Winning the Space Race

The Data Science Way

Debatreyo Roy
25 May 2024

Outline

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

Executive Summary

Methodologies

- **Data collection** using **SpaceX REST API** (third-party, open-source) and **web scraping** tools.
- **Data wrangling** to clean, process, and perform **feature engineering** with data.
- **Exploring** and **visualizing** the prepared dataset for gaining better insights.
- **Spatial analysis** of launch sites to understand how **location** influences mission outcomes.
- Build, train, and evaluate **machine learning models** to predict launch outcome based on multiple input factors.
- Find the model most suited to addressing the problem statement.

Results

Exploratory Data Analysis

- **Launch success** improved gradually over the years
- Most launches were destined for the **Geostationary Transfer Orbit** (GTO)
- Success rate of **Falcon 9** missions to the **Low Earth Orbit** (LEO) improved as it gained more experience
- Missions carrying heaviest payloads were destined for the **Very Low Earth Orbit** (VLEO)
- **Kennedy Space Center Launch Complex** has been SpaceX's most successful launch site

Visualizations

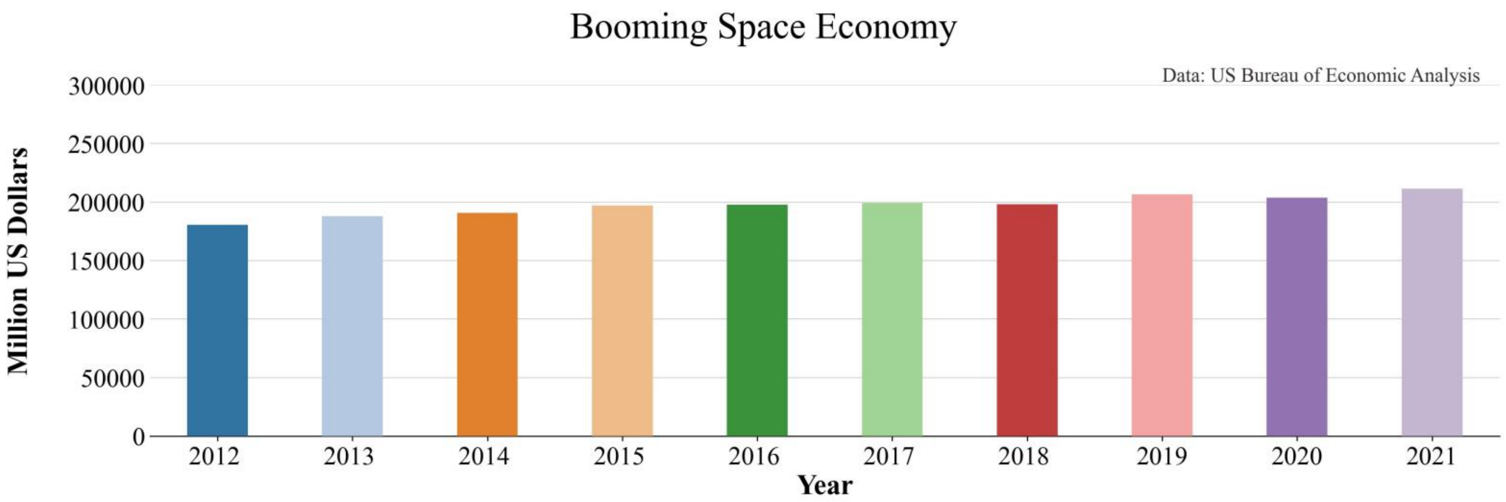
- Launch sites are strategically located close to the **Equator**, **coastline**, and **transport infrastructure**.

Predictive Analytics

- Most models performed similarly when evaluated based on **classification accuracy** score.

Introduction

Space- New Frontier, New Opportunities



- (OECD) **Space economy-** full range of activities and the use of resources that create value and benefits for human beings in the course of exploring, researching, understanding, managing, and utilizing space
  - Goods and services **produced in space for use in space**- includes **rocket launch services**.
- (World Economic Forum) Space economy is projected to be worth **$1.8 trillion** by **2035**.
- Decreasing **launch costs** and ongoing **commercial innovation**- Prime movers of space economy.

Falcon 9- Understanding the competition

- **Reusable**, **two-stage** rocket.
    - Can carry payloads into Earth's orbits
- **Engineering marvel**
    - Rocket’s **first stage** designed to return to Earth and **land vertically**.
    - Can reuse for future missions
- Rocket’s **First stage**- One of the most expensive parts of any rocket
    - **Reuse** results in reduced cost per launch.
    - Key to success enjoyed by SpaceX in this 21st century space race.

Falcon 9

Space Economy- The Race is On …

- SpaceX and its **Falcon 9** rocket- Game changer
  - (SpaceX) Advertised cost per launch- **$62 Million**
  - Cost advertised by other players (average)- **$165 Million**
- Reusing the rocket’s **first stage**
  - Key to SpaceX’s savings and success
- **Problem Statement**- Determining if the first stage will be able to land back successfully (*and be reusable*)
  - Can estimate **cost of a launch**

Falcon 9 v1.2 (Full Thrust)- Block 5
Image: [Wevolver]

Methodology

Summary

- **Data collection methodology**
  - Past launch data was fetched from open source third-party SpaceX API.
  - Past launch data was scraped from a publicly maintained Wikipedia article.
- **Data wrangling**
  - Different SpaceX launch sites were analyzed.
  - Destination orbit for past launches were analyzed.
  - Separate category for second stage landing outcome was created.
  - Encoded into two classes (Success/Failure) for further predictive analysis.

Summary

- Exploratory data analysis (EDA) using **visualization** and **SQL**
  - Visualization techniques including scatter plots, bar charts, and line charts were used to understand inter-relationships among different aspects of past launches.
- Interactive visual analytics using **Folium** and **Plotly Dash**
  - Location of the four SpaceX Launch sites and their proximate elements were analyzed.
  - Dashboard of SpaceX launch records was created to enable stakeholders understand the data in an interactive easy to interpret manner.

Summary

- **Predictive analysis** using **classification** models
  - Separate datasets were generated for **training** and **testing** the classification models.
  - **Four different models** trained, tested and compared to find the most suitable model.
  - Performance of each model was quantified using metrics such as **accuracy score**.
  - **Confusion matrix** for classification models used to analyze performance of models.
  - Best **hyperparameters** were obtained for the models for further finetuning model performance.

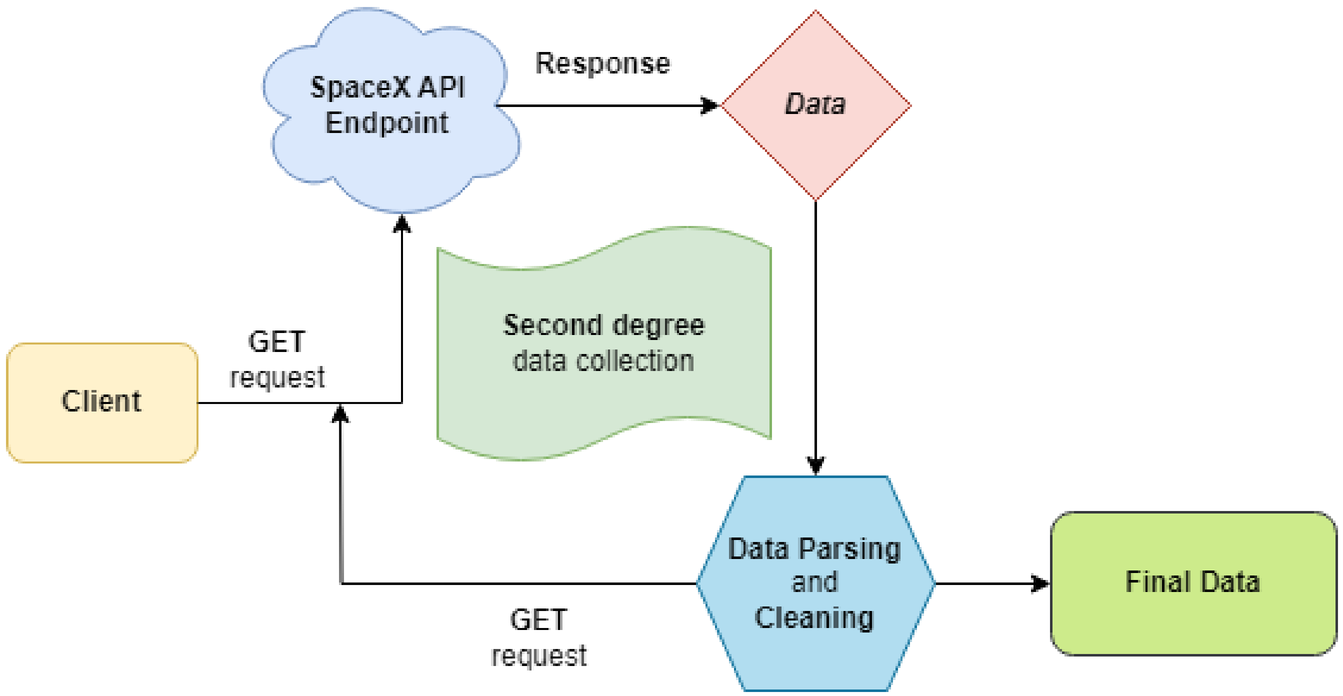
Data Collection

- **SpaceX REST API**
  - Made multiple API calls as needed for fetching relevant details of past launches
- Fetching relevant data step-by-step:
  - Getting the **past launch records**
  - Extracting **second-degree information** from the records returned by the SpaceX API
    - From \`cores\` the **landing outcome**, **landing pad usage**, **number of flights** with that core etc. were extracted
    - From the \`rocket ID\` its **name** was extracted by making further API calls at a different end point
    - From the \`launchpad ID\` its **name** and **location coordinates**
    - From \`payload ID\` its **mass** (kg) and destination **space orbit**

Data Collection- SpaceX API

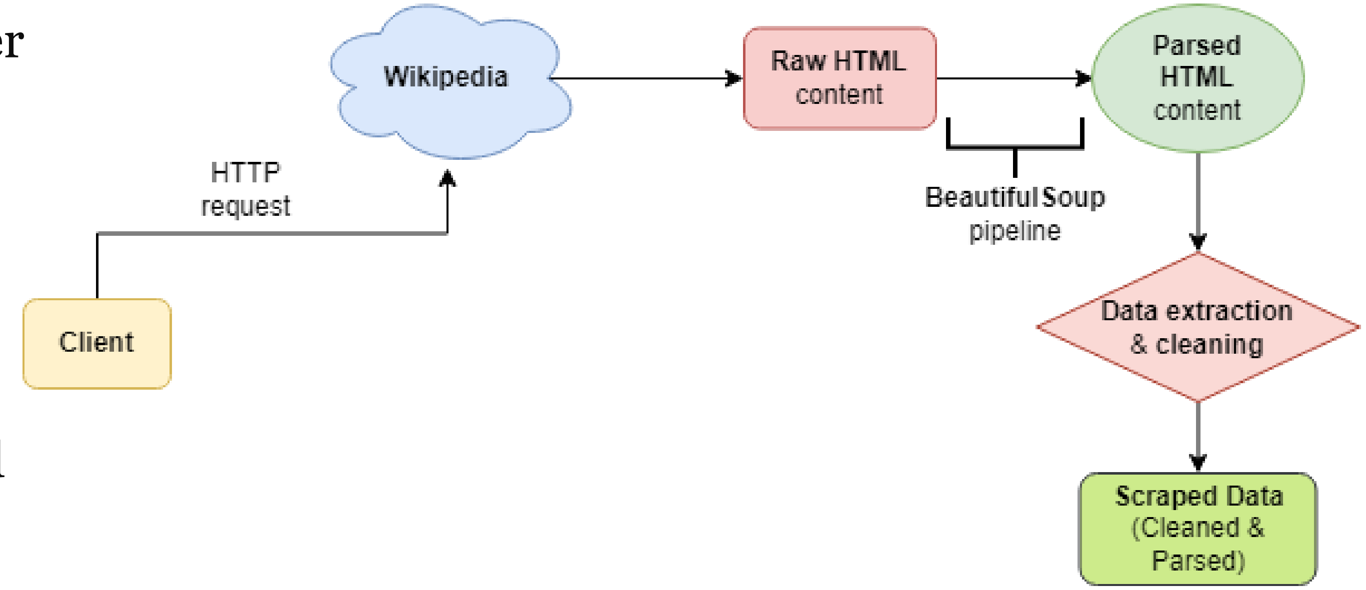
- **GET request** to multiple **SpaceX API endpoints** were made to fetch following details of past launches, the rocket, launch site and payload
- Separate **functions** were coded in Python for different API endpoint calls to make data collection method **modular** and streamlined

- GitHub URL of Jupyter Notebook:
    - https://rb.gy/5pex0o



Data Collection- Web Scraping

- **HTML** content of **Wikipedia** article List of *Falcon 9 and Falcon Heavy launches* was retrieved using Python library **requests**
- **HTML structure** of probed to uncover fields containing data of interest
- Relevant data (in tabular format) was inside `**<table>**` tags with class starting with 'wikitable'
- Data was scraped using open-source library **BeautifulSoup**
- **Regular expressions** were employed as needed
- Separate functions were coded to parse the scraped data
- [Github URL](https://rb.gy/3z9inb) of Jupyter Notebook:
  - https://rb.gy/3z9inb

