



IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Debra Elkins
October 14, 2025



Outline

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

Executive Summary

REPORT OVERVIEW: This report summarizes the Coursera / IBM Applied Data Science Capstone Project.

CAPSTONE PROJECT & VALUE: This project seeks to **predict if SpaceX's Falcon 9 spacecraft's first stage will land successfully after launch**. The results can be used to evaluate a SpaceX cost proposal versus other providers. SpaceX estimates its cost per launch of \$62 M dollars (based on its approach for landing and reusing the first stage of its spacecraft), whereas other providers charge a cost per launch of \$165+ M dollars.

METHODOLOGIES USED: **Visual data analysis** and four classifier models (**Logistic Regression, Decision Tree, Support Vector Machine (SVM), and K-Nearest Neighbor**) are used to classify if a launch will result in a successful first stage landing after launch.

RESULTS:

- Data Wrangling showed **1st Stage Landing Success Rate of 67%** (60/90), based on prior launch data
- Visual analysis showed SpaceX demonstrating **“organizational learning”** as it increased its launch number. Notably, SpaceX showed improvement in successfully landing the 1st stage of the rocket at different launch sites, for different orbits, for different payload masses (kg), and year over year
- Folium geospatial mapping shows **Kennedy Space Center LC-39A** achieved $10/13 = 76.9\%$ **successful 1st stage landings**
- **Predictive Modeling** yielded 4 classifier models (**Logistic Regression, Decision Tree, Support Vector Machine (SVM), and K-Nearest Neighbor**) with **83.3% accuracy** (i.e., all 4 models make mostly correct predictions regarding 1st stage landing success or failure) and **100% recall** (i.e., all 4 models were able to identify all successful 1st stage landings, with no “misses”)
- The **Logistic Regression model** can also be used **to predict the probability of SpaceX successfully landing the 1st phase of its spacecraft given launch input parameters** such as launch site, orbit, payload mass, booster variant, etc. For the best Logistic Regression model, we estimate the **probability of successfully landing the 1st phase of its spacecraft at 67.2%**, using all available launch data

CONCLUSION: **SpaceX can deliver launch capabilities at the cost of \$62 M per launch, because they can successfully land and reuse the first stage of its spacecraft.** Other providers should NOT bid, unless they can demonstrate a comparable cost per launch.

Introduction

PROJECT BACKGROUND AND CONTEXT:

- Multiple companies offer space launch-related services, including SpaceX, Virgin Galactic, Rocket Lab, and Blue Origin.
- Space launches are used by companies to:
 - Continue exploring space travel
 - Deploy satellites
 - Deliver materials via spacecraft (e.g., deliveries to the International Space Station)
- SpaceX estimates its cost per launch of \$62 M dollars (based on its approach for first stage landing and reuse), whereas other providers charge a cost per launch of \$165+ M. dollars

PROBLEM STATEMENT:

We want to estimate the probability of SpaceX's successfully landing its first stage. This result will permit us to evaluate if SpaceX can successfully deliver launch capabilities at a significantly lower cost compared to other potential providers.

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology
- Data wrangling (i.e., cleaning and preparing data for analysis & decision-making)
- Exploratory data analysis (EDA) using visualization and SQL
- Interactive visual analytics using Folium and Plotly Dash
- Predictive modeling & analysis using four classification models
 - Logistic Regression Classifier
 - Support Vector Machine Classifier
 - Decision Tree Classifier
 - K Nearest Neighbor Classifier

Data Collection

- **APPROACH:** Develop Jupyter Labs Notebook and Python Code to acquire data from two data sources:
 1. SpaceX REST API data
 2. Web Scraping a SpaceX Wikipedia page
- **Dataset evaluation criteria:**
 1. Y (target variable) availability to assess first stage landing success
 2. Amount of complete launch observations (Y and X's availability) vs. missing data (and "fixes" required)
 3. Quality & quantity of Features (X variables) for predictive modeling

Potential Y (Target Var) for Predictive Modeling

[DATASET_PART_1.CSV](#)

- *Outcome* – results of the first stage landing (True=success or False=failure, also includes locations of landing such as Ocean, Ground Pad, Drone Ship)

[SPACEX_WEB_SCRAPED.CSV](#)

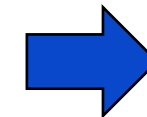
- *BoosterLanding* – categorical var for Success, Failure, etc.
- *LaunchOutcome* – Success, Failure or Blank

Potential X's (Features) for Predictive Modeling

- *Block* – number used to distinguish versions of the core
- *BoosterVersion* – booster used on the launch vehicle
- *Customer* – who is paying for the launch
- *Date* – date of launch
- *FlightNumber* – sequential count of flight launches
- *Flights* – number of flights with the specific core
- *GridFins* – were gridfins used on the launch vehicle
- *LandingPad* – was the landing pad used
- *Latitude* – longitude coordinate for the launch site
- *LaunchSite* – where the launch occurred
- *Legs* – were legs used on the launch vehicle
- *Longitude* – latitude coordinate for the launch site
- *Orbit* – categorical var for distance above Earth (e.g., LEO – 160 km to 2000 km)
- *Payload* – text description of launch delivery (e.g., satellite)
- *PayloadMass* – launch load in kg delivered to space
- *Reused* – was the core reused on the launch vehicle
- *ReusedCount* – number of times a specific core has been reused
- *Serial* – serial number of the core
- *Time* – time of the launch

Dataset Initial Preparation Notes

1. Filter data only to Falcon 9 launches
2. Deal with missing values in data for [DATASET_PART_1.CSV](#)
 - *Payload Mass* – replace 5 missing values with average payload mass of other similar launch observations
 - *LandingPad* – these 26 missing values are left blank, since the missing values indicate a LandingPad was NOT used
3. No missing data in [SPACEX_WEB_SCRAPED.CSV](#)
4. Store in intermediate datasets
 - SpaceX REST API data ([90 launch observations](#)) in [DATASET_PART_1.CSV](#)
 - Web Scraping a SpaceX Wikipedia page ([121 launch observations](#)) in [SPACEX_WEB_SCRAPED.CSV](#)

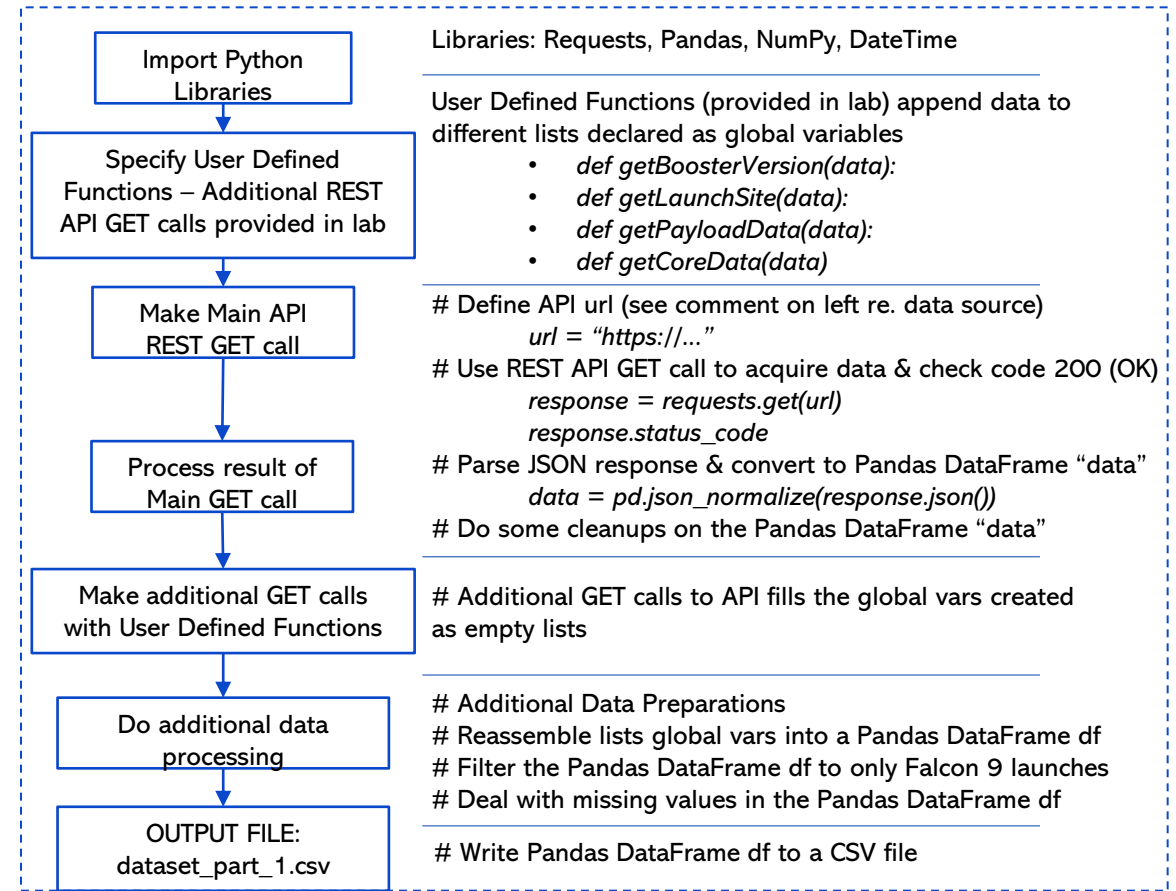


Dataset Decision

Use [SpaceX REST API data](#) (90 launches) in [DATASET_PART_1.CSV](#) because it has the Y (Target Variable) of interest (Outcome – results of the first stage landing)

