IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Maia Otermin
May 4, 2024

Outline

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

Executive Summary

- In order to determine what a competitive bid against SpaceX would be, SpaceY needs to determine SpaceX's cost of a launch. One component of cost is the rate of successful and unsuccessful launches under different conditions
- Data on SpaceX Falcon9 launches was gathered, cleaned, and analyzed in order to identify what factors relate to successful mission outcomes
  - Key factors identified: payload mass, orbit, launch site, flight number, and booster version
  - Interactive maps and dashboards were also created to examine patterns in the data
- Four different modeling techniques were tested under a variety of parameters to identify the model & parameter combination that yields the most accurate results
  - (1) Logistic Regression, (2) Smooth Vector Classification, (3) Decision Tree, & (4) K-Nearest Neighbor
- Decision Tree produced the most accurate model, with an accuracy score of 94%
- All analysis, code, and project documentation can be found on Github.

Introduction

- SpaceY is an company seeking to compete with SpaceX, the leader in the space rocket industry
- Part of SpaceX's success is due to the company's cost to launch a Falcon9 rocket:
  - Falcon9 Rocket: $62 million
  - Other providers: $165+ million
- SpaceX is able to offer such a (comparatively) low price because it can reuse the first stage of its rockets, by far the most expensive component
- If SpaceY can determine when and why a first stage rocket lands (and thus be available for reuse), SpaceY can determine the cost of a launch – and develop a competitive bid against SpaceX for a rocket launch
- This project seeks to answers the following questions:
  - Will a SpaceX first-stage rocket land successfully? What factors contribute to a mission outcome?
  - What is the overall cost of each rocket launch?

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - Collected directly from Space X's API (https://api.spacexdata.com/v4/launches/past)
  - Collected by web-scraping the Wikipedia page for Falcon 9 Launches (https://en.wikipedia.org/wiki/List_of_Falcon_9_and_Falcon_Heavy_launches)
- Perform data wrangling
  - The data was processed to remove missing fields and develop indicator variables
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - Multiple techniques were used to build a model, each assessed using accuracy scores

Data Collection

- Two methods were used to collect data on Falcon9 Launches:
  - Accessing the Space X's API directly (https://api.spacexdata.com/v4/launches/past)
  - Collected by web-scraping the Wikipedia page for Falcon 9 Launches (https://en.wikipedia.org/wiki/List_of_Falcon_9_and_Falcon_Heavy_launches)
- The same variables were collected using either method:
  - Flight Number
  - Date
  - Booster Version
  - Payload Mass
  - Orbit
  - Launch Site
  - Outcome
  - Flights
  - Grid Fins
  - Reused
  - Legs
  - Landing Pad
  - Block
  - ReusedCount
  - Serial
  - Longitude
  - Latitude
- See Appendix for a full data dictionary

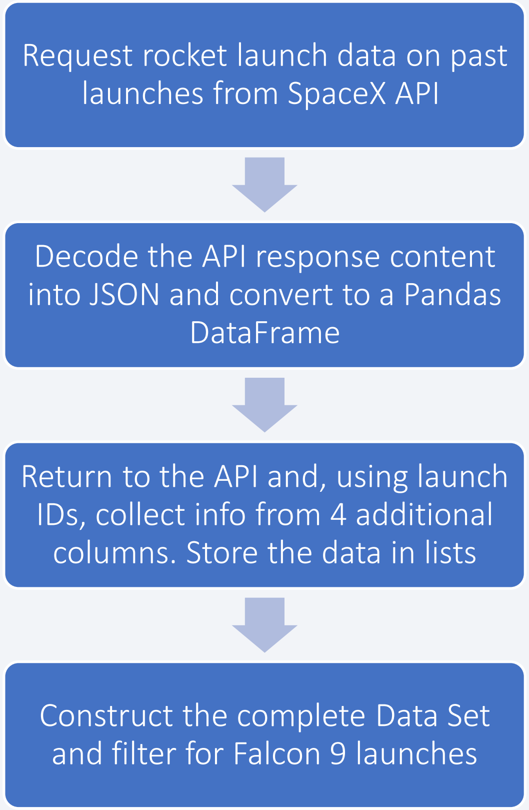
Data Collection – SpaceX API

- Data was collected directly from the SpaceX API using REST calls

```
https://api.spacexdata.com/v4/launches/past
```

- The process for accessing the API involved collecting information from specific columns:
  - Rocket Column
  - Launchpad Column
  - Payload Column
  - Cores Column
- Jupyter Notebook #1 contains the completed SpaceX API calls

Data Collection Process using SpaceX API



Data Collection - Scraping

- Data was scraped from the Wikipedia page for Falcon 9 and Falcon Heavy Launches

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

- Information was collected from HTML an table with information for 78 flights

- [Jupyter Notebook #2](#) contains the code for this web-scraping process

Data Collection Process Using Web-scraping

Request the Falcon9 Launch Wiki page from its URL (HTTP GET)

↓

Use the HTTP response to create a BeautifulSoup Library

↓

Extract all column/variable names from the HTML table header

↓

Create a Pandas DataFrame by parsing the launch HTML Tables

Data Wrangling

- The data collection process resulted in a dataset with17 variables:
  - For additional information on these variables, a data dictionary is included in Appendix A1
- After exploration, some variables were of particular interest:
  - 3 launch sites:
    - Cape Canaveral Space Force Station Space Launch Complex (“CCSFS SLC-40” or “CCSFS LC-40”)
    - Vandenberg Air Force Base Space Launch Complex 4E ("VAFB SLC 4E")
    - Kennedy Space Center Launch Complex 39A ("KSC LC 39A")
  - 11 possible orbits (see Appendix for specific values)
- For easier data analysis, the 'Outcome' variable was used to create a binary variable, 'Class', that abstracts all outcomes to Success (1) or Failure (0)
  - Space X's Falcon 9 success rate, across all launch sites and orbits, is **66.67%**
- Jupyter Notebook #3 contains the code for the data wrangling portion of this project

EDA with Data Visualization

- Exploratory Data Analysis ("EDA") was conducted using data visualization techniques that rely on Python' Pandas and MatPlotlib libraries. In particular, these relationships were examined:
  - Payload by Flight Number (Has the size of the payload changed as more launches took place?)
  - Launch Site by Flight Number (Is there a pattern in which launch sites were chosen over time?)
  - Launch Site by Payload Mass (Is the size of the payload correlated with the chosen launch site?)
  - Success Rate by Orbit Type (Is the success rate of a launch correlated with the targeted orbit?)
  - Orbit by Flight Number (Has the target orbit of a launch changed over time?)
  - Orbit Type Payload Mass (Is the size of the payload correlated with the targeted orbit for a launch?)
  - Success Rate by Year (Has the success rate for a launch improved over time?)
- Data visualization indicated 12 variables would be useful for modeling
- Jupyter Notebook #5 contains the EDA conducted with data visualization techniques

EDA with SQL

- EDA was also performed using Structured Query Language ("SQL") queries
- The following were examined to search for patterns in the data:
  - Unique launch sites, including records for those marked "CCA"
  - Total payload mass carried by boosters launched by NASA (CRS)
  - Average Payload mass carried by booter version F9 v1.1
  - Date when first successful landing outcome on a ground pad was achieved
  - List of boosters that have successfully landed on a drone ship with a payload greater than 4,000 kg but less than 6,000 kg.
  - Total number of successful and failed mission outcomes
  - Names of the booster versions that have carried the maximum payload mass
  - List of launches that failed (the month, booster version, and launch site) in 2015
  - Successful and failed launches between June 4, 2010 and March 20, 2017
- Jupyter Notebook #4 contains the code for EDA with SQL

Build an Interactive Map with Folium

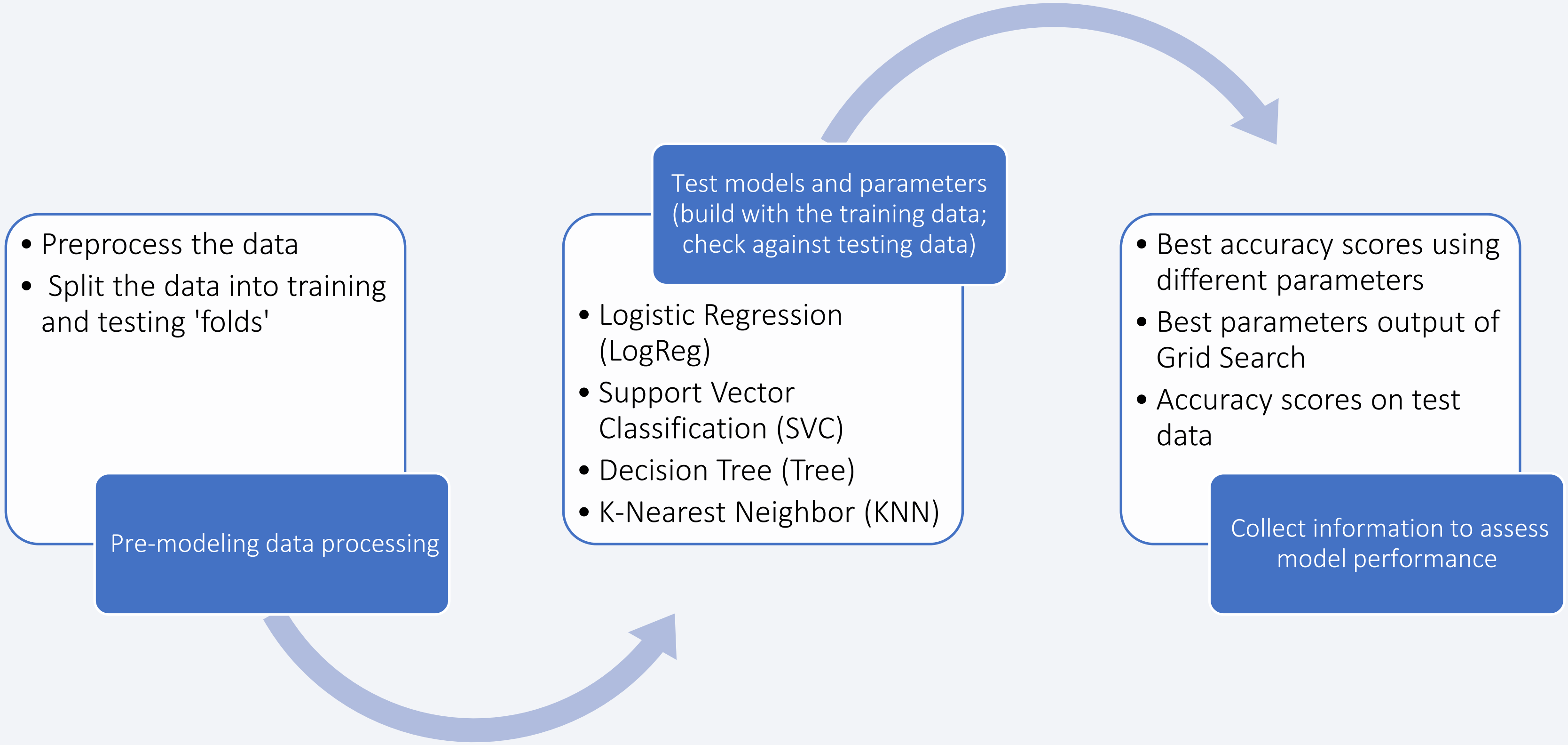
- Using the Python library Folium, a geographic analysis of the launch sites was conducted
  - Marked all launch sites using Folium Circles on a map of the United States, centered on NASA's Johnson Space Center in Houston
  - Marked all the successful and failed launches for each site on the map using color-indexed Folium markers
  - Calculated the distances between a launch site to important proximities like railways, highways, and coastlines – and marked an example with a line
- With these visual cues, it was easier to analyze the relationship between launch sites and success rates, as well was what factors might influence launch site selection
- Jupyter Notebook #6 contains the interactive maps created with Folium

Build a Dashboard with Plotly Dash

- Using the Python libraries Plotly and Dash, a Space X Launch Records Dashboard was developed.
- Two plots displayed data on all launch sites (or one of the four, depending on user selection)
  - Total successful launches were shown in a pie chart
  - Total successful launches were shown by payload mass and booster version
- These visualizations of the success rates, and the ability to filter them by launch site, made it easy to see patterns by launch site and payload sizes for launches.
- The completed Plotly Dash lab code can be found in Python script file #7

Predictive Analysis (Classification)

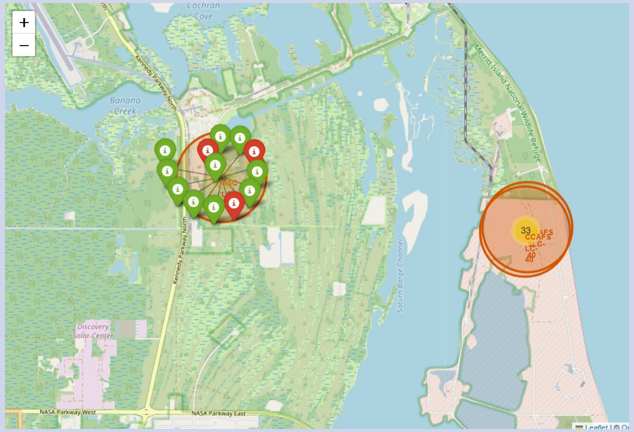
- Multiple modeling techniques and strategies were explored to determine the best predictor of launch success or failure. The process followed a standard flow:



- Jupyter Notebook #8 contains the code for predictive analysis

Results

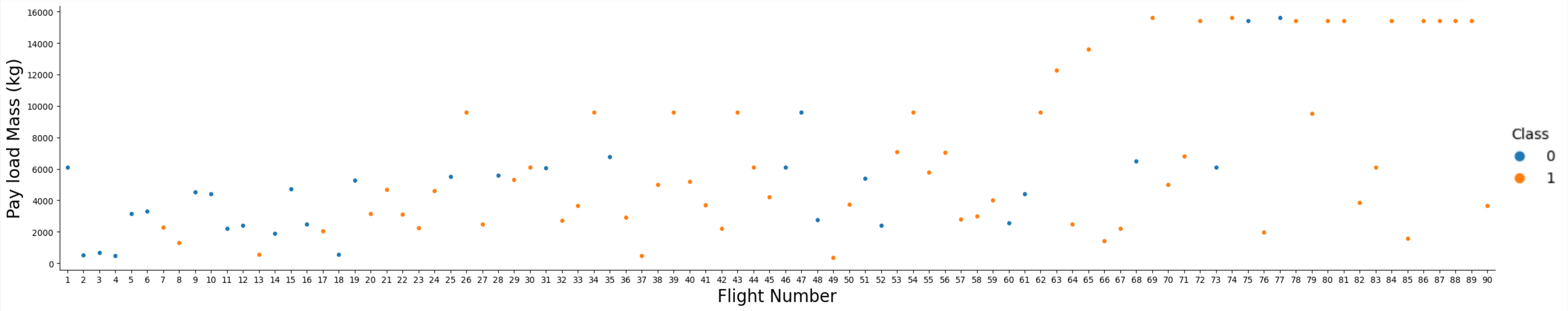
Exploratory Data Analysis	Interactive Analytics Demo	Predictive Analysis

• EDA identified the following features as potentially influential on successful landing outcomes:
  • Payload Mass
  • Orbit
  • Launch site
  • Flight number (experience)
  • Booster Version
	• Launch sites are always located near major waterways (i.e. oceans) and have easy access to railways and highways
	• Launch cites in Florida have higher success rates.
	
