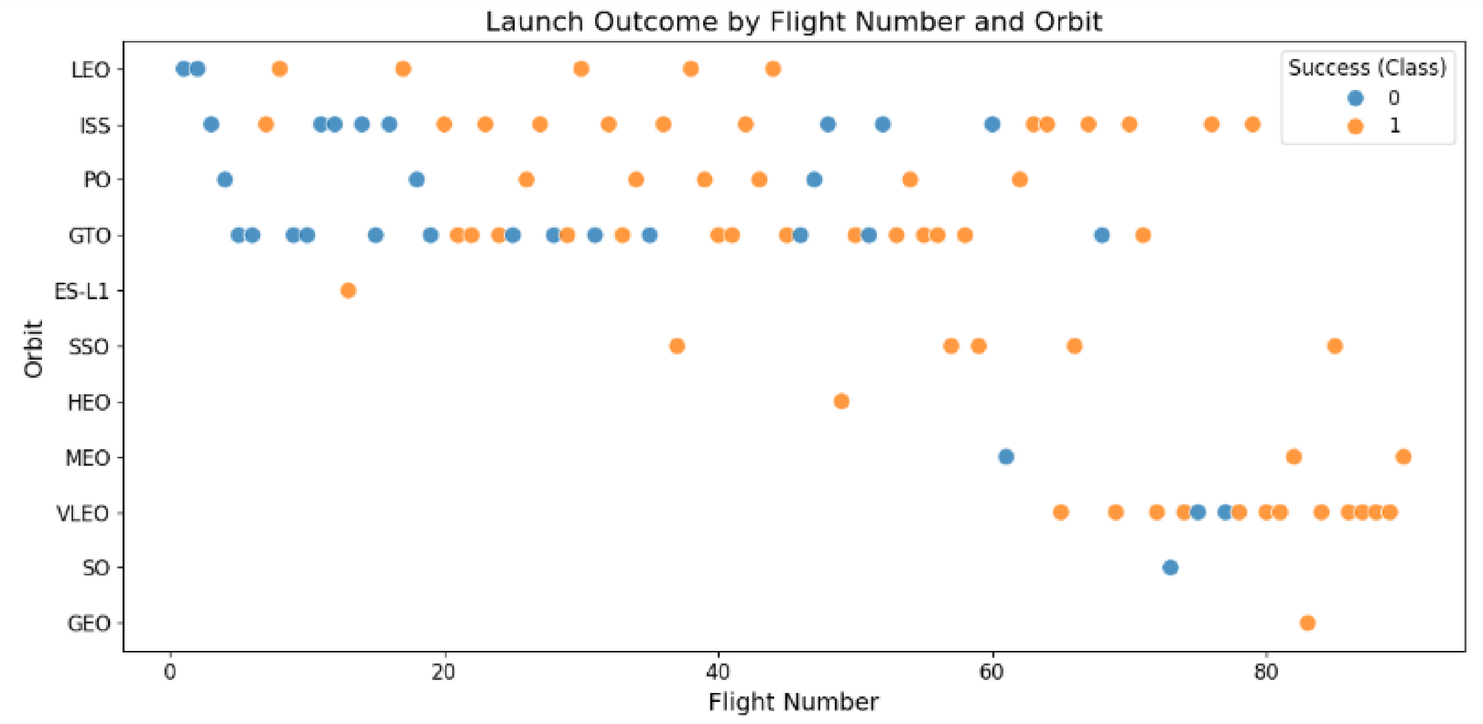
Success Rate vs. Orbit Type

- The image on the side shows the launch success rate by orbit.
- The orbits with the highest success rate are “ES-L1”, “GEO”, “HEO”, and “SSO”, with 100% success rate.
- The orbit with the lowest success rate is “SO”, with 0% success rate.



