IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Jose Vitor Ferrari Costa
21/07/2025

Outline

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

Executive Summary

Methodology

- Data Collection:
  Obtained from public SpaceX records and organized into CSV files for analysis.
- Data Preparation:
  Cleaned data by handling missing values, standardizing categories, and ensuring proper formats.
- Exploratory Data Analysis (EDA):
  Used visualizations and SQL queries to identify trends like launch success rates by site and payload.
- Interactive Visualizations:
  Created maps and charts for better understanding of launch sites and outcomes.
- Predictive Modeling:
  Built and tuned classification models (Logistic Regression, SVM, Decision Tree, KNN) to predict launch success.
- Model Evaluation:
  Assessed models with accuracy scores and confusion matrices; Decision Tree showed highest performance.
- Conclusion:
  The Decision Tree model is most effective for predicting SpaceX launch success, supporting better planning and decision-making.

Results

- Built four models: Logistic Regression, SVM, Decision Tree, and KNN.
- Tuned model settings to get the best results.
- Tested models and got these accuracies:
  - Logistic Regression: 83%
  - SVM: 83%
  - Decision Tree: **94%** (best)
  - KNN: 83%
- Decision Tree was the most accurate.
- It made very few mistakes:
  - Correctly predicted 12 successful launches (true positives)
  - Correctly predicted 5 failures (true negatives)
  - Only 1 wrong positive prediction (false positive)
  - No missed successes (false negatives)
- Conclusion: Decision Tree is the best model for predicting SpaceX launch outcomes.

Introduction

Project Background and Context

This capstone project simulates the work of a data scientist at a private space launch company.

The objective is to apply a complete data science workflow to support strategic decisions related to successful mission outcomes and launch performance.

Problems to Explore

- What factors influence the success of a space launch?
- Can we predict whether a future launch will be successful?
- How do different launch sites and payload characteristics impact outcomes?

Section 1

Methodology

Methodology

Executive Summary

- Data Collection Methodology:
    - Data was obtained from SpaceX public records and processed into structured CSV format for analysis.
- Data Wrangling:
    - Cleaned missing values, renamed columns, standardized categories, and ensured correct data types.
- Exploratory Data Analysis (EDA):
    - Performed using visualizations (Matplotlib, Seaborn) and SQL queries to uncover trends and correlations (e.g., success rates by launch site and payload mass).
- Interactive Visual Analytics:
    - Created using Plotly Dash for real-time exploration and Folium for launch site mapping.
- Predictive Analysis:
    - Built and tested classification models (e.g., Logistic Regression, SVM) to predict launch success based on features like payload, orbit, and site.
- Model Development Process:
- Built models with scikit-learn
- Tuned hyperparameters using GridSearchCV
- Evaluated performance using accuracy, confusion matrix, and cross-validation

Data Collection

- **SpaceX REST API**

  Used the "api.spacexdata.com/v4/launches/past" endpoint to collect structured launch data.

  Each record included references (IDs) to related entities: rocket, payload, launchpad, and core.

  Additional endpoints were queried to resolve these IDs into descriptive information.

- **Web Scraping from Wikipedia**

  Launch records and details related to Falcon 9 missions were scraped using the BeautifulSoup library.

  HTML tables were parsed and converted to structured DataFrames.

  Provided supplementary details such as mission outcome and landing status.

- **Integration and Cleaning**

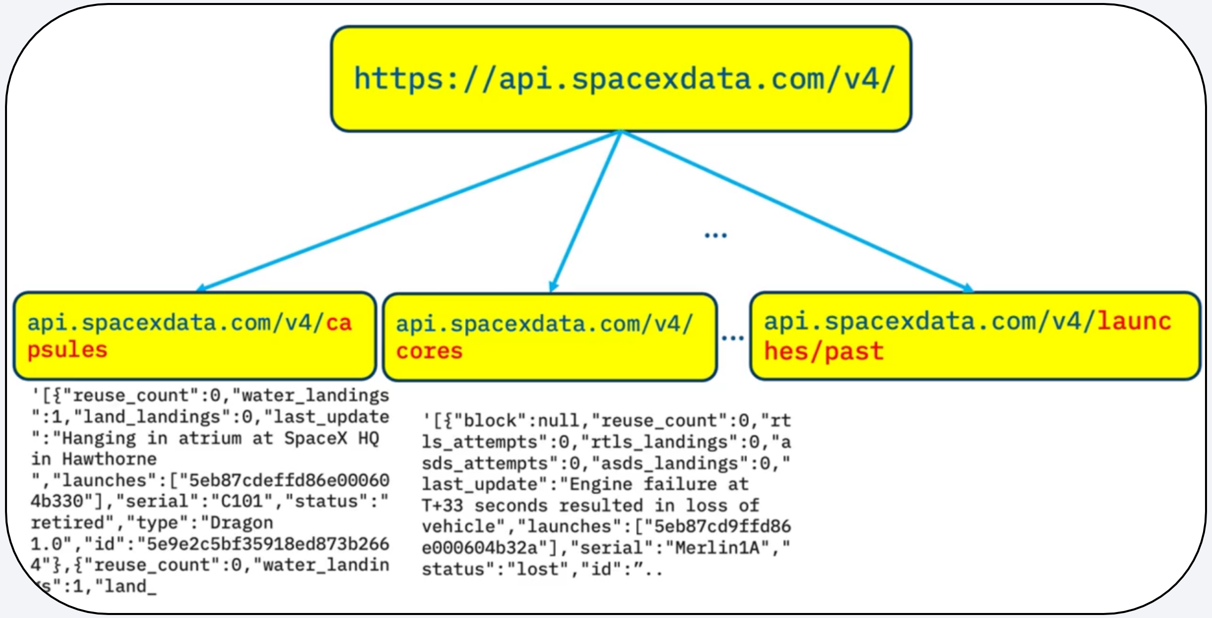
  All data sources were combined into an unified dataset.

  Data was filtered to include only Falcon 9 launches.

  Missing values were handled (e.g., imputation of `PayloadMass` using column mean).

Data Collection – SpaceX API

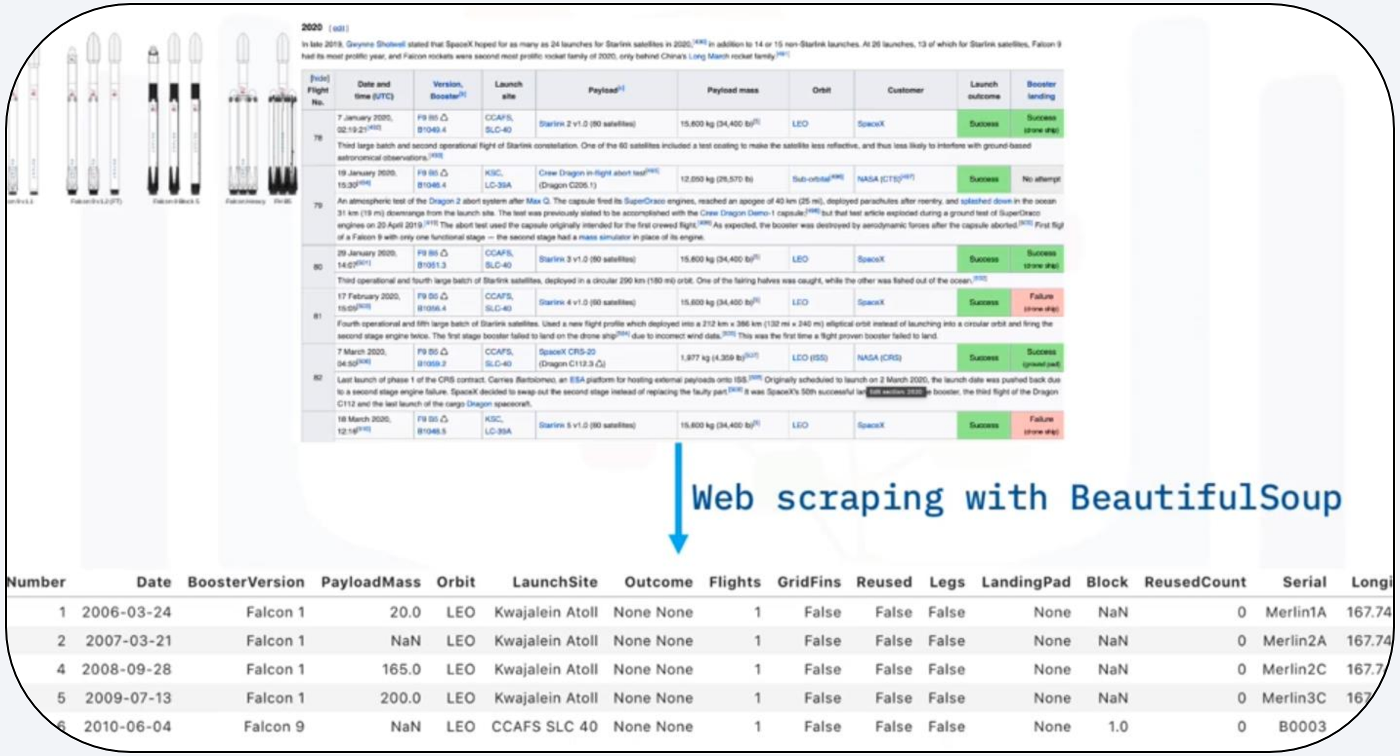
- Queried SpaceX REST API at /v4/launches/past
- Used Python requests library to perform GET request
- Retrieved data as JSON list of launch records
- Each record included rocket, payload, core, and launchpad IDs
- Converted JSON to tabular format using json_normalize()
- Created initial DataFrame of past launch events



