IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Manuel E. Cano Nesbet
18-05-2025

Outline

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

Executive Summary

Summary of Methodologies:

- This project follows these steps:
  - Data Collection
  - Data Wrangling
  - Exploratory Data Analysis
  - Interactive Visual Analytics
  - Predictive Analysis (Classification)

Summary of Results:

- This project produced the following outputs and visualizations:
  - Exploratory Data Analysis (EDA) results
  - Geospatial analytics
  - Interactive dashboard
  - Predictive analysis of classification models

Introduction

- SpaceX's Falcon 9 rockets are launched at a significantly lower cost, around $62 million per flight, compared to the industry average of over $165 million.
- A major reason for this cost advantage is the company's ability to recover and reuse the first stage of its rockets, reducing overall mission expenses.
- Accurately predicting whether the first stage of the Falcon 9 will successfully land is critical for estimating the true cost of a launch.
- This insight can help other companies decide whether to compete with SpaceX for contracts in the satellite launch market.
- The goal of this project is to develop a model that can reliably forecast the likelihood of a successful landing for the Falcon 9 first stage.

Section 1

Methodology

Methodology

Data Collection

- Making GET requests to the SpaceX REST API
- Web Scraping

Data Wrangling

- Using the .fillna() method to remove NaN values
- Using the .value_counts() method to determine the following:
  - Number of launches on each site
  - Number and occurrence of each orbit
  - Number and occurrence of mission outcome per orbit type
- Creating a landing outcome label that shows the following:
  - 0 when the booster did not land successfully
  - 1 when the booster did land successfully

Exploratory Data Analysis

- Using SQL queries to manipulate and evaluate the SpaceX dataset
- Using Pandas and Matplotlib to visualize relationships between variables, and determine patterns

Interactive Visual Analytics

- Geospatial analytics using Folium
- Creating an interactive dashboard using Plotly Dash

Data Modelling and Evaluation

- Using Scikit-Learn to:
  - Pre-process (standardize) the data
  - Split the data into training and testing data using train_test_split
  - Train different classification models
  - Find hyperparameters using GridSearchCV
- Plotting confusion matrices for each classification model
- Assessing the accuracy of each classification model

Data Collection

To gather comprehensive data for this analysis, a combination of API requests from the SpaceX REST API and web scraping from SpaceX's Wikipedia entry was employed. This approach was necessary to ensure complete and accurate data coverage for each launch.

Data Columns Collected via SpaceX REST API:

- FlightNumber, Date, BoosterVersion, PayloadMass, Orbit, LaunchSite, Outcome, Flights, GridFins, Reused, Legs, LandingPad, Block, ReusedCount, Serial, Longitude, Latitude

Data Columns Collected via Wikipedia Web Scraping:

- Flight No., Launch Site, Payload, Payload Mass, Orbit, Customer, Launch Outcome, Booster Version, Booster Landing, Date, Time

Data Collection – SpaceX API

Requesting	Requesting rocket launch data from SpaceX API
Decoding	Decoding the response content using .json() and turning it into a dataframe using .json_normalize()
Requesting	Requesting needed information about the launches from SpaceX API by applying custom functions
Constructing	Constructing data we have obtained into a dictionary
Creating	Creating a dataframe from the dictionary
Filtering	Filtering the dataframe to only include Falcon 9 launches
Replacing	Replacing missing values of Payload Mass column with calculated .mean() for this column
Exporting	Exporting the data to CSV

GitHub URL: Data Collection API

Data Collection - Scraping

Requesting	Requesting Falcon 9 launch data from Wikipedia
Creating	Creating a BeautifulSoup object from the HTML response
Extracting	Extracting all column names from the HTML table header
Collecting	Collecting the data by parsing HTML tables
Creating	Creating a dataframe from the dictionary
Constructing	Constructing data we have obtained into a dictionary
Exporting	Exporting the data to CSV

GitHub URL: Data Collection with web scraping

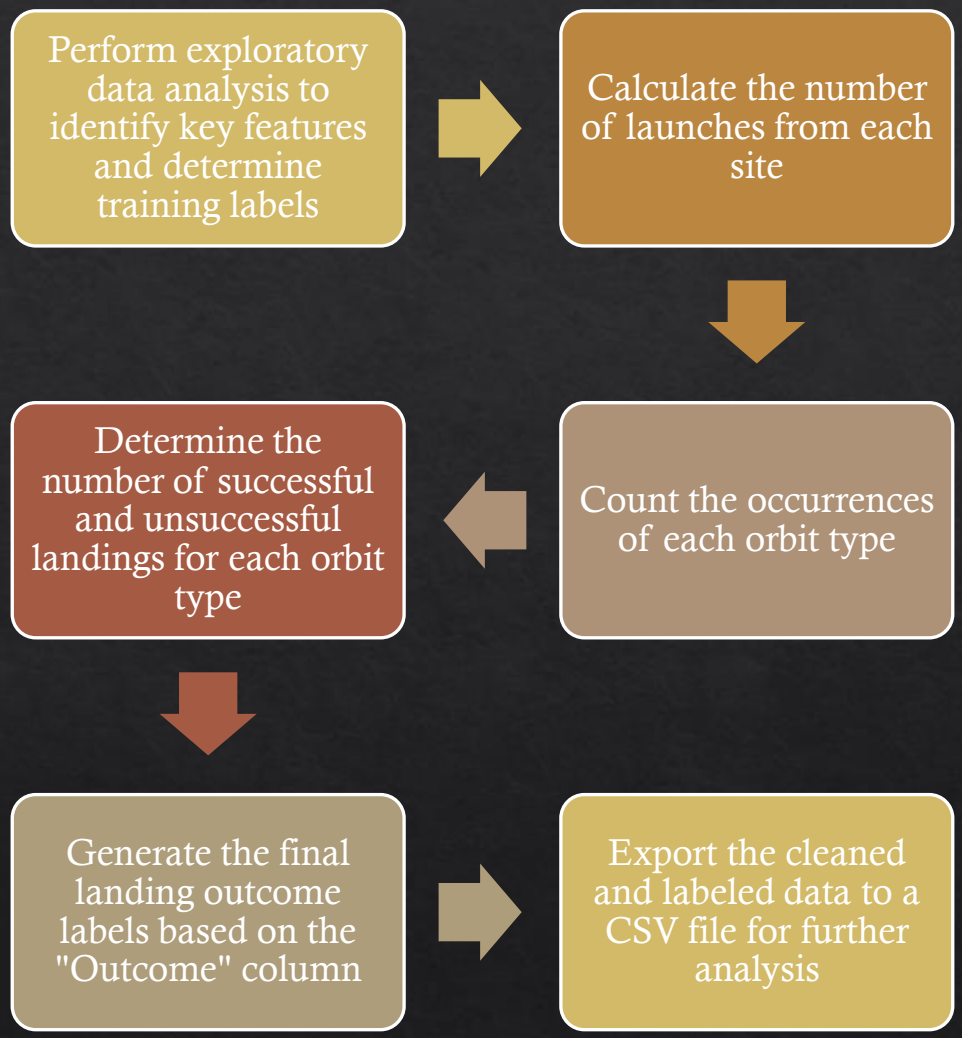
Data Wrangling

In this dataset, there are multiple possible outcomes for booster landings, not all of which are successful. For instance:

- **True Ocean**: The booster successfully landed in a designated ocean region.
- **False Ocean**: The booster attempted to land in the ocean but was unsuccessful.
- **True RTLS (Return to Launch Site)**: The booster successfully returned to a ground pad.
- **False RTLS**: The booster attempted to return to a ground pad but failed.
- **True ASDS (Autonomous Spaceport Drone Ship)**: The booster successfully landed on a drone ship.
- **False ASDS**: The booster attempted to land on a drone ship but was unsuccessful.

For training the machine learning model, these outcomes are simplified into binary labels:

- **1**: Successful landing
- **0**: Unsuccessful landing



GitHub URL: Data Wrangling

EDA with Data Visualization

Exploratory Data Analysis (EDA) with Visualizations

Charts Created:

- Flight Number vs. Payload Mass
- Flight Number vs. Launch Site
- Payload Mass vs. Launch Site
- Orbit Type vs. Success Rate
- Flight Number vs. Orbit Type
- Payload Mass vs. Orbit Type
- Yearly Trends in Success Rate

Chart Types and Their Purposes:

- **Scatter Plots:** Used to identify potential relationships between pairs of numerical variables, helping to uncover patterns that could be valuable for training machine learning models.
- **Bar Charts:** Effective for comparing discrete categories, illustrating the distribution and differences among key data points.
- **Line Charts:** Ideal for tracking trends over time, capturing variations and changes in key metrics across different periods.

GitHub URL: EDA with Data Visualization

EDA with Data Visualization

Exploratory Data Analysis (EDA) with SQL

Performed SQL Queries:

- Retrieve the unique launch site names used in space missions.
- Select the first 5 launch site names that begin with the prefix "CCA".
- Calculate the total payload mass for all boosters launched by NASA (CRS missions).
- Determine the average payload mass for the booster version "F9 v1.1".
- Find the date of the first successful ground pad landing.
- List the boosters that successfully landed on a drone ship and carried payloads between 4000 and 6000 kg.
- Count the total number of successful and unsuccessful mission outcomes.
- Identify the booster versions that have carried the maximum payload mass.
- List the failed drone ship landing outcomes, including booster versions and launch site names, for the months in 2015.
- Rank the landing outcomes (e.g., "Failure" on a drone ship or "Success" on a ground pad) between June 4, 2010, and March 20, 2017, in descending order.

