Final Course Project in Applied Data Science

Pedro Henrique Brito Liberal

Project Introduction
Falcon 9 First Stage Landing Prediction – SpaceX

Objective:

Predict whether the first stage of the Falcon 9 rocket will successfully land using real-world data collected from the SpaceX API.

Context:

- A Falcon 9 launch costs **$62 million**;
- Competitors can charge over **$165 million**;
- SpaceX’s lower cost is mainly due to **reusing the first stage**;
- If we can predict whether the first stage will land, this insight can help other companies bid competitively for launches.

Data Collection

Data Source:

- Public SpaceX API
- Data fetched using HTTP requests (`requests` library)

Goal of this step:

- Extract data on Falcon 9 launches;
- Ensure the data is in a proper format for cleaning, exploration, and modeling.

Web Scraping (Wikipedia Dataset)

Data Enrichment via Web Scraping

Objective:
To enhance the dataset with historical launch records of Falcon 9 from an external source: Wikipedia.

Tools Used:

- `requests` (to fetch the HTML page)
- `BeautifulSoup` (to parse and extract table data)

What was done:

- Extracted launch tables from the [Wikipedia page]
- Parsed and cleaned relevant fields:
  `Flight No.`, `Date`, `Time`, `Launch Site`, `Booster Version`, `Payload`, `Payload Mass`, `Orbit`, `Customer`, `Launch Outcome`, `Booster Landing`
- Built a clean DataFrame to be merged with the API data

Data Wrangling – Converting Landing Outcome to Label

Transformation Applied:

Context:

Landing outcome values include:

- `True ASDS`, `True RTLS`, `True Ocean` → Successful landings
- `False ASDS`, `False RTLS`, `False Ocean` → Failed landings

```
def classify_landing(outcome):
    return 1 if 'True' in outcome else 0

df['Class'] = df['Outcome'].apply(classify_landing)
```

Result:

A new binary column `Class` was added to the dataset:

- `1` = Booster successfully landed
- `0` = Booster did not land

EDA with SQL

Exploring Falcon 9 Launch Data

Goal:

Explore trends and patterns in Falcon 9 launch data using SQL queries.

Key Questions Answered:

- What are the unique launch sites?
- What's the average payload mass for F9 v1.1?
- When was the first successful ground pad landing?
- Which boosters carried the heaviest payload?
- How do landing outcomes vary across time and location?

Example Insight:

CCAFS LC-40 had lower success rates compared to KSC LC-39A and VAFB SLC-4E.
Payloads > 10,000kg from CCAFS LC-40 had a **100% landing success rate**.

Tools Used:

- SQLite + SQLAlchemy
- `%sql` magic in Jupyter Notebook

Outcome:

Discovered key features like launch site, booster version, payload mass, and orbit that impact landing outcome.

Visual EDA & Feature Engineering

Goal:

Identify visual trends and transform the dataset for modeling.

EDA Highlights:

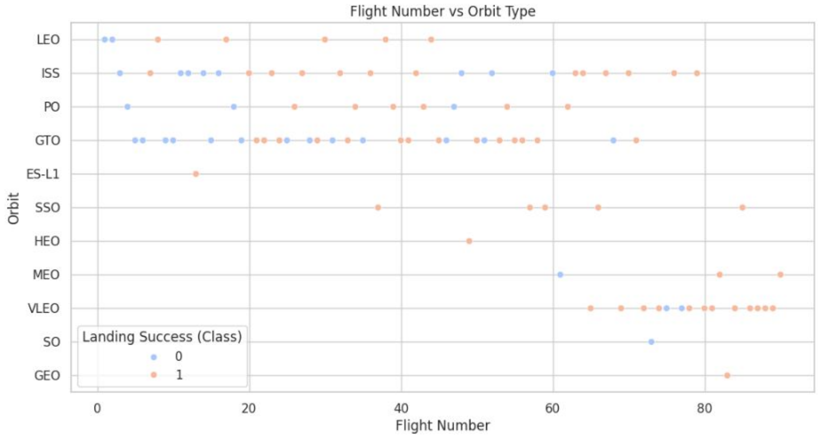
- Payload Mass vs. Launch Site → heavier payloads succeed more often in specific locations
- Orbit vs. Flight Number → LEO shows increasing success with flight experience
- Yearly success trend → continuous improvement in landing rates