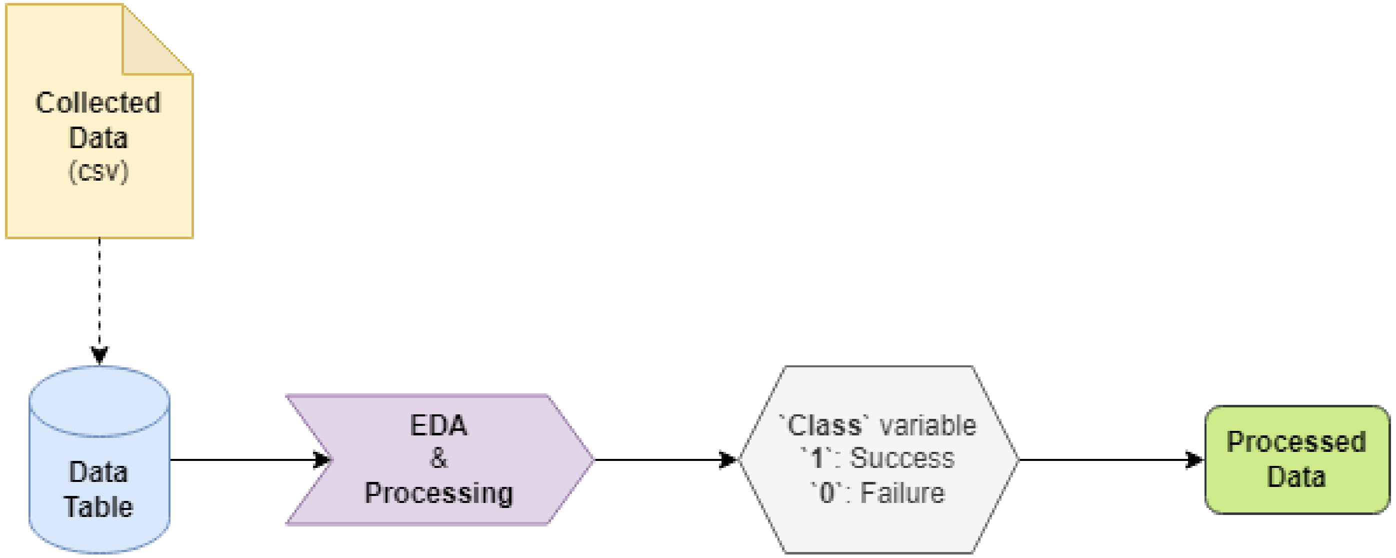


Data Wrangling

- Data collected in previous phase was loaded in a **tabular format** for further processing.
- Each **column** of table, corresponding to some **data variable**, was inspected to answer following questions:
  - Is data **missing** in any column?
  - What **type** of data is being stored in each column?
  - EDA revealed multiple possible **landing outcomes** were possible such as:
  - `**True Ocean**`: Successful landing over ocean,
`**True RTLS**`: Successful return to launch site i.e. ground pad,
`**None**`: Failure to land

Data Wrangling

- **Landing outcomes** data was processed further to make it suitable for **predictive analysis**
- Separate data variable `**Class**` was created
- `Class` was **binary one-hot encoded** by combining similar outcomes
  - `1`: Successful landing
  - `0`: Failed to land
- GitHub URL of Jupyter Notebook:
  - https://rb.gy/3nrti6



EDA with Data Visualization

- Visualizing data to uncover trends and relationships among the data variables
- **Scatter plot**
  - Visualizing **relationship** among two data variables and how landing outcome is impacted by them
  - Example: `FlightNumber` (indicates continuous launch attempts) and `PayloadMass` (payload mass in kg)
- **Bar chart**
  - Comparing **categorical data** visually
  - Example: Launch success rate for different target destination orbits
- **Line chart**
  - Visualizing **trends**
  - Example: Launch success rate over the years
- GitHub URL of Jupyter Notebook:
  - https://rb.gy/m5hcuz

EDA with SQL

- Displaying names of **unique launch sites**
- Displaying 5 records where launch **sites begin with 'CCA'**
- Displaying total payload mass carried by **boosters launched by NASA** (CRS)
- Displaying average payload mass carried by **booster version F9 v1.1**
- Listing the date of **first successful landing in ground pad** (Return to launch site)
- Listing names of boosters which have attained **successful landing in drone ship** and have payload mass greater than 4000 kg but less than 6000 kg
- Displaying total number of successful and failure mission outcomes
- Listing names of booster versions which carried **maximum payload mass**

EDA with SQL

- Listing multiple records for **launches in the year 2015**
  - Month
  - Failure landing outcomes in drone ship
  - Booster version
  - Launch site
- Ranking in **descending** order the total count of landing outcomes for launches between dates 2010-06-04 and 2017-03-20
- GitHub URL of Jupyter Notebook:
  - https://rb.gy/xkjh20

Interactive Map with Folium

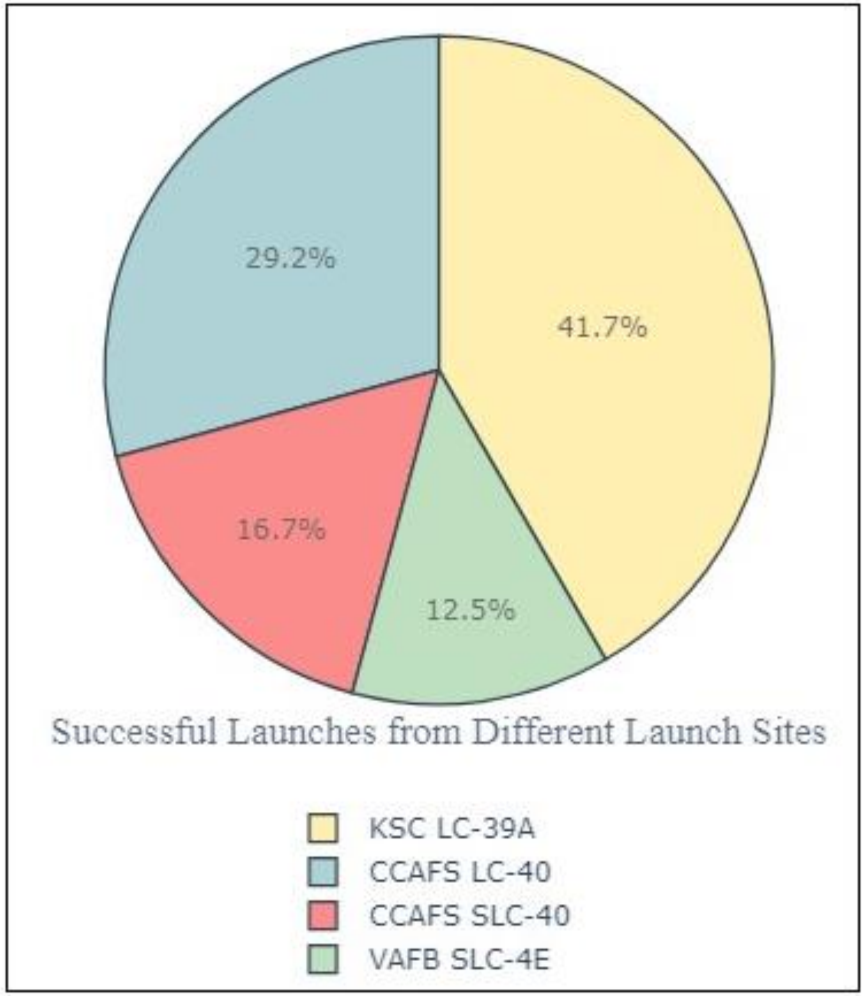
- Open-source Python library **`Folium`** was used to build interactive maps for the **launch sites**
- Following **map objects** were created for this purpose:
- **`folium Map`**
  - Creating a base map of United States of America centered over NASA Johnson Space center.
  - All further interactive features were created over this Map object.
- **`folium Circle`**
  - Creating circular coloured markers on the map indicating location of launch sites.
- **`folium Popup`**
  - Adding interactivity to the map.
  - Popup with detailed information of the selected launch site.

Interactive Map with Folium

- **`folium Marker`**
  - Adding indicative **icons** over launch sites.
  - Example: Name of the launch site highlighted by circle marker.
- **`folium PolyLine`**
  - Adding details to convey helpful additional information.
  - Example: Line of Equator, Distance of the launch sites to the coastline, highway, nearest town etc.
- **`MarkerCluster`**
  - Adding interactivity and details to each marked launch site.
  - Example: Cluster of launches made from each launch site with distinct color and icon signifying whether mission was successful or failure.
- GitHub URL of Jupyter Notebook:
  - https://rb.gy/ay1493

Interactive Dashboard with Plotly Dash

- Open-source Python libraries `**Dash**` and `**Plotly**` were used to create interactive dashboard for visual inspection of **past SpaceX launch records**
- Multiple plots to understand data visually
- **Pie chart**
    - To visualize share of launch **success rate** across **all launch sites**
    - To get an idea of success rate of each **selected launch site**



Interactive Dashboard with Plotly Dash

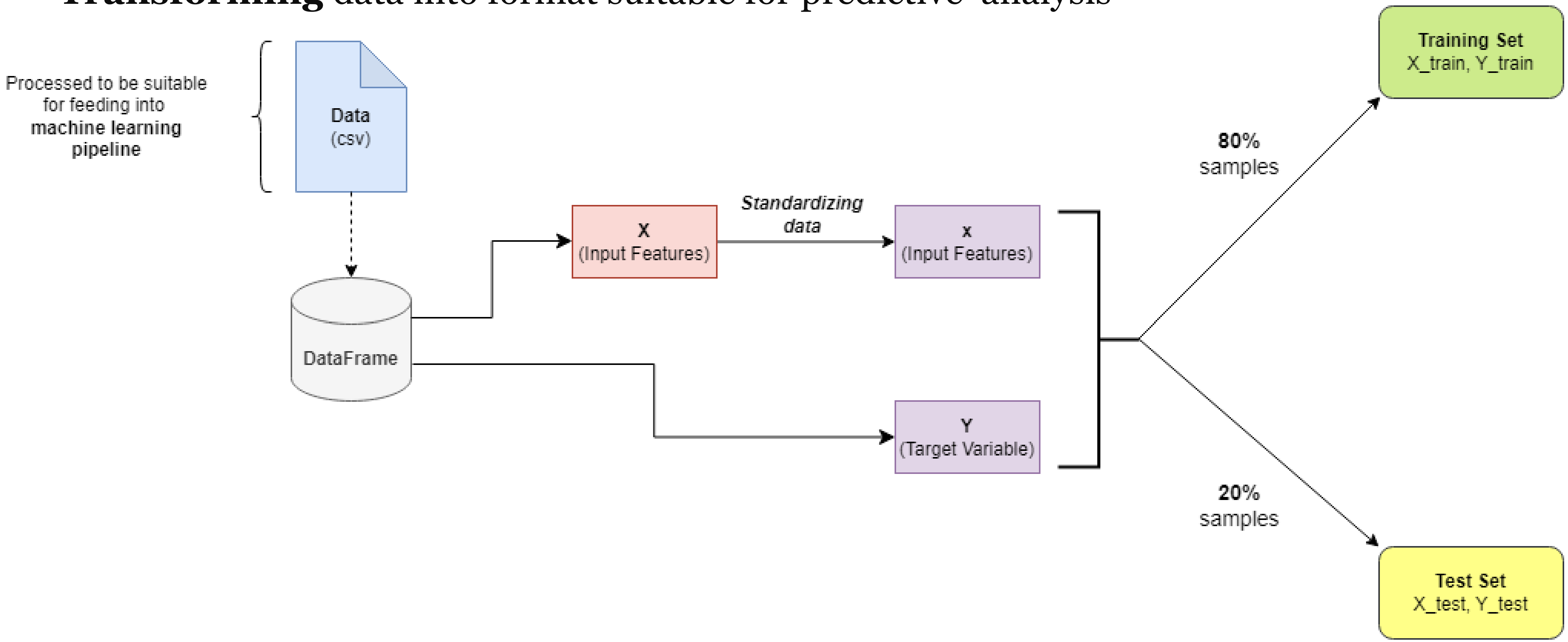
- **Scatter plot**
  - To visualize **correlation** between Payload mass (kg) and mission outcome.
  - Further detail was added to indicate the **booster version** and **mission outcome** for each launch in selected payload mass range.
- GitHub URL of Python code:
  - https://shorturl.at/fwXlt

Predictive Analysis

- Four **Classification** models were built and evaluated to find the most suitable model for the current task.
  - Support Vector Machine (SVM)
  - Classification Trees
  - Logistic Regression
  - K-Nearest Neighbours Classifier
- **Model development-** A 3-stage process
  - Model **creation** and **training**
  - Model **testing**
  - Model **evaluation** and **finetuning**
- Open-source machine learning library `**scikit learn**` was used for model development

Predictive Analysis

- Model development process began with **data preparation**
  - Creating separate datasets for **training** and **testing**
  - **Transforming** data into format suitable for predictive analysis