https://github.com/josevitor22/Applied-Data-Science-Capstone/blob/main/1-%20jupyter-labs-spacex-data-collection-api.ipynb

Data Collection - Scraping

- Got web page from Wikipedia about Falcon 9 launches
- Used Python requests to download the page
- Used BeautifulSoup to find tables in the page
- Extracted launch data from HTML tables
- Turned tables into Pandas DataFrames for analysis

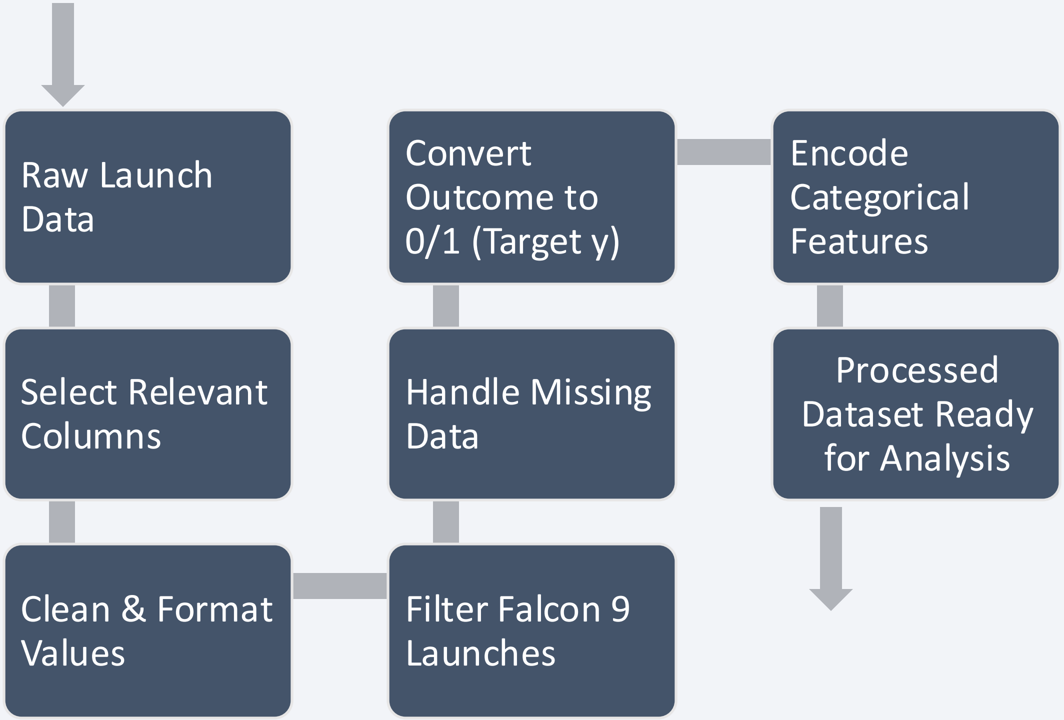


Number	Date	BoosterVersion	PayloadMass	Orbit	LaunchSite	Outcome	Flights	GridFins	Reused	Legs	LandingPad	Block	ReusedCount	Serial	Longi
1	2006-03-24	Falcon 1	20.0	LEO	Kwajalein Atoll	None None	1	False	False	False	None	NaN	0	Merlin1A	167.74
2	2007-03-21	Falcon 1	NaN	LEO	Kwajalein Atoll	None None	1	False	False	False	None	NaN	0	Merlin2A	167.74
4	2008-09-28	Falcon 1	165.0	LEO	Kwajalein Atoll	None None	1	False	False	False	None	NaN	0	Merlin2C	167.7
5	2009-07-13	Falcon 1	200.0	LEO	Kwajalein Atoll	None None	1	False	False	False	None	NaN	0	Merlin3C	167
6	2010-06-04	Falcon 9	NaN	LEO	CCAFS SLC 40	None None	1	False	False	False	None	1.0	0	B0003	

https://github.com/josevitor22/Applied-Data-Science-Capstone/blob/main/2-%20jupyter-labs-webscraping.ipynb

Data Wrangling

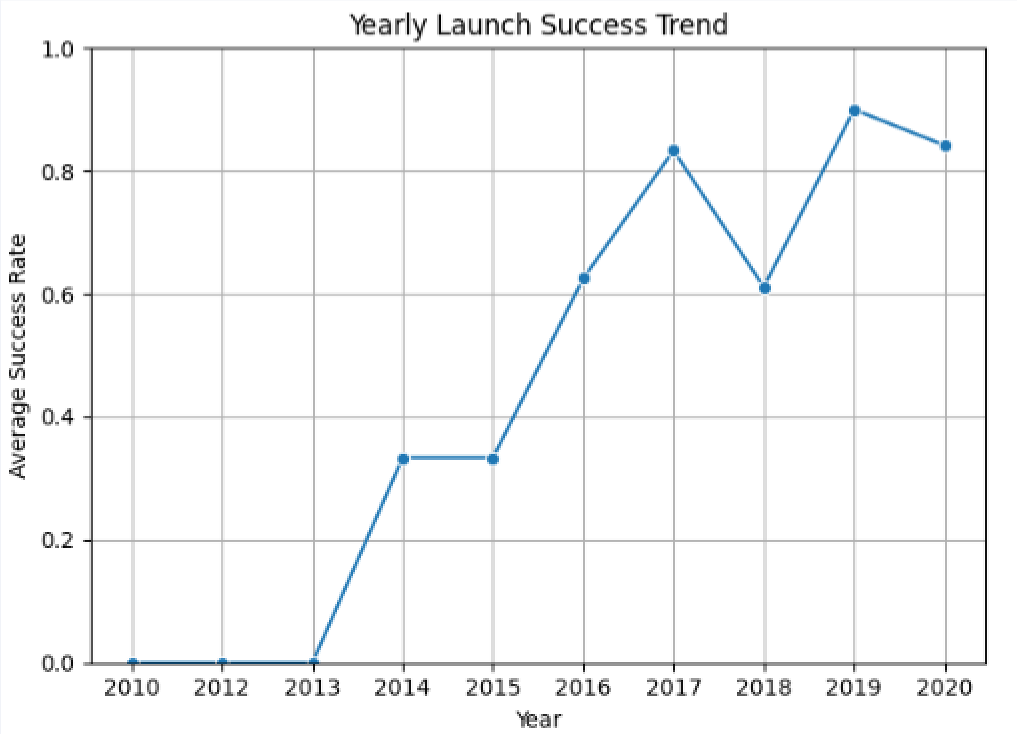
- Selected key variables
- Filtered dataset to include only Falcon 9 launches
- Handled missing values:
  - Replaced nulls in PayloadMass with the column mean
- Converted landing outcomes into binary classification:
  - 1 for successful landing
  - 0 for failed landing
- Extracted and added geographic data: Longitude and Latitude
- Encoded categorical variables using one-hot encoding for modeling
- Final result: a clean, structured dataset ready for analysis and modeling



https://github.com/josevitor22/Applied-Data-Science-Capstone/blob/main/3-%20labs-jupyter-spacex-Data%20wrangling.ipynb

EDA with Data Visualization

- Flight Number vs. Payload Mass
  To examine how launch experience and payload size relate to landing outcome.
- Flight Number vs. Launch Site
  To explore how launch frequency varies across different sites and its relationship to success.
- Payload Mass vs. Launch Site
  To assess payload distribution by launch site.
- Orbit Type vs. Success Rate
  To compare average landing success rates across different orbit types.
- Flight Number vs. Orbit Type
  To visualize how launch experience relates to success across orbit types.
- Payload Mass vs. Orbit Type
  To analyze how payload mass affects success for different orbit types.
- Yearly Launch Success Trend
  To observe trends in average launch success rate over time.



https://github.com/josevitor22/Applied-Data-Science-Capstone/blob/main/5-%20edadataviz.ipynb

EDA with SQL

- Retrieved unique launch site names.
- Selected 5 records where launch site starts with 'CCA'.
- Calculated total payload mass for boosters launched by NASA (CRS).
- Computed average payload mass for F9 v1.1 booster version.
- Identified first successful ground pad landing date.
- Listed booster names with drone ship success and payload between 4000–6000 kg.
- Counted total successful and failed mission outcomes.
- Used subquery to find booster versions with maximum payload mass.
- Filtered 2015 failure outcomes on drone ship with booster version, launch site, and month.
- Ranked landing outcome counts between 2010-06-04 and 2017-03-20 in descending order.

