IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

John Lingo

23 August 2023

Outline

Executive Summary

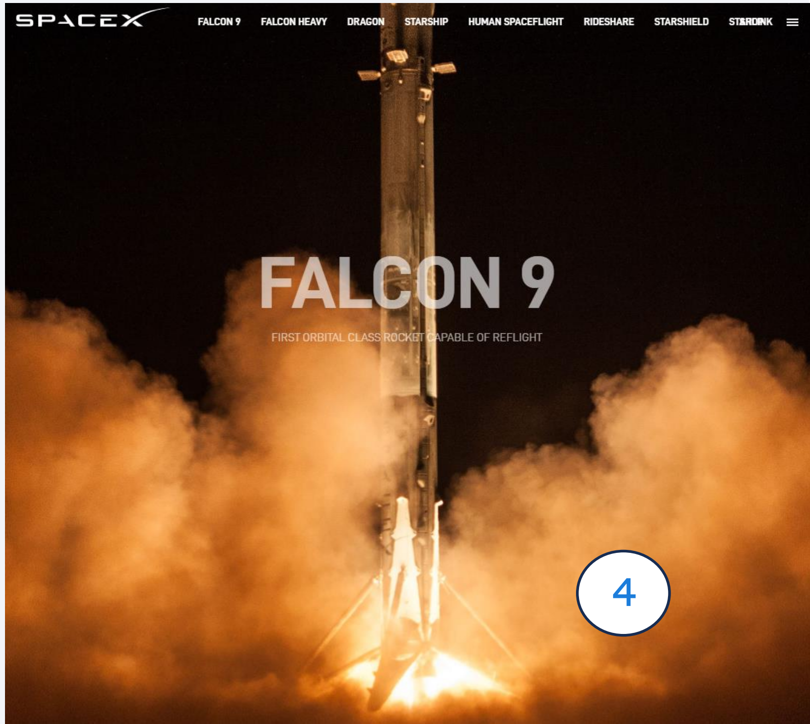
Return to Outline

- Summary of methodologies
  - The Executive Summary of the methodologies for this study is available here
- Summary of results
  - This study identified multiple interesting correlations between launch-related parameters and launch outcomes, including success rate versus target orbit (link) and success ratio versus launch site (link)
  - The study identified that the predictive model that was built using a Decision Tree Classifier was the most accurate of the model types evaluated
  - However, while this model performed very well in terms of avoiding “False Negative” predictions, the model had a high rate (20%) of “False Positive” predictions
  - The next recommended step is to identify whether supplemental parametric data (e.g., weather conditions during landing attempt, indication of “intent” for failed landing) exists that would warrant repeating this analysis in an attempt to develop a more accurate predictive model

Introduction

[Return to Outline](#)

- Project background and context
  - SpaceX can perform Falcon 9 rocket launches at a cost of around 62 million dollars, versus competitors’ costs of at least 165 million dollars for an equivalent launch
  - SpaceX achieves this lower cost based on their ability to re-use the first stage of their Falcon 9 rocket
- Project goal
  - This project will create a machine learning pipeline to predict whether the first stage of the Falcon 9 will successfully land
  - Based on this prediction, a company could more accurately estimate the cost of a launch
    - SpaceX could use this information to tune their pricing process
    - Competitors could use this information to help develop bids against SpaceX



Section 1

Methodology

[Return to Outline]

Methodology

[Return to Executive Summary]

Executive Summary

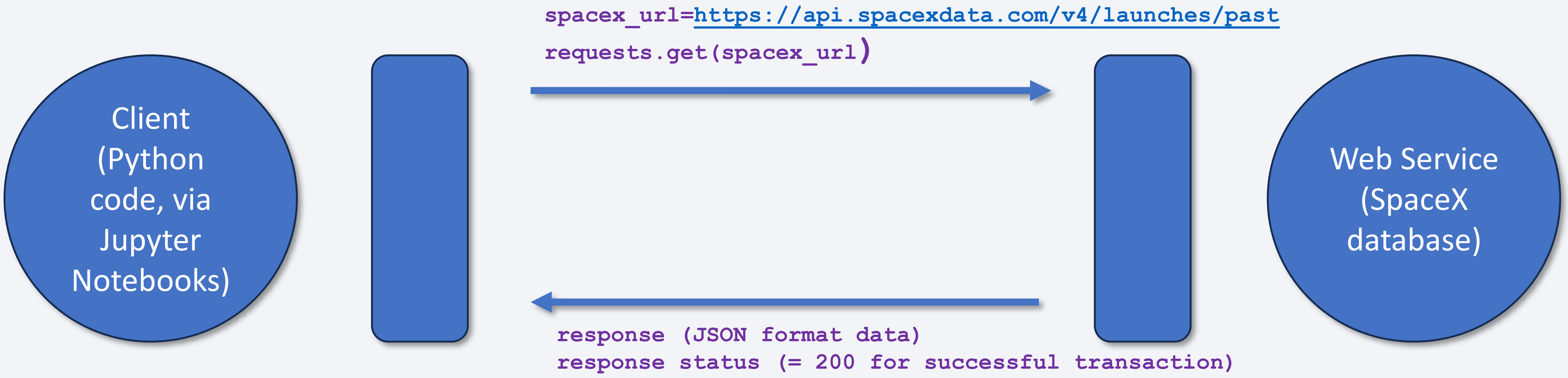
- Data collection methodology
  - Data was collected via querying the SpaceX REST API, and via web scraping of a wiki page on Falcon 9 launches
- Perform data wrangling
  - Data was processed to get it into a Pandas dataframe, filter the desired values, handle missing information, convert categorical information into numerical information, and create a binary field for the target predictor for landing success
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models

Data Collection

- The data sets for this study were accessed from the following sources
  - Download of launch data via the SpaceX REST API
    - https://api.spacexdata.com/v4/launches/past (used to obtain initial set of launch information)
    - https://api.spacexdata.com/v4/rockets (used to obtain booster name)
    - https://api.spacexdata.com/v4/launchpads/ (used to obtain launch pad names and coordinates)
    - https://api.spacexdata.com/v4/payloads/ (used to obtain payload mass and target orbit)
  - Web scraping of data from Wikipedia
    - https://en.wikipedia.org/w/index.php?title=List_of_Falcon_9_and_Falcon_Heavy_launches&oldid=1027686922

Data Collection – SpaceX API

- Overview of data collection via SpaceX REST API calls

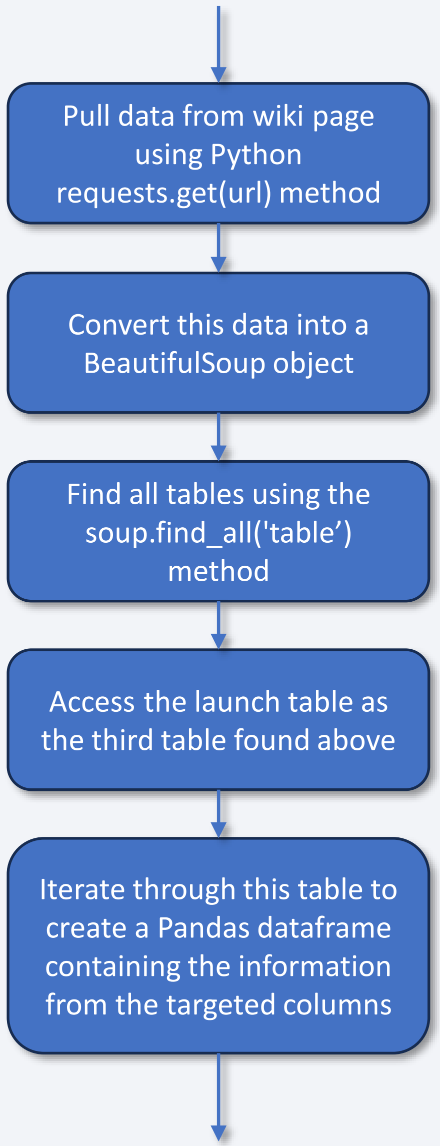


- GitHub URL of completed "SpaceX API calls" notebook: ([link](https://github.com/John-Lingo/Capstone/blob/master/jupyter-labs-spacex-data-collection-api.ipynb))
  (https://github.com/John-Lingo/Capstone/blob/master/jupyter-labs-spacex-data-collection-api.ipynb)

Data Collection - Scraping

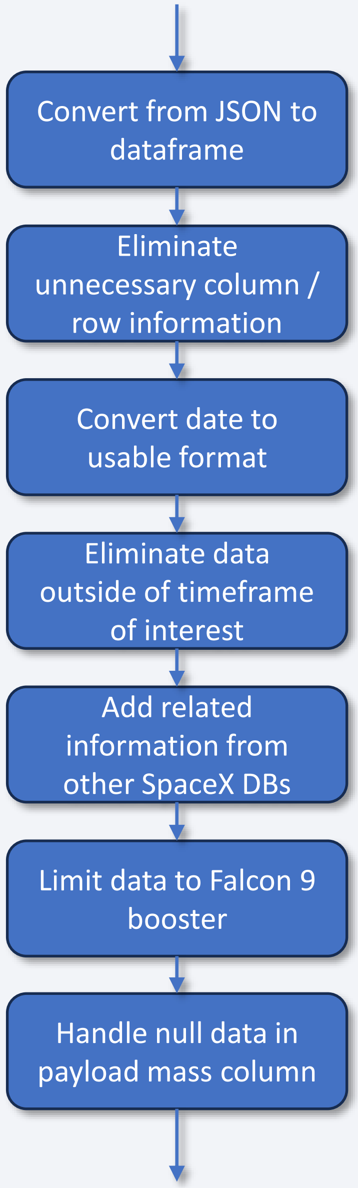
- Overview of data collection via webscraping of wiki page entitled "List of Falcon 9 and Falcon Heavy launches"
  - The data for this portion of the study was extracted from the following wiki page: https://en.wikipedia.org/w/index.php?title=List_of_Falcon_9_and_Falcon_Heavy_launches&oldid=1027686922
  - See flowchart at right for the key high-level steps in the web scraping process

- GitHub URL of completed "web scraping" notebook: (link)
  (https://github.com/John-Lingo/Capstone/blob/master/jupyter-labs-webscraping.ipynb)



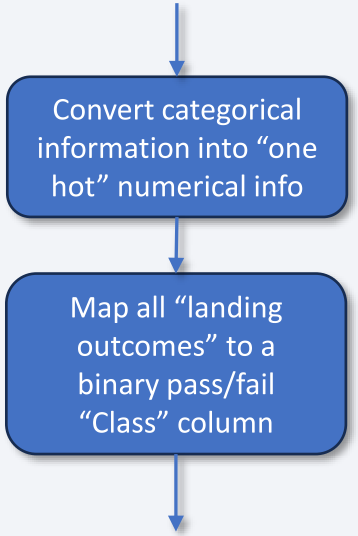
Data Wrangling

- Data Wrangling for the initial data pulled from the SpaceX API included:
  - Converting the data from JSON format to a Pandas dataframe using the Pandas `json_normalize()` method
  - Reducing the columns in this dataframe to include only the columns of interest
  - Removing all rows in the dataframe that listed multiple cores
    - These rows represented Falcon rockets with 2 extra boosters
  - Removing all rows in the dataframe that listed multiple payloads
  - Converting the “data_utc” column to a new “date” column containing only the date (not the time of day) in Python datetime format
  - Removing all rows corresponding to launches that occurred after 11/13/2020
  - Extracting related information (booster name, payload mass, target orbit, launch site name and coordinates, landing details and outcomes, etc.) from additional SpaceX databases
  - Filtering the results to remove all rows for boosters other than the Falcon 9
  - For any rows that specified a payload mass of 0, replaced this mass with the mean value of all non-zero payload masses

