IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Alex Mraz

6/7/2025

Outline

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

Executive Summary

- Summary of Methodologies
  - Data Collection
  - Data Wrangling & Cleaning
  - Exploratory Data Analysis (EDA) with Visualization
  - Exploratory Data Analysis using SQL
  - Creating Interactive Maps with Folium
  - Developing Dashboards with Plotly Dash
  - Predictive Analytics: Solving Classification Problems

Executive Summary

- Summary of All Results
  - Results from Exploratory Data Analysis
  - Interactive Analysis Demo (Screenshots)
  - Outcomes of Predictive Modeling

Introduction

- Project Background and Context
  - Commercial spaceflight is **booming**
  - SpaceX: Leading company in commercial space travel
  - Falcon 9 launch cost: **~$62 million** vs. competitors at $165+ million
  - SpaceX cuts launch costs via **reusable** rockets
  - **Predicting** first stage landing success is **key** to estimating launch costs
  - Using **public data** and **machine learning models** to forecast reusability

Introduction

- Problems You Want to Find Answers
- How do factors like:
  - Payload mass
  - Launch site
  - Number of flights
  - Orbit type

  affect first stage landing success?
- Has the landing success rate improved over time?
- Which algorithm performs best for binary classification of landing success?

Introduction

- Project Objective
- Assess the viability of a new company, *SpaceY*, by analyzing SpaceX launch data
- Develop predictive models to forecast first stage landing success

- Key Questions
- What is the best way to estimate total launch costs?

  → By predicting first stage landing success
- Which launch sites yield the highest success rates?

Section 1

Methodology

Methodology

- Data collection methodology:
  - SpaceX REST API
  - Web scraping from Wikipedia
- Perform data wrangling
  - Filter relevant data
  - Handle missing values
  - Apply one-hot encoding for binary classification
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - How to build, tune, evaluate classification models

Data Collection

Data Collection Overview

- Combined data from **SpaceX REST API** and **Wikipedia web scraping** to ensure complete launch information
- Used both sources to enrich data for detailed analysis

API Data Included:

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

Wikipedia Data Included:

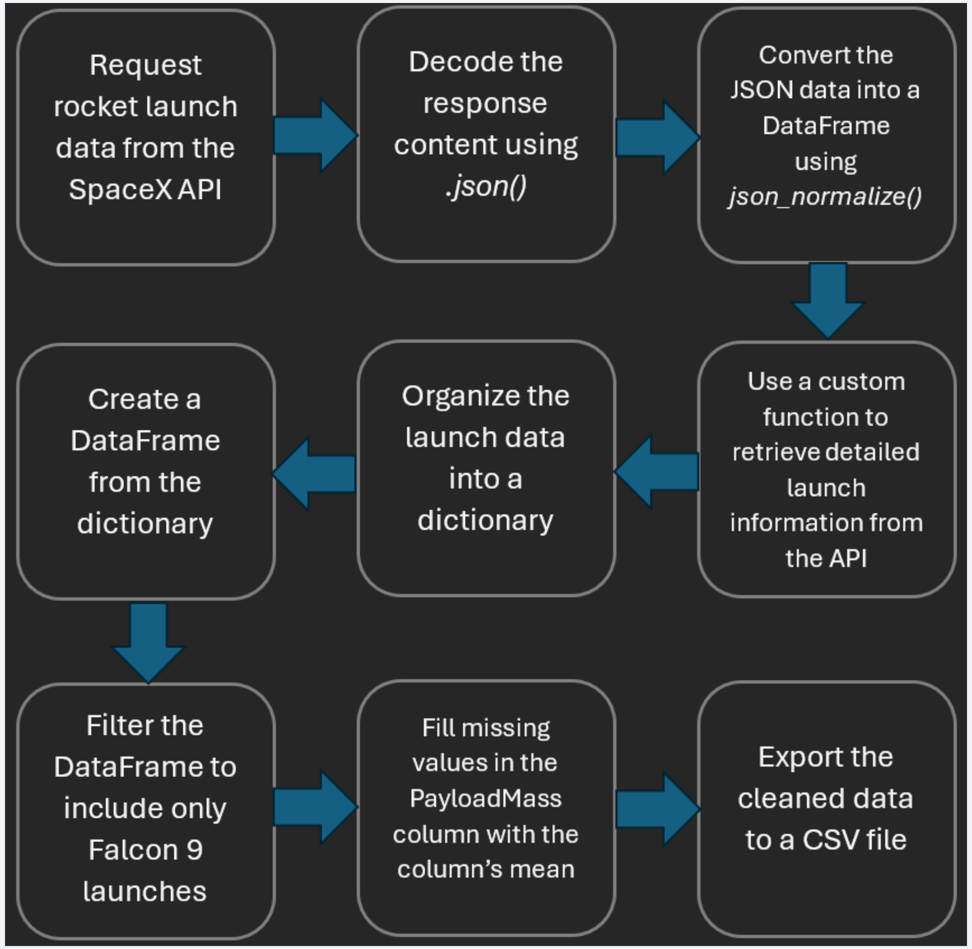
- Flight No., Launch Site, Payload, Payload Mass, Orbit, Customer, Launch Outcome, Booster Version, Booster Landing, Date, Time

Data Collection – SpaceX API

- Data collection with SpaceX REST calls

- Refer to GitHub URL for the completed SpaceX API call notebook:

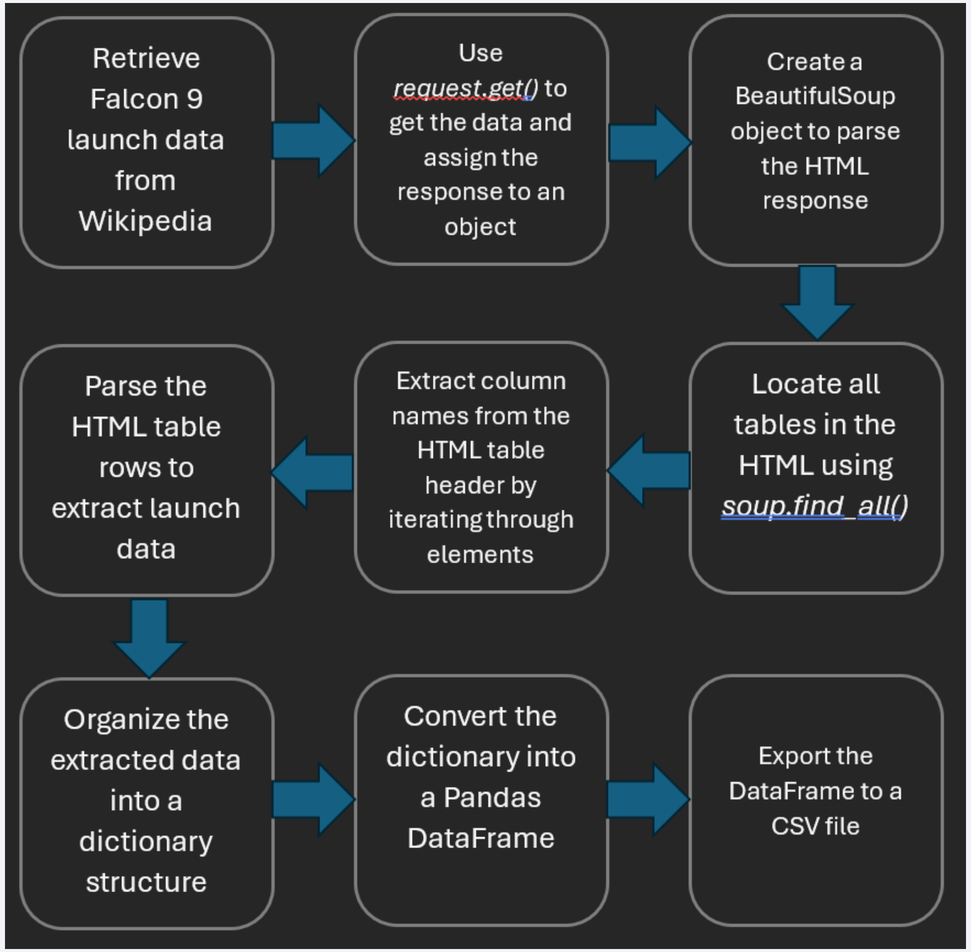
- https://github.com/amraz39/spaceY/blob/main/jupyter-labs-spacex-data-collection-api.ipynb