		• The model technique with the most accurate predictions is Decision Tree
		  • 94% Accuracy Score
		• Using a Decision Tree Model, we can predict the success or failure of a Falcon9 landing based on a number of key inputs
		• The model will help determine the overall cost of a launch

Section 2

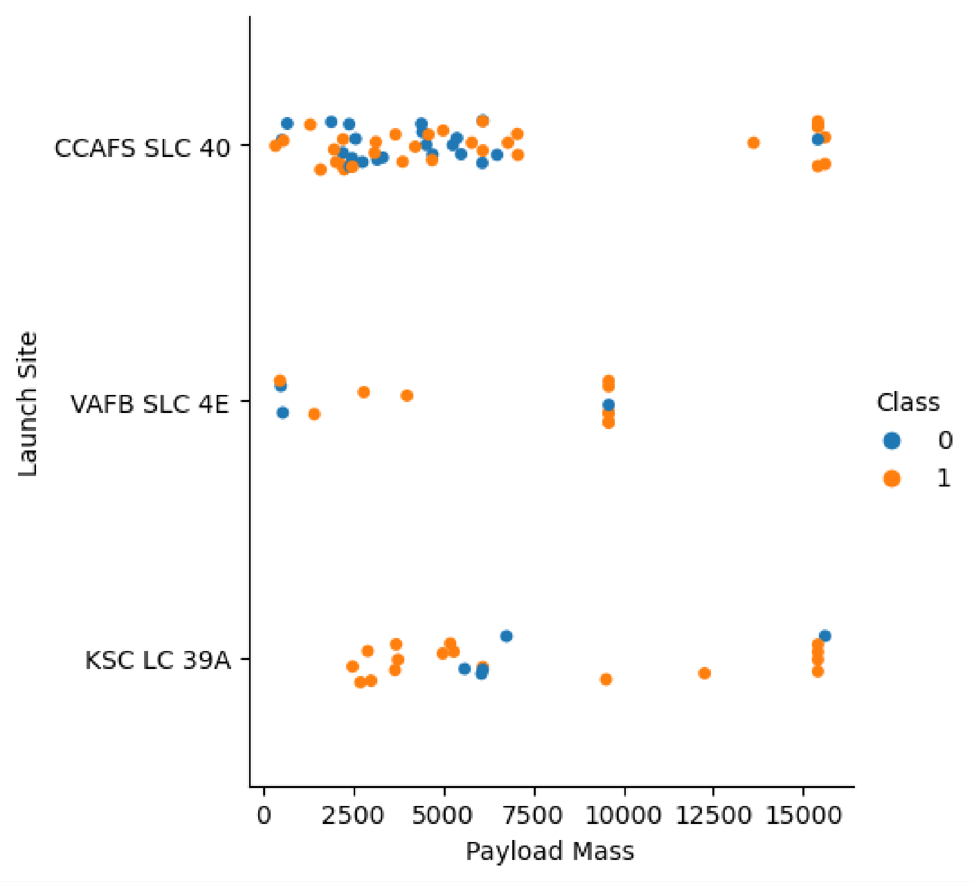
Insights drawn from EDA

Flight Number vs. Launch Site



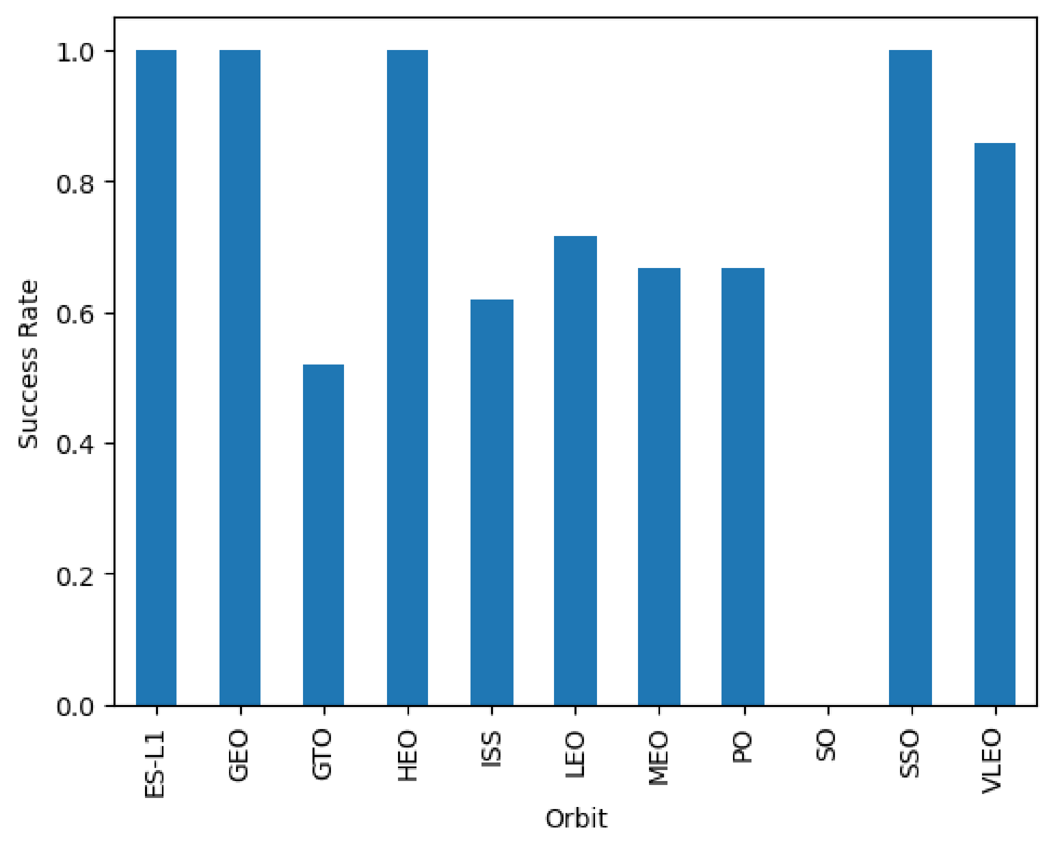
- In a plot that maps the size of the rocket's payload in kgs over flight number, it is clear that there has been an increase in the max payload size over time, with payloads becoming larger and larger until achieving approx. 10,000 kg around flight 70
- Additionally, it looks like the dots trend towards orange (successful launches) over blue (failed launches), indicating an improvement in success rates as more launches have taken place

Payload vs. Launch Site



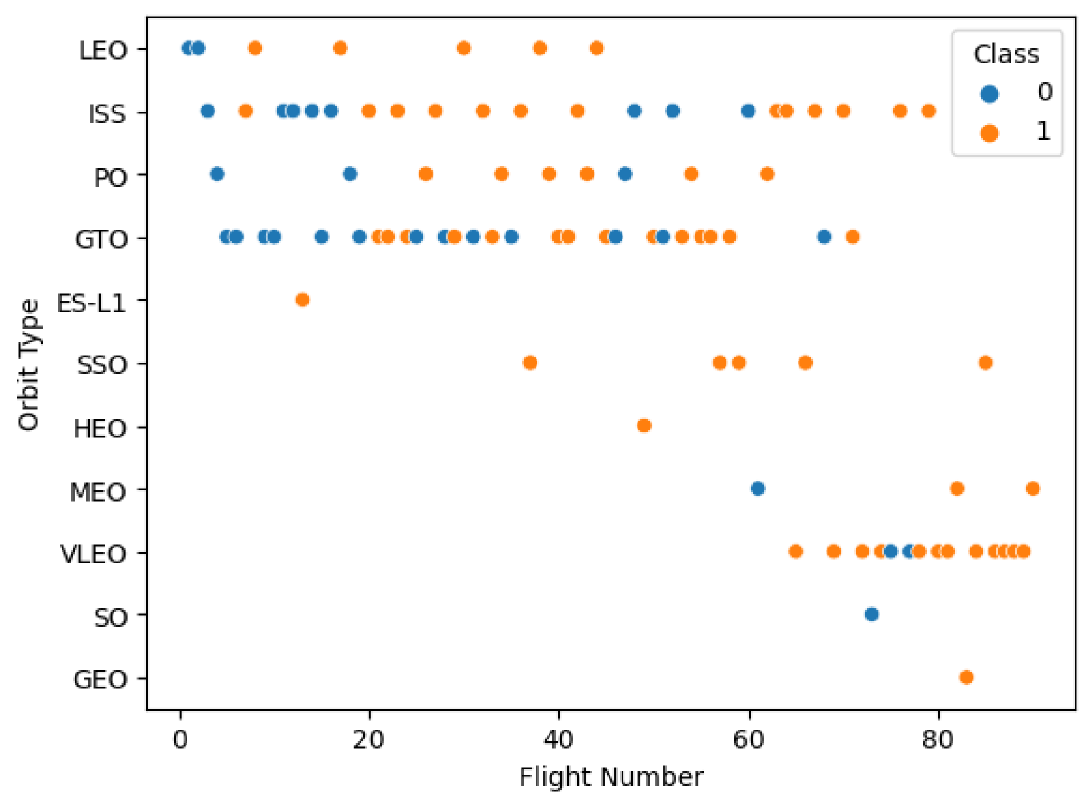
- Comparing the 3 launch sites, it seems like CCAFS SLC 40 (Cape Canaveral, Florida) has had the most launches, including some of the highest payloads
- VAFB SLC 4E (Vandenberg) has trended towards lower payload sizes, on average
- The relationship between payload mass and launch success is unclear, as both sites have a mix – with higher payloads generally having higher success rates than lower payloads at all sites.

Success Rate vs. Orbit Type



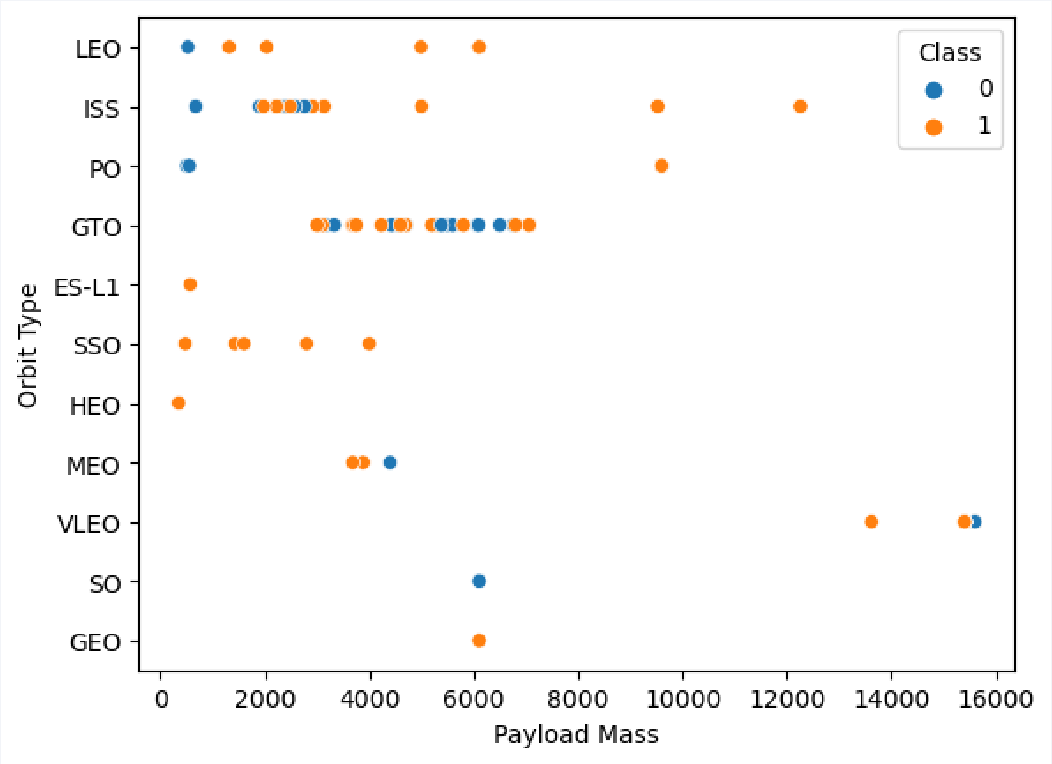
- The success rates are higher for orbits that are father from earth (see Appendices A3 & A4 for a description of orbit types):
  - LEO (Low Earth Orbit) and MEO have lower success rates than GEO and HEO
  - GTO had the lowest success rate, apart from SO (which is similar to SSO)
  - These success rates would be more informative if weighted by the number of times the orbit was attempted – the ones with lower success rates have been attempted more often.
- From this graph, it seems there is a relationship between the type of orbit and the success rate

