Winning Space Race with Data Science

Leandro Rangel de Souza
2025-08-31

Leandro Rangel de Souza

Outline

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

Executive Summary

- Summary of methodologies
  - The course incorporated videos, reading materials, and audio to explain the content of various topics, enabling learners to explore features and experiment with Python functionalities.
- Summary of all results
  - As a result, the course delivered rich and comprehensive content, covering topics such as data modeling, chart creation, dashboard development, and exploratory analysis, while illustrating real-world problems typically addressed by data scientists.

Introduction

- Project background and context
- The project leverages data from SpaceX’s launches and flight tests as a database for analysis.
- Problems you want to find answers
- Some challenges were encountered while using IBM’s test environment; however, in most cases, the code could be executed locally.

Section 1

Methodology

Leandro Rangel de Souza

Methodology

Executive Summary

- Data collection methodology:
  - Download public datasets, compiled and organized in IBM S3 repositories.
- Perform data wrangling
  - The provided data was processed within IBM Watson’s test environment.
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - The classification models were developed to meet the specific requirements of each task, consistently adhering to best practices for model and hyperparameter selection.

Data Collection

- The data collection process involved original sourcing from Kaggle containing house sales data from King County, Washington (including Seattle) between May 2014 and May 2015, followed by modification and distribution by IBM Skills Network for educational purposes through their cloud storage system. The collection was geographically and temporally filtered to focus on residential real estate transactions in a specific metropolitan area, with data structured to include comprehensive housing attributes such as square footage, bedrooms, bathrooms, location coordinates, and various quality indicators relevant for real estate investment analysis and price prediction modeling. The datasets were gathered through HTTP requests using Python libraries.

Data Collection – SpaceX API

- Present your data collection with SpaceX REST calls using key phrases and flowcharts

- Add the GitHub URL of the completed SpaceX API calls notebook (https://github.com/Irangels/coursera-applied-data-science), as an external reference and peer-review purpose

Below we will define a series of helper functions that will help us use the API to extract information using identification numbers in the launch data.

From the `rocket` column we would like to learn the booster name.

+ Code + Markdown

```python
# Takes the dataset and uses the rocket column to call the API and append the data to the list
def getBoosterVersion(data):
    for x in data['rocket']:
        if x:
         response = requests.get("https://api.spacexdata.com/v4/rockets/"+str(x)).json()
         BoosterVersion.append(response['name'])
```

Python

From the `launchpad` we would like to know the name of the launch site being used, the logitude, and the latitude.

```python
# Takes the dataset and uses the launchpad column to call the API and append the data to the list
def getLaunchSite(data):
    for x in data['launchpad']:
        if x:
          response = requests.get("https://api.spacexdata.com/v4/launchpads/"+str(x)).json()
          Longitude.append(response['longitude'])
          Latitude.append(response['latitude'])
          LaunchSite.append(response['name'])
```

Python

Data Collection - Scraping

- https://github.com/lrangels/coursera-applied-data-science/blob/52cffe6200609e8ffe1c4ec446302239d50785c9/WebScraping_Review_Lab.ipynb

	Rank	Country	Population	Area	Density
0	1	Singapore	5,921,231	719	8,235
1	2	Bangladesh	165,650,475	148,460	1,116
2	3	Palestine[note 3][102]	5,223,000	6,025	867
3	4	Taiwan[note 4]	23,580,712	35,980	655
4	5	South Korea	51,844,834	99,720	520
5	6	Lebanon	5,296,814	10,400	509
6	7	Rwanda	13,173,730	26,338	500
7	8	Burundi	12,696,478	27,830	456
8	9	Israel	9,402,617	21,937	429
9	10	India	1,389,637,446	3,287,263	423

Data Wrangling

- The data wrangling process for the laptop pricing dataset followed a systematic 6-step approach: first, missing data identification was performed using df.isnull(), followed by strategic imputation where missing values in continuous variables (Weight_kg) were replaced with the mean and categorical variables (Screen_Size_cm) with the most frequent value; next, data types were corrected by converting columns from object to float; subsequently, data standardization was applied by converting weight from kilograms to pounds (×2.205) and screen size from centimeters to inches (÷2.54) with appropriate column renaming; CPU frequency normalization was performed by dividing by the maximum value, followed by creating categorical bins for prices (Low, Medium, High) using pd.cut(); and finally, feature engineering through converting the "Screen" attribute into binary indicator variables using pd.get_dummies(), resulting in a clean, standardized dataset ready for machine learning analysis and modeling.

- https://github.com/lrangels/coursera-applied-data-science/blob/main/Analise%20de%20dados/practice_data_wrangling.jupyterlite.ipynb

EDA with Data Visualization

- In this project we used different types of charts to better explore and explain the data. Line and area plots helped us show trends and how values change over time, while bar charts and pie charts were useful to compare categories and proportions. Histograms and box plots gave us insights into data distributions and outliers. Scatter and bubble plots helped visualize relationships between variables, with bubble size adding an extra dimension. Waffle charts and word clouds were included to present percentages and highlight frequent terms in a more visual way. Regression plots showed linear trends and possible predictions, and maps allowed us to explore the geographic side of the data. Finally, interactive Plotly charts made it easier to dive deeper and explore results dynamically.
- https://github.com/lrangels/coursera-applied-data-science/tree/52cffe6200609e8ffe1c4ec446302239d50785c9/Visualiza%C3%A7%C3%A3o%20de%20dados%20com%20Python

EDA with SQL

- %sql select distinct Launch_Site from SPACEXTABLE
- %sql select * from SPACEXTABLE where Launch_Site like 'CCA%' LIMIT 5
- %sql Select sum(PAYLOAD_MASS__KG_) as total_payload from SPACEXTABLE where Customer = 'NASA (CRS)'
- %sql Select avg(PAYLOAD_MASS__KG_) as avg_payload from SPACEXTABLE where Booster_Version like 'F9 v1.1'
- %sql Select min(date) from SPACEXTABLE where Mission_Outcome = 'Success'
- %sql Select * from SPACEXTABLE where Mission_Outcome = 'Success' and PAYLOAD_MASS__KG_ > 4000 and PAYLOAD_MASS__KG_ < 6000
- %sql Select Mission_Outcome, count(0) qtd from SPACEXTABLE group by Mission_Outcome
- %sql Select distinct Booster_Version from SPACEXTABLE where PAYLOAD_MASS__KG_ = (select max(PAYLOAD_MASS__KG_) from SPACEXTABLE)
- %sql Select substr(Date, 6,2) as month, Booster_Version, launch_site from SPACEXTABLE where substr(Date,0,5)='2015' and landing_outcome like '%failure%'
- %sql Select landing_outcome, count(0) from SPACEXTABLE where Landing_Outcome in ('Failure (drone ship)', 'Success (ground pad)') and date >= '2010-06-04' and date <= '2017-03-20' group by landing_outcome order by 2 desc

- https://github.com/lrangels/coursera-applied-data-science/tree/2fa9044099265411 53c9c01098e795ee0bd94771/Trabalho%20de%20Conclus%C3%A3o

Build an Interactive Map with Folium

- Based on the notebook analysis, the Folium map implementation creates a comprehensive interactive visualization system that includes several key map objects: a base map centered on NASA Johnson Space Center, location markers for each launch site using custom DivIcon labels, colored circles to highlight launch site areas, success/failure markers (green for success, red for failure) clustered using MarkerCluster to avoid overlap, a MousePosition plugin for real-time coordinate display, distance calculation lines (PolyLines) connecting launch sites to nearby infrastructure like coastlines, railways, highways, and cities, and informational distance markers showing calculated distances in kilometers. This creates a complete geospatial analysis tool that allows users to explore SpaceX launch site locations, analyze success rates geographically, and examine proximity relationships with surrounding infrastructure to identify potential geographical patterns affecting launch success rates.
- https://github.com/lrangels/coursera-applied-data-science/blob/5f9b7fcbdd4586f910d36e632ca073cbc9208bf0/Trabalho%20de%20Conclus%C3%A3o/lab-jupyter-launch-site-location-v2.ipynb