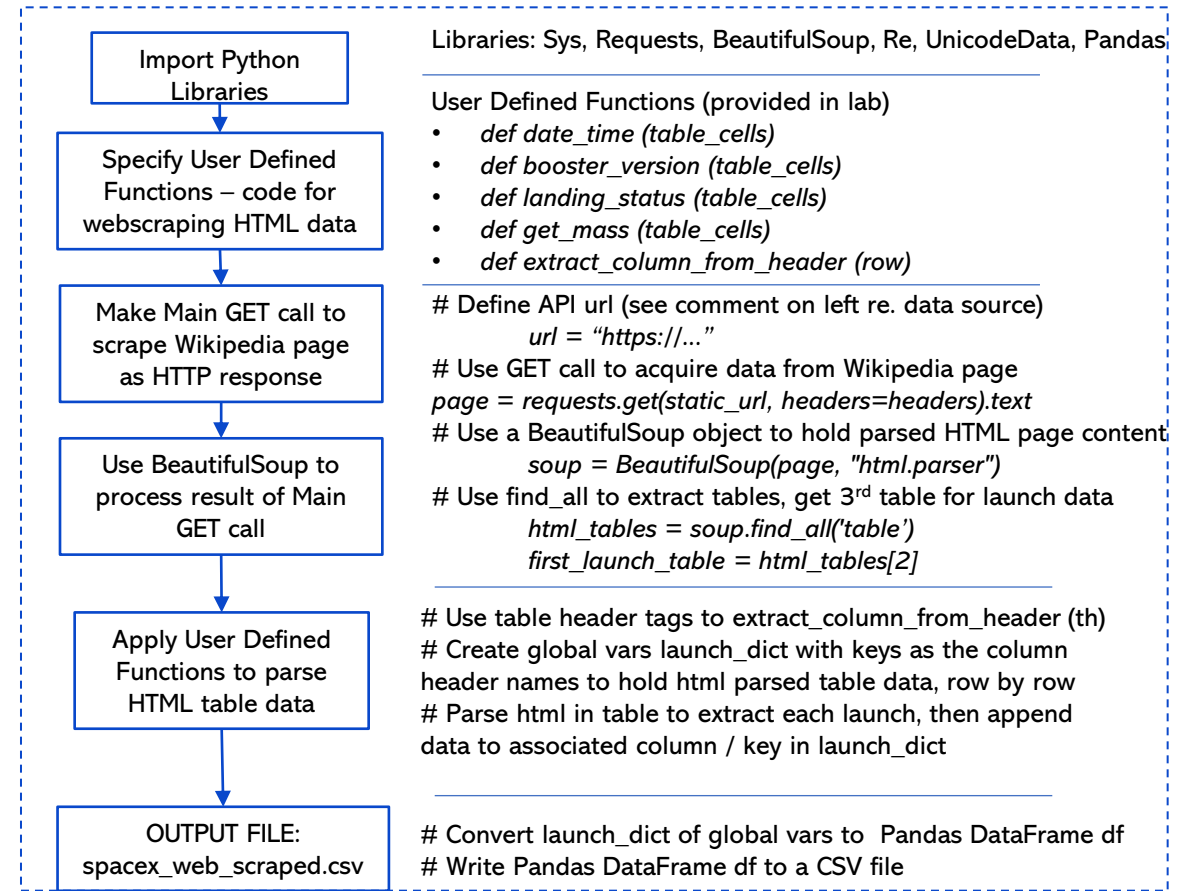
Data Collection – SpaceX API

- Here we outline how we used API REST GET requests to acquire SpaceX launch data
- **SpaceX API Data Source:**
<https://api.spacexdata.com/v4/launches/past>
- Note: For this course project, we used the **static API Data Source:** https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBM-DS0321EN-SkillsNetwork/datasets/API_call_spacex_api.json
- **GitHub URL of completed SpaceX API calls Jupyter Notebook:**
<https://github.com/DebraElkins/Coursera-DataScienceCapstoneProject/blob/main/jupyter-labs-spacex-data-collection-api.ipynb>



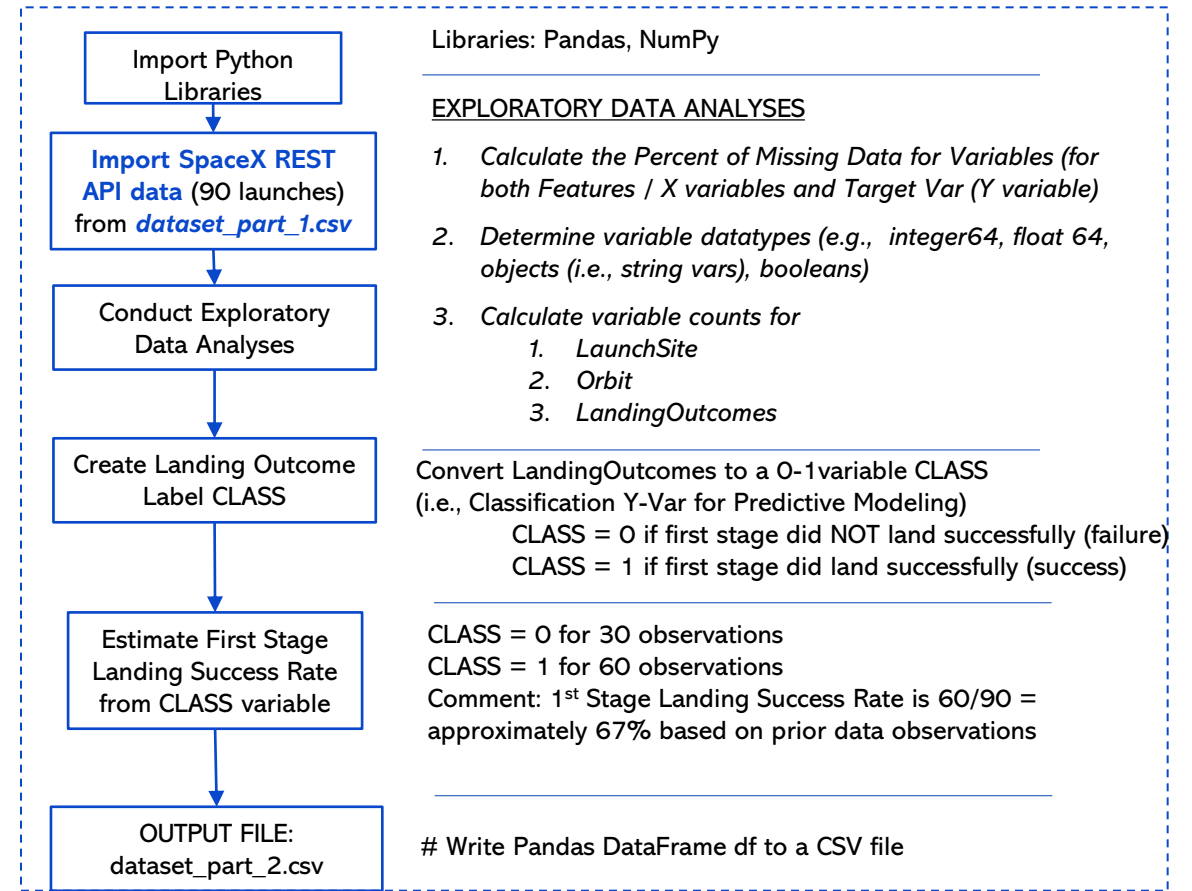
Data Collection - Scraping

- Here we outline how we used web scraping techniques to acquire SpaceX launch data
- **Wikipedia page for data web scraping:**
https://en.wikipedia.org/wiki/List_of_Falcon_9_and_Falcon_Heavy_launches
- Note: For this course project, we used the **static Wikipedia page for data web scraping:**
https://en.wikipedia.org/w/index.php?title=List_of_Falcon_9_and_Falcon_Heavy_launches&oldid=1027686922
- **GitHub URL of completed web scraping Jupyter Notebook:**
<https://github.com/DebraElkins/Coursera-DataScienceCapstoneProject/blob/main/jupyter-labs-webscraping%20.ipynb>



Data Wrangling

- Here we outline Data Wrangling performed to prepare data for visual analysis & predictive modeling
- Note: For this course project, we used the **static Space X dataset** https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBM-DS0321EN-SkillsNetwork/datasets/dataset_part_1.csv
- **GitHub URL of completed Data Wrangling Jupyter Notebook:**
<https://github.com/DebraElkins/Coursera-DataScienceCapstoneProject/blob/main/labs-jupyter-spacex-Data%20wrangling.ipynb>



Exploratory Data Analysis (EDA) with Data Visualization

- Used Jupyter Labs and Python libraries (Pandas, NumPy, Matplotlib-PyPlot, and Seaborn) to perform Exploratory Data Analysis (EDA) with Data Visualization using INPUT FILE: dataset_part_2.csv
- PART 1: Data Visualizations
 - Scatter Plot of Payload Mass versus Flight Number (x-axis) – to assess 1st stage landing success/failure as the payload mass and the flight number both increase
 - Scatter Plot of Launch Site (y-axis) versus Flight Number (x-axis) – to assess 1st stage landing success/failure as the flight number increases for different launch sites
 - Scatter Plot of Launch Site (y-axis) versus Payload Mass (x-axis) – to assess 1st stage landing success/failure for different launch sites as they attempt launches with a variety of payload masses
 - Bar Chart (Pareto Chart) for Orbital Types and Success Rates - to assess 1st stage landing success for types of orbital launches attempted
 - Scatter Plot of Orbital Types (y-axis) versus Flight Number (x-axis) - to assess 1st stage landing success/failure as the flight number increases for different types of orbital launches attempted
 - Scatter Plot of Orbital Types (y-axis) versus Payload Mass (x-axis) - to assess 1st stage landing success/failure as different types of orbital launches are attempted with different payload masses
 - Line Chart of Average Mission Success Rate (y-axis) by Year (x-axis) - to assess 1st stage landing success/failure by year
- PART 2: Features Engineering & creation of OUTPUT FILE: dataset_part_3.csv
 - Created dummy variables for categorical variables (Orbits, LaunchSite, LandingPad, and Serial) in preparation for predictive modeling / classifier development
 - Cast all variables to float64 for numerical computation / predictive modeling

GitHub URL of completed EDA with Data Visualization Jupyter Notebook:

<https://github.com/DebraElkins/Coursera-DataScienceCapstoneProject/blob/main/jupyter-labs-eda-dataviz-v2.ipynb>

EDA with SQL

- INPUTS: The lab-provided dataset SpaceX.csv was obtained
- OUTPUTS: Using python code, the csv data was imported into a Pandas DataFrame, then written to a SQLite Database (my_data1.db) for analysis
- Python Libraries / Packages used: sqlalchemy, ipython-sql, i-python-sql->prettytable, csv, sqlite, pandas, numpy