Flight Number vs. Orbit Type



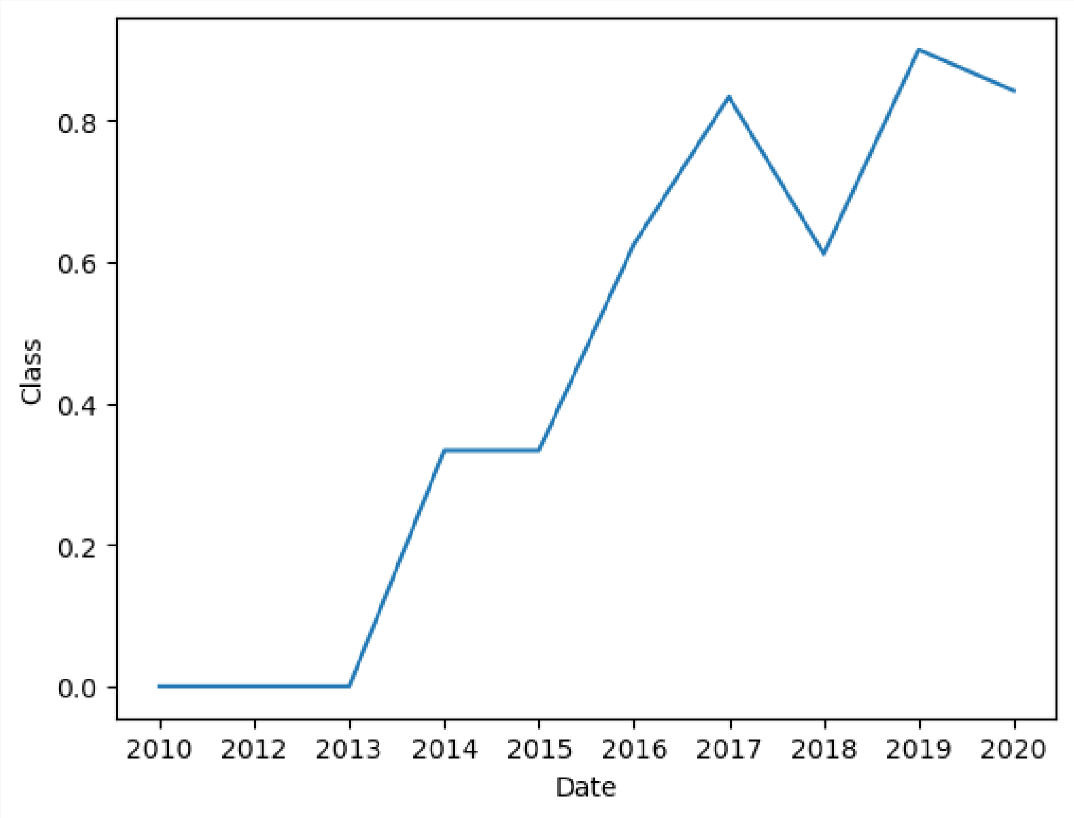
- SpaceX has attempted new orbit types as it has conducted more flights:
  - Early flights focused on LEO (Low Earth Orbit), ISS, and GTO orbits
  - With more flights under their belt, SpaceX began conducting launches to ES-L1, SSO, HEO, MEO, VLEO, SO, and GEO
  - While some orbits have only been attempted once, VLEO has become a common orbit after flight no. 60
- While there is a relationship between flight number and orbit type, it does not seem overly strong

Payload vs. Orbit Type



- Payload mass does affect which orbits are attempted:
    - VLEO has only been attempted for payloads higher than 13,000 kg
    - Payloads for GTO orbit are always between 2,000 kg and 8,000 kg
    - ISS orbit (for International Space Station) has the widest range of payloads, from about 1,000 kg to over 12,000 kg.
- There seems to be a strong relationship between orbit type and payload mass – some orbit types are only attempted if under a payload threshold.

Launch Success Yearly Trend



- SpaceX has improved its launch capabilities over time, resulting in higher success rates year over year
- From 210 to 2013, all landings were unsuccessful
- Starting in 2014, SpaceX began successfully landing Falcon9 rockets
- Over time, SpaceX has improved its annual success rate to about 80%

All Launch Site Names

SQL Query (in Jupyter Notebook with SQL Magic)
%%sql select distinct Launch_Site from SPACEXTABLE

[Output]:

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

- This SQL query extracts a list of unique launch site values
- Note: Cape Canaveral is represented by two values: CCAFS LC-40" and "CCAFS SLC-40"

Launch Site Names Begin with 'CCA'

SQL Query (in Jupyter Notebook with SQL Magic)

```
%%sql
select * from SPACEXTABLE
where Launch_Site like 'CCA%'
limit 5
```

[Output]:

Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Orbit	Customer	Mission_Outcome	Landing_Outcome
F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success	Failure (parachute)
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)
F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success	No attempt
F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500	LEO (ISS)	NASA (CRS)	Success	No attempt
F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677	LEO (ISS)	NASA (CRS)	Success	No attempt

- This query identifies the first 5 launches that took place in Cape Canaveral
  - See Appendix A6 for a complete table with all columns.

Total Payload Mass

SQL Query (in Jupyter Notebook with SQL Magic)

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

[Output]:

Nasa_CRS_payload_sum
45596

- This query calculates the total payload amount launched by NASA (CRS) over all launches

Average Payload Mass by F9 v1.1

SQL Query (in Jupyter Notebook with SQL Magic)

```
%%sql
select avg(PAYLOAD_MASS__KG_) as f9_1v1_payload_avg
from SPACEXTABLE
where Booster_Version like 'F9 v1.1%'
```

[Output]:

f9_1v1_payload_avg
2534.6666666666665

- This query calculates the average payload for a Falvon9 rocket that's version 1.1

First Successful Ground Landing Date

SQL Query (in Jupyter Notebook with SQL Magic)
%%sql select min(Date) from SPACEXTABLE where Landing_Outcome = 'Success (ground pad)'

[Output]:

min(Date)
2015-12-22

- This query returns the date of the first successful landing on a ground pad

Successful Drone Ship Landing with Payload between 4000 and 6000

SQL Query (in Jupyter Notebook with SQL Magic)

```
%%sql
select Booster_Version from SPACEXTABLE
where Landing_Outcome = 'Success (drone ship)'
and PAYLOAD_MASS__KG_ > 4000
and PAYLOAD_MASS__KG_ < 6000
```

[Output]:

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

- This query returns a list of versions of the Falcon9 rocket that have successfully landed on a drone ship after carrying payloads greater than 4,000kg and less than 6,000kg

Total Number of Successful and Failure Mission Outcomes

SQL Query (in Jupyter Notebook with SQL Magic)

```
%%sql
 select distinct
        (select count(Mission_Outcome) from SPACEXTABLE
        where Mission_Outcome Like '%Success%’)
as success_count,
        (select count(Mission_Outcome)
        from SPACEXTABLE
        where Mission_Outcome Like '%failure%’)
as failure_Count FROM SPACEXTABLE
```

[Output]:

success_count	failure_Count
100	1

- This query returns a total count of both successful and failed mission outcomes

Boosters Carried Maximum Payload

SQL Query (in Jupyter Notebook with SQL Magic)

```
%%sql
select distinct Booster_Version from SPACEXTABLE
where PAYLOAD_MASS__KG_ =
        (select max(PAYLOAD_MASS__KG_)
        from SPACEXTABLE)
```

[Output]:

Booster_Version	
F9 B5 B1048.4	F9 B5 B1049.5
F9 B5 B1049.4	F9 B5 B1060.2
F9 B5 B1051.3	F9 B5 B1058.3
F9 B5 B1056.4	F9 B5 B1051.6
F9 B5 B1048.5	F9 B5 B1060.3
F9 B5 B1051.4	F9 B5 B1049.7

- This query returns a list of the Falcon9 versions that have carried ever carried the maximum payload

2015 Launch Records

SQL Query (in Jupyter Notebook with SQL Magic)

```
%%sql
select substr(Date, 6, 2) as month,
substr(Date, 0, 5) as year,
Booster_Version, Launch_Site
from SPACEXTABLE
where landing_outcome = "Failure (drone ship)“
and year = '2015'
```

[Output]:

month	year	Booster_Version	Launch_Site
01	2015	F9 v1.1 B1012	CCAFS LC-40
04	2015	F9 v1.1 B1015	CCAFS LC-4

- This query returns a list of all 2015 launch records with unsuccessful landings on a drone ship, including the launch site and version of the Falcon9 booster used

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

SQL Query (in Jupyter Notebook with SQL Magic)

```
%%sql
select launch site, count(‘Mission_Outcome’)
as MISSION_OUTCOME_COUNT
where date between '2010-06-04' and '2017-03-20'
from SPACEXTABLE
group by Launch_Site
order by MISSION_OUTCOME_COUNT DESC
```

[Output]:

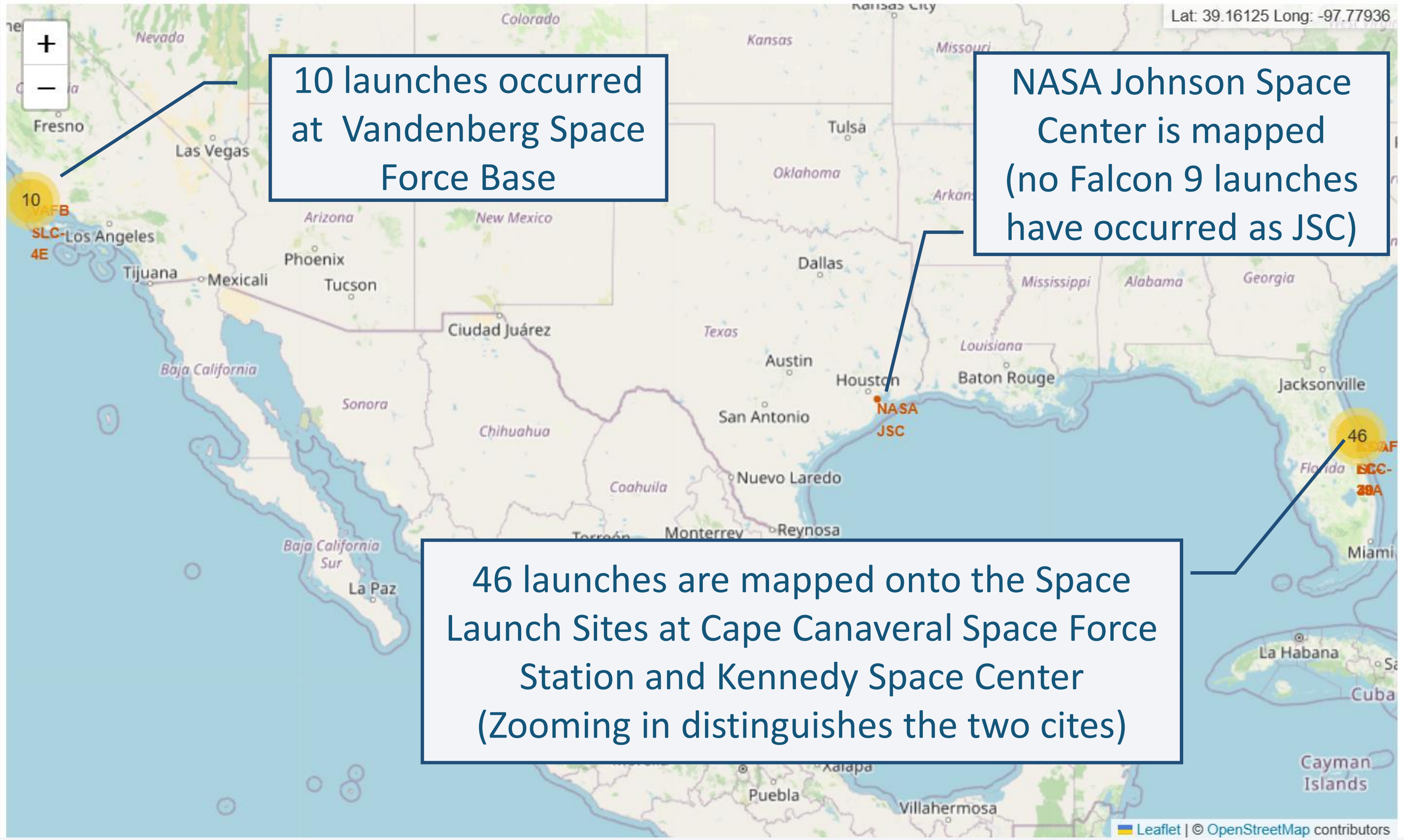
Launch_Site	MISSION_OUTCOME_COUNT
CCAFS LC-40	26
VAFB SLC-4E	3
KSC LC-39A	2

- This query returns a list of launch sites and a count of the launches/missions conducted at each site, in descending order by mission count, for launches that occurred between June 4, 2010 and March 20, 2017