GitHub URL: EDA with SQL

Build an Interactive Map with Folium

Building an Interactive Map with Folium

Launch Site Markers:

·Placed a circle marker for the NASA Johnson Space Center using its latitude and longitude coordinates as the initial focal point.

·Added circle markers for all known launch sites, including popup labels and text annotations, to display their geographic positions and relative proximity to the equator and coastlines.

Color-Coded Markers for Launch Outcomes:

·Incorporated color-coded markers to differentiate successful (green) and failed (red) launches using a marker clustering approach, making it easier to visualize the relative success rates at each launch site.

Measuring Distances to Nearby Features:

·Added colored lines to indicate distances from the KSC LC-39A launch site (used as an example) to nearby infrastructure such as railways, highways, coastlines, and the closest city.

GitHub URL: Interactive Visual Analytics with Folium

Build a Dashboard with Plotly Dash

Building a Dashboard with Plotly Dash

Launch Site Dropdown Selector:

·Implemented a dropdown menu to allow users to filter data by specific launch sites.

Pie Chart for Launch Success Rates (All Sites / Specific Site):

·Added a pie chart to visualize the proportion of successful and failed launches, either for all sites collectively or for a selected launch site.

Payload Mass Range Slider:

·Included a slider component to dynamically adjust the payload mass range, providing more control over data visualization.

Scatter Plot for Payload Mass vs. Success Rate:

·Created a scatter plot to reveal the relationship between payload mass and launch success for different booster versions, highlighting potential performance trends.

GitHub URL: SpaceX Dash App

Predictive Analysis (Classification)

Preparing the Data:	Extract a NumPy array from the "Class" column to serve as the target variable
Data Standardization:	Standardize the feature data using StandardScaler, fitting and transforming it for improved model performance
Data Splitting:	Divide the dataset into training and testing subsets using the train_test_split function to evaluate model accuracy.
Hyperparameter Tuning:	Create a GridSearchCV object with 10-fold cross-validation (cv=10) to identify the best model parameters.
Model Training and Evaluation:	Apply GridSearchCV to various models, including Logistic Regression, SVM, Decision Trees, and KNN.
Model Accuracy Assessment:	Calculate the accuracy of each model on the test data using the .score() method.
Confusion Matrix Analysis:	Generate confusion matrices to assess model performance across different classes.
Selecting the Best Model:	Use the Jaccard score and F1 score metrics to determine which model performs the best.

GitHub URL: Machine Learning prediction

Results

- Exploratory Data Analysis Results
- Interactive Analytics demo in screenshots
- Predictive Analysis Results

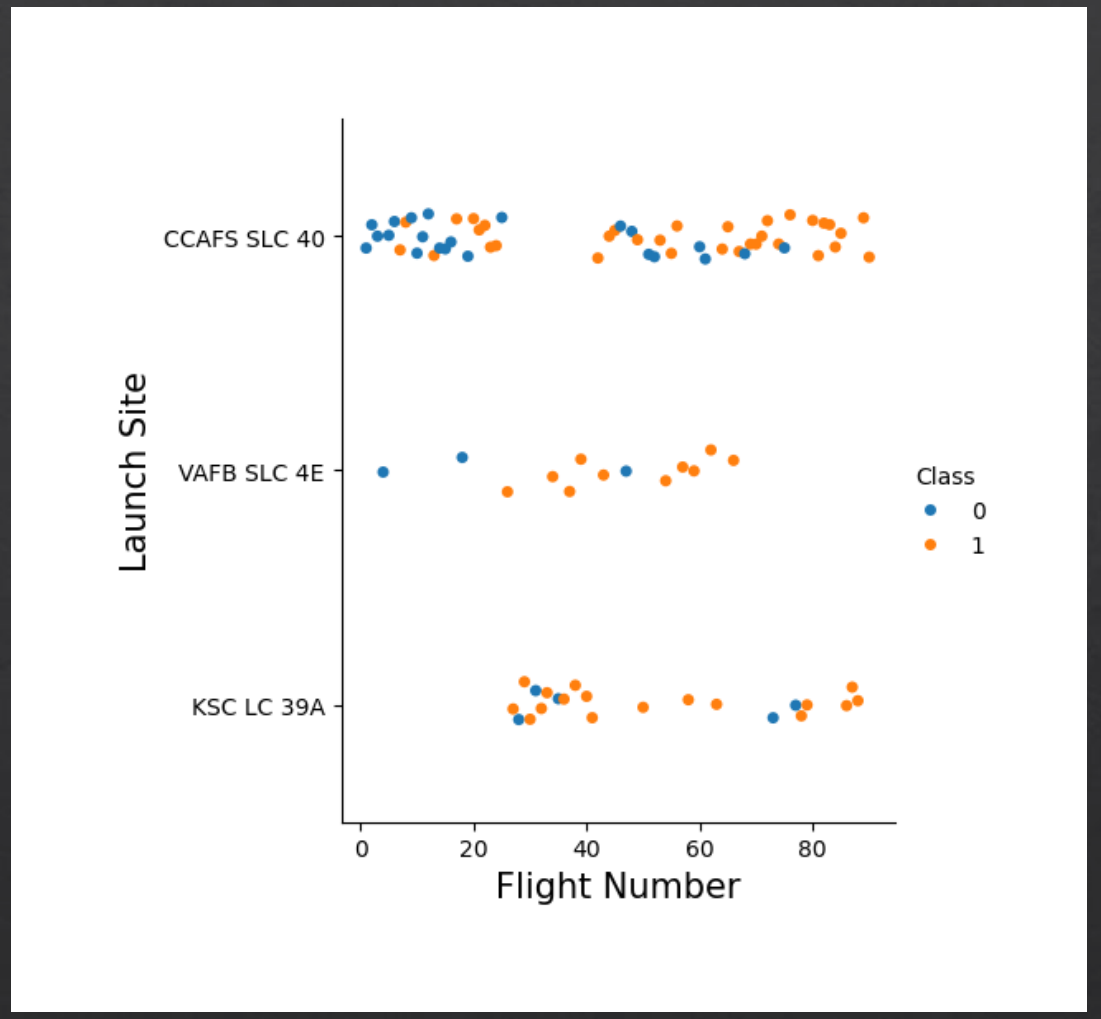
Section 2

Insights drawn from EDA

Flight Number vs. Launch Site

Insights from the Scatter Plot: Launch Site vs. Flight Number

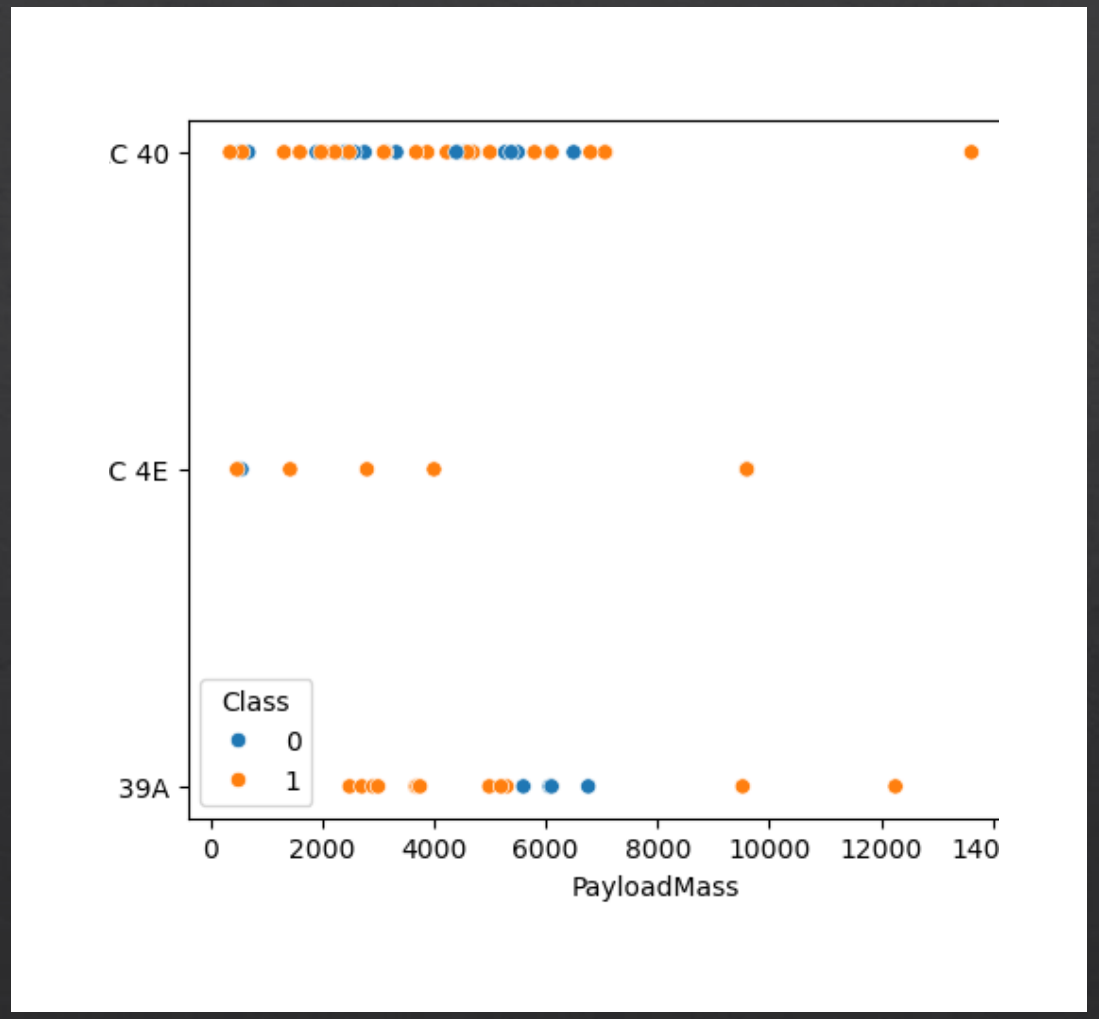
- **Increasing Success with More Flights:**
  - The overall success rate at each launch site tends to improve as the number of flights increases.
- **Early Launch Challenges at CCAFS SLC 40:**
  - Most early flights (with flight numbers below 30) were conducted from CCAFS SLC 40, and these missions had a relatively low success rate.
- **Similar Trends at VAFB SLC 4E:**
  - Early launches from VAFB SLC 4E also showed lower success rates, reinforcing the trend that early missions often faced more challenges.
- **High Success Rates at KSC LC 39A:**
  - KSC LC 39A, which did not host any early flights, has a notably higher success rate, suggesting that it benefited from more mature technology and operational experience.
- **Significant Improvement After 30 Flights:**
  - Once flight numbers surpass 30, the success rate increases significantly, indicating a learning curve effect as SpaceX refined its technology and operations.