Query Used – Description / Purpose	SQLite Code Statement
Query to find out names and datatypes of columns in the SQLite Database	<code>%sql PRAGMA TABLE_INFO(SPACEXTABLE);</code>
Query to extract the 1st 25 rows of SPACEXTABLE in My_Data1.DB	<code>%sql SELECT * FROM SPACEXTABLE LIMIT 25;</code>
10 Analysis Queries: 1. Display the names of the unique launch sites in the space mission 2. Display 5 records where launch sites begin with the string 'CCA' 3. Display the total payload mass carried by boosters launched by NASA (CRS) 4. Display average payload mass carried by booster version F9 v1.1 5. List the date when the first successful landing outcome in ground pad was achieved. 6. List the names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000 7. List the total number of successful and failure mission outcomes 8. List all the booster_versions that have carried the maximum payload mass, using a subquery with a suitable aggregate function. 9. List the records which will display the month names, failure landing_outcomes in drone ship ,booster versions, launch_site for the months in year 2015. 10. Rank the count of landing outcomes (such as Failure (drone ship) or Success (ground pad)) between the date 2010-06-04 and 2017-03-20, in descending order.	<pre>1. %sql SELECT DISTINCT "Launch_Site" FROM SPACEXTABLE; 2. %sql SELECT "Launch_Site" FROM SPACEXTABLE WHERE "Launch_Site" LIKE '%CCA%' LIMIT 5; 3. %sql SELECT SUM("Payload_Mass_Kg") AS Total_Payload FROM SPACEXTABLE WHERE "Customer" = 'NASA (CRS)'; 4. %sql SELECT AVG("Payload_Mass_Kg") AS Avg_Payload FROM SPACEXTABLE WHERE "Booster_Version" = 'F9 v1.1'; 5. %sql SELECT MIN("Date") AS FirstSuccessfulGPLanding FROM SPACEXTABLE WHERE "Landing_Outcome" = 'Success (ground pad)' ORDER BY "Date" ASC; 6. %sql SELECT "Booster_Version" FROM SPACEXTABLE WHERE ("Landing_Outcome" = 'Success (drone ship)') AND ("Payload_Mass_Kg" BETWEEN 4000 AND 6000); 7. %sql SELECT "Mission_Outcome", COUNT(*) AS "Count" FROM SPACEXTABLE GROUP BY "Mission_Outcome"; 8. %sql SELECT "Booster_Version", "Payload_Mass_Kg" FROM SPACEXTABLE WHERE "Payload_Mass_Kg" = (SELECT MAX("Payload_Mass_Kg") FROM SPACEXTABLE); 9. %sql SELECT SUBSTR("Date", 6,2) AS "Month", "Landing_Outcome", "Booster_Version", "Launch_Site" FROM SPACEXTABLE WHERE SUBSTR("Date", 0, 5) = '2015' AND "Landing_Outcome" = 'Failure (drone ship)'; 10. %sql SELECT "Landing_Outcome", Count(*) AS "Count" FROM SPACEXTABLE GROUP BY "Landing_Outcome" HAVING "Date" BETWEEN '2010-06-04' AND '2017-03-20' ORDER BY Count DESC;</pre>

GitHub URL of completed EDA with SQL Jupyter Notebook:

https://github.com/DebraElkins/Coursera-DataScienceCapstoneProject/blob/main/jupyter-labs-eda-sql-coursera_sqlite.ipynb

Interactive Mapping of Launch Sites with Folium

- Used Jupyter Labs and Python libraries (Pandas, Folium) to build an interactive map using lab-provided INPUT FILE: `spacex_launch_geo.csv`
- Developed a geographic visual map with the following information:
 1. Launch Sites by latitude & longitude, each with a circular blast proximity area, to show that all the sites are close to the Earth's equator, and near coastlines (to allow for rocket destruction over the ocean if a launch / landing experiences issues)
 2. Markers for successful and unsuccessful 1st stage landings, clustered by launch site to show a visual representation of success/failure by launch site
 3. A distance proximity tool & capability to estimate potential launch / landing failure impacts (e.g., an explosion's potential blast radius). For example, the tool was used to estimate distance between:
 - Kennedy Space Center LC-39A and the coastline
 - Kennedy Space Center LC-39A and the nearby airstrip

GitHub URL of completed Interactive Map with Folium in Jupyter Notebook:

https://github.com/DebraElkins/Coursera-DataScienceCapstoneProject/blob/main/lab_jupyter_launch_site_location.ipynb

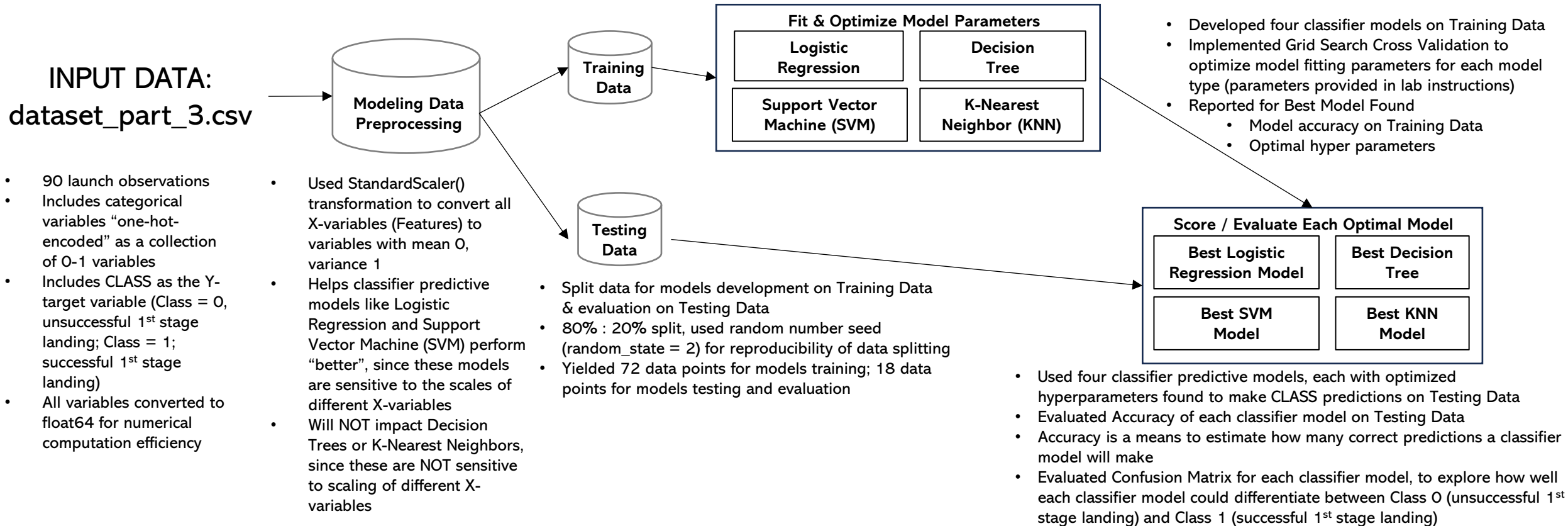
Interactive Data Visualization Dashboard with Plotly Dash

- Used Plotly / Dash and Python scripting to create an interactive data visualization dashboard
- The dashboard provides:
 1. Pie Chart to display a relative comparison of successful 1st stage landings across all launch sites
 2. Pie Chart to display the proportion of successful vs. unsuccessful 1st stage landings, when a launch site is selected
 3. Scatter plot to display successful vs. unsuccessful 1st stage landings for different Booster Versions used, when a particular payload mass (kg) range is provided (i.e., from 0 kg to 10,000 kg)

GitHub URL of completed Dashboard with Plotly Dash

- Python Script: https://github.com/DebraElkins/Coursera-DataScienceCapstoneProject/blob/main/SpaceX_Dash_App_FINAL_CODE.py
- Screenshots of Interactive Dashboard: <https://github.com/DebraElkins/Coursera-DataScienceCapstoneProject/blob/main/Dashboard%20Screenshots.pptx>

Predictive Analysis (Classification)



GitHub URL of completed First Stage Successful Landing Predictive Analysis Jupyter Notebook:

[https://github.com/DebraElkins/Coursera-DataScienceCapstoneProject/blob/main/SpaceX Machine%20Learning%20Prediction Part 5.ipynb](https://github.com/DebraElkins/Coursera-DataScienceCapstoneProject/blob/main/SpaceX%20Machine%20Learning%20Prediction%20Part%205.ipynb)

Results Summary

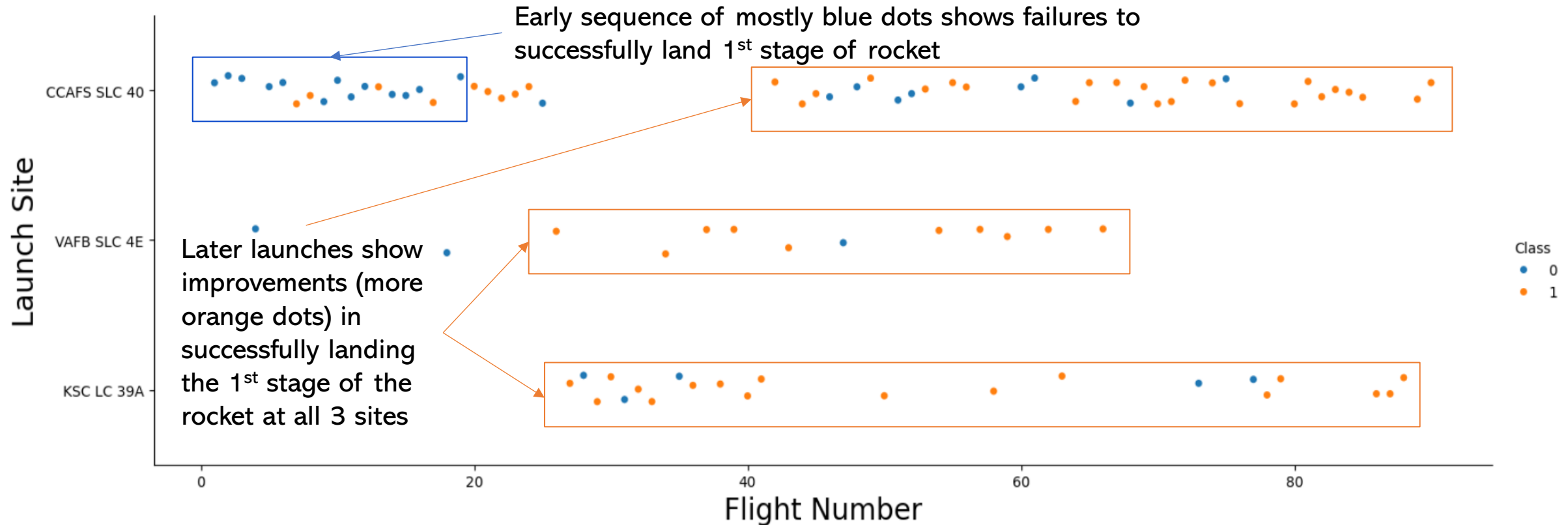
- Exploratory data analysis results
 - Data Wrangling showed **1st Stage Landing Success Rate of 67%** (60/90), based on prior launch data observations
 - SQL data analysis showed the average payload was 2,928.4 kg, and a maximum payload of 15,600 kg for all launches
- Interactive analytics demo in screenshots
 - Folium geospatial interactive mapping shows **Kennedy Space Center LC-39A** achieved 10/13 = **76.9% successful 1st stage landings**
 - Using Plotly / Dash for interactive analytics, we observed that SpaceX experienced **“organizational learning”** as it increased its launch number, and showed improvement in successfully landing the 1st stage of the rocket at different launch sites, for different orbits, for different payload masses (kg), and year over year
- Predictive analysis results: Four classifier models (Logistic Regression, Support Vector Machine, Decision Tree, and K Nearest Neighbor):
 - **83.3% accuracy** (i.e., all 4 models make mostly correct predictions regarding 1st stage landing success or failure)
 - **100% recall** (i.e., all 4 models were able to identify all successful 1st stage landings, with no “misses”)
 - The best **Logistic Regression** model was used to **predict the probability of SpaceX successfully landing the 1st phase of its spacecraft at 67.2%**, using all available launch data

The background of the slide is an abstract composition. It features a dark blue base color. Overlaid on this are numerous diagonal streaks in shades of blue and red, creating a sense of motion or data flow. A faint, light blue grid pattern is also visible, particularly in the lower half of the image. The overall effect is high-tech and digital.