Data Collection - Scraping

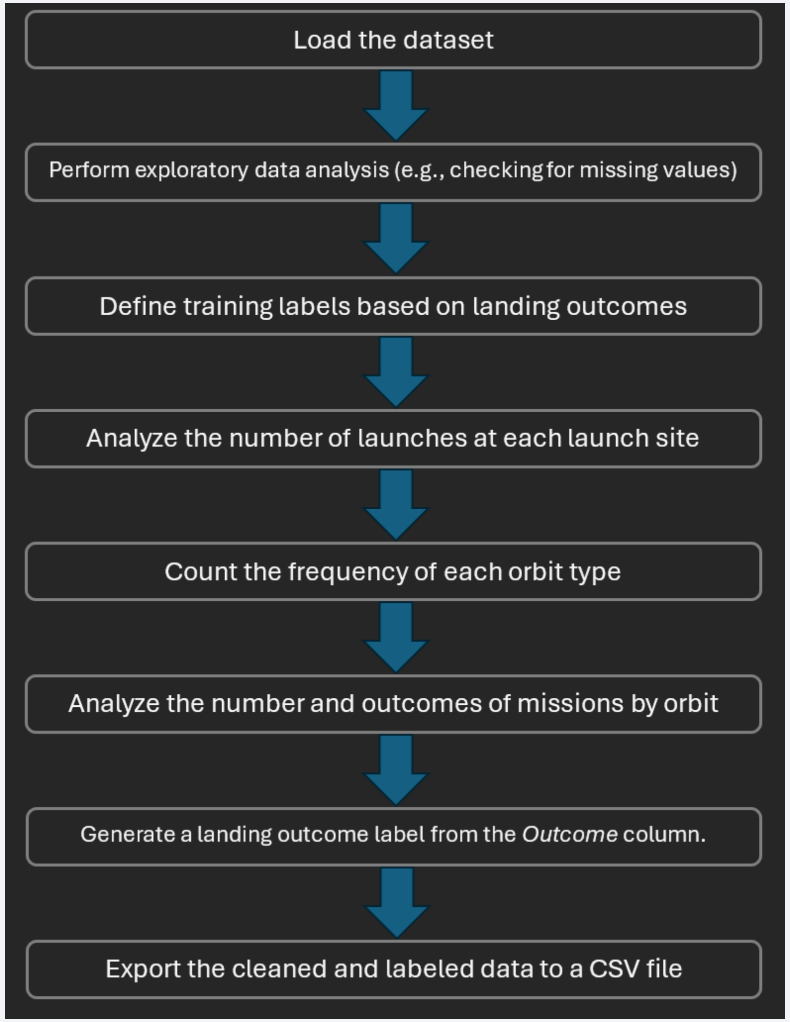
- Present your web scraping process using key phrases and flowcharts

- Add the GitHub URL of the completed web scraping notebook, as an external reference and peer-review purpose



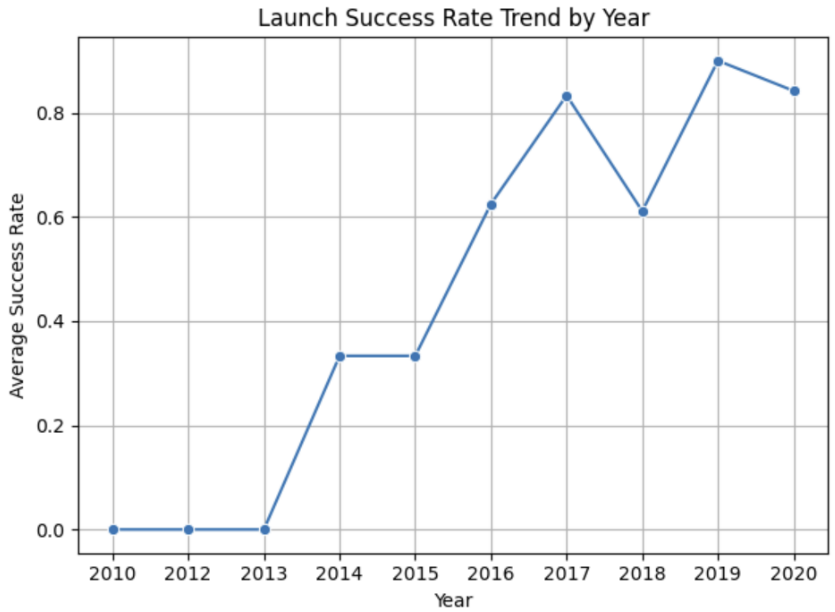
Data Wrangling

- The dataset includes various cases where the booster landing was (un)successful
- Landing attempts may fail due to accidents
- Landing outcome examples:
  - True Ocean: Successful landing in the ocean
  - False Ocean: Unsuccessful ocean landing
  - True RTLS: Successful ground pad landing
  - False RTLS: Unsuccessful ground pad landing
  - True ASDS: Successful drone ship landing
  - False ASDS: Unsuccessful drone ship landing
- These outcomes are converted into training labels:
  - 1 = Successful landing
  - 0 = Unsuccessful landing
- GitHub URL: https://github.com/amraz39/spaceY/blob/main/labs-jupyter-spacex-Data%20wrangling.ipynb



EDA with Data Visualization

- To explore the dataset, scatter plots, bar plots, and line plots were used to visualize relationships / trends between pairs of features:
  - Payload Mass vs. Flight Number
  - Launch Site vs. Flight Number
  - Launch Site vs. Payload Mass
  - Orbit vs. Flight Number
  - Orbit vs. Success Rate
  - Launch Success Yearly Trend



- Scatter plots: identify trends, correlations, and potential outliers between numerical features
- Bar plots: compare categorical features (like launch sites or orbits) and their frequencies or distributions across missions
- GitHub URL: https://github.com/amraz39/spaceY/blob/main/edadataviz.ipynb

EDA with SQL

- Executed SQL queries for data retrieval and analysis
- Names of the unique launch sites in the space mission
- Top 5 launch sites whose name begin with the string 'CCA'
- Total payload mass carried by boosters launched by NASA (CRS)
- Average payload mass carried by booster version F9 v1.1
- Date when the first successful landing outcome in ground pad was achieved
- Names of the boosters which have success in drone ship and have payload mass between 4000 and 6000 kg
- Total number of successful and failure mission outcomes
- Names of all booster versions which have carried the maximum payload mass
- Failed landing outcomes in drone ship, their booster versions, and launch site names for in year 2015
- Rank of the count of landing outcomes (such as Failure (drone ship) or Success (ground pad)) between the date 2010-06-04 and 2017-03-20
- GitHub: https://github.com/amraz39/spaceY/blob/main/jupyter-labs-eda-sql-coursera_sqllite.ipynb