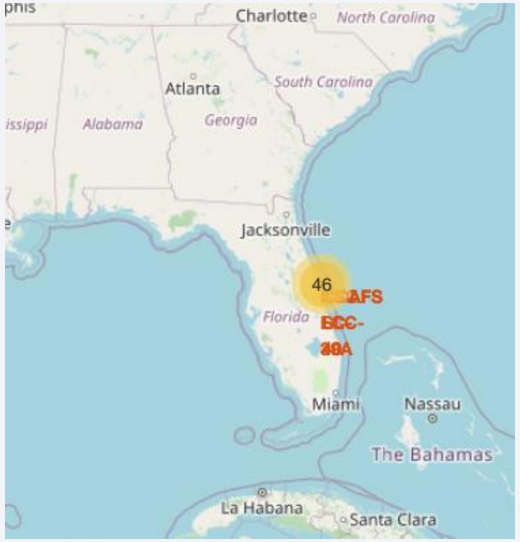
```
%sql
SELECT "Landing_Outcome", COUNT(*) AS outcome_count
FROM SPACEXTABLE
WHERE "Date" BETWEEN '2010-06-04' AND '2017-03-20'
GROUP BY "Landing_Outcome"
ORDER BY outcome_count DESC;
```

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

https://github.com/josevitor22/Applied-Data-Science-Capstone/blob/main/4-%20jupyter-labs-eda-sql-coursera_sqllite-Copy1.ipynb

Build an Interactive Map with Folium

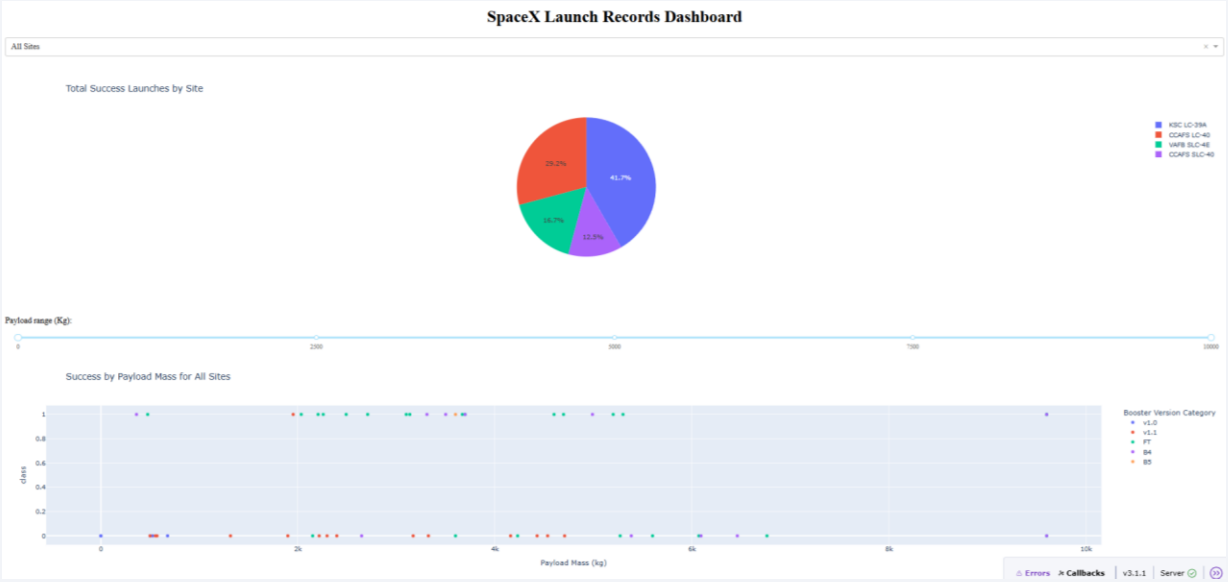
- **Markers with labels (Task 1):** To visually identify and label each SpaceX launch site on the map, making it easier to locate and distinguish them.
- **Circles (Task 1):** To highlight the geographic area of each launch site, helping emphasize their location and scale.
- **Colored Markers with MarkerCluster (Task 2):** To represent individual launch outcomes with intuitive color coding (green for success, red for failure) and to manage overlapping markers by clustering, improving map readability.
- **Polylines with Distance Labels (Tasks 3):** To illustrate and quantify the spatial relationships between launch sites and nearby geographic features like coastlines, highways, railways, and cities, which is important for assessing site proximities and logistical considerations.
- **MousePosition Plugin (Additional):** To enable interactive coordinate retrieval on the map, facilitating precise identification of points of interest for distance calculations.



https://github.com/josevitor22/Applied-Data-Science-Capstone/blob/main/6-%20lab_jupyter_launch_site_location.ipynb

Build a Dashboard with Plotly Dash

- **Launch Site Dropdown**
  Enables selection of specific site or all sites for dynamic filtering.
- **Success Pie Chart**
  - All Sites: Shows total successful launches per site
  - Selected Site: Shows success vs. failure ratio
- **Payload Range Slider**
  Filters data based on payload mass (kg) range.
- **Scatter Plot (Payload vs. Success)**
  Visualizes correlation between payload mass and launch success, colored by booster version.



https://github.com/josevitor22/Applied-Data-Science-Capstone/blob/main/8-%20spacex-dash-app.py

Predictive Analysis (Classification)

1. Data Preparation

Imported and loaded structured feature data (X) and target labels (Y = Class).

Standardized the feature data using StandardScaler to ensure uniform scaling.

Performed an 80/20 split into training and testing sets using train_test_split.

2. Model Training and Hyperparameter Optimization

Trained and optimized the following classification models using GridSearchCV with 10-fold cross-validation:

- Logistic Regression
- Support Vector Machine (SVM)
- Decision Tree
- K-Nearest Neighbors (KNN)

Evaluated a range of hyperparameters for each model to determine optimal configurations.

3. Model Evaluation

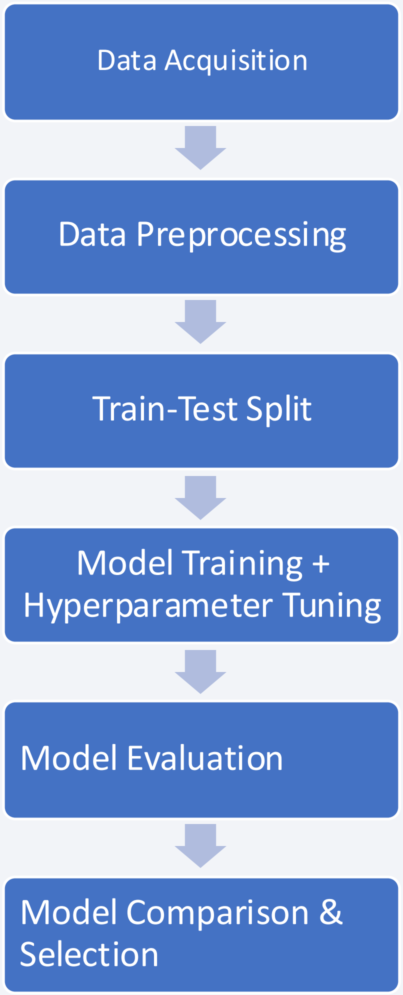
- Cross-validation accuracy
- Test accuracy
- Confusion matrix analysis