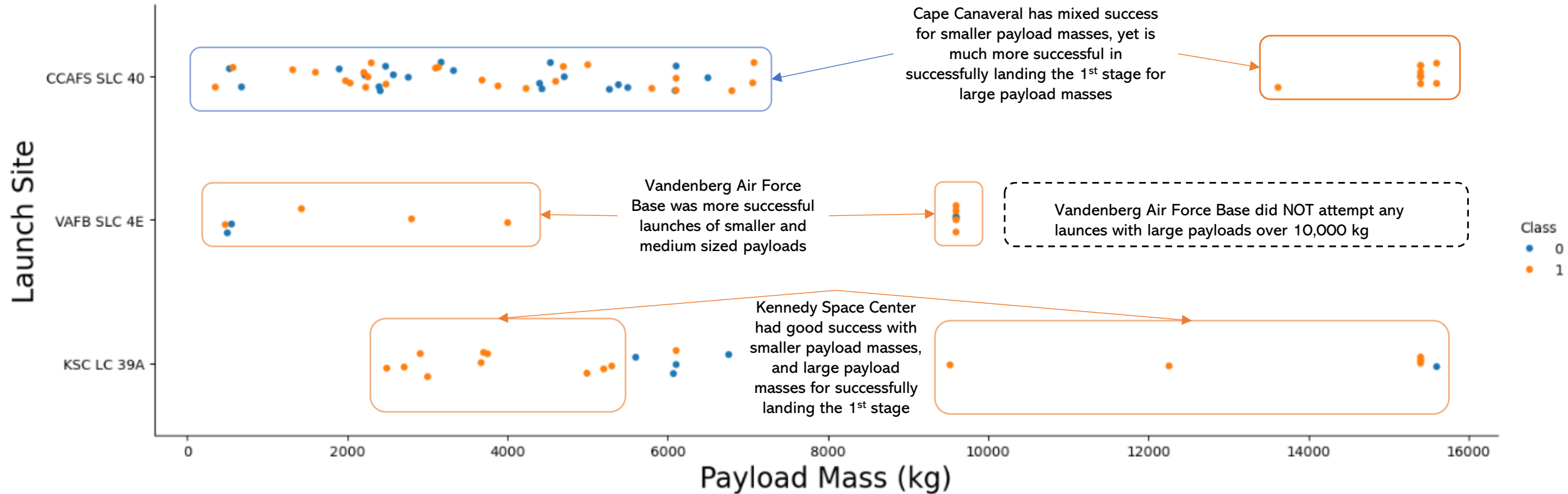
Section 2

Insights drawn from EDA

Flight Number vs. Launch Site

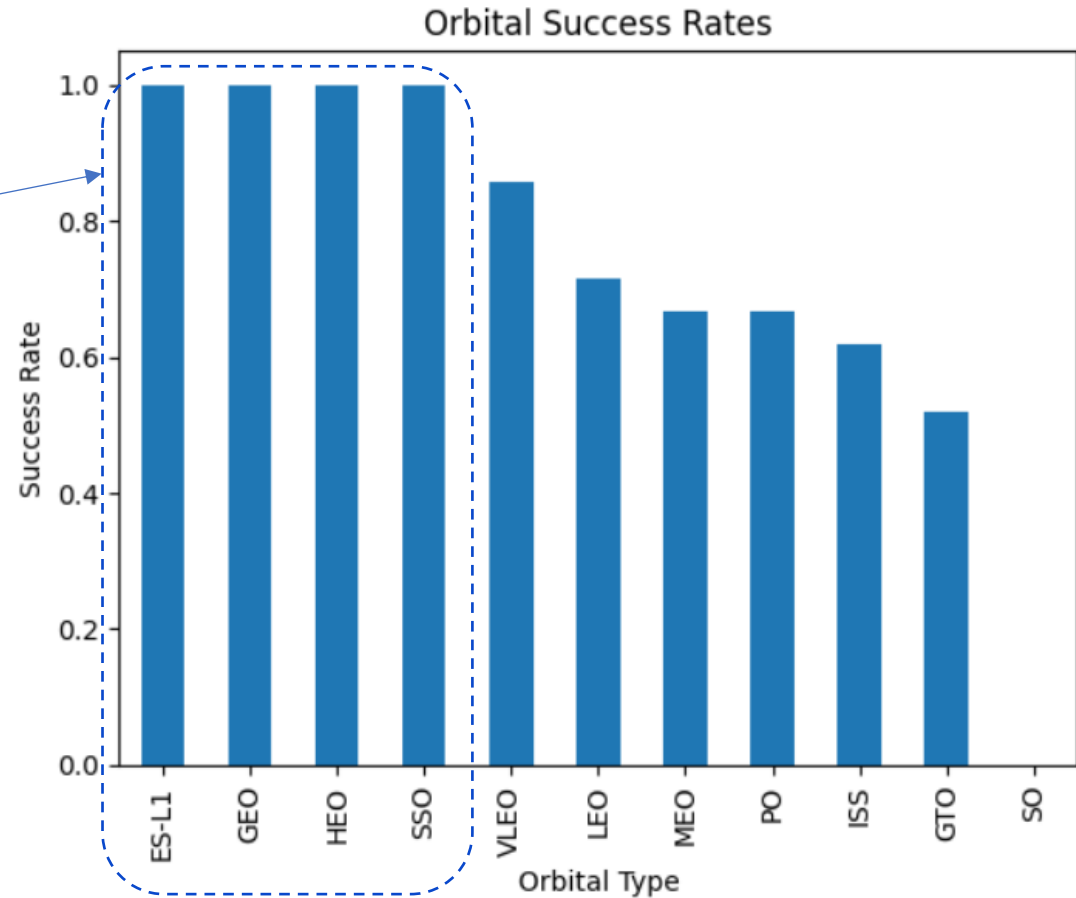


Payload vs. Launch Site

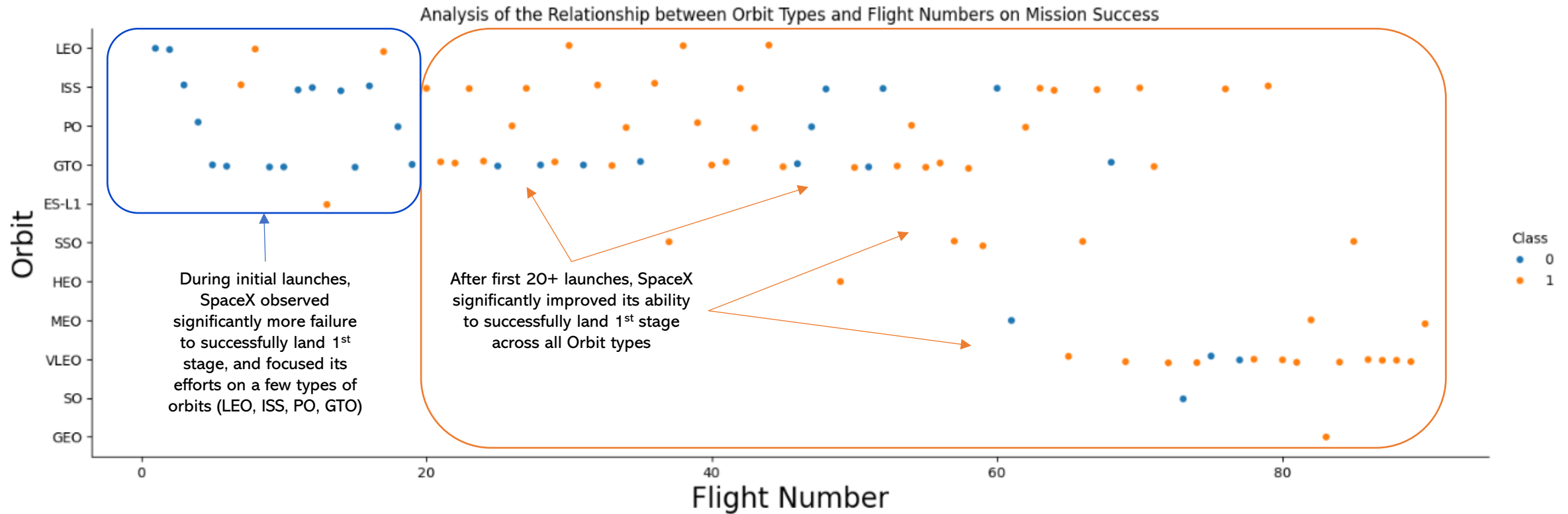


Success Rate vs. Orbit Type

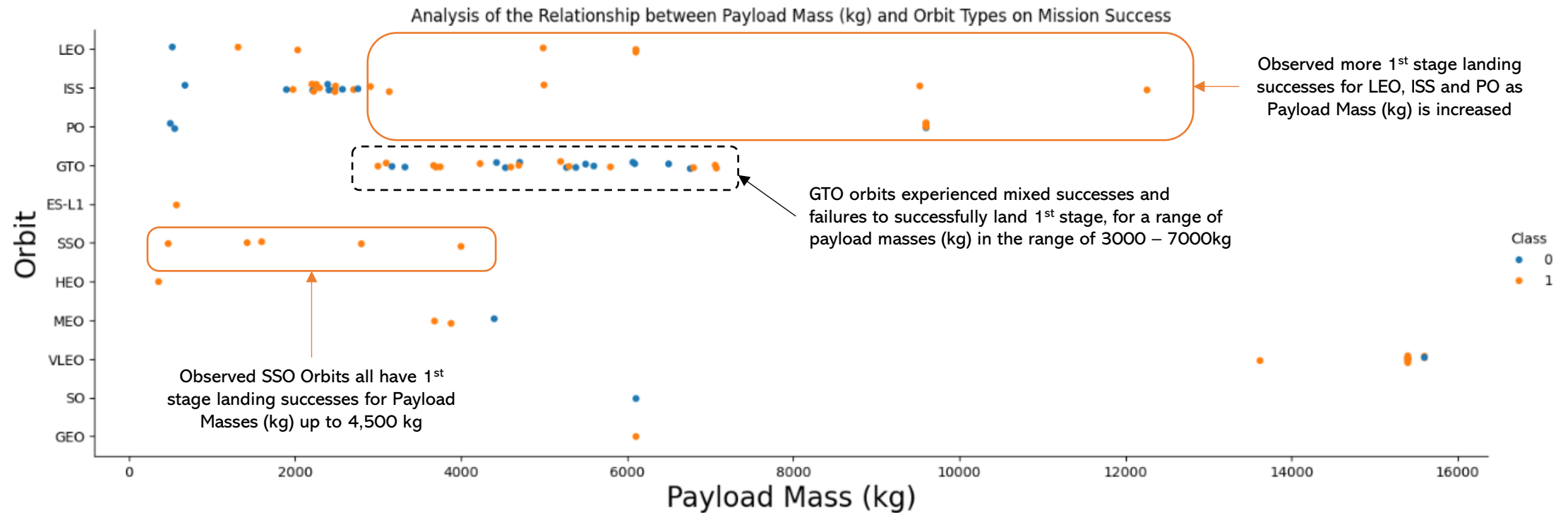
- 4 types of orbits (ES-L1, GEO, GEO and SSO) had the highest average success rates of 1.00 for 1st stage successful landing
- VLEO and LEO had the next highest average success rates with 0.857 and 0.714 respectively
- Remaining orbits (MEO, PO, ISS, GTO and SO) all were 0.667 or lower success rate on average