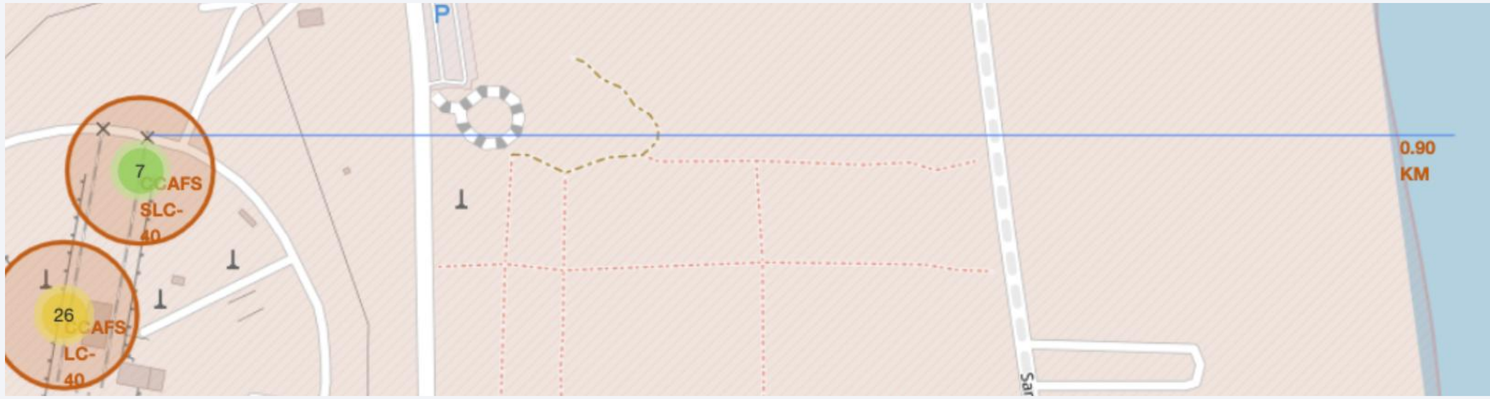
Build an Interactive Map with Folium

- **Markers for All Launch Sites**
  - Plotted NASA Johnson Space Center using latitude and longitude as the starting point
  - Added circle markers with popup and text labels for all launch sites
  - Displayed geographical positions relative to the Equator and nearby coastlines

- **Launch Outcome Visualization with Colored Markers**
  - Green for successful launches (Class=1)
  - Red for failed launches (Class=0)
  - Applied Marker Clustering to highlight success rate patterns by site

Build an Interactive Map with Folium

- **Distance to Nearby Infrastructure**
- Colored lines from Launch Site KSC LC-39A to nearby features:
  - Railway
  - Highway
  - Coastline
  - Closest city



- GitHub: https://github.com/amraz39/spaceY/blob/main/lab_jupyter_launch_site_location.ipynb

Build a Dashboard with Plotly Dash

- Launch Site Selection:
  - Implemented a dropdown menu to allow users to select a specific launch site
- Launch Success Pie Chart:
  - Displays overall successful launches across all sites
  - If a specific site is selected, shows Success vs. Failure distribution for that site
- Payload Mass Range Slider:
  - Enables users to filter launches by selecting a specific payload mass range
- Payload vs. Success Scatter Plot:
  - Visualizes the correlation between Payload Mass and Launch Success
  - Differentiates between Booster Version categories for deeper insight.

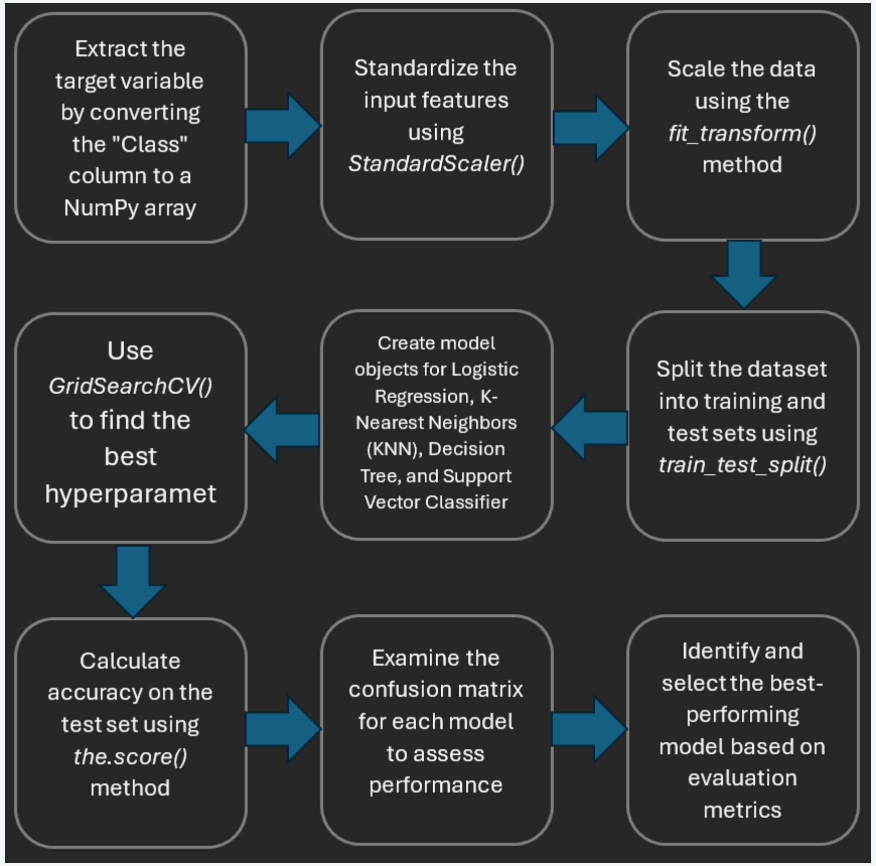
GitHub URL: https://github.com/amraz39/spaceY/blob/main/spacex-dash-app.py

Predictive Analysis (Classification)

- Building and Evaluating the Best Classification Model

- We will adopt a scientific methodology to build, train, evaluate, and select the most effective models for making accurate predictions.

- GitHub: https://github.com/amraz39/spaceY/blob/main/SpaceX_Machine%20Learning%20Prediction_Part_5.ipynb



Predictive Analysis (Classification)

- Model Development Summary
  - Prepared and cleaned the dataset, handling missing values and encoding categorical features
  - Standardized feature data using StandardScaler() to ensure consistent input scales
  - Split the data into training and test sets using train_test_split()
  - Built classification models used GridSearchCV() to tune hyperparameters and optimize each model
  - Evaluated model performance using accuracy, confusion matrix, and other metrics (e.g., precision, recall, F1)
  - Compared results across all models to identify the best performer
  - Selected the model(s) with the highest accuracy and balanced performance as the final classifier

Results

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

Section 2

Insights drawn from EDA

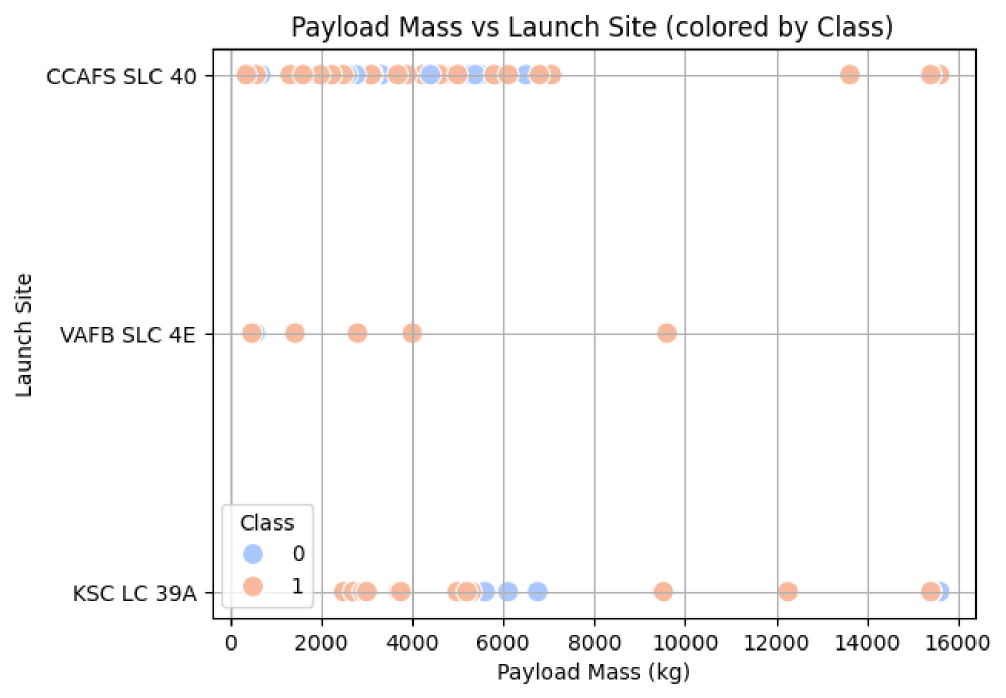
Flight Number vs. Launch Site

- Flight Number vs. Launch Site

- Early launches tended to fail, while recent launches have shown consistent success
- CCAFS SLC 40 accounts for approximately half of all launches
- VAFB SLC 4E and KSC LC 39A demonstrate higher success rates compared to other sites
- There is a **clear trend** suggesting that newer launches have a higher likelihood of success, likely due to improved technology, processes, and experience