Payload vs. Launch Site

Insights from the Scatter Plot: Launch Site vs. Payload Mass

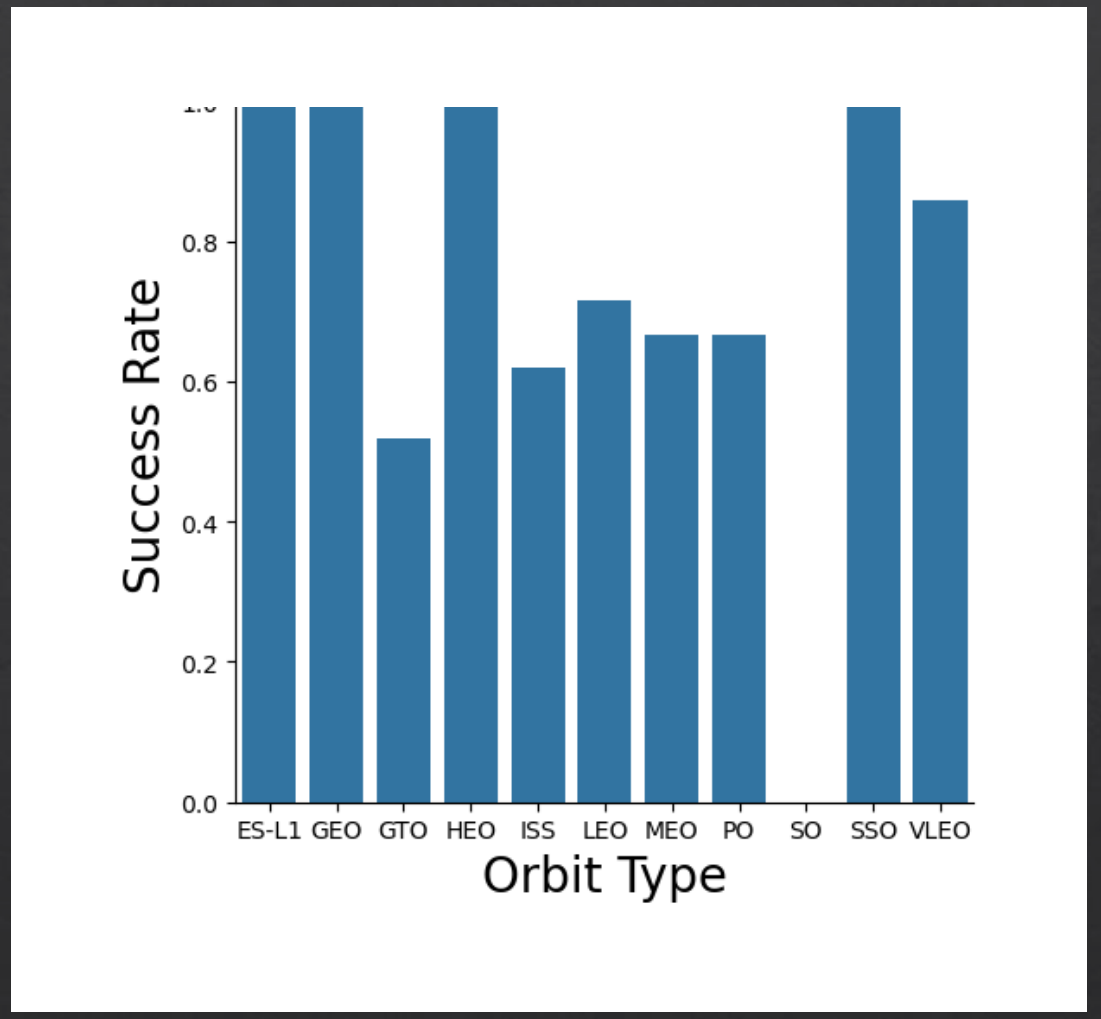
- **Heavier Payloads Show Higher Success:**
  - For payloads above approximately 7000 kg, unsuccessful landings are rare, though this may be partly due to the limited number of heavier launches in the dataset.
- **Weak Correlation Between Mass and Success:**
  - There is no strong or consistent relationship between payload mass and success rate at individual launch sites.
- **Wide Range of Payloads Across Sites:**
  - All launch sites have handled a broad range of payload masses, though CCAFS SLC 40 has a higher concentration of lighter payloads, with a few notable outliers.



Success rate vs. Orbit Type

Insights from the Bar Chart: Success Rate vs. Orbit Type

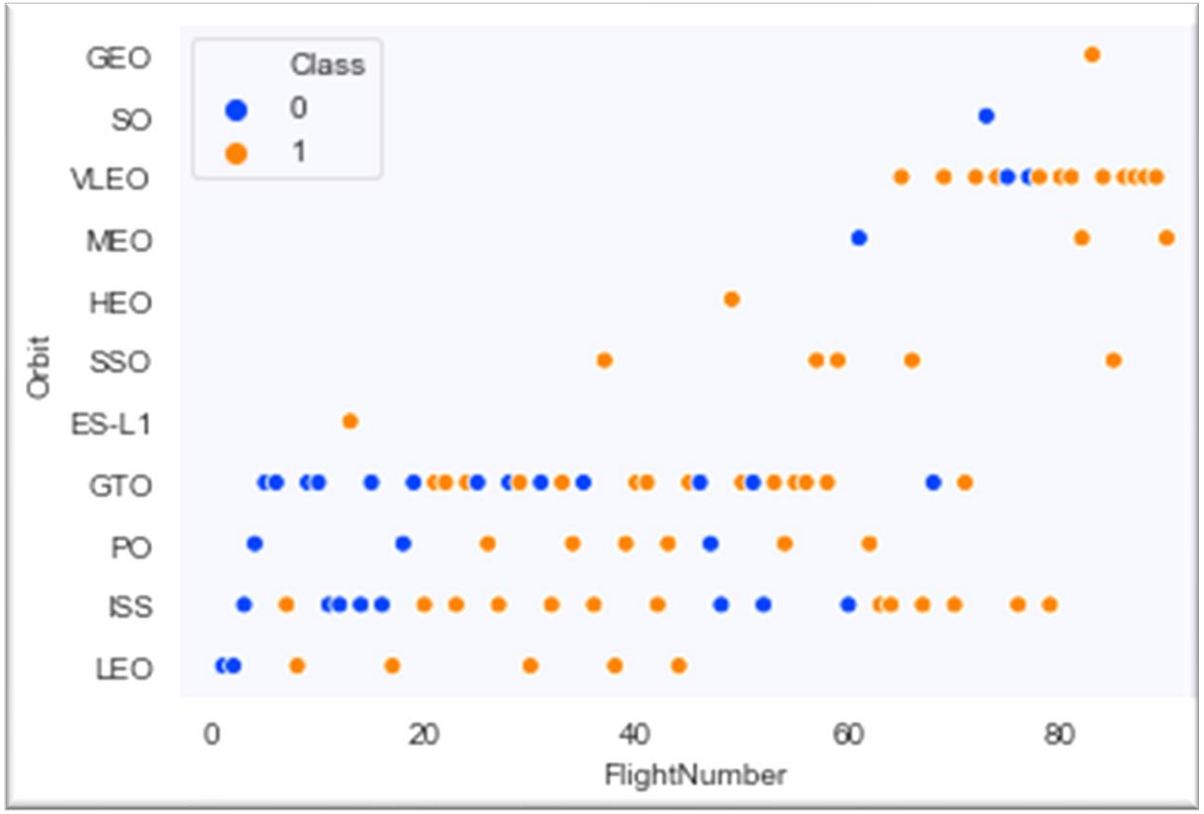
- **Orbits with a 100% Success Rate:**
  - **ES-L1 (Earth-Sun First Lagrangian Point)**
  - **GEO (Geostationary Orbit)**
  - **HEO (High Earth Orbit)**
  - **SSO (Sun-synchronous Orbit)**
- **Orbit with a 0% Success Rate:**
  - **SO (Heliocentric Orbit)**