Section 3

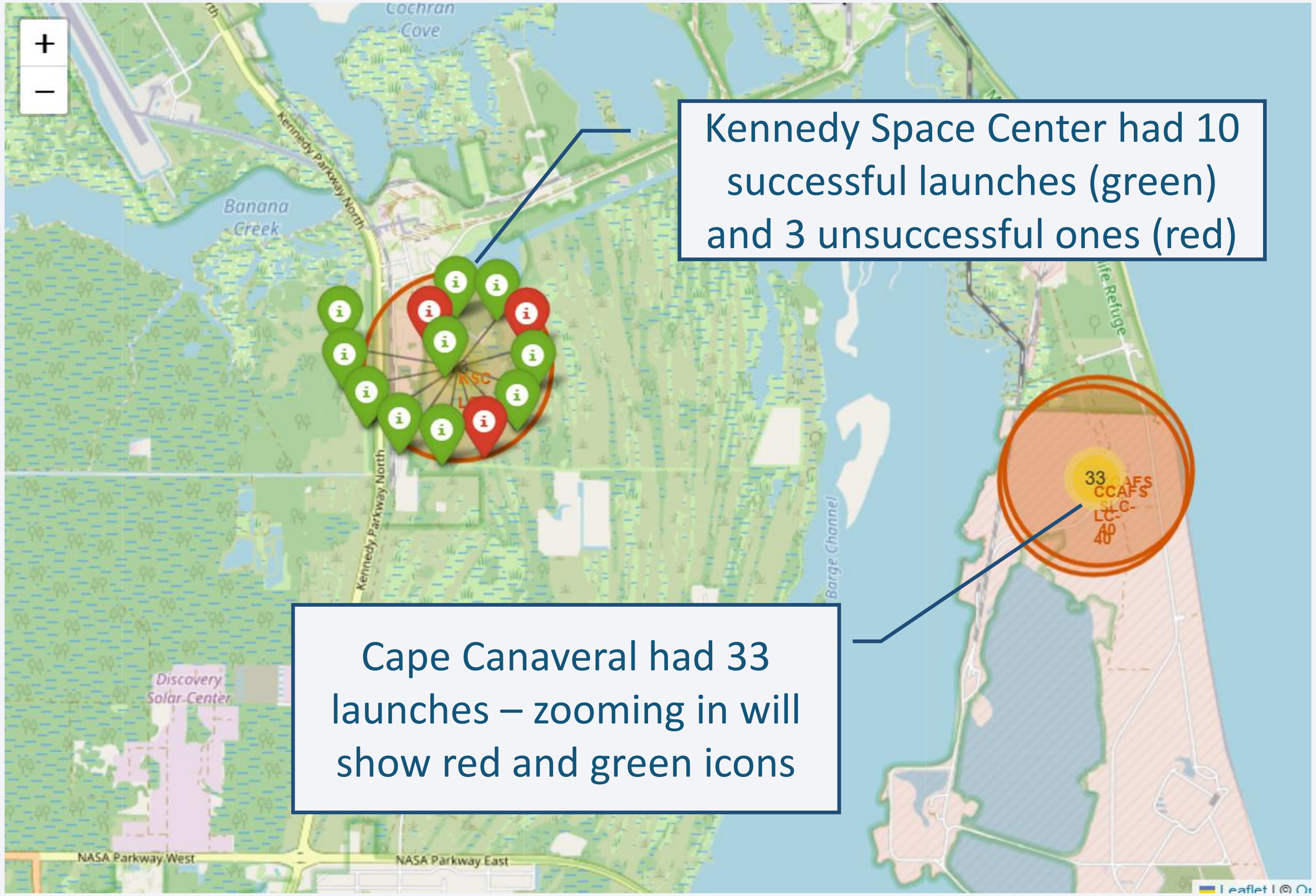
Launch Sites Proximities Analysis

Folium maps key locations for Space X Launches



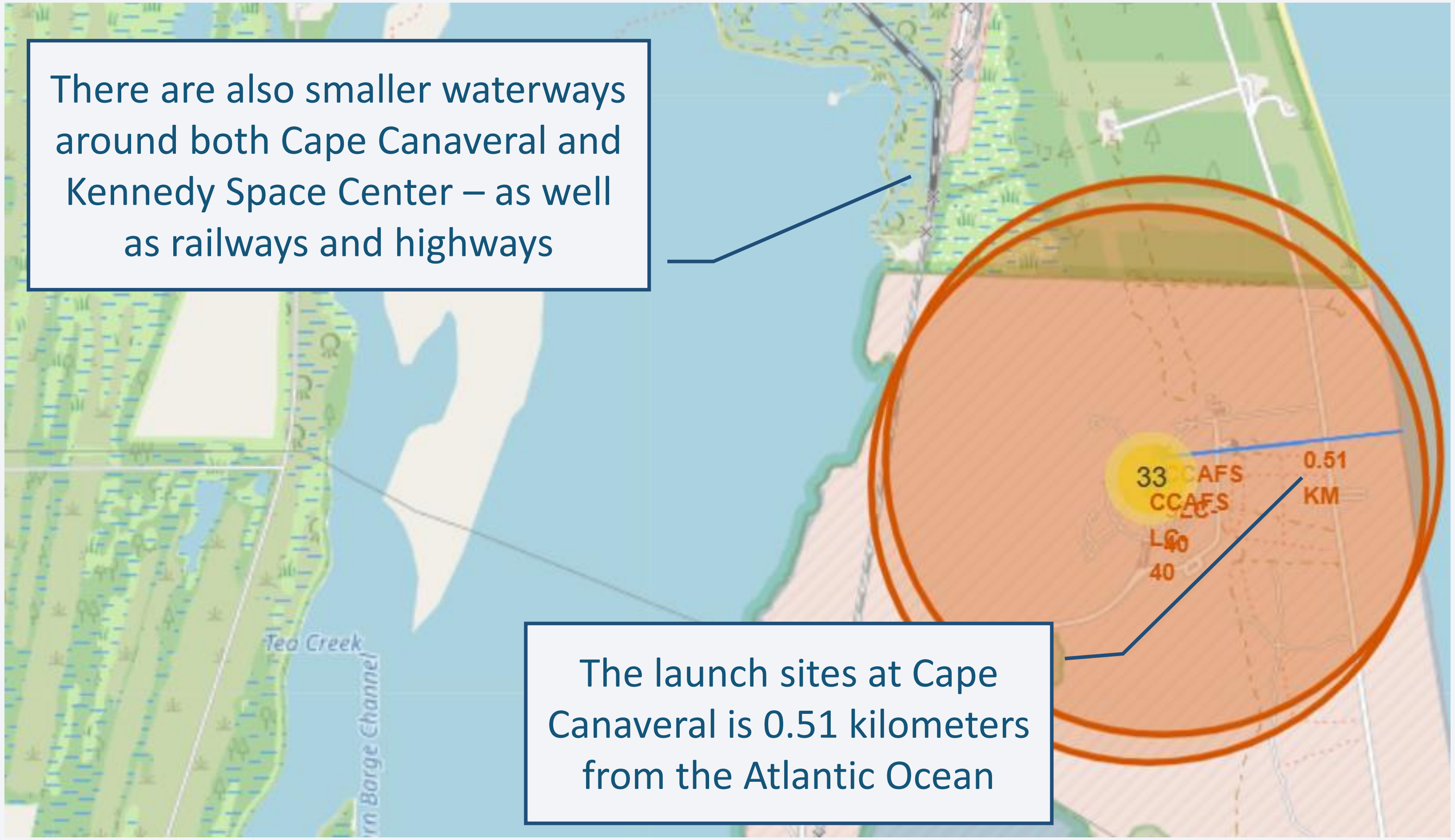
- All launch sites located in the southern United States
- Most have occurred in Central Florida, implying that the warmer climates are worth potential interference by stormy weather

Individual launches are mapped and color-coded



- The folium map was modified to have markers for each launch, color-coded to visually represent success or failure
- As a user zooms in (or out), they see more (or grouped) markers.

Folium also maps proximity to key geographic features

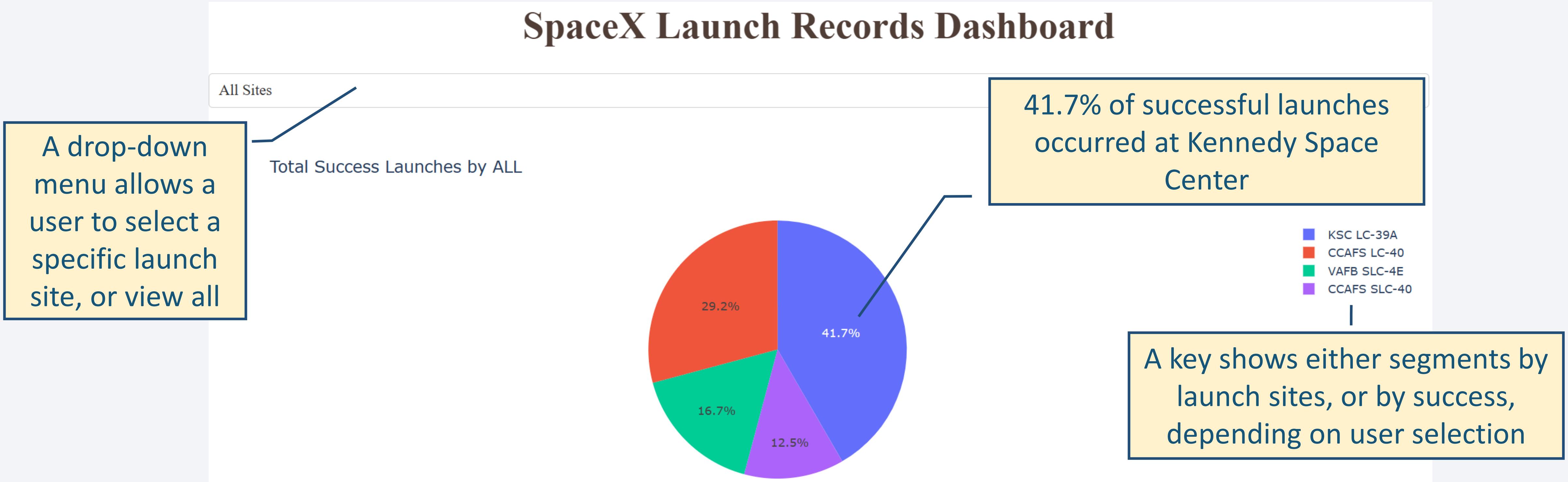


- All 3 launch sites are close to water or a coastline
- All 3 launch sites also are close to highways and railways
- This can be visually represented and explored using a Folium map

Section 4

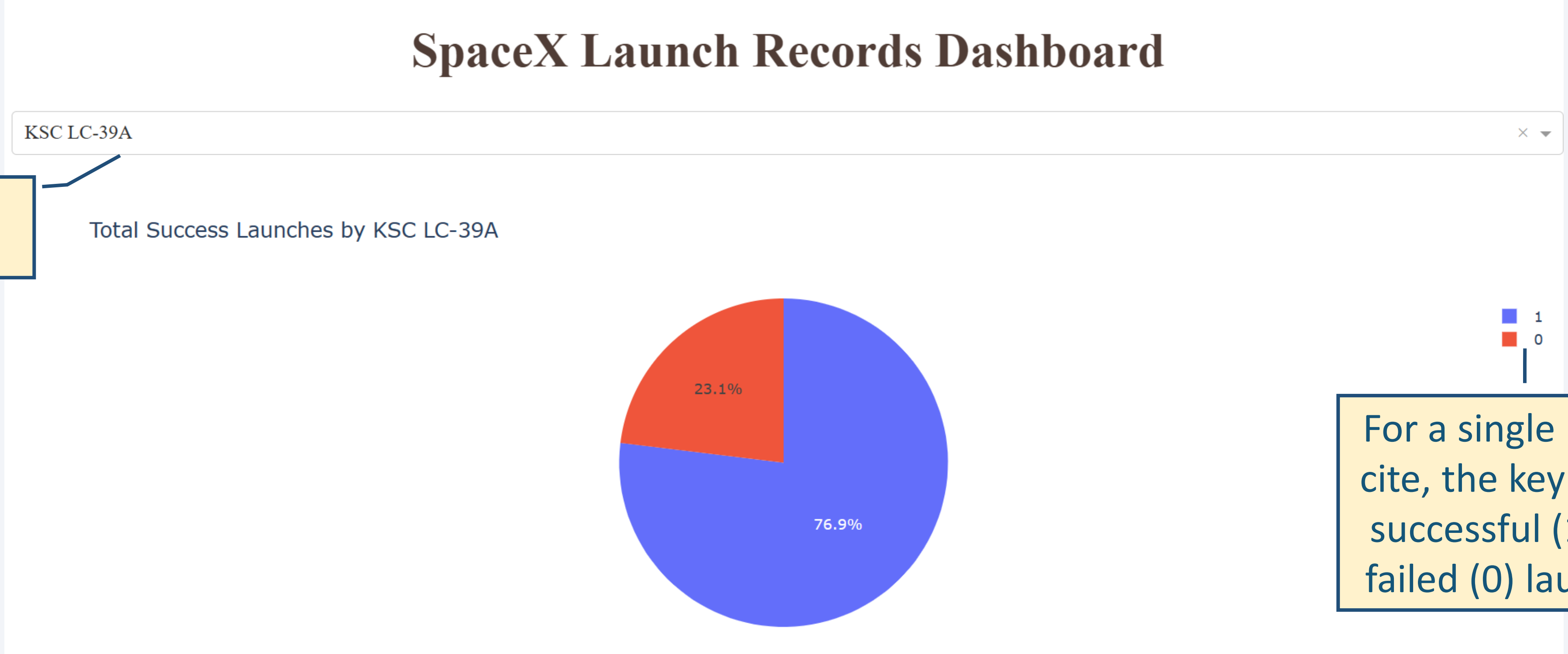
Build a Dashboard with Plotly Dash

A dashboard easily visualizes launch site success rates



- The majority of successful launches have occurred in Central Florida, either at Cape Canaveral (~**41.6%**) or Kennedy Space Center (~**41.7%**)
- Vandenberg Space Force Base is responsible for ~**16.7%** of successful launches

Individual Site Information can be seen using a filter

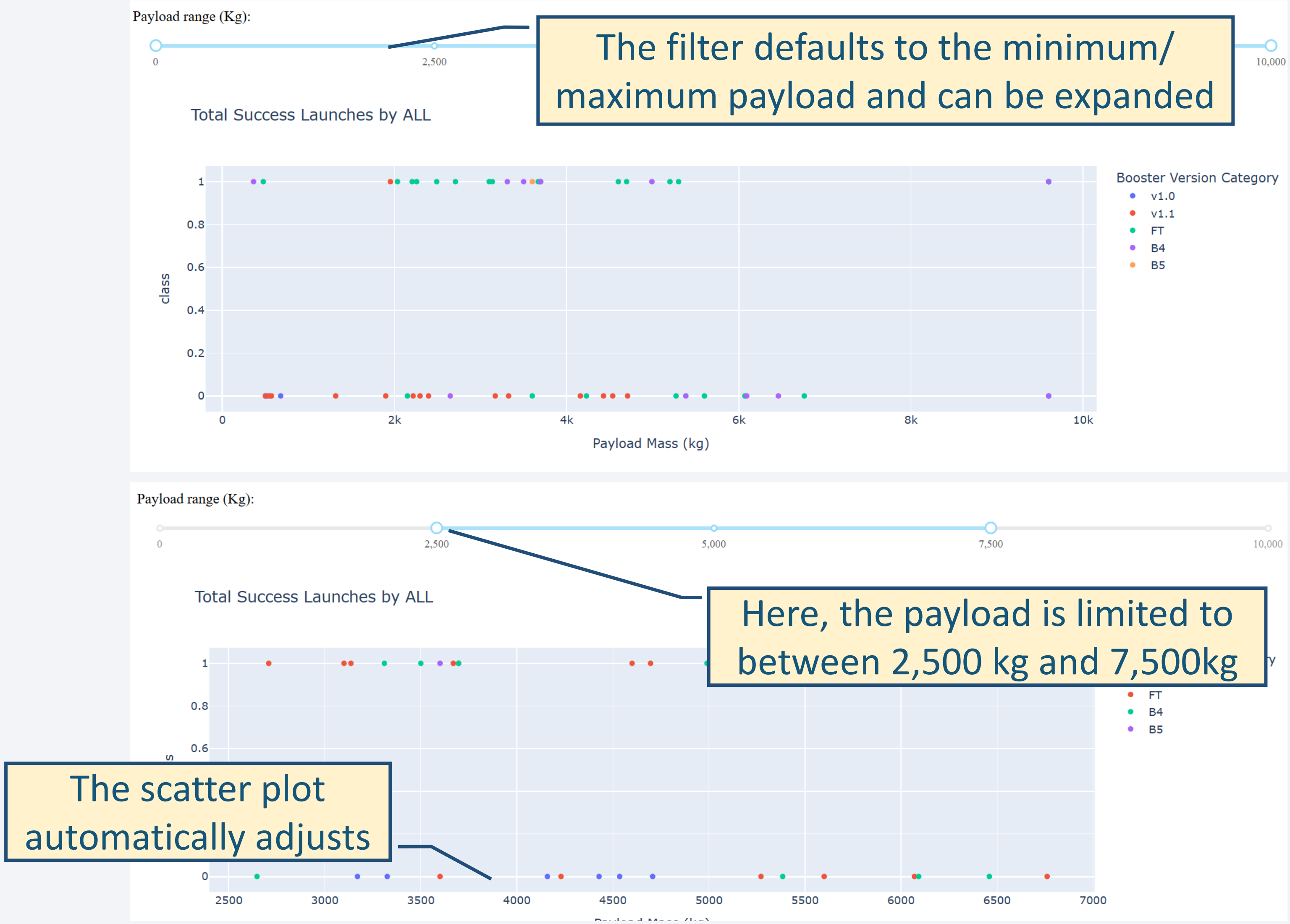


A user selects KSC LC-39A

For a single launch cite, the key shows successful (1) and failed (0) launches

- At **76.9%**, Vandenberg Space Force Base has the highest launch success ratio of all launch sites
- Dashboard results for the other launch sites are available in Appendices A8, A9, A10, & A11

