IBM Developer
SKILLS NETWORK

Winning the Modern Space Race with Data Science

12/28/23

Patrick Murray

Python, SQL, Visualizations, ML

Outline

- Introduction
- Abstract
- Section 1 - Methodology
  - Data collection
  - Data wrangling
  - Exploratory data analysis
  - Interactive visual analytics
- Section 2 – Predictive Analysis
- Section 3 – Conclusion

(Appendix)
- Section 4 – Insights drawn from EDA
- Section 5 – Launch Sites Proximities Analysis
- Section 6 - Build a Dashboard with Plotly Dash

Introduction

This project explores the possibility of predicting the success or failure of a theoretical SpaceX Falcon 9 rocket launch. To achieve this, a wide breadth of technical skills are employed including Python, SQL, data visualization, multi-variate regression, machine learning for predictive analysis, and reporting of findings.

Several caveats are made with regards to statistical and model robustness for the purpose of completing this project. The goal isn't a deep dive into sound hypothesis testing, but to demonstrate a variety of techniques used in data analytics and data science. For example, the data is not randomly sampled nor believed to follow a normal distribution.

Links to code samples and technical printouts are provided where relevant. Sections 1-3 of this project comprise the analytical work while sections 4-6 contain code excerpts and greater detail on select analysis.

Abstract

The commercial space sector is a highly lucrative market with contract bids valued in the hundreds of millions. SpaceX holds a unique advantage in that it can re-use its first-stage Falcon 9 rocket boosters, significantly reducing operating costs and thus lowering their contract bids.

The objective of this study is to see if the success or failure of a SpaceX landing can be predicted. Launches that are likely to fail will cost SpaceX approximately $165 million while successes cost only $62 million. Predicting the outcome of a SpaceX launch allows a theoretical firm to offer competitive bids and thereby win contracts.

This study utilizes SpaceX launch data that is collected with the SpaceX REST API as well as conventional web-scraping. This data includes 101 observations from 2010 through 2020. The data is cleaned and normalized before machine-learning based statistical analysis is performed to predict future launch success. All coding is done in Python in a Jupyter Notebook.

Preliminary analysis shows that the launch site plays a role in success, followed by orbit type and payload weight. A trend is observed where success has increased over time, meaning later flights are more likely to be successful and must be controlled for in the model.

The predictive model therefore incorporates the following variables of interest – flight number, payload mass, orbit, and launch site. Test set data is run through four different supervised learning models. All models – logistic regression, support vector machine, decision tree, and k-nearest neighbors produce nearly identical goodness of fit results, approximately 83% accuracy.

Section 1

Methodology

Data

- Publicly available data of 101 SpaceX launches from 2010 through 2020 is collected using both the SpaceX REST API as well as web scraping using Python package BeautifulSoup. Data is sourced from SpaceX's data library at spacexdata.com as well as Wikipedia's Falcon 9 page. The raw data is converted into Pandas data frames for ease of further wrangling and filtering.
- The variables include date, time of launch, booster version, launch site, payload type, payload mass, orbit type, customer, mission outcome, landing outcome.
- Wikipedia entry for Falcon 9 Heavy Rockets

Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Orbit	Customer	Mission_Outcome	Landing_Outcome
7/14/2014	15:15:00	F9 v1.1	CCAFS LC-40	OG2 Mission 1 6 Orbcomm-OG2 satellites	1316	LEO	Orbcomm	Success	Controlled (ocean)