Flight Number vs. Orbit Type

Insights from the Scatter Plot: Orbit Type vs. Flight Number

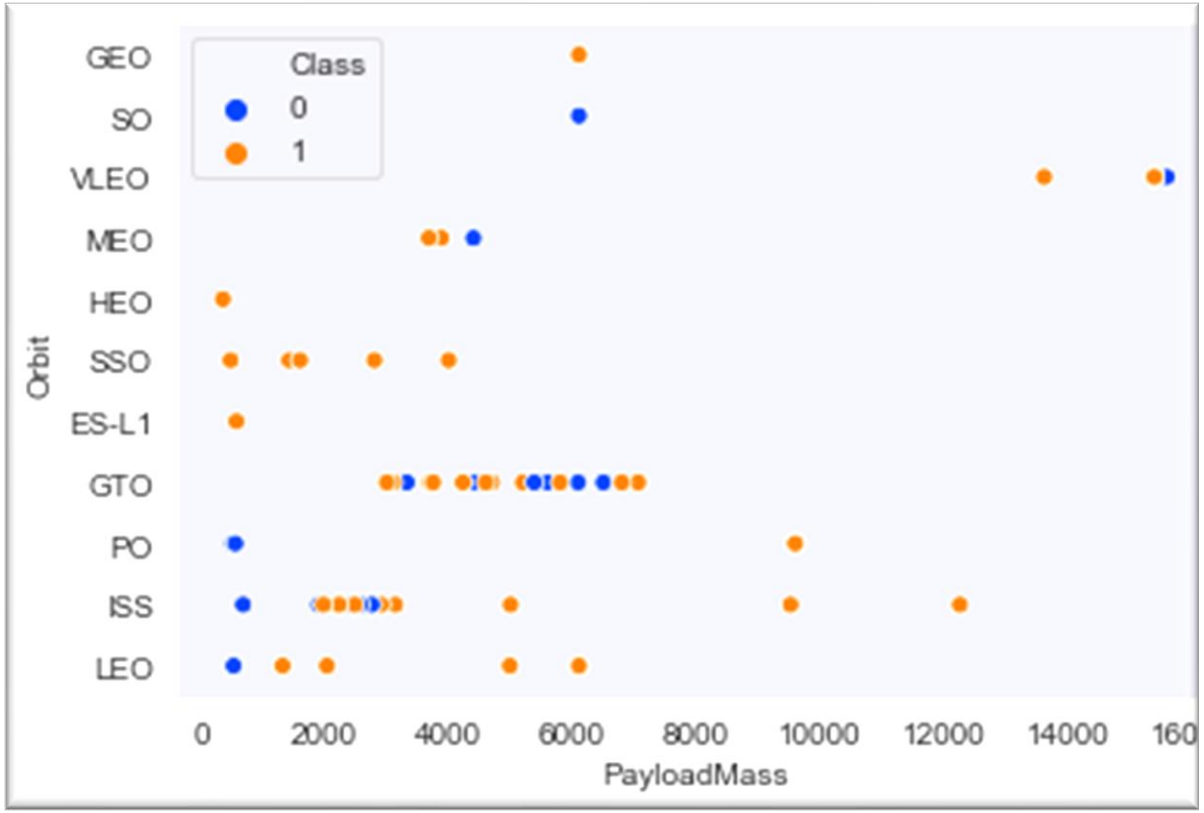
- **High Success Rates with Limited Flights:**
  - The 100% success rates for **GEO (Geostationary Orbit)**, **HEO (High Earth Orbit)**, and **ES-L1 (Earth-Sun First Lagrangian Point)** are largely due to each having only a single recorded flight.
- **Consistent Success for SSO:**
  - The **SSO (Sun-synchronous Orbit)** stands out with a more meaningful 100% success rate, achieved across **5 successful flights**.
- **Weak Correlation for GTO:**
  - There appears to be little or no clear relationship between flight number and success rate for **GTO (Geostationary Transfer Orbit)**.
- **Improving Success with Experience:**
  - In general, success rates tend to improve as flight numbers increase, especially for **LEO (Low Earth Orbit)**, where most unsuccessful landings occurred during early launches.



Payload vs. Orbit Type

Insights from the Scatter Plot: Orbit Type vs. Payload Mass

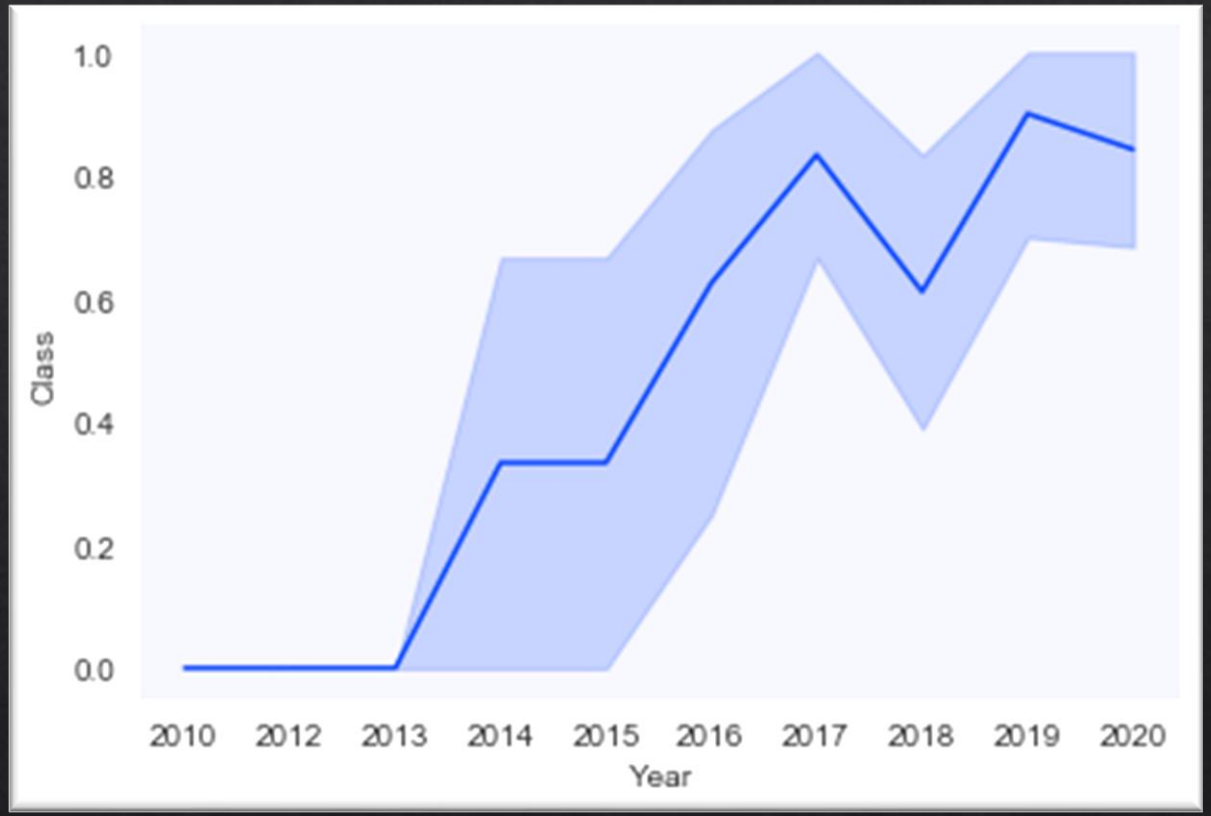
- **Orbits with Higher Success for Heavier Payloads:**
  - **PO (Polar Orbit) –** Despite having a limited number of data points, this orbit shows a trend of successful heavy payload launches.
  - **ISS (International Space Station) –** Consistently supports heavier payloads with high success rates.
  - **LEO (Low Earth Orbit) –** Also shows a positive relationship between heavier payloads and successful missions.
- **Unclear Trends for GTO:**
  - For **GTO (Geostationary Transfer Orbit)**, the relationship between payload mass and success rate remains ambiguous.
- **VLEO Association with Heavy Payloads:**
  - **VLEO (Very Low Earth Orbit)** launches are typically associated with heavier payloads, which aligns with expectations given the nature of this orbit.



Launch Success Yearly Trend

Insights from the Line Chart: Yearly Average Success Rate

- **Early Challenges (2010-2013):**
  - From 2010 to 2013, all landings were unsuccessful, resulting in a 0% success rate during this period.
- **Steady Improvement After 2013:**
  - Success rates generally improved after 2013, although there were minor setbacks in 2018 and 2020.
- **Consistent High Performance After 2016:**
  - Since 2016, the success rate has consistently remained above 50%, indicating a significant increase in overall reliability.