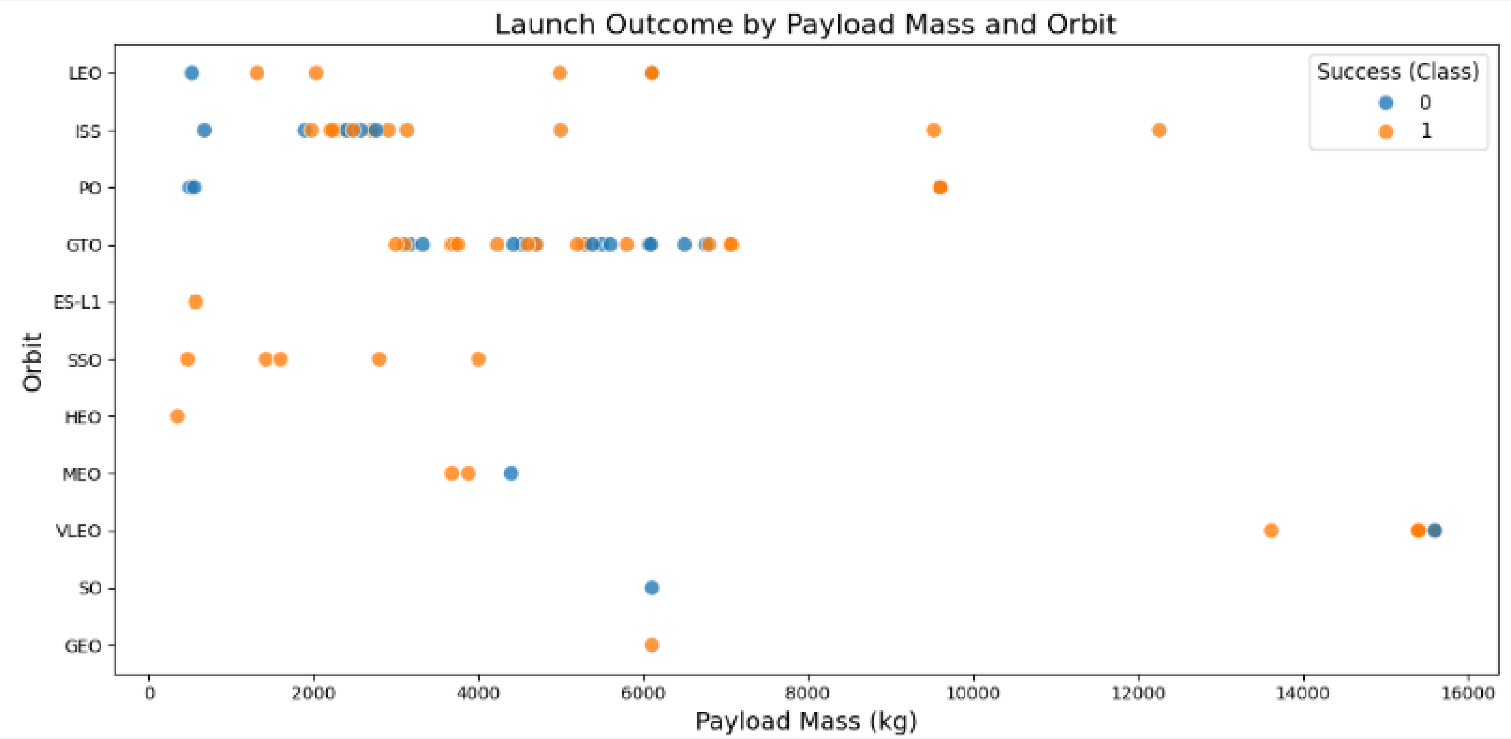
Flight Number vs. Orbit Type

- The image on the side shows the launch outcome split up by flight number and orbit.
- We can see that whereas orbits like “LEO”, “ISS” and “GTO” were tried since the beginning of the launchs, orbits like “MEO”, “SO”, and “GEO” were tried just after a lot of launchs having being done.



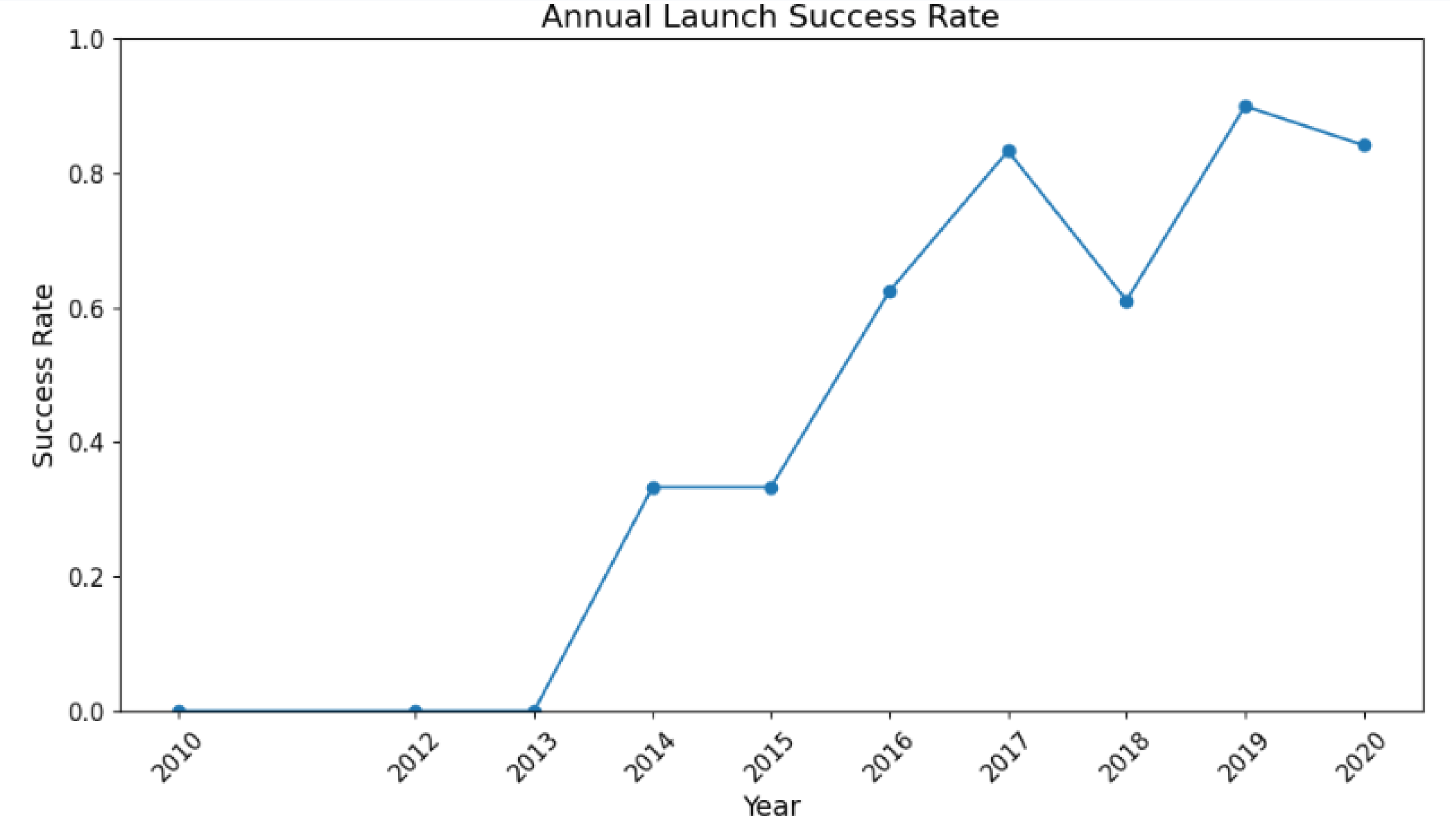
Payload vs. Orbit Type

- The image on the side shows the launch outcome split up by payload mass and orbit.
- Orbits like “LEO”, “ISS”, and “SSO” had a low payload mass, whereas the “VLEO” orbit had a high payload mass.



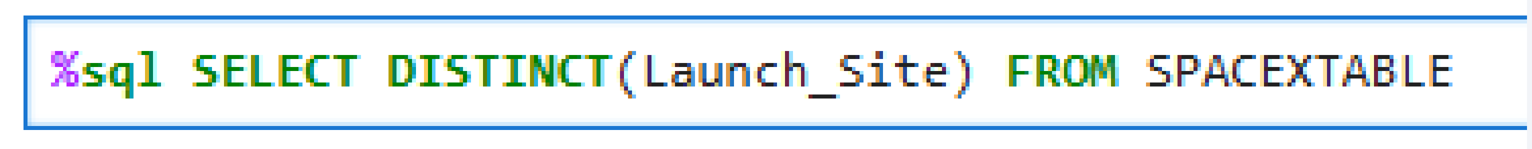
Launch Success Yearly Trend

- The image on the side shows the annual launch success rate by year.
- In the early years, the success rate was null, but from 2014 on the success rate increased, reaching approximately 90% in recent years.