Payload vs. Launch Site

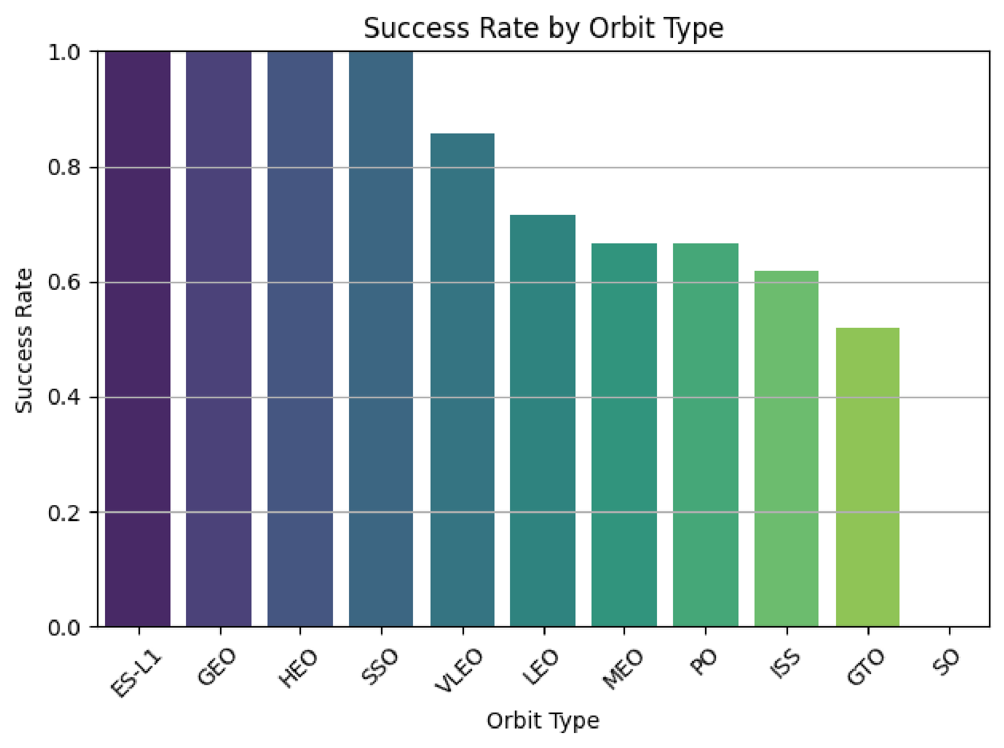
- Payload vs. Launch Site



- Across all launch sites, **higher payload mass** is generally associated with **higher success rates**
- Most launches exceeding 7,000 kg in payload mass were **successful**
- KSC LC 39A stands out with **high success rate** even for payloads under 5,500 kg, indicating strong reliability across different mission profiles
- VAFB SLC 4E has no rockets launched for heavy payload mass (greater than 10,000 kg)

Success Rate vs. Orbit Type

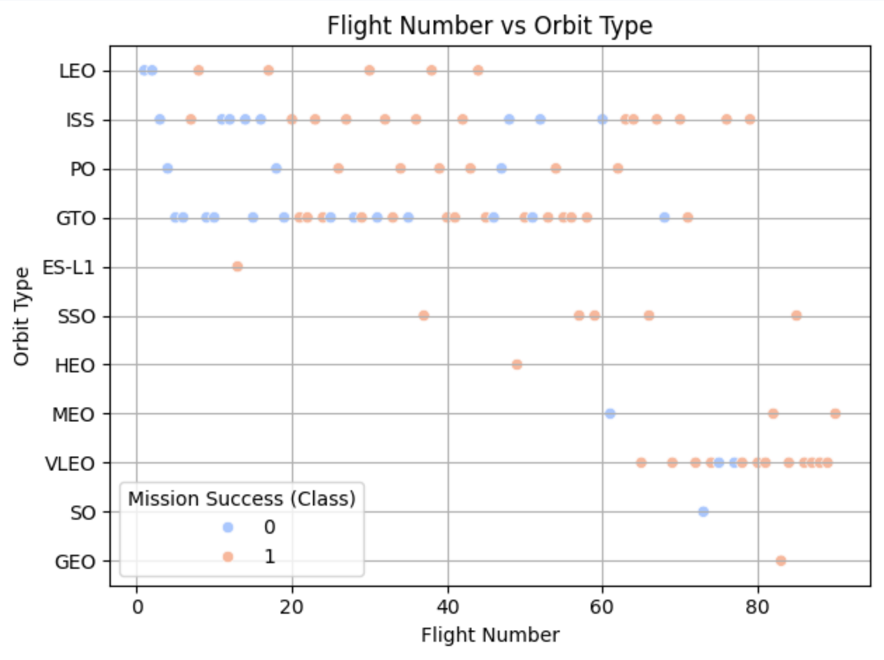
- Success Rate of Each Orbit Type



- Orbits with 100% success rate:
  - ES-L1, GEO, HEO, SSO
- Orbit with 0% success rate:
  - SO
- Orbits with moderate success rates (50%–85%):
  - VLEO, LEO, MEO, PO, ISS, GTO

Flight Number vs. Orbit Type

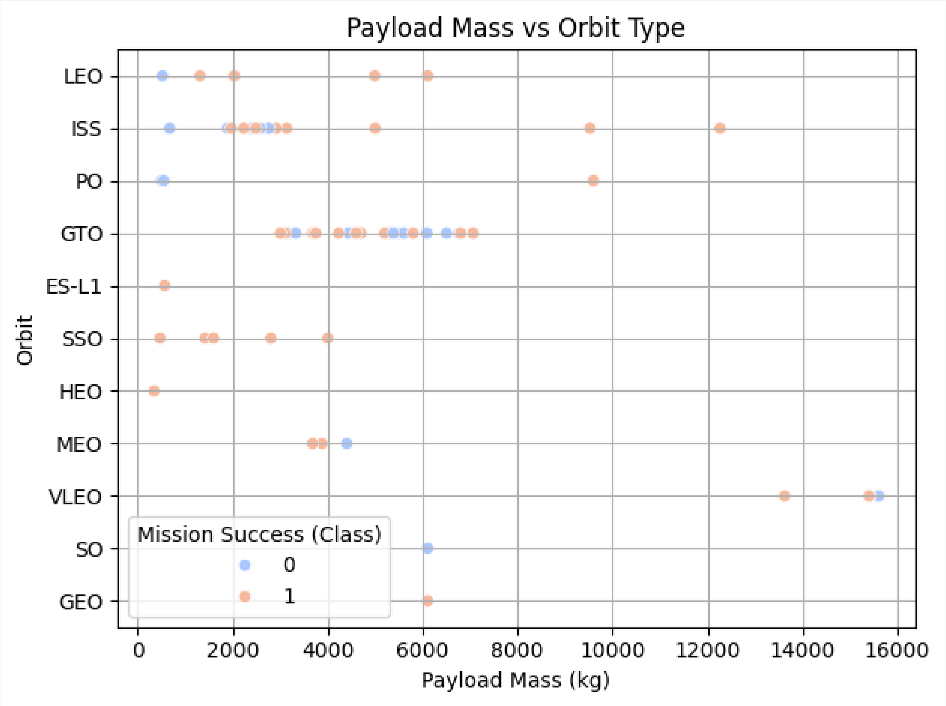
- Flight Number vs. Orbit Type



- In LEO (Low Earth Orbit), success rates tend to improve with the number of flights, suggesting learning or refinement over time
- In contrast, for GTO (Geostationary Transfer Orbit), there appears to be no clear correlation between the number of flights and success rates
- In SSO, even though there are limited number of flights, they have 100% success rate