All Launch Site Names

```
%sql select distinct launch_site from SPACEXTBL;
```

Displaying 5 records where launch sites begin with the string 'CCA'.

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

The word `distinct` returns only unique values from the `LAUNCH_SITE` column of the `SPACEXTBL` table.

Launch Site Names Begin with 'CCA'

```
%sql select * from SPACEXTBL where launch_site like 'CCA%' limit 5;
```

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

`limit 5` fetches only 5 records, and the `like` keyword is used with the wild card `'CCA%'` to retrieve string values beginning with 'CCA'.

Total Payload Mass

```
%sql select sum(payload_mass__kg_) as total_payload_mass from SPACEXTBL where customer = 'NASA (CRS)';
```

Displaying the total payload mass carried by boosters launched by NASA (CRS)

total_payload_mass
45596

Average Payload Mass by F9 v1.1

```
%sql select avg(payload_mass__kg_) as average_payload_mass from SPACEXTBL where booster_version like '%F9 v1.1%';
```

Displaying average payload mass carried by booster version F9 v1.1

average_payload_mass
2534.6666666666665

First Successful Ground Landing Date

```
%sql select min(date) as first_successful_landing from SPACEXTBL where Landing_Outcome = 'Success (ground pad)';
```

Listing the date when the first successful landing outcome in ground pad was achieved

first_successful_landing
2015-12-22

Successful Drone Ship Landing with Payload between 4000 and 6000

```
%sql select booster_version from SPACEXTBL where Landing_Outcome = 'Success (drone ship)' and PAYLOAD_MASS__KG_ between 4000 and 6000;
```

Listing the names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000

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

Total Number of Successful and Failure Mission Outcomes

```
%sql select mission_outcome, count(*) as total_number from SPACEXTBL group by mission_outcome;
```

Listing the total number of successful and failure mission outcomes

Mission_Outcome	total_number
Failure (in flight)	1
Success	98
Success	1
Success (payload status unclear)	1

Boosters Carried Maximum Payload

```
%sql select booster_version from SPACEXTBL where payload_mass__kg_ = (select max(payload_mass__kg_) from SPACEXTBL);
```

Listing the names of the booster versions which have carried the maximum payload mass

Booster_Version
F9 B5 B1048.4
F9 B5 B1049.4
F9 B5 B1051.3
F9 B5 B1056.4
F9 B5 B1048.5
F9 B5 B1051.4
F9 B5 B1049.5
F9 B5 B1060.2
F9 B5 B1058.3
F9 B5 B1051.6
F9 B5 B1060.3
F9 B5 B1049.7

2015 Launch Records

```
%%sql
SELECT
    strftime('%m', date) AS month,
    date,
    Booster_Version,
    launch_site,
    Landing_Outcome
FROM SPACEXTBL
WHERE
    Landing_Outcome = 'Failure (drone ship)'
    AND substr(date, 1, 4) = '2015';
```

Listing the failed landing outcomes in drone ship, their booster versions and launch site names for the months in year 2015

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

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

```
%%sql select landing_outcome, count(*) as count_outcomes from SPACEXTBL
      where date between '2010-06-04' and '2017-03-20'
      group by landing_outcome
      order by count_outcomes desc;
```

Landing_Outcome	count_outcomes
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

Ranking the count of landing outcomes (such as Failure (drone ship) or Success (ground pad)) between the date 2010-06-04 and 2017-03-20 in descending order

Section 3

Launch Sites Proximities Analysis

All launch sites' location markers on a global map

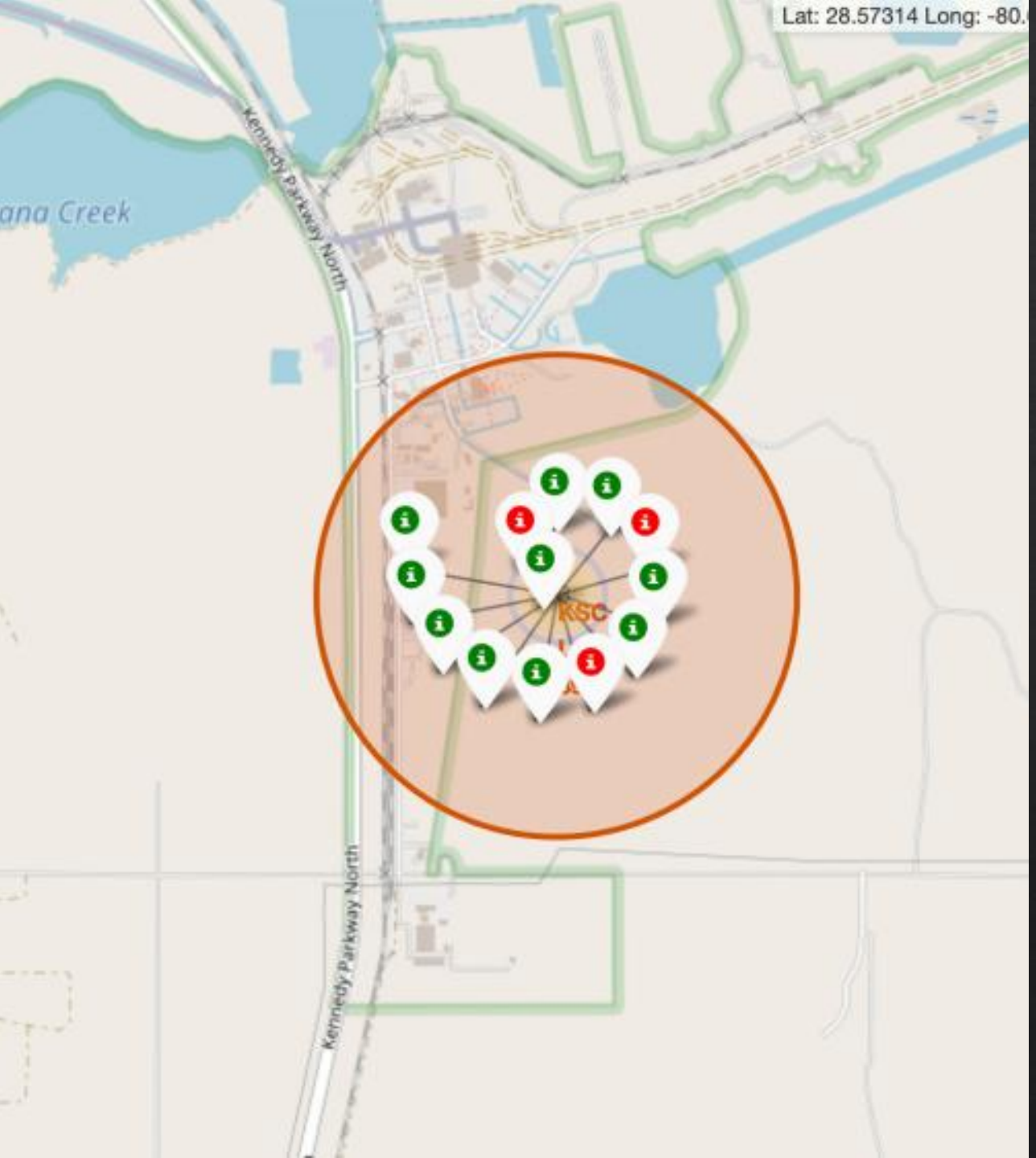
Proximity to the Equator:

Most launch sites are located close to the equator. This is advantageous because the rotational speed of the Earth is greatest at the equator, approximately **1670 km/hour**.

Launching from this region provides a "speed boost" due to inertia, as the spacecraft already has this eastward velocity when it leaves the ground, reducing the amount of fuel needed to reach orbital speed.

Coastal Locations for Safety:

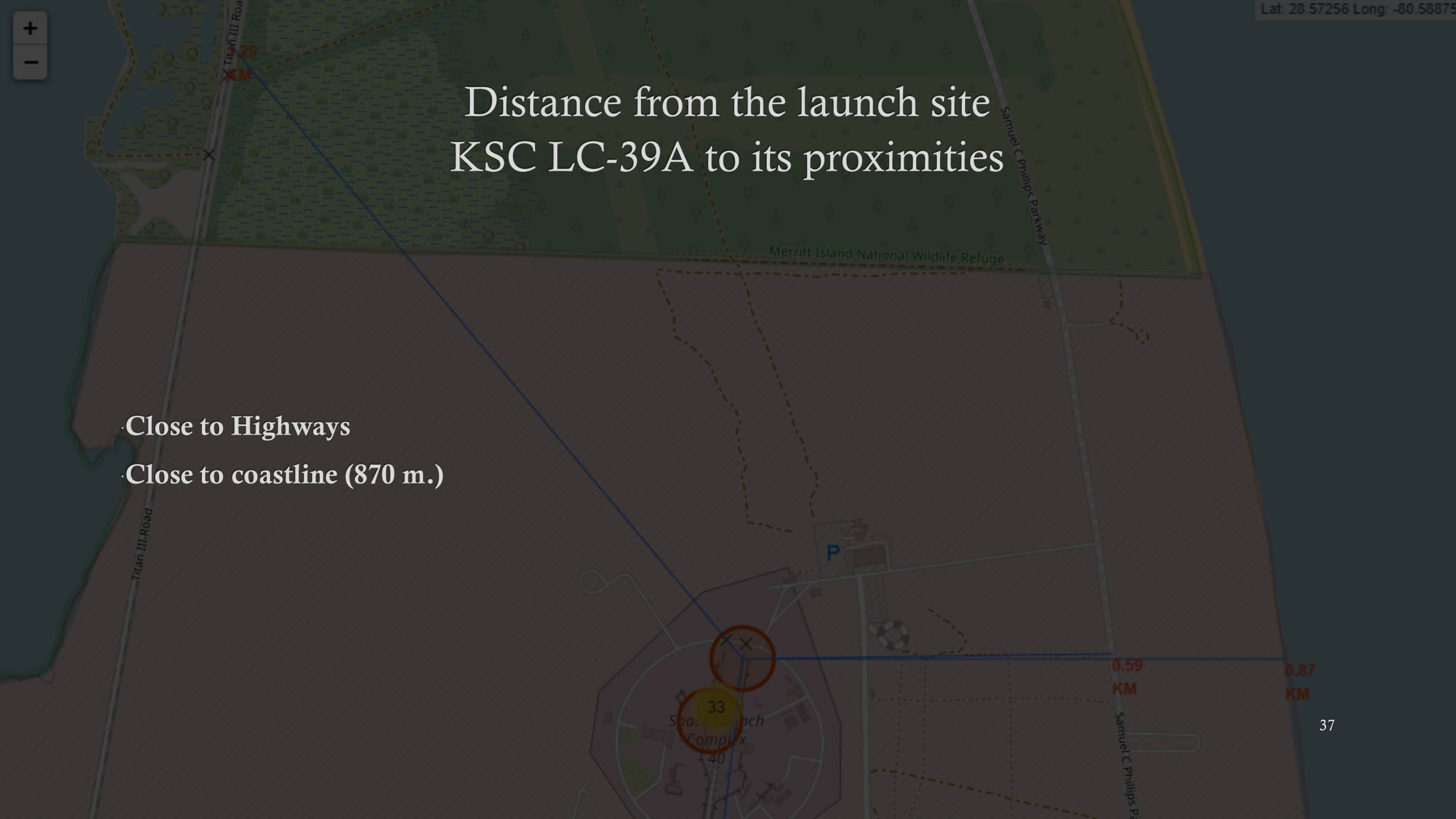
Launch sites are typically near coastlines to ensure that rockets launch over open water. This minimizes the risk to populated areas from potential debris, launch failures, or explosions, making the process significantly safer.



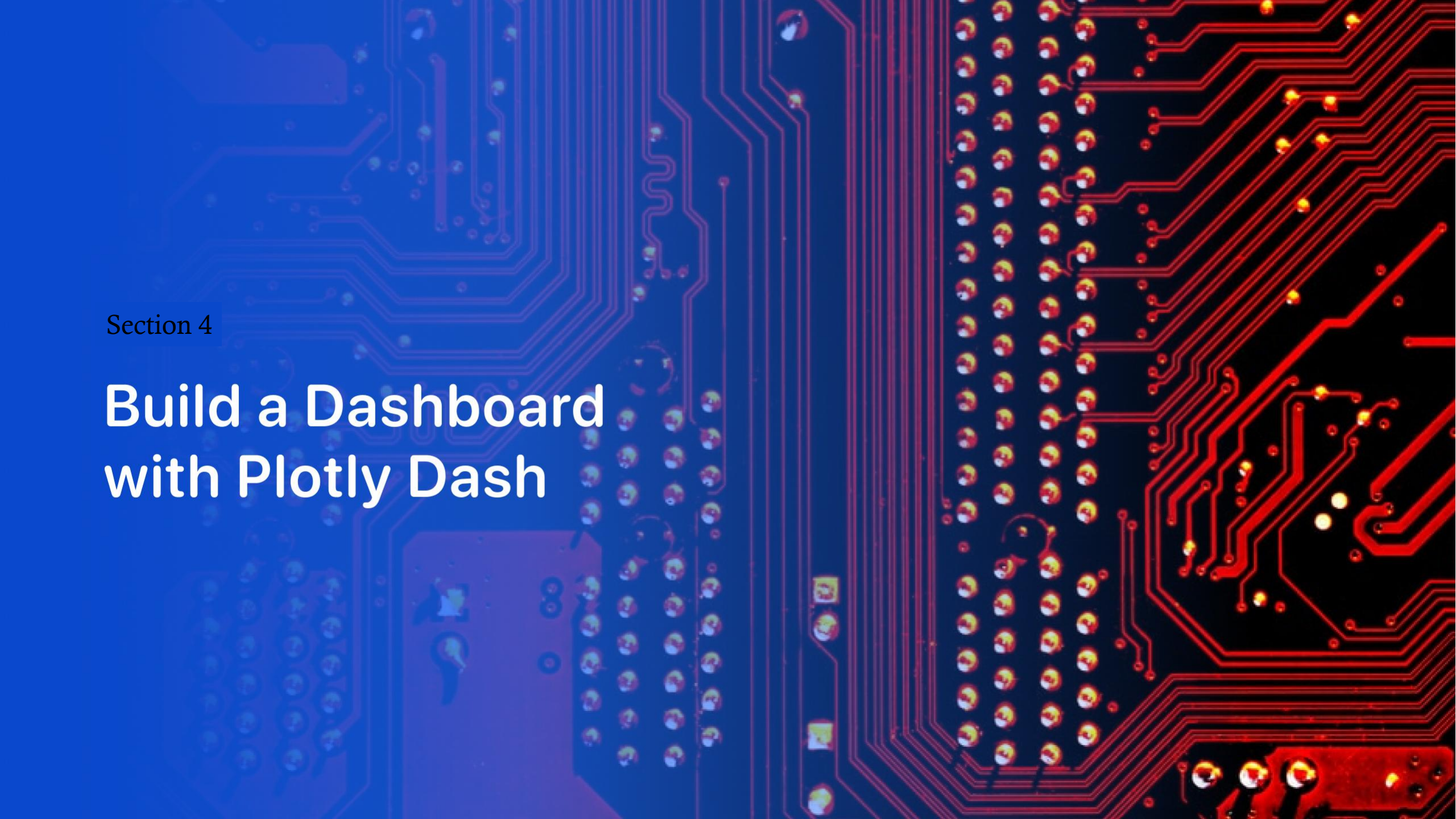
Colour labeled launch records on the map

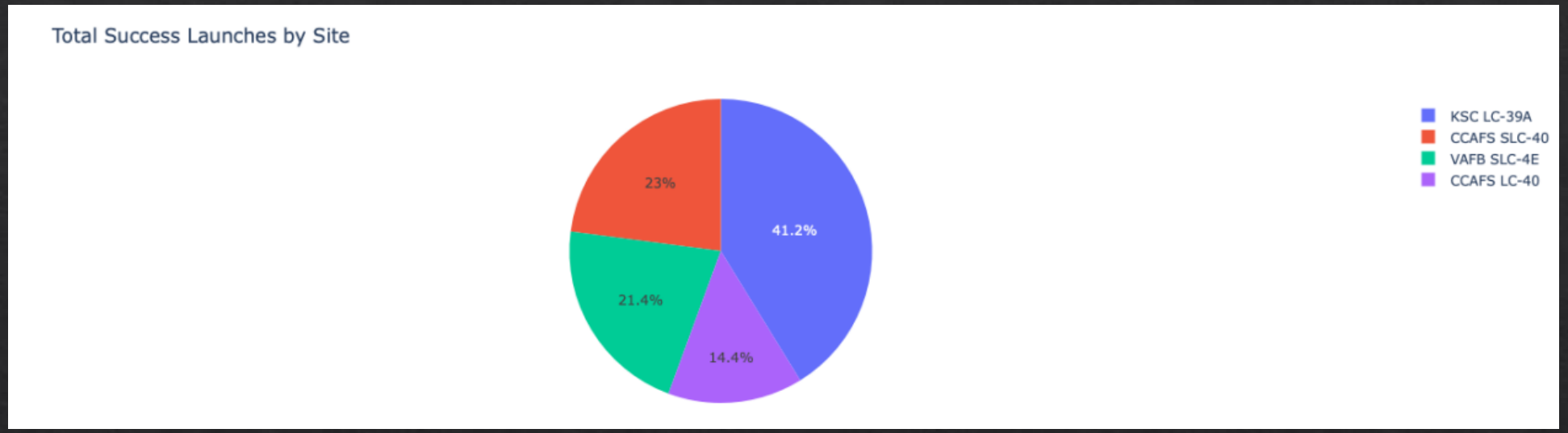
- **Color-Coded Success Indicators:**
- The use of color-labeled markers makes it straightforward to identify which launch sites have relatively high success rates.
- **Green Marker** = Successful Launch
- **Red Marker** = Failed Launch
- **High Success at KSC LC-39A:**
- The **KSC LC-39A** launch site stands out for its notably high success rate, indicating reliable performance over multiple missions