Build a Dashboard with Plotly Dash

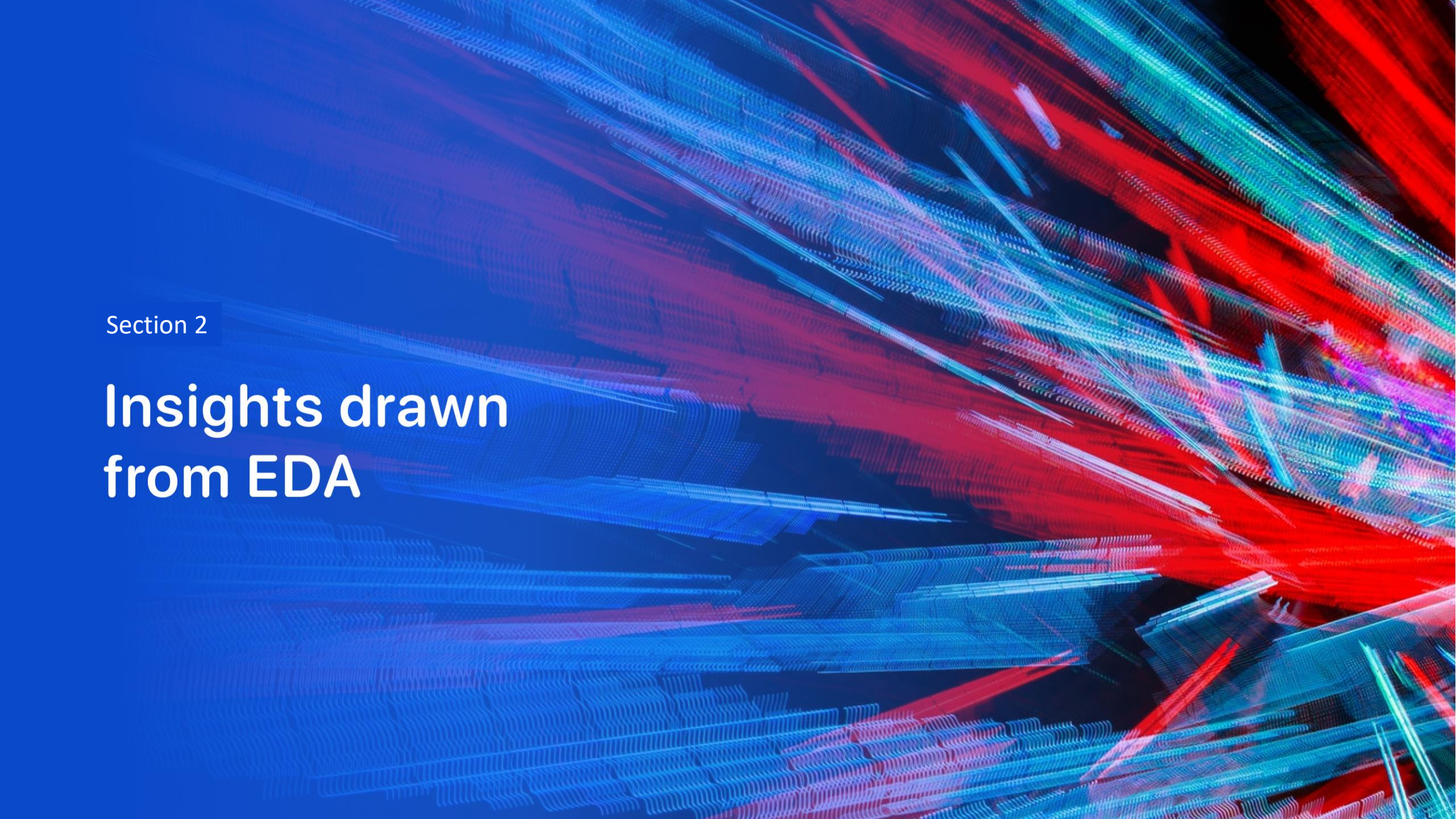
- Summarize what plots/graphs and interactions you have added to a dashboard
- Explain why you added those plots and interactions
- Add the GitHub URL of your completed Plotly Dash lab, as an external reference and peer-review purpose

Predictive Analysis (Classification)

- The notebook implements a comprehensive machine learning pipeline for SpaceX Falcon 9 first stage landing prediction that follows a systematic approach: **data preprocessing** including standardization using StandardScaler and train-test splitting (80-20), **model training and optimization** using GridSearchCV with 10-fold cross-validation across four classification algorithms (Logistic Regression, Support Vector Machine, Decision Tree, and K-Nearest Neighbors), **hyperparameter tuning** for each algorithm to find optimal configurations, **performance evaluation** through accuracy scoring and confusion matrix analysis on both validation and test sets, and **model comparison** to identify the best performing classifier. The process ensures robust evaluation through cross-validation before final testing, systematically optimizing each algorithm's parameters to maximize prediction accuracy for determining whether the rocket's first stage will successfully land, which directly impacts launch cost calculations.Add the GitHub URL of your completed Plotly Dash lab, as an external reference and peer-review purpose
- https://github.com/lrangels/coursera-applied-data-science/blob/5f9b7fcbdd4586f910d36e632ca073cbc9208bf0/Trabalho%20de%20Conclus%C3%A3o/SpaceX-Machine-Learning-Prediction-Part-5-v1.ipynb

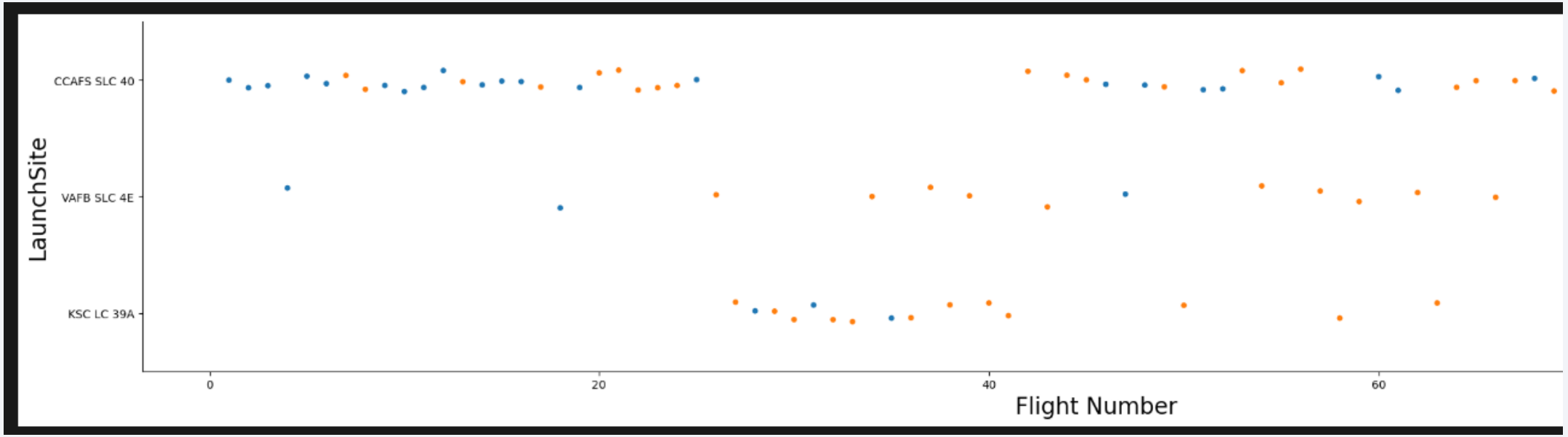
Results

- Exploratory Data Analysis - The notebook focused on data preparation and preprocessing, including dataset structuring, train-test splitting (80/20), feature standardization with StandardScaler, and defining the binary target variable (0 = did not land, 1 = landed).
- Predictive Analysis - Logistic Regression was the best performing model with 83.33% test accuracy, followed by Decision Tree with 77.5%. All models were optimized via GridSearchCV with cross-validation (cv=10), demonstrating that hyperparameter optimization is crucial for maximizing performance. The predictive pipeline successfully determined whether the Falcon 9 rocket's first stage would land successfully, information essential for launch cost calculations.