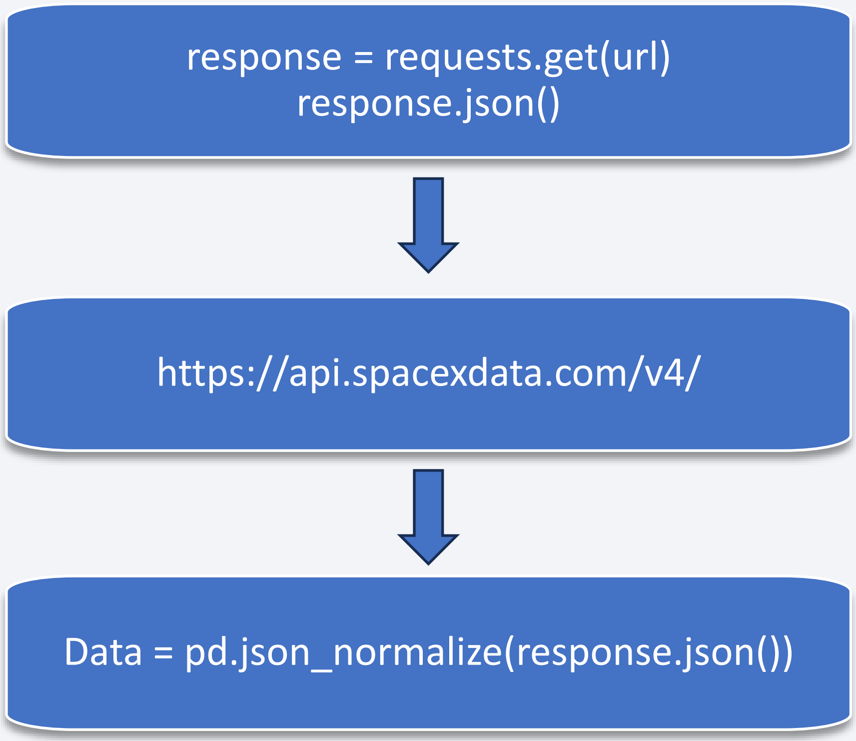
Excerpt from cleaned data showing a single flight entry.

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/.

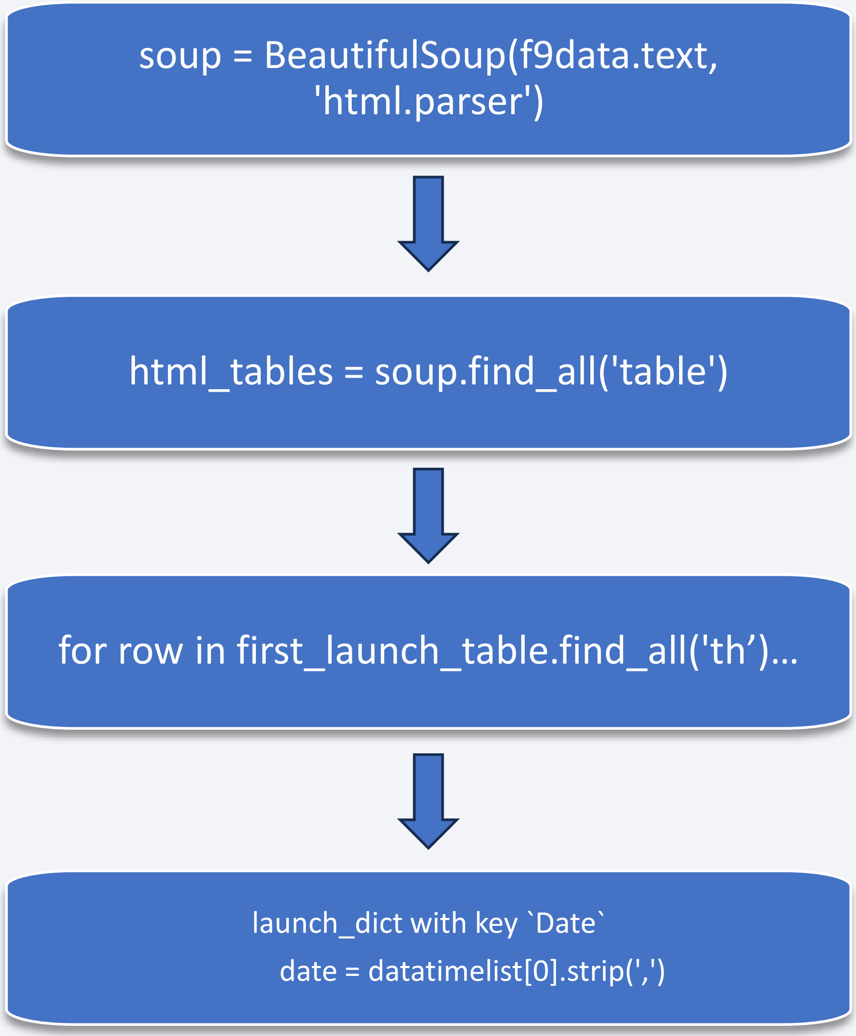
  /rockets/, /launchpads/, /payloads/, /cores/

- 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.