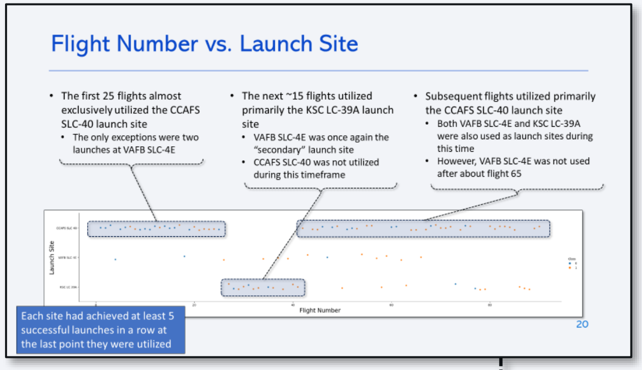


Data Wrangling *(continued)*

- Subsequent Data Wrangling included:
  - Conversion of categorical columns (“Orbit", "LaunchSite", "LandingPad", "Serial") to numeric (a.k.a. “One Hot”) columns using the Pandas get_dummies() method
    - This is a critical step that allows classification models to incorporate this data into their predictions
  - Mapping all landing outcomes into a single “Class” column that constituted the target prediction for this study (representing whether a given launch did/would fail)
    - See appendix for information on mapping of launch outcomes to the “Class” field
- GitHub URL of “Data Wrangling” lab notebook: (link)
  (https://github.com/John-Lingo/Capstone/blob/master/IBM-DS0321EN-SkillsNetwork_labs_module_1_L3_labs-jupyter-spacex-data_wrangling_jupyterlite.jupyterlite.ipynb)



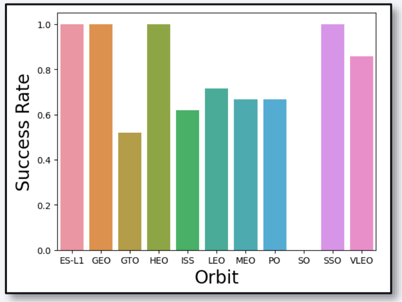
EDA with Data Visualization

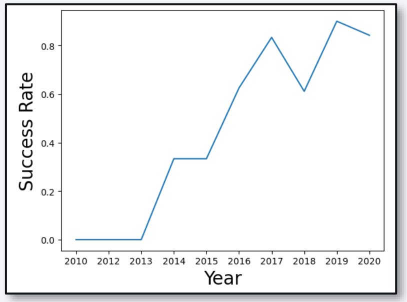


- This portion of the study utilized primarily scatter plots as a method of exploring the impact of multiple variable combinations on the launch outcome, including:
  - Payload Mass versus Flight Number (to see the effect of both the payload and "time and experience" on launch outcomes)
  - Launch Site versus Flight Number (to help visualize how the launch outcomes had varied over time by site)
  - Launch Site versus Payload Mass (to help understand which sites were utilized for which ranges of payload mass values)
  - Orbit versus Flight Number (to help visualize how the target orbits had changed over time)
  - Orbit versus Payload Mass (to help understand which orbits were associated with which ranges of payload mass)

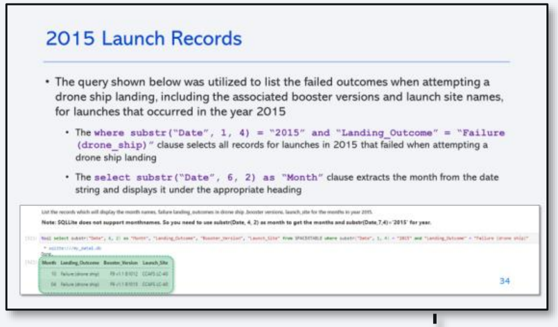
EDA with Data Visualization *(continued)*

- The study also utilized:
  - A bar chart to explore the launch success rate by target orbit type (to help identify the target orbits that had high – or low – success rates)
  - A line chart to view overall launch success rate by year (to visualize how outcomes had generally improved over time)
- GitHub URL of completed “EDA with data visualization” notebook: ([link](https://github.com/John-Lingo/Capstone/blob/master/IBM-DS0321EN-SkillsNetwork_labs_module_2_jupyter-labs-eda-dataviz.ipynb.jupyterlite.ipynb))
  (https://github.com/John-Lingo/Capstone/blob/master/IBM-DS0321EN-SkillsNetwork_labs_module_2_jupyter-labs-eda-dataviz.ipynb.jupyterlite.ipynb)



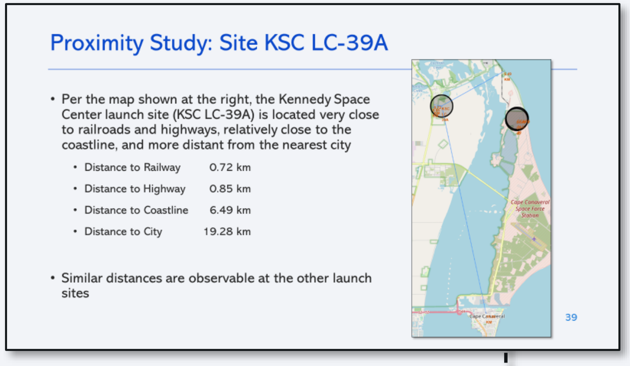


EDA with SQL



- This portion of the data exploration included the following SQL queries (see notebook or presentation pages for details):
  - Names of unique launch sites
  - Total payload mass for a specific customer
  - Average payload mass carried by the F9 v1.1 booster
  - Names of boosters that have successfully landed on a drone ship when carrying payloads in a specific range of mass
  - Date of first successful ground pad landing
  - Total number of successful/failed mission outcomes
  - Names of boosters that have carried the maximum payload value
  - Information on failed landings in 2015
  - Count/ranking of all landing outcomes
- GitHub URL of completed “EDA with SQL” notebook: (link)
  (https://github.com/John-Lingo/Capstone/blob/master/jupyter-labs-eda-sql-coursera_sqllite.ipynb)

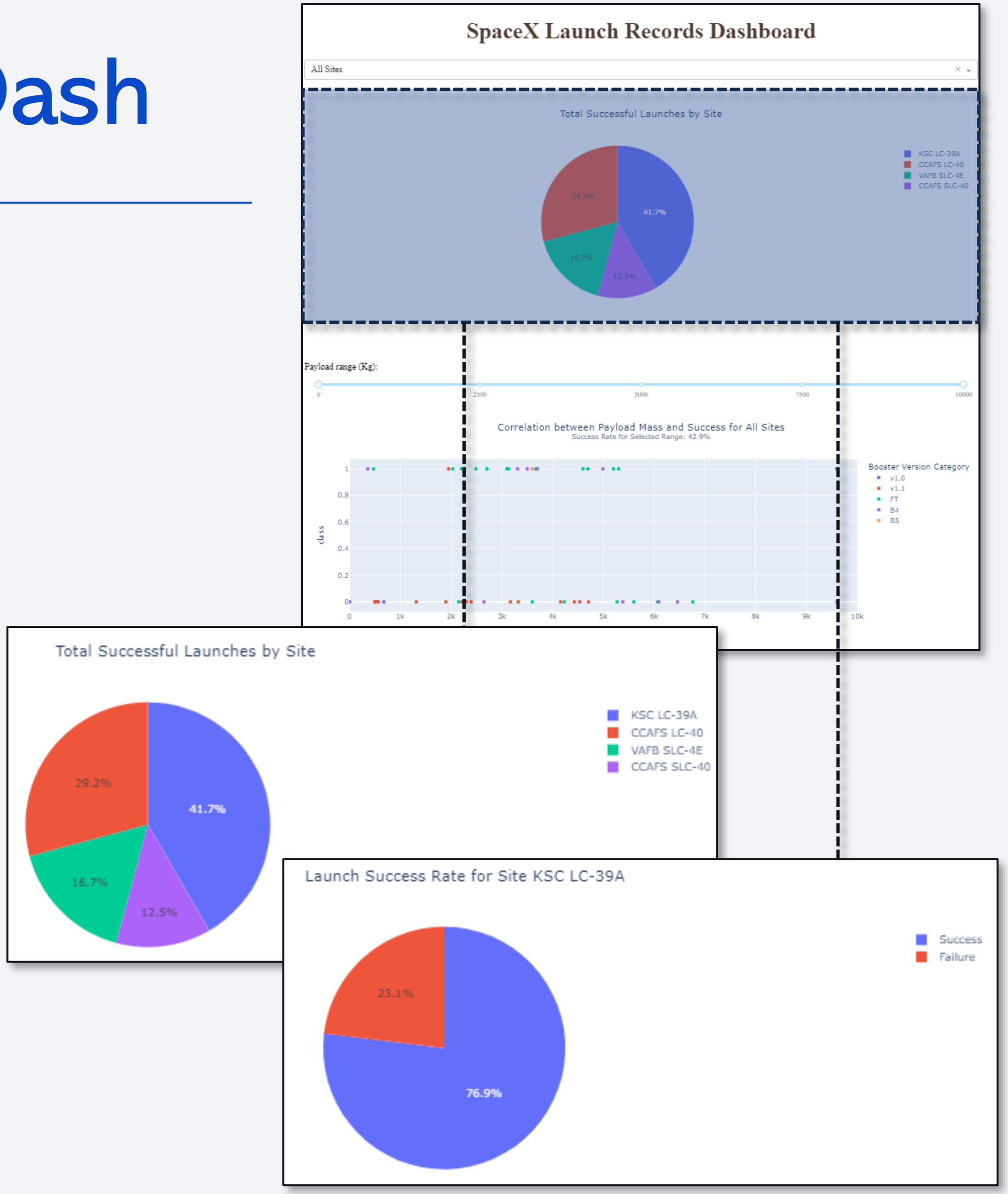
Build an Interactive Map with Folium



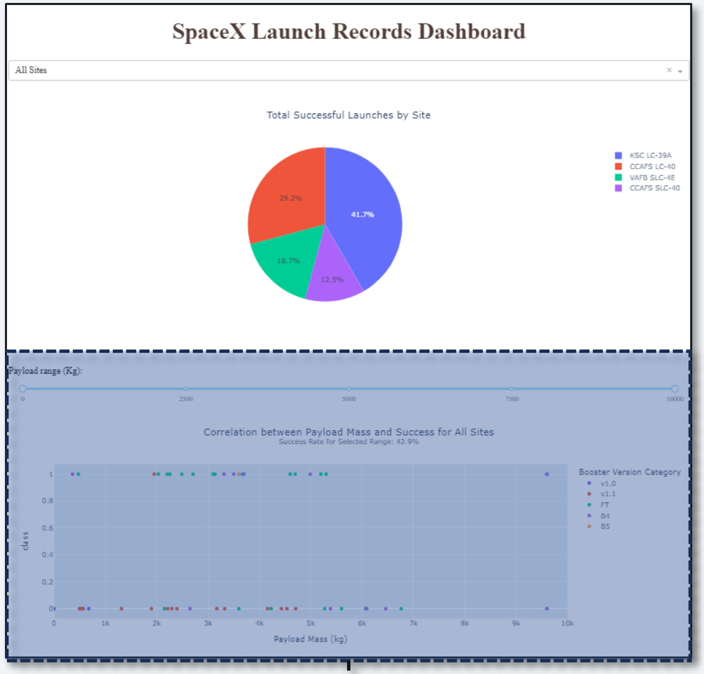
- This portion of the study used many features of Folium maps to explore and visualize landing outcomes, including:
  - Markers: Used to provide the placement and text for labeling launch sites
  - Circles: Used to identify the locations of launch sites
  - PolyLines: Used to measure / portray the distance from a launch site to nearby features of interest (railroads, highways, coastlines, cities)
  - Marker Clusters: Used to provide a grouping of launch outcomes at a specific launch site
- GitHub URL of completed “interactive map with Folium” notebook: (link)
  (https://github.com/John-Lingo/Capstone/blob/master/IBM-DS0321EN-SkillsNetwork_labs_module_3_lab_jupyter_launch_site_location.jupyterlite.ipynb)