A scatter plot maps successes by Booster Version



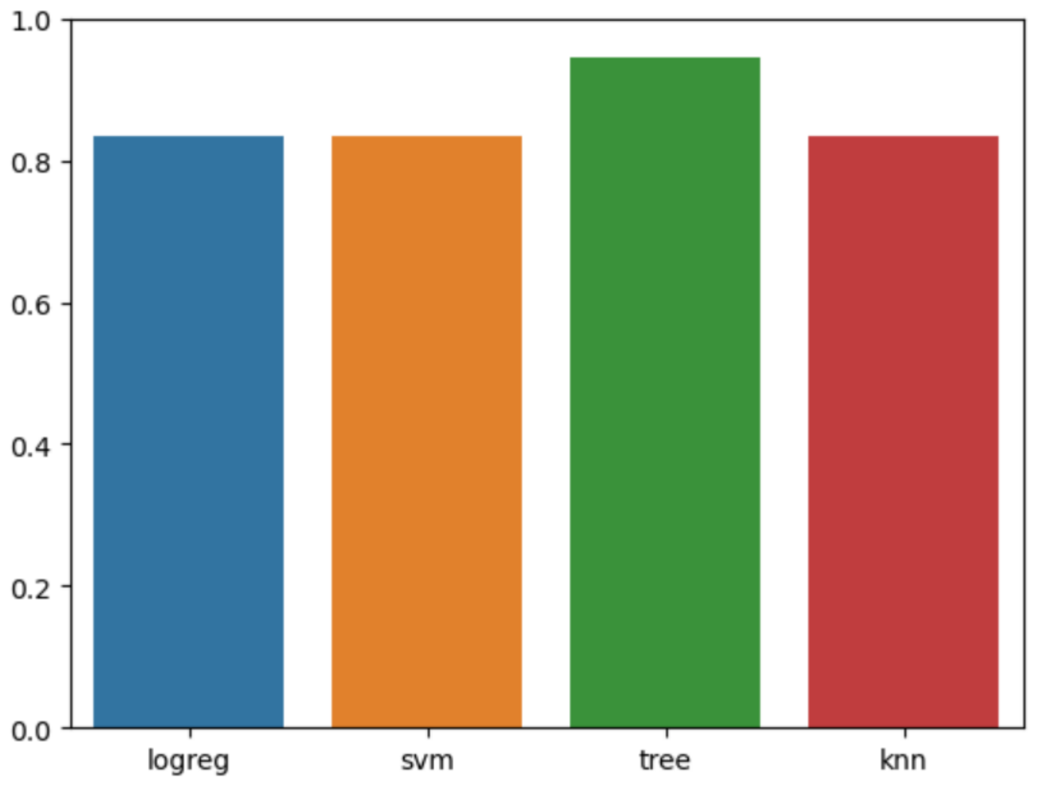
- A second plots allows further examination of the selected launch site(s)
- This graph shows more information and can be adjusted by a slider
- Color-coded dots indicate that the “FT” Booster version has had the most successful launches of all versions, while “v1.1” has had the least
- Over time, launches have shifted away from “v1” and “v1.1” and presumably towards newer versions

Section 5

Predictive Analysis (Classification)

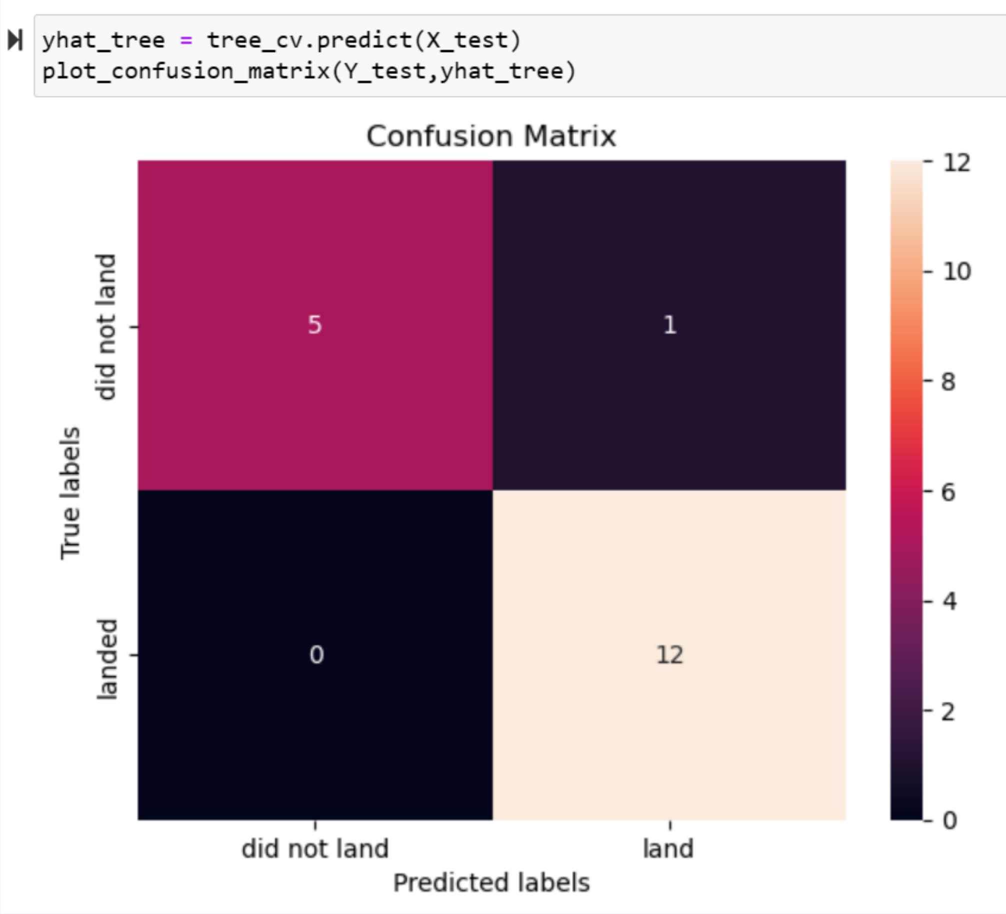
Classification Accuracy

Accuracy Scores for Test Data



- Accuracy scores were calculated for each of the 4 model types:
  - Logistical Regression (Appendix A13)
  - Support Vector Machine/Classification (Appendix A14)
  - Decision Trees (Appendix A15)
  - K-Nearest Neighbor (Appendix A16)
- Additionally, for each of the Grid Searches conducted to identify the strongest version of a specific model type, we identified:
  - The optimal parameters
  - That version's accuracy score
- The Decision Tree modeling technique had the highest Accuracy Score, at **0.94**

Confusion Matrix – Decision Tree



- The Decision Tree technique produced the most accurate model, with a test accuracy score of 0.944444
  - The best parameters, per the grid search, were:
    - criterion: 'gini’
    - max_depth: 16
    - max_features: ‘sqrt’
    - min_samples_leaf: 2
    - min_samples_split: 10
    - splitter: 'random'
  - The best accuracy score was 0.873214
- As shown in the confusion matrix, the model had only one false positive
  - 5 landing failures were predicted correctly
  - 12 landing successes were predicted correctly

Conclusions

- There is a relationship between the success of a launch and its payload, orbit, and launch site.
  - The flight number (indicative of experience) is also correlated to success – with larger flight numbers having higher success rates
  - Booster version is also correlated with success, though there is a correlation between booster version and experience (flight number)
- Launch sites are always located near major waterways (i.e. oceans)
  - Easy access to railways and highways is also prioritized
  - Launch cites in Florida have higher success rates
- The model technique with the most accurate predictions is Decision Tree
- Using a Decision Tree Model, we can predict the success or failure of a Falcon9 landing based on a number of key inputs – which will help determine the overall cost of a launch

Appendix

The following 18 slides include relevant assets like Python code snippets, SQL queries, charts, Notebook outputs, or data sets that were created during this project and provide useful, if more detailed, information on the items discussed in the report.

List of Appendix Slides	
1. Data Dictionary 2. List of All Launch Sites 3. Types of Orbits 4. Diagram of 4 Possible Orbits 5. Detailed Types of Launch Outcomes 6. First 5 CCA Launches (Full Table) 7. Dashboard for All Launch Sites 8. Dashboard for CCAFS LC-40 9. Dashboard for CCAFS SLC-40	10. Dashboard for KSC LC-39A 11. Dashboard for VAFB SLC-4E 12. Summary Table of Model Outcomes 13. Confusion Matrix – Logarithmic Regression 14. Confusion Matrix – SVM 15. Confusion Matrix – Decision Tree 16. Confusion Matrix – K-Nearest Neighbors 17. Bar Chart of Accuracy Scores (Test) 18. Bar Chart of Accuracy Scores (Best GridSearch)

A1: Data Dictionary

Variable	Data Type	Description
FlightNumber	Int64	A launch’s flight number, starting at 1 and ending at 90
Date	Object	The data a launch occurred; formatted as YYYY-MM-DD
BoosterVersion	Object	The specific version of the booster that was used for the launched
PayloadMass	Float64	The size or mass, in kilograms, of the payload sent to space in a launch
Orbit	Object	The type of orbit that for a specific launch (11 values)
LaunchSite	Object	The location from which a rocket was launched (4 values for 3 space launch centers)
Outcome	Object	Indicates the outcome of the launch – a successful landing or failed one
Flights	Int64	The number of flights with this core (Falcon 9 Rocket)
GridFins	Bool	Indicated whether a GridFin was used (True) or not (False)
Reused	Bool	Indicated whether the first stage rocker was reused (True) or not (False)
Legs	Bool	Indicates whether legs were used (True) or (Not)
LandingPad	Object	The landing pad used; “None” in instances where a landing failed.
Block	Float64	The block number of the core (Falcon 9 rocket), which is used to separate version of cores
ReusedCount	Int64	The number of times the first-stage rocket has been reused
Serial	Object	The serial number of the core (Falcon 9 rocket) used in a launch
Longitude	Float64	The longitude of the Launch Site
Latitude	Float64	The latitude of the Launch Site
Class	Float64	Binary value based on ‘Outcome’ that generalizes outcomes details: success (1) and failure (0)

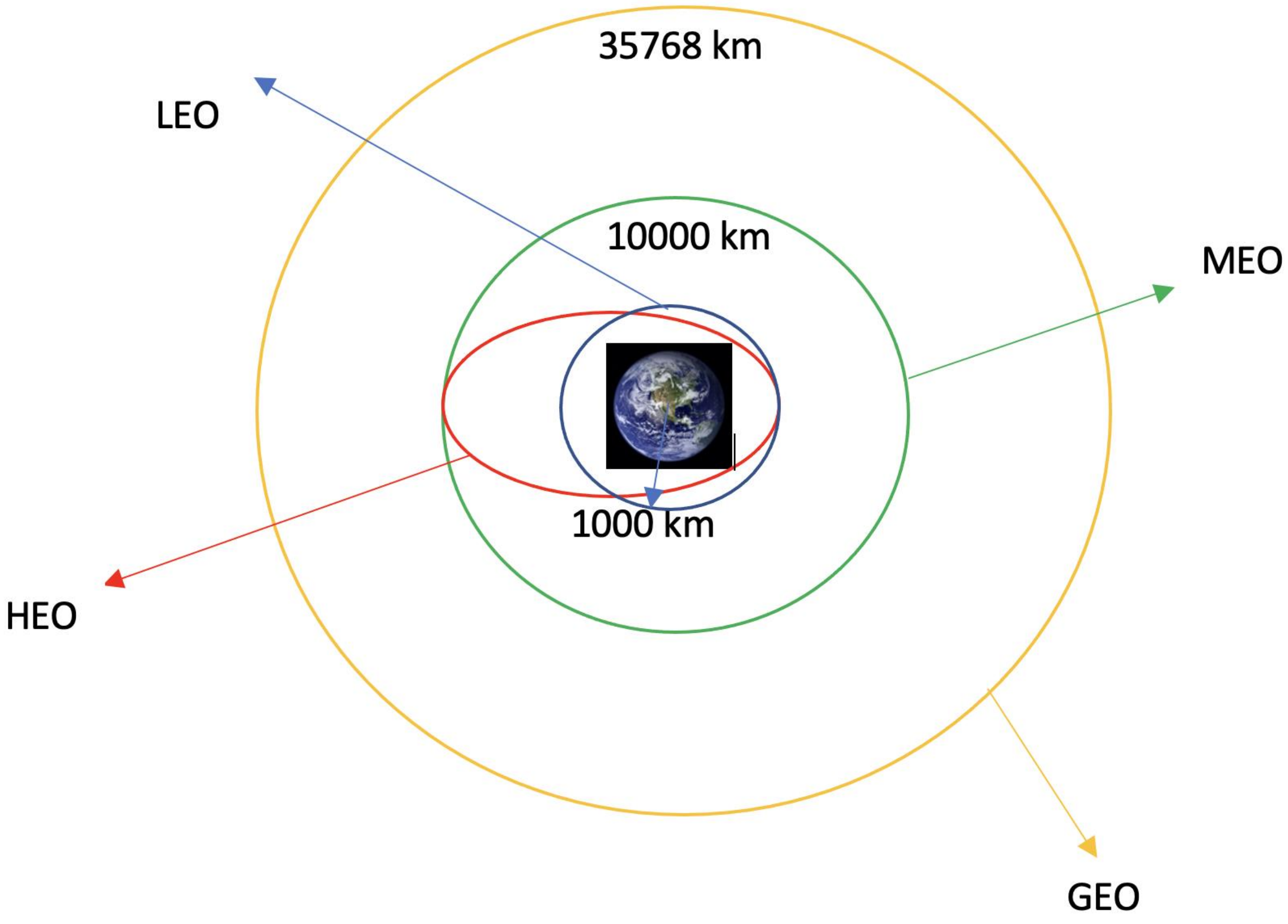
A2: List of All Launch Sites

Value in Data	Launch Site Name	Location (State)	Count
CCAFS SLC 40	Cape Canaveral Space Force Station Space Launch Complex	Florida	55
KSC LC 39A	Kennedy Space Center Launch Complex 39A	Florida	22
VAFB SLC 4E	Vandenberg Air Force Base Space Launch Complex 4E	California	13
		Total:	90