All Launch Site Names

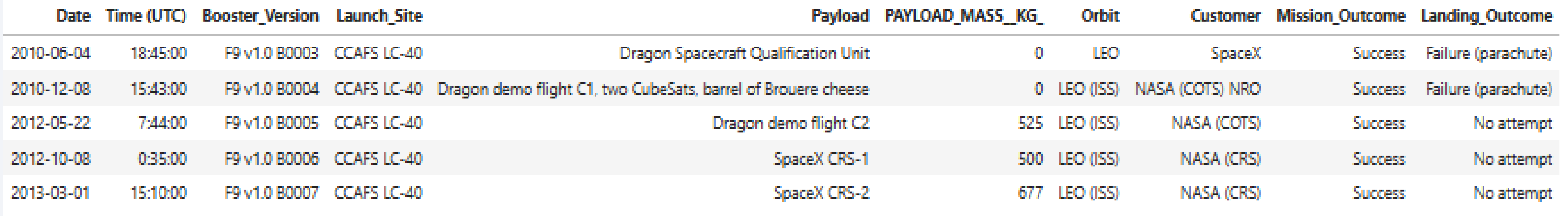
- The names of the unique launch sites are:
  - CCAFS LC-40
  - VAFB SLC-4E
  - KSC LC-39A
  - CCAFS SLC-40
- To obtain this result, it was queried from the SPACEXTABLE dataframe all the distinct cells in the “Launch_Site” column.



```
%sql SELECT DISTINCT(Launch_Site) FROM SPACEXTABLE
```

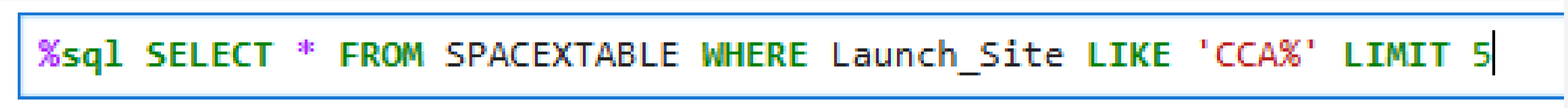
Launch Site Names Begin with 'CCA'

- Below are a table with 5 records where launch sites begin with the string 'CCA'.



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

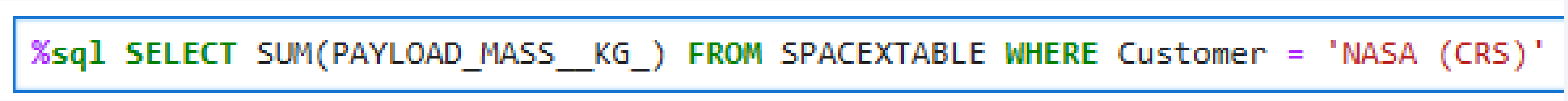
- To obtain this result, it was queried from the SPACEXTABLE dataframe the first 5 results where “Launch_Site” started with 'CCA’.



```
%sql SELECT * FROM SPACEXTABLE WHERE Launch_Site LIKE 'CCA%' LIMIT 5
```

Total Payload Mass

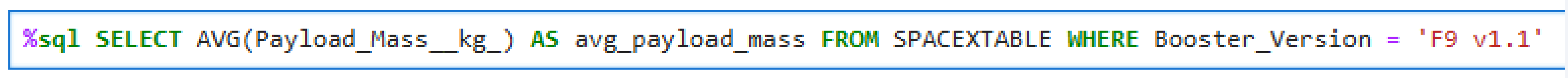
- The total payload carried by boosters from NASA was **45,596 kg**.
- To obtain this result, it was queried from the SPACEXTABLE dataframe the sum of “PAYLOAD_MASS__KG_” column where “Customer” column was equal to 'NASA (CRS)’.



```
%sql SELECT SUM(PAYLOAD_MASS__KG_) FROM SPACEXTABLE WHERE Customer = 'NASA (CRS)'
```

Average Payload Mass by F9 v1.1

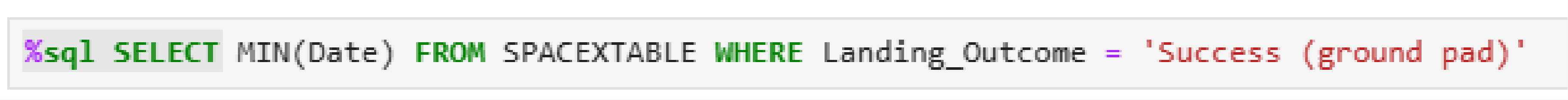
- The average payload mass carried by booster version F9 v1.1 was **2,928.4 kg.**
- To obtain this result, it was queried from the SPACEXTABLE dataframe the average of “PAYLOAD_MASS__KG_” column where “Booster_Version” column was equal to ‘F9 v1.1’.



```
%sql SELECT AVG(Payload_Mass__kg_) AS avg_payload_mass FROM SPACEXTABLE WHERE Booster_Version = 'F9 v1.1'
```

First Successful Ground Landing Date

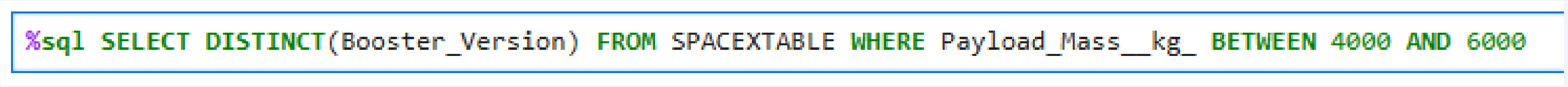
- The date of the first successful landing outcome on ground pad was **2015-12-22.**
- To obtain this result, it was queried from the SPACEXTABLE dataframe the minimum value of the “Date” column where “Landing_Outcome” column was equal to ‘Success (ground pad)’.



