IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Patrick Murray
12/28/23

Outline

- Executive Summary
- Introduction
- Methodology
- Results
- Conclusion

Executive Summary

The commercial space sector is a rapidly developing and highly lucrative market. Competition among firms is intense with bids for contracts that are valued in the hundreds of millions. SpaceX holds a unique advantage in that it can re-use its first-stage rocket boosters, significantly reducing operating costs and subsequently offering lower contract bid prices.

The objective of this study is to examine data on SpaceX launches to see if we can predict the potential success or failure of a first-stage booster landing. Launches that are likely to fail will cost SpaceX approximately $165 million while successes cost only $62 million. Predicting the success or fail of a SpaceX launch allows us at SpaceY to offer competitive bids and thereby win contracts.

This study utilizes public SpaceX data that is collected with the SpaceX REST API as well as conventional web-scraping from the Falcon 9 Wikipedia page. This data encompasses launch data on Falcon 9 boosters, launch payloads, orbit, launch-site locations, and outcomes. The data is cleaned and normalized before machine-learning based statistical analysis is performed to identify factors affecting launch success. All coding is done in Python.

Preliminary analysis shows that launch site plays a significant role in success, with the KSC LC-39A and CCAFS LC-40 sites yielding approximately 10% more frequent failure outcomes. Orbit type also has a significant affect on outcomes with GTO, ISS, LEO, MEO, and PO orbits all underperforming relative to other orbit types. Heavy payloads (above 80,000 kg) are more likely to be successful in PO, LEO, and ISS orbits. A general trend is observed where flight success has increased over time, meaning later flight numbers are more likely to be successful and must be controlled for in our model.

Our predictive model therefore incorporates the following variables of interest – flight number, payload mass, orbit, and launch site. Data is split into training and test sets and run through four different supervised learning models. Grid search is used to find the hyperparameters giving best model fit. All four model types – logistic regression, support vector machine, decision tree, and k-nearest neighbors produce nearly identical goodness of fit results, approximately 83% accuracy. We recommend using logistic regression because of the lower computational cost on large datasets.

Given the assumptions and caveats of this project, it is proposed that a properly trained logistic regression model can be used to predict the success or failure of a SpaceX launch with sufficient accuracy for the purposes of calculating SpaceY's potential contract bid.

Introduction

The space race has re-ignited. This time the competition is among various commercial firms offering various space flight operations, including launch of payloads into orbit. Firms are vying for lucrative contracts that can total in the hundreds of millions of dollars. Key players include SpaceX, Virgin Galactic, Rocket Lab, and SpaceY – our own startup. SpaceX is the current leader in winning contracts due in large part to their ability to offer highly competitive bids at prices below what others offer. The primary advantage that SpaceX has over the competition is their Falcon 9 booster rocket. The Falcon 9 has the ability to automatically land after separating from the second stage, which significantly reduces operating costs of having to engineer and manufacture new boosters for each flight. This in turn allows SpaceX to offer lower bid prices that most firms can't match. The Falcon 9 isn't perfect however, and failed landings end up costing SpaceX considerably and in turn drive up their bidding prices.

We at SpaceY currently lack the ability to land and re-use our first stage boosters. Therefore if we wish to be competitive in this space and score contracts then we must find another way to price our bids appropriately and secure contracts that SpaceX is over-bidding for. The game theory assumption is that SpaceX knows ahead of time if their first-stage will succeed and thus they offer a very low bid and secure the contract. If SpaceX knows that their landing will fail then they offer a much higher bid to recoup the cost. It's here that we have the opportunity to offer our bid below SpaceX and secure the contract. Therefore, if we can anticipate the probability of a SpaceX first-stage landing failure ahead of the launch then we can price our bids accordingly.

There are multiple steps involved with our research process. In order to create a valid predictive model we first need to determine which variables have a statistically significant affect on launch success. We use exploratory data analysis to determine which variables have a significant causal effect on launch success. From here we need to develop a predictive model that can accurately calculate the likelihood of success for theoretical launch conditions.

The problems we seek to find answers for: What data is publicly available for this project? What cleaning and wrangling must be done so we can work with the data? Which variables are likely indicators for launch success? Which supervised machine learning method produces the most accurate predictive results? With these questions answered by this project it is possible for SpaceY to craft competitive bids and win contracts.

Section 1

Methodology

Methodology

The methodology for this project is divided as follows

- Data collection
- Data wrangling
- Exploratory data analysis (EDA) using visualization and SQL
- Interactive visual analytics using Folium and Plotly Dash
- Predictive analysis using classification models

Data Collection

- Methodology

- Publicly available data from SpaceX launches from 2010 through 2020 is collected using both the SpaceX REST API as well as conventional web scraping using Python package BeautifulSoup. Data is sourced from SpaceX's own data library at spacexdata.com as well as Wikipedia's Falcon 9 page, respectively. The raw data is converted into Pandas data frames for ease of further wrangling and filtering.

- Wikipedia entry for Falcon 9 Heavy Rockets: https://en.wikipedia.org/wiki/List_of_Falcon_9_and_Falcon_Heavy_launches

Data Collection – SpaceX API

- The SpaceX REST API is used to request multiple datasets from the SpaceX public data library at https://api.spacexdata.com/v4/launches/past

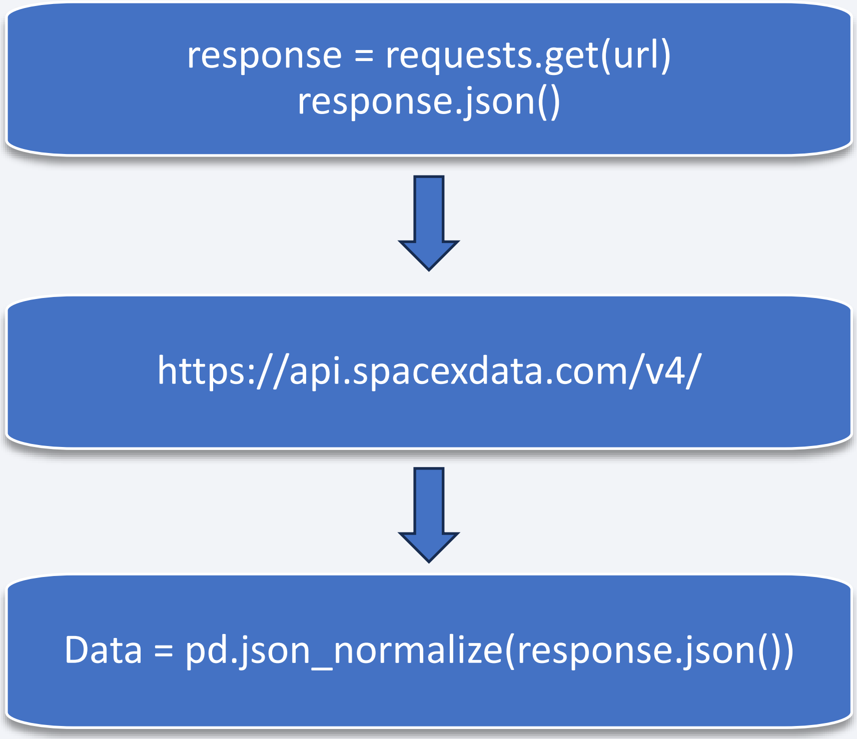
- Four requests are made, each to a specific sub-folder within https://api.spacexdata.com/v4/.

  /rockets/, /launchpads/, /payloads/, /cores/

- This is done to consolidate data into one primary dataset. For example, launchpad related data is amended with latitude, longitude, and location names from the /launchpads/ sub-folder.

- The collected data is normalized into a Pandas data frame for further cleaning and analysis.

- GitHub repository of Python code for SpaceX API.



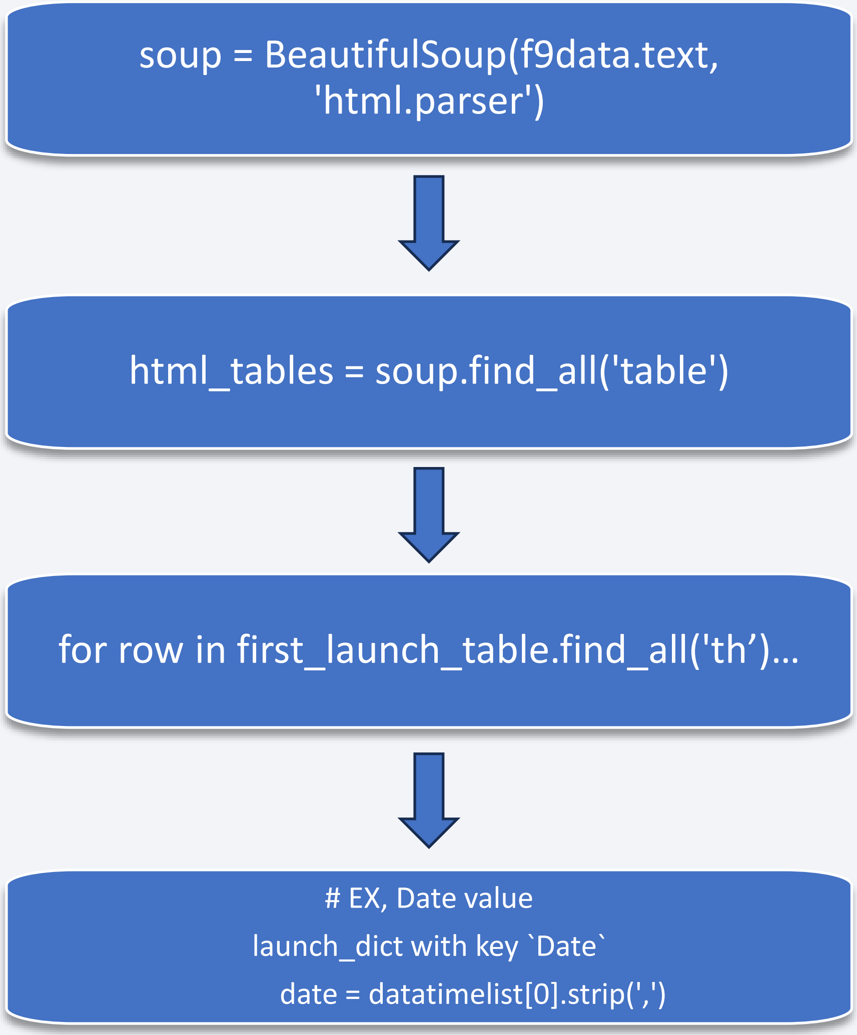
```
response = requests.get(url)
response.json()
```

https://api.spacexdata.com/v4/

```
Data = pd.json_normalize(response.json())
```

Data Collection - Scraping

- Additional data is collected from the Wikipedia entry for the Falcon 9 heavy rocket using the BeautifulSoup Python library. The Wikipedia entry is Here.
- The raw table data is scraped using the soup.find_all function on keyword ‘table’.
- Column names are extracted from the raw table data and assigned to their respective columns.
- Lastly, a data frame object named launch_dict is created and populated with the parsed data from the raw table.
- GitHub repository of Python code for BeautifulSoup based web scraping.



soup = BeautifulSoup(f9data.text, 'html.parser')

html_tables = soup.find_all('table')

for row in first_launch_table.find_all('th’)…

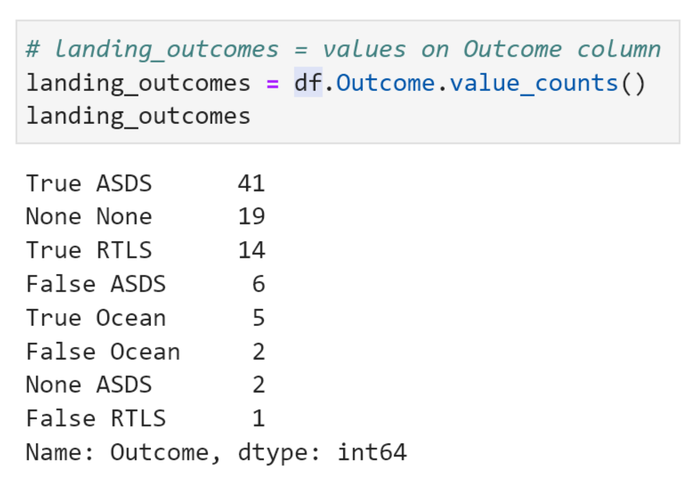
# EX, Date value
launch_dict with key `Date`
date = datatimelist[0].strip(',')

Data Wrangling

The first step after collecting the raw data is data wrangling. Data wrangling helps reveal initial patterns, distributions, data category types, and missing values. It helps us understand what the dataset contains and what issues may need to be addressed before beginning any statistical analysis. One notable shortcoming with the raw data is that the flight outcome is divided up into eight different categorical outcomes. We create a new binary variable “Class” which takes the value of 1 to indicate a true success while 0 indicates failure.

One final finding with the data wrangling EDA is that SpaceX has only a 66.6% success rate of all launches from 2010 to 2022. The implications of this are discussed further in the conclusion section.