4. Model Comparison and Selection

Compared test set accuracies of all models:

- Logistic Regression: 83.3%
- SVM: 83.3%
- KNN: 83.3%
- Decision Tree: **94.4%**

Selected the Decision Tree model as the best-performing classifier based on highest test accuracy.



https://github.com/josevitor22/Applied-Data-Science-Capstone/blob/main/7-%20SpaceX_Machine%20Learning%20Prediction_Part_5.ipynb

Results

- **EDA results are located here (example of a query in Image 1):** https://github.com/josevitor22/Applied-Data-Science-Capstone/blob/main/4-%20jupyter-labs-eda-sql-coursera_sqllite-Copy1.ipynb
- Also for the EDA, was used an interactive Dashboard, presented in the image 2.
- Predictive analysis results:

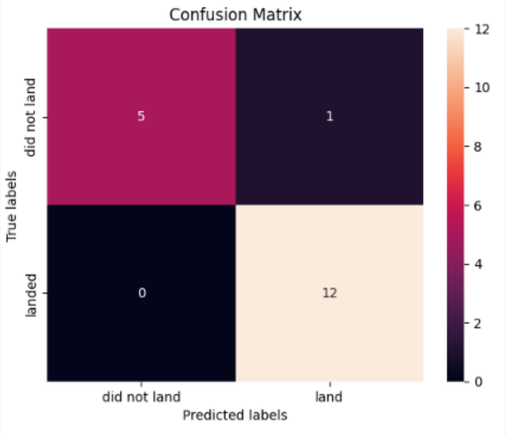
  Logistic Regression: 83.3%

  SVM: 83.3%

  KNN: 83.3%

  Decision Tree: **94.4%**

  Selected the Decision Tree model as the best-performing classifier based on highest test accuracy.