Distance from the launch site KSC LC-39A to its proximities



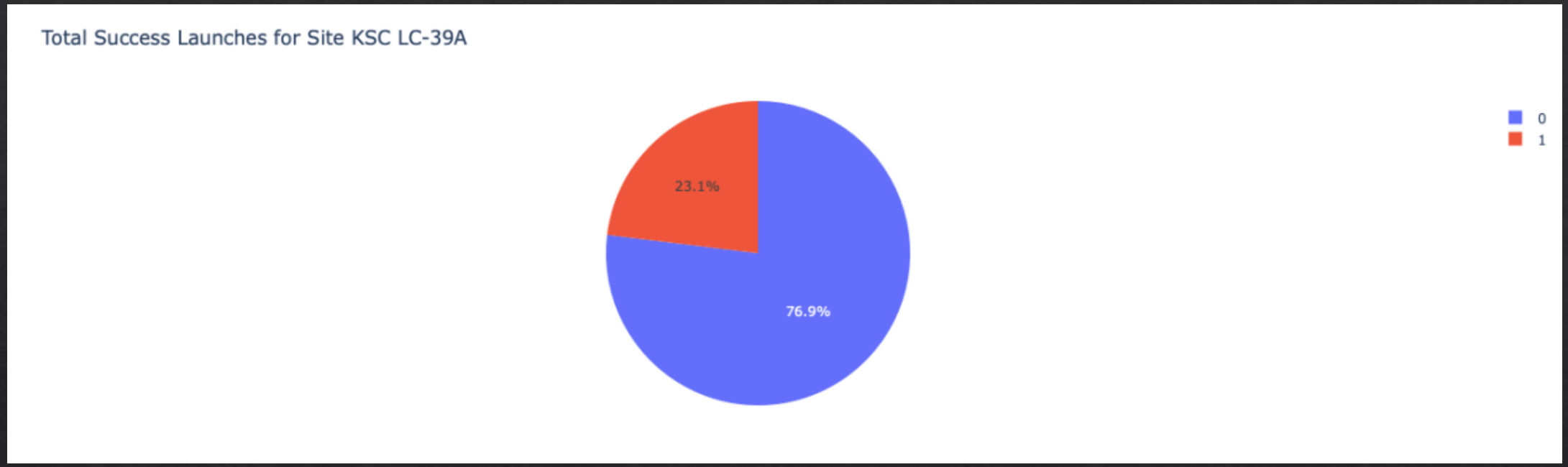
- Close to Highways
- Close to coastline (870 m.)


Section 4
Build a Dashboard with Plotly Dash



Launch success count for all sites

The chart clearly shows that from all the sites, KSC LC-39A has the most successful launches

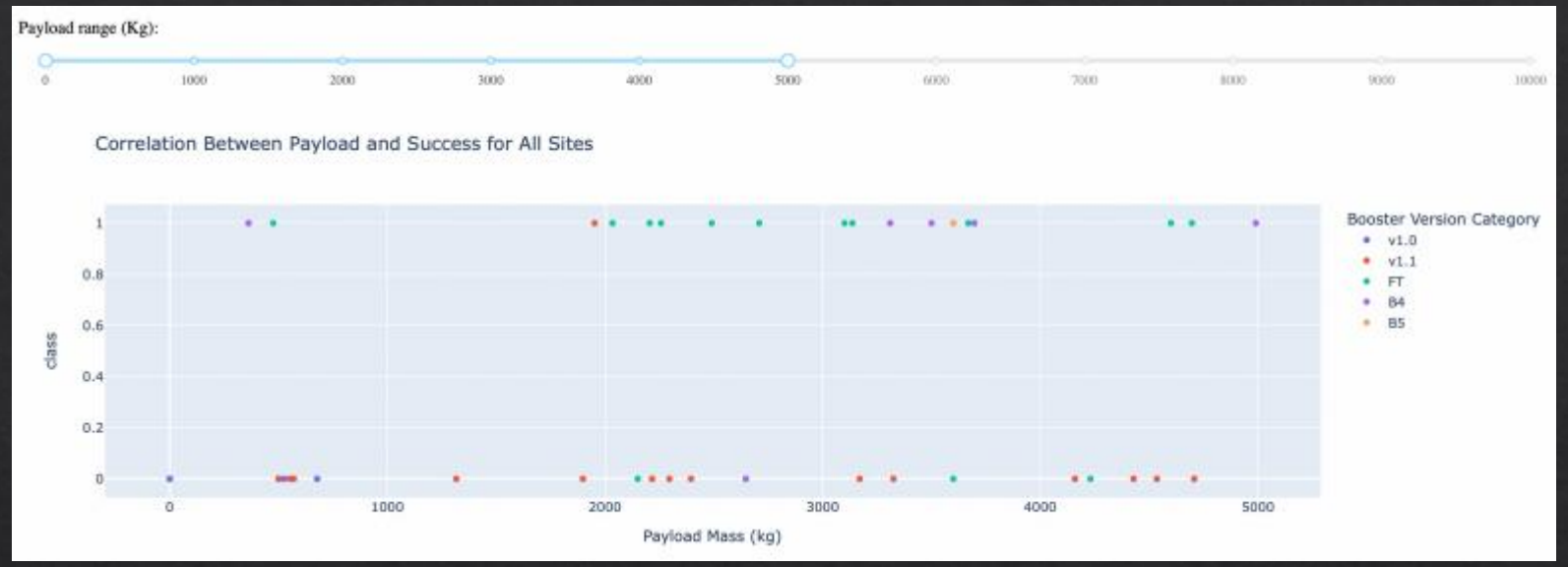


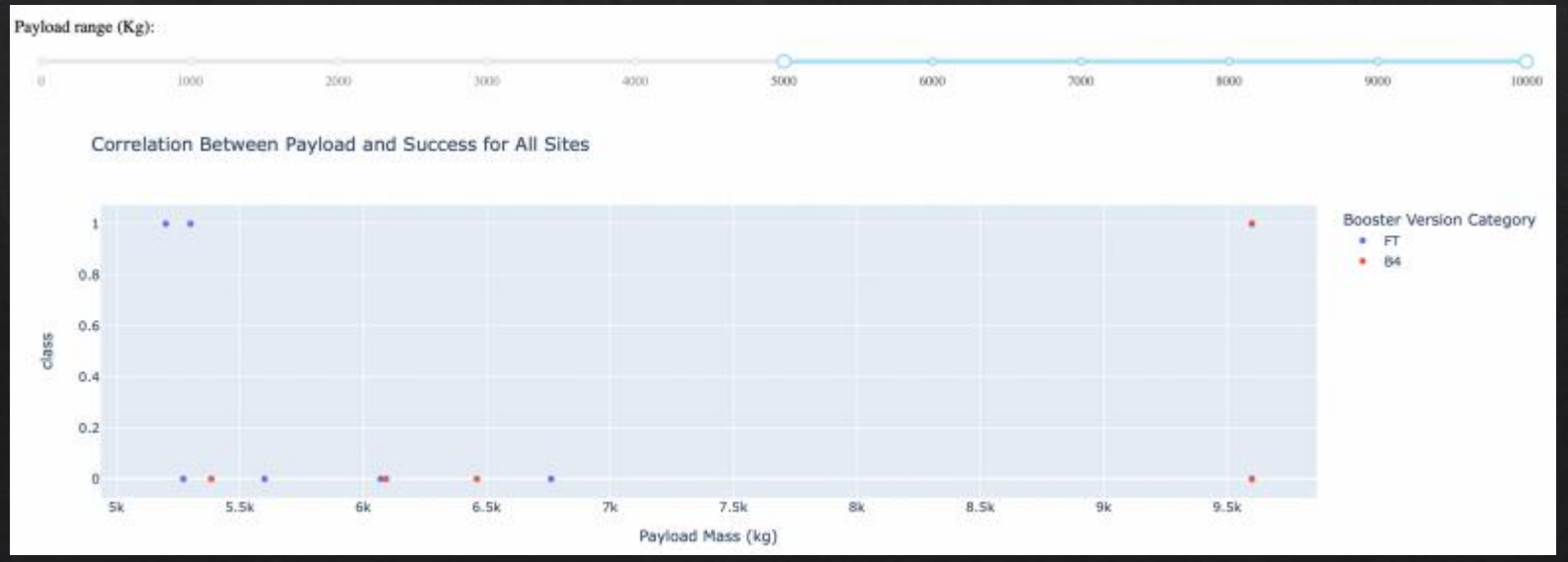
Launch site with highest launch success ratio

- KSC LC-39A has the highest launch success rate (76.9%) with 10 successful and only 3 failed landings.