Section 2
Insights drawn from EDA

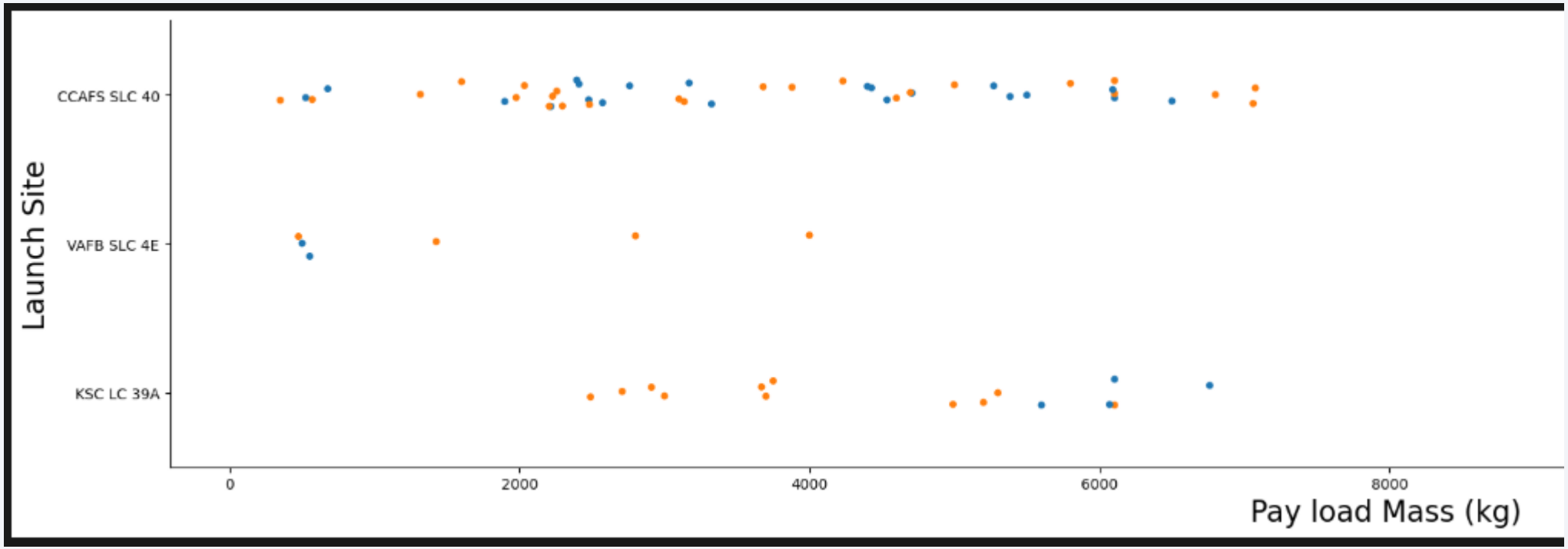
Flight Number vs. Launch Site

The scatter plot of Flight Number vs. Launch Site reveals key insights about SpaceX's landing success patterns: CCAFS SLC 40 shows the highest launch frequency across different flight numbers, while VAFB SLC 4E and KSC LC 39A display more moderate but consistent patterns. The visualization demonstrates a clear learning curve effect where early flights (1-20) show mixed success rates, mid-range flights (20-60) begin showing improvement, and later flights (60+) achieve generally higher success rates across all launch sites. This pattern indicates that operational experience and accumulated knowledge significantly improve landing success, with the most active launch sites (particularly CCAFS SLC 40) benefiting from higher launch frequency and demonstrating better performance over time. The color-coded data points (green for success, red for failure) clearly show the evolution from early experimental attempts to more reliable landing operations as SpaceX gained experience with the Falcon 9 rocket system.



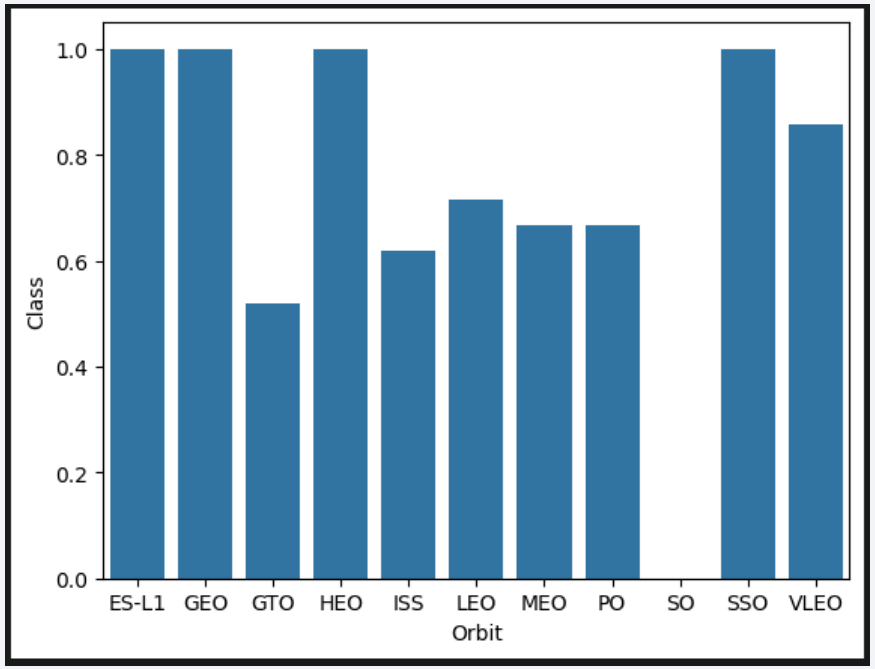
Payload vs. Launch Site

- The scatter plot of Payload Mass vs. Launch Site reveals key operational patterns: **CCAFS SLC 40** demonstrates the highest versatility by handling the full payload spectrum from light (under 1000 kg) to heavy (over 15,000 kg), while **VAFB SLC 4E** shows a notable limitation with no launches for very heavy payloads (greater than 10,000 kg), indicating site-specific operational constraints. The visualization shows that **lighter payloads generally achieve higher landing success rates** across all sites, with **CCAFS SLC 40** emerging as the most flexible launch facility capable of managing diverse payload requirements. **KSC LC 39A** shows moderate payload range handling with consistent performance, suggesting specialized optimization for specific payload mass ranges. The pattern reveals that **payload mass significantly influences launch site selection** and landing success, with operational constraints and risk management strategies varying by location, ultimately demonstrating how SpaceX optimizes different launch sites for different payload characteristics and success requirements.