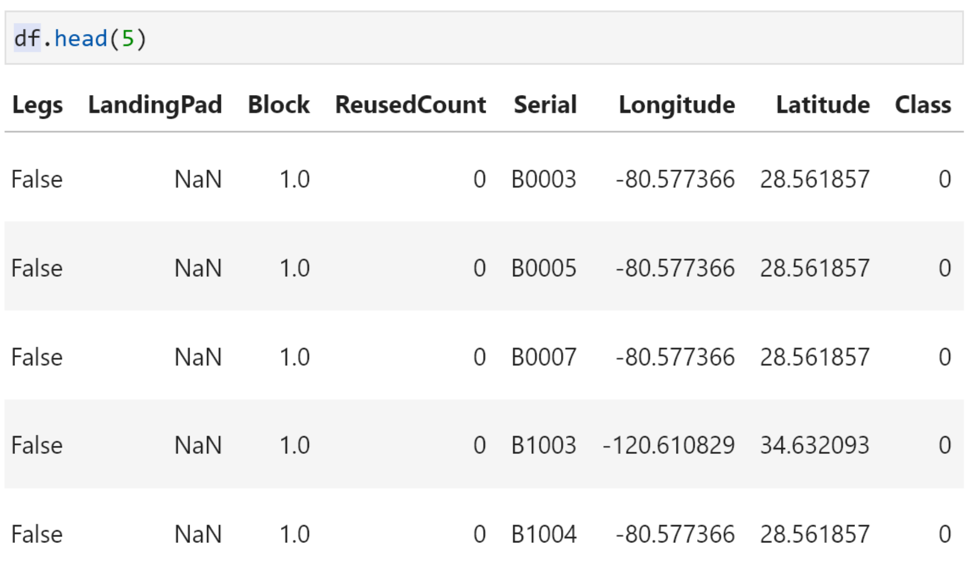
GitHub repository of preliminary data wrangling.



```
# landing_outcomes = values on Outcome column
landing_outcomes = df.Outcome.value_counts()
landing_outcomes
```

```
True ASDS      41
None None      19
True RTLS      14
False ASDS      6
True Ocean      5
False Ocean     2
None ASDS       2
False RTLS      1
Name: Outcome, dtype: int64
```

The eight different outcomes are consolidated into a binary variable. Here, a print of the first five outcomes are all true failures.



```
df.head(5)
```

Legs	LandingPad	Block	ReusedCount	Serial	Longitude	Latitude	Class
False	NaN	1.0	0	B0003	-80.577366	28.561857	0
False	NaN	1.0	0	B0005	-80.577366	28.561857	0
False	NaN	1.0	0	B0007	-80.577366	28.561857	0
False	NaN	1.0	0	B1003	-120.610829	34.632093	0
False	NaN	1.0	0	B1004	-80.577366	28.561857	0

EDA with Data Visualization

- Now the data is ready for more in-depth exploratory data visualization. This is done using the Python libraries Numpy, Pandas, and Seaborn.

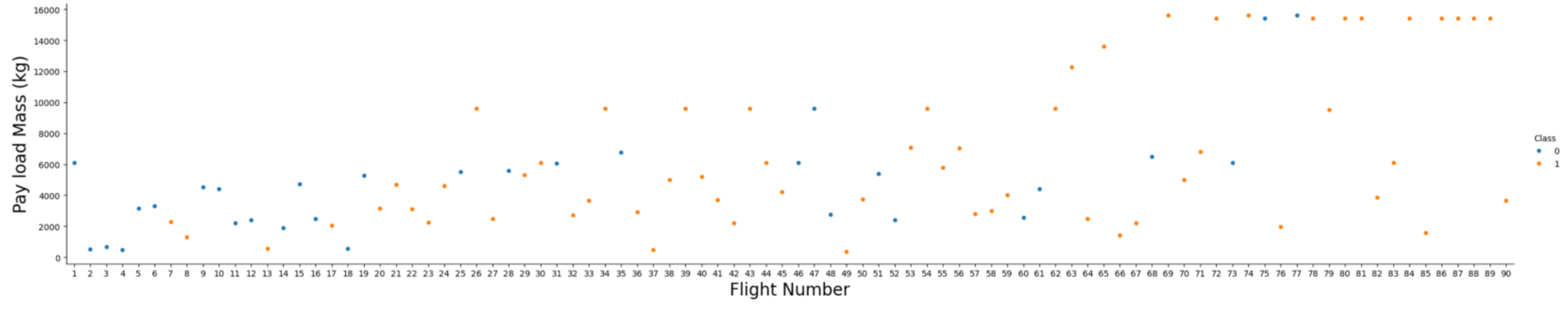
- One of the first relationships we're interested in is that of payload weight on flight success. A simple scatter chart gives the following information – There appears to be a positive correlation with flight number and success and there appears to be some positive correlation between payload mass and failure. These relationships warrant further statistical investigation in later stages of analysis and suggest potential variables of interest.

- Similar scatter charts were plotted for flight number vs launch, payload mass vs launch site, flight number vs orbit type, and payload mass vs orbit type. All these charts were used for a quick visual inference of potential relationships.

- Finally, a basic bar chart for orbit vs success rate and a line chart of year vs success rate gave general information about trends occurring independent of potential variables of interest – these trends were for higher success rates in certain orbits* as well as an increasing success rate over time, respectively.

- [GitHub repository of EDA.]()

*Higher success orbits included
ES-L1, GEO, HEO, SSO, and VLEO



Scatter plot of Flight Number vs Payload Mass (KG)

EDA with SQL

- SQL is used for quick queries on the SpaceX dataset and offer a glimpse at various outcomes and statistics that may be useful for formulating our predictive model.
- First, the raw data is imported and converted from a .csv file to a Pandas data frame. Blank rows are removed from the dataset.
- Next, a series of high-level queries are performed, including identifying the names of the unique launch sites, finding total payload mass launched by a specific buyer (ex, NASA), finding the total number of successful and failed missions, filtering queries by dates and weights, and ranking landing outcomes by count. Other more specified queries such as listing the names of boosters that have carried the maximum payload weight are used for investigating potential outliers or trends.
- GitHub repository of SQL code is available here.

Build an Interactive Map with Folium

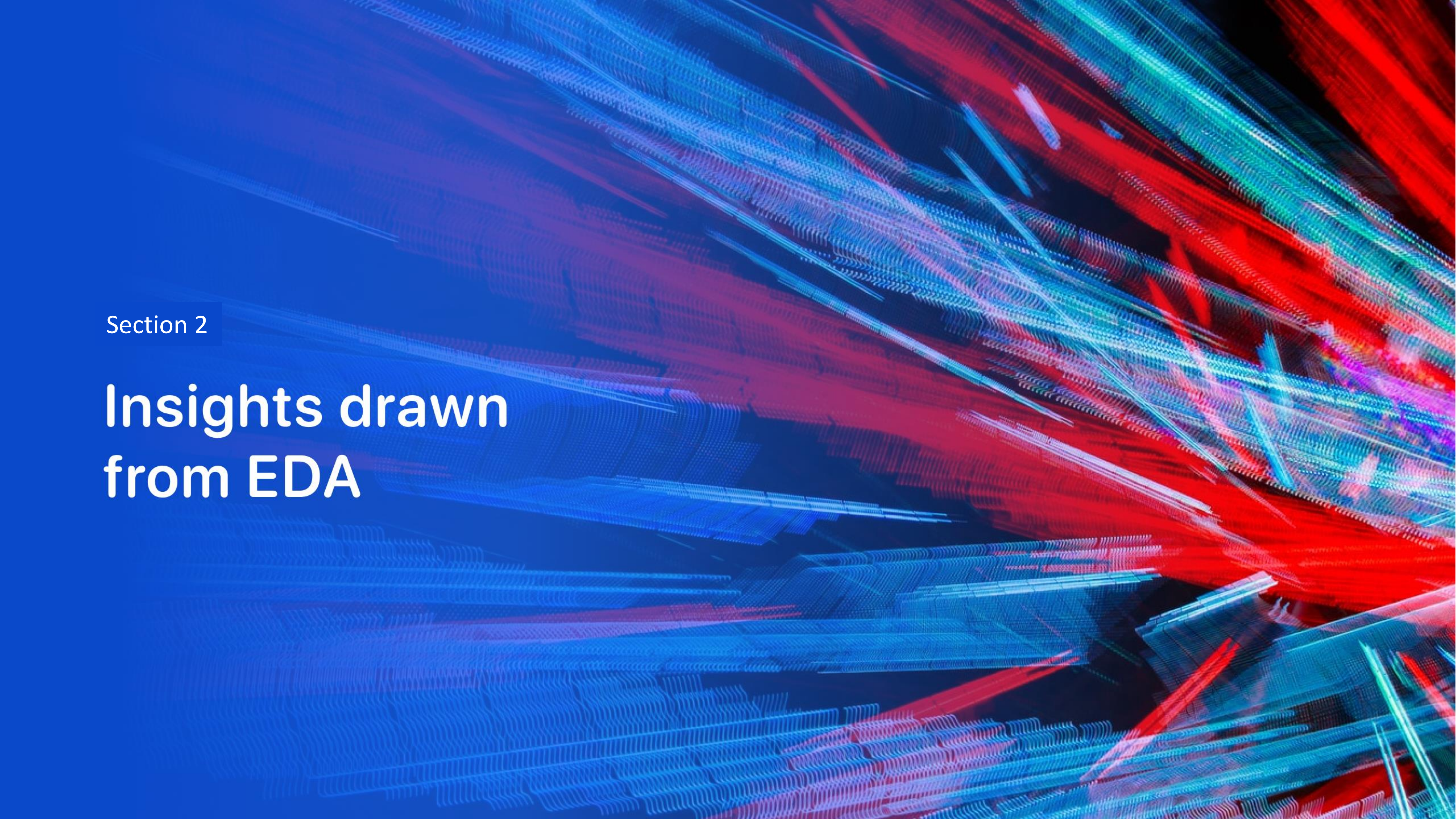
- Geographic mapping is another useful tool for identifying data points and other aspects of potential interest to our predictive model. For example, we may find that launch sites share common characteristics such as proximity to infrastructure, bodies of water, or similar longitudes. This information is useful to know because some launch sites may be better suited for different orbits or mission types than others and thus have higher success rates. This should be controlled for in our model.
- To complete this goal, we added various map markers indicating launch sites with corresponding success or failure indicators for those specific launches, lines to nearby objects of interest for calculating distances, as well as labels for general convenience.
- A few general trends were noted thanks to this mapping. First, is that SpaceX launch sites are located nearby to Oceans, no further than 10km from the Pacific and less than 1km from the Atlantic. No launch site is further North than -120.6 longitude. Florida launch sites saw the majority of missions with 46 total launches compared to California’s 10. Launch sites are located either within wildlife reserves or non-private areas and are at least 10km away from the nearest city or residential area. All launch sites have immediate adjacent access to rail lines.
- GitHub repository for Folium mapping here.

Build a Dashboard with Plotly Dash

- At this point we have a general idea of variables of interest and what sorts of relationships they may have with launch success. To further explore the determinants of a successful launch we drill down to greater specifics within the data. Comparisons can easily be made by utilizing a dashboard with dynamic graphs and plots that change depending on which data we want presented.
- Because we've seen possible trends with launch site location affecting success we're focusing the dashboard on filtering the data by specific launch site or by showing all launches. Specifically, this dashboard has one dropdown menu that allows us to select which launch site data to display. By default the dashboard shows data for all launch sites.
- Additionally, each launch site has a pie chart and a scatter plot. The pie chart shows a total site specific count of successful and failed launches. The scatter plot graphs payload mass (in kg) versus outcome. Finally, a slider is added so the user can filter by a specified payload or range in weight.
- The dashboard confirms some earlier observations. Rocket booster v1.1 has the greatest percentage of failures regardless of site or payload while the FT booster has the greatest percentage of success, also independent of site or payload. Launch site CCAFS SLC-40 has the greatest success rate at 42.9% of launches, however this site only had a total of 7 recorded launches in the data.
- GitHub repository of dashboard code is available here.

Predictive Analysis (Classification) and Results

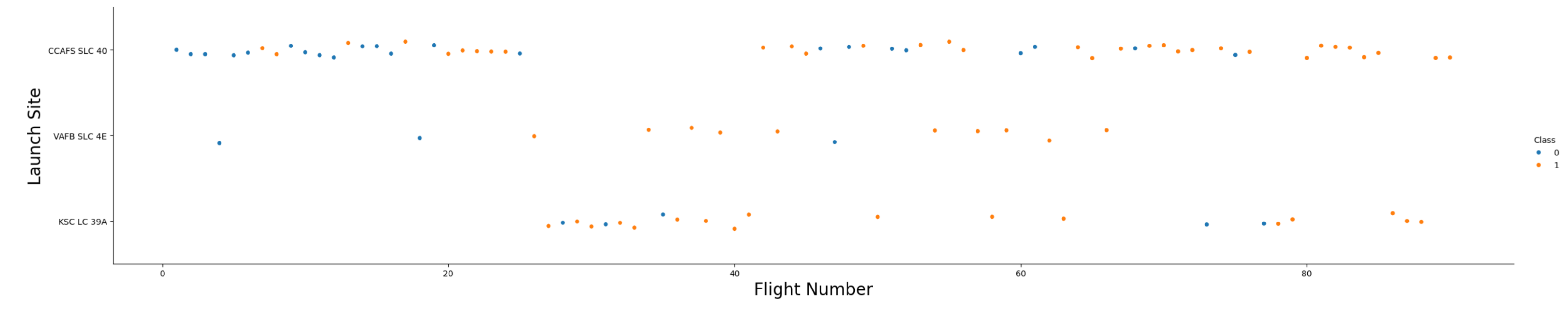
- The final step in this project is predictive analysis. Specifically, determining what sort of classification model is best suited for our business problem of predicting SpaceX launch successes.
- From our earlier analysis we have identified a number of potential determinants for launch success including payload mass, orbit type, launch site, and booster rocket type. Our model will therefore be multi-variable.
- We test four different supervised learning models to determine which is most suitable for predictive classification; logistic regression, support vector machine, decision tree, and k-nearest neighbors. Our dataset was split into training and testing sets and run on each of the four model types. The models are trained on the training test set and a grid search is performed to identify the hyperparameters that best fit the model to the data. Goodness of fit scores as well as confusion matrices are calculated and compared to determine the best model.
- We find that all four models perform very similarly, approximately 83% accuracy in correctly predicting outcomes. This is due to the limited sample size in this study, only 18 test samples are available. We recommend revisiting this model testing in the future if more data becomes available.
- Since the performance of the four models is so similar we recommend using logistic regression because it is one of the lest computationally intensive model types and scales well with more data.
- GitHub repository is available here.