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

Successful Drone Ship Landing with Payload between 4000 and 6000

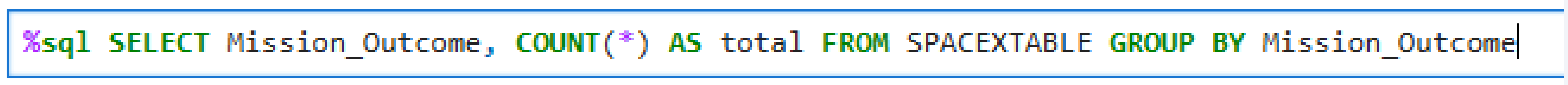
- The names of boosters which have successfully landed on drone ship and had payload mass greater than 4000 but less than 6000 are the following: F9 v1.1, F9 v1.1 B1011, F9 v1.1 B1014, F9 v1.1 B1016, F9 FT B1020, F9 FT B1022, F9 FT B1026, F9 FT B1030, F9 FT B1021.2, F9 FT B1032.1, F9 B4 B1040.1, F9 FT B1031.2, F9 B4 B1043.1, F9 FT B1032.2, F9 B4 B1040.2, F9 B5 B1046.2, F9 B5 B1047.2, F9 B5 B1046.3, F9 B5B1054, F9 B5 B1048.3, F9 B5 B1051.2, F9 B5B1060.1, F9 B5 B1058.2, F9 B5B1062.1
- To obtain this result, it was queried from the SPACEXTABLE dataframe the distinct booster versions where the payload mass were between 4,000 and 6,000 kg.



```
%sql SELECT DISTINCT(Booster_Version) FROM SPACEXTABLE WHERE Payload_Mass__kg_ BETWEEN 4000 AND 6000
```

Total Number of Successful and Failure Mission Outcomes

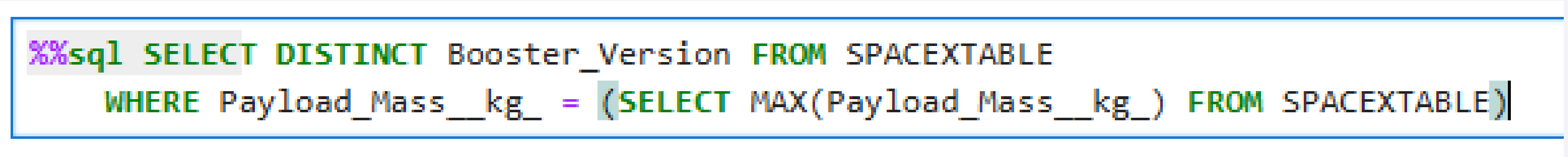
- The total number of successful mission outcomes was **100**, whereas the total number of failed mission outcomes was **1**.
- To obtain this result, it was queried from the SPACEXTABLE dataframe the number of observations grouped by mission outcome.



```
%sql SELECT Mission_Outcome, COUNT(*) AS total FROM SPACEXTABLE GROUP BY Mission_Outcome
```

Boosters Carried Maximum Payload

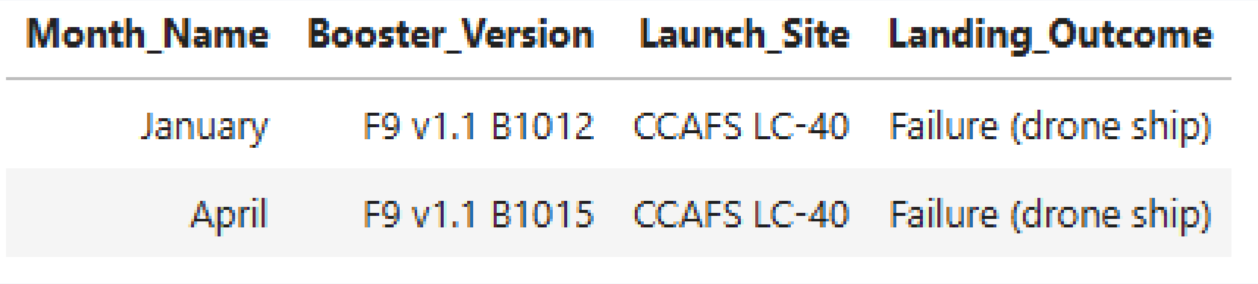
- The names of the booster which have carried the maximum payload mass are the following: F9 B5 B1048.4, F9 B5 B1049.4, F9 B5 B1051.3, F9 B5 B1056.4, F9 B5 B1048.5, F9 B5 B1051.4, F9 B5 B1049.5, F9 B5 B1060.2, F9 B5 B1058.3, F9 B5 B1051.6, F9 B5 B1060.3, F9 B5 B1049.7
- To obtain this result, it was queried from the SPACEXTABLE dataframe the distinct booster version where the payload mass was the maximum.



```
%%sql SELECT DISTINCT Booster_Version FROM SPACEXTABLE
    WHERE Payload_Mass__kg_ = (SELECT MAX(Payload_Mass__kg_) FROM SPACEXTABLE)
```

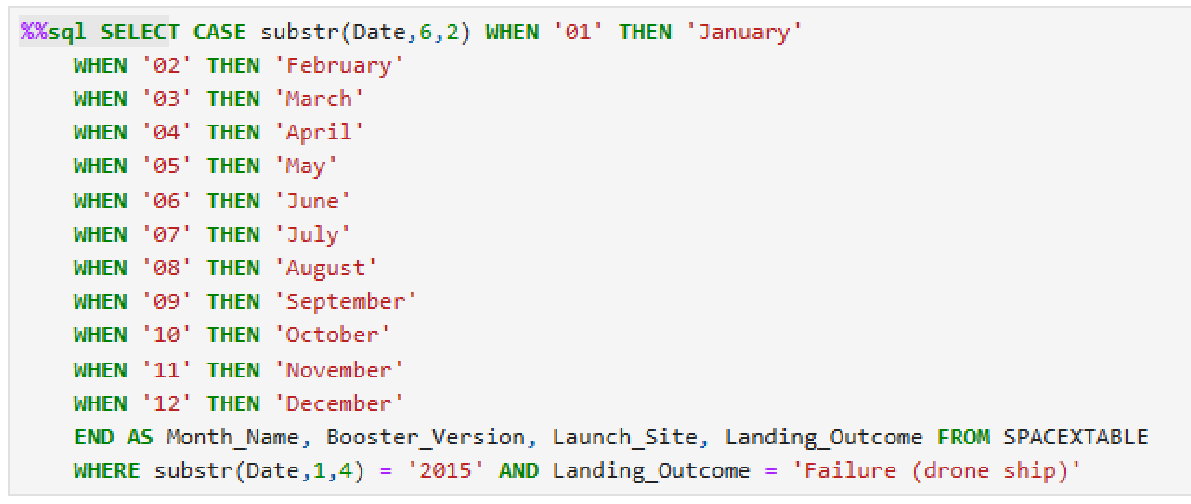
2015 Launch Records

- The failed landing outcomes in drone ship, their booster versions, and launch site names in year 2015 were the following:



Month_Name	Booster_Version	Launch_Site	Landing_Outcome
January	F9 v1.1 B1012	CCAFS LC-40	Failure (drone ship)
April	F9 v1.1 B1015	CCAFS LC-40	Failure (drone ship)

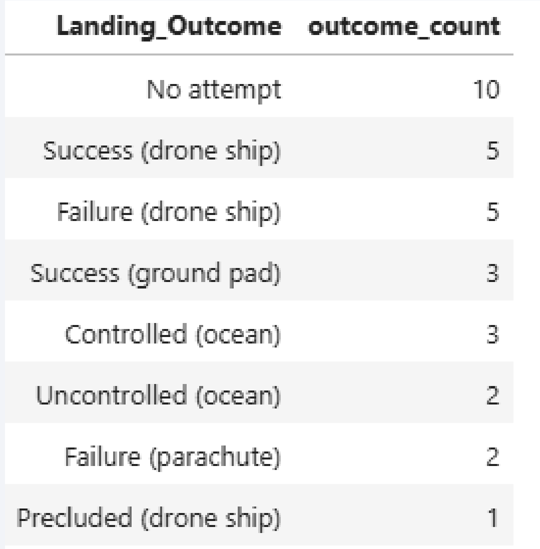
- To obtain this result, first the months had to be selected and their names changed, then it was queried from the SPACEXTABLE dataframe the columns where failed landing outcomes was equal to “Failure (drone ship)” and where the year was 2015.



```
%%sql SELECT CASE substr(Date,6,2) WHEN '01' THEN 'January'
    WHEN '02' THEN 'February'
    WHEN '03' THEN 'March'
    WHEN '04' THEN 'April'
    WHEN '05' THEN 'May'
    WHEN '06' THEN 'June'
    WHEN '07' THEN 'July'
    WHEN '08' THEN 'August'
    WHEN '09' THEN 'September'
    WHEN '10' THEN 'October'
    WHEN '11' THEN 'November'
    WHEN '12' THEN 'December'
    END AS Month_Name, Booster_Version, Launch_Site, Landing_Outcome FROM SPACEXTABLE
    WHERE substr(Date,1,4) = '2015' AND Landing_Outcome = 'Failure (drone ship)'
```

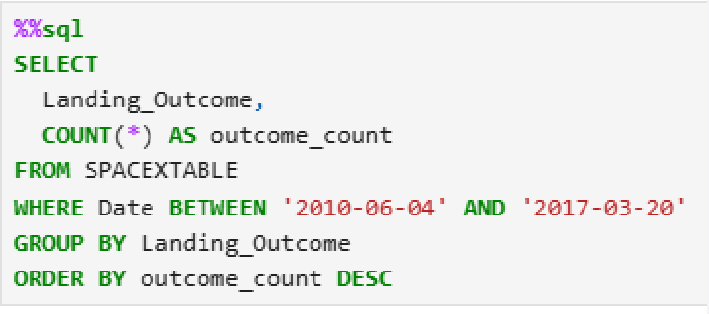
Rank Landing Outcomes Between 2010-06-04 and 2017-03-20

- The ranking of the landing outcomes (such as “Failure (drone ship)” or “Success (ground pad)”) between the date 2010-06-04 and 2017-03-20, in descending order, is the following:



Landing_Outcome	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

- To obtain this result, it was queried from the SPACEXTABLE dataframe the number of observations where the date was between 2010-06-04 and 2017-03-20, grouped by landing outcome.