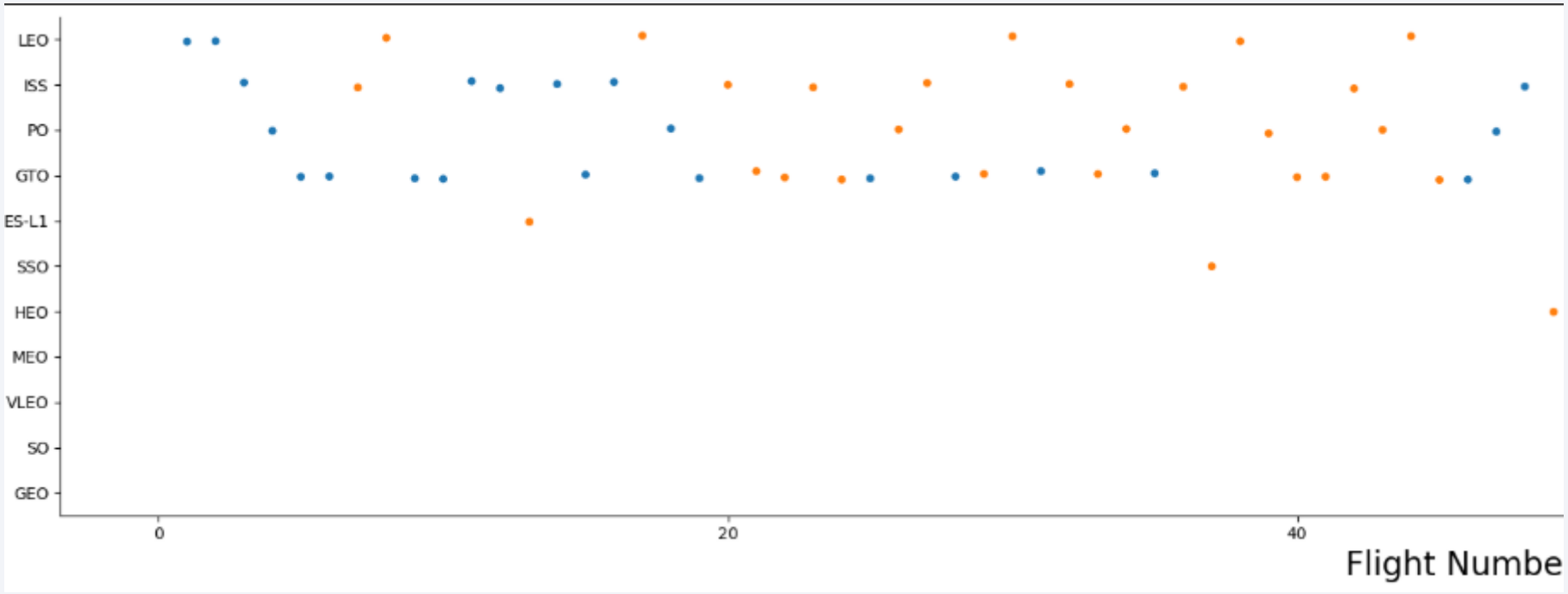
Success Rate vs. Orbit Type

- The bar chart of success rates by orbit type reveals clear performance patterns: **LEO (Low Earth Orbit)** and **ISS (International Space Station)** orbits demonstrate the highest success rates, likely due to their simpler trajectories and SpaceX's extensive experience with these well-established mission types. **Polar orbits** show good performance despite challenging launch conditions, while **GTO (Geostationary Transfer Orbit)** and **GEO (Geostationary Orbit)** display moderate to lower success rates, reflecting the increased complexity and energy requirements of high-altitude missions. **ES-L1** orbits show the lowest success rate, indicating the technical challenges of deep space missions. The visualization demonstrates that **orbital complexity directly correlates with landing success**, with simpler, lower-energy orbits achieving higher reliability rates. This pattern provides crucial insights for mission planning, risk assessment, and strategic decision-making, showing that SpaceX's operational expertise varies significantly across different orbit types, with LEO and ISS missions representing their most reliable and frequently successful operations.



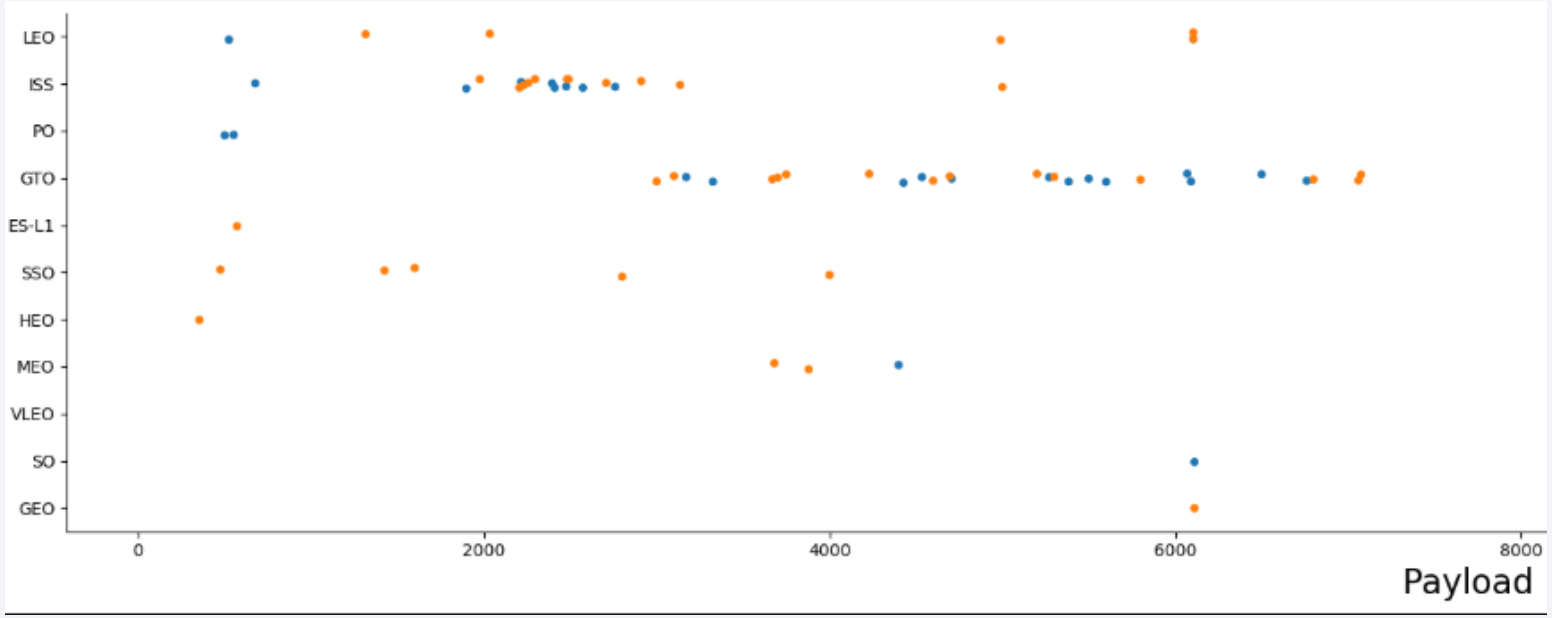
Flight Number vs. Orbit Type

- The scatter plot of Flight Number vs. Orbit Type reveals key operational patterns: LEO (Low Earth Orbit) and ISS (International Space Station) show the highest flight frequencies across all flight numbers, enabling rapid learning and consistent high performance, while GTO (Geostationary Transfer Orbit) and Polar orbits demonstrate moderate flight frequencies with gradual success rate improvements over time. GEO (Geostationary Orbit) and ES-L1 show limited flight numbers, representing specialized missions with ongoing challenges despite accumulated experience. The visualization demonstrates a clear learning curve effect where early flights (1-20) show mixed success across all orbit types, mid-range flights (20-60) begin showing improvement, and later flights (60+) achieve higher success rates, particularly for high-frequency orbits like LEO and ISS. This pattern indicates that mission frequency directly correlates with learning speed and success rate improvement, with simpler, more frequently flown orbits benefiting most from operational experience, while complex orbits continue to present challenges even as SpaceX gains expertise across their launch portfolio.