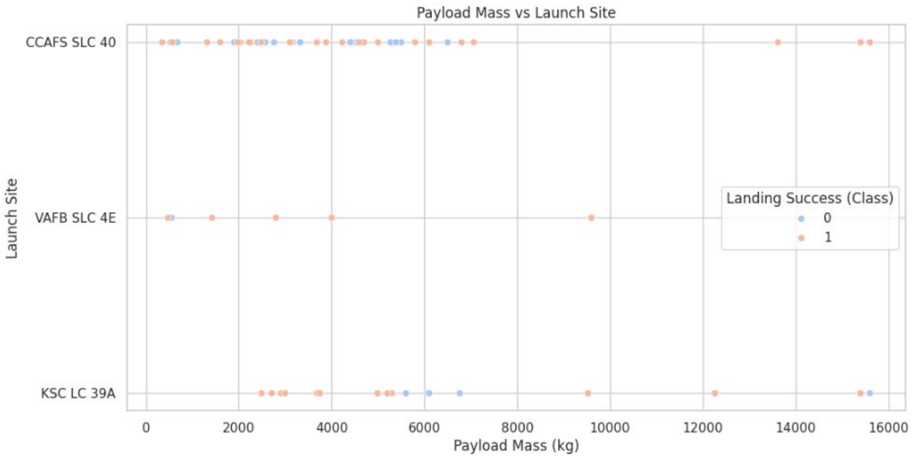
Feature Engineering:

- Selected core variables (Flight Number, Payload, Orbit, etc.)
- Applied One-Hot Encoding on categorical features: `Orbit`, `LaunchSite`, `LandingPad`, `Serial`
- Cast all data to `float64` for model compatibility
- Exported final dataset for modeling stage

Tools Used:
`pandas`, `matplotlib`, `seaborn`

Visual EDA & Feature Engineering





Interactive Map with Folium

Interactive Launch Site Mapping with Folium

Goal:

Visualize the location of SpaceX launch sites and their proximity to coastlines, cities, railways, and highways.

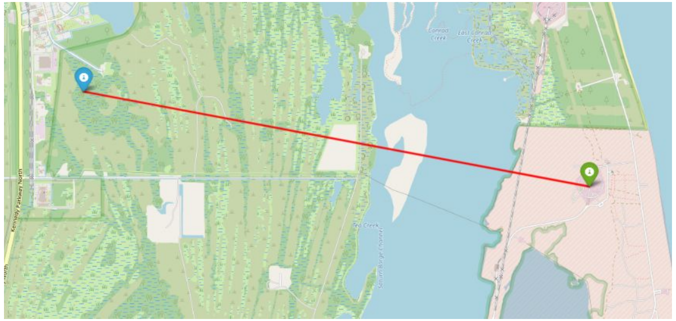
What was done:

- Mapped all SpaceX launch sites using their coordinates
- Marked individual launches as green/red icons (success/failure)
- Calculated distances to:
  - 🏖️ Coastline
  - 🛣️ Highway
  - 🏙️ Nearby city
  - 🚆 Railway
- Connected points using `folium.PolyLine`
- Displayed distance labels directly on the map (`DivIcon`)

Tools Used:

- `folium`, `numpy` (haversine formula)

Example Output: 📍 Map showing KSC LC-39A, nearest coastline and a distance of **0.59 KM**

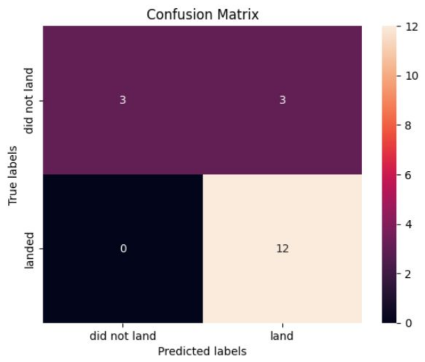


Machine Learning Classification Results
Predicting Falcon 9 First Stage Landing Success

Goal:
Train machine learning models to predict whether the Falcon 9 first stage will land successfully.