Section 2

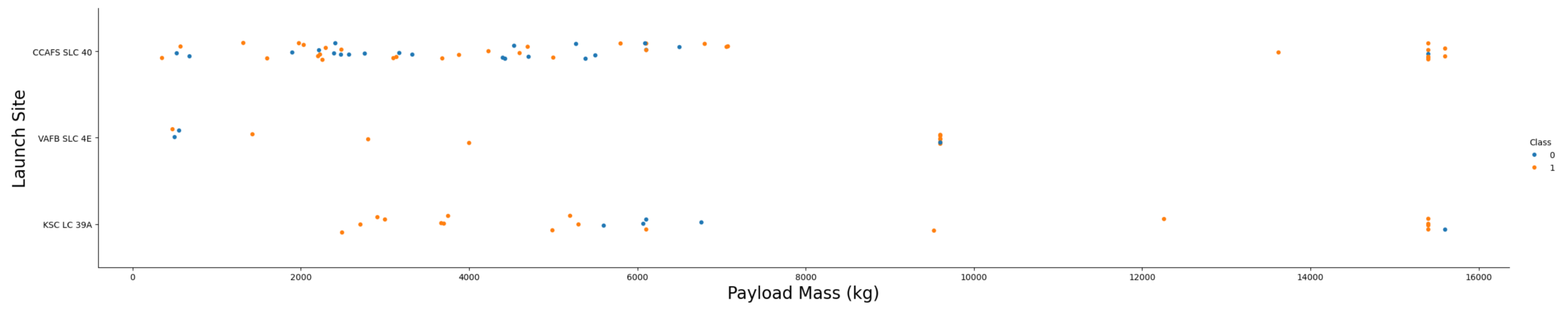
Insights drawn from EDA

Flight Number vs. Launch Site



The scatter plot of flight numbers versus launch site is useful to check for trends over time. We notice that launch site CCAFS SLC 40 had the majority of early flights, a pause around flights 25-42 where it appears SpaceX shifted to using KSC LC 39A for most flights, and then afterwards once again had the majority of flights. There is also a notable increase in successes from CCAFS SLC 40 after that pause, possibly indicating some improvements to the site itself that increased the odds of success.

Payload vs. Launch Site



Plotting payload mass against launch site shows a fairly even distribution of payload masses amongst the launch sites, it doesn't appear there was any preference for one site over another.

The only notable observations are that heavier payloads, above 8,000kg, were more likely to succeed. Launch site KSC LC 39A had no failures at lower payload weights, below 5,000kg.

Success Rate vs. Orbit Type

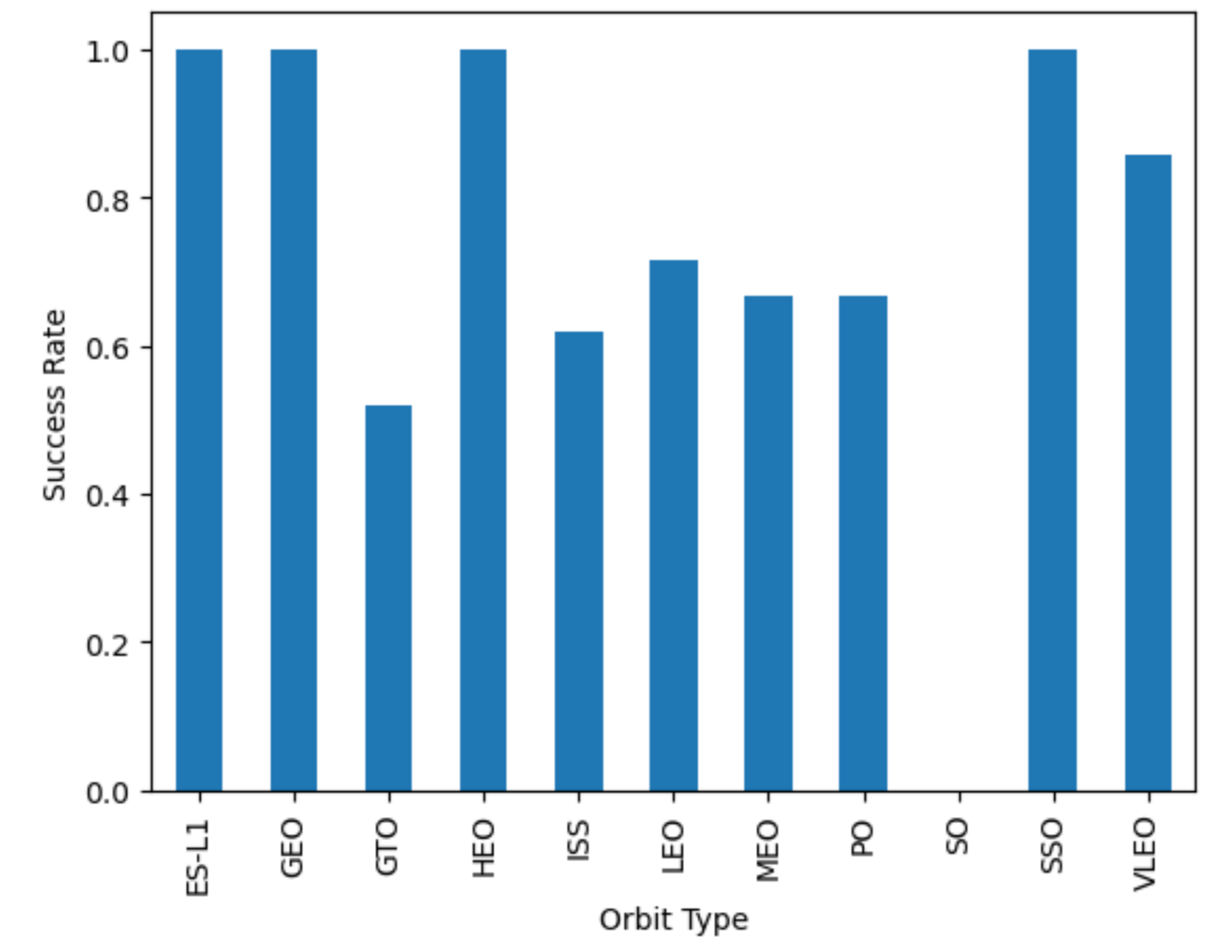
Orbit type appears to play an important factor in launch success as there is a clear separation in outcome probabilities.

ES-L1, GEO, HEO, and SSO all had perfect success rates for launch outcome.

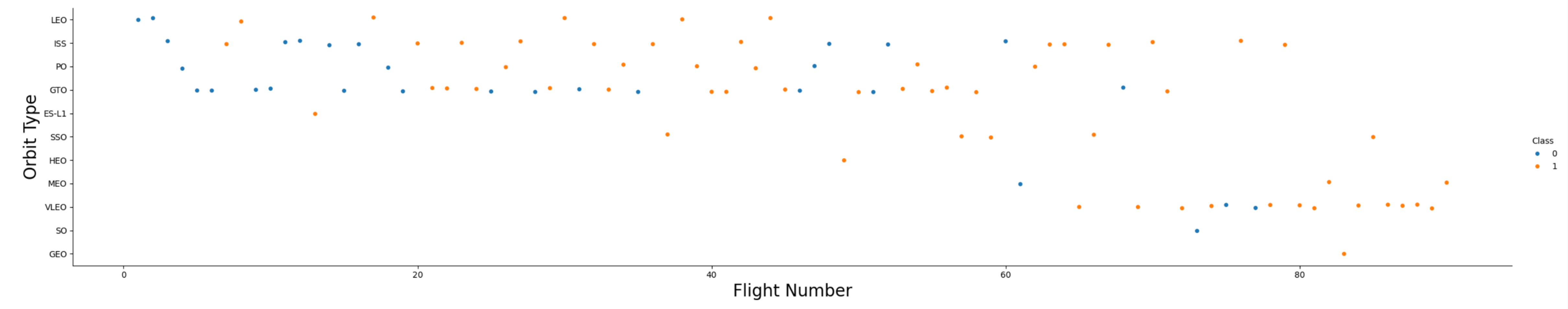
There is no discernable pattern to the orbit type and success however. GTO is a high altitude orbit (approximately 35,786km) similar to ES-L1 and GEO but with a significantly worse success rate.

Similarly, VLEO has an extremely low orbit but a very high success rate.

Because orbit type is not consistent in predicting success rate we deem it likely that there are other factors that are stronger indicators for success.



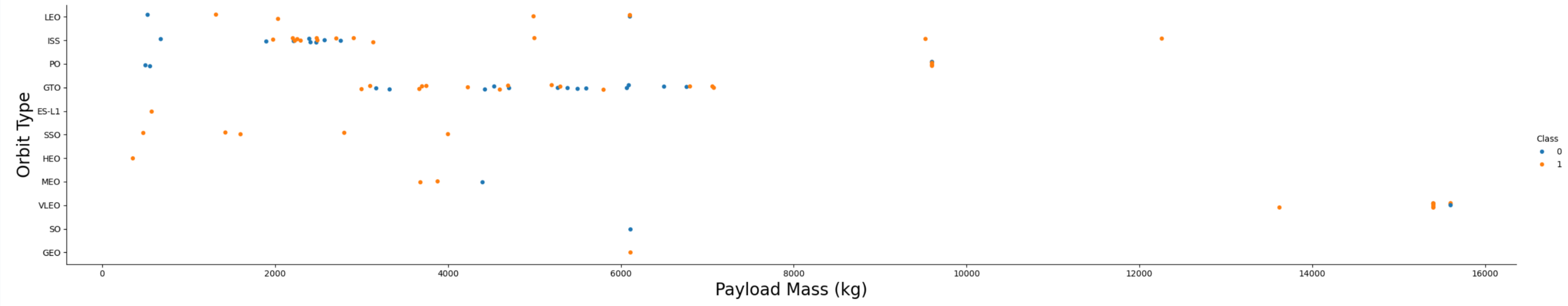
Flight Number vs. Orbit Type



An interesting trend emerges when viewing flight numbers plotted against orbit type. Around flight number 60 there appears to be a shift away from ISS, PTO, and GTO orbit types and towards VLEO.

GTO orbits show an increase in success rates. Of the 27 flights in GTO orbit there is an increase in success after the first 13 flights. Put in other words, the second half of flights in GTO orbit had a greater success rate than the first half. There doesn't appear to be a similar increase in success rates for other orbit types around this range in total flight numbers. This suggests some sort of improvement was made specific to this orbit type.

Payload vs. Orbit Type



Plotting out payload mass versus orbit type doesn’t yield much obvious information. The heaviest payloads, 13,000 kg and greater, were exclusively launched to VLEO orbit. ISS orbit has the largest spread in payloads, ranging from approximately 1,000 kg to 12,000 kg.

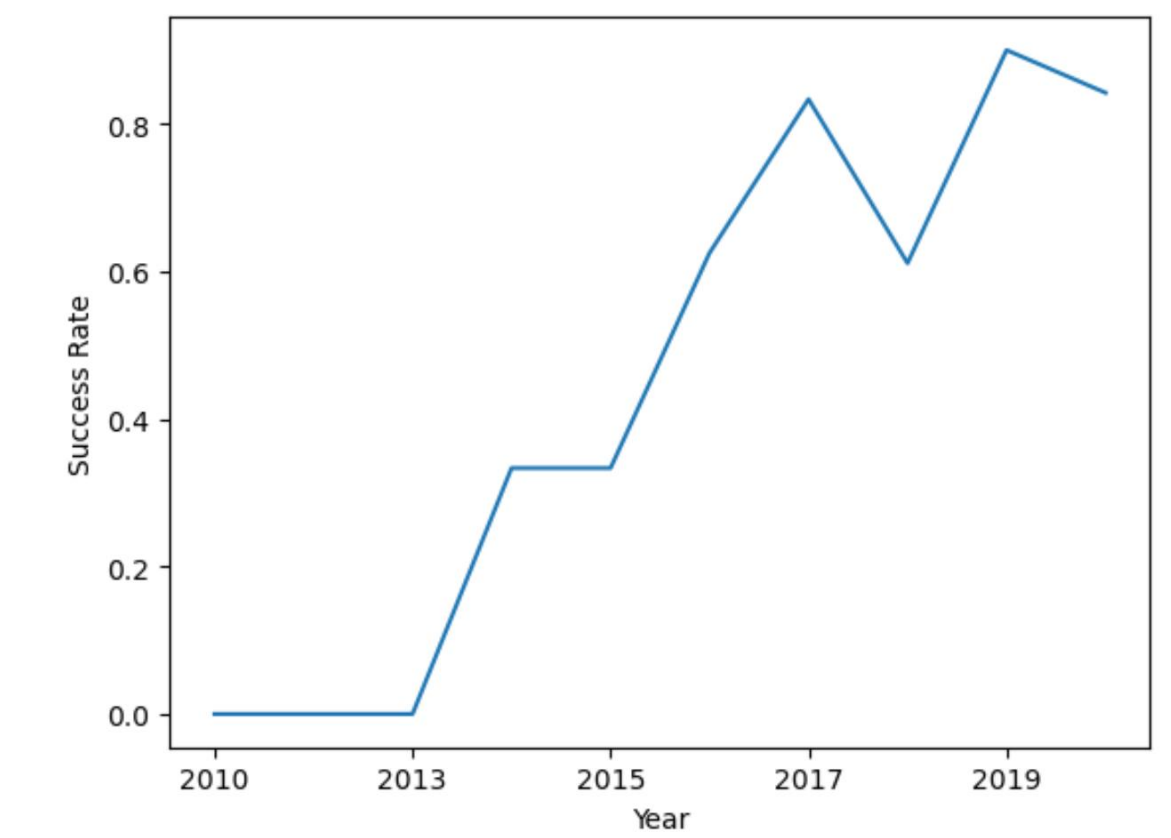
Ultimately there aren’t enough data points to draw any solid conclusions about payload and orbit type.