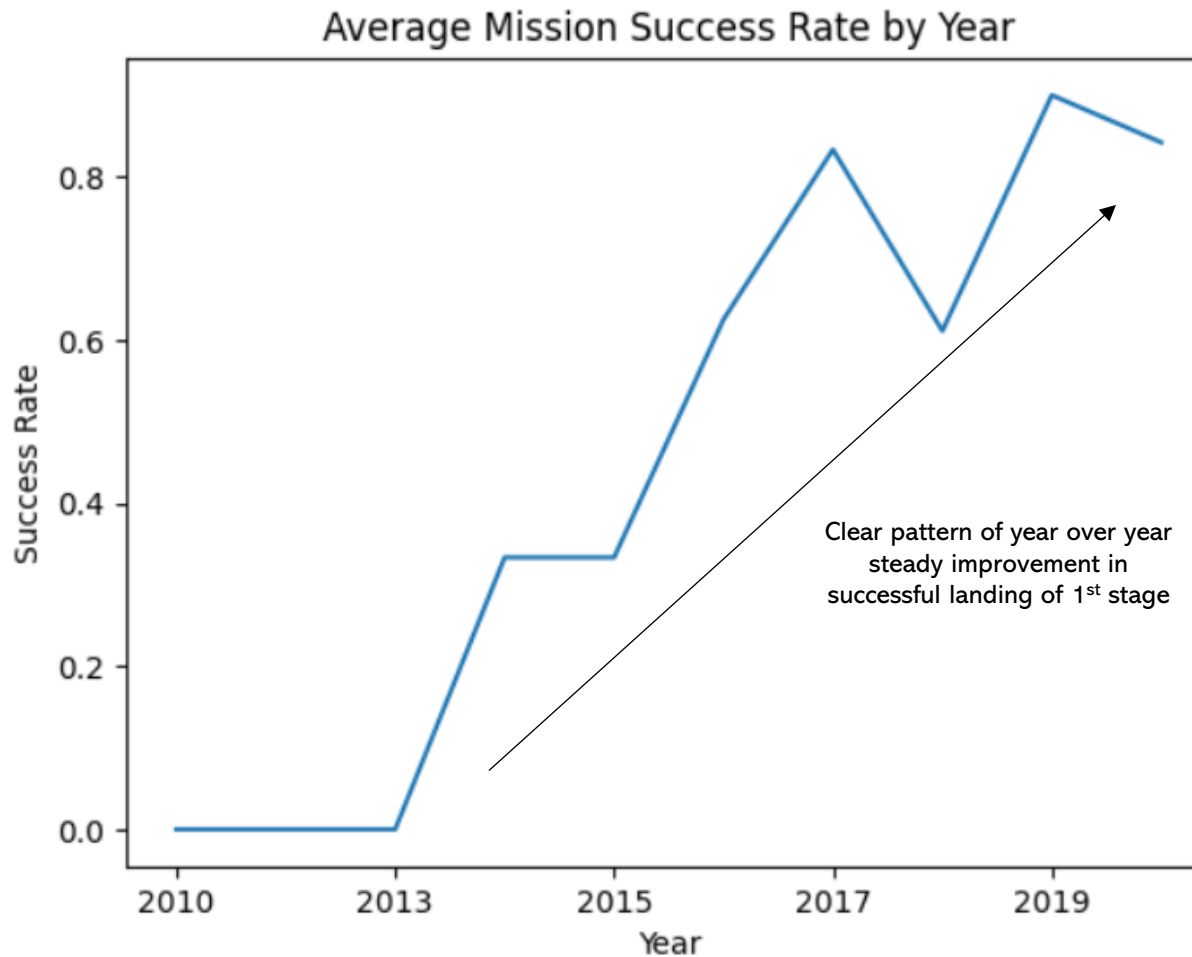
Flight Number vs. Orbit Type



Payload vs. Orbit Type



Launch Success Yearly Trend



All Launch Site Names

Using SQLite, there are 4
distinct / unique launch sites

Launch Site Code	Launch Site Longer Name
CCAFS LC-40	Cape Canaveral Launch Center-40
VAFB SLC-4E	Vandenberg Air Force Base Space Launch Center 4E
KSC LC-39A	Kennedy Space Center Launch Center 39A
CCAFS SLC-40	Cape Canaveral Space Launch Center-40

Display the names of the unique launch sites in the space mission

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

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

```
Done.
```

Launch_Site

CCAFS LC-40

VAFB SLC-4E

KSC LC-39A

CCAFS SLC-40

Launch Site Names Begin with 'CCA'

Display 5 records where launch sites begin with the string 'CCA'

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

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

Done.

```
] : Launch_Site
```

CCAFS LC-40

CCAFS LC-40

CCAFS LC-40

CCAFS LC-40

CCAFS LC-40

This SQL query...

Has a WHERE clause to search for Launch_Sites that have the string pattern "CCA" somewhere in their names (the % before and after CCA allows 'padding' of a site name string with white spaces)

Limits the number of records returned using "LIMIT 5" clause

Total Payload Mass

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)';
```

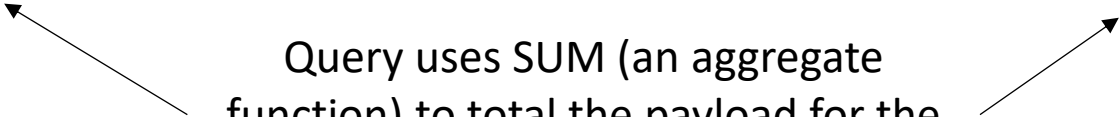
```
* sqlite:///my_data1.db
```

```
Done.
```

Total_Payload

45596

Query uses SUM (an aggregate function) to total the payload for the specified customer = NASA (CRS)



Average Payload Mass by F9 v1.1

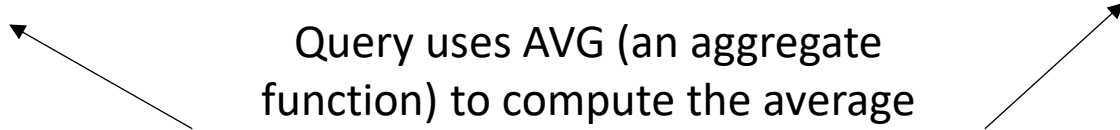
Display average payload mass carried by booster version F9 v1.1

```
] : %sql SELECT AVG("Payload_Mass_Kg") AS Avg_Payload FROM SPACEXTABLE WHERE "Booster_Version" = 'F9 v1.1';
```

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

```
] : Avg_Payload  
-----  
2928.4
```

Query uses AVG (an aggregate function) to compute the average payload over all launches using Booster_Version = "F9 v1.1"



First Successful Ground Landing Date

List the date when the first succesful landing outcome in ground pad was acheived.

Hint: Use min function

```
[32]: %sql SELECT MIN("Date") AS FirstSuccessfulGPLanding FROM SPACEXTABLE WHERE "Landing_Outcome" = 'Success (ground pad)' ORDER BY "Date" ASC;
```

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

```
[32]: FirstSuccessfulGPLanding  
-----  
2015-12-22
```

Query uses MIN (an aggregate function) to find the earliest date over all launches where the Landing_Outcome = "Success (ground pad)"

Note:

- We added the ORDER By "Date" ASC clause in the query to sort the result of the query in ascending order of date, in case there were several successful landing outcomes on the same date, and we might need to do additional work to resolve a tie using some other variable (e.g., time or location)
- In this case, there was only 1 launch on this date, so no ties occurred.

Successful Drone Ship Landing with Payload between 4000 and 6000

List the names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000

```
[34]: %sql SELECT "Booster_Version" FROM SPACEXTABLE WHERE ("Landing_Outcome" = 'Success (drone ship)') AND ("Payload_Mass_Kg" BETWEEN 4000 AND 6000)
```

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

```
[34]: Booster_Version
```

F9 FT B1022

F9 FT B1026

F9 FT B1021.2

F9 FT B1031.2

There are 4 different F9 booster versions that have successfully been used for 1st stage landing on drone ships, with payload masses in the range of 4,000 – 6,000 kg

Logic Filter 1

AND

Logic Filter 2

The SQL statement has a compound WHERE clause with two logical filters separated by AND:

1. "Landing_Outcome" = "Success (drone ship)"
2. "Payload_Mass_Kg" BETWEEN 4000 AND 6000

Total Number of Successful and Failure Mission Outcomes

List the total number of successful and failure mission outcomes

```
[41]: %sql SELECT "Mission_Outcome", COUNT(*) AS "Count" FROM SPACEXTABLE GROUP BY "Mission_Outcome";
```

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

Done.

```
[41]:
```

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



For this dataset, there are two different counts for “Success” – so Mission_Outcome may have some different whitespace characters that distinguish “Success” from “Success_” or “_Success” (underscore added explicitly here to show where a whitespace may be occurring)

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_Kg") FROM SPACEXTABLE);
```

```
* 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

Booster Versions that
have carried the
maximum payload
mass of 15,600 kg are
all F9 B5 Boosters

Subquery to find all
launches that have
carried maximum
payload mass

Subquery uses MAX (an
aggregate function) find
the largest Payload Mass
value over all launches

2015 Launch Records

List the records which will display the month names, failure landing_outcomes in drone ship ,booster versions, launch_site for the months in year 2015.

Note: SQLite does not support monthnames. So you need to use substr(Date, 6,2) as month to get the months and substr(Date,0,5)='2015' for year.