A3: Types of Orbits

- **LEO**: Low Earth orbit (LEO)is an Earth-centred orbit with an altitude of 2,000 km (1,200 mi) or less (approximately one-third of the radius of Earth),[1] or with at least 11.25 periods per day (an orbital period of 128 minutes or less) and an eccentricity less than 0.25.[2] Most of the manmade objects in outer space are in LEO [1].
- **VLEO**: Very Low Earth Orbits (VLEO) can be defined as the orbits with a mean altitude below 450 km. Operating in these orbits can provide a number of benefits to Earth observation spacecraft as the spacecraft operates closer to the observation[2].
- **GTO** A geosynchronous orbit is a high Earth orbit that allows satellites to match Earth's rotation. Located at 22,236 miles (35,786 kilometers) above Earth's equator, this position is a valuable spot for monitoring weather, communications and surveillance. Because the satellite orbits at the same speed that the Earth is turning, the satellite seems to stay in place over a single longitude, though it may drift north to south," NASA wrote on its Earth Observatory website [3] .
- **SSO (or SO)**: It is a Sun-synchronous orbit also called a heliosynchronous orbit is a nearly polar orbit around a planet, in which the satellite passes over any given point of the planet's surface at the same local mean solar time [4] .
- **ES-L1** :At the Lagrange points the gravitational forces of the two large bodies cancel out in such a way that a small object placed in orbit there is in equilibrium relative to the center of mass of the large bodies. L1 is one such point between the sun and the earth [5] .
- **HEO** A highly elliptical orbit, is an elliptic orbit with high eccentricity, usually referring to one around Earth [6].
- **ISS** A modular space station (habitable artificial satellite) in low Earth orbit. It is a multinational collaborative project between five participating space agencies: NASA (United States), Roscosmos (Russia), JAXA (Japan), ESA (Europe), and CSA (Canada) [7]
- **MEO** Geocentric orbits ranging in altitude from 2,000 km (1,200 mi) to just below geosynchronous orbit at 35,786 kilometers (22,236 mi). Also known as an intermediate circular orbit. These are "most commonly at 20,200 kilometers (12,600 mi), or 20,650 kilometers (12,830 mi), with an orbital period of 12 hours [8]
- **HEO** Geocentric orbits above the altitude of geosynchronous orbit (35,786 km or 22,236 mi) [9]
- **GEO** It is a circular geosynchronous orbit 35,786 kilometres (22,236 miles) above Earth's equator and following the direction of Earth's rotation [10]
- **PO** It is one type of satellites in which a satellite passes above or nearly above both poles of the body being orbited (usually a planet such as the Earth [11]

A4: Diagram of 4 Possible Orbits



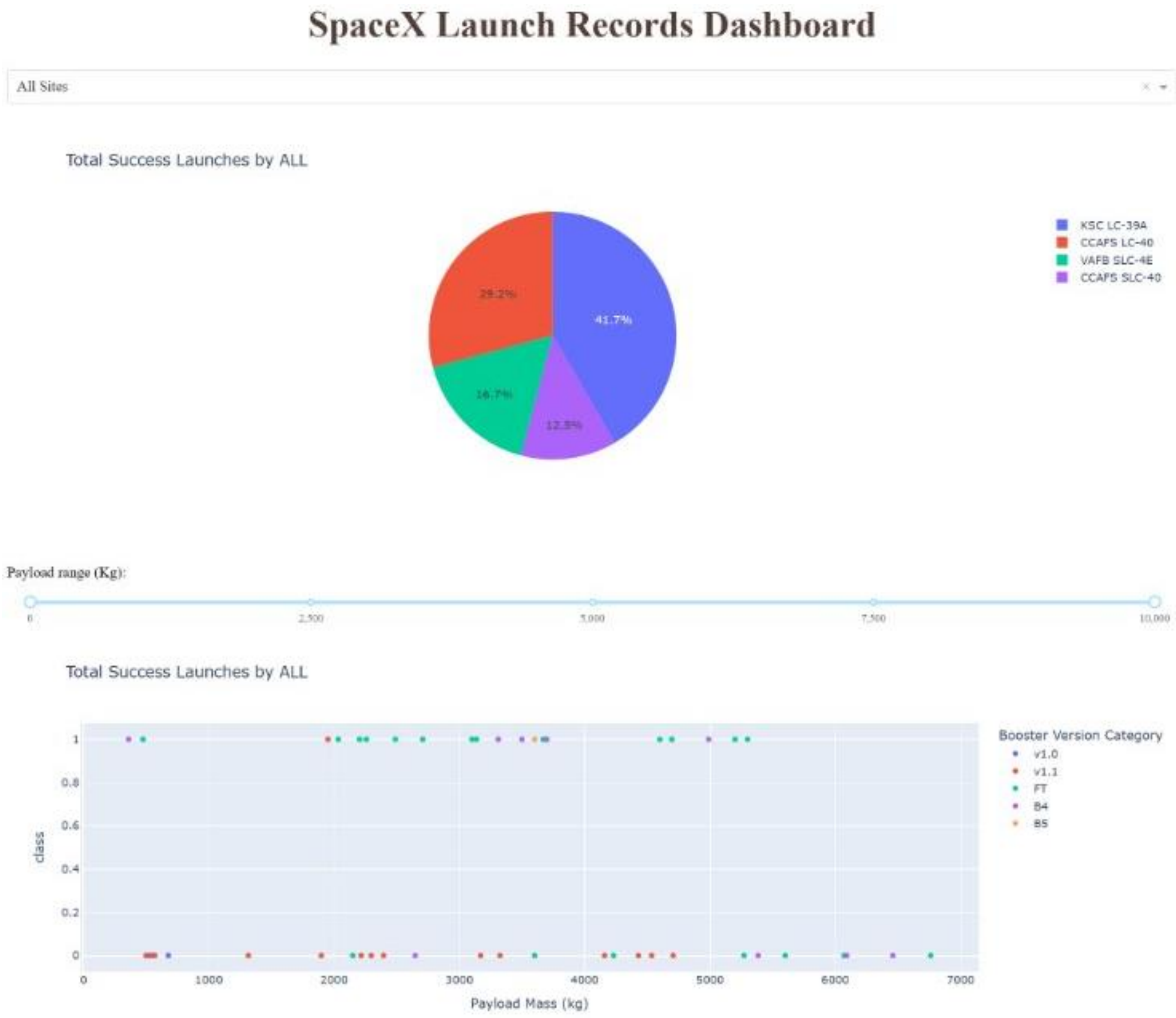
A5: Detailed Types of Launch Outcomes

Value in Data	Description	Count
True ASDS	Successfully landed on a drone ship	41
False ASDS	Unsuccessfully landed on a drone ship	6
None ASDS	Failure to land	2
True RTLS	Successfully landed on a ground pad	14
False RTLS	Unsuccessfully landed on a ground pad	1
True Ocean	Successfully landed on a specific region of the ocean	5
False Ocean	Unsuccessfully landed on a specific region of the ocean	2
None None	Failure to land	19
	Total:	**90**

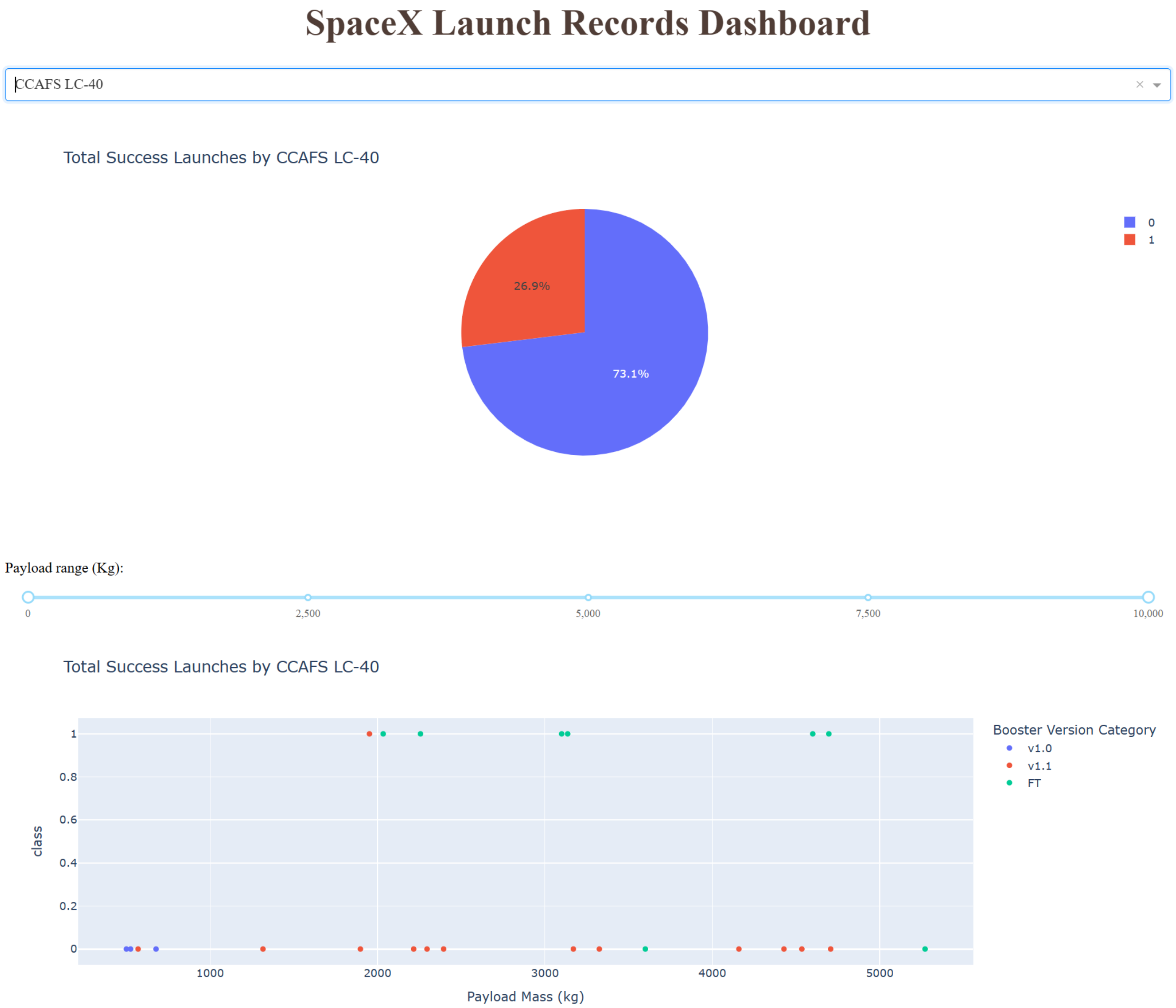
A6: First 5 CCA Launches (Full Table)