Launch Success Yearly Trend

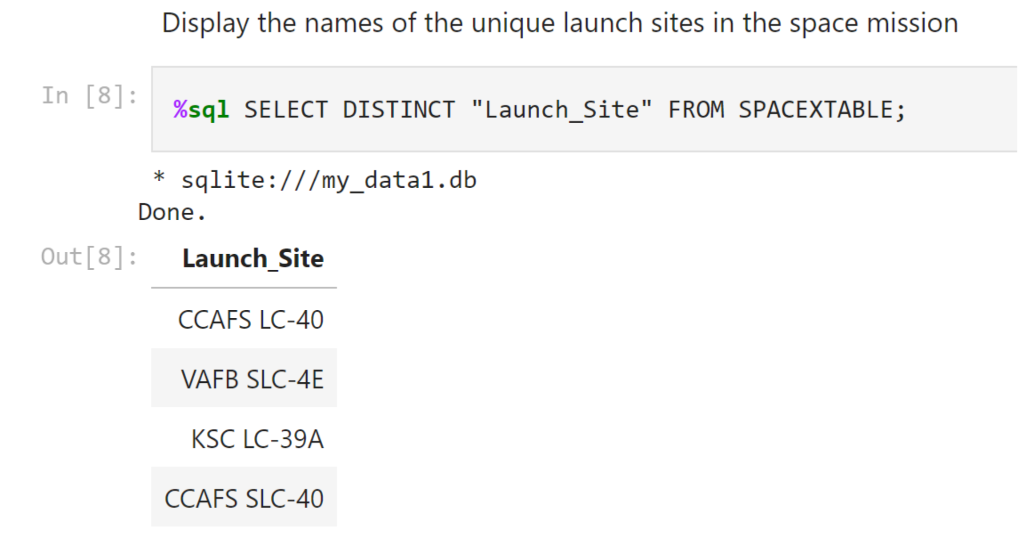
SpaceX has clearly demonstrated an increasing success rate over time, with steady improvements starting in 2013 and peaking in 2019.

This means we need to account for the fact that launches with higher flight numbers (i.e., launched in later years) are more likely to be successful due to improvements not captured by our other explanatory variables.

If we believe this trend will continue then future launches will have a naturally higher likelihood of success independent of any other explanatory variable. This will hurt the accuracy of our predictive model, so we will include the year a rocket is launched as a control variable to account for this positive trend over time.



All Launch Site Names



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

```
In [8]: %sql SELECT DISTINCT "Launch_Site" FROM SPACEXTABLE;
```

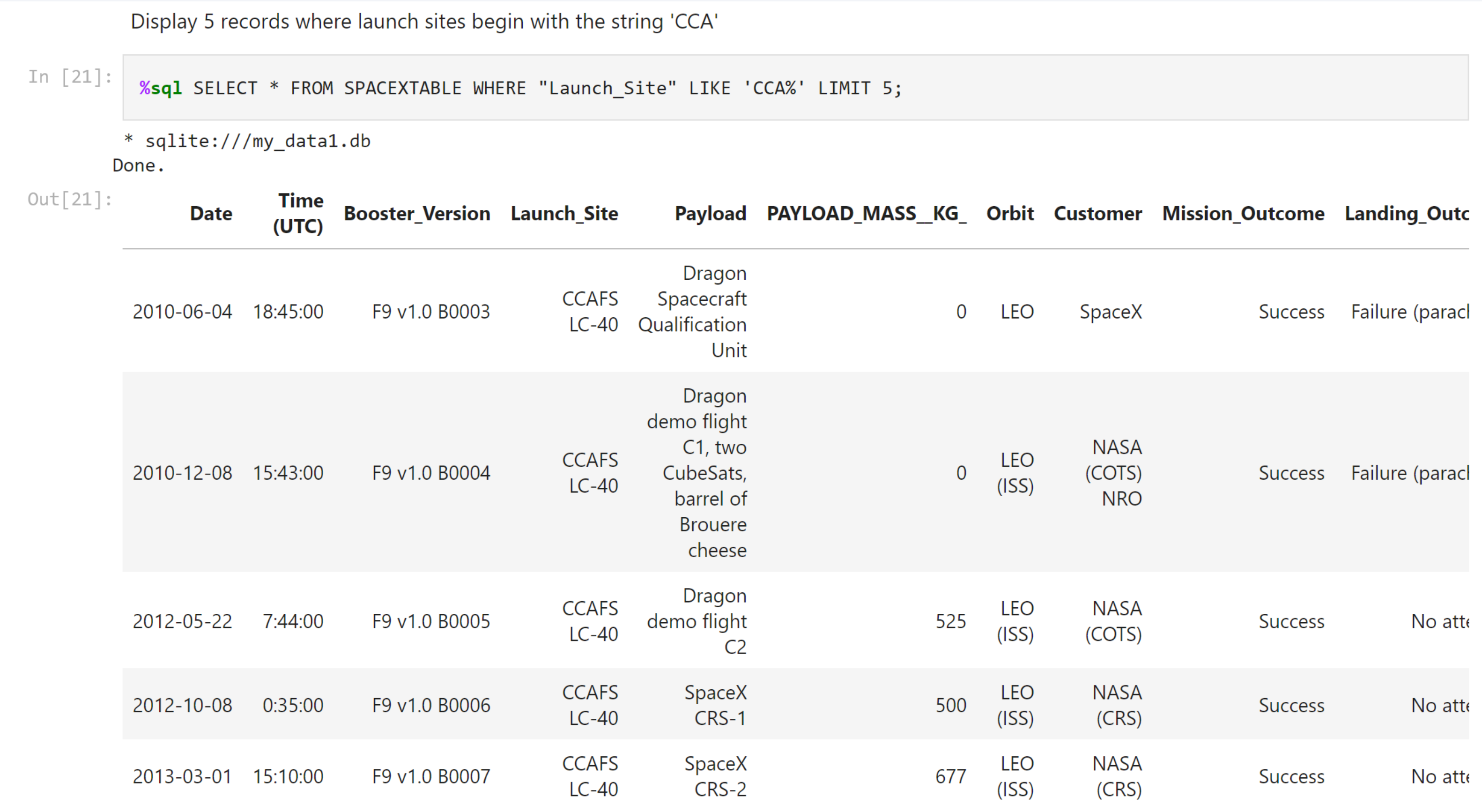
```
 * sqlite:///my_data1.db
Done.
```

Out[8]:

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

- SQL magic command to query for all distinct launch sites within the SPACEXTABLE table. There are four sites returned.

Launch Site Names Begin with 'CCA'



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

```
In [21]: %sql SELECT * FROM SPACEXTABLE WHERE "Launch_Site" LIKE 'CCA%' LIMIT 5;
```

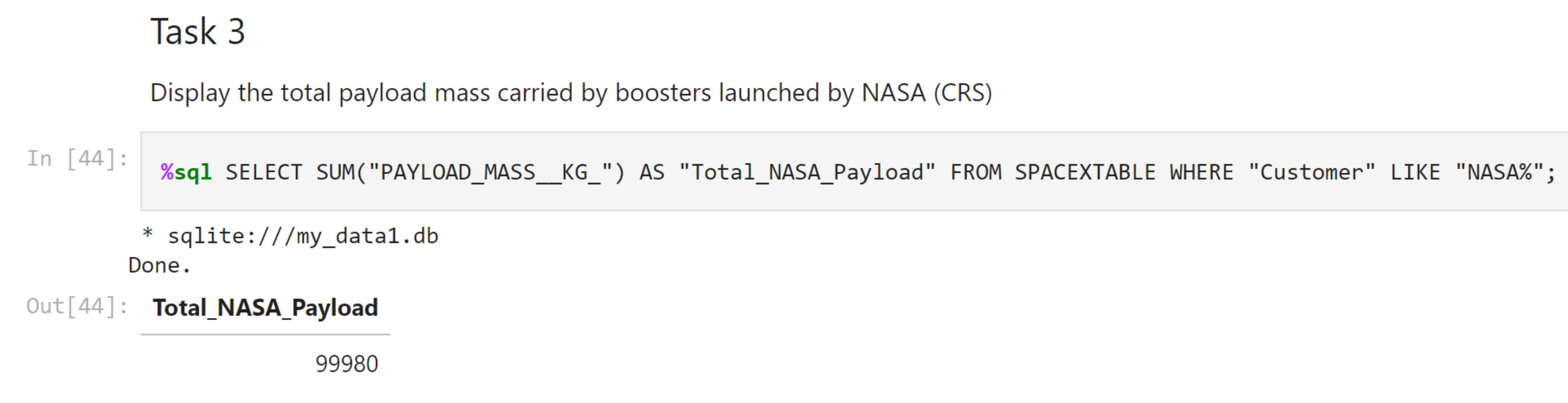
```
 * sqlite:///my_data1.db
Done.
```

Out[21]:

Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Orbit	Customer	Mission_Outcome	Landing_Outc
2010-06-04	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success	Failure (paracl
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 (paracl
2012-05-22	7:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success	No atte
2012-10-08	0:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500	LEO (ISS)	NASA (CRS)	Success	No atte
2013-03-01	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677	LEO (ISS)	NASA (CRS)	Success	No atte

- SQL magic query using LIKE operator to return any launch site starting with ‘CCA’.

Total Payload Mass



Task 3

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

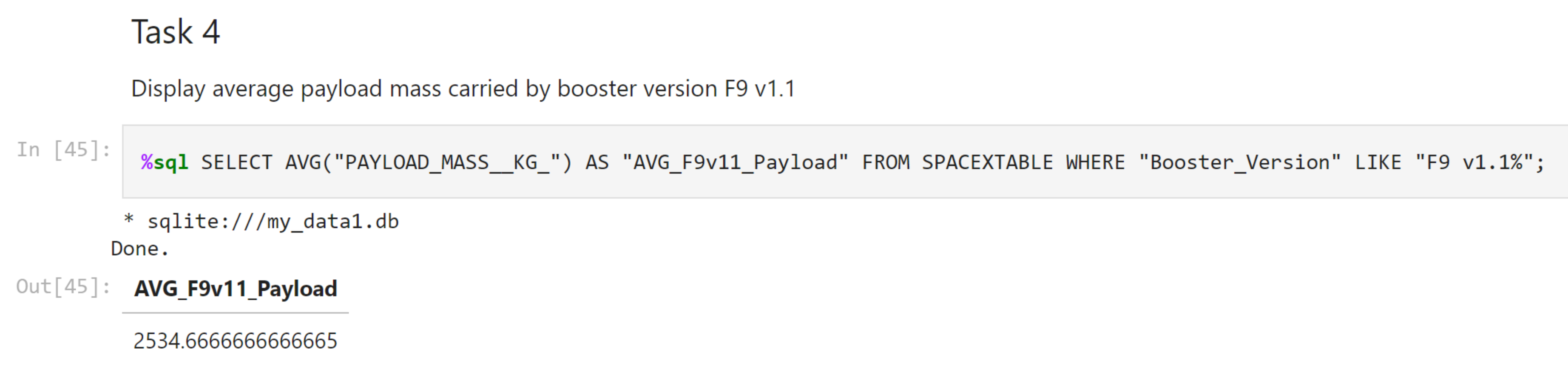
```
In [44]: %sql SELECT SUM("PAYLOAD_MASS__KG_") AS "Total_NASA_Payload" FROM SPACEXTABLE WHERE "Customer" LIKE "NASA%";

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

Out[44]:

Total_NASA_Payload
99980

- SQL magic query using SUM function on the column “PAYLOAD_MASS__KG__” using WHERE operator to filter out only entries where the Customer starts with “NASA”. The total payload for all missions ordered by NASA is 99,980 kg.

Average Payload Mass by F9 v1.1



Task 4

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