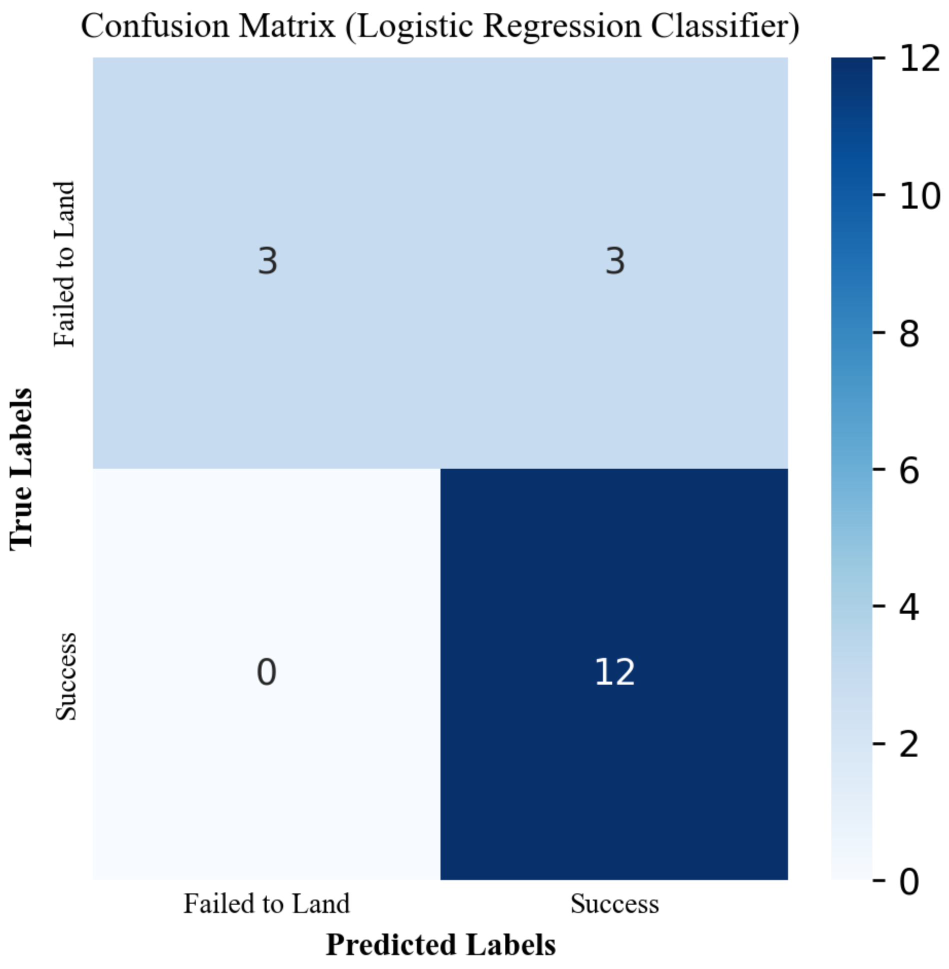


Phase I- Model Creation and Training

- **Model object** was instantiated
- **Hyperparameters** to be tried out were defined
- Model was trained on **Training set**
- \`**GridSearchCV**\` of \`scikit learn\` was used to evaluate best combination of hyperparameters and their values

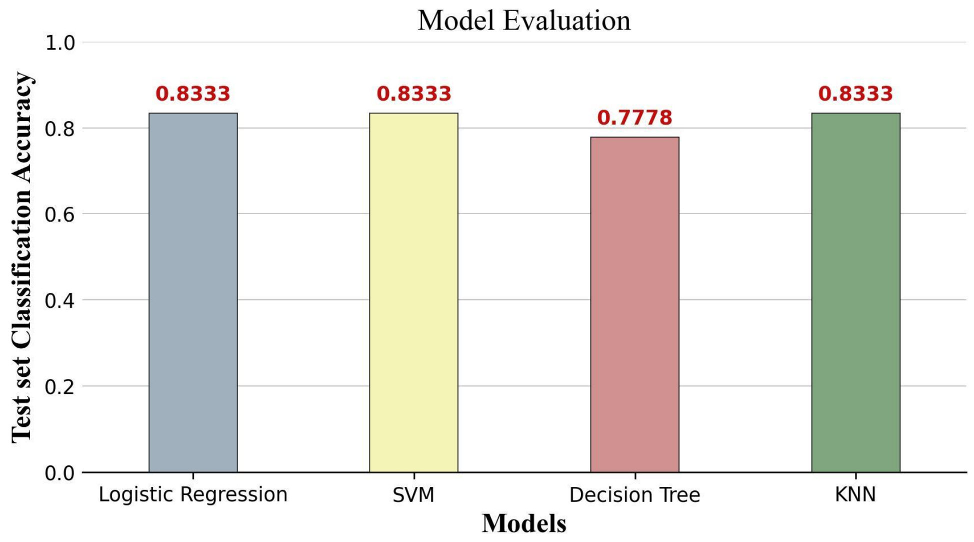
Phase II- Model Evaluation and Improvement

- Computing **Accuracy score** on **Test set** of the trained classification model
- **Confusion matrix** was plotted to visually understand model performance and get deeper insights
  - Precision
  - Recall
  - False positives and False negatives

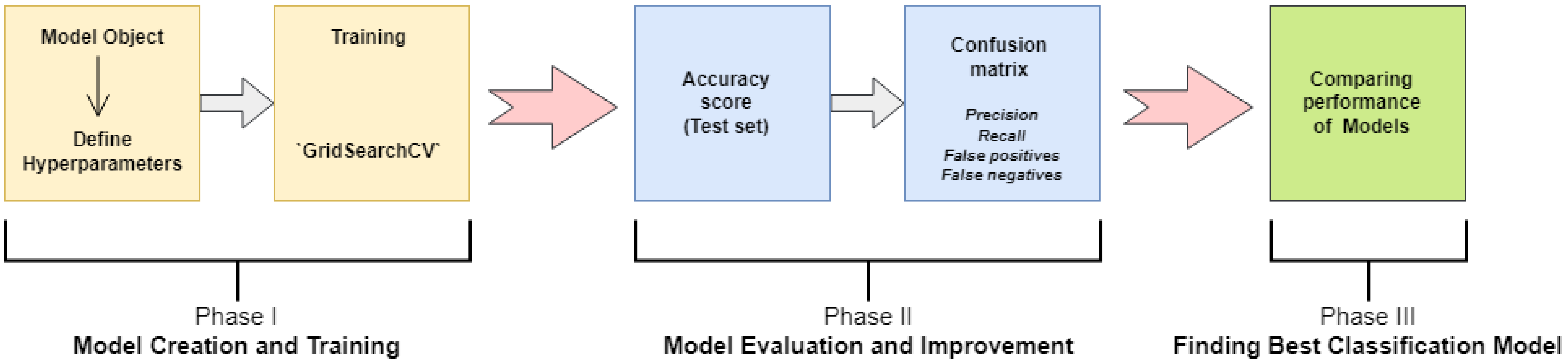


Phase III- Comparing the Models

- **Test set Accuracy** score of all four trained models were presented in **tabular** format
- **Bar chart** was plotted to visually compare the four models and find the best model



Predictive Analysis Pipeline

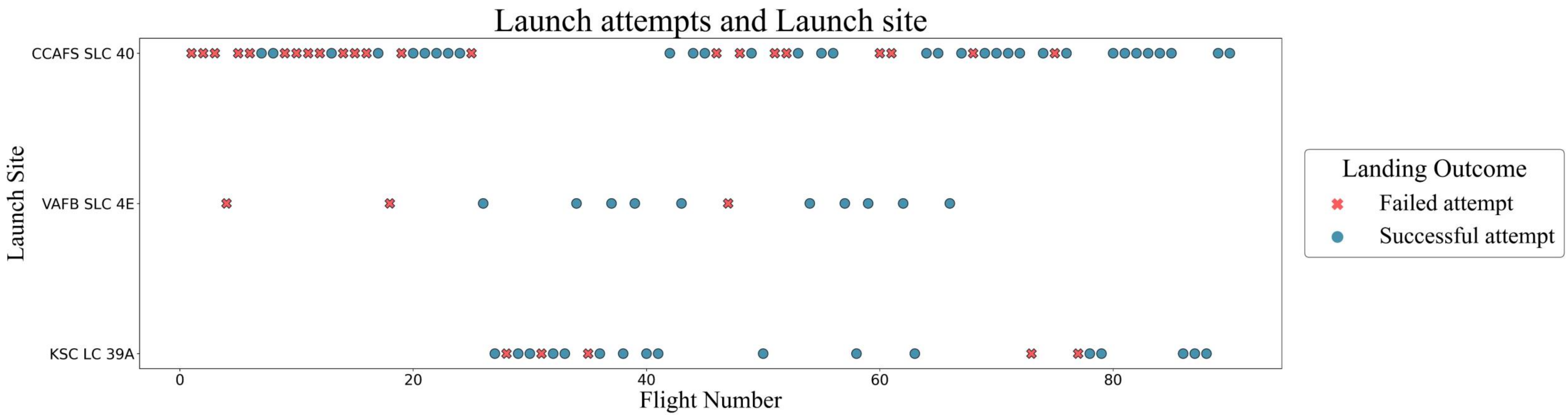


Results

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

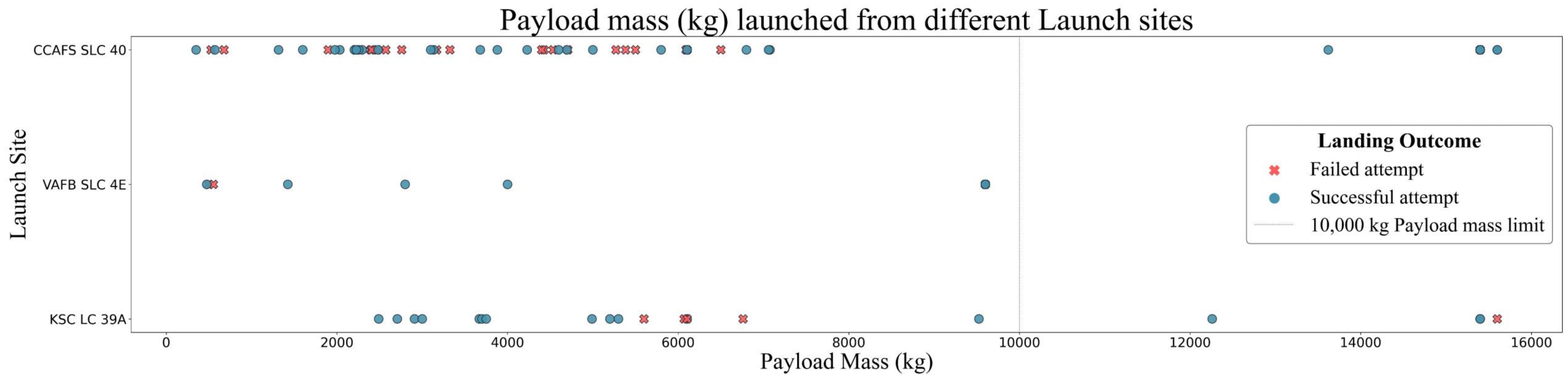
Insights drawn from EDA

Flight Number vs. Launch Site



- Launches from **CCAFS SLC 40** have show **higher failure rate** compared to the other sites
- Rockets launched from **KSC LC 39A** and **VAFB SLC 4E** seem to have much **higher success rate**