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

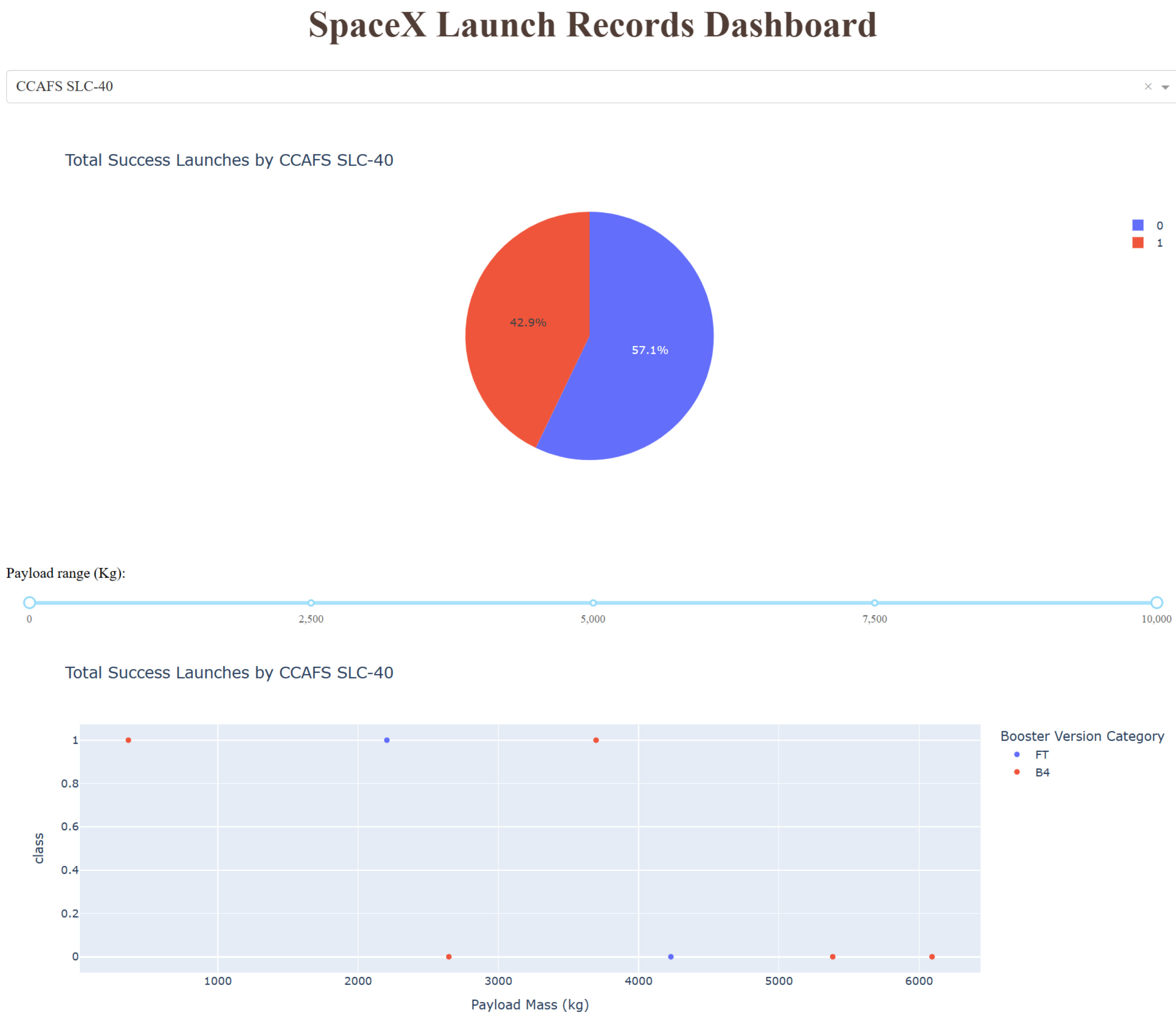
A7: Dashboard for All Launch Sites



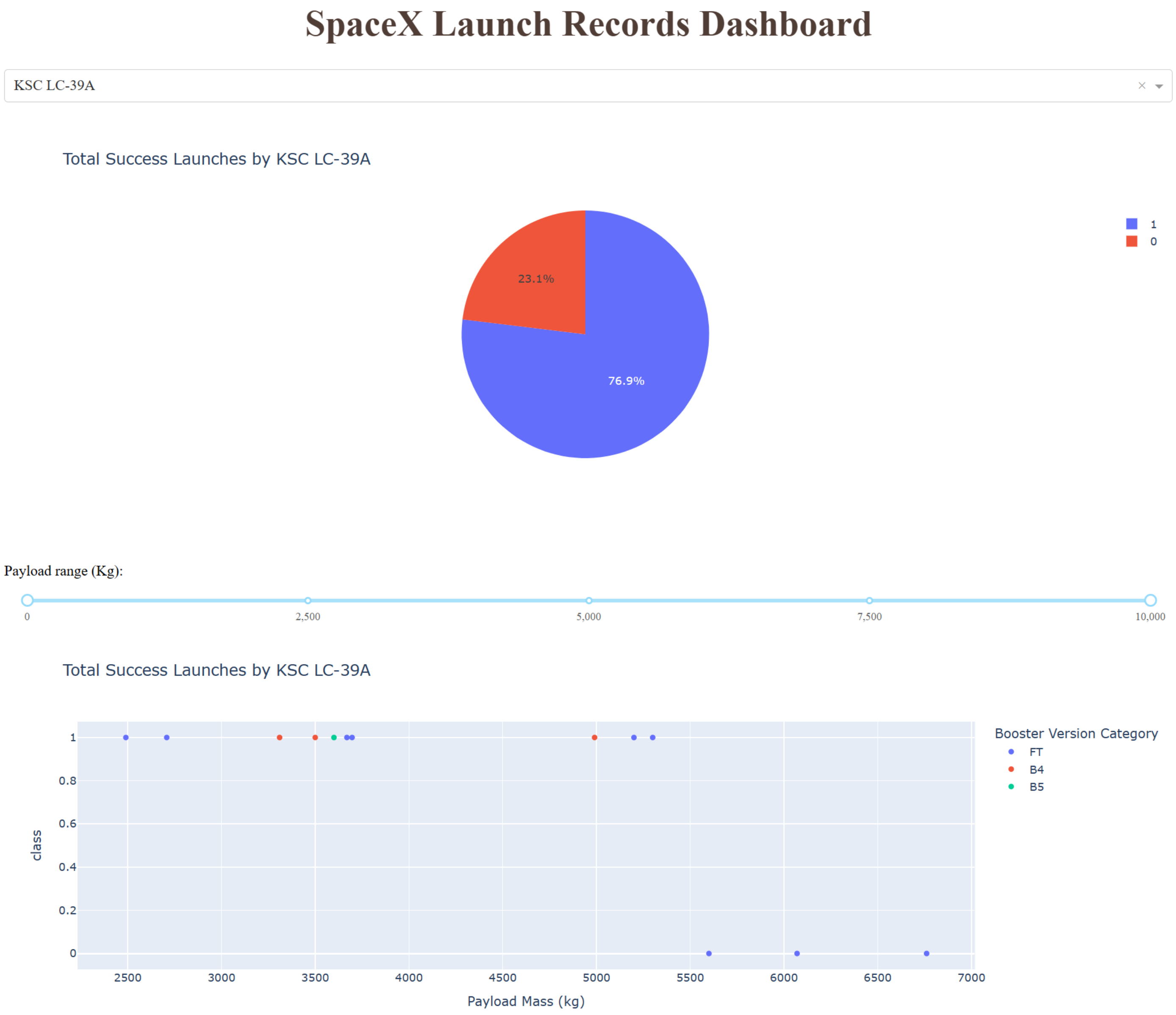
A8: Dashboard for CCAFS LC-40



A9: Dashboard for CCAFS SLC-40

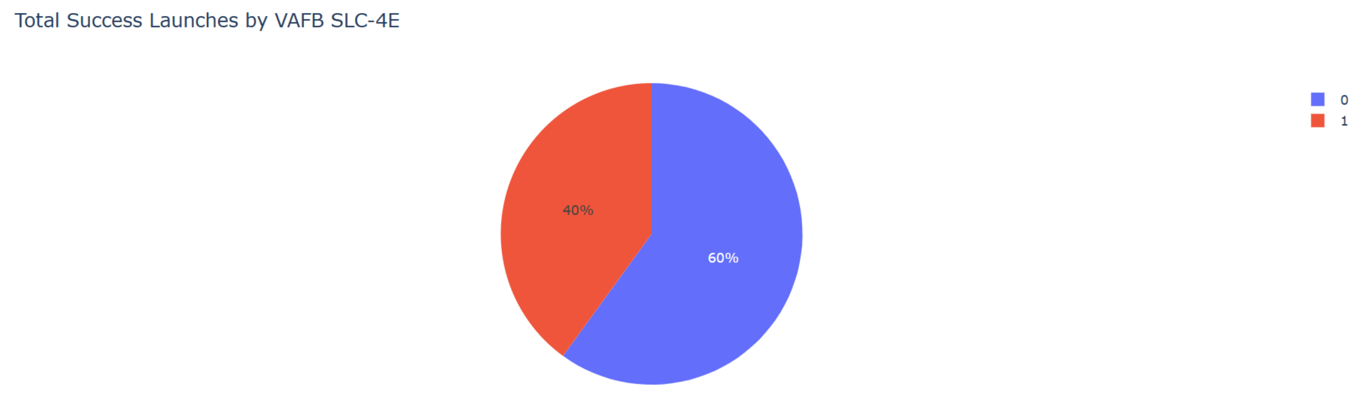


A10: Dashboard for KSC LC-39A



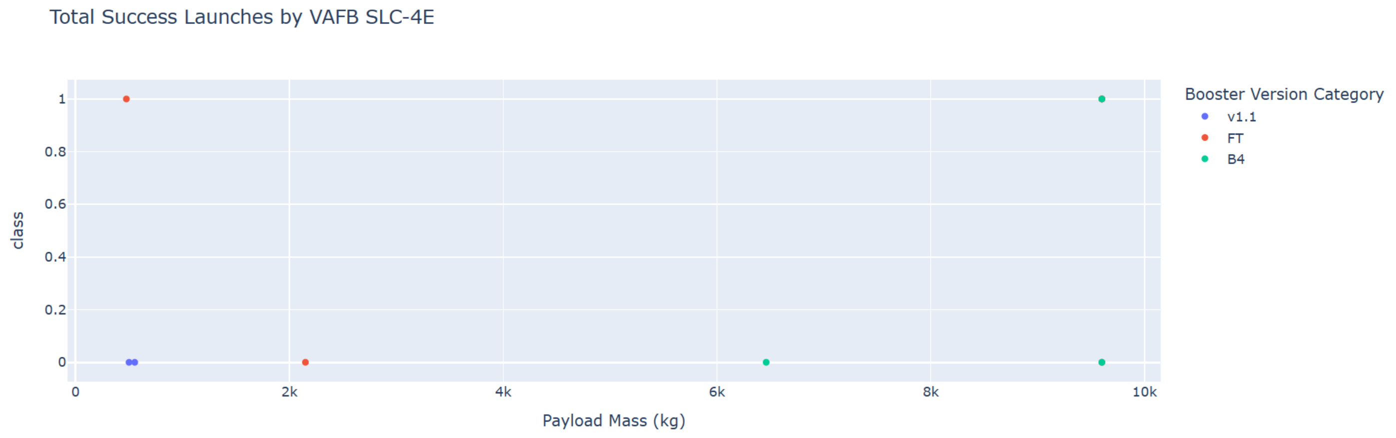
A11: Dashboard for VAFB SLC-4E

VAFB SLC-4E



Payload range (Kg):

0 2,500 5,000 7,500 10,000

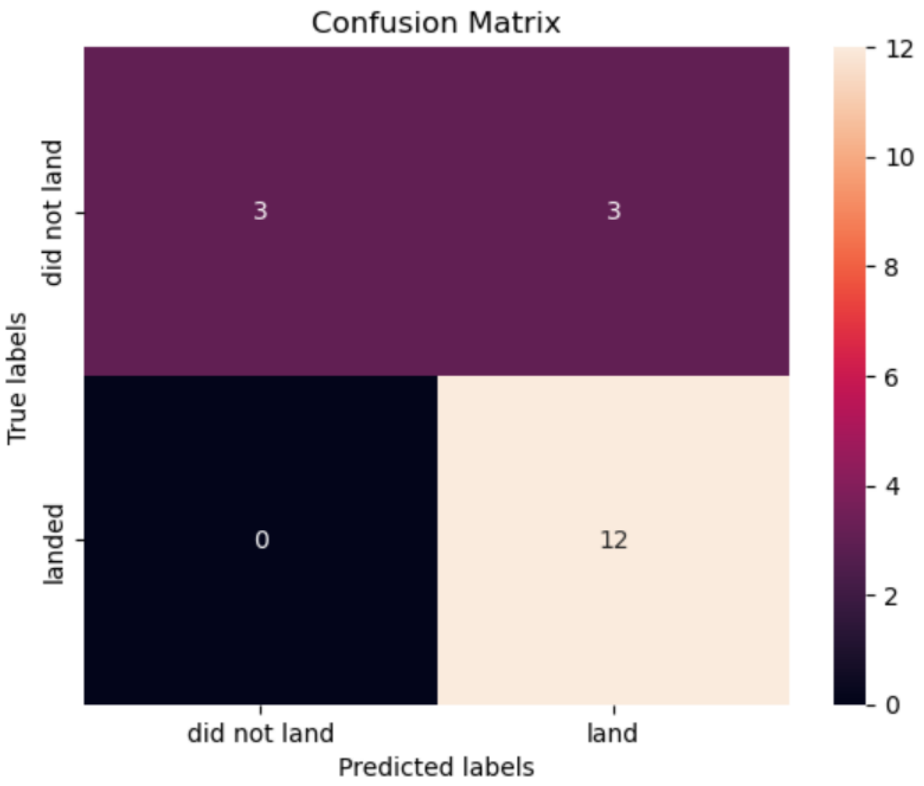


A12: Summary Table of Model Outcomes

	logreg	svm	tree	knn
Accuracy Test (Test Data)	0.833333	0.833333	0.944444	0.833333
Best Parameters	▪ 'C': 0.01, ▪ 'penalty': 'l2’ ▪ 'solver': 'lbfgs'	▪ 'C': 1.0 ▪ 'gamma': 0.03162277660168379 ▪ 'kernel': 'sigmoid'	▪ 'criterion': 'gini’ ▪ max_depth': 16 ▪ 'max_features': 'sqrt’ ▪ min_samples_leaf’: 2 ▪ ‘min_samples_split': 10 ▪ 'splitter': 'random'	▪ 'algorithm': 'auto’ ▪ 'n_neighbors': 10 ▪ 'p': 1
Best Accuracy Score (Training)	0.846429	0.848214	0.873214	0.84821

A13: Confusion Matrix – Logarithmic Regression

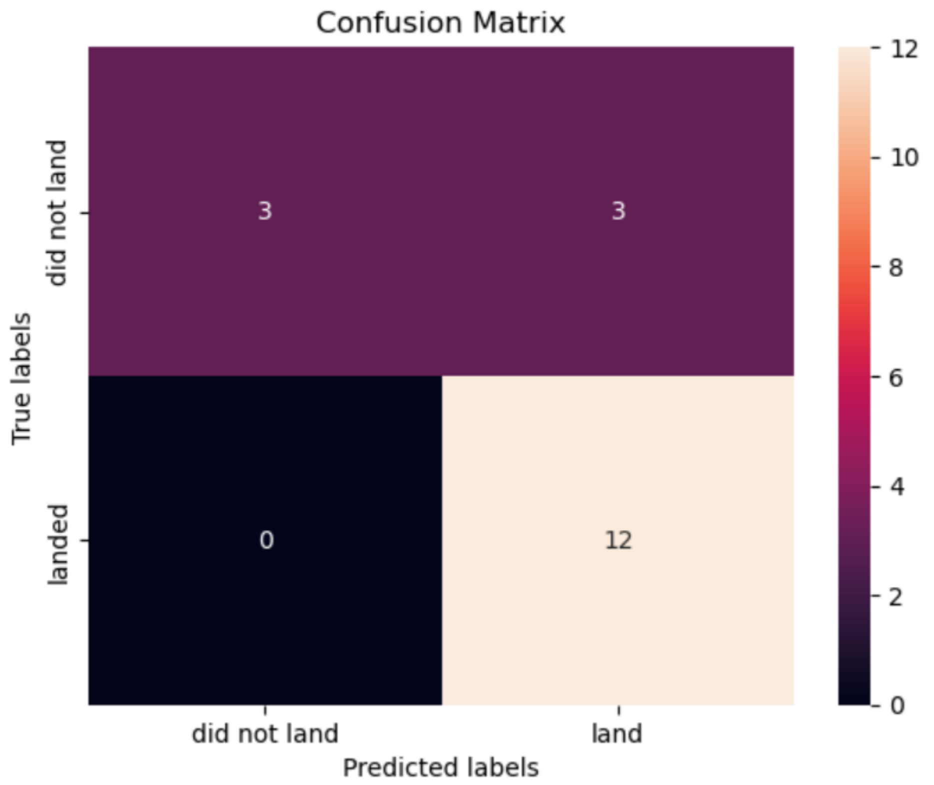
```
yhat=logreg_cv.predict(X_test)
plot_confusion_matrix(Y_test,yhat)
```