```
In [45]:  %sql SELECT AVG("PAYLOAD_MASS__KG_") AS "AVG_F9v11_Payload" FROM SPACEXTABLE WHERE "Booster_Version" LIKE "F9 v1.1%";
```

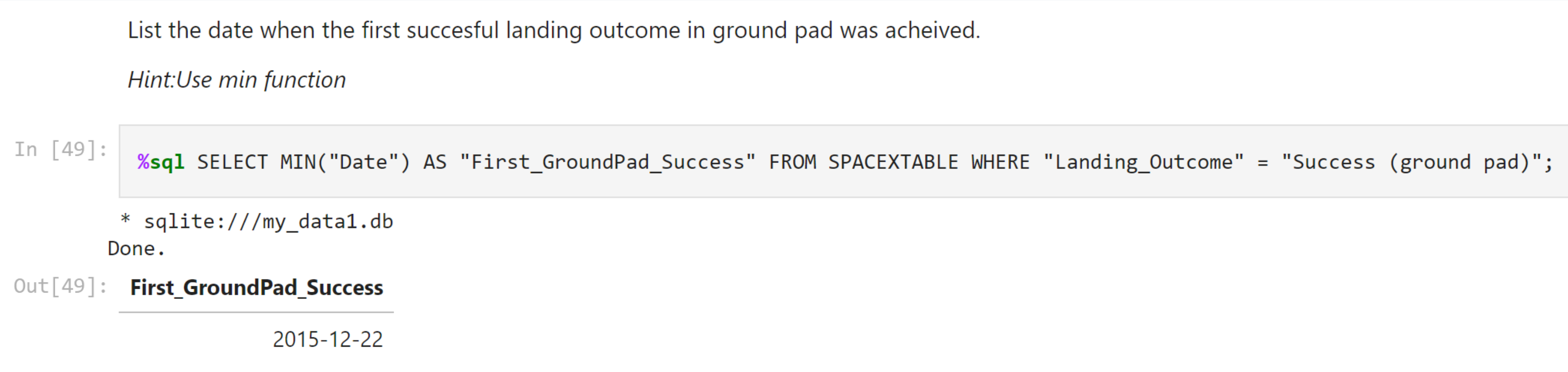
```
 * sqlite:///my_data1.db
Done.
```

Out[45]:

AVG_F9v11_Payload
2534.6666666666665

- SQL magic query using AVG function and the WHERE clause to filter for only “F9 v1.1” boosters. The average payload of all F9 v1.1 booster rockets is 2534.6 kg.

First Successful Ground Landing Date



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

Hint:Use min function

```
In [49]: %sql SELECT MIN("Date") AS "First_GroundPad_Success" FROM SPACEXTABLE WHERE "Landing_Outcome" = "Success (ground pad)";
```

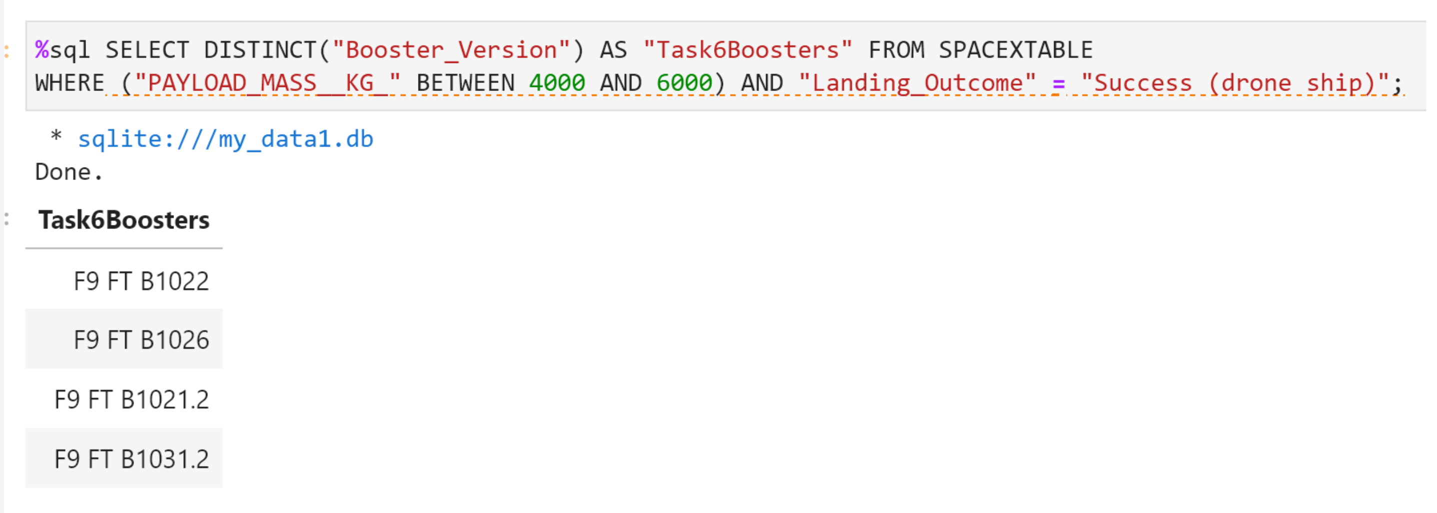
```
 * sqlite:///my_data1.db
Done.
```

Out[49]:

First_GroundPad_Success
2015-12-22

- SQL magic query using MIN function on the “Date” column to find the ‘smallest’ value, which is the earliest date. This MIN function is performed on all values for which the landing outcome was the specific outcome of “success (ground pad). The first successful ground pad landing was December 23, 2015.

Successful Drone Ship Landing with Payload between 4000 and 6000



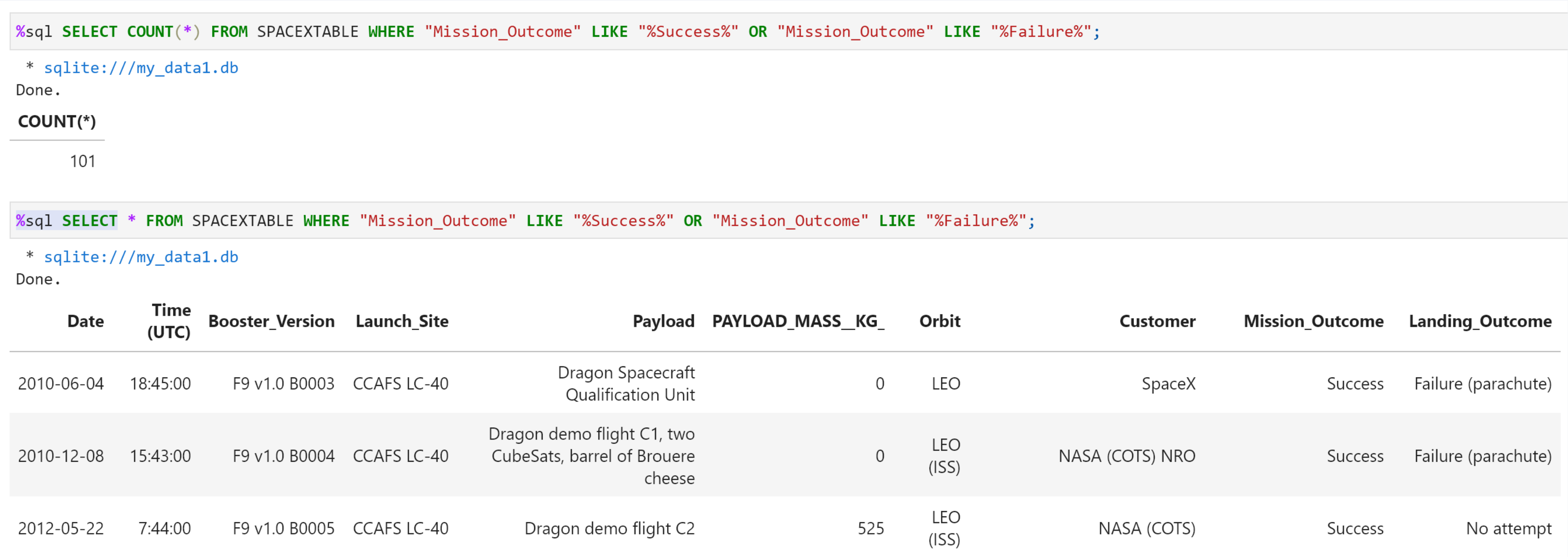
```
%sql SELECT DISTINCT("Booster_Version") AS "Task6Boosters" FROM SPACEXTABLE
WHERE ("PAYLOAD_MASS__KG_" BETWEEN 4000 AND 6000) AND "Landing_Outcome" = "Success (drone ship)";
```

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

Task6Boosters
F9 FT B1022
F9 FT B1026
F9 FT B1021.2
F9 FT B1031.2

- SQL magic query using the AND operator to combine two conditions – the query must be within the payload weight range of 4,000 and 6,000 and have a successful landing on a drone ship. Four boosters fit this criteria and are returned.

Total Number of Successful and Failure Mission Outcomes



```
%sql SELECT COUNT(*) FROM SPACEXTABLE WHERE "Mission_Outcome" LIKE "%Success%" OR "Mission_Outcome" LIKE "%Failure%";
```

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

COUNT(*)
101

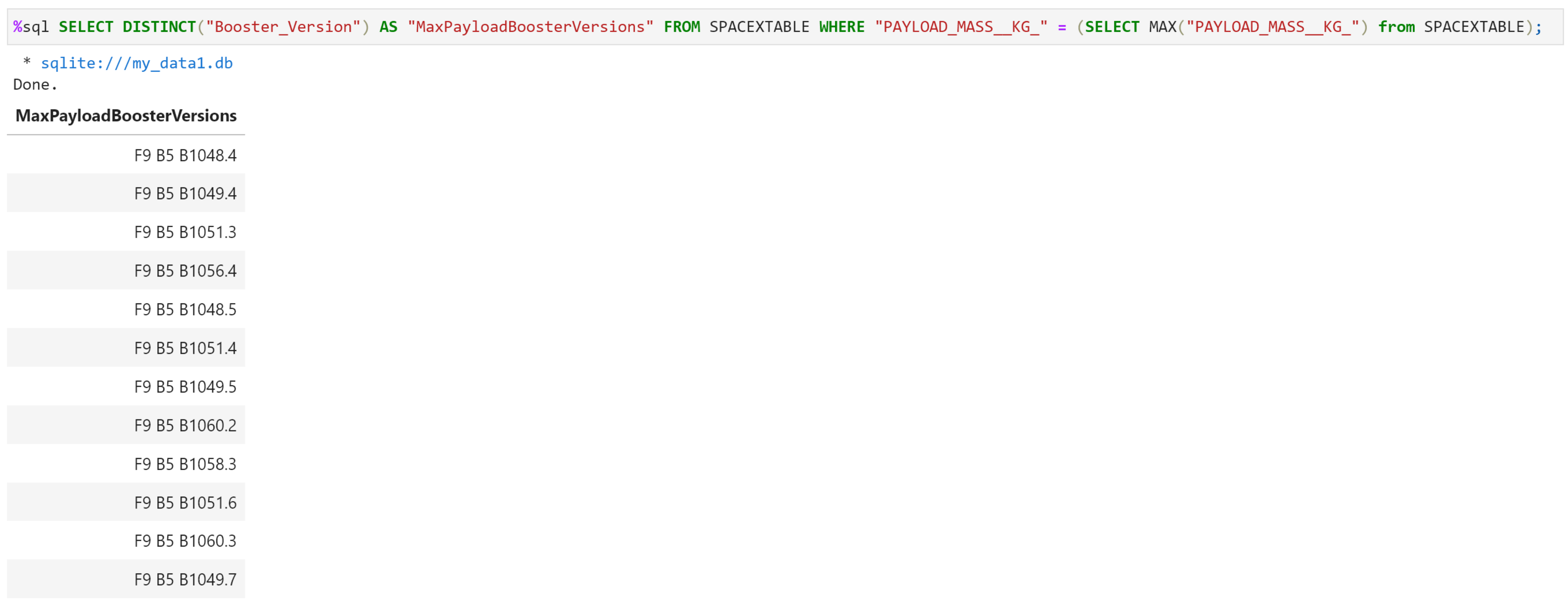
```
%sql SELECT * FROM SPACEXTABLE WHERE "Mission_Outcome" LIKE "%Success%" OR "Mission_Outcome" LIKE "%Failure%";
```

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

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

- The SQL lab doesn’t ask for the total count, so this code was added explicitly for this slide. The total count is 101 missions. An excerpt of the returned missions (the original code from the lab) is also shown.

Boosters Carried Maximum Payload



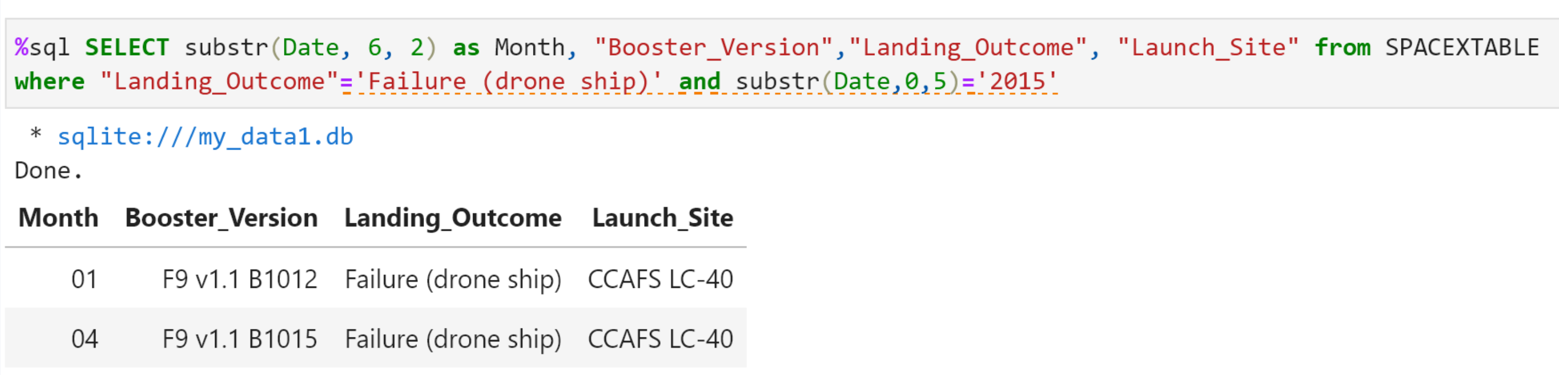
```
%sql SELECT DISTINCT("Booster_Version") AS "MaxPayloadBoosterVersions" FROM SPACEXTABLE WHERE "PAYLOAD_MASS__KG_" = (SELECT MAX("PAYLOAD_MASS__KG_") from SPACEXTABLE);
```

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

MaxPayloadBoosterVersions
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

- A subquery is used to first select the maximum value of the payload mass column, and then to return all booster rockets that have at least one launch with that weight.

2015 Launch Records



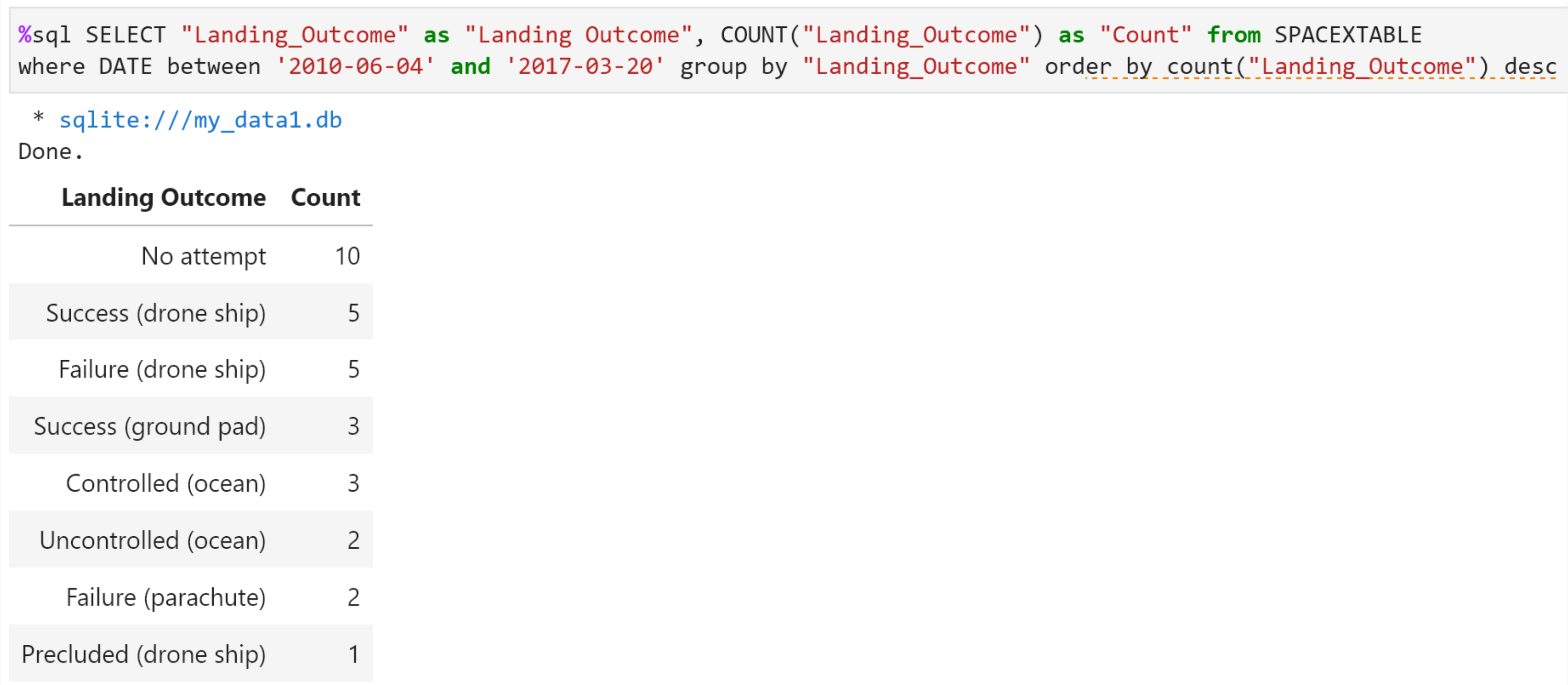
```
%sql SELECT substr(Date, 6, 2) as Month, "Booster_Version","Landing_Outcome", "Launch_Site" from SPACEXTABLE
where "Landing_Outcome"='Failure (drone ship)' and substr(Date,0,5)='2015'
```

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

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

- A substring is used for this query because SQLLite doesn’t support ‘monthnames’. There were two failed landings on the drone ship in the year 2015.

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