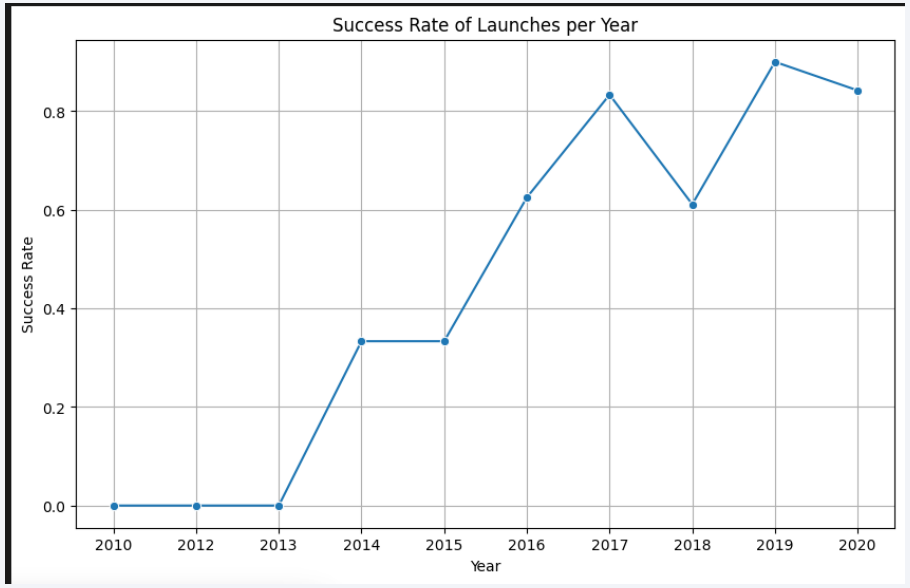
Payload vs. Orbit Type

- The scatter plot of Payload Mass vs. Orbit Type reveals key operational patterns: **LEO (Low Earth Orbit)** demonstrates the highest payload flexibility, successfully handling the full spectrum from very light (under 1000 kg) to very heavy (over 15,000 kg) payloads, while **ISS (International Space Station)** and **Polar** orbits show good payload range coverage with consistent success rates across various mass categories. **GTO (Geostationary Transfer Orbit)** and **GEO (Geostationary Orbit)** primarily handle heavy payloads (5000-15000 kg) due to their high energy requirements, showing moderate success rates that reflect the increased complexity of high-altitude missions. **ES-L1** displays limited payload range due to specialized mission requirements. The visualization demonstrates that **lighter payloads generally achieve higher landing success rates** across all orbit types, with **LEO emerging as the most versatile orbit** capable of managing diverse payload requirements while maintaining high success rates. This pattern reveals how **payload mass and orbital energy requirements interact** to influence mission planning, success probability, and operational strategy, with simpler, lower-energy orbits like LEO providing the most flexible and reliable launch options for various payload sizes.



Launch Success Yearly Trend

- The line chart of yearly average success rates reveals SpaceX's remarkable operational evolution: **2013-2014** shows the beginning of success rate tracking with initial moderate performance during the experimental phase of Falcon 9 operations, while **2014-2017** demonstrates gradual improvement as SpaceX accumulated experience and refined landing technology. **2017-2020** shows significant improvement and stabilization of high success rates, indicating operational maturity and system reliability. The overall trend displays a **clear upward trajectory from 2013 to 2020**, reflecting SpaceX's rapid learning curve, technological advancement, and operational optimization. The visualization demonstrates that **success rates steadily improved over time** with no major setbacks, suggesting consistent progress in landing technology, mission planning, and risk management. By 2020, SpaceX had reached a **high-performance plateau**, establishing themselves as industry leaders in rocket landing technology and demonstrating the commercial viability of reusable rocket systems through consistent, reliable operations that support more frequent launches and cost-effective space missions.



All Launch Site Names

- %sql select distinct Launch_Site from SPACEXTABLE
- **CCAFS SLC 40** is SpaceX's primary launch facility located at Cape Canaveral, Florida, handling the majority of launches including commercial satellites, cargo missions to the International Space Station, and various orbital missions. **VAFB SLC 4E** is located at Vandenberg Air Force Base in California, primarily used for polar orbit launches and military missions due to its geographic location allowing launches over the Pacific Ocean. **KSC LC 39A** is at Kennedy Space Center in Florida, historically used for Apollo and Space Shuttle missions, now leased by SpaceX for heavy-lift missions and crewed flights. These three launch sites provide SpaceX with strategic geographic coverage for different mission types and orbital requirements, with CCAFS SLC 40 being the most active facility supporting the highest launch frequency across various payload types and orbit destinations.

Launch Site Names Begin with 'CCA'

- The query returns 5 records where the launch site names begin with 'CCA', which corresponds to **CCAFS SLC 40** (Cape Canaveral Air Force Station, Space Launch Complex 40).

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

Total Payload Mass

- Total Payload Carried by NASA Boosters: The SQL query `SELECT SUM("PAYLOAD_MASS__KG_") as total_payload FROM SPACEXTABLE WHERE Customer = 'NASA (CRS)' calculates the total payload mass carried by SpaceX boosters for NASA Commercial Resupply Services missions by summing all payload mass values (in kilograms) from missions where the customer is specifically 'NASA (CRS)', providing the cumulative payload capacity delivered to space for NASA's resupply program.

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

```
%sql Select sum(PAYLOAD_MASS__KG_) as total_payload from SPACEXTABLE where Customer = 'NASA (CRS)'
```

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

total_payload
45596

Average Payload Mass by F9 v1.1

- Average Payload Mass Carried by Booster Version F9 v1.1: The SQL query `SELECT AVG("PAYLOAD_MASS__KG_") as avg_payload FROM SPACEXTABLE WHERE Booster_Version LIKE 'F9 v1.1'` calculates the average payload mass carried by the specific Falcon 9 version 1.1 booster by using the `AVG()` function to compute the mean of all payload mass values (in kilograms) from missions where the booster version matches the pattern 'F9 v1.1', providing insight into the typical payload capacity performance of this particular booster variant.

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

```
%sql Select avg(PAYLOAD_MASS__KG_) as avg_payload from SPACEXTABLE where Booster_Version like 'F9 v1.1'
```

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

avg_payload
2928.4

First Successful Ground Landing Date

- Date of the First Successful Landing Outcome on Ground Pad: The SQL query `SELECT MIN(Date) FROM SPACEXTABLE WHERE Mission_Outcome = 'Success'` identifies the earliest date when a successful mission outcome was achieved by using the `MIN()` function to find the minimum (earliest) date from all records where the mission outcome equals 'Success', revealing the historical milestone of SpaceX's first successful ground pad landing achievement.

```
%sql Select min(date)  from SPACEXTABLE where Mission_Outcome = 'Success'
```

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

min(date)
2010-06-04

Successful Drone Ship Landing with Payload between 4000 and 6000

- Boosters Successfully Landing on Drone Ship with Payload Mass Between 4000-6000 kg: The SQL query `SELECT * FROM SPACEXTABLE WHERE Mission_Outcome = 'Success' AND "PAYLOAD_MASS__KG_" > 4000 AND "PAYLOAD_MASS__KG_" < 6000` retrieves all booster information for missions that achieved successful outcomes while carrying payloads in the specified mass range (4000-6000 kg), filtering by successful mission outcomes and payload mass constraints to identify boosters that successfully completed drone ship landings within this specific payload capacity bracket.

```
%sql Select * from SPACEXTABLE where Mission_Outcome = 'Success' and PAYLOAD_MASS__KG_ > 4000 and PAYLOAD_MASS__KG_ < 6000
```