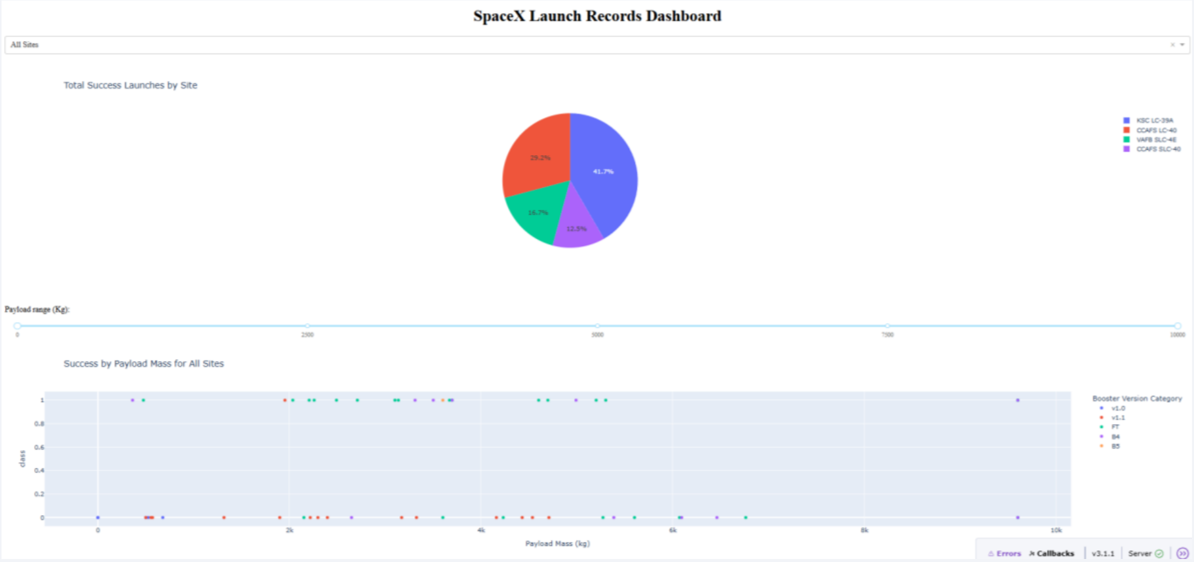
Task 4

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

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

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

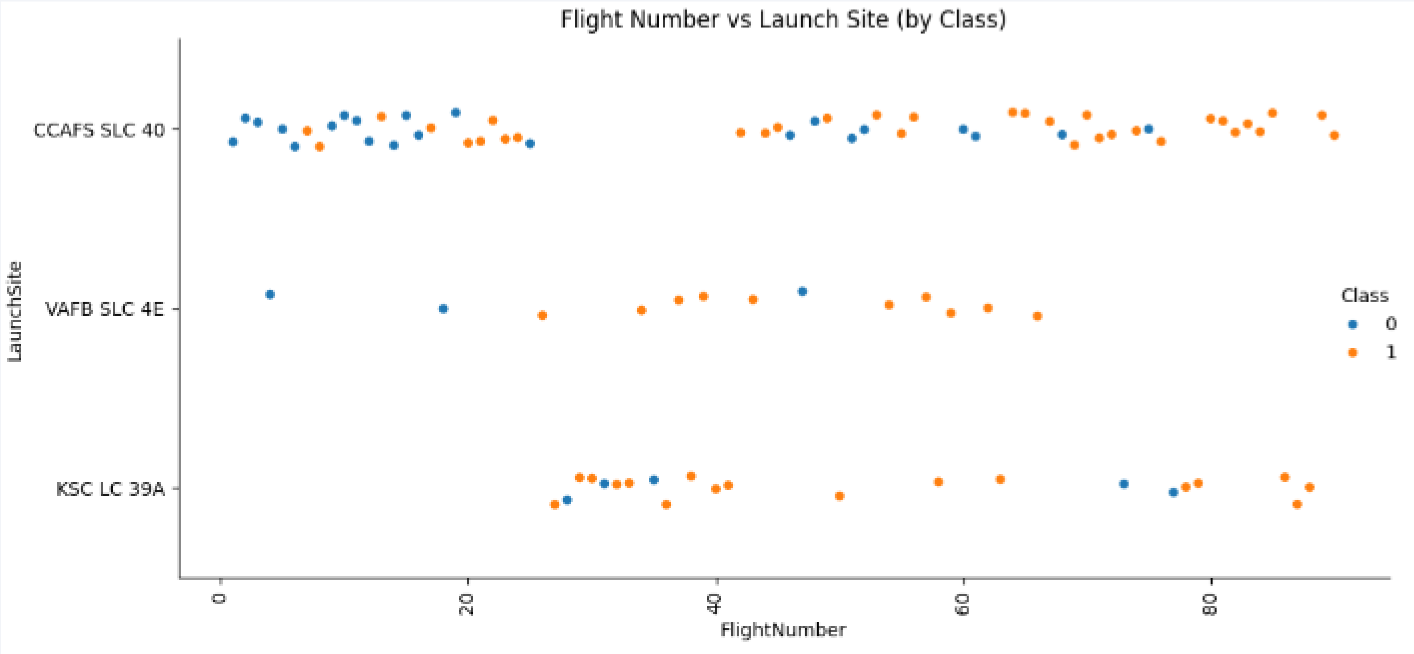
avg_payload_mass
2928.4



Section 2

Insights drawn from EDA

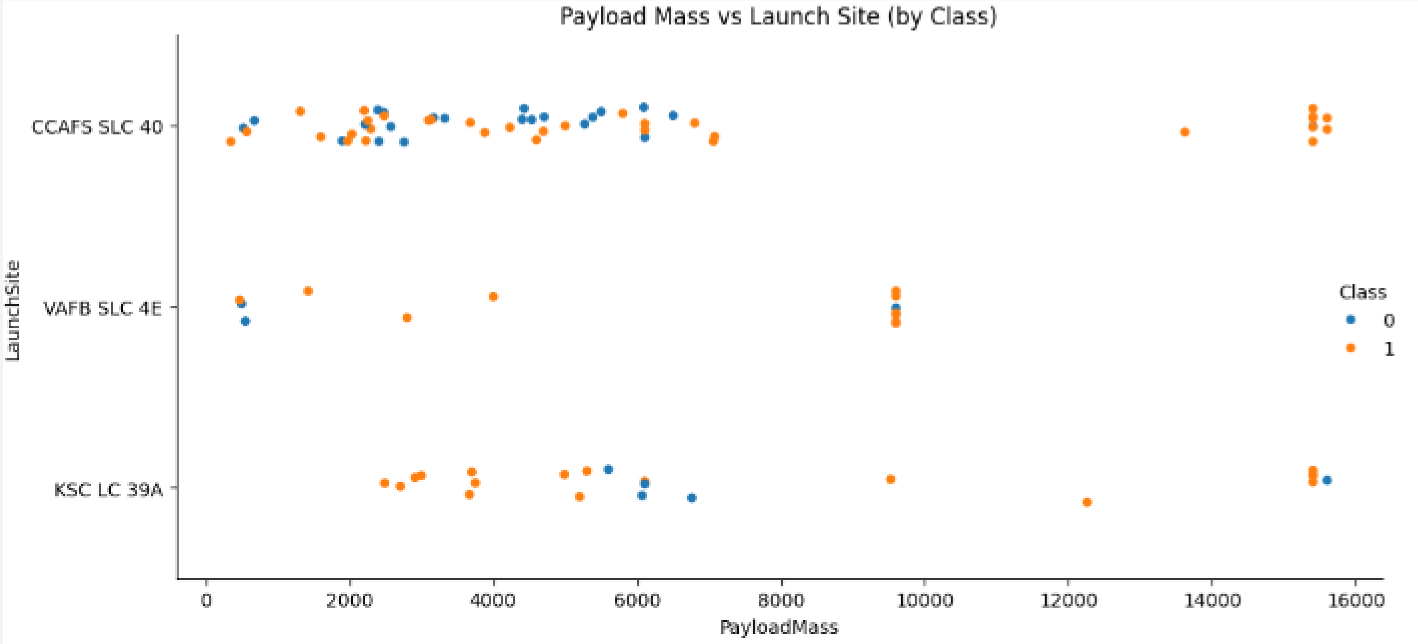
Flight Number vs. Launch Site



Explanation

- Flight experience matters: More recent flights tend to be more successful.
- Launch site performance varies:
  - CCAFS SLC 40: Largest sample size, showing gradual improvement.
  - KSC LC 39A: Fewer but more successful launches—possibly due to selective use.

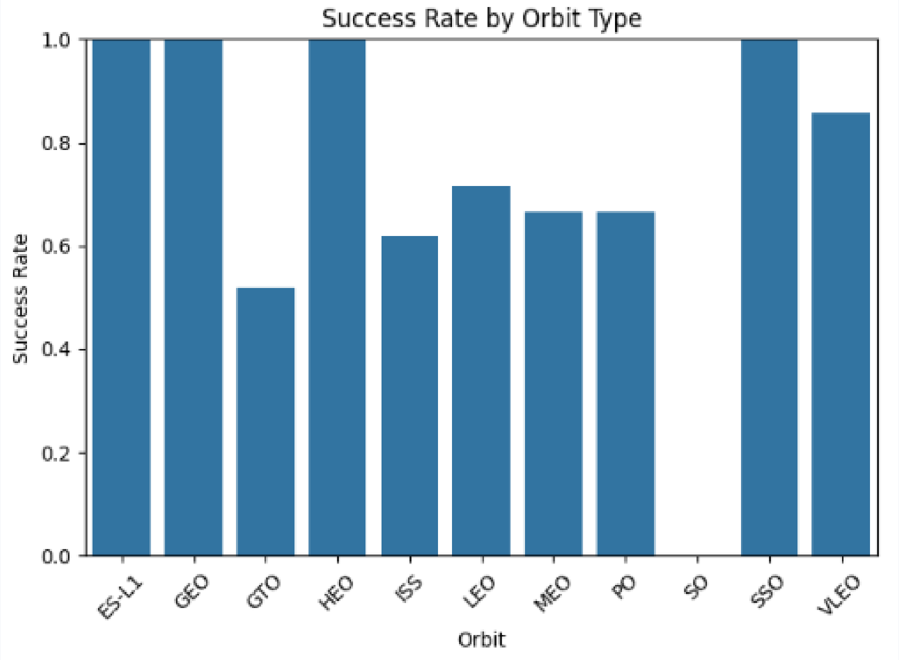
Payload vs. Launch Site



Explanation

- Heavier payloads don’t guarantee failure—some successful missions carried up to 15,000+ kg.
- Launch site and technology maturity may play a stronger role than payload size alone.
- KSC LC 39A handles heavy missions with high success, suggesting it's used for critical or capable missions.

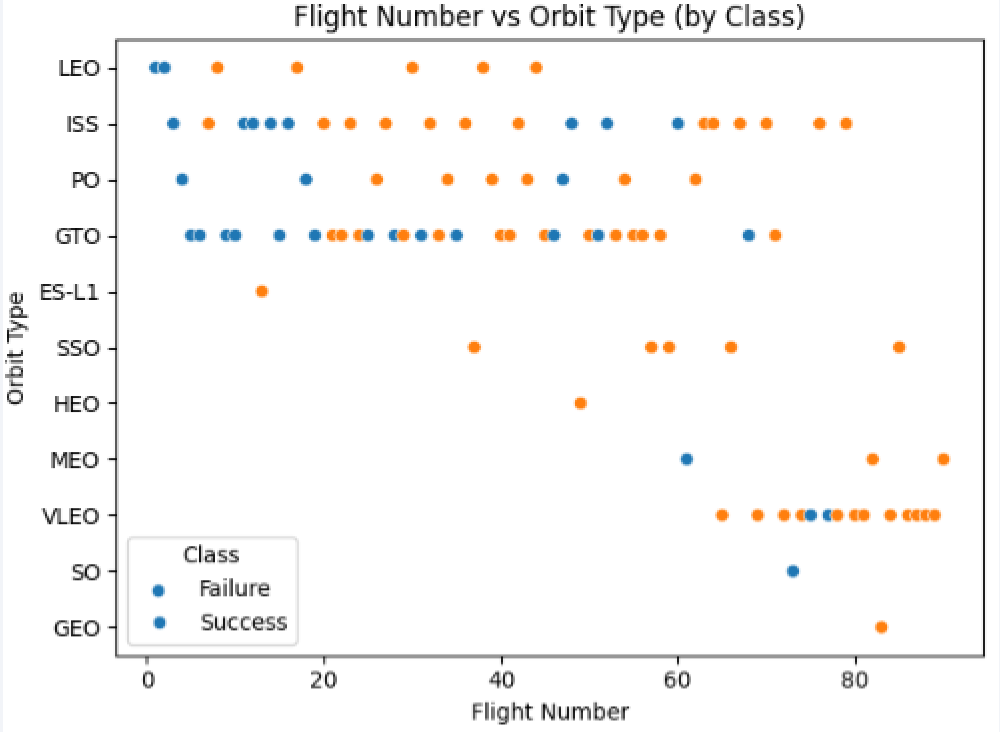
Success Rate vs. Orbit Type



Explanation

- Highest success rates observed for ES-L1, GEO, HEO, and SSO orbits.
- These orbits show no failed launches in the dataset.
- However, this is likely due to very few attempts, not necessarily higher reliability.
- Caution: Small sample sizes can skew success rate perception.

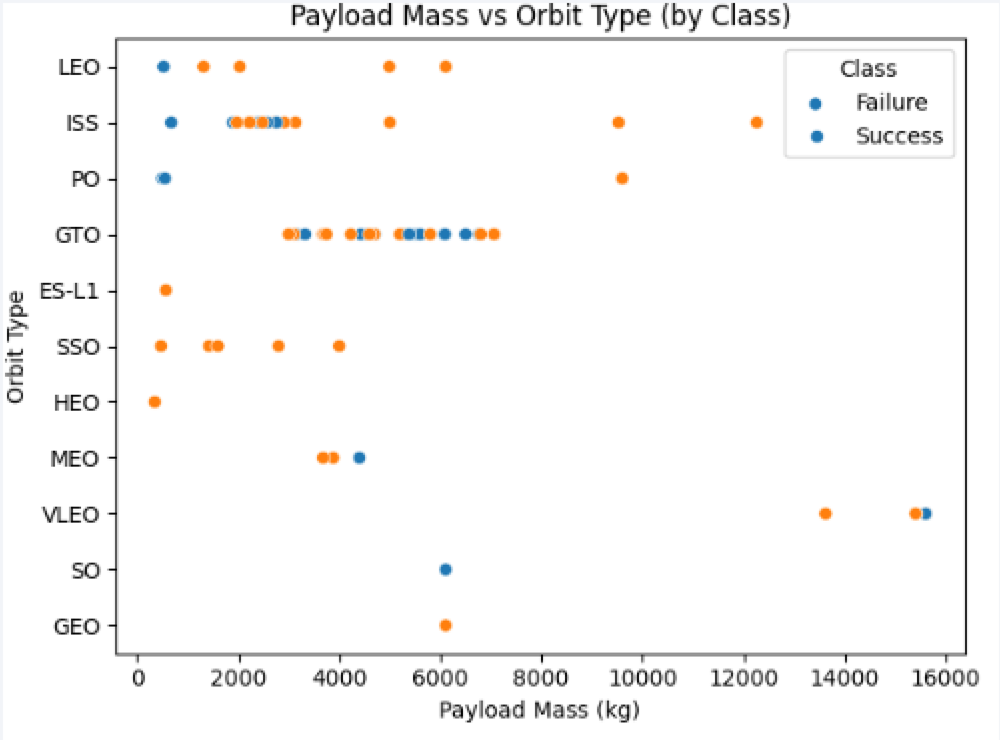
Flight Number vs. Orbit Type



Explanation

- Here it's possible to see that the most used orbits are LEO, ISS, GTO, PO and VLEO
- Also possible to identify that flight experience matters: More recent flights tend to be more successful.
- You can observe that in the LEO orbit, success seems to be related to the number of flights. Conversely, in the GTO orbit, there appears to be no relationship between flight number and success.