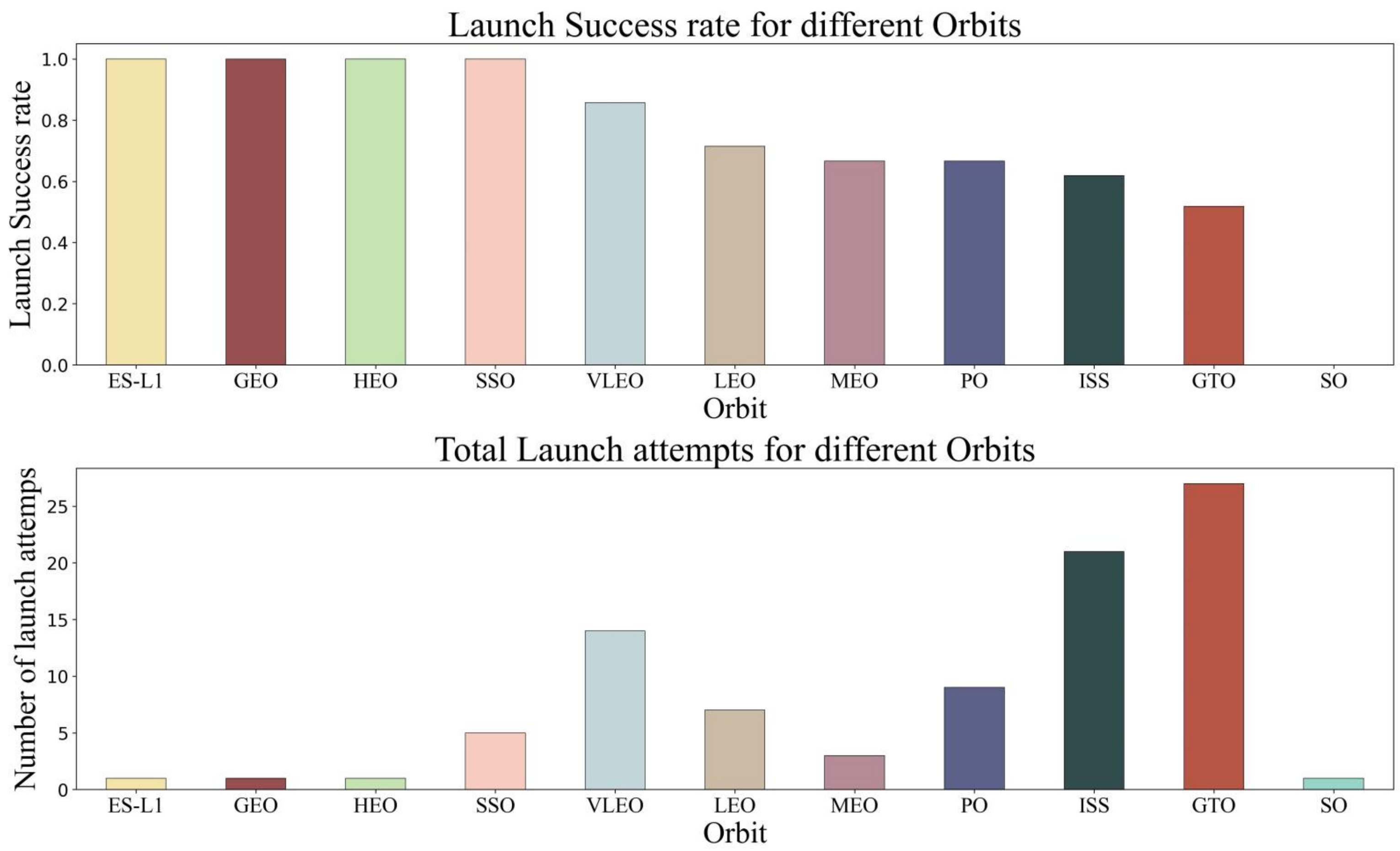
Payload vs. Launch Site



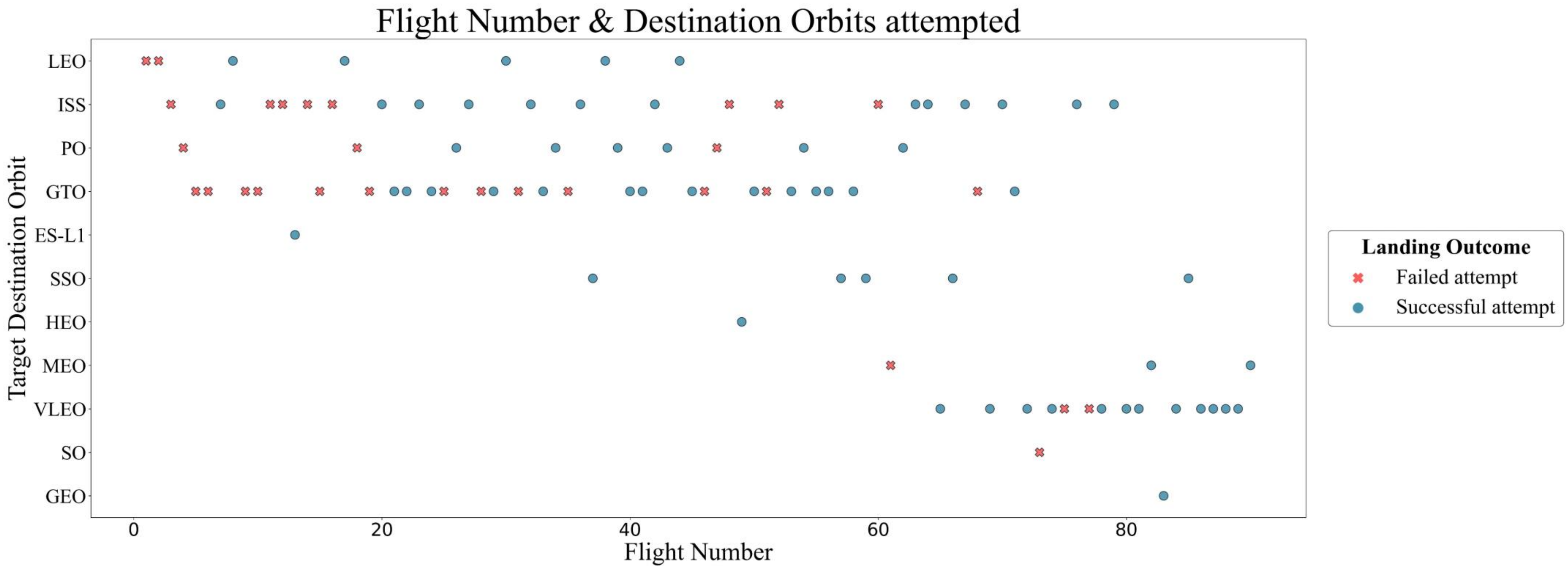
- **VAFB SLC 4E** does not have any launches with **payload mass >10,000** ***kg***
- Launches from **KSC LC 39A** and **CCAFS SLC** are mostly for **payloads <8,000** ***kg***
- **KSC LC 39A** site has a much better record in terms of landing success

Success Rate vs. Orbit Type

- **ES-L1**, **GEO**, **HEO**, **SSO** have a **perfect success rate** of 100%
  - Lowest number of **launch attempts**
- Most launches had target destination of **GTO**
  - Low **success rate** of barely ≈50%
- Several launches were destined for **VLEO**
  - It has shown a **relatively high success rate**
- Only one of the launch attempts had **SO** as its target
  - Landing outcome was a **failure**

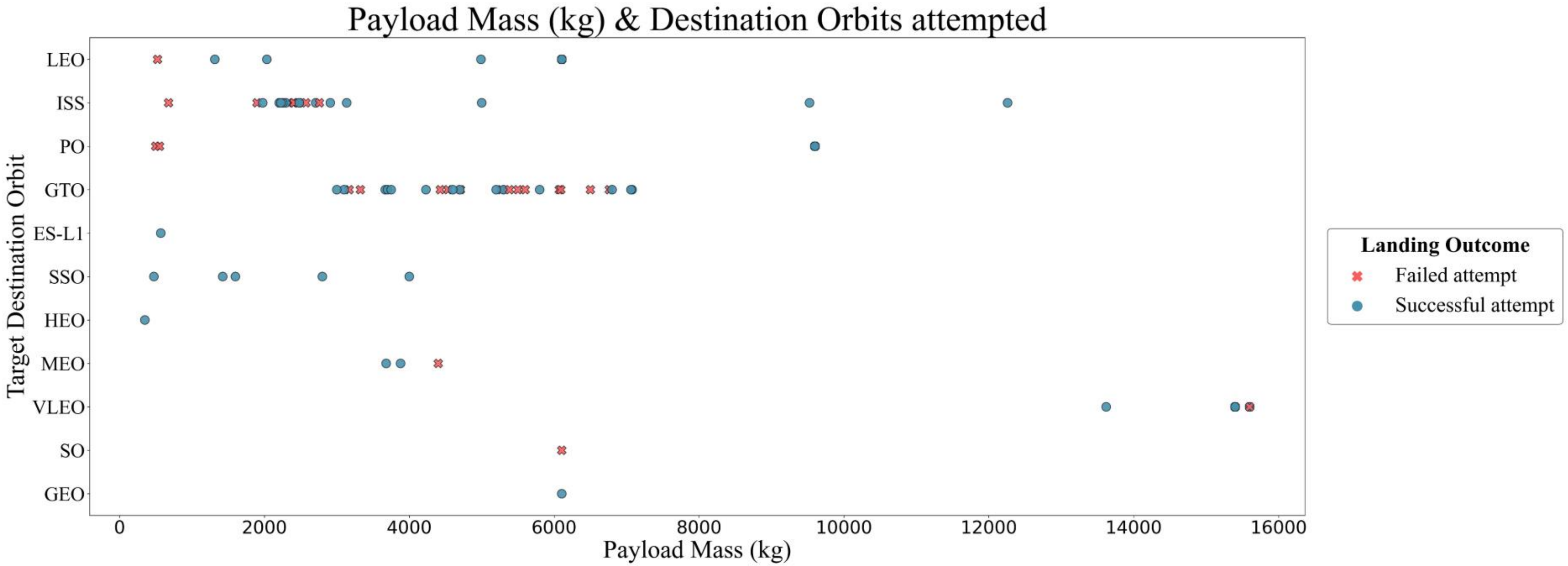


Flight Number vs. Orbit Type



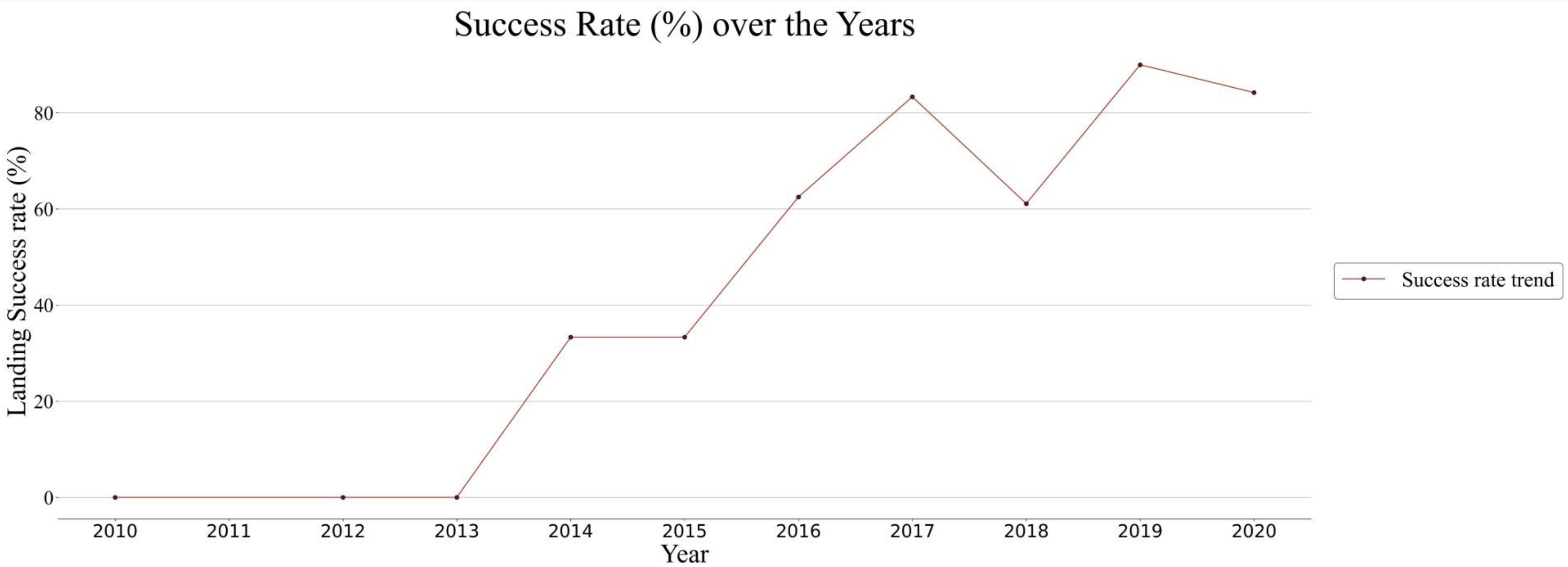
- **LEO-** As **number of flights** (*experience of Falcon 9*) increases **success rate** of landing attempts also seem to increase
- **Other orbits-** Correlation between number of flights and success seems to be missing.
- **GTO-** Success and flight number seem to have **no correlation** at all

Payload vs. Orbit Type



- **Heaviest payloads** have been attempted for **VLEO**
  - One of the nearest satellite orbits of Earth ($\leq 400km$)
  - Heavier the payload, greater **volume of fuel** is needed to fly per unit distance
- Relatively heavy payloads show higher success rate for **Polar**, **LEO**, and **ISS** orbits
- Most launches have been planned for **GTO**
  - Payload mass have always been **$<8{,}000kg$**
  - Success rate has been on the lower side ($\approx 50\%$)

Payload vs. Orbit Type



- SpaceX saw **no success** in landing back the rocket until year **2013**
  - **Patience** is key when competing in the space race
- Year **2013-2017**: Success rate grew monotonically
- **General trend** of success rate has been **upward**
  - Year **2017** and **2020-** Success rate fluctuated
  - **2017** & **2018**, and **2019** & **2020-** Success rate dipped

All Launch Sites

- `**DISTINCT**` SQL function was used to extract names of unique SpaceX launch sites
- `**SPACEXTABLE**`- Table storing the relevant data

```
SELECT DISTINCT `Launch_Site` AS
`Unique Launch sites`
FROM SPACEXTABLE;
```

Unique Launch sites
CCAFS LC-40
VAFB SLC-4E
KSC LC-39A
CCAFS SLC-40

Launch Sites beginning with ‘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

- \`**LIKE**\` SQL function enables searching for patterns
- ‘**CCA**’ pattern in Launch site names was searched, matched, and extracted from \`SPACEXTABLE\` table

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

Total Payload Mass

- **`SUM`**, **`LIKE`** and **`GROUP BY`** SQL functions were used to retrieve the total payload mass for NASA's missions
- **`GROUP BY`**: For grouping the data among **unique customers**
- **`LIKE`**: For retrieving NASA (customer) missions
- **`SUM`**: For **totaling** the payload masses

```
SELECT
Customer,
SUM(PAYLOAD_MASS__KG_) AS `Total Payload mass (kg)`
FROM SPACEXTABLE
WHERE Customer LIKE '%CRS%'
GROUP BY `Customer`;
```

Customer	Total Payload mass (kg)
NASA (CRS)	45596
NASA (CRS), Kacific 1	2617

Average Payload Mass by F9 v1.1

- `**REGEXP**` function was employed to use **regular expression** pattern matching to filter the data
- Past records whose Booster Version **starts** ('^' in the pattern) with '**F9 v1.1**'
- Result shows average payload mass for all records whose booster version starts with 'F9 v1.1'

```
SELECT
Booster_Version AS `Booster Version`,
AVG(PAYLOAD_MASS__KG_) AS `Average Payload mass (kg)`
FROM SPACEXTABLE
WHERE `Booster_Version` REGEXP '^F9 v1.1'
GROUP BY `Booster_Version`;
```

Booster Version	Average Payload mass (kg)
F9 v1.1	2928.4
F9 v1.1 B1003	500.0
F9 v1.1 B1010	2216.0
F9 v1.1 B1011	4428.0
F9 v1.1 B1012	2395.0
F9 v1.1 B1013	570.0
F9 v1.1 B1014	4159.0
F9 v1.1 B1015	1898.0
F9 v1.1 B1016	4707.0
F9 v1.1 B1017	553.0
F9 v1.1 B1018	1952.0