Build a Dashboard with Plotly Dash

- The upper section of the dashboard allows users to display either:
  - A pie chart indicating the ratio of the distribution of successful launches by launch site (to help visualize the relationship between launch sites and successful outcomes as aggragated across all sites), or
  - A pie chart indicating the launch success rate for a selected launch site (to help visualize the launch success rate at a specific launch site)



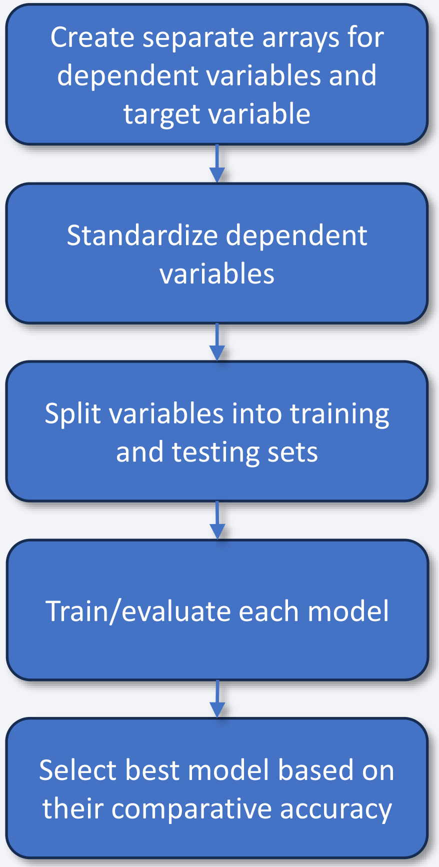
Build a Dashboard with Plotly Dash



- The lower section of the dashboard allows users to display the correlation between the payload mass and the successful launch rate, either for all sites in aggregate or for a specific selected site
  - This data is also filtered (using a range slider) by a selected range of payload masses
  - This portion of the dashboard helps to explore the effect of payload mass on success rate, either in total or in association with a specific launch site
- GitHub URL of Python script from "Plotly Dash" lab: (link)
  (https://github.com/John-Lingo/Capstone/blob/master/spacex_dash_app.py)

Predictive Analysis (Classification)

- Classification models were created for each targeted method (Logistic Regression, Support Vector Machine, Decision Tree Classifier, and K-Nearest Neighbors)
  - These models were tuned using the `GridSearchCV()` facility from Scikit-Learn
  - The models were then trained using the `fit()` method
- Each model was evaluated based on:
  - The accuracy score (`best_score_` parameter) associated with the results of the `fit()` method
  - The accuracy score measured by the `score()` method on the test data sets
  - The Confusion Matrix based on the predictions generated by the `predict()` method on test data
- The best model was selected using the accuracy score from the `fit()` method, since it was the only method that provided sufficient differentiation among the results
- GitHub URL of completed "predictive analysis" lab: (link)
  (https://github.com/John-Lingo/Capstone/blob/master/IBM-DS0321EN-SkillsNetwork_labs_module_4_SpaceX_Machine_Learning_Prediction_Part_5.jupyterlite.ipynb)



Results

Return to Outline

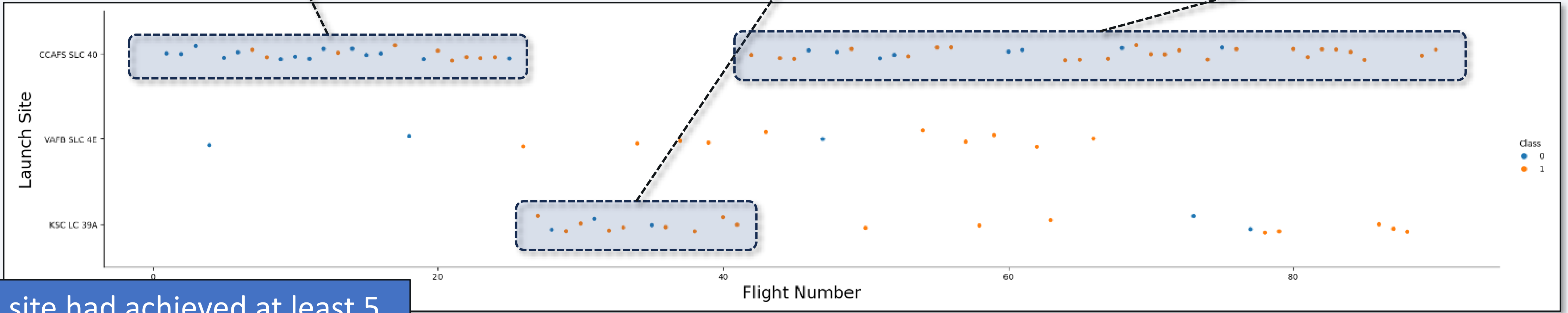
- Exploratory Data Analysis (EDA) results (link)
- Launch sites proximities analysis (link)
- Interactive analytics demo in screenshots (link)
- Predictive analysis results (link)

Section 2

Insights drawn from EDA

Flight Number vs. Launch Site

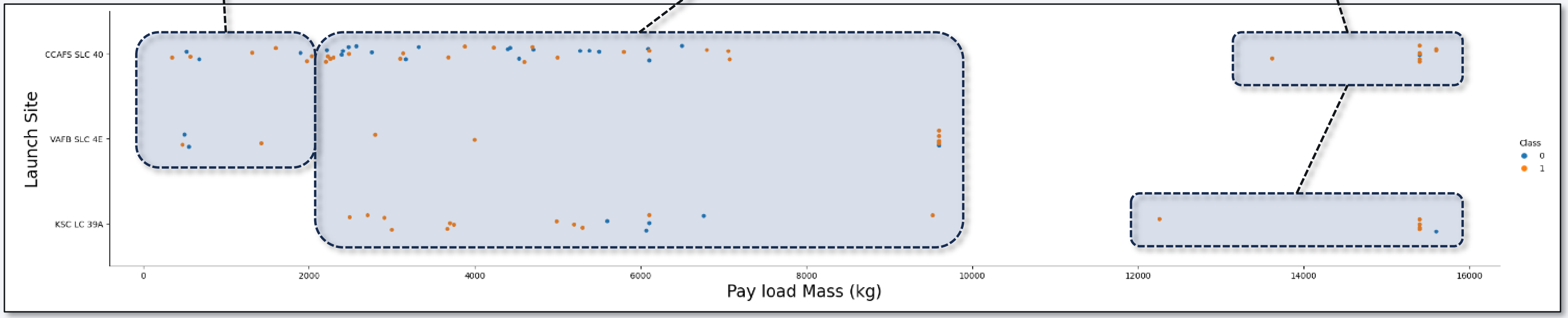
- The first 25 flights almost exclusively utilized the CCAFS SLC-40 launch site
  - The only exceptions were two launches at VAFB SLC-4E
- The next ~15 flights utilized primarily the KSC LC-39A launch site
  - VAFB SLC-4E was once again the "secondary" launch site
  - CCAFS SLC-40 was not utilized during this timeframe
- Subsequent flights utilized primarily the CCAFS SLC-40 launch site
  - Both VAFB SLC-4E and KSC LC-39A were also used as launch sites during this time
  - However, VAFB SLC-4E was not used after about flight 65



Each site had achieved at least 5 successful launches in a row at the last point they were utilized

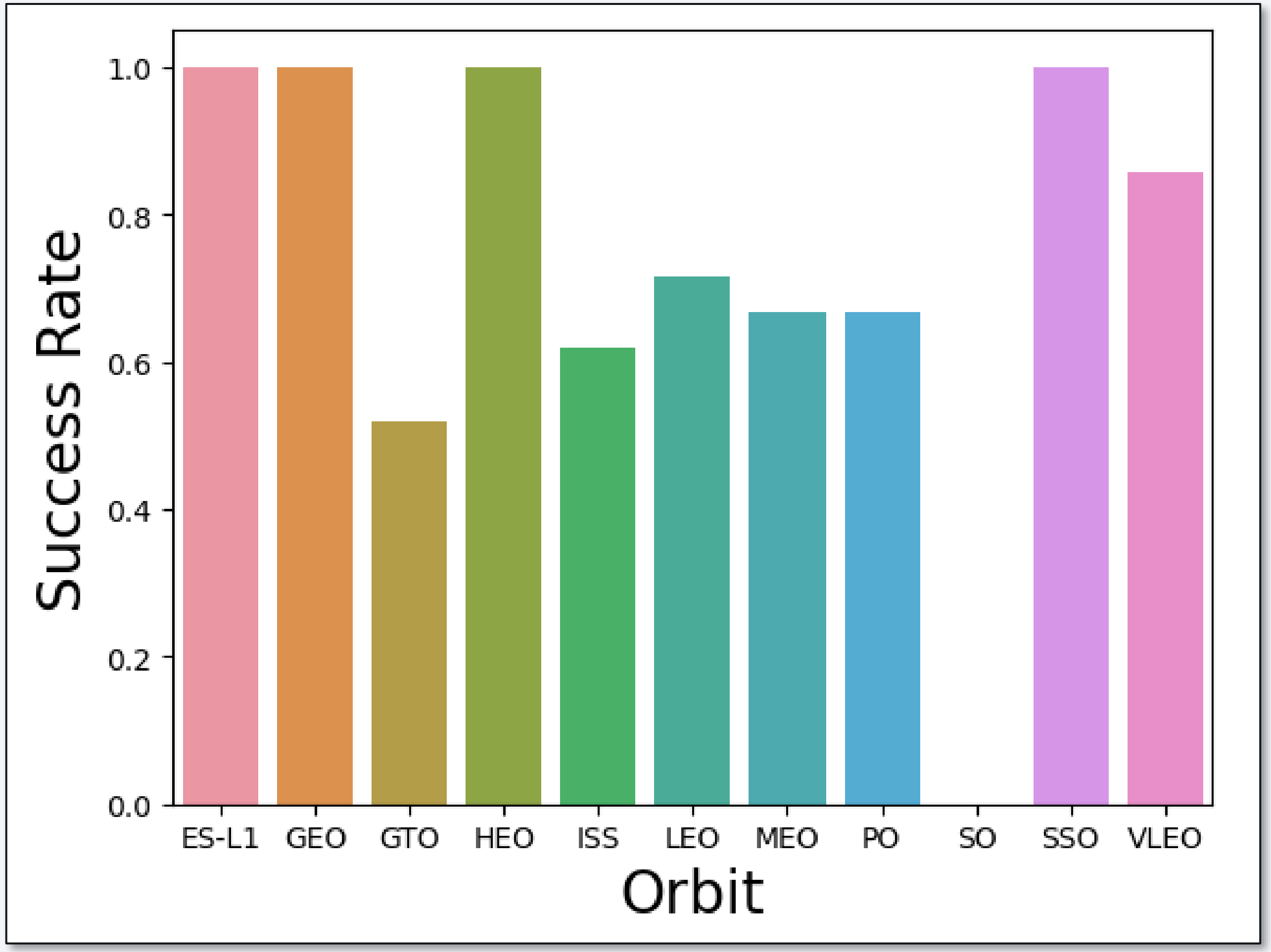
Payload vs. Launch Site

- Launches containing payloads of 2000 kg or less only occurred from the CCAFS SLC-40 and VAFB SLC-4E launch sitesThe only exceptions were two launches at VAFB SLC-4E
- All three launch sites were utilized for payloads in the range of 2000 kg to 10000 kg
  - VAFB SLC-4E did not have any launches with payloads exceeding 9600 kg
- Launches of payloads above 10000 kg were attempted from the CCAFS SLC-40 and KSC LC-39A launch sites
  - The success rate was high for launches in this payload range