Data Collection – Web 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.



Data Wrangling

Data wrangling helps reveal initial patterns, distributions, data types, and missing values. It helps us understand what the dataset contains and what issues may need to be addressed before performing any statistical analysis. One shortcoming with the raw data is that flight outcome is divided into eight different categorical outcomes. I create a new binary variable “Class” (1 to indicate a true success while 0 indicates failure) and populate the variable.

One finding 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
```

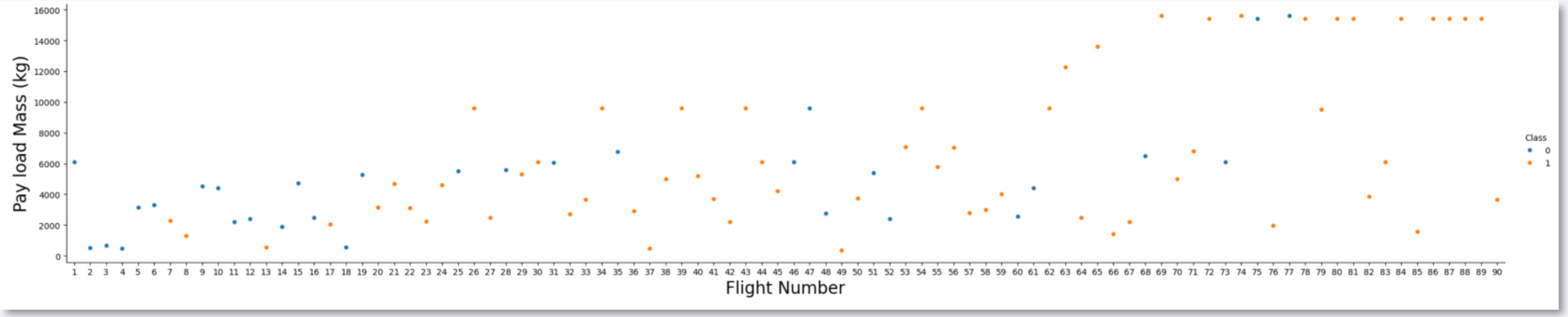
The eight different outcomes are consolidated into a single binary variable, “Class”.

```
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 data visualization using the Python libraries Numpy, Pandas, and Seaborn.
- One relationship I'm interested in is that of payload weight on flight success. A simple scatter chart shows the appearance of a positive correlation with flight number and success as well as between payload mass and failure. These relationships warrant further statistical investigation and suggest potential variables of interest.
- Similar scatter charts are 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 are used for a quick visual inference of potential relationships.
- Finally, a basic a line chart of year vs success rate gave general information about trends occurring independent of potential variables of interest – I note there is an increasing success rate over time.
- GitHub repository of EDA.



Scatter plot of Flight Number vs Payload Mass (KG)

EDA with SQL

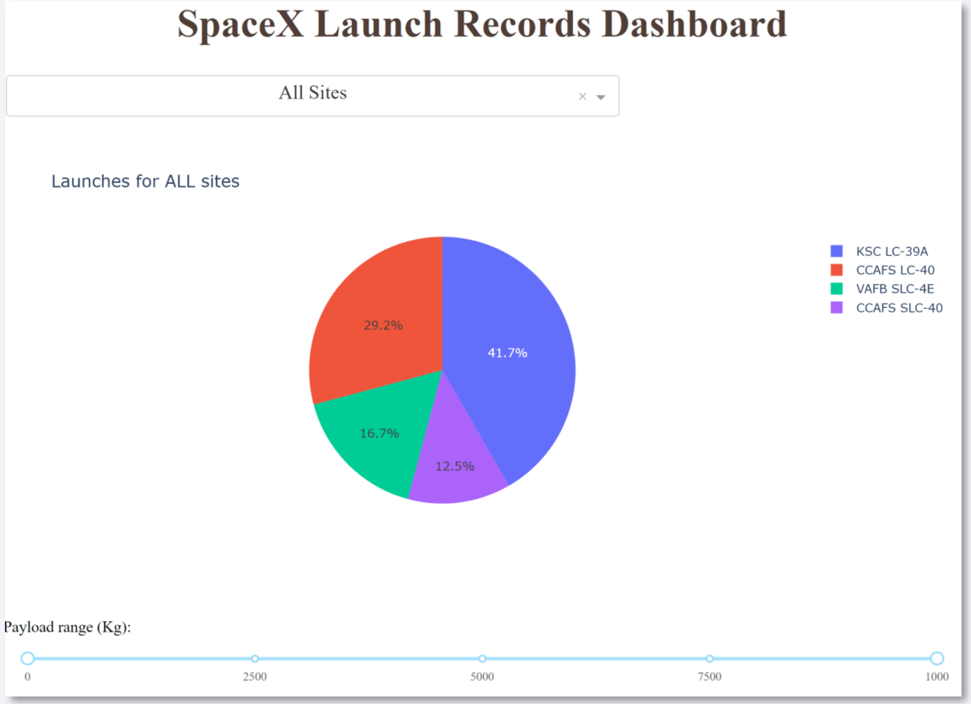
- SQL is used for quick queries on the SpaceX dataset and offer a glimpse at various outcomes and statistics that may be useful for formulating the predictive model.
- First, the raw data is imported and converted from a .csv file to a Pandas data frame. Blank rows and Null entries are removed from the 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 client (ex, NASA), finding the total number of successful and failed missions, etc.
- Other more specified queries such are used for investigating potential outliers or trends.
- GitHub repository of SQL code is available here.