First Successful Ground Landing

- **`REGEXP`** function was employed to use **regular expression** pattern matching to filter the data
- Past records where landing outcome **starts** ('**^**' in the pattern) with 'Success'
- **`MIN`** function used on date of all data which matched the pattern allows fetching the **earliest** (first) **date**

First Successful Landing	Landing_Outcome
2015-12-22	Success (ground pad)

```
SELECT
MIN(`DATE`) AS `First Successful Landing `
`Landing_Outcome`
FROM SPACEXTABLE
WHERE `Landing_Outcome` REGEXP '^Success';
```

Landing with Payload between 4000-6000 kg

- **`REGEXP`** function was employed to use **regular expression** pattern matching to filter the data
- Past records where landing outcome **starts** ('^' in the pattern) with 'Success' and includes '**drone**' **sub-string**
- **`BETWEEN`**- Restricts past records to only those with payload mass (kg) **within 4000** and **6000**

Booster	Payload Mass (kg)	Landing Outcome
F9 FT B1022	4696	Success (drone ship)
F9 FT B1026	4600	Success (drone ship)
F9 FT B1021.2	5300	Success (drone ship)
F9 FT B1031.2	5200	Success (drone ship)

```
SELECT
Booster_Version AS `Booster`,
PAYLOAD_MASS__KG_ AS `Payload Mass (kg)`,
Landing_Outcome AS `Landing Outcome`
FROM SPACEXTABLE
WHERE
Landing_Outcome REGEXP '^Success\s*\S*drone\s*\S*'
AND
PAYLOAD_MASS__KG_ BETWEEN 4000 AND 6000;
```

Tracking Success and Failure

- **`COUNT`** and **`GROUP BY`** SQL functions were used to retrieve the total number of missions in each distinct mission outcome
- **`GROUP BY`**- Groups the past records based on **unique mission outcome categories**
  - Failure (in flight)
  - Success
  - Success (payload status unclear)
- **`COUNT`**- Totals the number of records in **each category**

Mission Outcome	Total
Failure (in flight)	1
Success	98
Success	1
Success (payload status unclear)	1

```sql
SELECT
Mission_Outcome AS `Mission Outcome`,
COUNT(Mission_Outcome) AS `Total`
FROM SPACEXTABLE
GROUP BY Mission_Outcome;
```

Big Boosters- Which carried Maximum Payload?

- SQL **Subquery** was used to retrieve the maximum payload mass carried by each type of booster
  - **Subquery-** Nested SQL query written and executed inside another **outer query**
- **`WHERE`** (outer query) function's condition (inner subquery) filters out the maximum payload mass (kg) under each unique booster

```
SELECT
Booster_Version AS `Booster`,
PAYLOAD_MASS__KG_ AS `Maximum Payload Mass (kg)`
FROM SPACEXTABLE
WHERE
PAYLOAD_MASS__KG_ = (
                    SELECT MAX(PAYLOAD_MASS__KG_)
                    FROM SPACEXTABLE
                    );
```

Booster	Maximum Payload Mass (kg)
F9 B5 B1048.4	15600
F9 B5 B1049.4	15600
F9 B5 B1051.3	15600
F9 B5 B1056.4	15600
F9 B5 B1048.5	15600
F9 B5 B1051.4	15600
F9 B5 B1049.5	15600
F9 B5 B1060.2	15600
F9 B5 B1058.3	15600
F9 B5 B1051.6	15600
F9 B5 B1060.3	15600
F9 B5 B1049.7	15600

Launches in the year 2015

- Python `**sqlite3**` library does not support month names
    - `**SUBSTR**` function was used to overcome this limitation
- `**SUBSTR**` extracts the month and year from the date of launch and `**WHERE**` restricts the retrieved data to only launches of year 2015

Month	Year	Landing Outcome	Booster	Launch Site
01	2015	Failure (drone ship)	F9 v1.1 B1012	CCAFS LC-40
04	2015	Failure (drone ship)	F9 v1.1 B1015	CCAFS LC-40

```sql
SELECT
SUBSTR(Date, 6, 2) AS `Month`,
SUBSTR(Date, 1, 4) AS `Year`,
Landing_Outcome AS `Landing Outcome`,
Booster_Version AS `Booster`,
Launch_Site AS `Launch Site`
FROM SPACEXTABLE
WHERE
Landing_Outcome REGEXP '\s*\S*[f|F]ailure\s*\S*[d|D]rone\s*\S*'
AND
SUBSTR(Date, 1, 4) = '2015';
```

Ranking Outcomes- 2010-06-04 to 2017-03-20

- Multiple SQL functions including **`COUNT`**, **`RANK( )`**, and **`CAST`** were used to achieve following:
  - **Filter** and extract records of selected date ranges and **count** the total of the landing outcomes
  - **`GROUP BY`**- Groups the records by the **unique landing outcomes**
  - **`RANK() OVER (ORDER BY COUNT**(Landing Outcome) **DESC**) `- Assigns a rank to **each unique landing outcome** based on the **descending** ordered total count of records in each outcome category
  - **`ORDER BY`**- Rearranges the records in descending order based on **total count** of landing outcomes

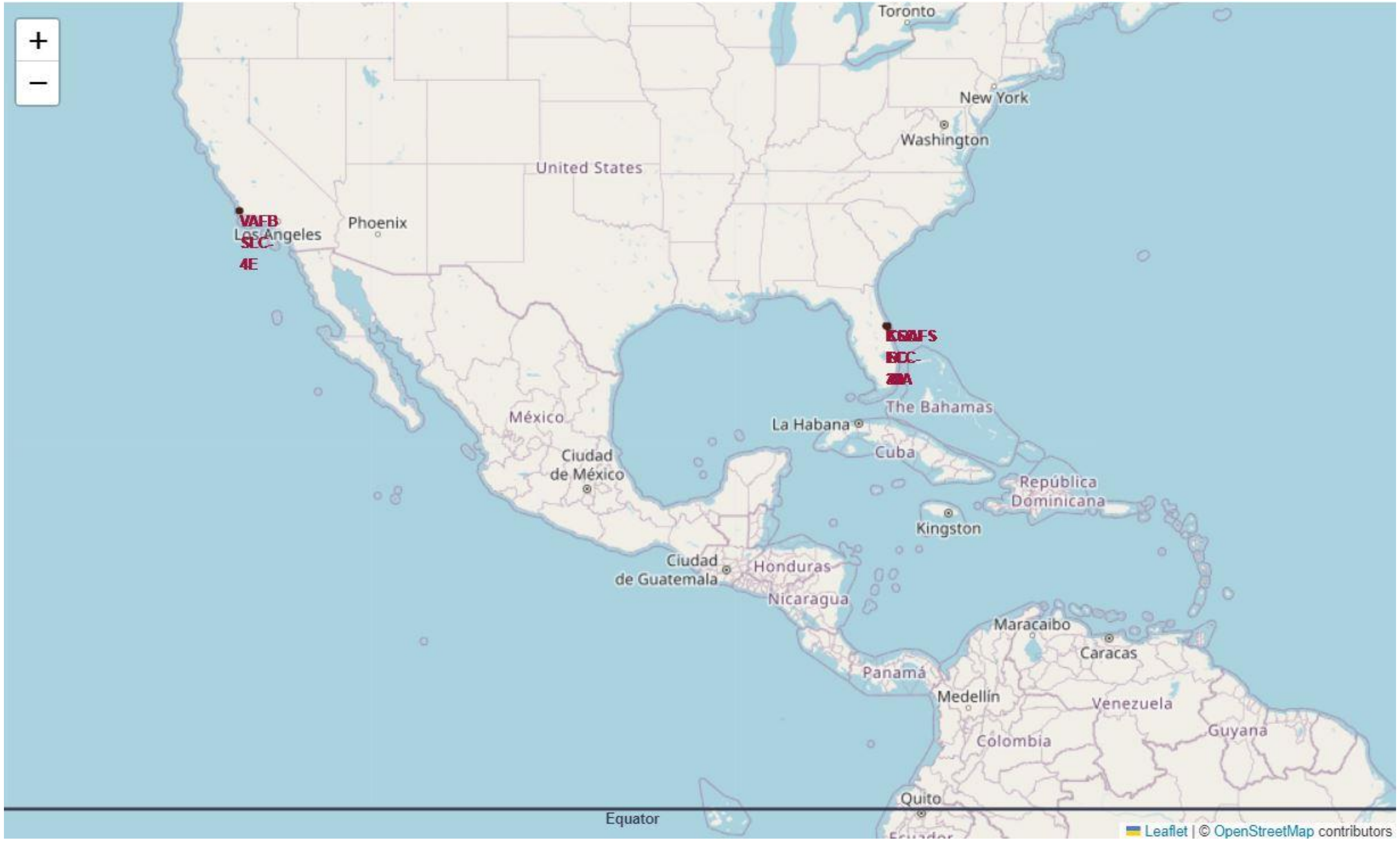
Date	Landing Outcome	Total Outcome	Rank
2012-10-08	No attempt	6	1
2016-07-18	Success (ground pad)	5	2
2016-08-14	Success (drone ship)	4	3
2015-01-10	Failure (drone ship)	4	3
2010-06-04	Failure (parachute)	2	5
2014-07-14	Controlled (ocean)	2	5

Ranking Outcomes- 2010-06-04 to 2017-03-20

```
SELECT

Date AS `Date`,
Landing_Outcome AS `Landing Outcome`,
COUNT(Landing_Outcome) AS `Total Outcome`,
RANK() OVER (ORDER BY COUNT(Landing_Outcome) DESC) AS `Rank`

FROM SPACEXTABLE

WHERE
CAST(SUBSTR(Date, 1, 4) AS INT) BETWEEN 2010 AND 2017
AND
CAST(SUBSTR(Date, 6, 2) AS INT) IN (6, 7, 8, 9, 10, 11, 12, 1, 2, 3)
AND
CAST(SUBSTR(Date, 9, 2) AS INT) BETWEEN 4 AND 20

GROUP BY Landing_Outcome

ORDER BY `Total Outcome` DESC;
```