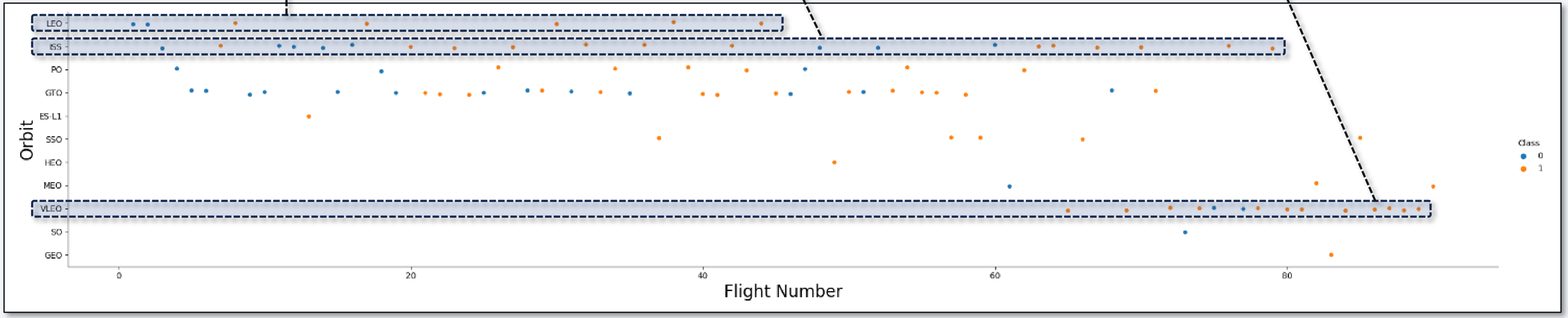
Success Rate vs. Orbit Type

- Launches that targeted the ES-L1, GEO, HEO, and SSO orbit types experienced a 100% success rate
  - The lowest success rate was for the SO orbit (0%, although there is very little data for this orbit type), followed by the GTO orbit (~50%)
  - Note: See Appendix for further information on Orbit types



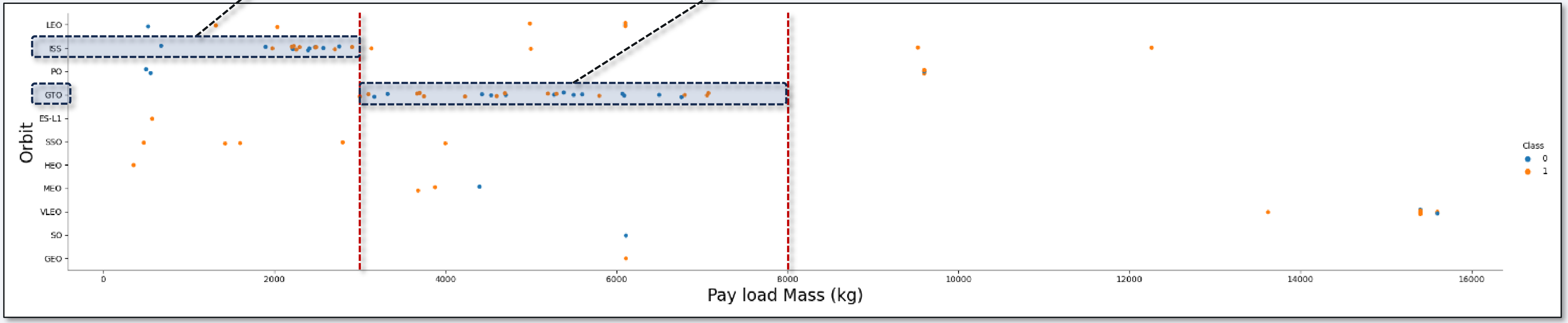
Flight Number vs. Orbit Type

- The success rate of launches targeting the LEO orbit was directly tied to the flight number (i.e., later flights were all successful)
- However, similar success rate relationships do not exist for most other orbit types (see for example the ISS orbit type)
- Later launches predominantly favored the VLEO orbit type, with >80% success rate



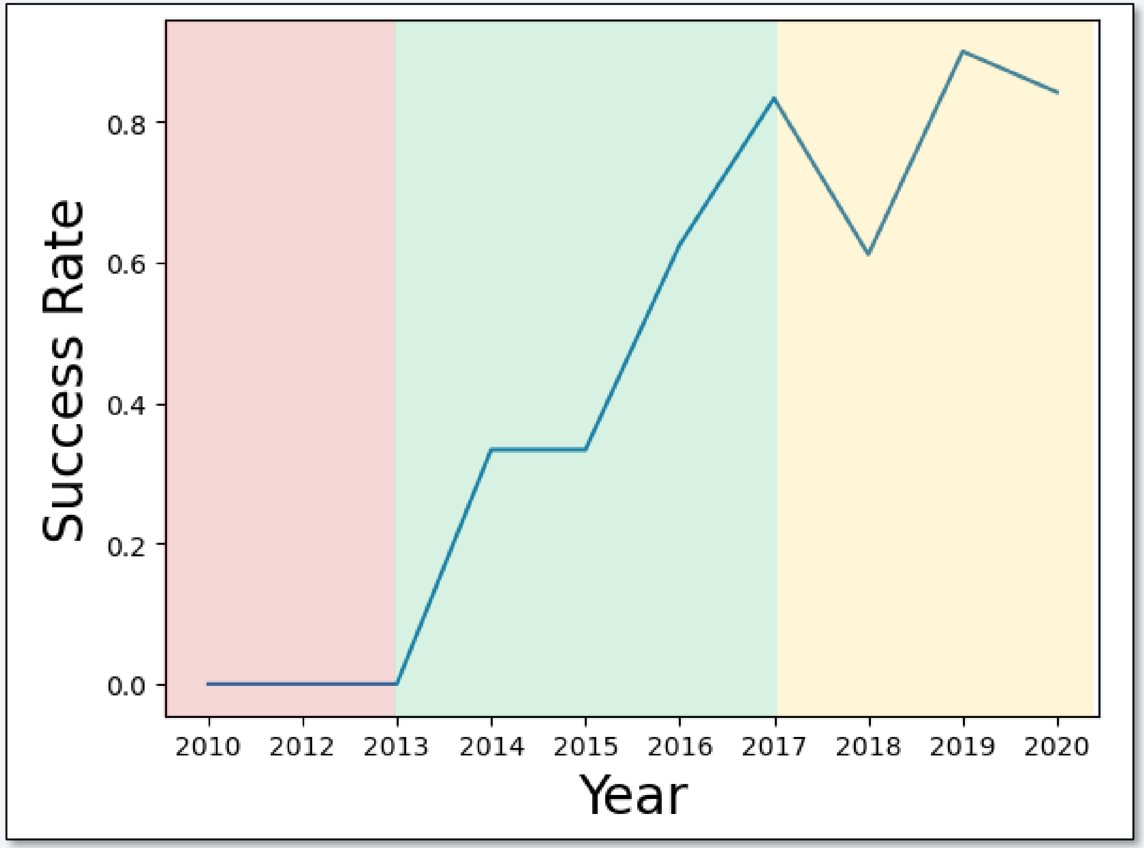
Payload vs. Orbit Type

- Payloads of 3000 kg or less targeted multiple orbit types
  - The most prominent target for these payloads was the ISS orbit type, particularly in the range of 2000 to 3000 kg
- Payloads of 3000 to 8000 kg also targeted multiple orbit types
  - The most prominent target for these payloads was the GTO orbit type
- Launches with payload mass above 8000 kg targeted the ISS, PO, and VLEO orbit types, with a high success rate



Launch Success Yearly Trend

- The launch success rate was 0% from 2010 through 2013
- The average success rate then climbed rapidly from 2013 through 2017
- From 2017 through 2020, the success rate varied non-monotonically, but remained above at least 60%

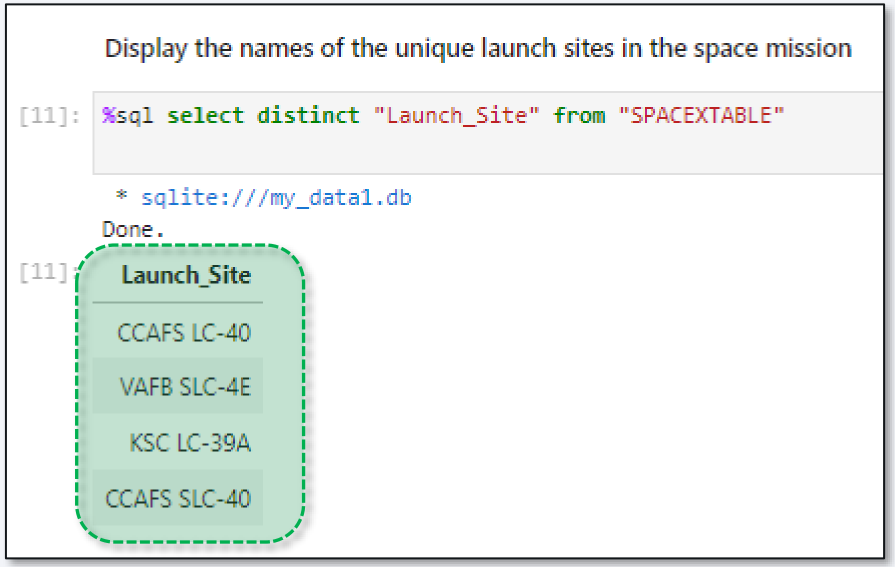


All Launch Site Names

- The query shown at the right was utilized to find the names of the unique launch sites
  - The key element to this query was the use of

    `distinct "Launch_Site"`

    as the selection criterion, to ensure that each launch site was only displayed once



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

```
[11]: %sql select distinct "Launch_Site" from "SPACEXTABLE"

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

[11]:

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

Launch Site Names Begin with 'CCA'

- The query shown below was utilized to find 5 records where launch sites begin with ‘CCA’
  - The **where “Launch_Site” like “CCA%”** clause was utilized to match sites that begin with ‘CCA’
  - The **limit 5** clause was utilized to ensure that only 5 records were displayed

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

```
[14]: %sql select * from SPACEXTABLE where "Launch_Site" like "CCA%" limit 5
```

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

[14]:

Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Orbit	Customer	Mission_Outcome	Landing_Outcome
2010-04-06	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success	Failure (parachute)
2010-08-12	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	07:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success	No attempt
2012-08-10	00:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500	LEO (ISS)	NASA (CRS)	Success	[illegible]empt
2013-01-03	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677	LEO (ISS)	NASA (CRS)	Success	No attempt

Total Payload Mass

- The query shown below was utilized to calculate the total payload carried by boosters launched on behalf of customer NASA (CRS)
  - The `where “Customer” = “NASA (CRS)”` clause selects only records for customer NASA (CRS)
  - The `sum(PAYLOAD_MASS__KG_)` clause sums the payload mass for the selected records

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

```
[16]: %sql select sum(PAYLOAD_MASS__KG_) from SPACEXTABLE where "Customer" = "NASA (CRS)"

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

[16]:

sum(PAYLOAD_MASS__KG_)
45596

Average Payload Mass by F9 v1.1

- The query shown below was utilized to calculate the average payload mass carried by booster version F9 v1.1
  - The **where "Booster_Version" like "F9 v1.1%"** clause ensures that only records for the F9 v1.1 booster version are selected
  - The **avg(PAYLOAD_MASS__KG_)** clause calculates the average payload mass for the selected records

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

```
[19]: %sql select avg(PAYLOAD_MASS__KG_) from SPACEXTABLE where "Booster_Version" like "F9 v1.1%"

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

[19]:

avg(PAYLOAD_MASS__KG_)
2534.6666666666665

First Successful Ground Landing Date

- The query shown below was utilized to find the date of the first successful landing outcome on a ground pad
  - The `where "Landing_Outcome" = "Success (ground pad)"` clause limits the selected records to those involving a successful ground pad landing
  - The `order by "Date" limit 1` clause ensures that only the earliest date is selected/displayed

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

Hint:Use min function