```
%sql SELECT "Landing_Outcome" as "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("Landing_Outcome") desc
```

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

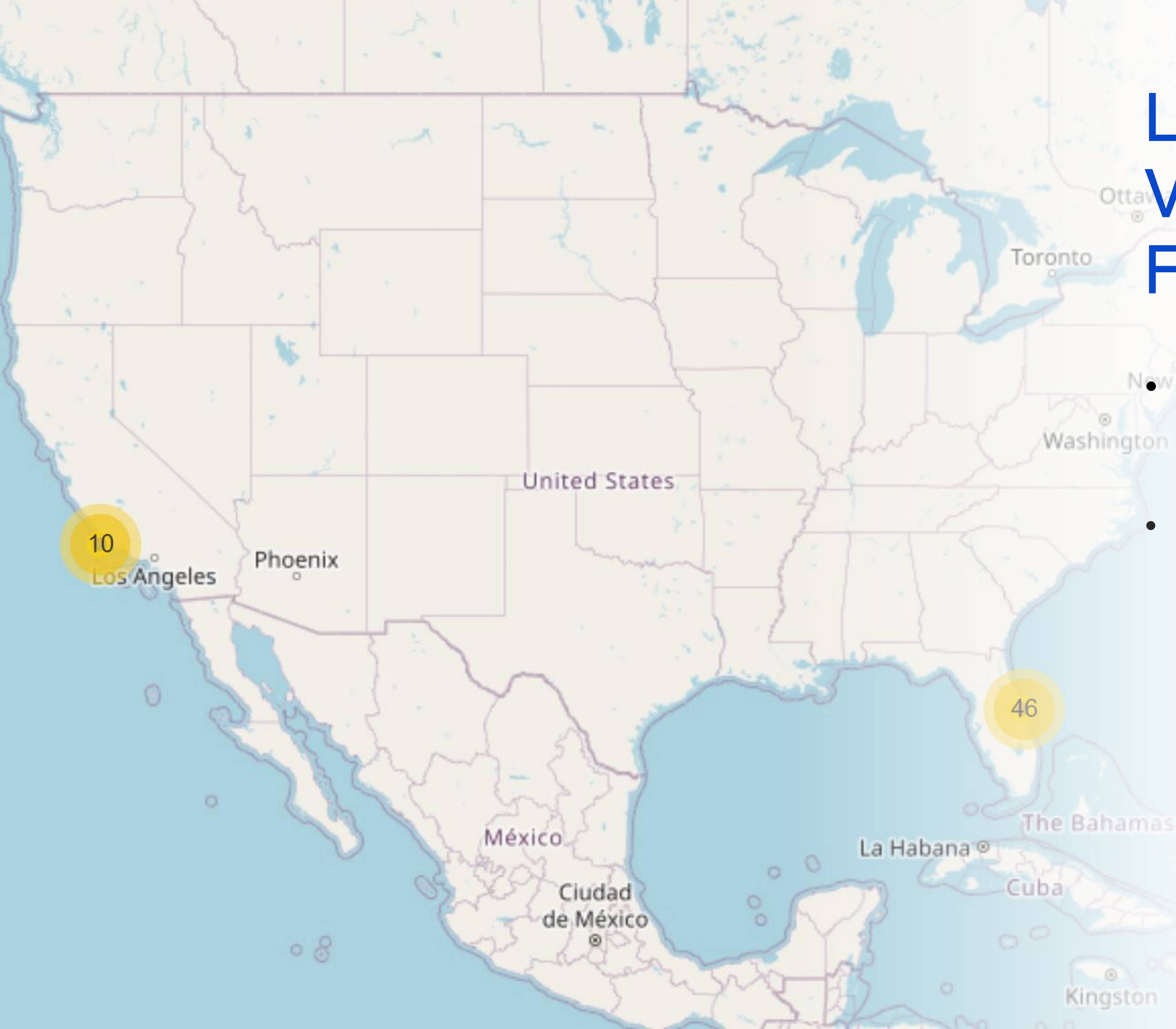
Landing Outcome	Count
No attempt	10
Success (drone ship)	5
Failure (drone ship)	5
Success (ground pad)	3
Controlled (ocean)	3
Uncontrolled (ocean)	2
Failure (parachute)	2
Precluded (drone ship)	1

- All landing outcomes from 4/6/10 through 3/20/17 are counted and then listed in descending order of frequency. The most frequent outcome was no attempt. This highlights an important issue that the different landing outcomes needed to be condensed down into a new binary variable for easier analysis, this was done in a later lab.

Section 3

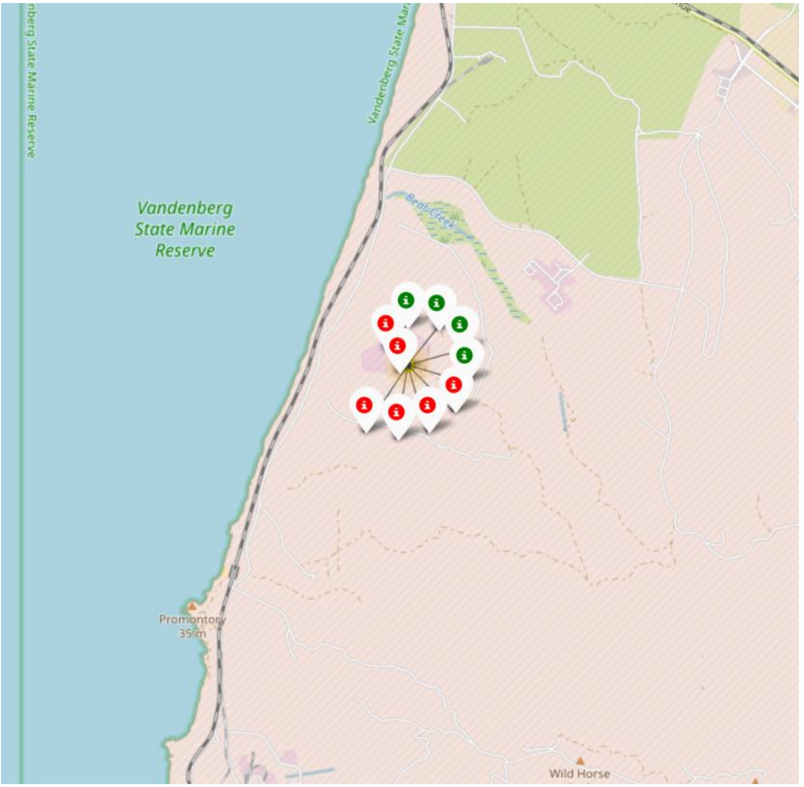
Launch Sites Proximities Analysis

Launch Site Visualization with Folium: Overview

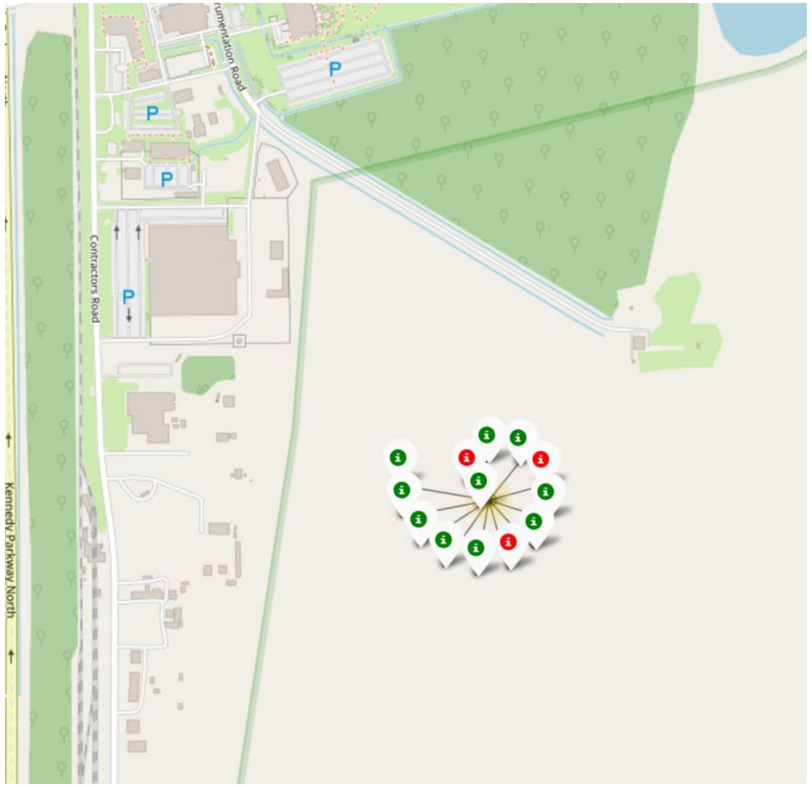


- Visually mapping out launch site locations provides additional insights to our dataset.
- SpaceX launch sites are located no further than 10km away from the nearest oceans. No launch site is further North than -120.6 longitude. Launch sites are located either within wildlife reserves or non-private areas and are at least 10km away from the nearest city or residential area.

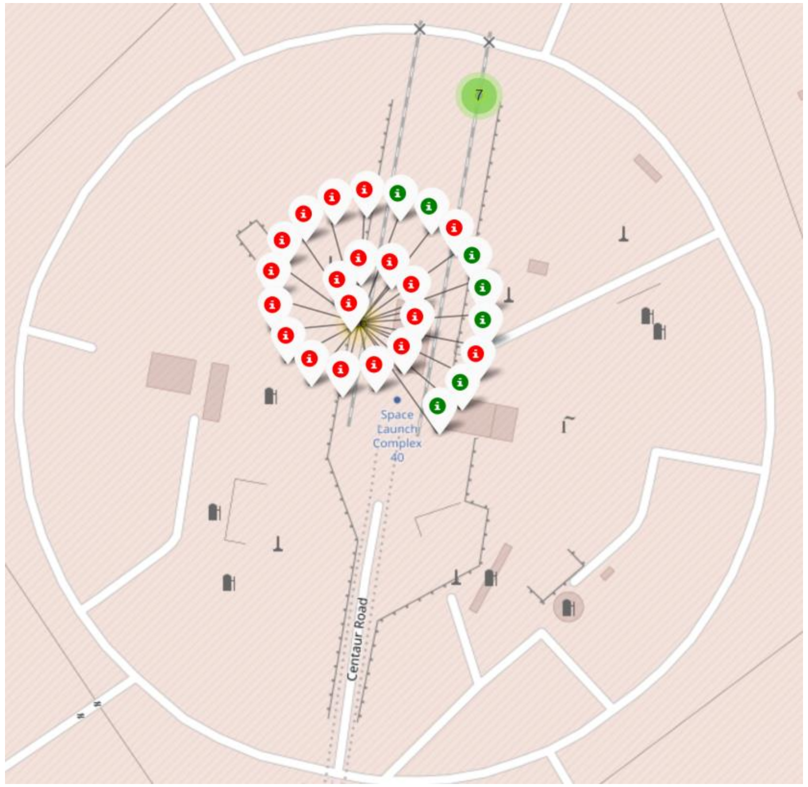
Outcomes by Launch Site


Vandenberg State Marine Reserve
Vandenberg State Marine Reserve
Bear Creek
Promontory
35 m
Wild Horse

VAFB SLC-4E - California

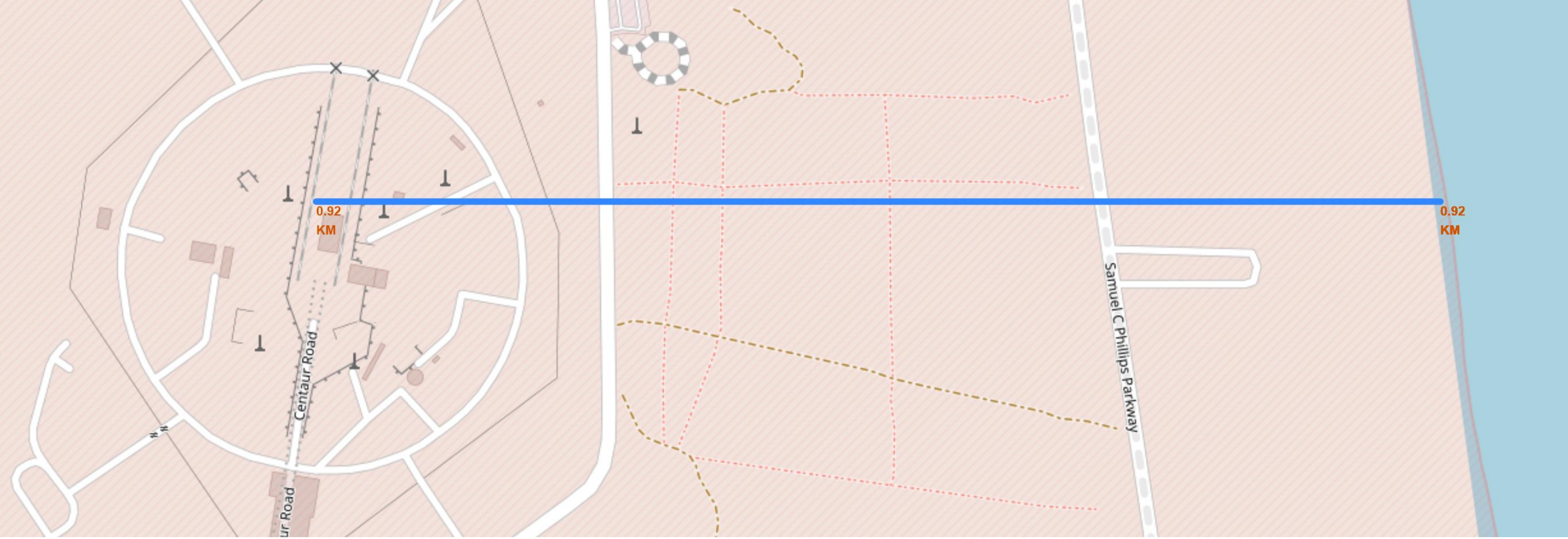


KSC LC-39A - Florida



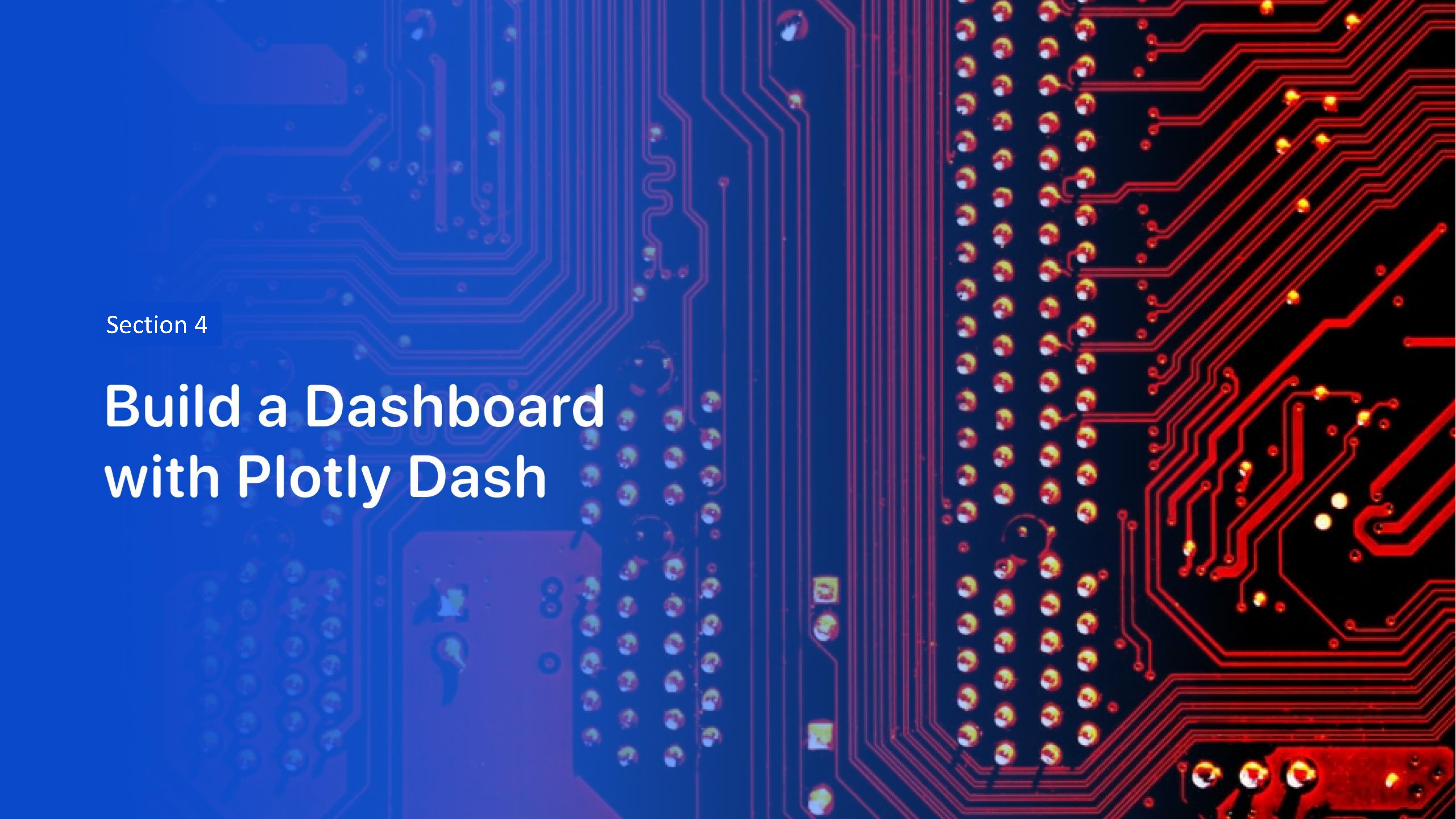
CCAFS LC-40 - Florida

- Each launch is color-coordinated to the landing outcome, green for success and red for failure. CCAFS LC-40 in Florida had the highest count of failed launches as a count (19) and percentage (73.1%) of total launches at that site.



Point of Interest Proximity

- Distance to nearby points of interest are plotted and measured. Shown above, launch site CCAFS LC-40 is approximately 0.92 km away from the Atlantic Ocean. Other proximities such as highways, rail lines, and residential zones are examined for potential correlation with launch success.