```
%sql SELECT SUBSTR("Date", 6,2) AS "Month", "Landing_Outcome", "Booster_Version", "Launch_Site" FROM SPACESTABLE
WHERE SUBSTR("Date", 0, 5) = '2015' AND "Landing_Outcome" = 'Failure (drone ship)';
```

* 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

Date seems to be stored as YYYYDDMM format

This query uses substr(Date, 6, 2) to parse the date string to start at character 6 in the string, then obtain the 2 following characters for the month

This query uses substr(Date, 0, 5) to parse the date string starting at character 0, and selecting the next 4 characters (stopping before character 5) to obtain the 4 characters for Year

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

Rank the count of landing outcomes (such as Failure (drone ship) or Success (ground pad)) between the date 2010-06-04 and 2017-03-20, in descending order.

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



* sqlite:///my_data1.db

Done.

Landing_Outcome	Count
No attempt	21
Success (drone ship)	14
Success (ground pad)	9
Failure (drone ship)	5
Controlled (ocean)	5
Uncontrolled (ocean)	2
Failure (parachute)	2
Precluded (drone ship)	1

This clause "ORDER BY Count DESC" ranks the Count of landing outcomes in descending order

Using the HAVING "Date" BETWEEN '2010-06-04' and '2017-03-20' Permits the GROUP BY "Landing_Outcome" to filter records to selected dates for the Count Aggregate Function to summarize

A satellite view of Earth from space, showing the curvature of the planet and city lights at night. The background is a deep blue gradient.

Section 3

Launch Sites Proximities Analysis

Overview of 4 U.S. Based SpaceX Launch Sites

All four U.S. launch sites are located within the connected U.S. (CONUS)

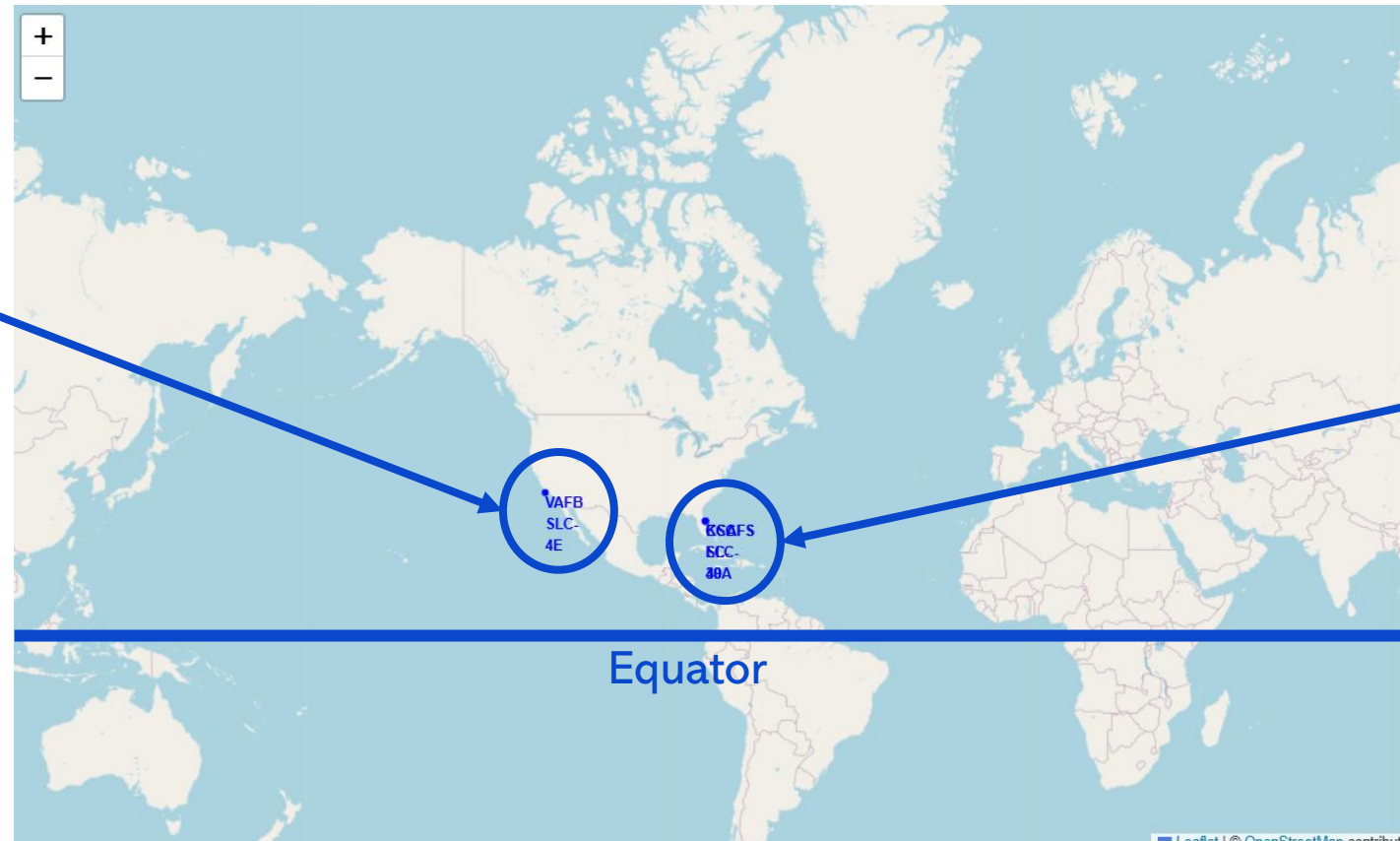
- Proximity to the equator allows launches to efficiently reach orbit around the Earth, escaping the Earth's gravity
- Proximity to the coastline allows launch or landing debris to fall into the ocean rather than a populated area

One launch site on the West Coast of the United States:

1. Vandenberg Air Force Base (SLC-4E)

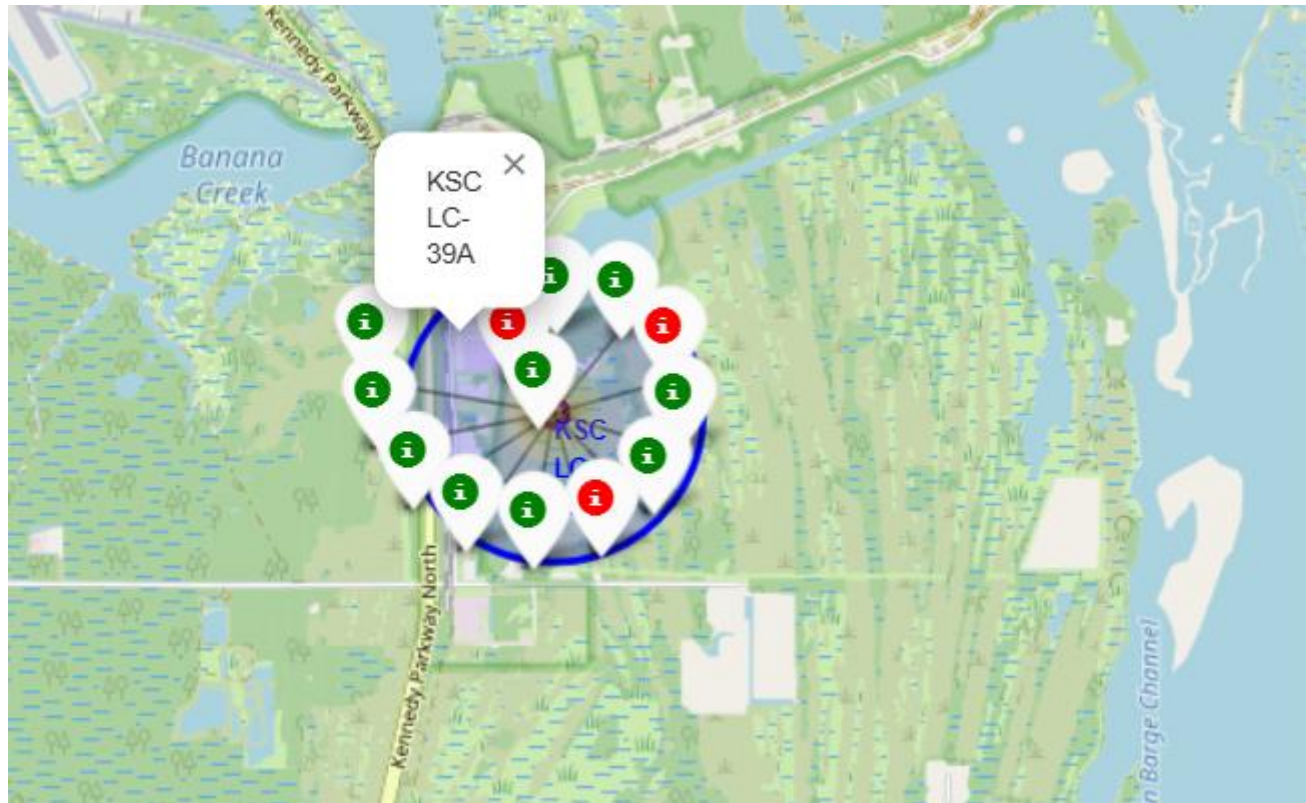
Three launch sites on the East Coast of the United States:

1. Cape Canaveral (LC-40)
2. Cape Canaveral (SLC-40)
3. Kennedy Space Center (LC-39-A)



Profile of Kennedy Space Center Launch Site

- This Folium map plot shows Kennedy Space Center LC-39A and its collection of launches with 10 successful (green) and 3 unsuccessful (red) 1st stage landings
- Based on the data shown, Kennedy Space Center achieved $10/13 = 76.9\%$ successful 1st stage landings



Proximity Map for Kennedy Space Center LC-39A Landing Site

Kennedy Space Center has several locations nearby that may be impacted by launch or landing debris including:

- An airstrip (6.46 km away, distance shown in purple)
- The coastline (6.78 km away, distance shown in orange)

It is also worth noting that the Kennedy Space Center is near Cape Canaveral Launch Center, shown in the bottom right corner of the map. Planning launches from Kennedy Space Center must also incorporate trajectories that avoid Cape Canaveral



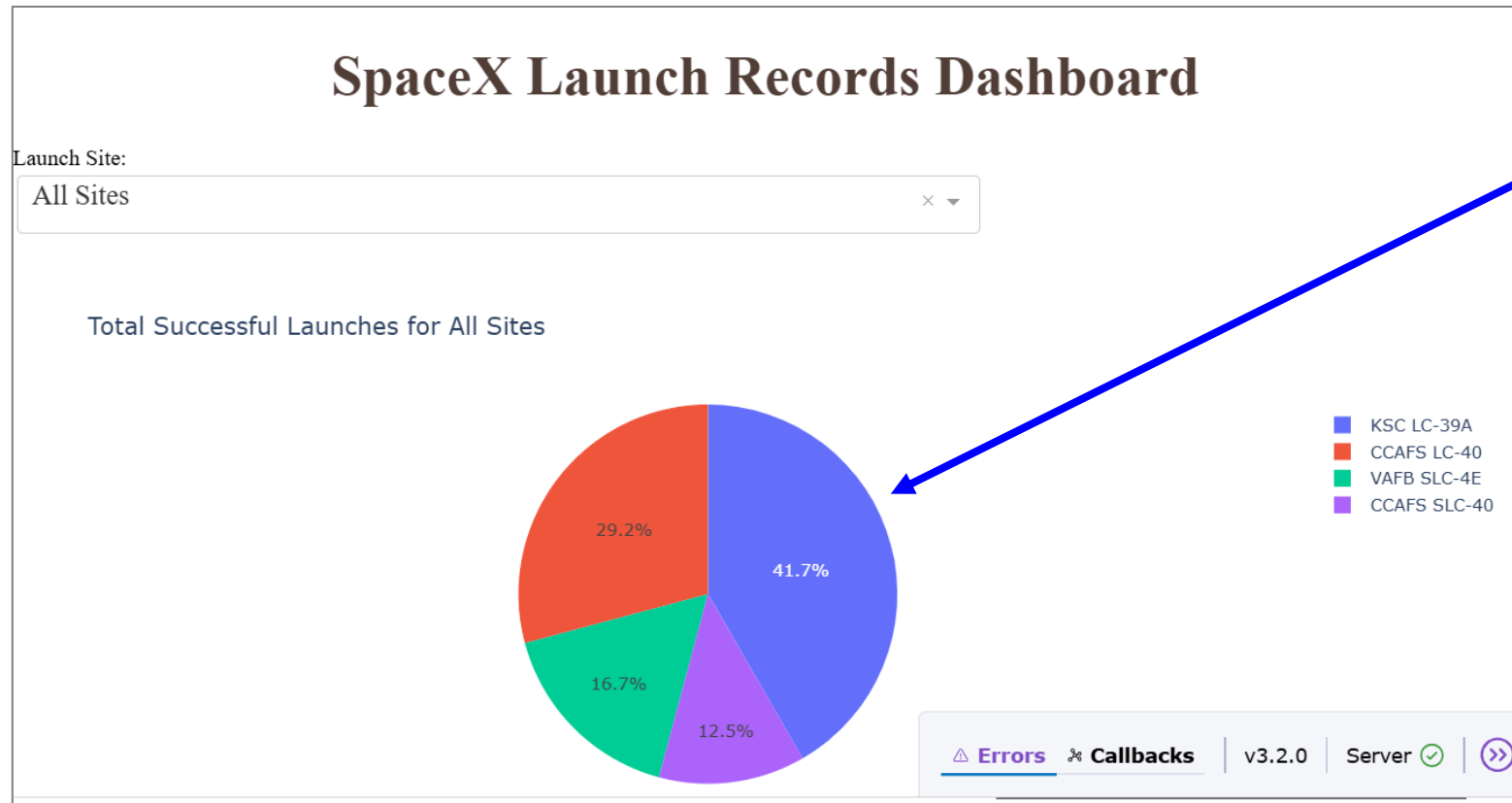


Section 4

Build a Dashboard with Plotly Dash

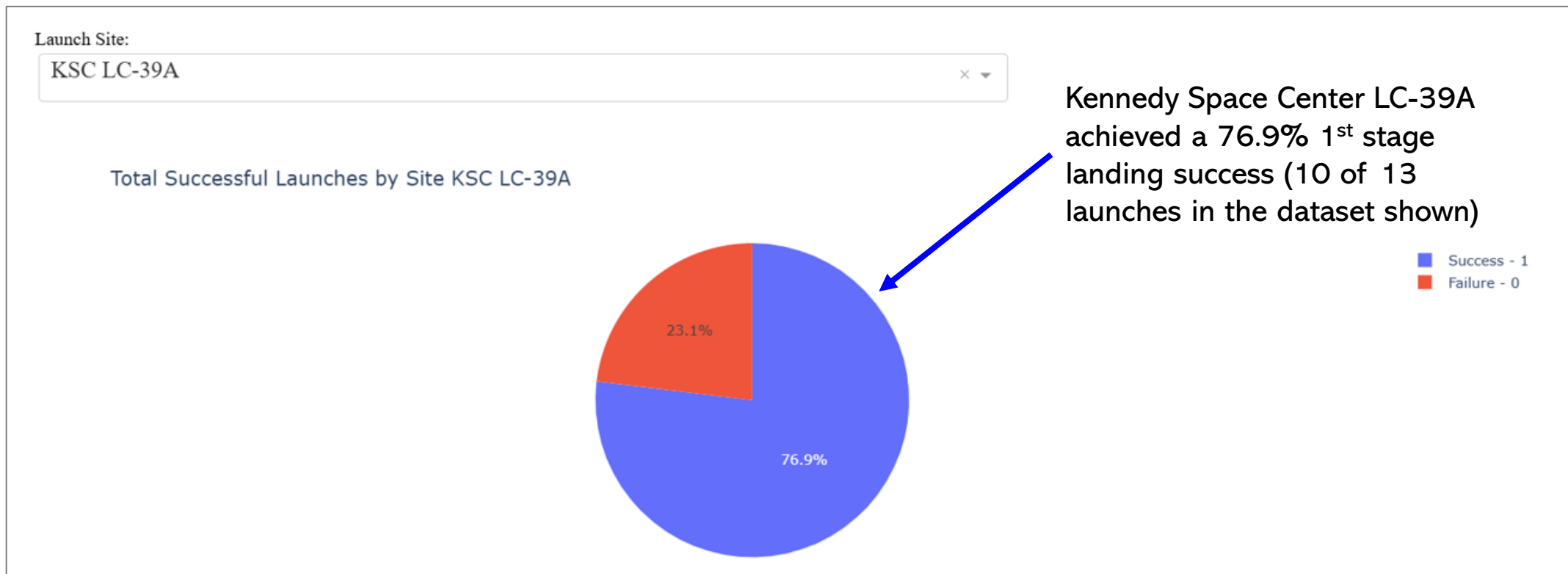
Comparison of Launch Sites for Successful Stage 1 Landings

Kennedy Space Center LC-39A has the highest successful 1st stage landings, when compared across all four sites