```
[24]: %sql select "Date" from SPACEXTABLE where "Landing_Outcome" = "Success (ground pad)" order by "Date" limit 1
```

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

[24]:

Date
2015-12-22

Successful Drone Ship Landing with Payload between 4000 and 6000 kg

- The query shown below was utilized to list the names of boosters that have successfully landed on a drone ship and that had a payload mass greater than 4000 but less than 6000 kg
  - The **where “Landing_Outcome” = “Success (drone ship)” and PAYLOAD_MASS__KG_ between 4000 and 6000** clause specifies the records of interest
  - The **distinct “Booster_Version”** clause ensures that each version is displayed only once

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

```
[29]: %sql select distinct "Booster_Version" from SPACEXTABLE where "Landing_Outcome" = "Success (drone ship)" and PAYLOAD_MASS__KG_ between 4000 and 6000
 * sqlite:///my_data1.db
Done.
```

[29]:

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

Total Number of Successful and Failure Mission Outcomes

- The query shown below was utilized to calculate the total number of successful and failure mission outcomes
  - The `group by “Mission_Outcome” like “Success%”` clause groups all outcomes beginning with “Success” together, and groups the failure types (of which there is only 1) uniquely
  - The `count(“Mission_Outcome”)` clause is specified as a selected column in order to count and display the instances of outcomes in each unique category

List the total number of successful and failure mission outcomes

```
[43]: %sql select "Mission_Outcome", count("Mission_Outcome") from SPACEXTABLE group by "Mission_Outcome" like "Success%"
```

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

[43]:

Mission_Outcome	count("Mission_Outcome")
Failure (in flight)	1
Success	100

Boosters Carried Maximum Payload

List the names of the booster_versions which have carried the maximum payload mass. Use a subquery

```
[47]: %sql select distinct "Booster_Version" from SPACEXTABLE where PAYLOAD_MASS__KG_ = (select max(PAYLOAD_MASS__KG_) from SPACEXTABLE)
```

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

[47]:

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

- The above query was utilized to list the names of the booster versions that have carried the maximum payload mass
  - The `where PAYLOAD_MASS__KG_ = (select max(PAYLOAD_MASS__KG_) from SPACEXTABLE)` sub-query clause was utilized to select only the records for launches that were carrying a payload equal to the maximum payload value
  - The `distinct “Booster_Version”` clause ensures that each booster version is only displayed once, regardless of how many times it carried the maximum payload

2015 Launch Records

- The query shown below was utilized to list the failed outcomes when attempting a drone ship landing, including the associated booster versions and launch site names, for launches that occurred in the year 2015
  - The `where substr("Date", 1, 4) = "2015" and "Landing_Outcome" = "Failure (drone_ship)"` clause selects all records for launches in 2015 that failed when attempting a drone ship landing
  - The `select substr("Date", 6, 2) as "Month"` clause extracts the month from the date string and displays it under the appropriate heading

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: SQLLite does not support monthnames. So you need to use substr(Date, 4, 2) as month to get the months and substr(Date,7,4)='2015' for year.

```
[52]: %sql select substr("Date", 6, 2) as "Month", "Landing_Outcome", "Booster_Version", "Launch_Site" from SPACEXTABLE where substr("Date", 1, 4) = "2015" and "Landing_Outcome" = "Failure (drone ship)"
```

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

[52]:

Month	Landing_Outcome	Booster_Version	Launch_Site
10	Failure (drone ship)	F9 v1.1 B1012	CCAFS LC-40
04	Failure (drone ship)	F9 v1.1 B1015	CCAFS LC-40

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.

```
[61]: %sql select "Landing_Outcome", count("Landing_Outcome") as "Count" from SPACEXTABLE where "Date" between "2010-06-04" and "2017-03-20" group by "Landing_Outcome" order by "Count" desc

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

[61]:

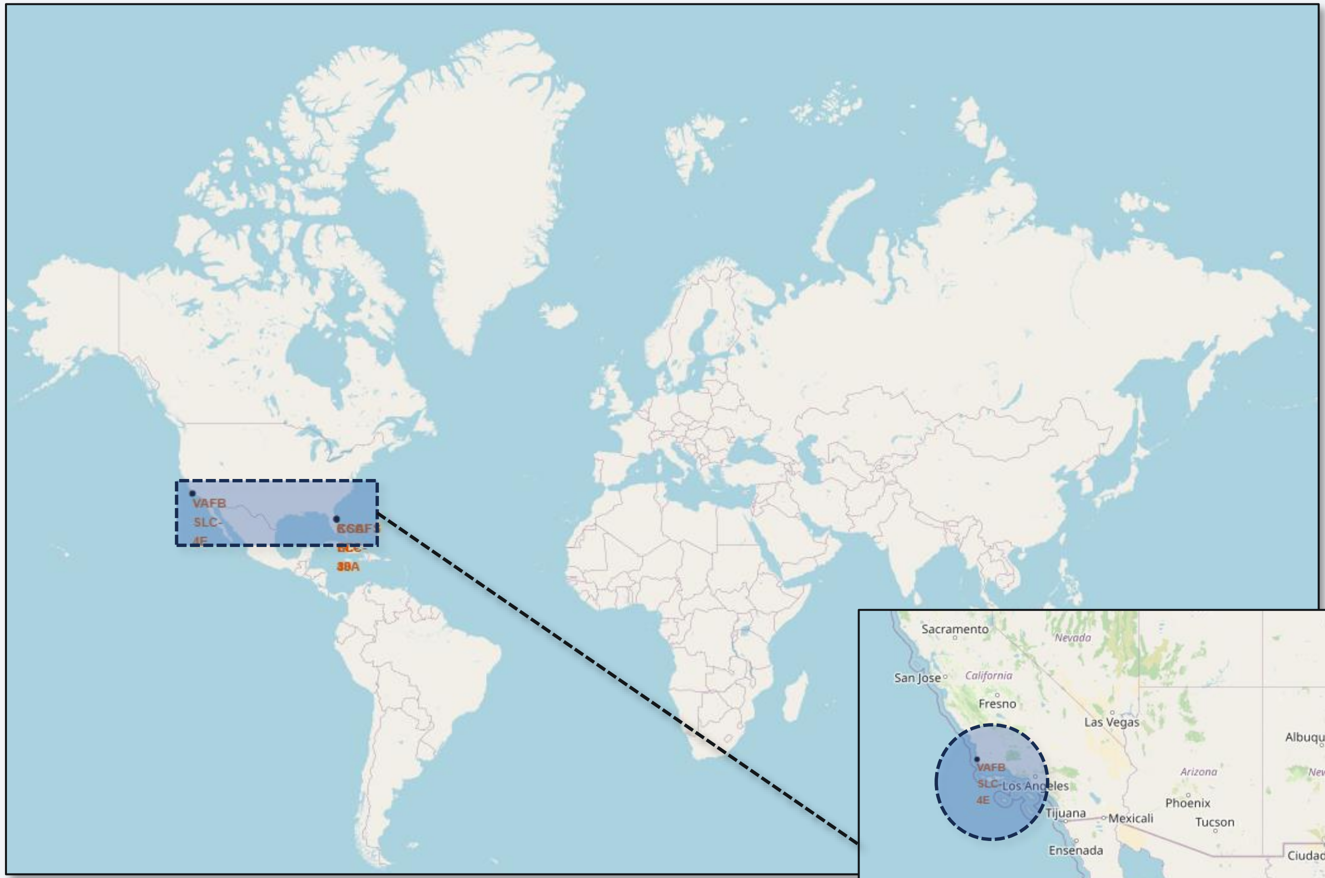
Landing_Outcome	Count
No attempt	10
Success (ground pad)	5
Success (drone ship)	5
Failure (drone ship)	5
Controlled (ocean)	3
Uncontrolled (ocean)	2
Precluded (drone ship)	1
Failure (parachute)	1

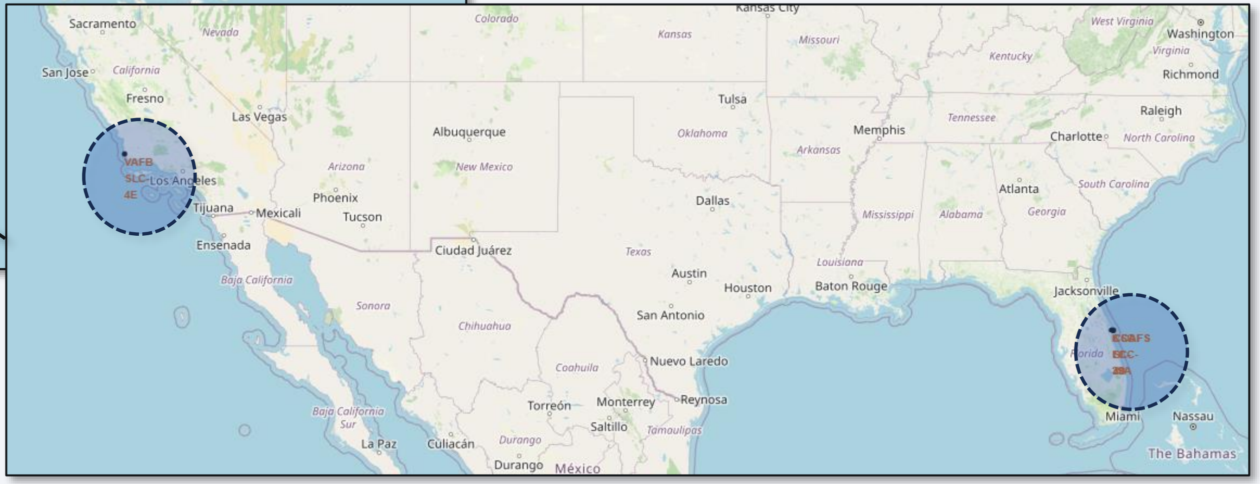
- The above query was utilized to rank the count of landing outcomes (such as “Failure (drone ship)” or “Success (ground pad)”) between the dates 2010-06-04 and 2017-03-20, in descending order
  - The `where “Date” between “2010-06-04” and “2017-03-20”` clause limits the selected records to the specified range
  - The `group by “Landing_Outcome”` clause, coupled with the `count(“Landing_Outcome”) as “Count”` clause, allow the landing outcomes to be grouped, counted, and displayed under an appropriate heading
  - The `order by “Count” desc` clause displays the results in descending order of landing outcome counts, as requested