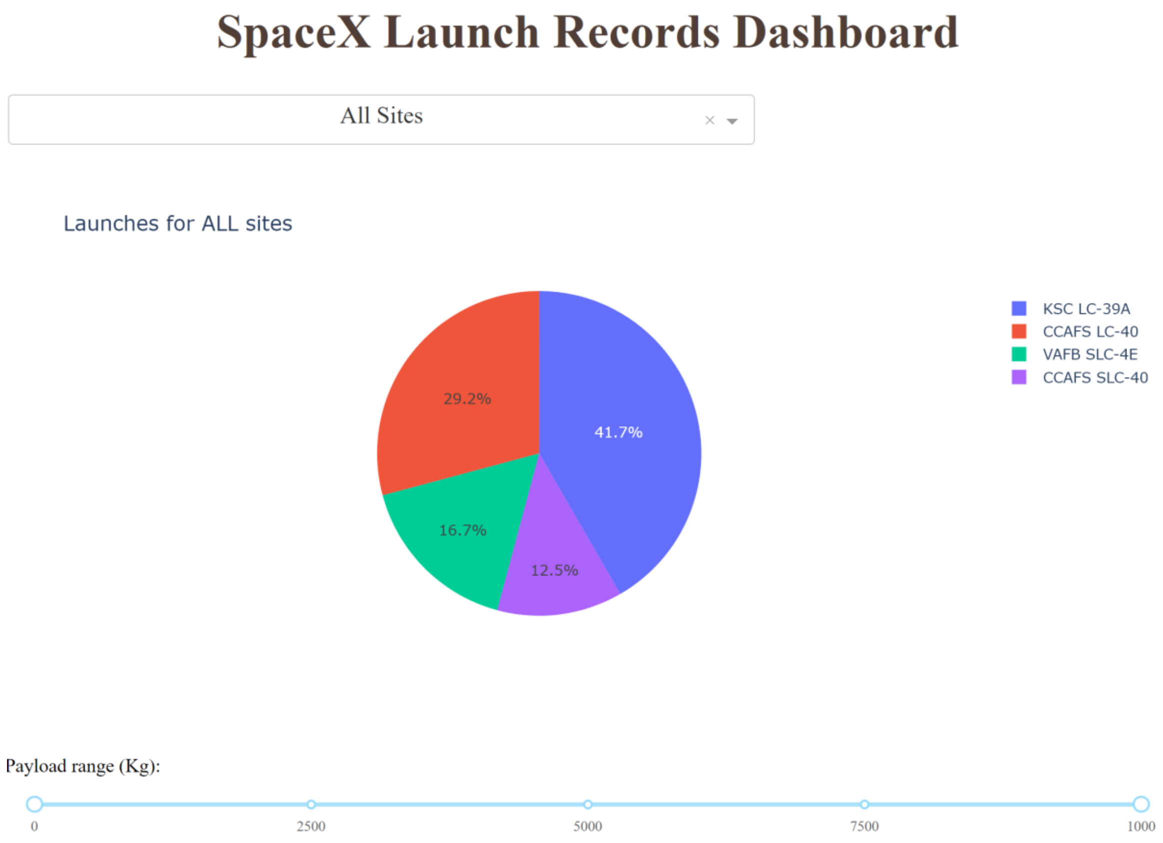


Section 4

Build a Dashboard with Plotly Dash

Plotly Dashboard: All Launch Sites

- Python library Plotly is used to generate a dashboard capable of filtering data by launch site and payload mass.
- A filtered view of all sites showing only successful launches reveals that KSC LC-39A had the greatest share of successful launches at 41.7%, while CCAFS SLC-40 had the lowest share at 12.5%.



SpaceX Launch Records Dashboard

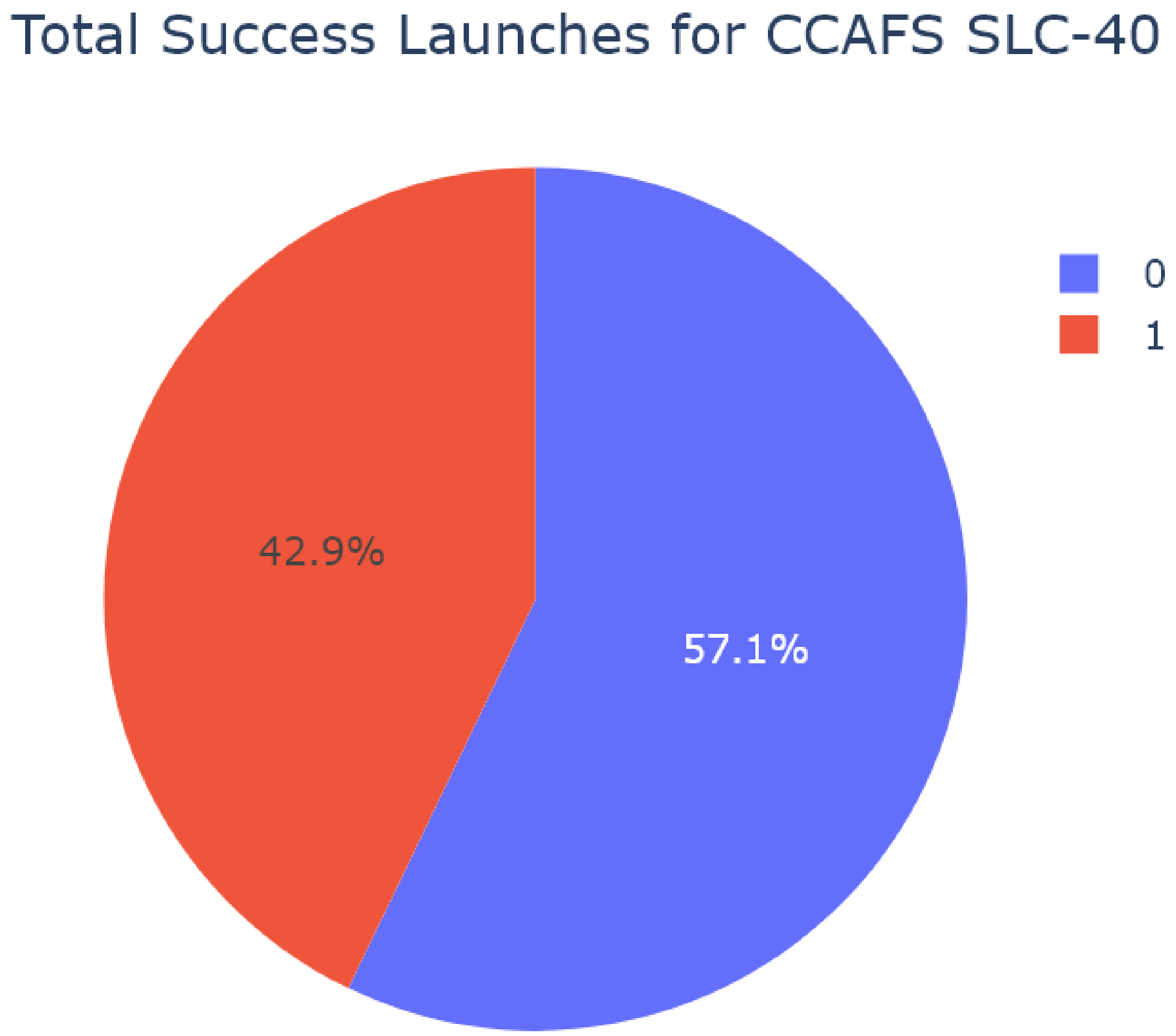
All Sites

Launches for ALL sites

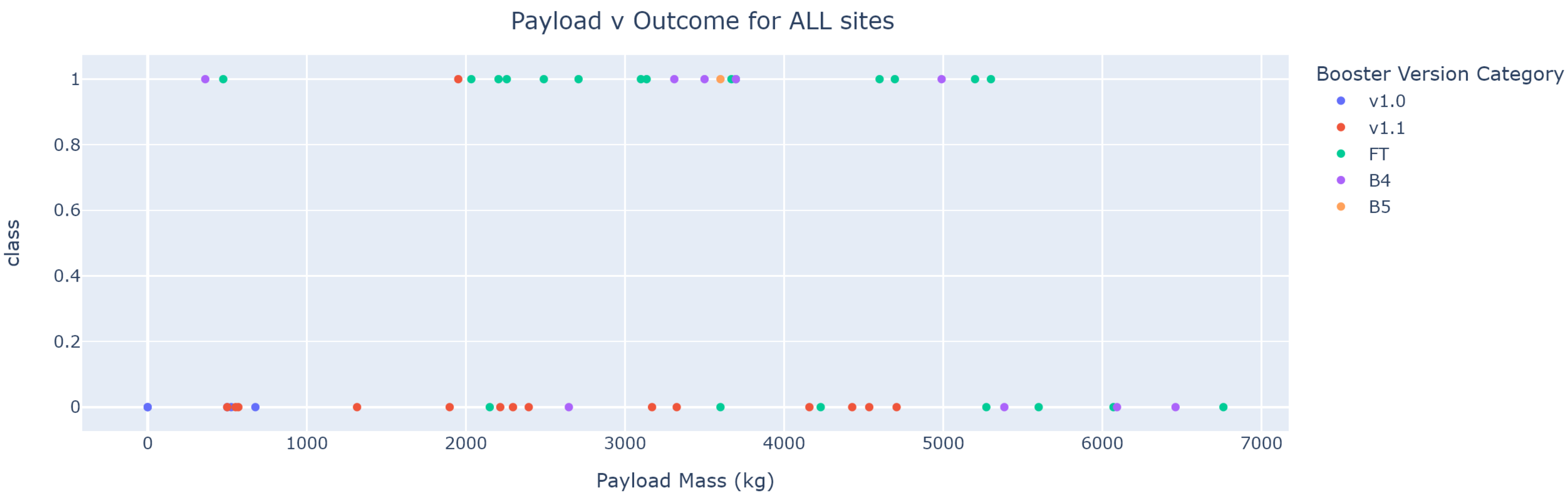
Payload range (Kg):

Plotly Dashboard: Most Successful Launch Site

- Site CCAFS SLC-40 was the most successful launch site, with 42.9% of launches ending in successful landings.
- This is an eye-opening figure, as even the best potential outcome for SpaceX flights is less than a coin-flip's odds of success. This leaves substantial opportunity for SpaceY to make competitive bids in anticipation of SpaceX's relatively high failure rates.



Plotly Dashboard: Outcome Filtered by Payload Outliers



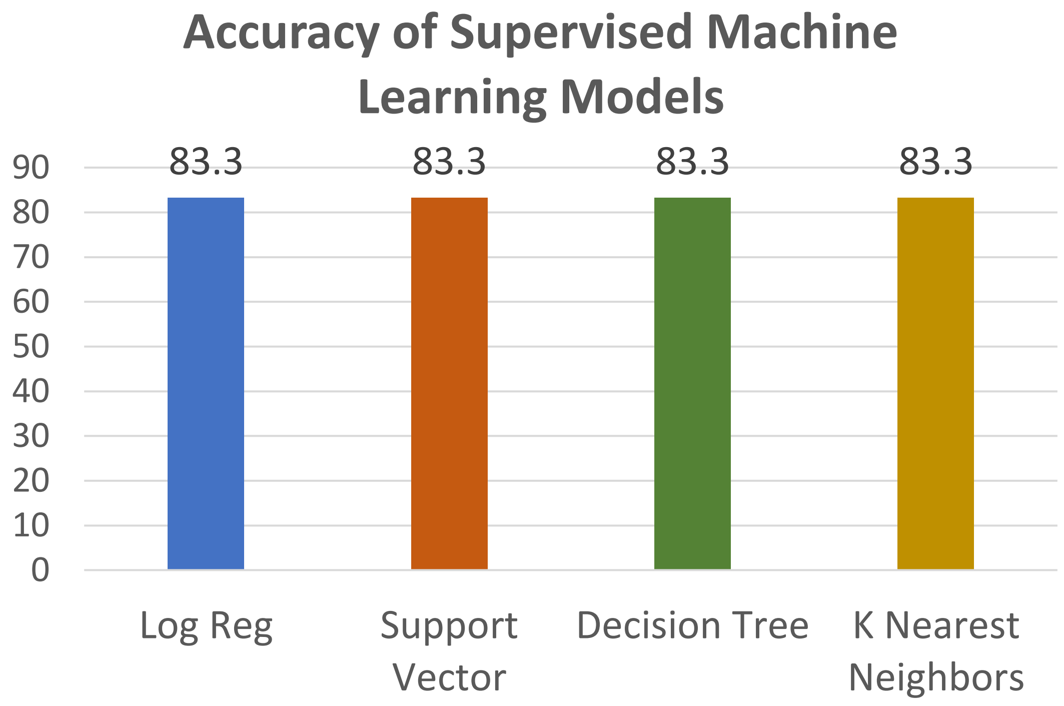
- A scatter plot of all launch sites with payloads of 8,000 kg or less to remove potential outliers. Recall our observation in earlier analysis that extremely heavy payloads are significantly more successful. Booster v1.1 has a higher rate of failure than other booster models.

Section 5

Predictive Analysis (Classification)

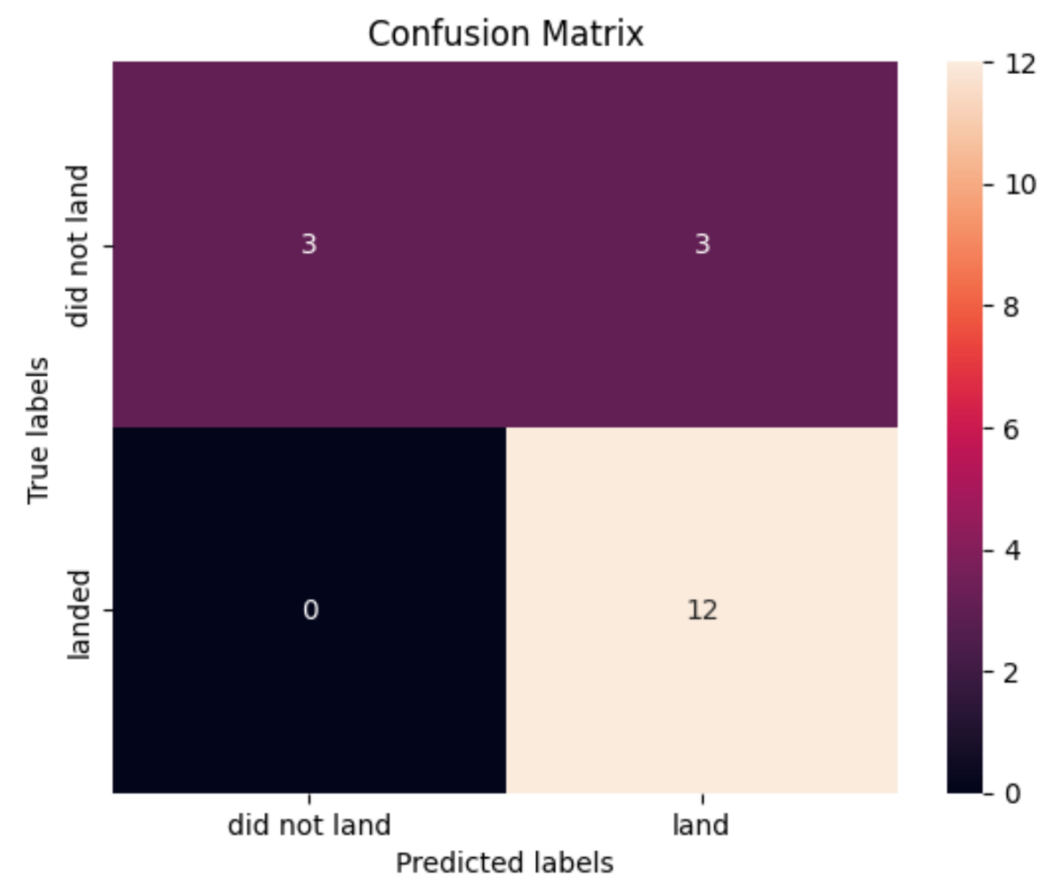
Classification Accuracy

- The four supervised learning models were run on the same test and training data sets.
- All four models scored identical results of 83.3% correctly predicted launch outcomes, with this figure supported by identical confusion matrix outputs.
- The identical accuracy results is due to the limited size of the dataset. As mentioned previously, it is worthwhile to re-examine model accuracy when more data is collected.
- For now we recommend using logistic regression for fast computational speeds and ease of interpreting coefficients of regression.



Confusion Matrix