Payload vs. Orbit Type

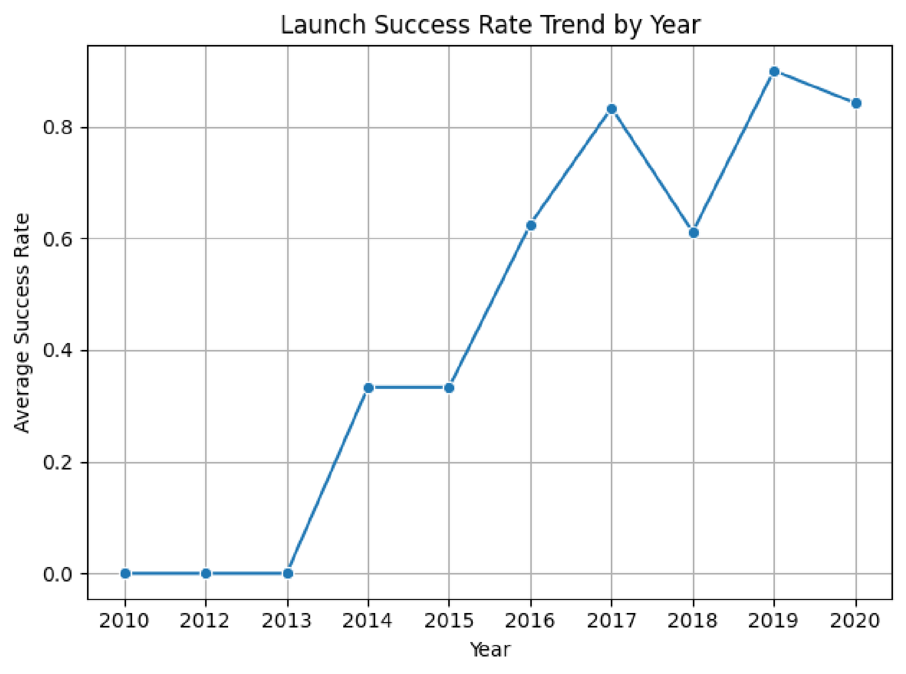
- Payload vs. Orbit Type



- Higher landing success rates are observed for heavy payloads in PO, LEO, and ISS orbits.
- For GTO missions, the trend is less clear — both successful and unsuccessful landings occur with heavy payloads, making it harder to establish a consistent pattern.
- For SSO missions, all landings were successful (all payloads were under 5,500 kg)

Launch Success Yearly Trend

- Yearly Average Success Rate



- Overall launch success rate steadily increased from 2013 to 2020
- This upward trend likely reflects:
  - Technological improvements in rocket design and systems
  - Operational experience gained from earlier launches
  - Enhanced quality control and pre-launch procedures
  - Data-driven refinements to navigation, landing, and recovery strategies

All Launch Site Names

```
%sql SELECT DISTINCT "Launch_Site" FROM SPACEXTABLE;
```

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

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

- The query is designed to filter for distinct launch sites from database
- CCAFS LC-40 and CCAFS SLC-40 may be duplicates due to naming inconsistency — should be verified
- Launch site codes alone do not reveal the geographical location of the site (require prior knowledge or a reference table to interpret)

Launch Site Names Begin with 'CCA'

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

MagicPython

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

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

- Filtered Launch Site Records
  - Displaying 5 records where launch site names start with 'CCA'
  - This filter helps isolate launches from Cape Canaveral-related facilities

Total Payload Mass

```
%sql SELECT SUM("PAYLOAD_MASS__KG_") AS Total_Payload_Mass FROM SPACEXTABLE WHERE "Customer" LIKE '%NASA (CRS)%';
```

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

Total_Payload_Mass
48213

- The query specifically filters for boosters launched under NASA's CRS (Commercial Resupply Services) program
- Filtering to generic CRS missions (all, including NASA’s CRS), the total payload mass amounted to 111,268 kg
- Just under half of the CRS missions were conducted for NASA

Average Payload Mass by F9 v1.1

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

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

Average_Payload_Mass
2928.4

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

First Successful Ground Landing Date

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

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

First_Successful_Ground_Landing
2015-12-22

- The query lists the **date of the first successful ground pad landing**

Successful Drone Ship Landing with Payload between 4000 and 6000

```
%sql SELECT DISTINCT Booster_Version FROM SPACEXTABLE WHERE "Landing_Outcome" = 'Success (drone ship)' AND PAYLOAD_MASS__KG_ > 4000 AND PAYLOAD_MASS__KG_ < 6000;
```

MagicPython

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

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

- The query retrieves **4 booster versions** that delivered payloads between 4,000 and 6,000 kg and successfully landed on a drone ship
- All four boosters are part of the **Falcon 9 family**

Total Number of Successful and Failure Mission Outcomes

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

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

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

- Query returns the summary table showing:
  - Unique mission outcome
  - Total number of launches that had that outcome
- **Most of the launches** were successful.

Boosters Carried Maximum Payload

```
%%sql

SELECT DISTINCT Booster_Version
FROM SPACEXTABLE
WHERE PAYLOAD_MASS__KG_ = (
    SELECT MAX(PAYLOAD_MASS__KG_)
    FROM SPACEXTABLE
)

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

→

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

- The query returns a **list of booster versions** that carried the maximum recorded payload.
- All of the returned boosters are from the **Falcon 9 family**, specifically the **Block 5 (B5)** variant

2015 Launch Records

```
%%sql

SELECT substr("DATE", 6, 2) AS "Month" , "landing_outcome", "BOOSTER_VERSION", "LAUNCH_SITE"
    FROM SPACEXTABLE WHERE landing_outcome = 'Failure (drone ship)'
    and substr("DATE", 0, 5) = '2015';
```

✓ 0.0s

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

Month	Landing_Outcome	Booster_Version	Launch_Site
01	Failure (drone ship)	F9 v1.1 B1012	CCAFS LC-40
04	Failure (drone ship)	F9 v1.1 B1015	CCAFS LC-40

- In 2015, there were **two failed landings** on **drone ships,** one in January and the other in April.
- Both missions used **Falcon 9** boosters.
- Both launches took place from CCAFS LC-40 launch site.

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