Launch sites Proximities Analysis

SpaceX Launch Sites



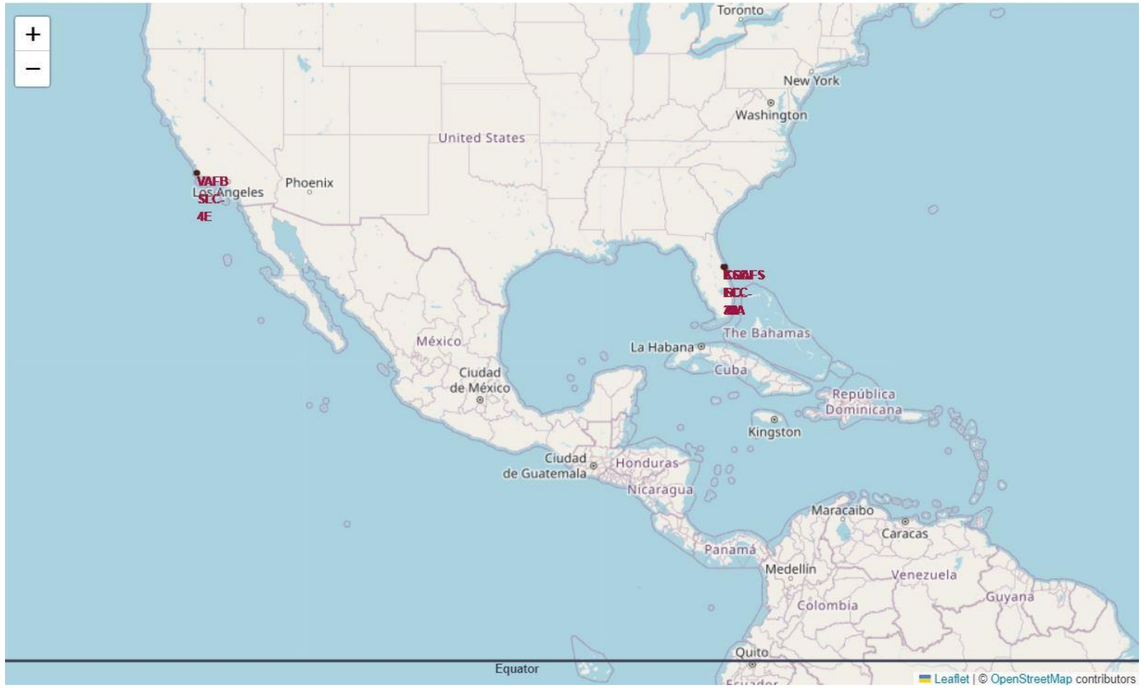
SpaceX Launch Sites

Proximity to Equator line

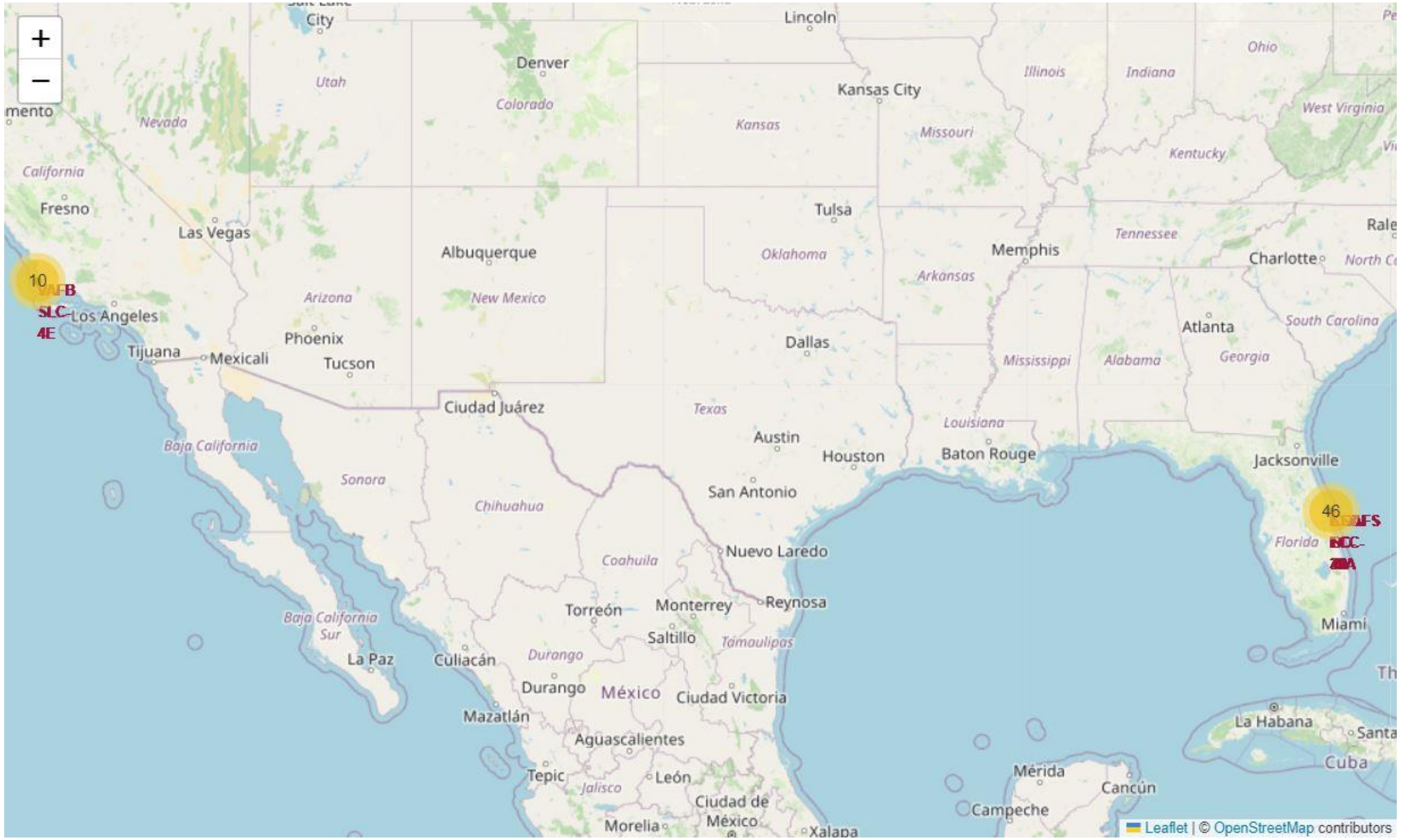
- Launch sites were strategically chosen to be as close to line of **equator** as possible (while remaining in mainland USA).
    - Rocket gets an extra boost from **Earth's rotational motion**

Proximity to the coast

- Sites are located along or near the coastline
- Strategic significance- **Safety** of human population
    - **Failed launches-** Rocket crashes over the ocean instead of mainland
    - (Contrary to what is preferred) **Vandenberg AFB Space Launch Complex 4** is on the west coast

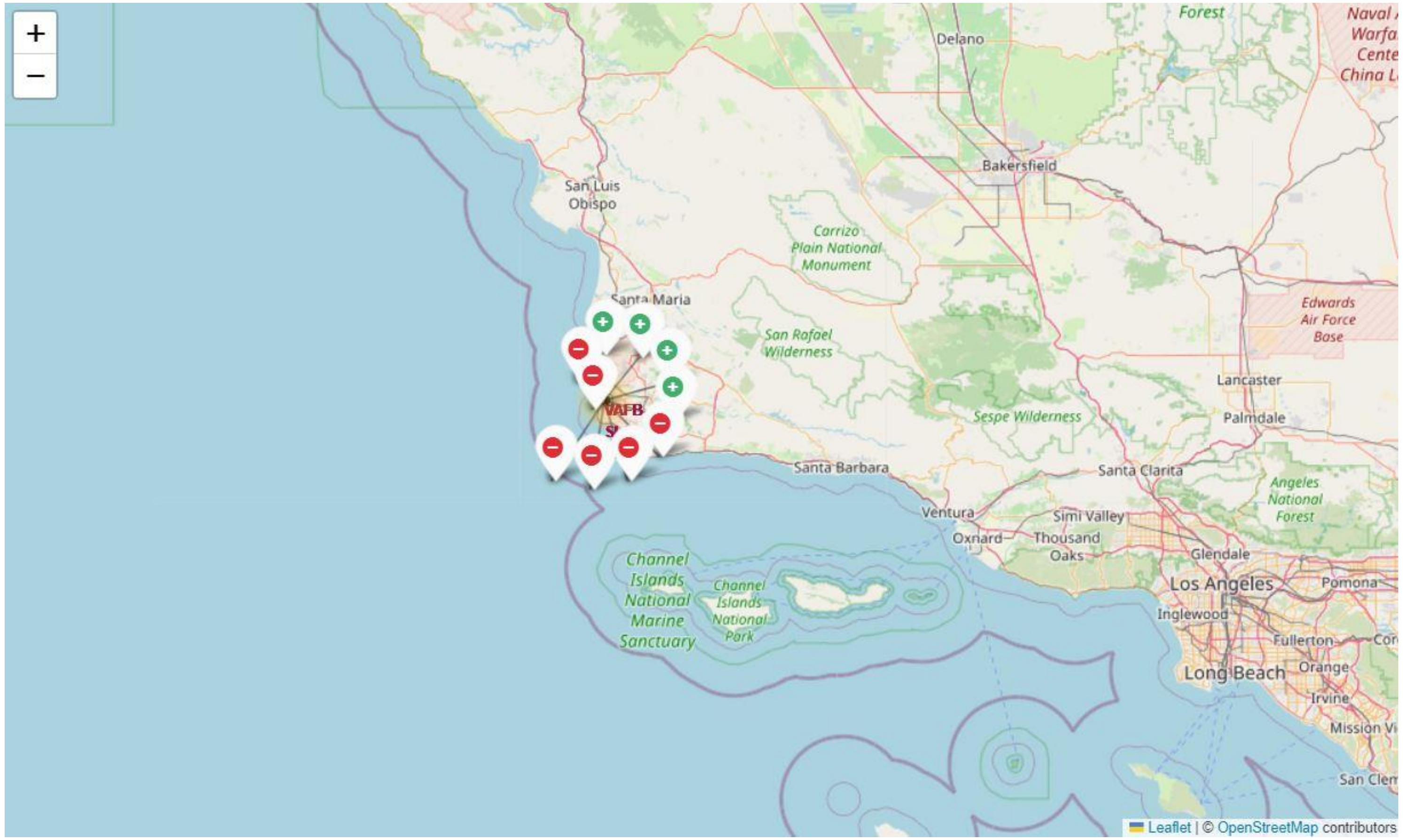


Location and Outcomes



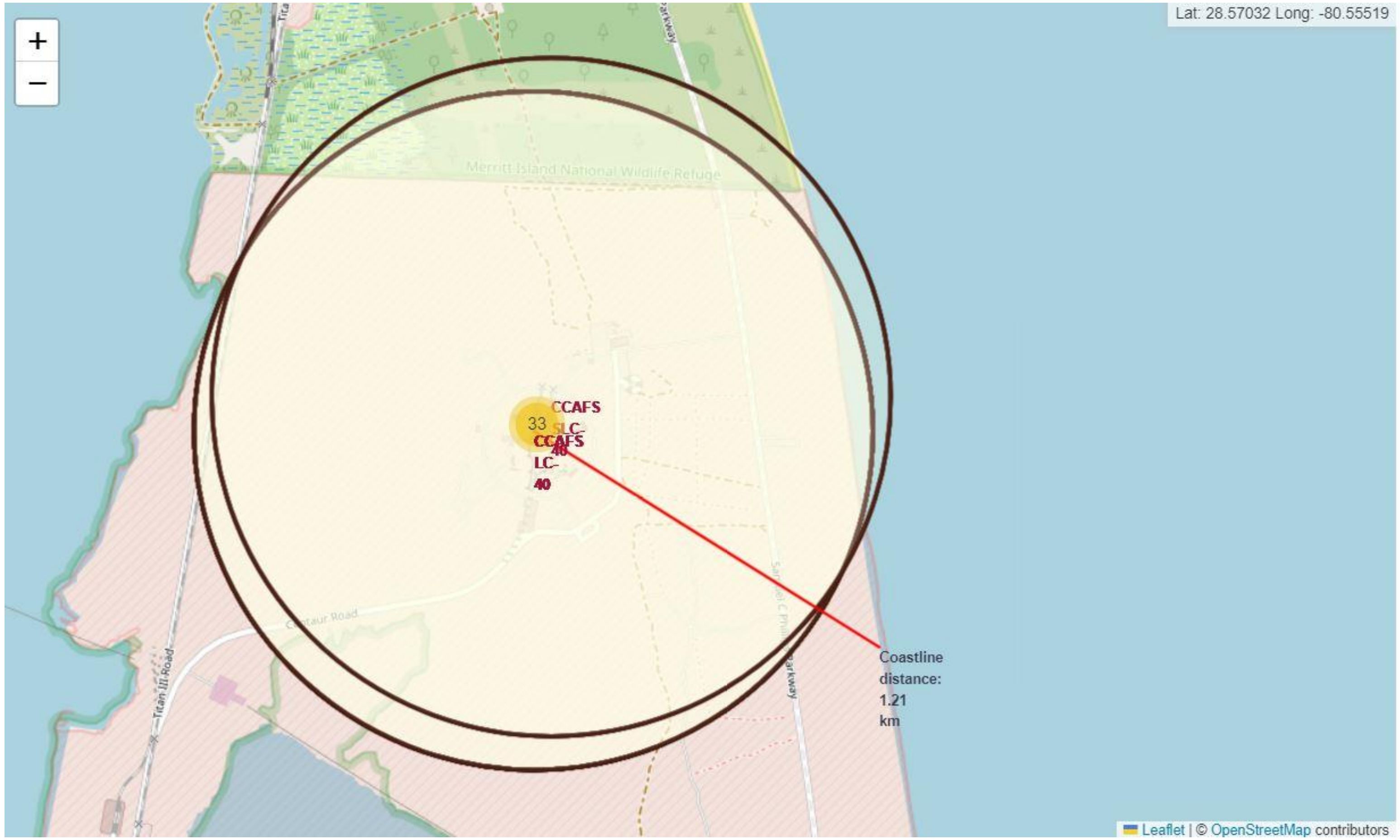
Location and Outcomes

- **Colour-coded** label markers were used in **marker clusters**
- Visually analyze **success rate** of each launch site



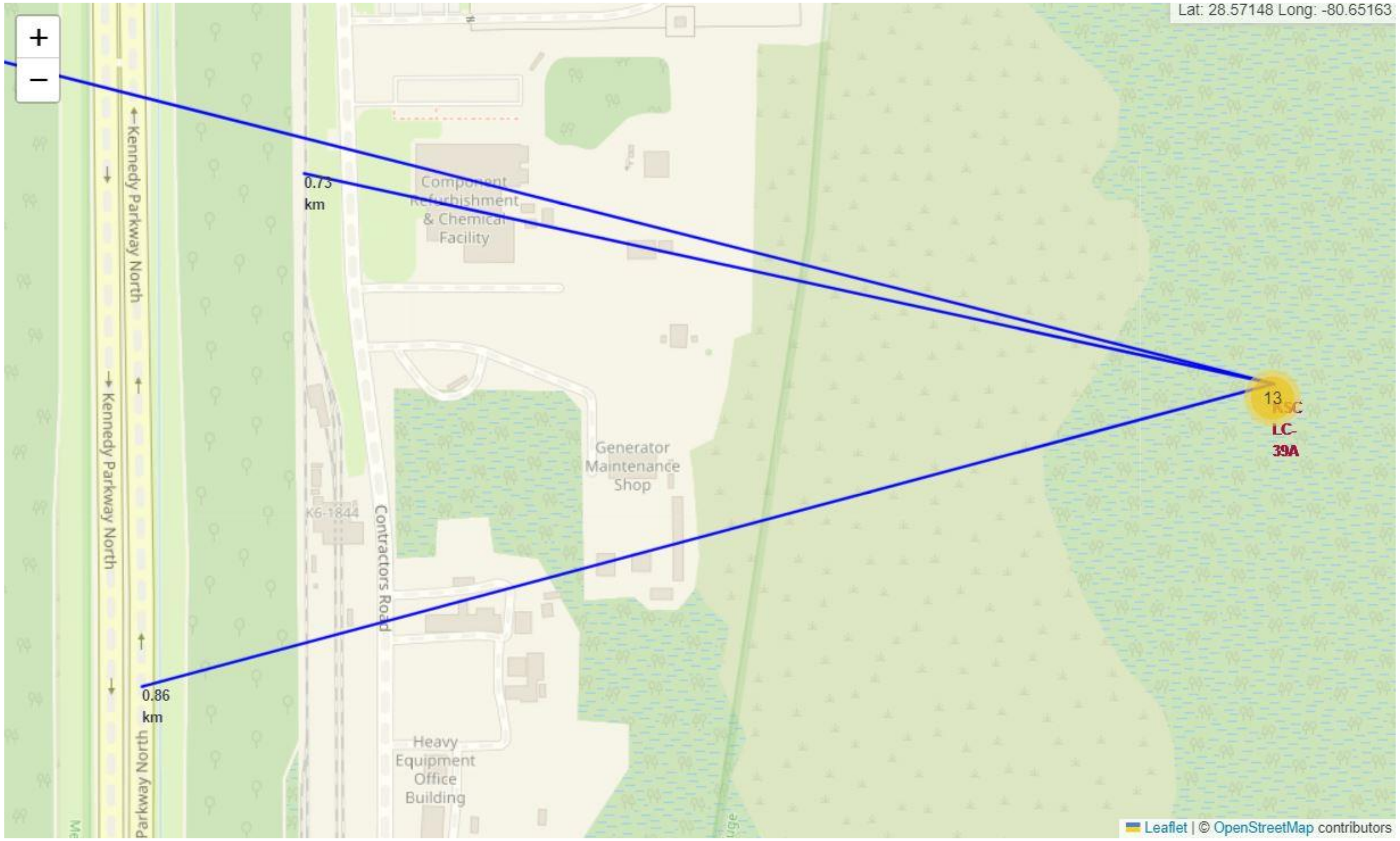
Launch Site and Proximities

- Launch site location is in close proximity to **coastline**
- **CCAFS SLC-40** center is located at a distance of ≈**1.21** *km* only from the coastline