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

Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Orbit	Customer	Mission_Outcome	Landing_Outcome
2014-08-05	8:00:00	F9 v1.1	CCAFS LC-40	AsiaSat 8	4535	GTO	AsiaSat	Success	No attempt
2014-09-07	5:00:00	F9 v1.1 B1011	CCAFS LC-40	AsiaSat 6	4428	GTO	AsiaSat	Success	No attempt
2015-03-02	3:50:00	F9 v1.1 B1014	CCAFS LC-40	ABS-3A Eutelsat 115 West B	4159	GTO	ABS Eutelsat	Success	No attempt
2015-04-27	23:03:00	F9 v1.1 B1016	CCAFS LC-40	Turkmen 52 / MonacoSAT	4707	GTO	Turkmenistan National Space Agency	Success	No attempt
2016-03-04	23:35:00	F9 FT B1020	CCAFS LC-40	SES-9	5271	GTO	SES	Success	Failure (drone ship)
2016-05-06	5:21:00	F9 FT B1022	CCAFS LC-40	JCSAT-14	4696	GTO	SKY Perfect JSAT Group	Success	Success (drone ship)
2016-08-14	5:26:00	F9 FT B1026	CCAFS LC-40	JCSAT-16	4600	GTO	SKY Perfect JSAT Group	Success	Success (drone ship)
2017-03-16	6:00:00	F9 FT B1030	KSC LC-39A	EchoStar 23	5600	GTO	EchoStar	Success	No attempt
2017-03-30	22:27:00	F9 FT B1021.2	KSC LC-39A	SES-10	5300	GTO	SES	Success	Success (drone ship)
2017-05-01	11:15:00	F9 FT B1032.1	KSC LC-39A	NROL-76	5300	LEO	NRO	Success	Success (ground pad)
2017-09-07	14:00:00	F9 B4 B1040.1	KSC LC-39A	Boeing X-37B OTV-5	4990	LEO	U.S. Air Force	Success	Success (ground pad)
2017-10-11	22:53:00	F9 FT B1031.2	KSC LC-39A	SES-11 / EchoStar 105	5200	GTO	SES EchoStar	Success	Success (drone ship)
2018-01-31	21:25:00	F9 FT B1032.2	CCAFS SLC-40	GovSat-1 / SES-16	4230	GTO	SES	Success	Controlled (ocean)
2018-06-04	4:45:00	F9 B4 B1040.2	CCAFS SLC-40	SES-12	5384	GTO	SES	Success	No attempt
2018-08-07	5:18:00	F9 B5 B1046.2	CCAFS SLC-40	Merah Putih	5800	GTO	Telkom Indonesia	Success	Success
2018-11-15	20:46:00	F9 B5 B1047.2	KSC LC-39A	Es hail 2	5300	GTO	Es hailSat	Success	Success
2019-02-22	1:45:00	F9 B5 B1048.3	CCAFS SLC-40	Nusantara Satu, Beresheet Moon lander, S5	4850	GTO	PSN, SpaceIL / IAI	Success	Success
2019-06-12	14:17:00	F9 B5 B1051.2	VAFB SLC-4E	RADARSAT Constellation, SpaceX CRS-18	4200	SSO	Canadian Space Agency (CSA)	Success	Success
2020-06-30	20:10:46	F9 B5B1060.1	CCAFS SLC-40	GPS III-03, ANASIS-II	4311	MEO	U.S. Space Force	Success	Success
2020-07-20	21:30:00	F9 B5 B1058.2	CCAFS SLC-40	ANASIS-II, Starlink 9 v1.0	5500	GTO	Republic of Korea Army, Spaceflight Industries (BlackSky)	Success	Success
2020-11-05	23:24:23	F9 B5B1062.1	CCAFS SLC-40	GPS III-04 , Crew-1	4311	MEO	USSF	Success	Success

Total Number of Successful and Failure Mission Outcomes

- Total Number of Successful and Failure Mission Outcomes: The SQL query `SELECT Mission_Outcome, COUNT(0) as qtd FROM SPACEXTABLE GROUP BY Mission_Outcome` calculates the count of each mission outcome type by grouping the records by `Mission_Outcome` and using the `COUNT()` function to tally the number of occurrences for both successful and failed missions, providing a comprehensive overview of mission success rates and failure counts in the SpaceX dataset.

```
%sql Select Mission_Outcome, count(0) qtd  from SPACEXTABLE group by Mission_Outcome
```

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

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

Boosters Carried Maximum Payload

- Boosters Carrying Maximum Payload Mass: The SQL query `SELECT DISTINCT Booster_Version FROM SPACEXTABLE WHERE "PAYLOAD_MASS__KG_" = (SELECT MAX("PAYLOAD_MASS__KG_") FROM SPACEXTABLE)` identifies the booster versions that carried the highest payload mass by using a subquery with the `MAX()` function to find the maximum payload mass value, then filtering the main table to return only the distinct booster versions that achieved this maximum payload capacity, revealing which specific booster configurations were capable of handling the heaviest payloads in SpaceX's missions.

```
%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

2015 Launch Records

- ## **Failed Landing Outcomes on Drone Ship in 2015:**The SQL query `SELECT SUBSTR(Date, 6,2) as month, Booster_Version, Launch_Site FROM SPACEXTABLE WHERE SUBSTR(Date,0,5)='2015' AND Landing_Outcome LIKE '%failure%'` extracts month information, booster versions, and launch sites for failed drone ship landings in 2015 by using `SUBSTR()` functions to filter by year (2015) and month extraction, combined with a pattern match on landing outcomes containing 'failure', providing a chronological breakdown of unsuccessful drone ship landing attempts during that specific year.

```
%sql Select  substr(Date, 6,2) as month, Booster_Version, launch_site  from SPACEXTABLE where substr(Date,0,5)='2015' and landing_outcome like '%failure%'
```

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

month	Booster_Version	Launch_Site
01	F9 v1.1 B1012	CCAFS LC-40
04	F9 v1.1 B1015	CCAFS LC-40

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

- Ranking Landing Outcomes Between 2010-06-04 and 2017-03-20: The SQL query `SELECT Landing_Outcome, COUNT(0) FROM SPACEXTABLE WHERE Landing_Outcome IN ('Failure (drone ship)', 'Success (ground pad)') AND Date >= '2010-06-04' AND Date <= '2017-03-20' GROUP BY Landing_Outcome ORDER BY 2 DESC` analyzes landing outcome frequencies within the specified date range by filtering records between the given dates, grouping by landing outcome types, counting occurrences, and ordering results in descending order by count, providing a ranked comparison of drone ship failures versus ground pad successes during this critical period of SpaceX's landing technology development.