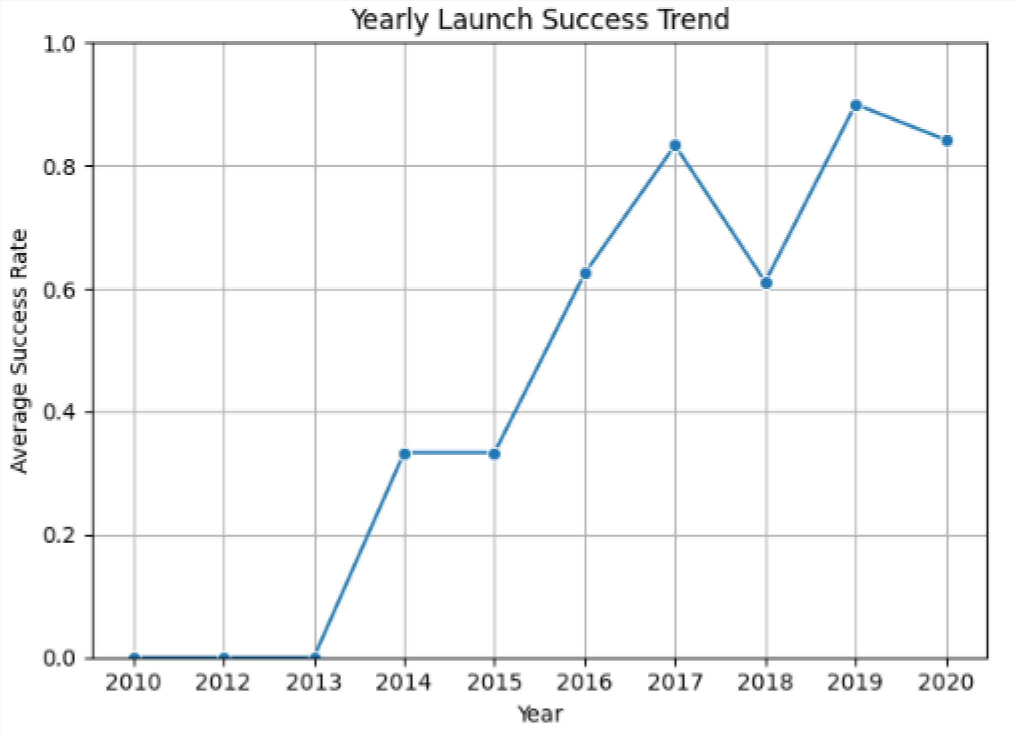
Payload vs. Orbit Type



Explanation

- With heavy payloads the successful landing or positive landing rate are more for Polar,LEO and ISS.
- However, for GTO, it's difficult to distinguish between successful and unsuccessful landings as both outcomes are present.

Launch Success Yearly Trend



Explanation

- It's possible to observe that the success rate since 2013 kept increasing till 2020

All Launch Site Names

In [10]:
```
%sql SELECT DISTINCT "Launch_Site" FROM SPACEXTABLE;
```

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

Out[10]:

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

Explanation

- There are four launch sites listed in the image to the left.
- These were extracted from a query directly to the SpaceX database

Launch Site Names Begin with 'CCA'

In [13]:
```
%sql SELECT * FROM SPACEXTABLE WHERE "Launch_Site" LIKE 'CCA%' LIMIT 5;
```

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

Out[13]:

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

Explanation

- Here are the first five records of lauch sites that begin with 'CCA'
- From the last query we also know that there are only two Launch Sites that begin with CCA

Total Payload Mass

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

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

total_payload_mass
45596

Explanation:

- To determine the total payload mass carried by NASA (CRS) missions, the analysis filtered the dataset to include only entries where the "Customer" field is 'NASA (CRS)'. Then, it computed the sum of the values in the "PAYLOAD_MASS__KG_" column.

Average Payload Mass by F9 v1.1

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

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

avg_payload_mass
2928.4

Explanation:

- To calculate the average payload mass for missions that used the F9 v1.1 booster version, the dataset was filtered to include only records where the "Booster_Version" is 'F9 v1.1'. The average of the "PAYLOAD_MASS__KG_" values was then computed.

First Successful Ground Landing Date

```
%sql SELECT MIN("Date") AS first_successful_landing_date FROM SPACEXTABLE WHERE "Landing_Outcome" LIKE '%Success (ground pa
```

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

first_successful_landing_date
2015-12-22

Explanation:

- To identify the first successful landing on a ground pad, the dataset was filtered for records where the "Landing_Outcome" contains the phrase 'Success (ground pad)'. Then, the earliest date of such landings was retrieved using the MIN() function.

Successful Drone Ship Landing with Payload between 4000 and 6000

```
%sql SELECT DISTINCT "Booster_Version" FROM SPACEXTABLE WHERE "Landing_Outcome" LIKE '%Success (drone ship)%' AND "PAYLOAD_
```

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

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

Explanation:

- To find boosters that had successful drone ship landings and carried a payload mass between 4000 kg and 6000 kg, the dataset was filtered using both a landing outcome condition and a payload mass range. The DISTINCT keyword was used to return unique booster versions.

Total Number of Successful and Failure Mission Outcomes

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

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

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

Explanation:

- To determine the total count of successful and failed mission outcomes, a SQL GROUP BY operation was performed on the Mission_Outcome column to group similar outcomes and count their occurrences.

Boosters Carried Maximum Payload

List all the booster_versions that have carried the maximum payload mass, using a subquery with a suitable aggregate function.

```
%sql SELECT "Booster_Version", "PAYLOAD_MASS__KG_" FROM SPACEXTABLE WHERE "PAYLOAD_MASS__KG_" = (SELECT MAX("PAYLOAD_MASS__
```

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

Booster_Version	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

Explanation:

- To identify which boosters carried the maximum payload mass, a subquery was used to first find the maximum value in the PAYLOAD_MASS__KG_ column. Then, the main query selected all records with that payload value.

2015 Launch Records

```
%%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,
"Landing_Outcome",
"Booster_Version",
"Launch_Site"
FROM SPACEXTABLE
WHERE substr("Date", 1, 4) = '2015'
  AND "Landing_Outcome" LIKE '%Failure (drone ship)%';
```

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

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

Explanation:

- This query extracted records from the year 2015 where the landing outcome was a failure on a drone ship. Since SQLite doesn't support month names directly, the substr function was used to derive the month number and map it to its respective name.

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
```

```
 * 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

Explanation:

- This query ranked the frequency of **landing outcomes** within a specified time window. The goal was to understand which outcomes were most common during the early to mid-stage of SpaceX's reusable booster development.