Section 3

Launch Sites Proximities Analysis

Map of all Launch Sites for SpaceX

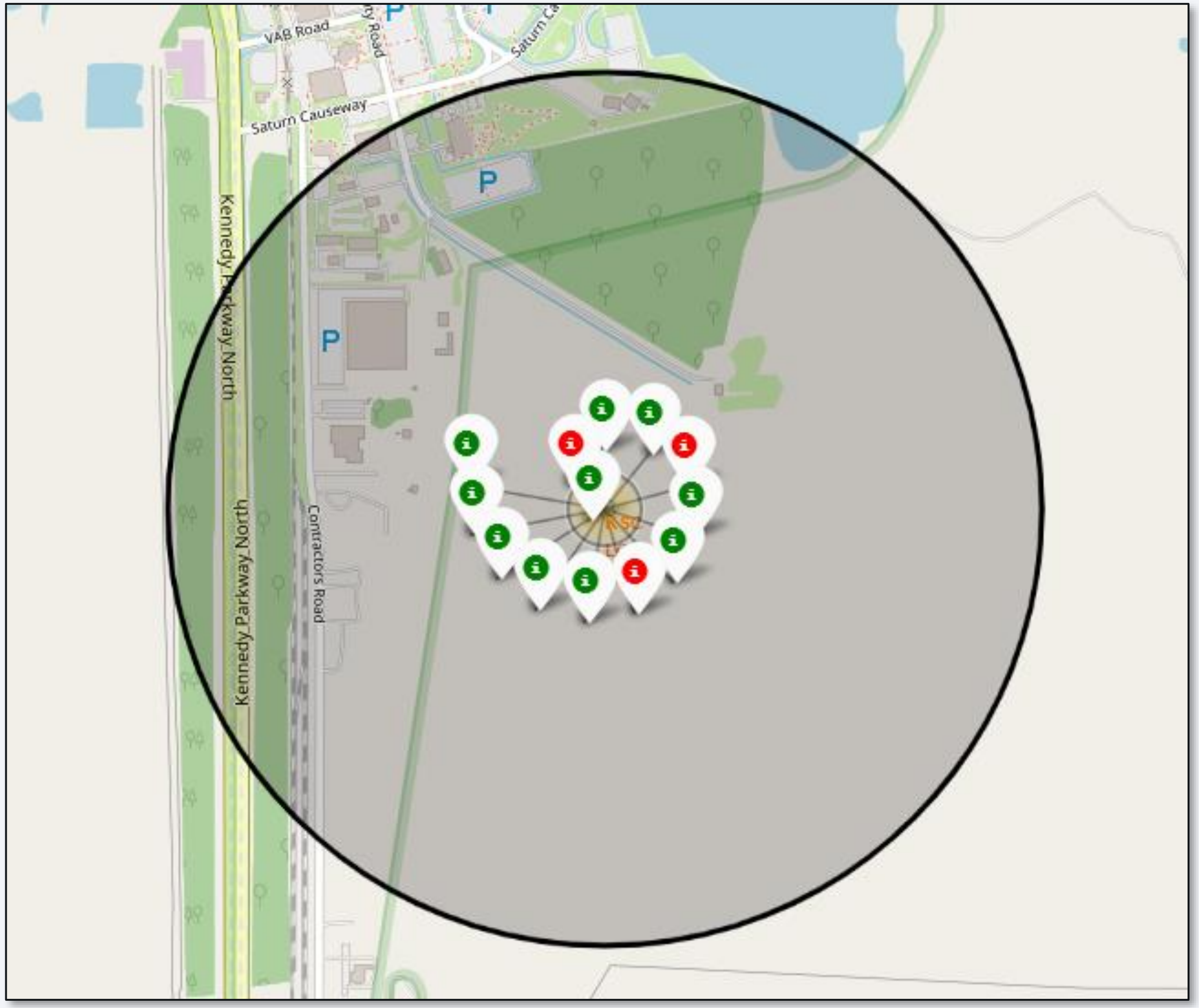




- All SpaceX launch sites are in the continental United States, and are located at facilities that are near either the Pacific or Atlantic Oceans

Launch Outcomes for Site KSC LC-39A

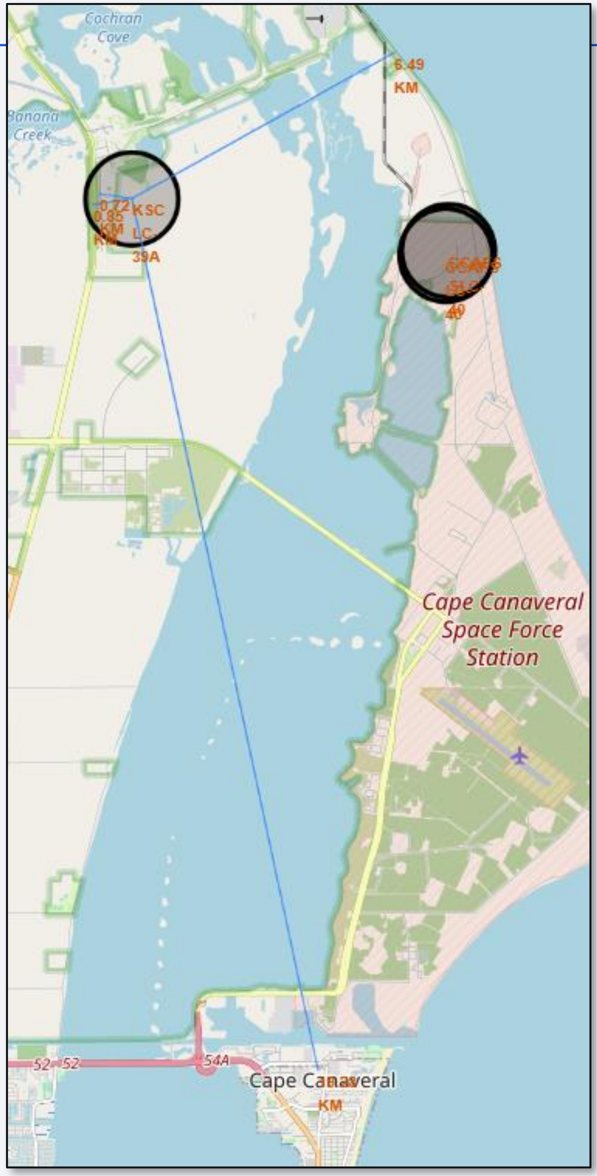
- The map at the right indicates a total of 13 launches at site KSC LC-39A
    - 10 launches were successful (green pins)
    - 3 launches were unsuccessful (red pins)



Proximity Study: Site KSC LC-39A

- Per the map shown at the right, the Kennedy Space Center launch site (KSC LC-39A) is located very close to railroads and highways, relatively close to the coastline, and more distant from the nearest city
  - Distance to Railway 0.72 km
  - Distance to Highway 0.85 km
  - Distance to Coastline 6.49 km
  - Distance to City 19.28 km

- Similar distances are observable at the other launch sites

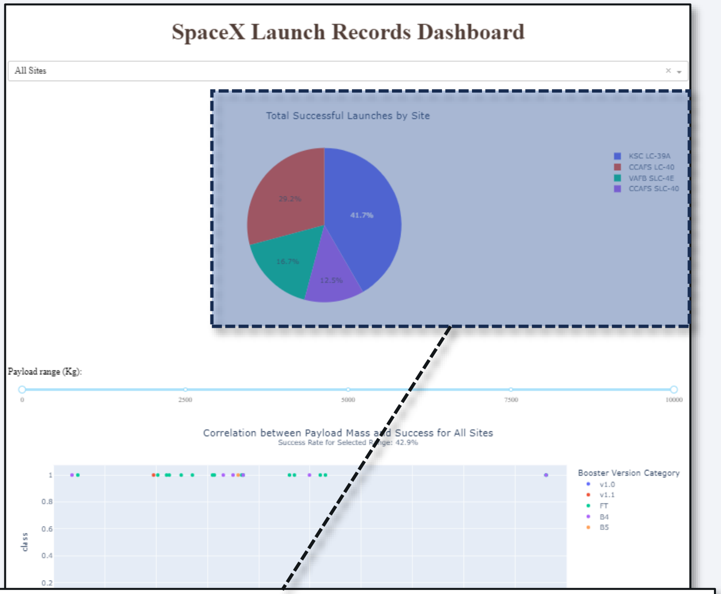


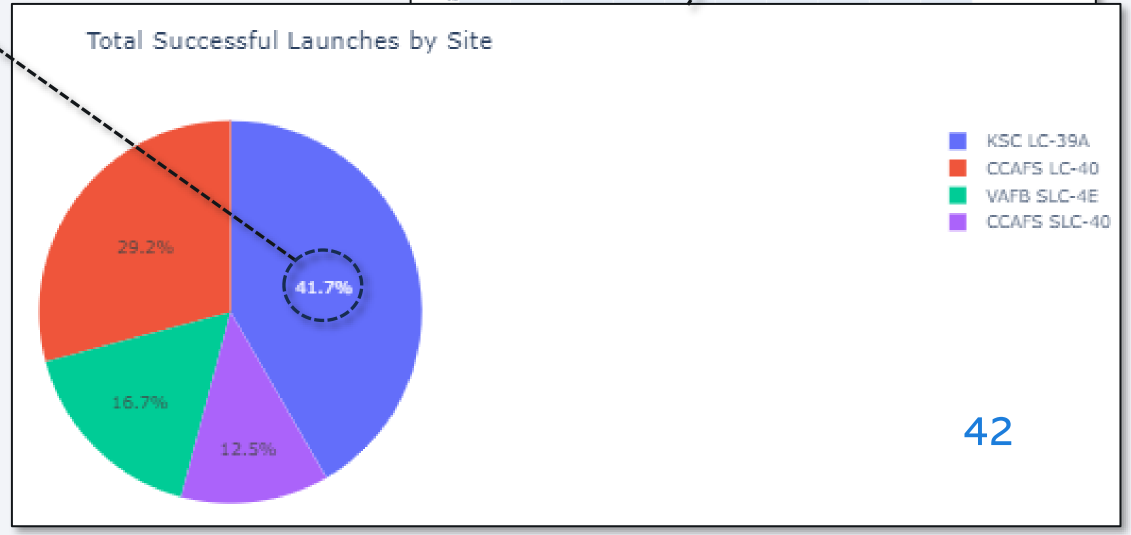
Section 4

Build a Dashboard with Plotly Dash

Successful Launches: Site Distribution

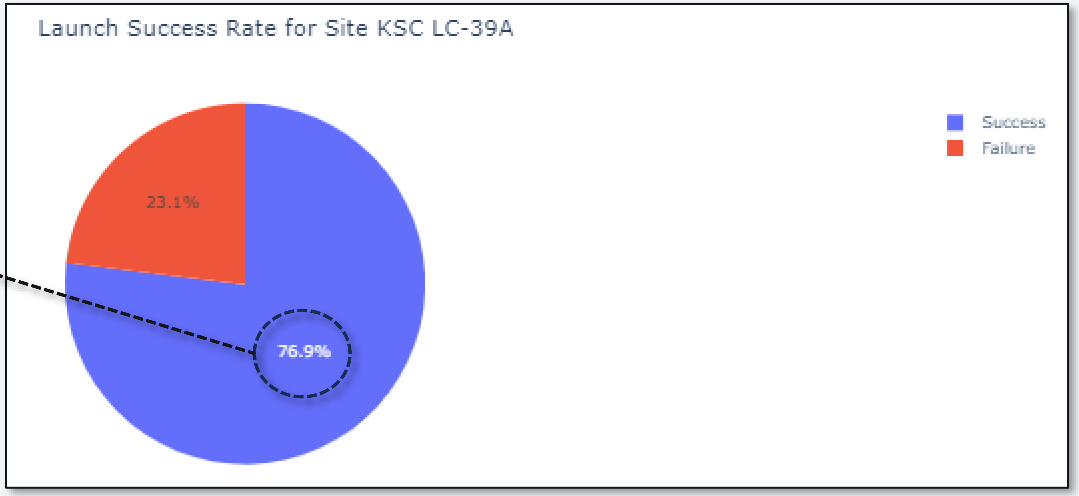
- Data indicated that the Kennedy Space Center (KSC LC-39A) launch site had the largest number of successful launches
  - Successful launches at KSC: 10
  - Successful launches, all Sites: 24
  - KSC contribution = 10/24 = 41.7%