Interactive Map with Folium

- Geographic mapping is another useful tool for identifying other aspects of interest to the predictive model. For example, launch sites may share common characteristics such as proximity to infrastructure, bodies of water, or similar longitudes. This should be controlled for.
- To complete this goal, I 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, and labels for general convenience.
- A few general trends are noted. 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.]()

Dashboard with Plotly Dash

- To further explore the determinants of a successful launch outcome I drill down to greater specifics within the data. Comparisons can easily be made by utilizing a dashboard.
- I've seen possible trends with launch site location affecting success. This dashboard features a dropdown menu that allows the user to select which launch site data to display.
- The dashboard affirms earlier observations. 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.



Section 2

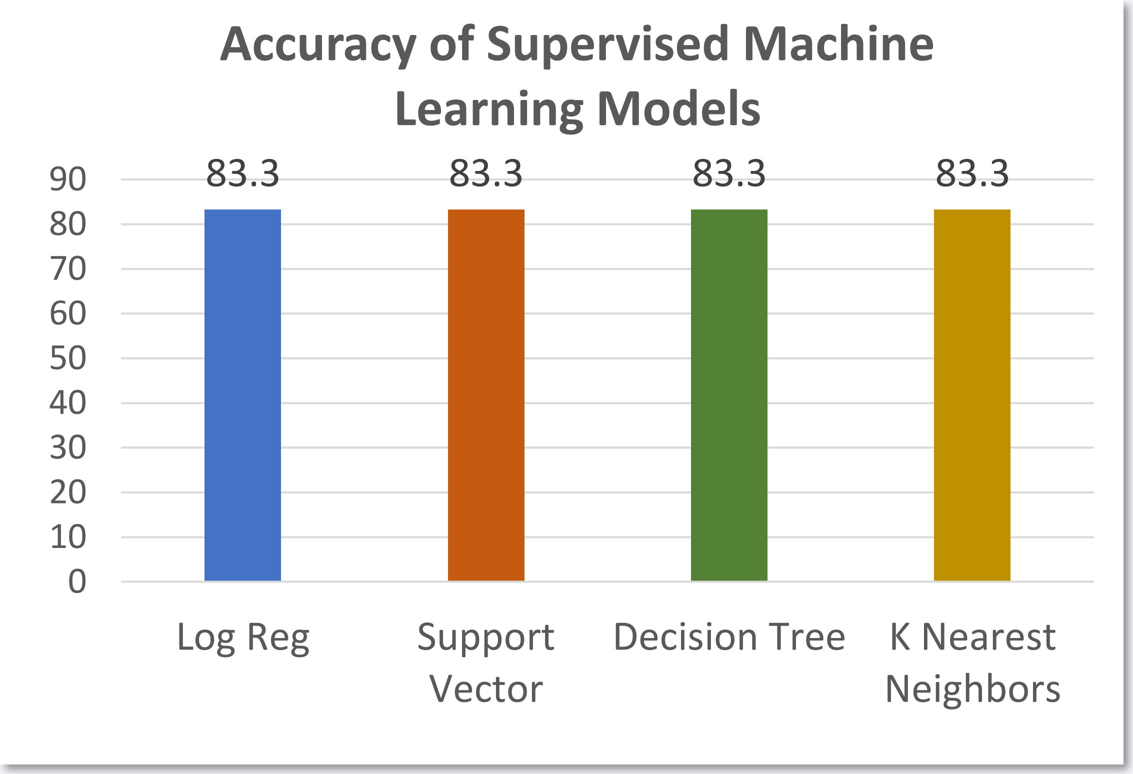
Predictive Analysis (Classification)