Kennedy Space Center LC-39A had the most successes in 1st stage landings among all four launch sites considered, with 41.7% of all successful landings (10 of 24 in the dataset shown)

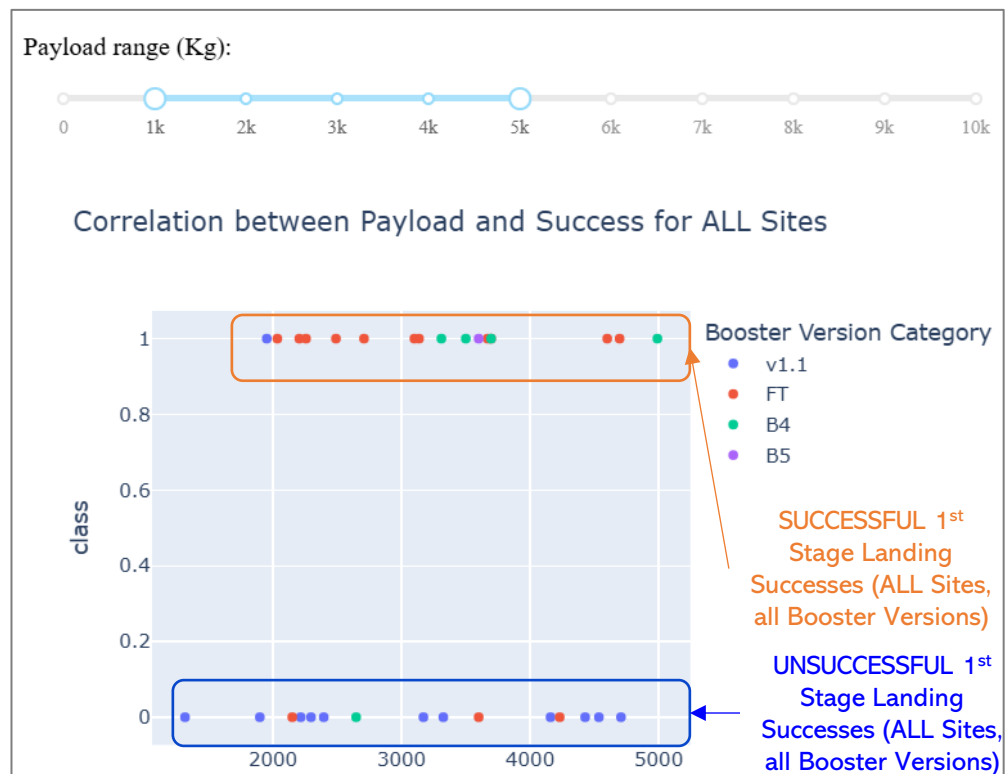
Kennedy Space Center's 1st Stage Landing Success Performance



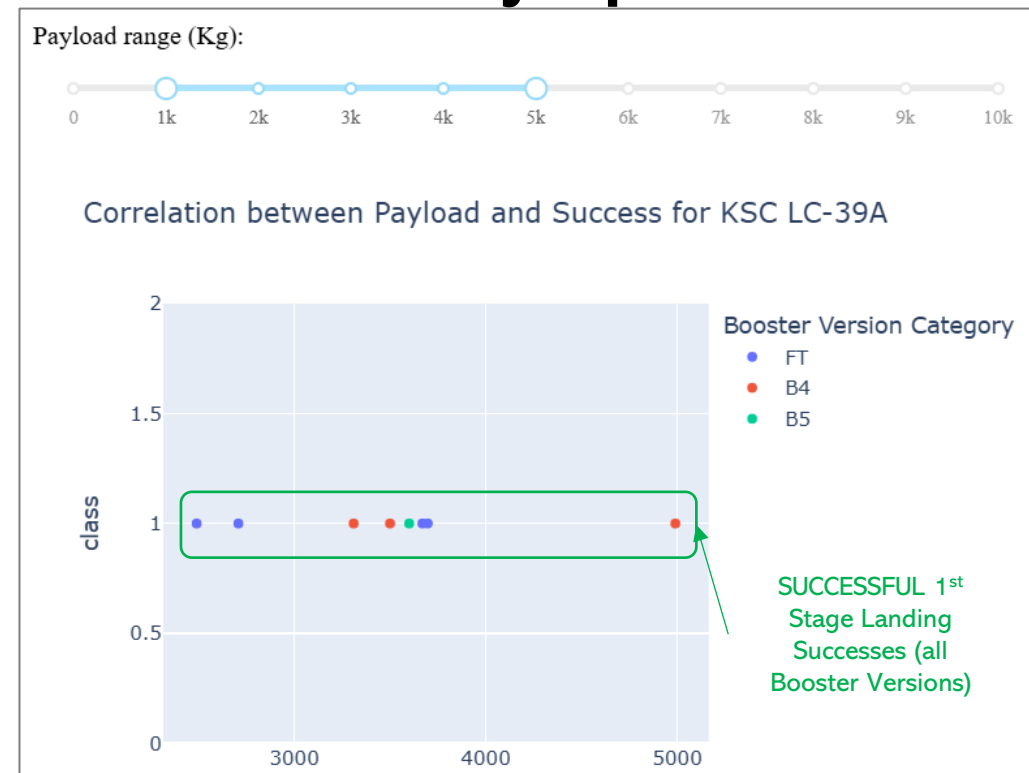
Visual Analysis of Payload Mass (Kg) and 1st Stage Landing Success

- For payload masses in the range of 1,000 kg to 5,000 kg, there is approximately an equal change of 1st stage landing success across ALL sites and using any of the four booster version types considered
- However, the Kennedy Space Center LC-39A achieved 100% successful 1st stage landing success (all 8 launches) for this payload mass range of 1,000 kg to 5,000 kg

ALL SITES



Kennedy Space Center



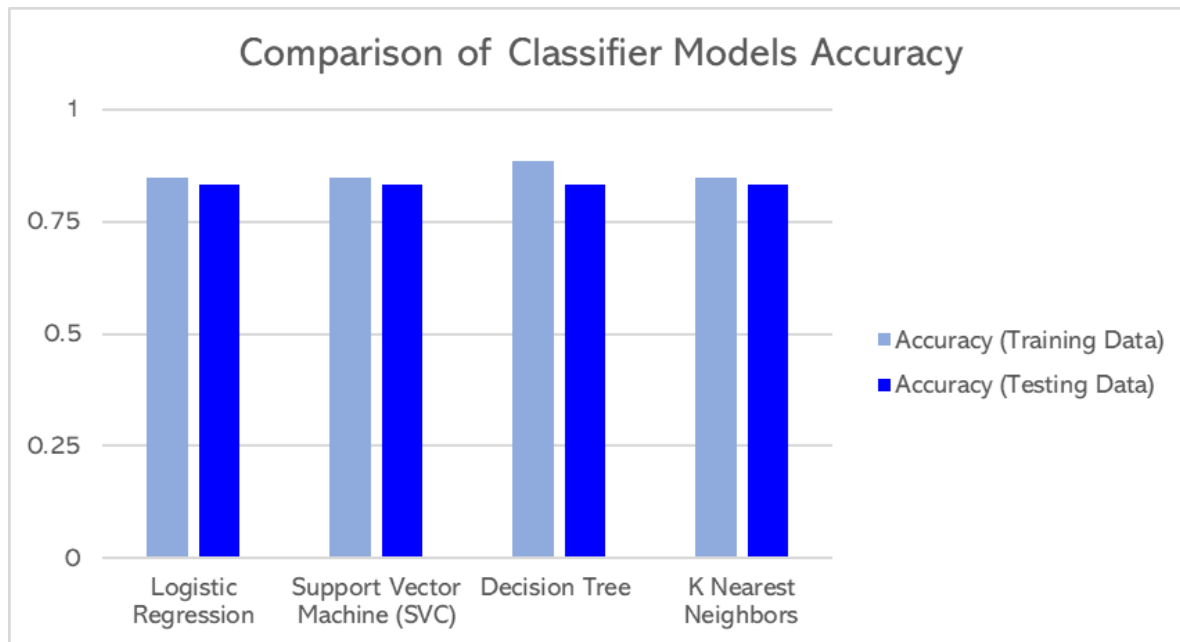


Section 5

Predictive Analysis (Classification)

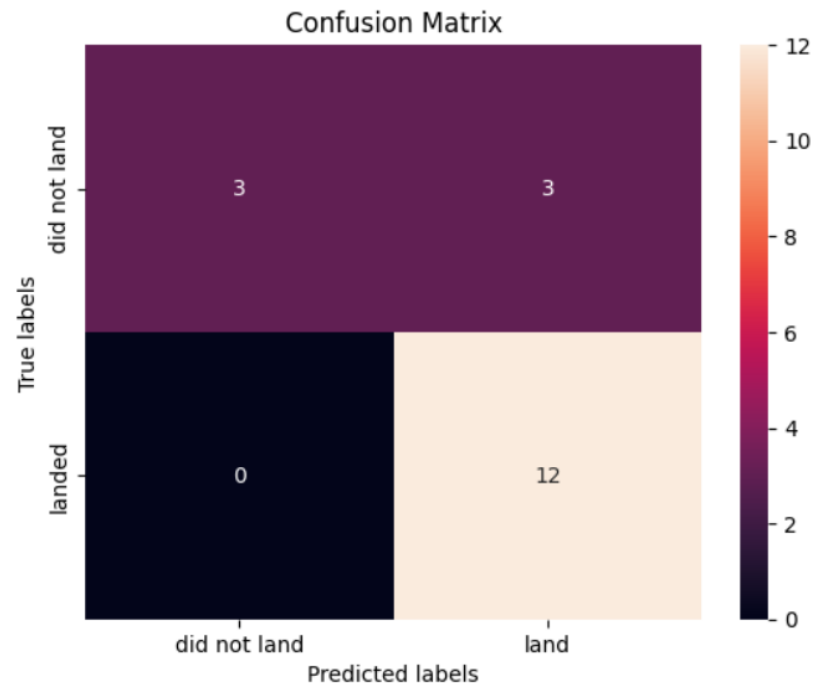
Classification Accuracy

- All four classifier models perform well on **training and testing data**, with over 80% accuracy
- All **four models perform equally** well in terms of **accuracy of correct classification, at 83.3%**, when compared on testing data
- The **Logistic Regression** model can also be used to **predict the probability of SpaceX successfully landing the 1st phase of its spacecraft** given launch input parameters such as launch site, orbit, payload mass, booster variant, etc. For the best Logistic Regression model, **we estimate the probability of successfully landing the 1st phase of its spacecraft at 67.2%**, using all available launch data



Classifier Model	Accuracy (Training Data)	Accuracy (Testing Data)	Optimized Parameters
Logistic Regression	0.8464	0.833	'C': 0.01, 'penalty': 'l2', 'solver': 'lbfgs'
Support Vector Machine (SVC)	0.8482	0.833	'C': 1.0, 'gamma': 0.03162277660168379, 'kernel': 'sigmoid'
Decision Tree	0.8857	0.833	'criterion': 'gini', 'max_depth': 8, 'max_features': 'sqrt', 'min_samples_leaf': 4, 'min_samples_split': 2, 'splitter': 'random'
K Nearest Neighbors	0.8482	0.833	'algorithm': 'auto', 'n_neighbors': 10, 'p': 1

Confusion Matrix



All four classifier models yielded the same confusion matrix

A **confusion matrix** shows how each model predicts 1st stage landing outcomes (x-axis) vs. actual outcomes (y-axis) on the testing dataset

In terms of **classifier model performance**, each model was able to assess on the testing data (18 launches)

- True Positive - 12 (Predicted 1st stage successful landing, Actual result was successful landing)
- False Positive - 3 (Predicted 1st stage successful landing, Actual result was NOT a successful landing)
- True Negative – 3 (Predicted 1st stage would NOT successfully land, Actual result was NOT a successful landing)
- False Negative – 0 (Predicted 1st stage would NOT successfully land, Actual result was a successful landing)

Accuracy is a means to estimate how many correct predictions a classifier model will make given sample observations.

Here, **Accuracy**

$$= (\text{True Positives} + \text{True Negatives}) / (\text{All Outcomes})$$

$$= (12 + 3) / (18)$$

$$= 83.3\%$$

Recall is a means to estimate how well a classifier model can identify all positive outcomes in a dataset (i.e., all successful 1st stage landings)

Here, **Recall**

$$= (\text{True Positives}) / (\text{True Positives} + \text{False Negatives})$$

$$= (12) / (12)$$

$$= 100.0\%$$