Section 3

Launch Sites Proximities Analysis

SpaceX Launch Sites

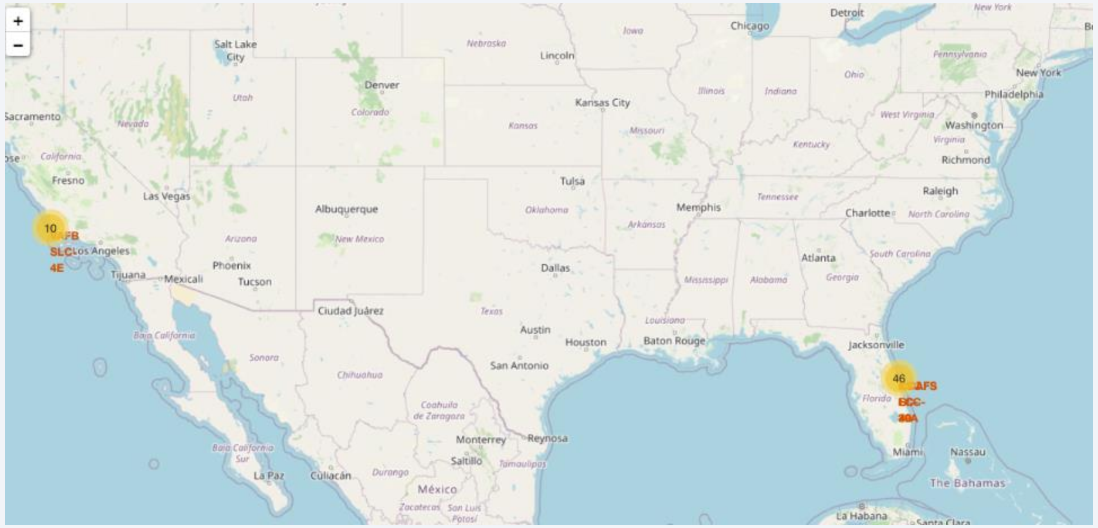
Explanation:

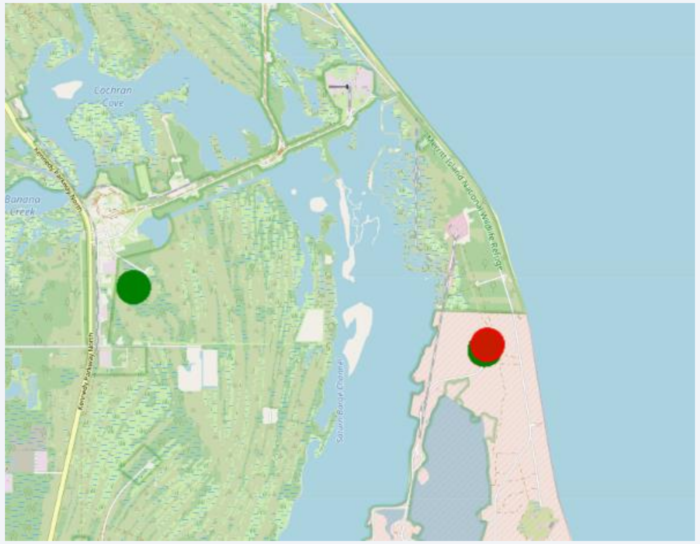
- Here you can see all SpaceX launch sites marked on a global map. The markers show their geographic locations, highlighting key launchpads in Florida and California. This distribution reflects SpaceX's strategic positioning for different orbital missions.

SpaceX Launch Outcomes

Explanation:

- This map displays SpaceX launch sites with markers color-coded by launch outcome:
- Green for successful landings
- Red for failures
- The color labels visually highlight the success rate at each site, showing where most successful recoveries have occurred and pinpointing locations with failures. This helps quickly assess operational performance across different launchpads.





SpaceX Launch Site Proximities

Explanation:

- The map zooms in on a SpaceX launch site, showing nearby coastline location. Distance lines connect the launch site to this point, illustrating its proximity and accessibility.

Section 4

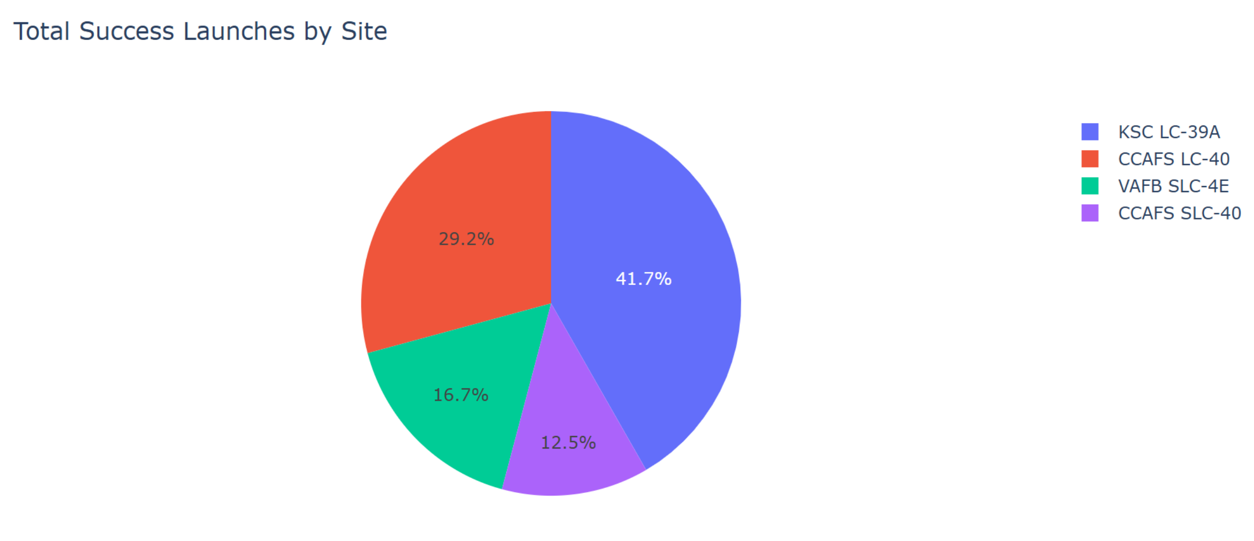
Build a Dashboard with Plotly Dash

Total Success Launches by Site

Explanation:

- The pie chart illustrates the distribution of successful launches across the four SpaceX launch sites.
- It highlights that some sites have a higher success rate than others, with KSC LC-39A accounting for the most successful launches.

https://github.com/josevitor22/Applied-Data-Science-Capstone/blob/main/SpaceX%20Launch%20Records%20Dashboard.pdf

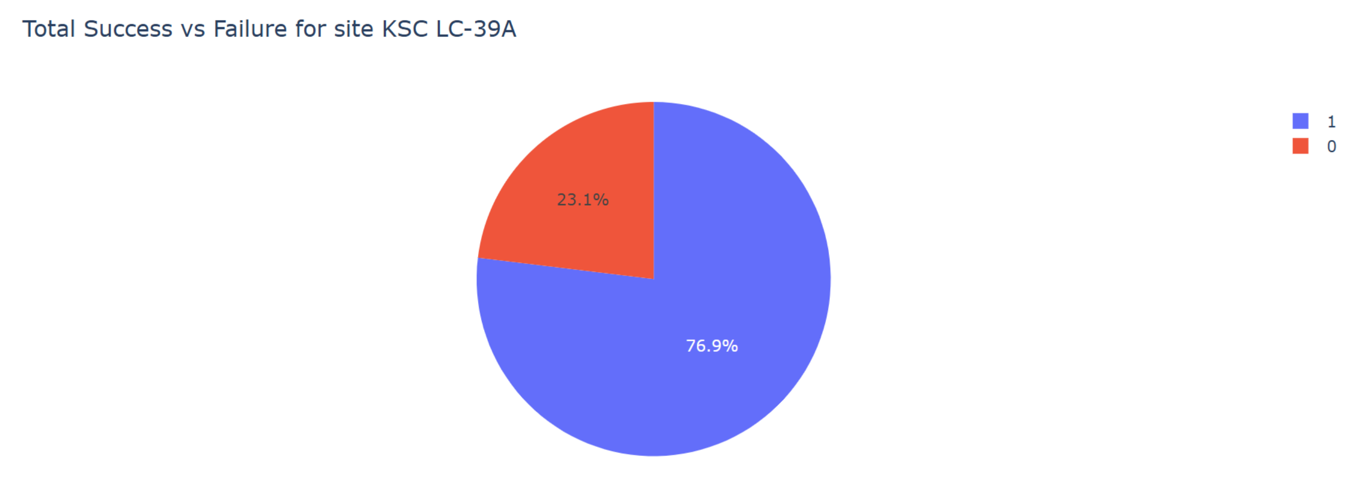


Launch Success Rate at KSC LC-39A

Explanation:

- KSC LC-39A accounts for the highest number of successful launches. The pie chart shows its launch success rate, with 76.9% of missions achieving success.
- This high percentage underscores the site's strong reliability and performance in SpaceX's launch operations.

https://github.com/josevitor22/Applied-Data-Science-Capstone/blob/main/SpaceX%20Launch%20Records%20Dashboard.pdf