Models Tested:

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



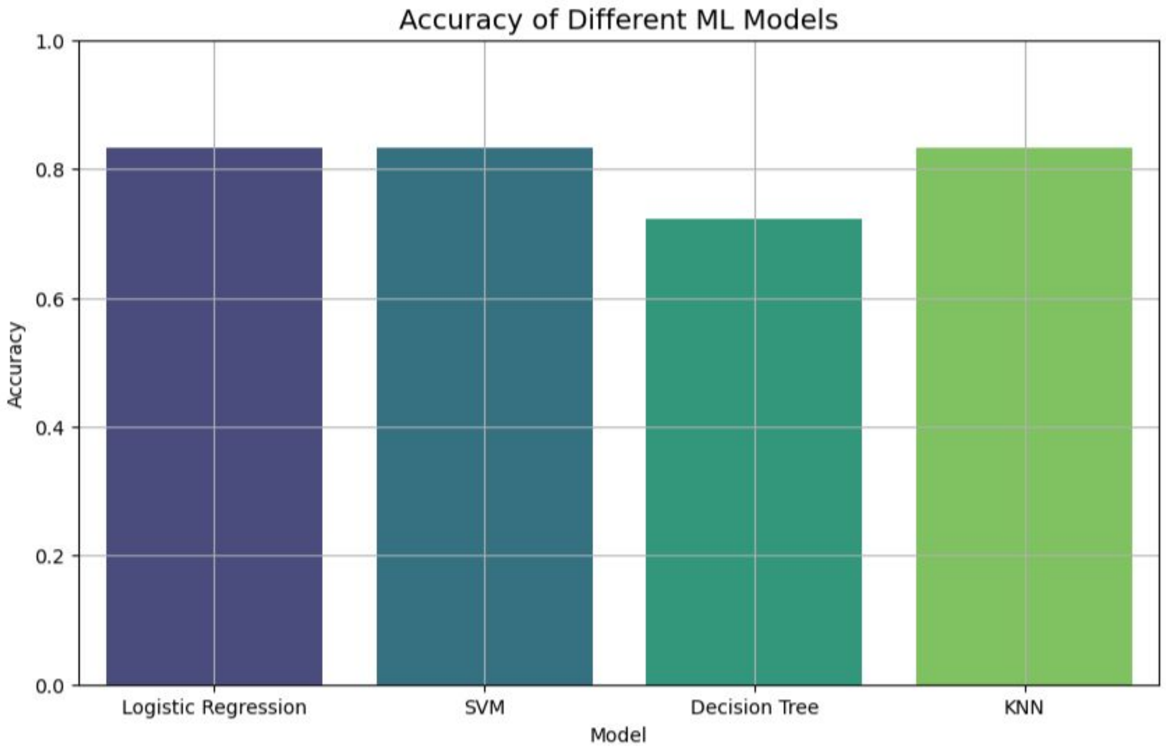
Evaluation:

- Accuracy measured using **10-fold Cross Validation**
- Best model: e.g. **SVM** with 84% accuracy
- Confusion matrix to assess precision and recall

Tools Used:

- `scikit-learn`, `GridSearchCV`, `StandardScaler`, `matplotlib`, `seaborn`

Predictive Accuracy for Falcon 9 Landing Success



Executive Summary

Executive Summary – Predicting Falcon 9 First Stage Landings

This project aims to predict whether the first stage of SpaceX's Falcon 9 rocket will successfully land.

To achieve this, we collected and analyzed historical data using public APIs and web scraping techniques.

After exploratory 📊 data analysis and feature engineering, we trained multiple 🤖 machine learning models to classify landing success.

Key insights include:

- Landing success rate has significantly improved over time
- Payload mass and launch site strongly influence landing outcome
- SVM and Logistic Regression achieved the best predictive accuracy (~84%)

An interactive dashboard and 🗺️map visualization were also developed to explore launch success factors.