Predictive Analysis (Classification) and Results

- The final step in this project is predictive analysis. Specifically, determining what sort of ML classification model is best suited for the problem of predicting SpaceX launch outcome success.
- From earlier analysis I have identified numerous potential determinants for success including payload mass, orbit type, launch site, and booster rocket type. Our model will therefore be multi-variable.
- I 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. The dataset was randomly split into training and testing sets and run on each of the four model types. Goodness of fit scores as well as confusion matrices are calculated and compared to determine the best model.
- I 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.
- Since the performance of the four models is so similar I recommend using logistic regression. It is one of the lest computationally intensive model types, scales well with more data, and is more easily interpretable.
- [GitHub repository is available here.]

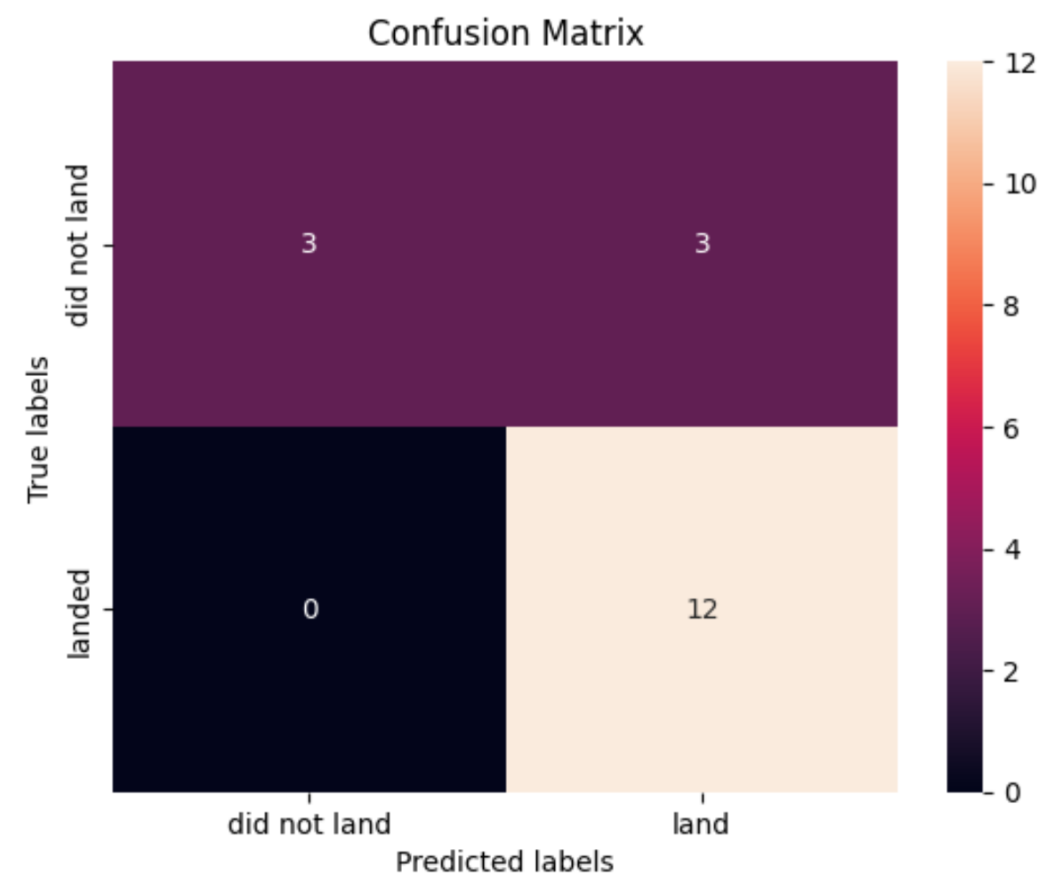
Classification Accuracy

- The four supervised learning models are run on the same test and training data sets.
- All four models scored identical results of 83.3% correctly predicted launch outcomes. This figure is 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.
- I 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 are included in the test data. 12 of which landed successfully, 6 are unsuccessful.