	Count
True Positives	12
True Negatives	3
False Positives	3
False Negatives	0
Total:	**18**

A14: Confusion Matrix – SVM

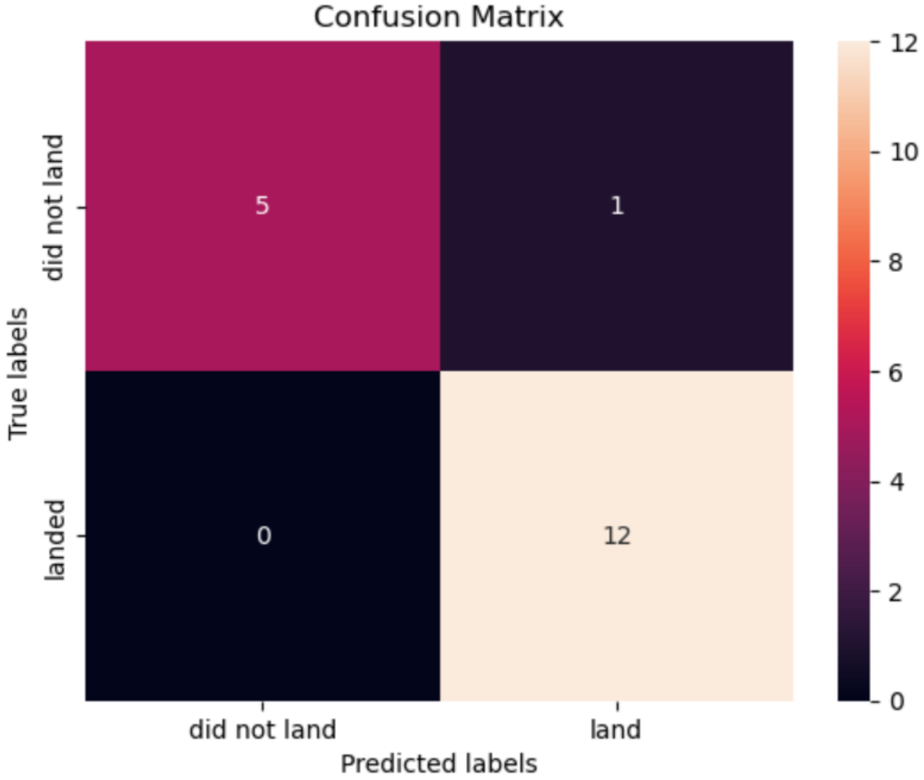
```
yhat=svm_cv.predict(X_test)
plot_confusion_matrix(Y_test,yhat)
```



	Count
True Positives	12
True Negatives	3
False Positives	3
False Negatives	0
Total:	**18**

A15: Confusion Matrix – Decision Tree

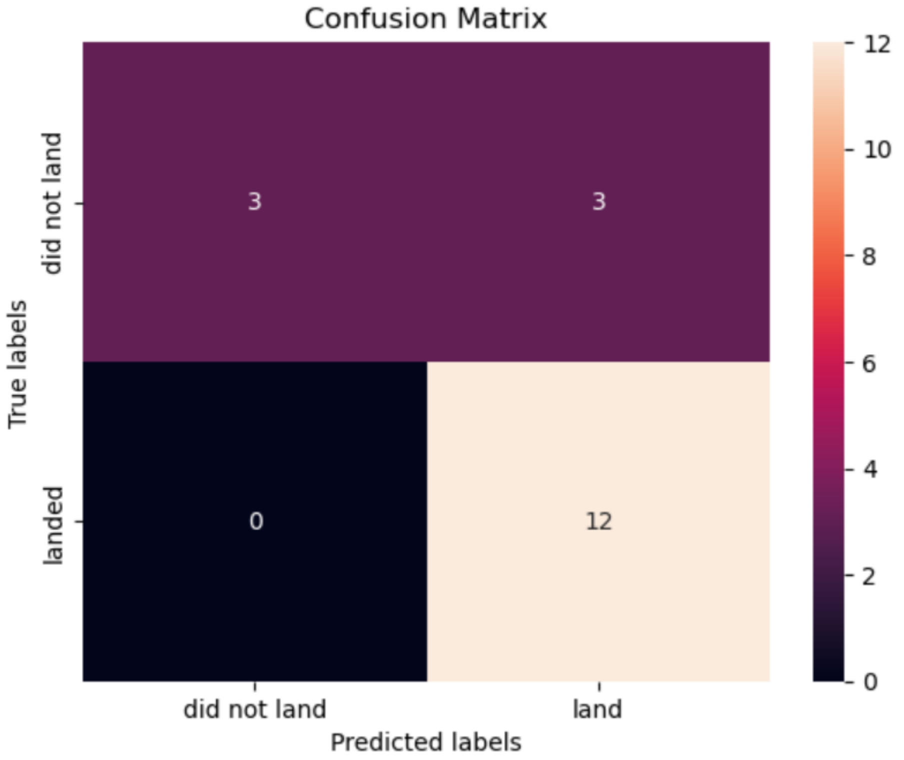
```
yhat_tree = tree_cv.predict(X_test)
plot_confusion_matrix(Y_test,yhat_tree)
```



	Count
True Positives	12
True Negatives	5
False Positives	1
False Negatives	0
Total:	**18**

A16: Confusion Matrix – K-Nearest Neighbors

```
yhat = knn_cv.predict(X_test)
plot_confusion_matrix(Y_test,yhat)
```

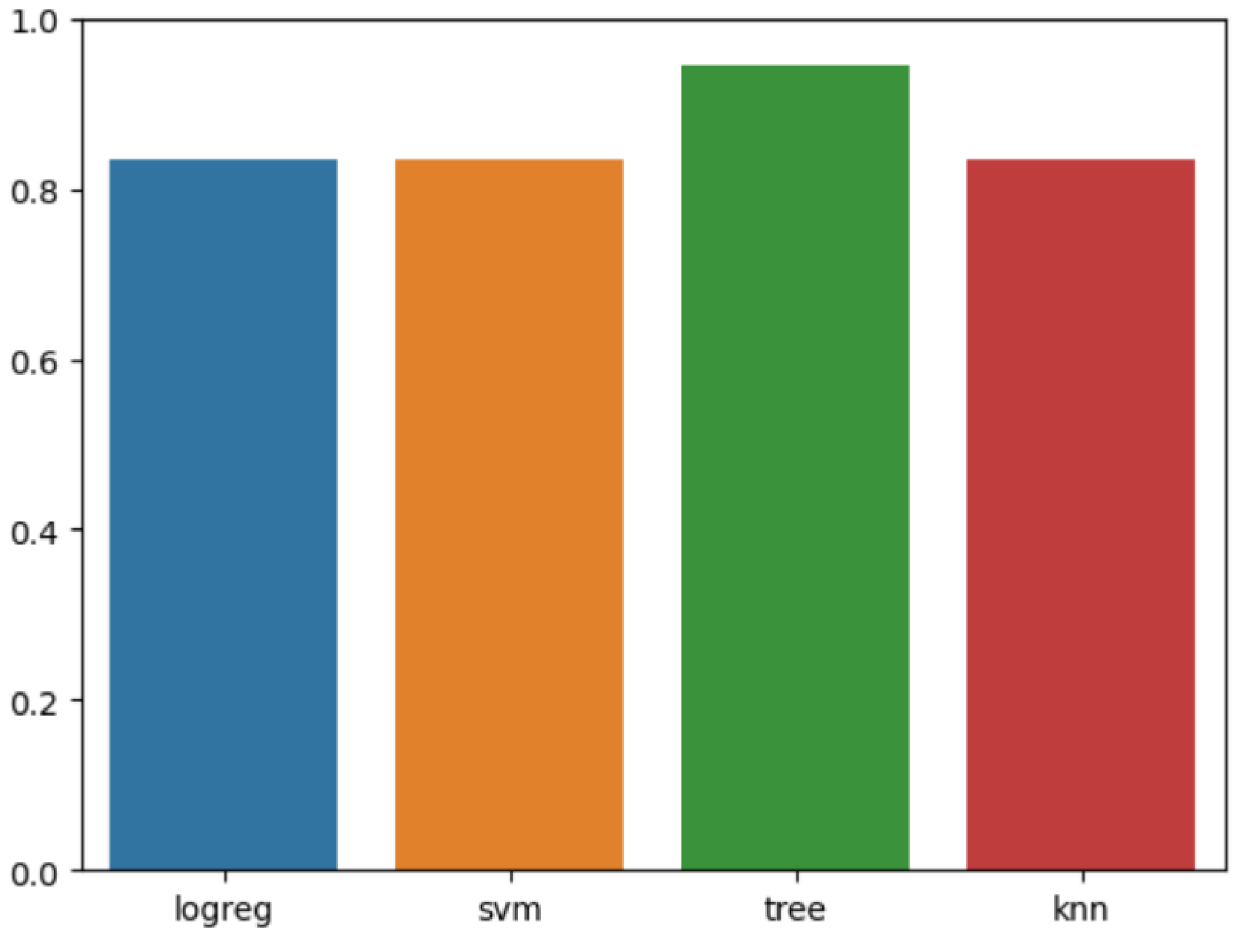


	Count
True Positives	12
True Negatives	3
False Positives	3
False Negatives	0
Total:	**18**

A17: Bar Chart of Accuracy Scores (Test)

```
bar_a=sns.barplot(data=all_scores_pd)
bar_a.set_ylim(0,1)
```

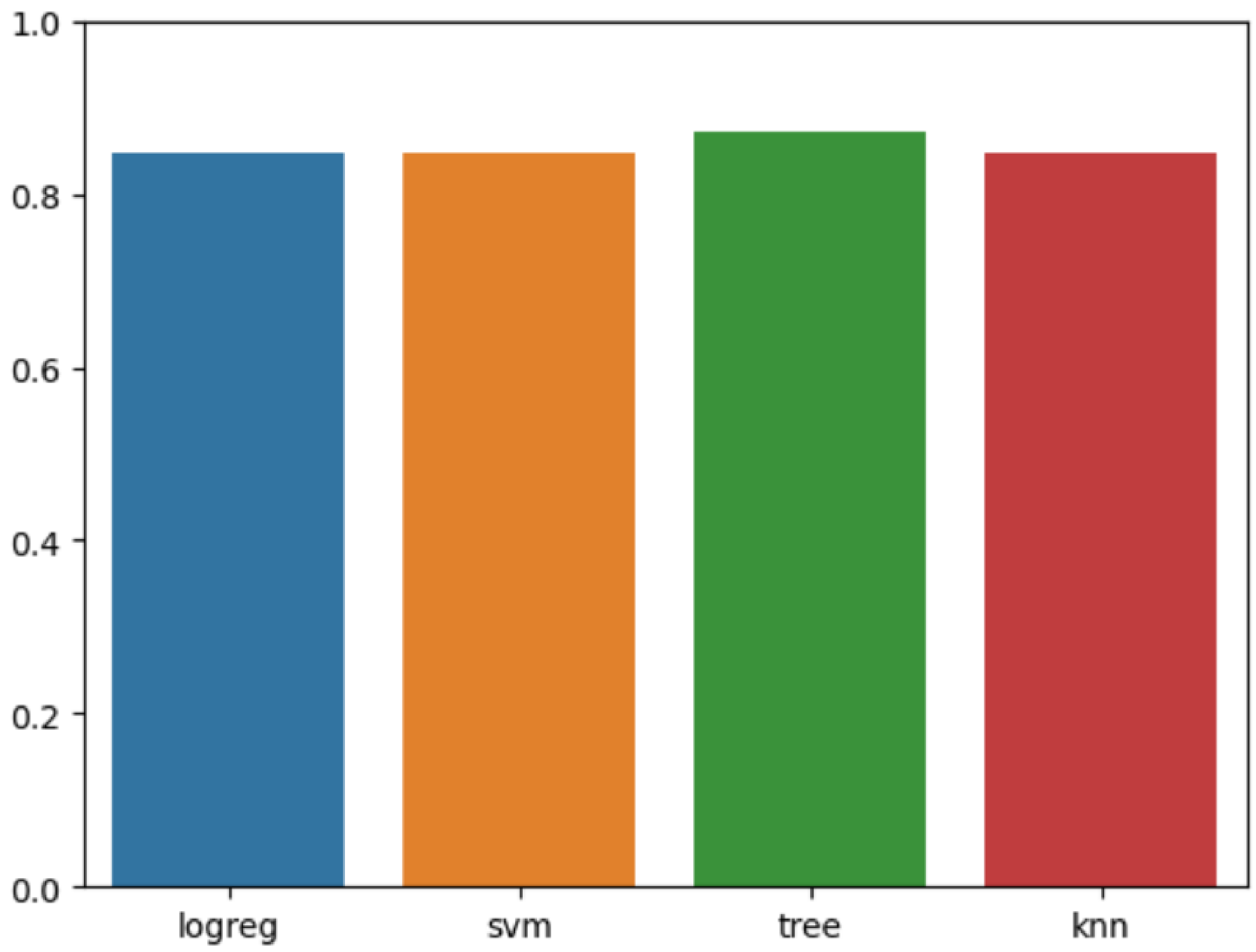
```
(0.0, 1.0)
```



A18: Bar Chart of Accuracy Scores (Best GridSearch)

```
bar_b=sns.barplot(data=all_best_scores_pd)
bar_b.set_ylim(0,1)
```

```
(0.0, 1.0)
```



Thank you!