Payload Mass vs. Launch Outcome for all sites

The charts show that payloads between 2000 and 5500 kg have the highest success rate



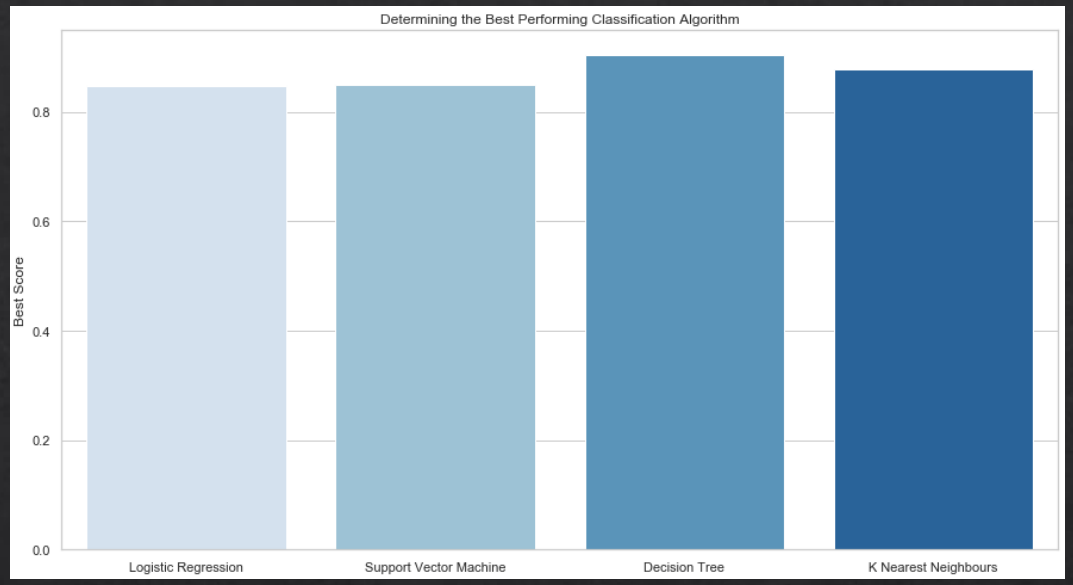


Section 5

Predictive Analysis (Classification)

Classification Accuracy

Algorithm	Accuracy Score	Best Score
Logistic Regression	0.833333	0.846429
Support Vector Machine	0.833333	0.848214
Decision Tree	0.944444	0.903571
K Nearest Neighbours	0.888889	0.876786

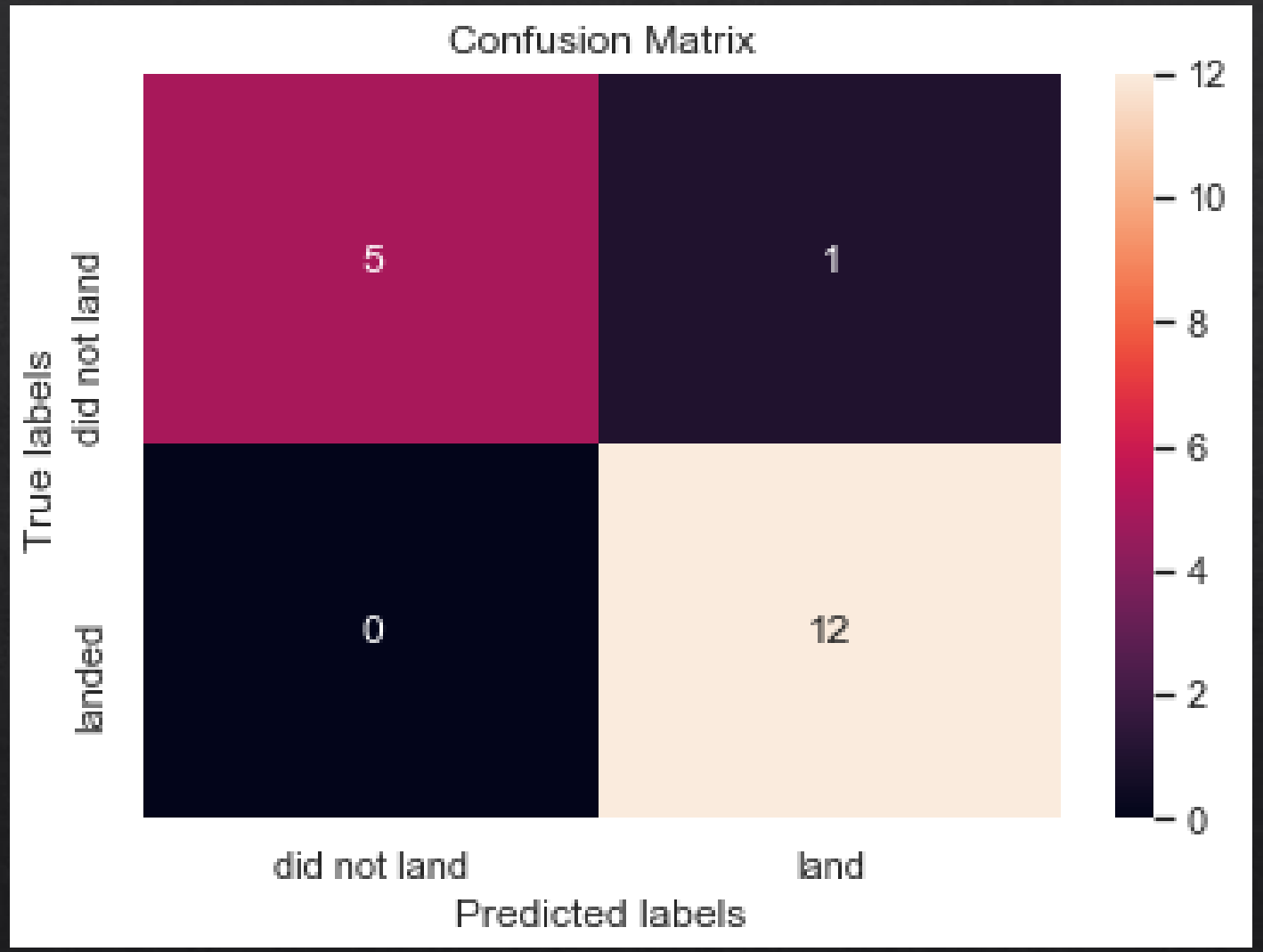


Plotting the Accuracy Score and Best Score for each classification algorithm produces the following result:

- The Decision Tree model has the highest classification accuracy
  - The Accuracy Score is 94.44%
  - The Best Score is 90.36%

Confusion Matrix

- As shown previously, best performing classification model is the Decision Tree model, with an accuracy of 94.44%.
- This is explained by the confusion matrix, which shows only 1 out of 18 total results classified incorrectly (a false positive, shown in the top-right corner).
- The other 17 results are correctly classified (5 did not land, 12 did land).



Conclusions

Key Findings:

- **Impact of Flight Experience on Success Rates:**
  - As the number of flights increases, the success rate at each launch site generally improves.
  - **Early Challenges:** From **2010 to 2013**, all landings were unsuccessful (0% success rate).
  - **Consistent Improvement:** After **2013**, success rates steadily improved, despite minor setbacks in **2018** and **2020**.
  - **High Reliability:** Since **2016**, the success rate has consistently remained above **50%**.
- **High Success Orbit Types:**
  - **100% Success Orbits:**
    - **ES-L1 (Earth-Sun First Lagrangian Point)**
    - **GEO (Geostationary Orbit)**
    - **HEO (High Earth Orbit)**
    - **SSO (Sun-synchronous Orbit)** - more impressive with **5 successful flights**.
  - **Heavy Payload Success:**
    - **PO (Polar Orbit)**, **ISS (International Space Station)**, and **LEO (Low Earth Orbit)** orbits tend to support heavier payloads with higher success rates.
  - **VLEO (Very Low Earth Orbit)** is also associated with heavier payloads, aligning with its design purpose.
- **Top-Performing Launch Site:**
  - **KSC LC-39A** had the **most successful launches**, accounting for **41.7%** of total successful launches, and the **highest individual site success rate** at **76.9%**.
- **Payload Size and Success Rates:**
  - The success rate for **massive payloads (over 4000 kg)** is generally lower than that for lighter payloads.
- **Best Classification Model:**
  - The **Decision Tree Model** emerged as the best-performing classification model, achieving an accuracy of **94.44%**.

Appendix

Auxiliary Functions for Data Collection – Web Scraping

```
def date_time(table_cells):
    """
    This function returns the data and time from the HTML  table cell
    Input: the  element of a table data cell extracts extra row
    """
    return [data_time.strip() for data_time in list(table_cells.strings)][0:2]

def booster_version(table_cells):
    """
    This function returns the booster version from the HTML  table cell
    Input: the  element of a table data cell extracts extra row
    """
    out=''.join([booster_version for i,booster_version in enumerate( table_cells.strings) if i%2==0][0:-1])
    return out

def landing_status(table_cells):
    """
    This function returns the landing status from the HTML table cell
    Input: the  element of a table data cell extracts extra row
    """
    out=[i for i in table_cells.strings][0]
    return out


def get_mass(table_cells):
    mass=unicodedata.normalize("NFKD", table_cells.text).strip()
    if mass:
        mass.find("kg")
        new_mass=mass[0:mass.find("kg")+2]
    else:
        new_mass=0
    return new_mass


def extract_column_from_header(row):
    """
    This function returns the landing status from the HTML table cell
    Input: the  element of a table data cell extracts extra row
    """
    if (row.br):
        row.br.extract()
    if row.a:
        row.a.extract()
    if row.sup:
        row.sup.extract()

    colunm_name = ' '.join(row.contents)

    # Filter the digit and empty names
    if not(colunm_name.strip().isdigit()):
        colunm_name = colunm_name.strip()
        return colunm_name
```

Thank you!