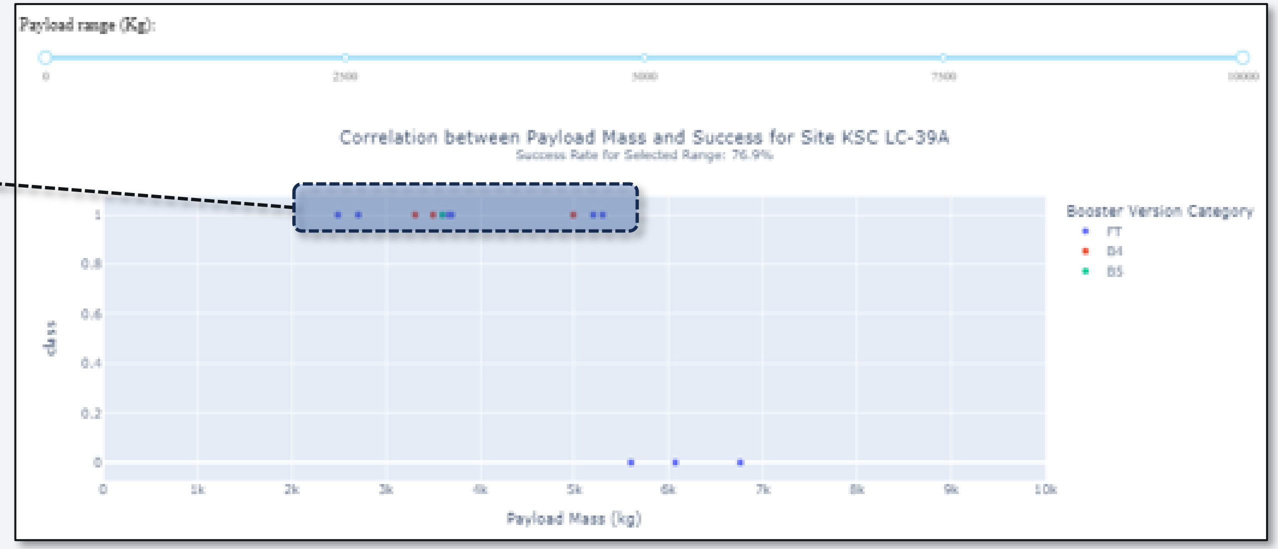




Site with Highest Launch Success Ratio

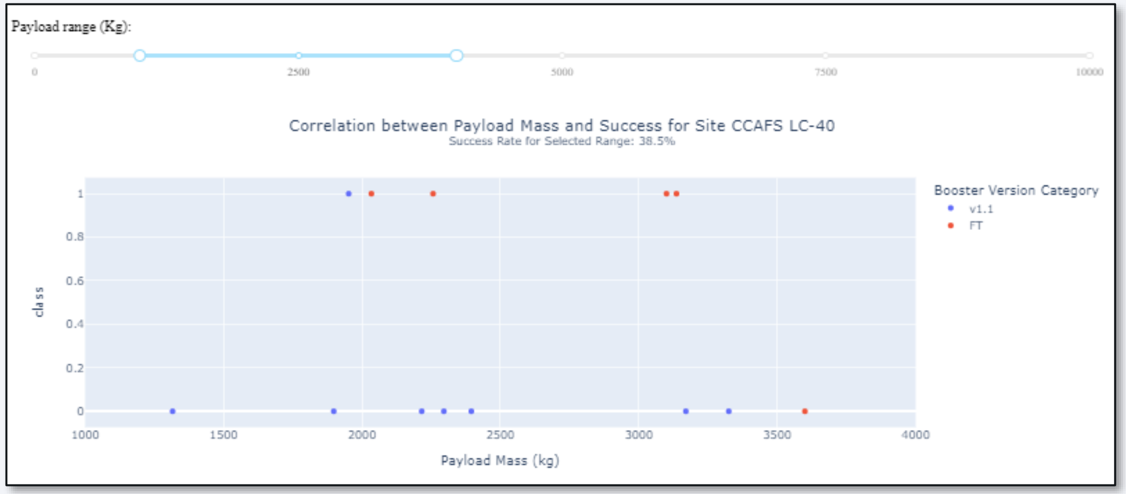
- Launch success ratio, by site:
  - KSC LC-39A 76.9%
  - CCAFS SLC-40 42.9%
  - VAFB SLC-4E 40.0%
  - CCAFS LC-40 26.9%
- Launches at KSC LC-39A were particularly successful when the payload mass was below 5500 kg



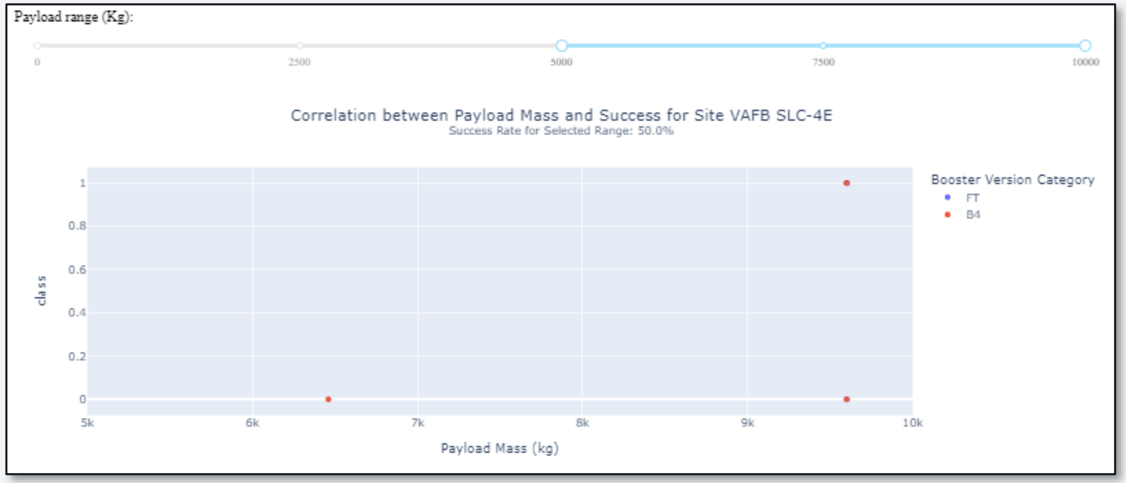


Effect of Payload Mass on Launch Outcome, by Site

- Site: CCAFS LC-40
  - Overall launch success rate was 26.9%
  - As shown at the right, the launch success rate was *higher* (38.5%) for payloads in the range of 1000 to 4000 kg

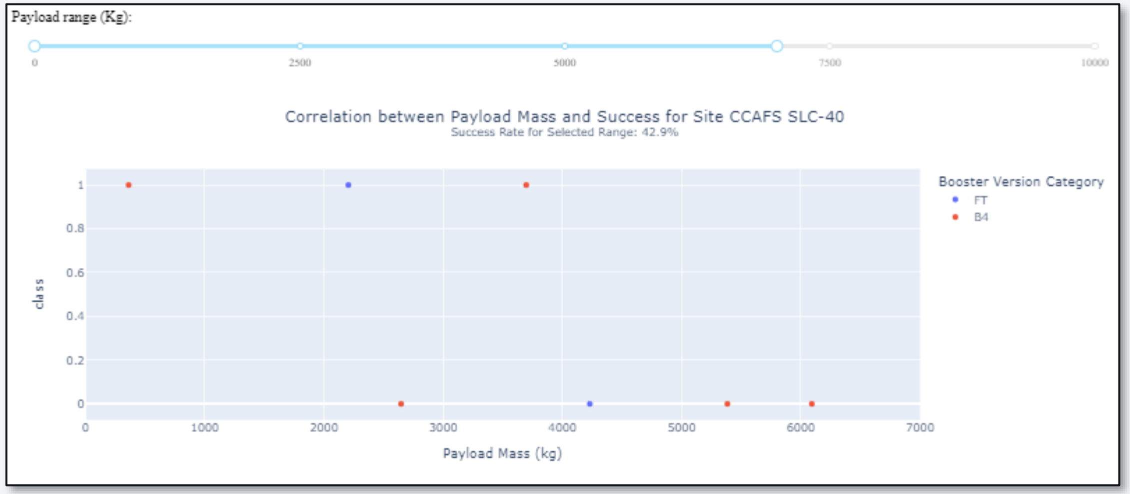


- Site: VAFB SLC-4E
  - Overall launch success rate was 40.0%
  - As shown at the right, the launch success rate was *higher* (50.0%) for payloads in the range of 5000 to 10000 kg

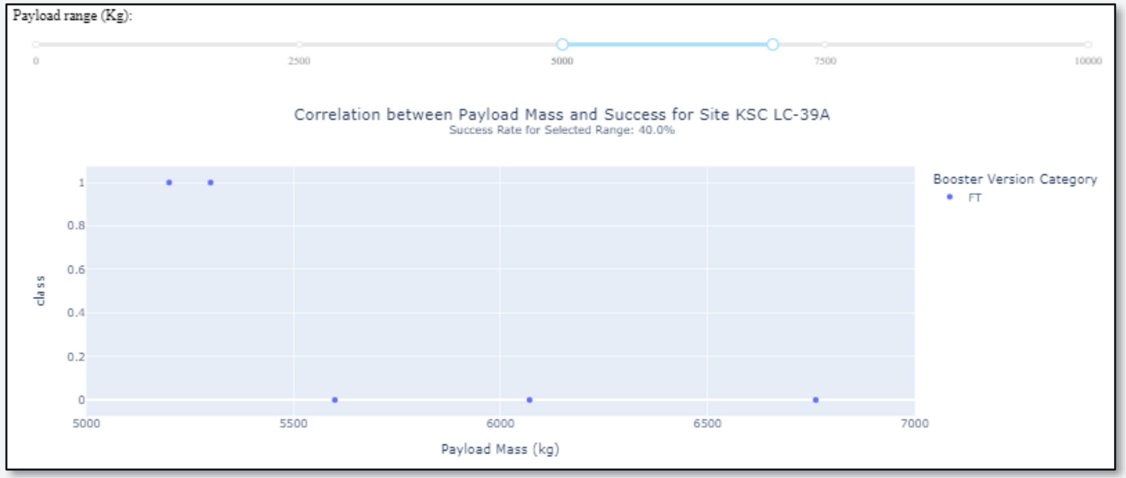


Effect of Payload Mass on Launch Outcome, by Site

- Site: CCAFS SLC-40
  - Overall launch success rate was 42.9%
  - As shown at the right, the launch success rate was *much lower* (0%) for payloads above 4000 kg

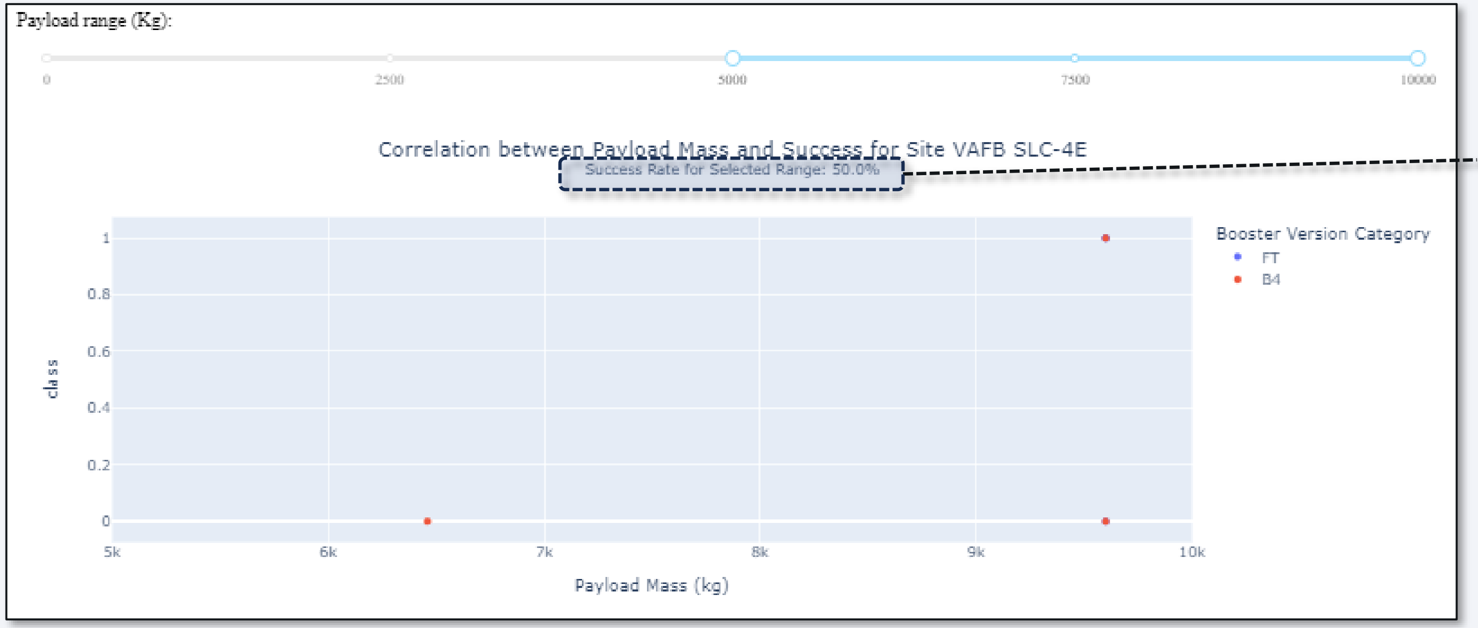


- Site: KSC LC-39A
  - Overall launch success rate was 76.9%
  - As shown at the right, the launch success rate was *lower* (40.0%) for payloads in the range of 5000 to 7000 kg, which all used the FT booster