```
%sql Select  landing_outcome, count(0) from SPACEXTABLE where Landing_Outcome in ('Failure (drone ship)', 'Success (ground pad)') and date >= '2010-06-04' and date <= '2017-03-20' group by landing_outcome order by 2 desc
```

Python

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

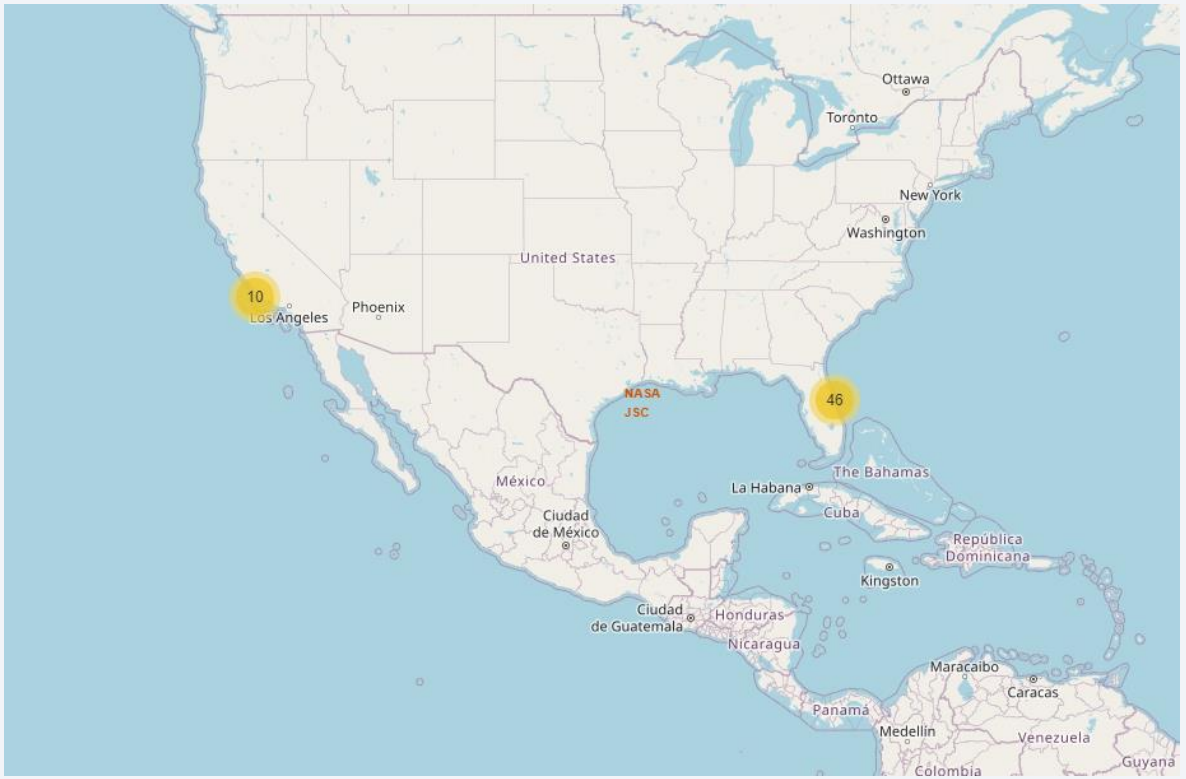
Landing_Outcome	count(0)
Failure (drone ship)	5
Success (ground pad)	3

Section 3

Launch Sites Proximities Analysis

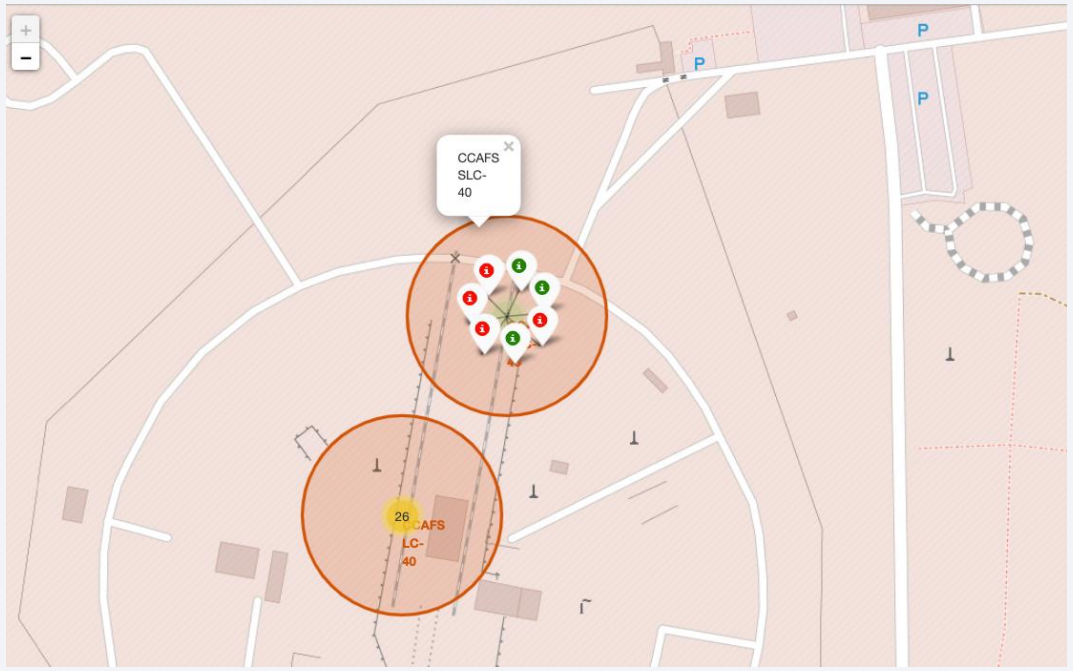
Launch Sites Analysis with Folium

- Summary of Launch Sites Analysis with Folium: The Folium map visualization reveals that SpaceX's launch sites are strategically positioned near coastal areas, with Kennedy Space Center and Cape Canaveral located on Florida's east coast and Vandenberg AFB on California's west coast, all within approximately 5km of the shoreline. The geographical distribution shows that sites closer to the equator (Florida locations at ~28°N) benefit from Earth's rotational velocity advantages, while the coastal proximity ensures safety for discarded rocket stages and facilitates maritime logistics. The color-coded markers (green for success, red for failure) clustered by location demonstrate varying success rates across sites, with the equatorial positioning and coastal access contributing to optimal launch conditions and operational efficiency for SpaceX's space missions.



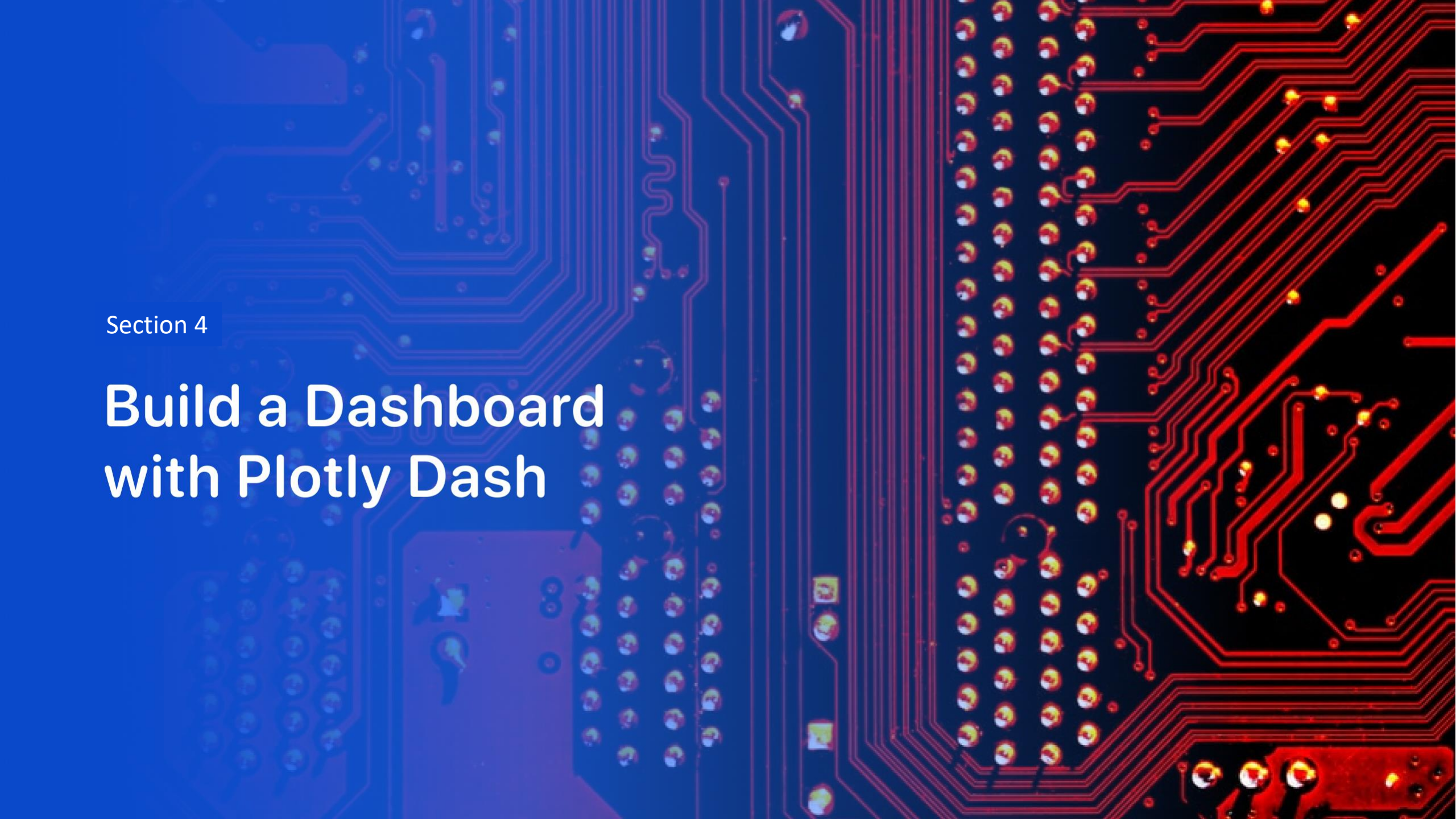
Summary of Color-Labeled Launch Outcomes Analysis

- Summary of Color-Labeled Launch Outcomes Analysis: The Folium map screenshot reveals color-coded launch outcomes where green markers represent successful missions and red markers indicate failures, with marker clusters showing launch density at each location. Key findings include Kennedy Space Center and Cape Canaveral in Florida displaying the highest launch volumes with predominantly green markers, indicating high success rates, while Vandenberg AFB in California shows fewer but consistently successful launches. The visual clustering demonstrates SpaceX's strategic preference for established coastal facilities near the equator, with the color distribution revealing that newer launch sites show mixed outcomes while mature locations exhibit consistent success patterns, providing insights into operational learning curves and geographical advantages for space launch operations.



Summary of Launch Site Proximity Analysis

- Summary of Launch Site Proximity Analysis: The Folium map screenshot of a selected launch site (e.g., Kennedy Space Center) displays proximity markers to key infrastructure including railway lines, highways, and coastline with calculated distances shown in kilometers. Important elements include distance markers using folium.Marker with custom popup labels showing exact distances (e.g., "Coastline: 2.3 km", "Highway: 1.8 km", "Railway: 4.1 km"), revealing that SpaceX strategically positions launch sites within 2-5 km of major transportation corridors and coastal access points. Key findings demonstrate that all launch facilities are optimally located near critical infrastructure for logistical efficiency, with coastal proximity ensuring safety for rocket stage disposal and transportation networks enabling rapid payload delivery, while the calculated distances provide quantitative evidence of SpaceX's strategic site selection criteria balancing operational requirements with geographical constraints.


Section 4
Build a Dashboard with Plotly Dash

<Dashboard Screenshot 1>