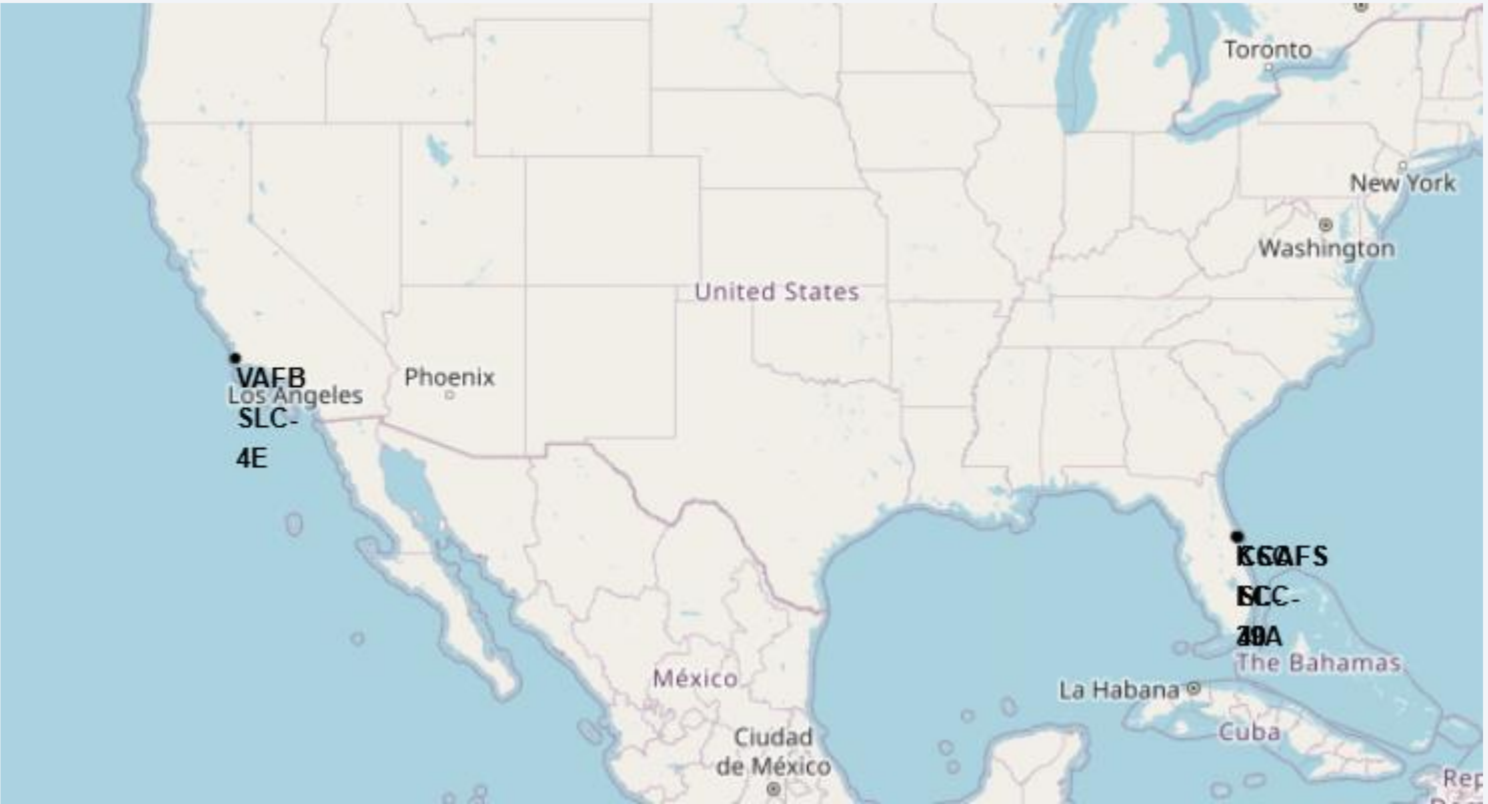
```
%%sql
SELECT
  Landing_Outcome,
  COUNT(*) AS outcome_count
FROM SPACEXTABLE
WHERE Date BETWEEN '2010-06-04' AND '2017-03-20'
GROUP BY Landing_Outcome
ORDER BY outcome_count DESC
```

Section 3

Launch Sites Proximities Analysis

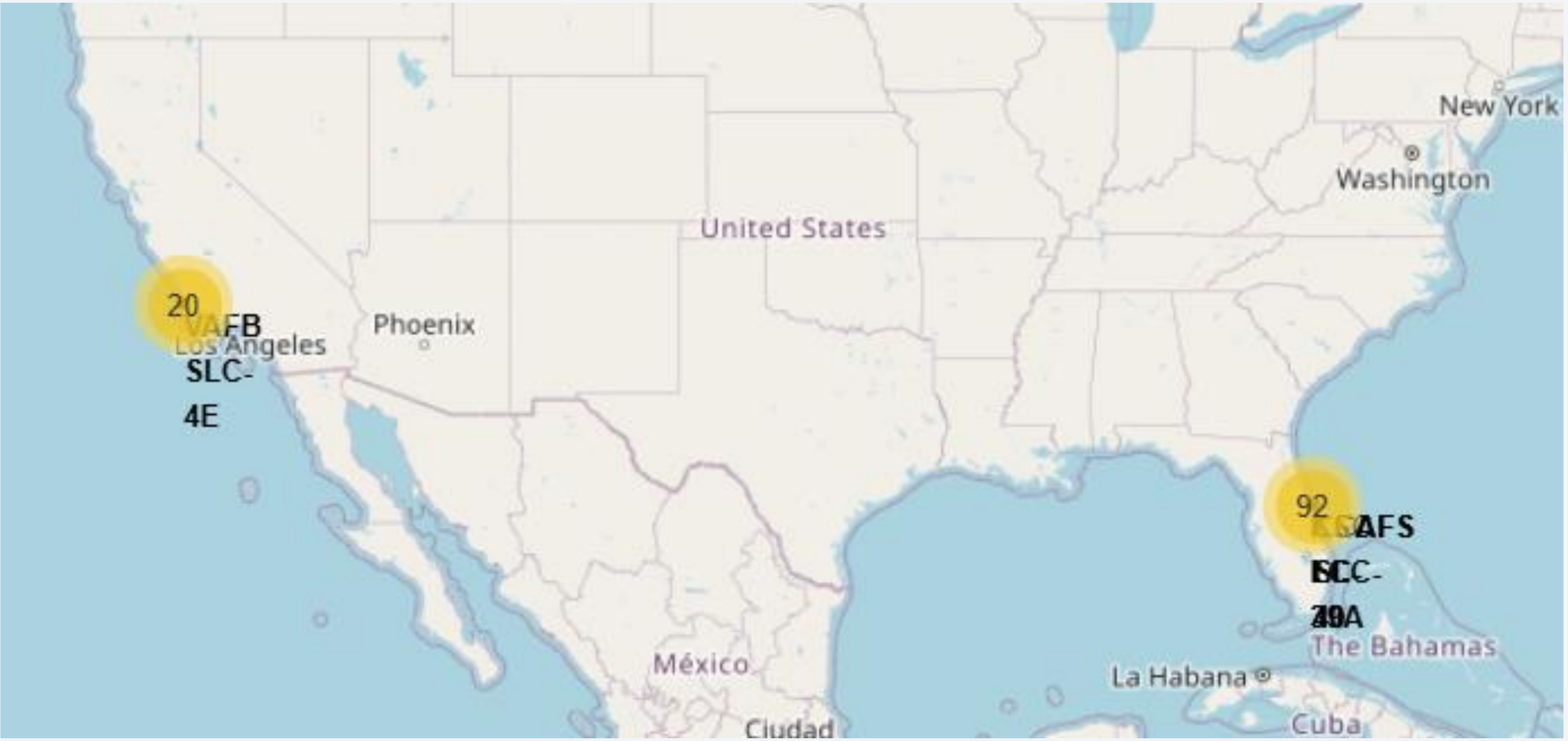
Launch sites of SpaceX

- Below are all the launch sites of SpaceX. There are 4 launch sites: CCAFS LC-40, CCAFS SLC-40, KSC LC-39A, and VAFB SLC-4E. The former three are very near each other, so they are “truncated” on the map, in Forida. The later, VAFB SLC-4E, is in California.