```
%%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;
```

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

→

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

- The query returns **eight unique landing outcomes** within **the selected date range**
- The **most frequent** outcome is "No attempt", **followed by** "Success (drone ship)"
- The least frequent outcome is "Precluded (drone ship)"

Section 3

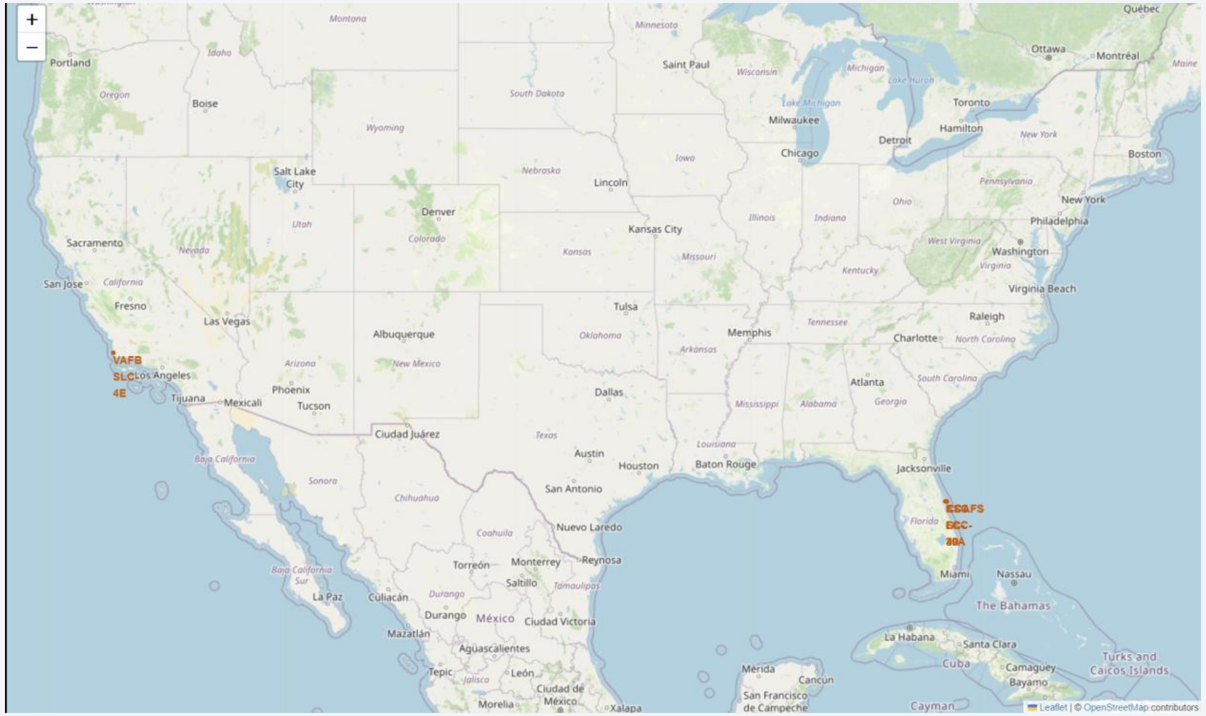
Launch Sites Proximities Analysis

Mapping Launch Site Locations Across the USA

- **Most launch sites** are located near the **Equator**, where the Earth's surface moves fastest, about 1,670 km/h due to Earth's rotation.

- Launching from the Equator gives rockets a **speed boost** from inertia, helping spacecraft achieve and maintain orbit more efficiently.

- **All launch sites** are positioned **close to the coast**, allowing rockets to be launched over the ocean, which minimizes risks from falling debris or explosions near populated areas.

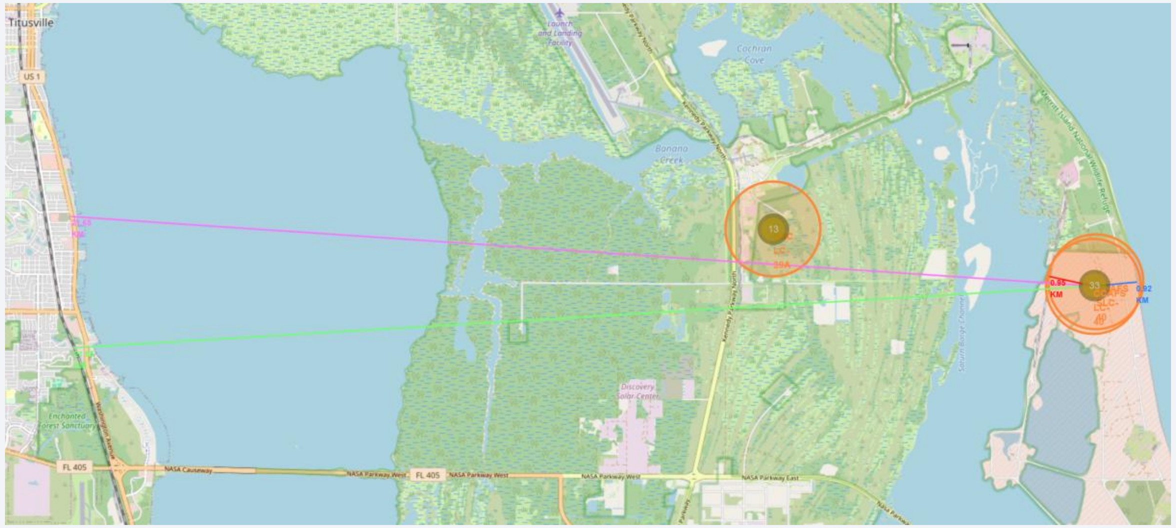


Color-Coded Launch Success Rates on the Map

- **Color-coded icon markers** help quickly identify launch site performance:
  - 🟢 Green Marker = Successful Launch
  - 🔴 Red Marker = Failed Launch
- Each launch site is **identifiable** by its **interactive popup marker** on the map.
- It is evident that KSC LC-39A stands out with a very **high success rate**.

Distance from Launch Site SLC-40 to Nearby Landmarks

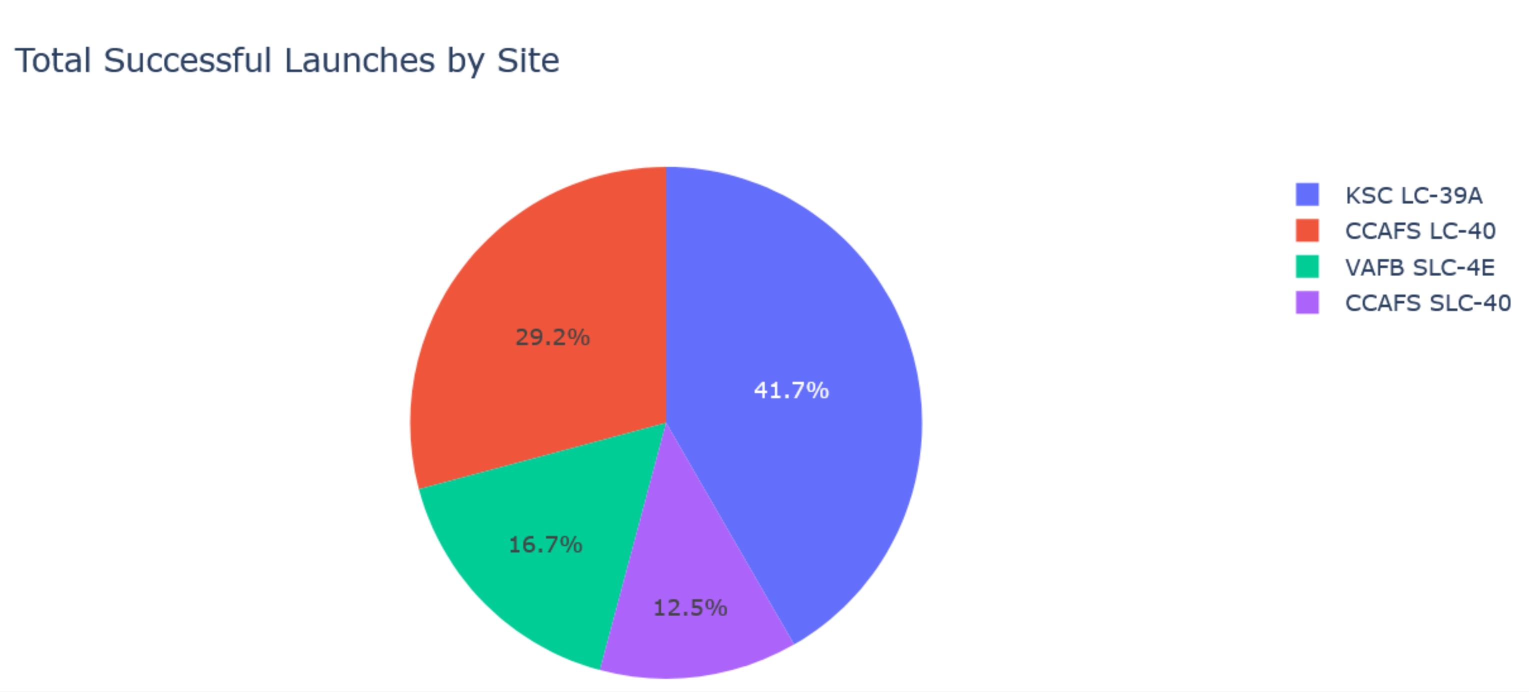
- Visual Analysis of SLC-40 Proximities:
- The site is relatively close to key infrastructure:
  - Railway: approximately 21.6 km away
  - Highway: approximately 21.65 km away
  - Coastline: approximately 0.92 km away
- It is near the city of Titusville, about 23 km from the launch site
- Considering that a failed rocket traveling at high speed can cover 15–20 km within seconds, proximity to populated areas like Titusville may pose potential safety risk
- Launch site LC-39A is about 16 km from city of Titusville which may also pose potential safety risk



Section 4

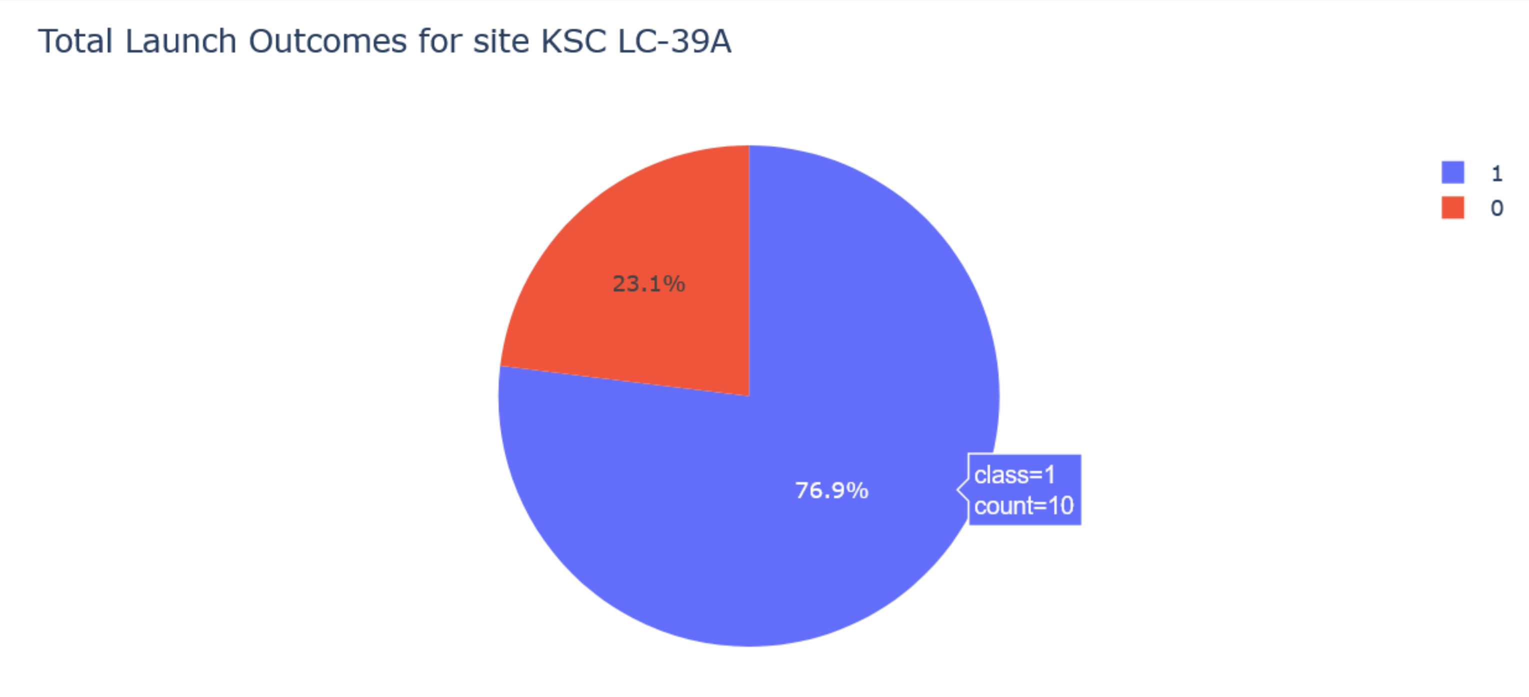
Build a Dashboard with Plotly Dash

Total Successful Launches for All Sites



- The chart clearly indicates that KSC LC-39A has the highest number of successful launches among all launch site

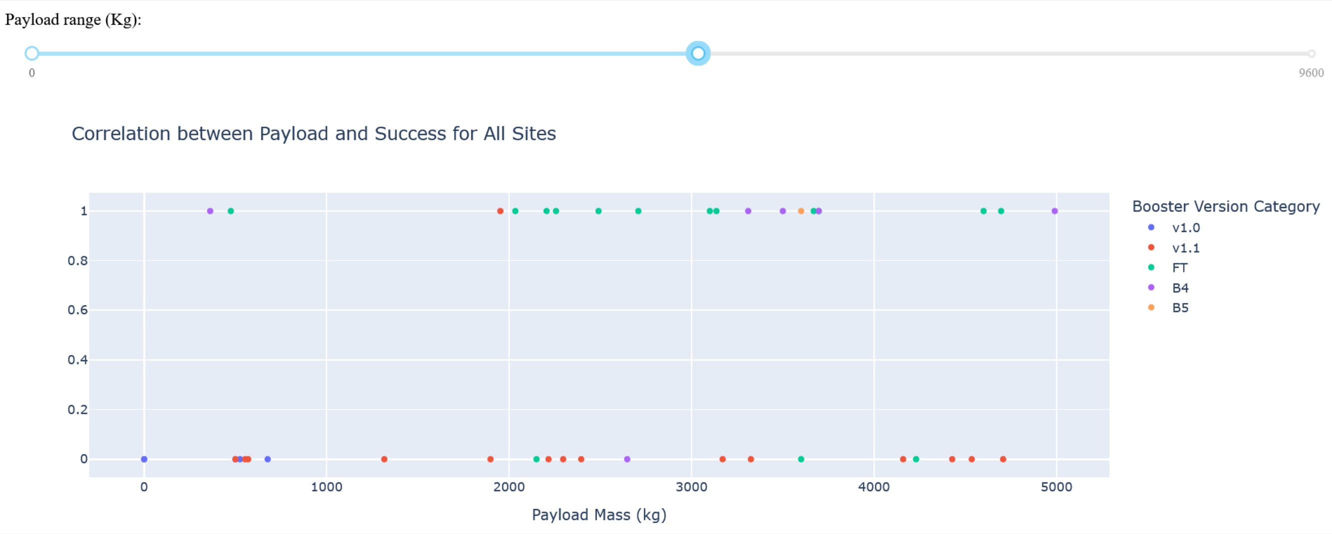