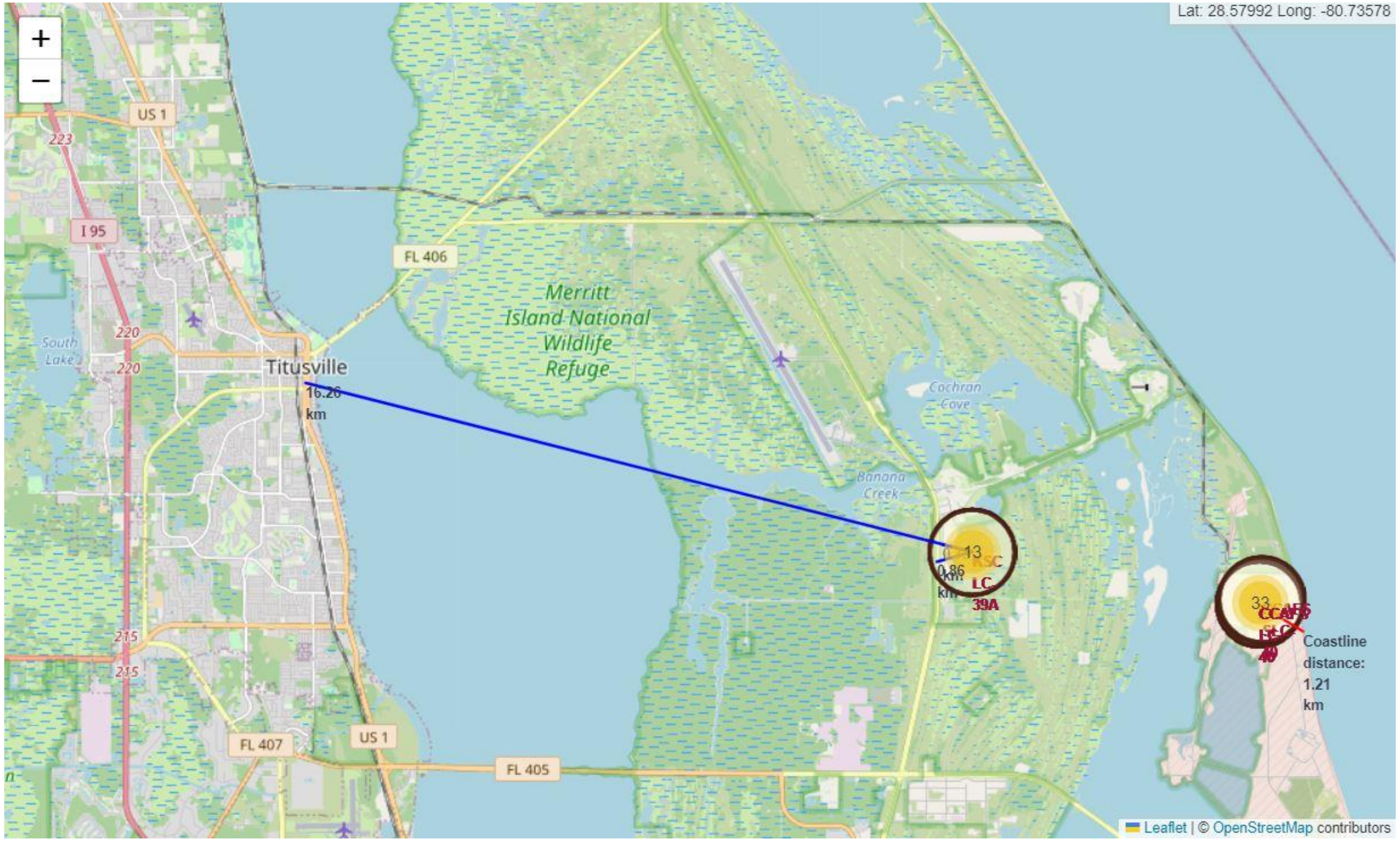
Launch Site and Proximities

- Launch site location is in close proximity to **railways**
  - Nearest railway line to the **KSC LC-39A** center is $\approx$**0.73***km* only
- Launch site location is in close proximity to **highways**
  - Nearest highway to the **KSC LC-39A** center, the **Kennedy Parkway**, is $\approx$**0.86***km* only



Launch Site and Proximities

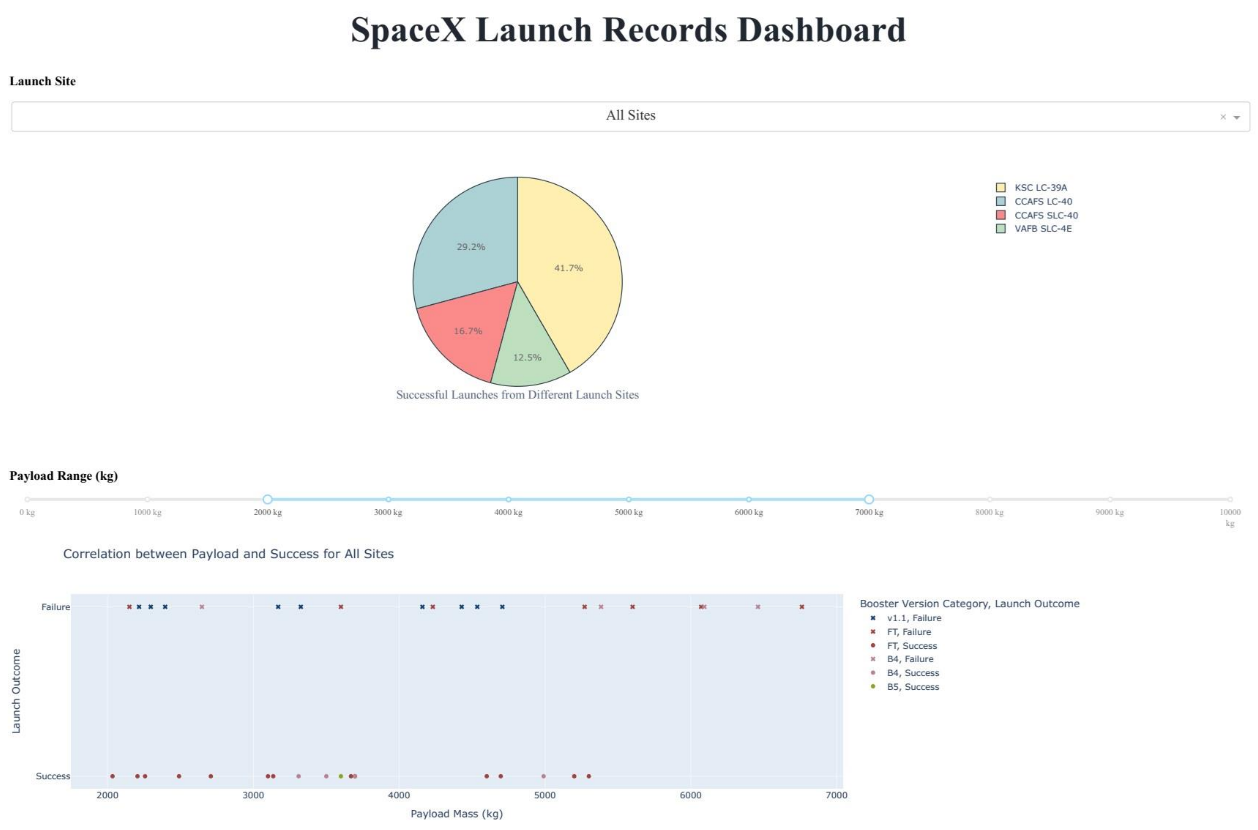
- *Compared to railway lines, highways, and coastlines*, the launch sites seem to be at a **larger distance away** from the **cities**
- Nearest city, **Titusville**, is ≈**16.26***km* from the **KSC LC-39A** center



Dashboard- Breathing Life into Data

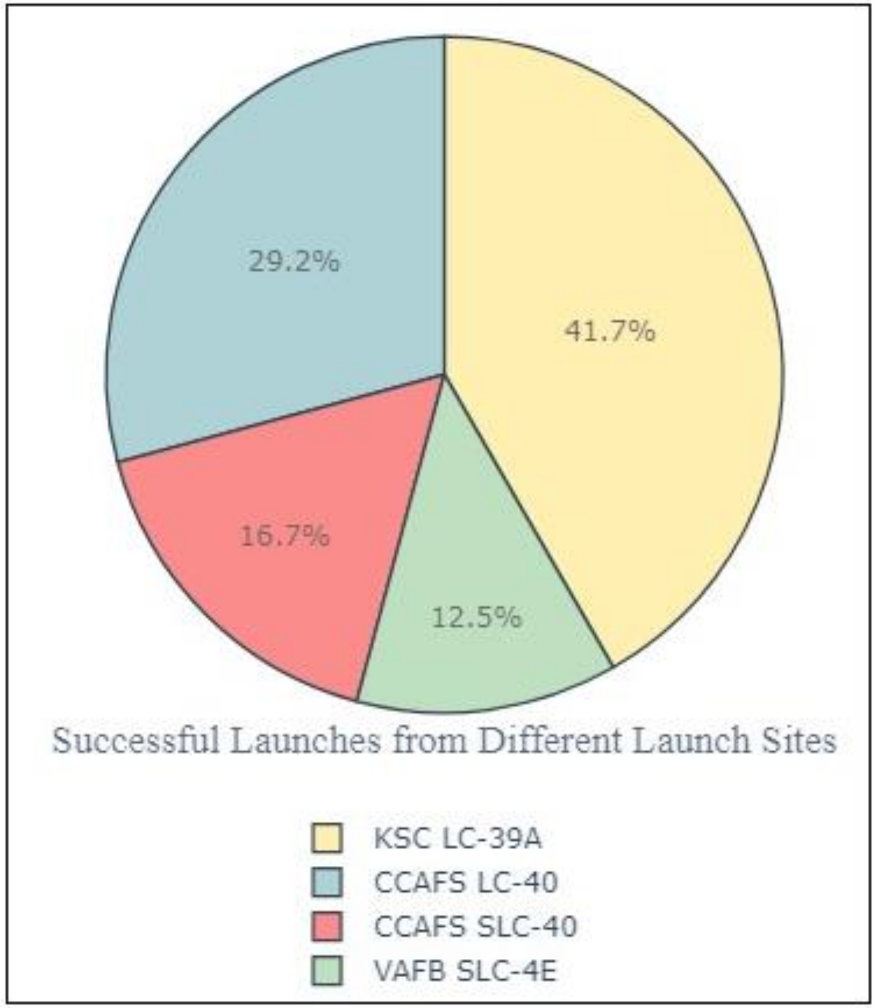
Dashboard with Plotly Dash

- Open-source Python library **Plotly** and **Dash** was used to build an **interactive dashboard** for **SpaceX past launch records**