- As stated previously, all four models had identical confusion matrix results. This screenshot is from the decision tree model but applicable to all models tested.
- 18 launches were included in the test data. 12 of which landed successfully, 6 were unsuccessful.
- The confusion matrix shows that our models correctly predicted the true successful landings but has some difficulty with launches that failed. Three predictions for success were actually true failures (top right quadrant) while three predictions were correctly made for true failure.
- Since we're trying to predict landing failures it would be considered a Type-II error, a false negative, for the three cases we predicted a success but the real outcome was a failure.



Conclusions and Final Thoughts

The results of this study can be summarized as the following points.

- SpaceX has a high failure rate for booster landings with only a 66.6% success rate. So there is potential to win contracts by under-bidding SpaceX where we believe they are likely to fail (and they would thus make a high bid to cover their operating costs.)
- Our predictive model, multi-variable logistic regression, is reasonably accurate. It was able to predict 83.3% of launch outcomes in our testing set. The variables used are FlightNumber', 'PayloadMass', 'Orbit', 'LaunchSite', 'Flights', 'GridFins', 'Reused', 'Legs', 'LandingPad', 'Block', 'ReusedCount', and 'Serial'.
- The predictive model is not given any further robustness tests nor a detailed breakdown of the model's coefficients, p-values, etc. These are not provided because it is beyond the scope of this course. A real-world study would go into substantially more detail discussing the variables of interest, what logic or theoretical backing makes us believe the variables are valid predictors, robustness tests to check for co-linearity, normal distribution, etc. and a thorough walkthrough of the model's output and implications. For example, this project doesn't determine which variable has the strongest causal effect on landing success and what that value is.
- The GitHub repository for all lab files is available here.

Thank you!