Insight: Improving Inference of Success Rate from Scatter Plots

- When looking at scatter plots alone, it is easy to misinterpret the success rate
  - For example in the scatter plot below, it looks like there is one data point for a successful launch, and two data points for unsuccessful launches (implying a success rate of 1/3 = 33.3%)
  - In fact, there is one failure at 6460 kg, two failures at 9600 kg, and 3 successes at 9600 kg (meaning the actual success rate is 3/6 = 50%), but the points are plotted on top of one another



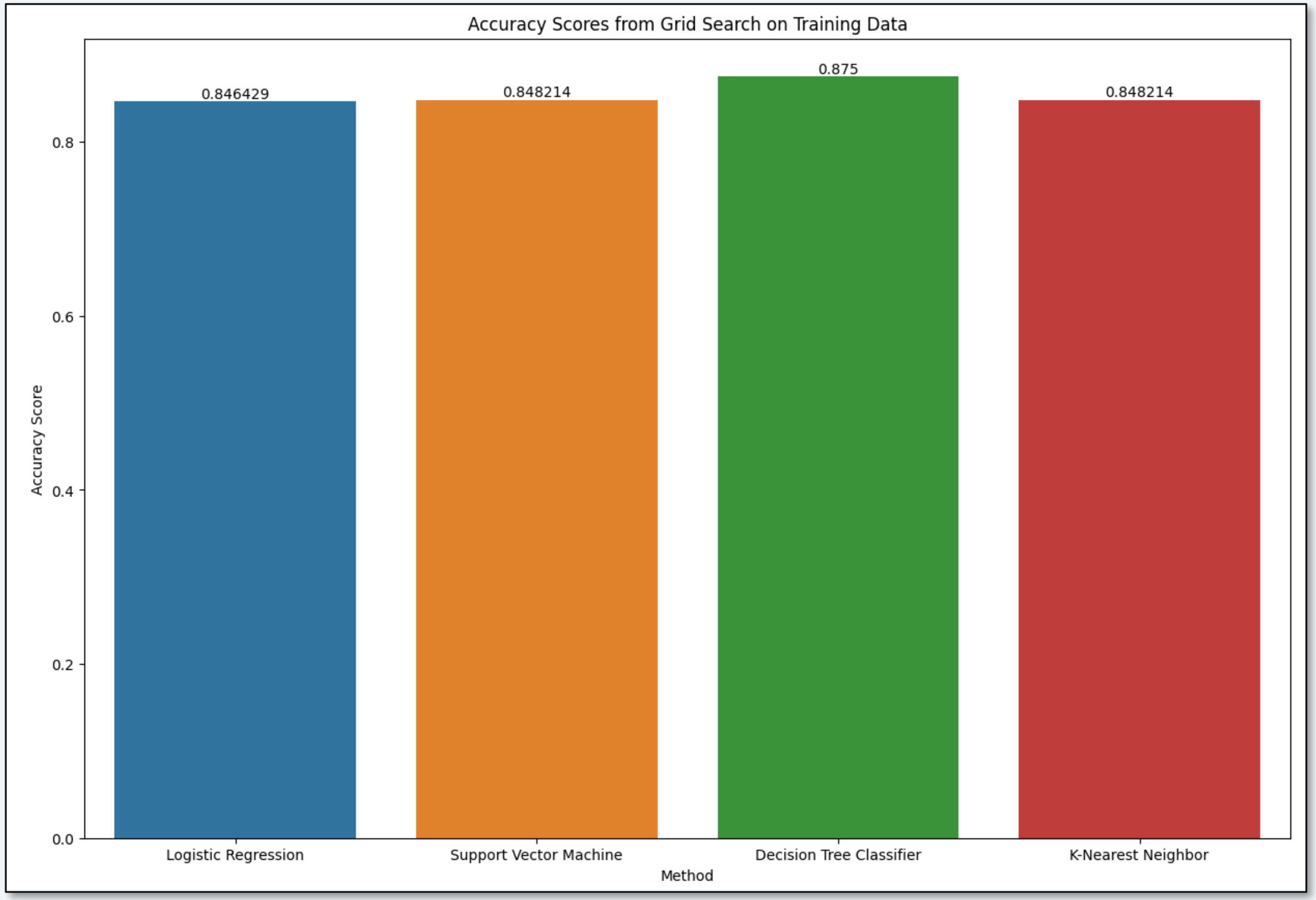
- To remedy this situation, this study added a sub-title that displays the actual success rate for the selected range

Section 5

Predictive Analysis (Classification)

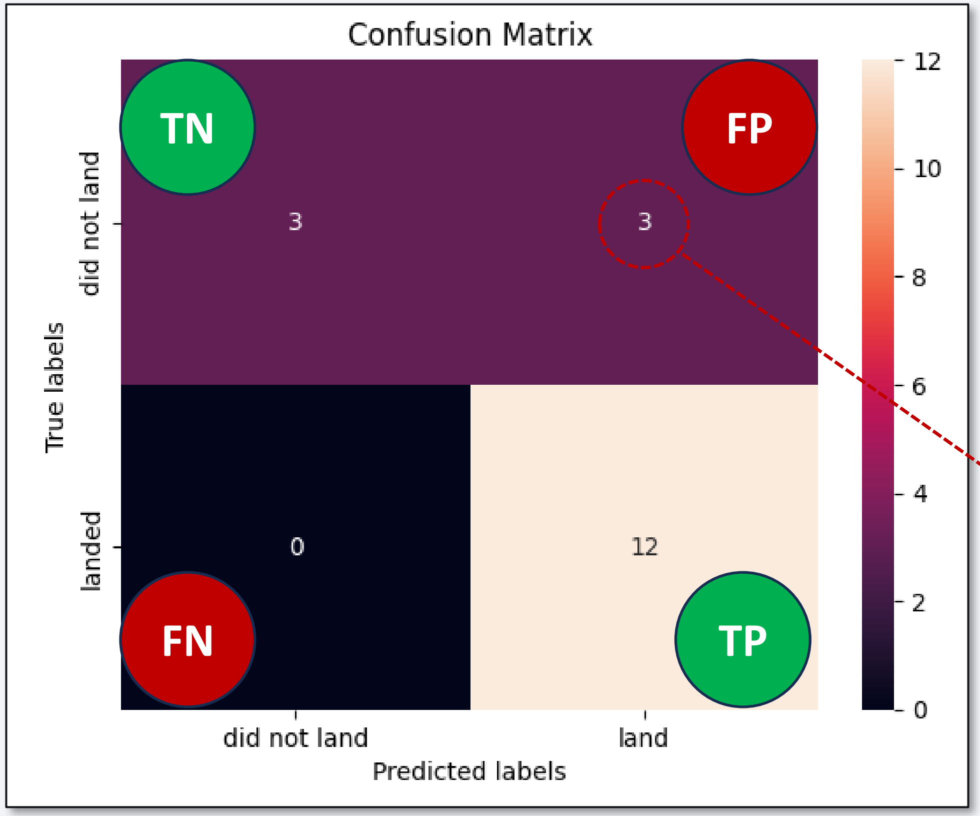
Classification Accuracy

- The accuracy scores for the various models (as measured by the `best_score_` attribute from the Grid Search fitting of the training data) were all fairly high and in a tight range
- However, the Decision Tree Classifier model had the best accuracy score (0.875)



Confusion Matrix

Decision Tree Classifier Model



- This Confusion Matrix indicates that the model performs well when making a “Negative” prediction (i.e., when predicting that a landing will fail)
  - The model did not have any “False Negative” (FN) predictions, although the total number of predicted negative outcomes was fairly small (3)
- However, the model does not perform as well when making a “Positive” prediction (i.e., when predicting that a landing will succeed)
  - The model had 3 “False Positive” (FP) predictions out of 15 total predictions – these were cases when the model predicted a successful landing but the actual result was a failure

Conclusions

Return to Outline

- This study identified multiple interesting correlations between launch-related parameters and launch outcomes, including success rate versus target orbit (link) and success ratio versus launch site (link)
- The study identified that the predictive model that was built using a Decision Tree Classifier was the most accurate of the model types evaluated
    - However, while this model performed very well in terms of avoiding "False Negative" predictions, the model had a high rate (20%) of "False Positive" predictions
- The next recommended step is to identify whether supplemental parametric data (e.g., weather conditions during landing attempt, indication of "intent" for failed landing) exists that would warrant repeating this analysis in an attempt to develop a more accurate predictive model

Appendix

Return to Outline

- Orbit Types
  - LEO: Low Earth orbit (LEO) is an Earth-centered orbit with an altitude of 2,000 km (1,200 mi) or less (approximately one-third of the radius of Earth), or with an orbital period of 128 minutes or less and an eccentricity less than 0.25
  - VLEO: Very Low Earth Orbits (VLEO) have a mean altitude below 450 km
  - GTO: A geosynchronous orbit is a high Earth (35,786 km = 22,236 mi) orbit that allows satellites to match Earth's rotation
  - SSO (or SO): A Sun-Synchronous Orbit (a.k.a. heliosynchronous orbit) is a nearly polar orbit around the earth, in which the satellite passes over any given point of the earth's surface at the same local mean solar time
  - ES-L1: This is a Lagrange point between the earth and the sun (i.e., a point where 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)
  - HEO: A Highly Elliptical Orbit (HEO) is an elliptic orbit with high eccentricity
  - ISS: The International Space Station (ISS) is a modular space station (habitable artificial satellite) in low Earth orbit
  - MEO: A geocentric orbit (a.k.a. intermediate circular orbits) that ranges in altitude from 2,000 km (1,200 mi) to just below geosynchronous orbit at 35,786 km (22,236 mi)
  - GEO: A circular geosynchronous orbit 35,786 km (22,236 mi) above Earth's equator and following the direction of Earth's rotation
  - PO: This is an orbit in which a satellite passes above or nearly above both poles of the earth

Appendix *(continued)*

- Mapping of “launch outcome” field

Landing Outcome	Meaning	“Class” mapping
True Ocean	Successfully landed to a specific region of the ocean	1
True ASDS	Successfully landed to a drone ship	1
True RTLS	Successfully landed to a ground pad	1
False Ocean	Unsuccessfully landed to a specific region of the ocean	0
False ASDS	Unsuccessfully landed to a drone ship	0
False RTLS	Unsuccessfully landed to a ground pad	0
None ASDS	Failure to land	0
None None	Failure to land	0

Thank you!