- The confusion matrix shows that the models correctly predicted truely successful landings but has some difficulty with failed landings. Three predictions for success are actually true failures (top right quadrant) while three predictions are correctly made for true failure.
- Falsely predicting a landing success would be considered a Type-I error, a false positive. For three observations the models predicted a success but the real outcome was a failure.



Section 3

Conclusion

Conclusions and Final Thoughts

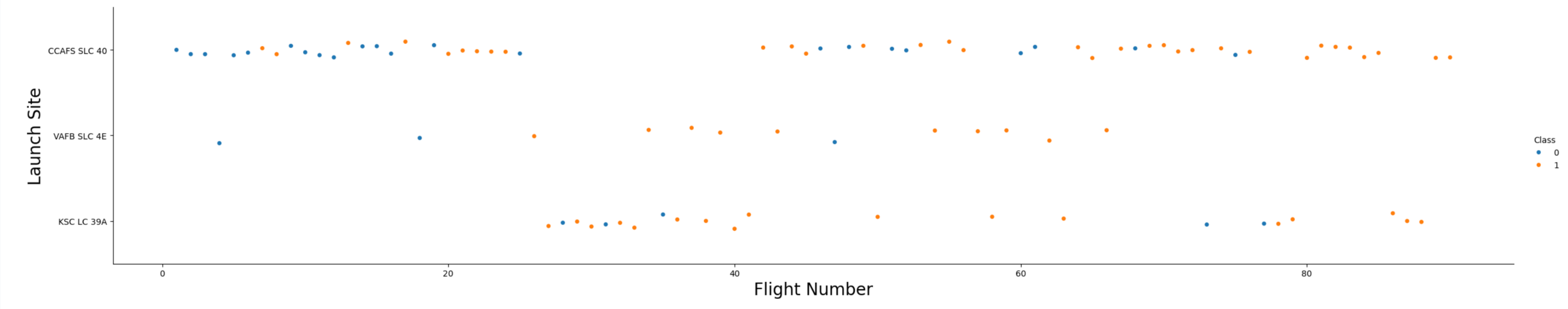
The results of this study can be summarized as the following points.

- 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 when it's believed they are likely to fail.
- The predictive model, multi-variable logistic regression, is reasonably accurate. It was able to correctly predict 83.3% of launch outcomes in the testing data. 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. A real-world study would go into substantially more detail discussing 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.

Section 4

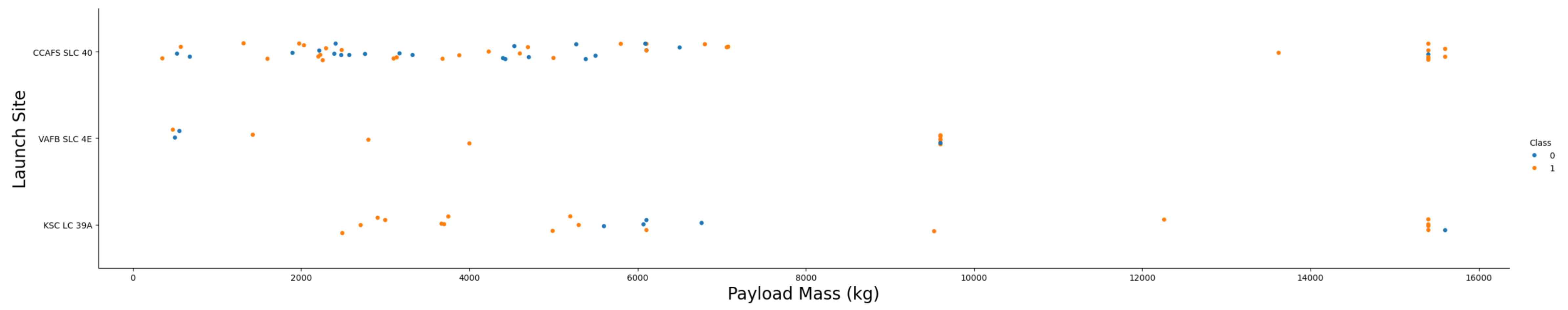
Insights drawn Conclusion from EDA

Flight Number vs. Launch Site



The scatter plot of flight numbers versus launch site is useful to check for trends over time. I 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, are more likely to succeed. Launch site KSC LC 39A had no failures at lower payload weights, below 5,000kg.