Conclusions

- Based on the analysis
 - SpaceX has a **1st Stage Landing Success Rate of 67%** (60/90), based on prior launch data
 - SpaceX demonstrated **“organizational learning”** as it increased its launch number. Notably, SpaceX showed improvement in successfully landing the 1st stage of the rocket at different launch sites, for different orbits, for different payload masses (kg), and year over year
 - Folium geospatial mapping shows **Kennedy Space Center LC-39A** achieved $10/13 = 76.9\%$ **successful 1st stage landings**
 - **Predictive Modeling** yielded 4 classifier models (Logistic Regression Support Vector Machine, Decision Tree, K Nearest Neighbor), each with **83.3% accuracy** (i.e., all 4 models make mostly correct predictions regarding 1st stage landing success or failure) and **100% recall** (i.e., all 4 models were able to identify all successful 1st stage landings, with no “misses”)
 - The **Logistic Regression** model can also be used to **predict the probability of SpaceX successfully landing the 1st phase of its spacecraft** given launch input parameters such as launch site, orbit, payload mass, booster variant, etc. For the best Logistic Regression model, **we estimate the probability of successfully landing the 1st phase of its spacecraft at 67.2%**, using all available launch data
- **SpaceX can deliver launch capabilities at the cost of \$62 M per launch, because they can successfully land and reuse the first stage of its spacecraft.** Other providers should NOT bid, unless they can demonstrate a comparable cost per launch.

Appendix

Problem-Solving Process: Files and Datasets

This diagram shows how files and datasets are used in different steps of the problem-solving process

GitHub Repository: <https://github.com/DebraElkins/Coursera-DataScienceCapstoneProject>

PHASE 1: Project Planning & Data Preparations

Jupyter Lab Notebook Files:

1. jupyter-labs-spacex-data-collection-api.ipynb

INPUT: SpaceX API

<https://api.spacexdata.com/v4/launches/past>

OUTPUT: dataset_part_1.csv

2. jupyter-labs-webscraping.ipynb

INPUT: Wikipedia page

https://en.wikipedia.org/wiki/List_of_Falcon_9_and_Falcon_Heavy_launches

OUTPUT: spacex_web_scraped.csv

3. labs-jupyter-spacex-Data wrangling.ipynb

INPUT: dataset_part_1.csv

OUTPUT: dataset_part_2.csv

PHASE 2: Exploratory Data Analysis

Jupyter Lab Notebook Files:

1. jupyter-labs-eda-dataviz-v2.ipynb

INPUT: dataset_part_2.csv

OUTPUT: dataset_part_3.csv

2. jupyter-labs-eda-sql-coursera_sqlite.ipynb

INPUT: SpaceX.csv (provided for lab)

OUTPUT: SQLite Database my_data1.db

PHASE 3: Data Visualization & Predictive Analysis

Jupyter Lab Notebook File:

1. lab_jupyter_launch_site_location.ipynb

INPUT: spacex_launch_geo.csv

OUTPUTS: Folium Interactive Maps

Python Script:

2. SpaceX_Dash_App_FINAL_CODE.py

INPUT: spacex_launch_dash.csv

OUTPUTS:

Plotly / Dash interactive web app; Dashboard Screenshots.pptx

Jupyter Lab Notebook File:

3. SpaceX_Machine Learning Prediction_Part_5.ipynb

INPUTS: X-vars: dataset_part_2.csv; CLASS Y-var: dataset_part_3.csv

OUTPUT: Parameters for Optimal Classifiers: Logistic Regression, SVM, Decision Tree, K Nearest Neighbor

PHASE 4: Results & Conclusions

Other:

- REPORT.pdf (this document)

Intermediate Data Files & Jupyter Lab Notebook Files Staging

- **GitHub Repository:** <https://github.com/DebraElkins/Coursera-DataScienceCapstoneProject>

Thank you!