Launch Site with Highest Launch Success Ratio



- KSC LC-39A has the highest launch success rate at 76.9%, with 10 successful landings and only 3 failures

Launch Outcomes by Payload Mass and Site

- Payloads of 2,000–5,500 kg show the highest success rates
- Payloads of 0–2,000 km and 5,500–9,000 show high failure rate



Section 5

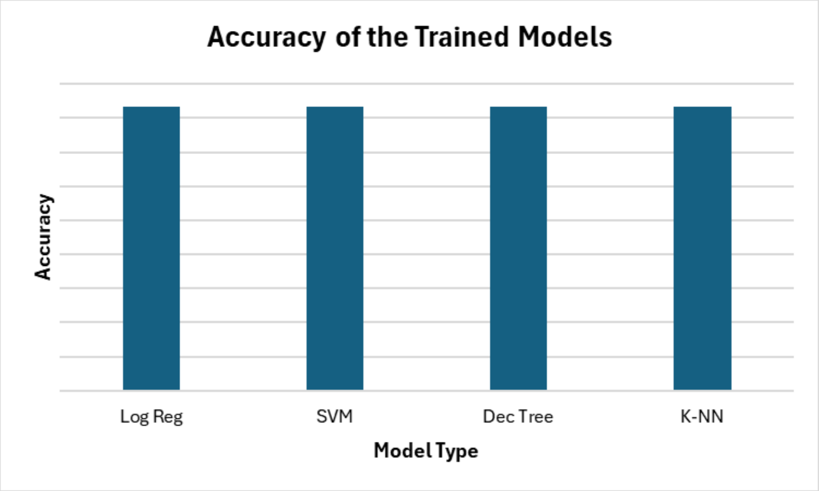
Predictive Analysis (Classification)

Classification Accuracy

- Training Accuracy:
  - Ranged from 84.6% to 89%
  - Decision Tree achieved the highest training accuracy
- Test Accuracy:
  - Identical across all models: 83.33%
  - All models correctly classified all 12 positives
  - Each misclassified 3 negatives
- Interpretation:
  - Slightly higher training accuracy of Decision Tree does not indicate overfitting
  - Uniform test results imply a performance ceiling, likely due to limitations in dataset test (only 18 samples) or features
- Best Performing Model:
  - Decision Tree

TRAIN DATASET	Log Reg	SVM	Dec Tree	K-NN
Accuracy	0.846	0.848	0.888	0.848
F1 Score	0.914	0.923	0.902	0.904

TEST DATASET	Log Reg	SVM	Dec Tree	K-NN
Accuracy	0.833	0.833	0.833	0.833
F1 Score	0.888	0.888	0.888	0.888

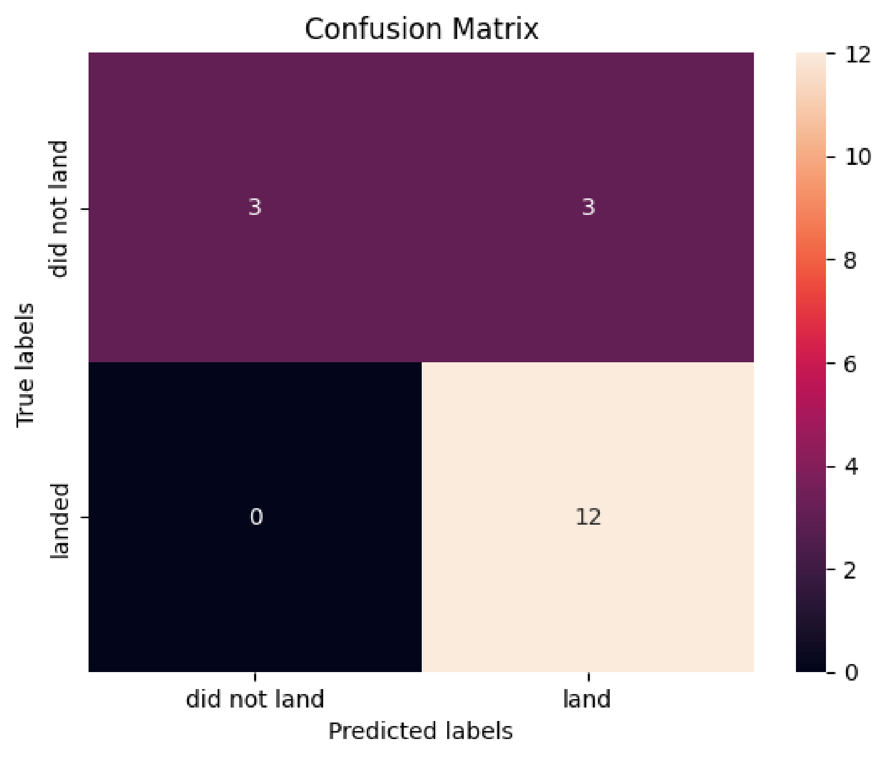


TEST DATASET	Confusion Matrix
Log Reg	[[3 3] [0 12]]
SVM	[[3 3] [0 12]]
Dec Tree	[[3 3] [0 12]]
K-NN	[[3 3] [0 12]]

Confusion Matrix

Confusion Matrix Insights – Logistic Regression

- The model **effectively distinguishes** between the classes
- The model performs well on detecting positives (no false negatives)
- However, it tends to misclassify some negatives as positives, resulting in **false positives**.
- This seems to be main source of error – false alarms may be costly.



Conclusions

- Most **launch sites are near the Equator**, benefiting from increased rotational speed, and are all located **close to coastlines** for safety
- **Launch success rates have steadily improved** year by year
- **Lower payload mass** is generally associated with **higher launch success**
- **KSC LC-39A** stands out with the **highest overall success rate** among all launch sites.
- **Orbits ES-L1, GEO, HEO, and SSO** show a **100% success rate**, indicating strong reliability in those mission types.
- **Decision Tree** performed best among all models for selected dataset

Appendix

- Snippet of the Plotly code