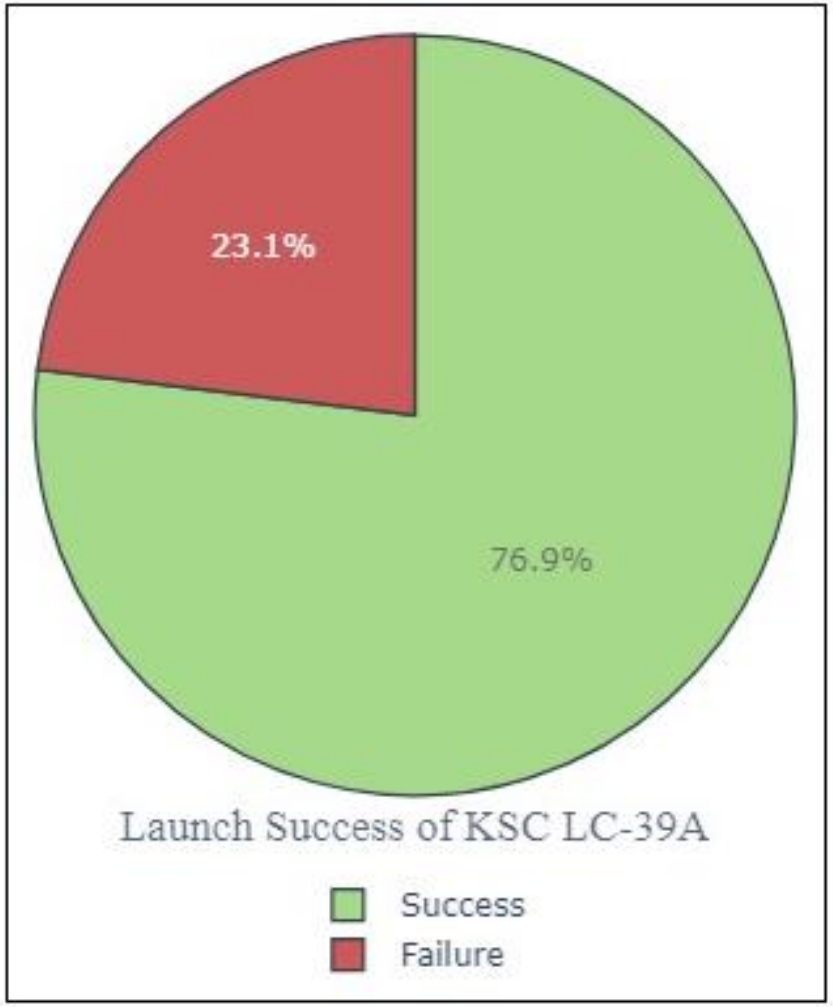
Success Stories- SpaceX Launch sites

- Almost **half** of the past successful missions were launched from **KSC LC-39A**
- **CCAFS** and **VAFB SLC-4E** both made approximately **equal** contribution to past successes

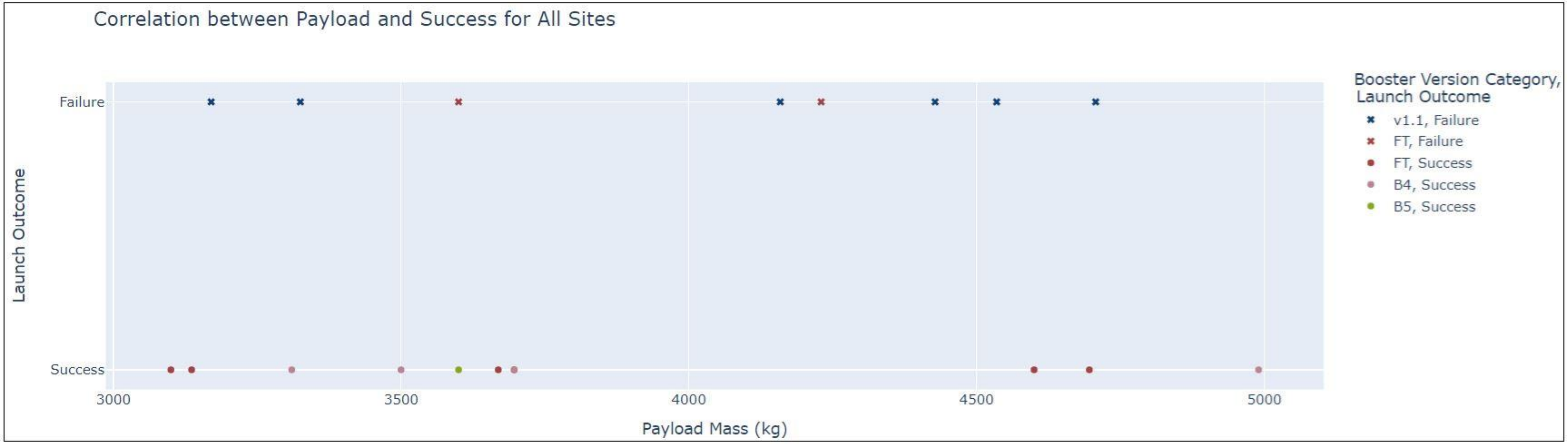


Kennedy Space Center- A Winner

- More than **two-thirds** of missions launched from **Kennedy Space Center Launch Complex** (KSC LC-39A) were successful.
- Hidden insights- Studying the **location**, **infrastructure**, mission **payloads**, target **orbits** etc. of missions may guide building/choosing appropriate launch sites in future.



Payload & Success Rate- Are they Correlated?



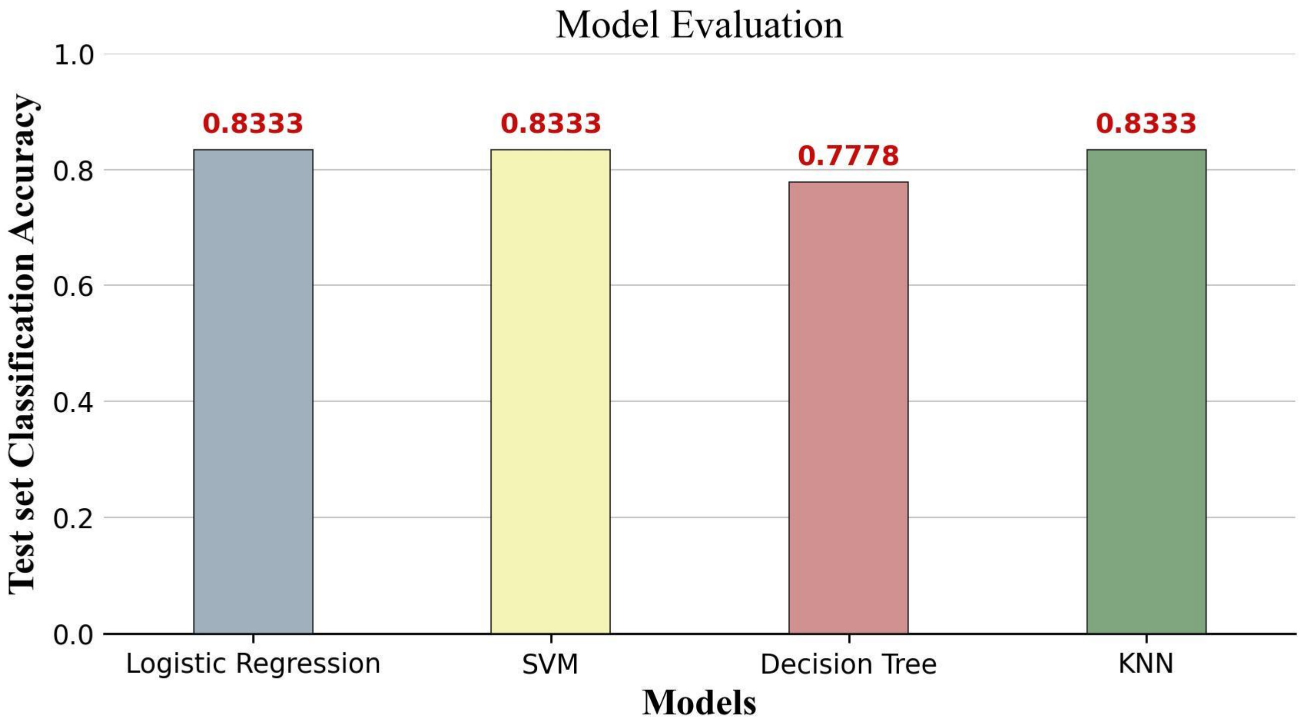
- **Success** rate & **payload mass** (kg) seems **inversely** correlated
- Payloads **below 4000 kg** have seen **higher successful outcomes** than failures.
- Falcon 9 **Full Thrust booster** (FT) has seen the **highest success** rate
  - Studying the FT booster **engineering** along with **payload mass dynamics** may be key to increasing success in this space race.

Predictive Analysis

Projecting the Future

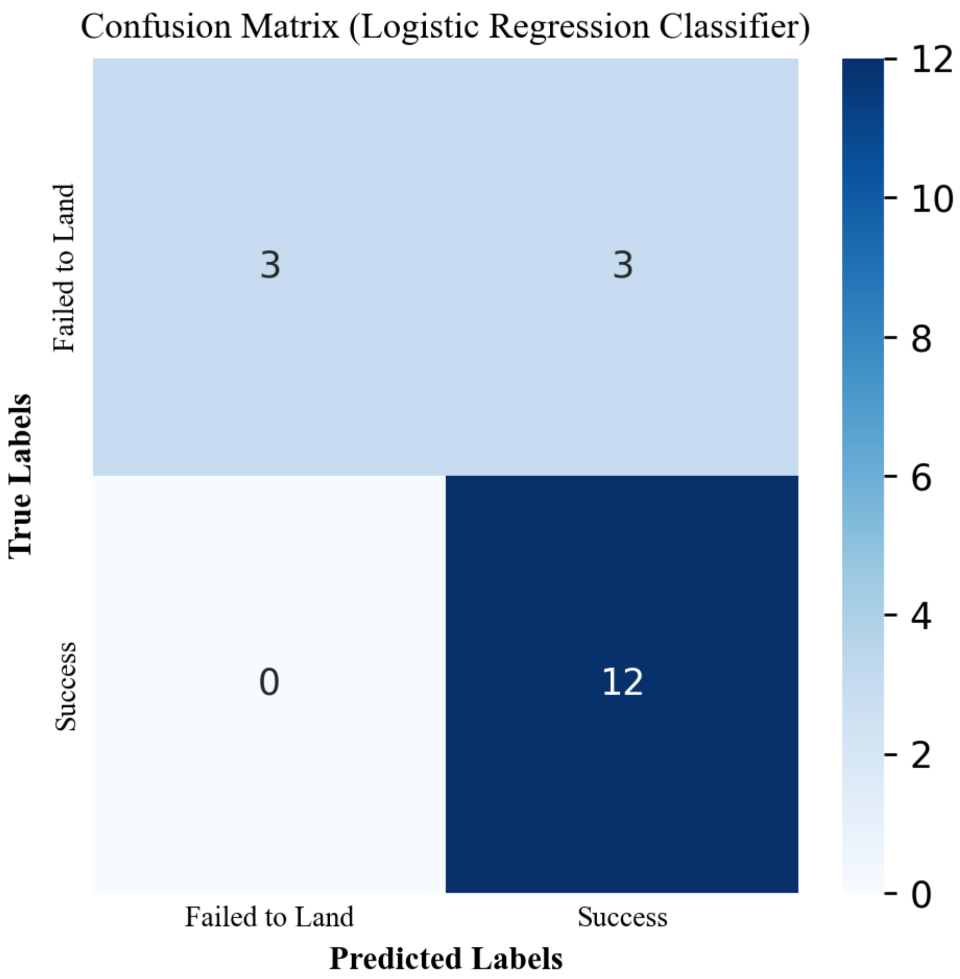
Classification Accuracy

- Classification Accuracy
  - Ratio of number of **correct predictions** and the total **number of predictions** (Test set size)
- **3 models** performed approximately **equally** on Test set data
- **Decision Tree classifier** accuracy was **lowest** among all



Confusion Matrix

- Confusion matrix for models were plotted for visualizing its performance across multiple metrics
  - **True Negatives** ($C0,0$) = 3
  - **False Positives** ($C0,1$) = 3
  - **False Negatives**($C1,0$) = 0
  - **True Positives** ($C1,1$) =12
- Can extract other metrics
  - **Precision**
    - $\frac{True\ Positive}{True\ Positive + False\ Positive}$
  - **Recall**
    - $\frac{True\ Positive}{True\ Positive + False\ Negative}$
  - **F1 Score**
    - $\frac{Precision \times Recall}{Precision + Recall}$



Conclusions

Conclusions

- **Patience** is important when contesting in this space race of 21st century.
  - Preliminary years of a newcomer competitor company may not see as much success in reusing the rocket by landing back successfully, thus raising cost projections.
  - With **insights drawn from data** such as demonstrated in previous slides even a newcomer company can overcome this hurdle to a large extent.
- As the **spatial analysis** revealed, selecting **launch site locations** is a strategic decision having influence on mission outcomes, hence not to be taken lightly.
  - Locations close to the **Equator**, preferably near the **Eastern coastline** are better suited for building a launch site.
  - Launch site location must have following features-
    - Good rail and road **connectivity to hinterland**.
    - Far away from areas with high **population density** such as towns and cities.

Conclusions

- Quality of **predictive analysis** depends on the chosen machine learning model and how well it **generalizes** to new unseen data.
- The problem of predicting **whether rocket's first stage will successfully land** back is an incredibly complex task with multiple factors each influencing the outcome in own different ways.
- Further options such as further model analysis, training other classification models and **deep learning** approaches may be explored to find better candidate solutions than the ones discussed in previous slides.
- Analyzing models further or other metrics such as **F1-score**, **Precision**, **Recall**.
    - Model with **higher Precision** may be more suitable in case of a **tie**
        - Launching a rocket is an expensive, resource-intensive affair with too many sensitive input factors
        - Ensures that model has **low number of false positives** i.e. if model predicts a successful landing then it is more likely to be correct even if that means missing some potential cases where outcome could also have been successful (*higher false negatives*)
- Collecting **more data** to train the model in case evaluation indicates high **variance** in model.
- **Feature engineering** to try out more complex models in case evaluation indicates high **bias** in model.

Conclusions

- Today's **space economy** offers plethora of hope in terms of shared value, driving innovation and world growth.
- The **data boom** seen in 21^{st} century along with newer tools at our disposal such as **data analytics** and **machine learning** enables everyone to reap the fruits of this flourishing space.
- As demonstrated in this report, **data driven decisions** will enable any new participant to position itself for glory in this new space race.

Appendix

- **Python code** snippets, **Jupyter Notebooks**, the **visualizations**, and PDF file of this **project report** can be accessed from this public [GitHub repository](https://github.com/debatreyo/space-race)-
  https://github.com/debatreyo/space-race

Thank you