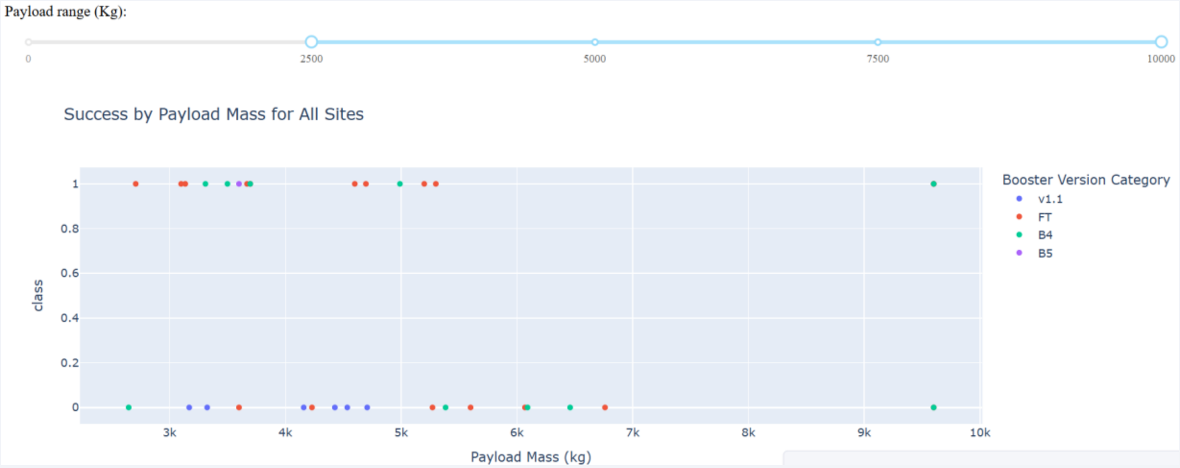


Launch Success Trends by Payload and Booster Version

Explanation:

- The scatter plot shows the relationship between payload mass and launch outcome across all SpaceX sites.
- Using the range slider to filter specific payload ranges, we can observe that medium-weight payloads (e.g., 2,000 – 5,000 kg) tend to have a higher success rate.
- Certain booster versions, such as FT and Block 5, also show consistently successful outcomes, especially within this payload range.

https://github.com/josevitor22/Applied-Data-Science-Capstone/blob/main/SpaceX%20Launch%20Records%20Dashboard.pdf

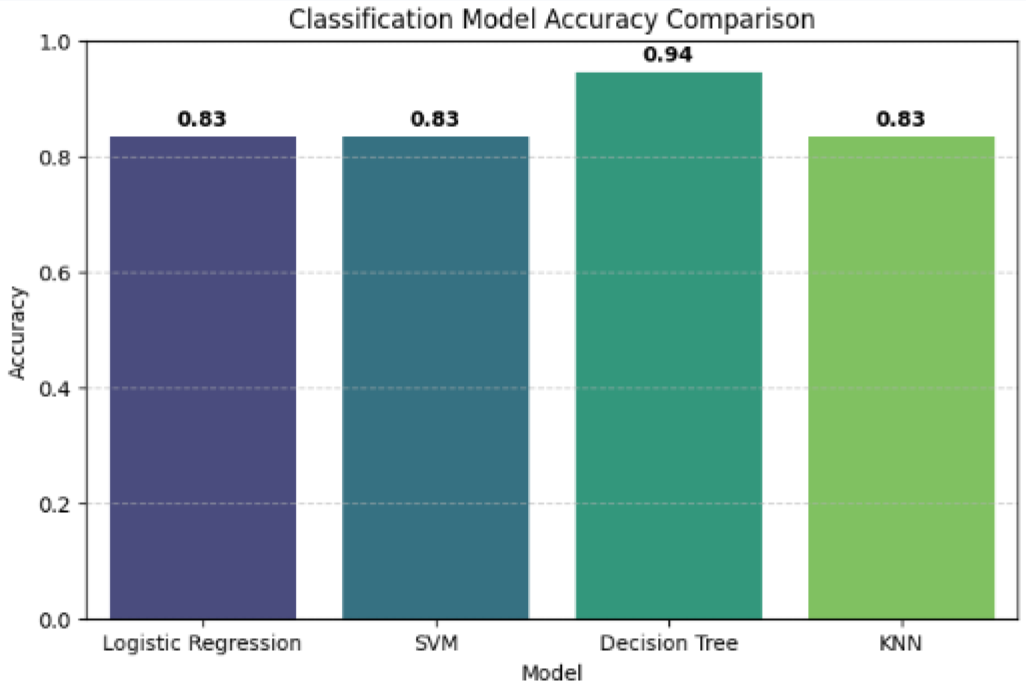


Section 5

Predictive Analysis (Classification)

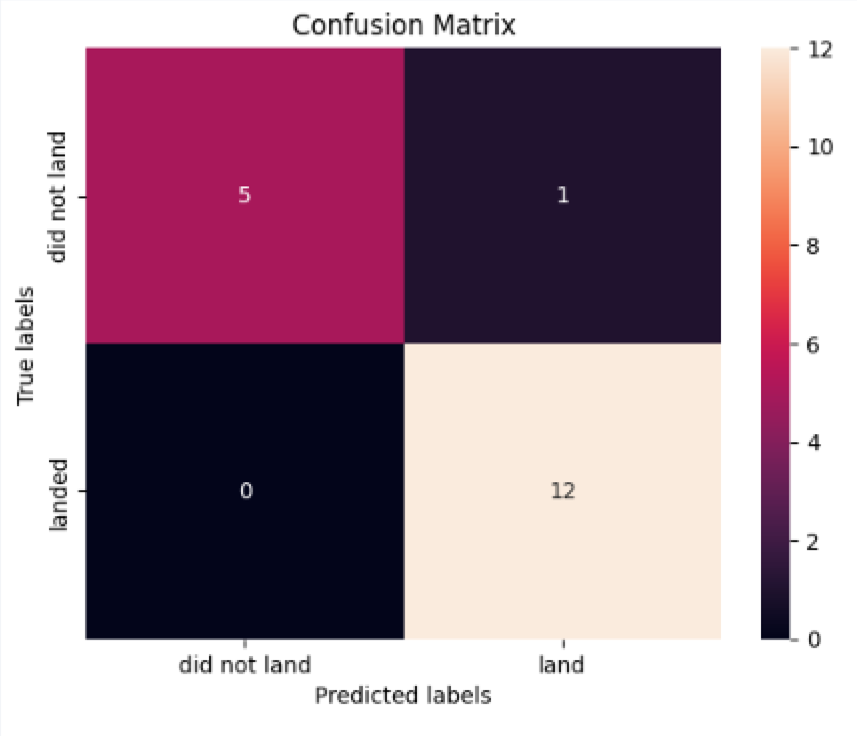
Classification Accuracy

- The bar chart compares the accuracy of all classification models built.
- It clearly shows that the Decision Tree Classifier achieved the highest accuracy, indicating its superior performance in predicting launch outcomes.



Confusion Matrix

- The confusion matrix shows the Decision Tree's classification results on the test data:
- True Positives (TP) = 12: The model correctly predicted 12 successful landings.
- True Negatives (TN) = 5: The model correctly identified 5 failures.
- False Positives (FP) = 1: The model incorrectly predicted 1 failure as a success.
- False Negatives (FN) = 0: The model did not miss any successful landings.
- These results demonstrate that the Decision Tree model achieves high accuracy with minimal misclassification, especially by avoiding false negatives, which is critical for correctly identifying successful landings.



Conclusions

- **Launch Site Usage:** The KSC LC-39A launch site is the most frequently used by SpaceX and shows a high launch success rate, indicating its reliability.
- **Payload Impact:** The success of launches varies with payload mass and booster version, highlighting the importance of these factors in mission planning.
- **Model Performance:** Among the classification models tested—Logistic Regression, SVM, Decision Tree, and KNN—the Decision Tree model achieved the highest accuracy (94.4%) on the test data.
- **Decision Tree Strength:** The Decision Tree model demonstrated strong predictive ability with minimal false negatives, making it a reliable tool for predicting launch outcomes.
- **Practical Use:** The developed models can assist SpaceX and similar organizations in improving launch success predictions, optimizing resources, and enhancing mission safety.

Appendix

- https://github.com/josevitor22/Applied-Data-Science-Capstone/tree/main

Thank you!