SpaceX Capstone: Full Slide Content Outline

# Executive Summary

* Goal: Predict the success of Falcon 9 first-stage landings.
* Business case: SpaceX saves cost through reusability.
* Approach: Data collection, EDA, visualization, machine learning.
* Tools: Python, Dash, Folium, SQL, Scikit-learn.

# Project Background

* SpaceX launches are significantly cheaper due to reusability.
* Predicting landing success helps understand launch economics.
* Project explores this using real launch data from various sources.

# Methodology Overview

* 1. Data Collection – API and Web Scraping.
* 2. Data Wrangling and Preparation.
* 3. Exploratory Data Analysis (EDA) and Visualization.
* 4. Interactive Dash and Geospatial Analysis.
* 5. Machine Learning Classification Modeling.

# Data Sources Overview

* • SpaceX REST API – launch data.
* • Wikipedia – rocket details and historical launches.
* • External JSON/CSV datasets.
* Data integrated and cleaned into a unified DataFrame for modeling.

# Data Collection – SpaceX API

* Collected historical launch data using SpaceX REST API.
* Data includes launch site, payload mass, orbit, and landing outcome.
* Sample API response parsed from JSON to DataFrame.
* Challenges: handling inconsistent or missing fields.

# Data Collection – Web Scraping

* Supplemented API data with Wikipedia scraping for rocket specs.
* Used BeautifulSoup to extract tabular data on Falcon 9 launches.
* Data merged and deduplicated based on flight number and date.

# Data Wrangling

* Merged datasets from API and web sources.
* Dropped nulls and redundant columns.
* Created new features: booster version, landing pad, etc.
* Filtered for Falcon 9 missions only.

# EDA – Visualization Highlights

* Used Seaborn and Matplotlib for visual analysis.
* Key charts: payload vs. success, launch site success rate, orbit type trends.
* Detected correlations and candidate features for modeling.

# EDA – SQL Queries

* Queried launch site names, payload distributions, and mission outcomes.
* Example: total payload mass from NASA launches.
* Used SQLite to interact with structured dataset and confirm patterns.

# Interactive Maps with Folium

* Mapped launch sites using Folium.
* Markers show site names, success/failure, and coordinates.
* Used Circle and PolyLine to highlight proximity to coastlines, highways.

# Plotly Dash Dashboard – Overview

* Interactive dashboard built using Dash and Plotly Express.
* Dropdown filter for launch site selection.
* Slider to filter by payload range.
* Output: pie chart (launch success) and scatter plot (payload vs. outcome).

# Plotly Dash – Launch Success Pie Chart

* Pie chart shows success count per site or for all sites combined.
* Highlights which sites have higher reliability.

# Plotly Dash – Payload vs. Outcome Scatter

* Scatter plot visualizes relationship between payload mass and landing outcome.
* Interactive payload slider allows dynamic filtering.
* Insights: mid-range payloads often show higher success.

# Machine Learning Overview

* Objective: classify if a launch will be successful.
* Features: payload, orbit, booster version, launch site, etc.
* Target: landing success (1) or failure (0).
* Models tested: Logistic Regression, Decision Tree, SVM, Random Forest.

# Model Building & Preprocessing

* Preprocessing steps: encoding categorical variables, feature scaling.
* Train-test split: 80/20 ratio with stratification to preserve class balance.
* StandardScaler used for numerical features like payload mass.
* OneHotEncoder used for orbit, launch site, and booster version.

# Model Evaluation Metrics

* Used Accuracy, Precision, Recall, and F1-Score.
* Confusion Matrix used to analyze false positives and false negatives.
* Cross-validation applied to improve reliability of scores.

# Logistic Regression Results

* Performed well with limited overfitting.
* Achieved moderate accuracy (~84%).
* Relatively interpretable coefficients.

# Decision Tree Results

* Prone to overfitting with full depth.
* Improved using pruning and depth limits.
* Less generalizable than ensemble methods.

# Support Vector Machine (SVM) Results

* High accuracy with tuned hyperparameters (kernel, C).
* Sensitive to feature scaling.
* Training time increased with dataset size.

# Random Forest Results

* Best overall performance (~88% accuracy).
* Robust to overfitting with sufficient estimators.
* Feature importance scores were insightful.

# Model Comparison

* Random Forest outperformed others on test data.
* Logistic Regression offered best interpretability.
* Evaluation involved confusion matrix and ROC curve.

# Feature Importance Analysis

* Top features: Payload Mass, Orbit Type, Launch Site, Booster Version.
* Helped interpret why predictions leaned toward success or failure.
* Guided potential simplification of feature set.

# Model Deployment Potential

* Model could be integrated into real-time dashboard or API.
* Useful for estimating cost/success likelihood of future launches.
* Could inform operational or business decisions.

# Modeling Limitations

* Class imbalance in original dataset may affect recall.
* Limited data for rare booster versions or orbits.
* Model performance is tied to the quality of public SpaceX data.

# Geospatial Analysis – Launch Sites

* Used Folium to map launch site coordinates.
* Visualized location-based patterns in mission outcomes.
* Markers indicated site name and outcome color-coded.

# Geospatial Insights – Proximities

* Assessed proximity of launch sites to coastlines, roads, railways.
* Displayed radius markers and calculated distances.
* Helped evaluate logistic feasibility and risk factors.

# Launch Outcomes by Region

* Success rates differed by location (e.g., KSC LC-39A vs. CCAFS).
* West Coast sites showed slightly lower success consistency.

# Dashboard Summary – Interactive Visuals

* Pie chart: Launch success by site.
* Scatter: Payload vs. outcome, filterable by range and site.
* Enabled exploratory insights and stakeholder interaction.

# Key Takeaways from Dashboard

* Highest success at KSC LC-39A and VAFB.
* Payload range of 2,000–6,000 kg had best success ratio.
* Orbital class had moderate influence on outcome.

# Overall Project Results

* Achieved 88% accuracy using Random Forest classifier.
* Identified payload mass and site as top predictors.
* Interactive dashboards and geospatial analysis enriched insights.

# Business Implications

* Predictive models help forecast cost and risk.
* Can be used by competitors or partners to benchmark performance.
* Supports data-driven decision-making in aerospace logistics.

# Recommendations

* Incorporate live API data for real-time prediction updates.
* Expand dataset with additional providers for broader comparison.
* Deploy the model in a monitoring dashboard for internal use.

# Future Work

* Deep learning models for sequential data (e.g., LSTMs).
* Add weather, time-of-day, or mission type as features.
* Explore unsupervised clustering for anomaly detection.

# Final Thoughts

* This project demonstrates how public data and open-source tools
* can offer valuable insights into one of the world’s most advanced private space programs.
* Future iterations could further bridge technical analysis with strategic applications.