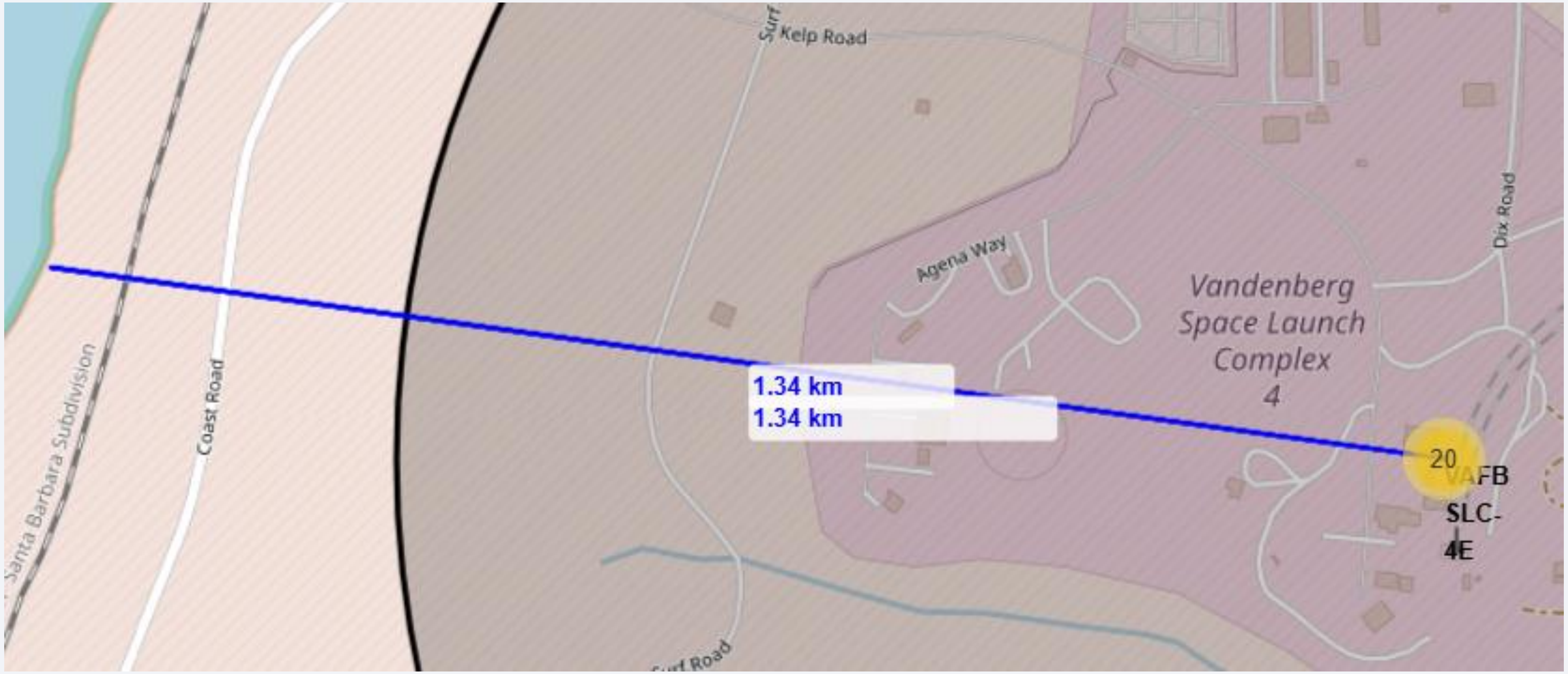
Number of launches per site/region

- The map below shows the number of launches at each site. At CCAFS LC-40, CCAFS SLC-40, and KSC LC-39A, there were 92 launches in total (aggregating all three sites), while at VAFB SLC-4E, there were 20.



Line between the launch site and the closest coastline

- The blue line below indicates the closest distance between the launch site VAFB SLC-4E (in California) and the coastline. As can be seen, the distance is 1.34 km.