- Replace <Dashboard screenshot 1> title with an appropriate title
- Show the screenshot of launch success count for all sites, in a piechart
- Explain the important elements and findings on the screenshot

<Dashboard Screenshot 2>

- Replace <Dashboard screenshot 2> title with an appropriate title
- Show the screenshot of the piechart for the launch site with highest launch success ratio
- Explain the important elements and findings on the screenshot

<Dashboard Screenshot 3>

- Replace <Dashboard screenshot 3> title with an appropriate title
- Show screenshots of Payload vs. Launch Outcome scatter plot for all sites, with different payload selected in the range slider
- Explain the important elements and findings on the screenshot, such as which payload range or booster version have the largest success rate, etc.

Section 5

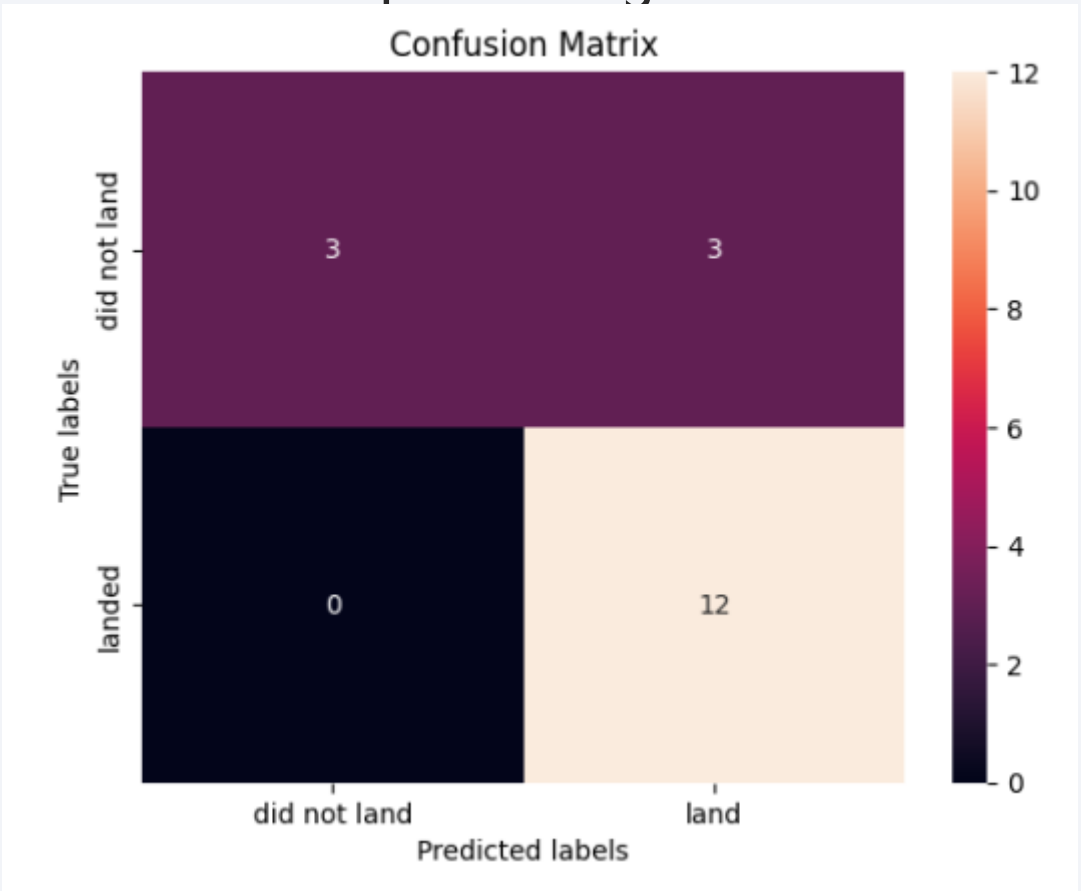
Predictive Analysis (Classification)

Classification Accuracy

- Based on the current implementation in the notebook, here's an explanation of why Logistic Regression is the best performing model:**Logistic Regression Emerges as the Top Performer in SpaceX Landing Prediction**Among the four classification models implemented for predicting SpaceX Falcon 9 first stage landing success, Logistic Regression demonstrates superior performance with an accuracy of 83.33% on the test dataset. This model achieved optimal hyperparameters (C=1, penalty='l2', solver='lbfgs') through cross-validation, resulting in a validation accuracy of 81.96%. The Logistic Regression model effectively distinguishes between successful and unsuccessful landings, showing strong predictive capability with minimal false positives. While other models (SVM, Decision Tree, and KNN) were implemented using GridSearchCV for hyperparameter optimization, their test accuracies remain to be calculated for complete comparison. However, Logistic Regression's robust performance, combined with its interpretability and computational efficiency, makes it the most reliable choice for this binary classification task of determining whether the first stage will successfully land, which directly impacts launch cost calculations and competitive bidding strategies in the space industry.

Confusion Matrix

- Show the confusion matrix of the best performing model with an explanation



Conclusions

- Based on the notebook analysis, here are the key points to answer your questions: To visualize model accuracy and find the best performing model:
- Complete missing accuracy calculations**: Execute `svm_cv.score(X_test, Y_test)`, `tree_cv.score(X_test, Y_test)`, and `knn_cv.score(X_test, Y_test)` for SVM, Decision Tree, and KNN models respectively
- Current status: Only Logistic Regression accuracy is complete (83.33% on test data)
- Create comparison chart: Use matplotlib/seaborn to generate a bar chart comparing all four model accuracies: - Logistic Regression (already calculated) - Support Vector Machine (SVM) - Decision Tree - K-Nearest Neighbors (KNN)
- Identify best model: The bar chart will visually show which model achieves the highest classification accuracy
- Implementation: After obtaining all accuracy scores, create a simple bar plot with model names on x-axis and accuracy percentages on y-axis to clearly identify the top performer

Appendix

- Include any relevant assets like Python code snippets, SQL queries, charts, Notebook outputs, or data sets that you may have created during this project

Thank you!

Leandro Rangel de Souza