```python
# TASK 2: Callback for pie chart
@app.callback(Output('success-pie-chart', 'figure'),
              Input('site-dropdown', 'value'))
def update_pie_chart(selected_site):
    if selected_site == 'ALL':
        # Total success launches by site
        fig = px.pie(spacex_df[spacex_df['class'] == 1],
                     names='Launch Site',
                     title='Total Successful Launches by Site')
    else:
        # Success vs failure for selected site
        filtered_df = spacex_df[spacex_df['Launch Site'] == selected_site]
        counts = filtered_df['class'].value_counts().reset_index()
        counts.columns = ['class', 'count']
        fig = px.pie(counts, names='class', values='count',
                     title=f'Total Launch Outcomes for site {selected_site}')
    return fig
```

Appendix

- Snippet of the Plotly code

```python
# TASK 4: Callback for scatter plot
@app.callback(Output('success-payload-scatter-chart', 'figure'),
              [Input('site-dropdown', 'value'),
               Input('payload-slider', 'value')])
def update_scatter_plot(selected_site, payload_range):
    low, high = payload_range
    df_filtered = spacex_df[(spacex_df['Payload Mass (kg)'] >= low) &
                            (spacex_df['Payload Mass (kg)'] <= high)]
    if selected_site == 'ALL':
        fig = px.scatter(df_filtered, x='Payload Mass (kg)', y='class',
                         color='Booster Version Category',
                         title='Correlation between Payload and Success for All Sites')
    else:
        df_site = df_filtered[df_filtered['Launch Site'] == selected_site]
        fig = px.scatter(df_site, x='Payload Mass (kg)', y='class',
                         color='Booster Version Category',
                         title=f'Correlation between Payload and Success for site {selected_site}')
    return fig
```

Appendix

- Snippet of the Folium code

```
# Create a marker at the highway location
distance_marker_highway = folium.Marker(
    [highway_lat, highway_lon],
    icon=folium.DivIcon(
        icon_size=(25, 25),
        icon_anchor=(0, 0),
        html='<div style="font-size: 12; color:purple;"><b>%s</b></div>' % "{:10.2f} KM".format(distance_rail),   ⇥ Tab to Jump
    )
)

# Add to map
site_map.add_child(distance_marker_coast)
site_map.add_child(distance_marker_road)
site_map.add_child(distance_marker_rail)
site_map.add_child(distance_marker_highway)

# Define coordinates
launch_site_coord = [launch_site_lat, launch_site_lon]    # CCAFS SLC-40
coastline_coord = [coastline_lat, coastline_lon]          # Closest coastline point
road_coord = [road_lat, road_lon]
rail_coord = [rail_lat, rail_lon]
highway_coord = [highway_lat, highway_lon]

# Create PolyLine objects with proper colors
line_coast = folium.PolyLine(locations=[launch_site_coord, coastline_coord], weight=2, color='blue')
line_road = folium.PolyLine(locations=[launch_site_coord, road_coord], weight=2, color='red')
line_rail = folium.PolyLine(locations=[launch_site_coord, rail_coord], weight=2, color='green')
line_highway = folium.PolyLine(locations=[launch_site_coord, highway_coord], weight=2, color='purple')

# Add lines to the map
site_map.add_child(line_coast)
site_map.add_child(line_road)
site_map.add_child(line_rail)
site_map.add_child(line_highway)
```

Thank you!