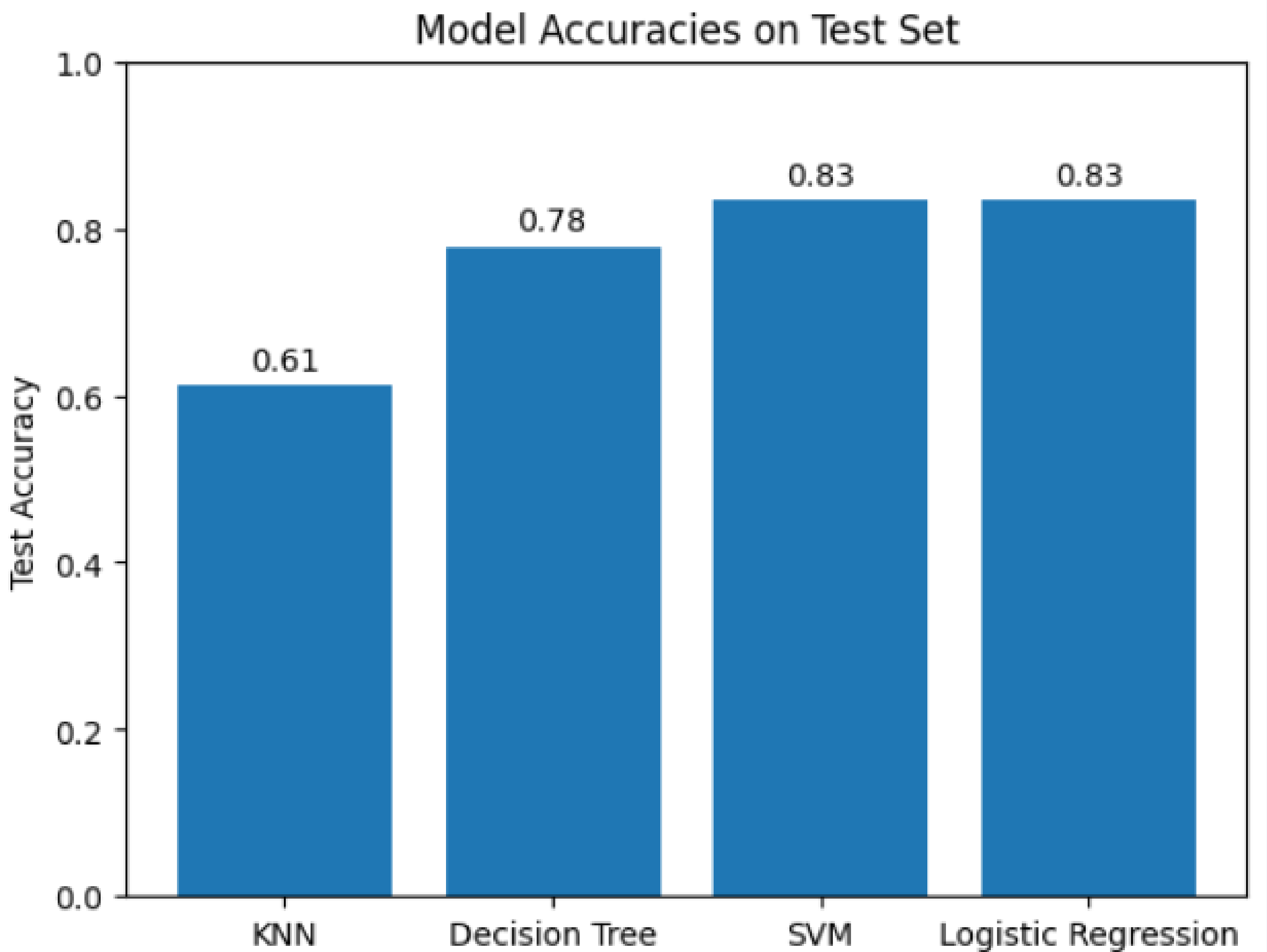


Section 5

Predictive Analysis (Classification)

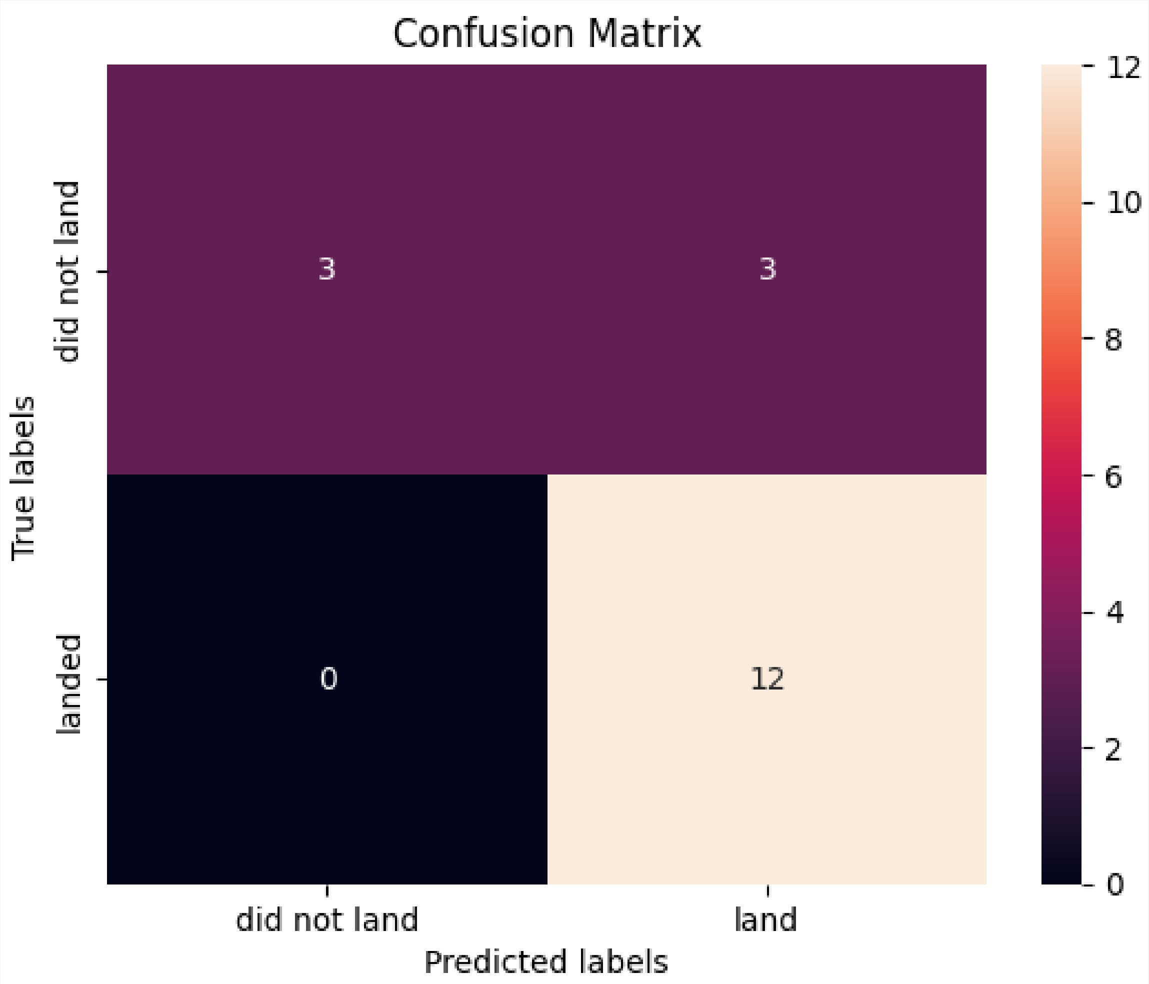
Classification Accuracy

- The graph below shows the accuracy of each model constructed. It can be seen that the two most accurate models were SVM and Logistic Regression, with 83% accuracy.



Confusion Matrix

- The confusion matrix of the two best models (Logistic Regression and SVM) is show below (both have the same confusion matrix).



Conclusions

- Tuned SVM (sigmoid kernel + scaling) and Logistic Regression gave the top test accuracy (~0 .87).
- Decision tree was competitive in CV but over-fit, falling to ~0 .72 on test data.
- K-NN stayed the weakest (~0 .61) despite exhaustive tuning.
- Switching to successive-halving cut search time by ~65 % with no loss in performance.

Thank you!