Success Rate vs. Orbit Type

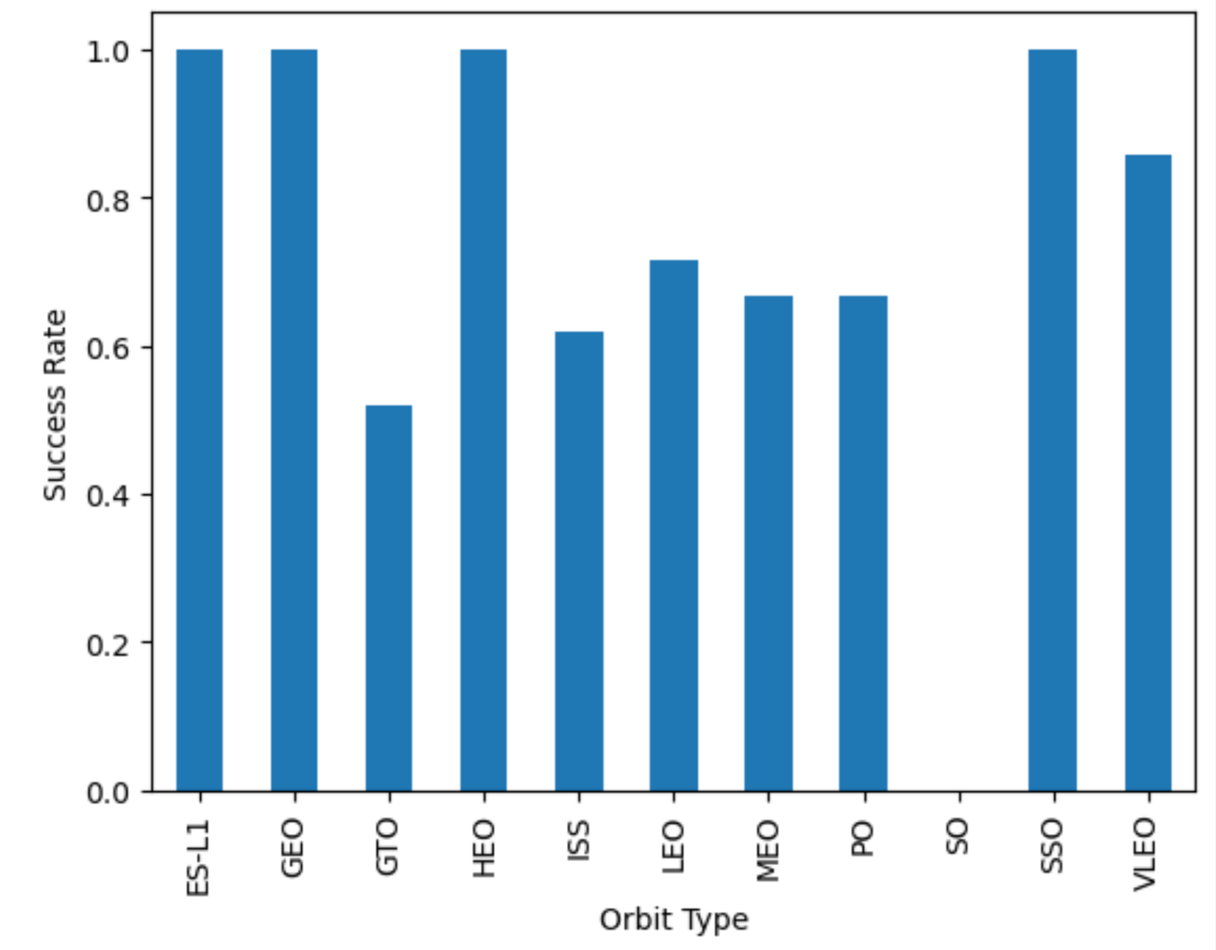
Orbit type appears to play an important factor in outcome success as there is a clear separation in success rates.

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

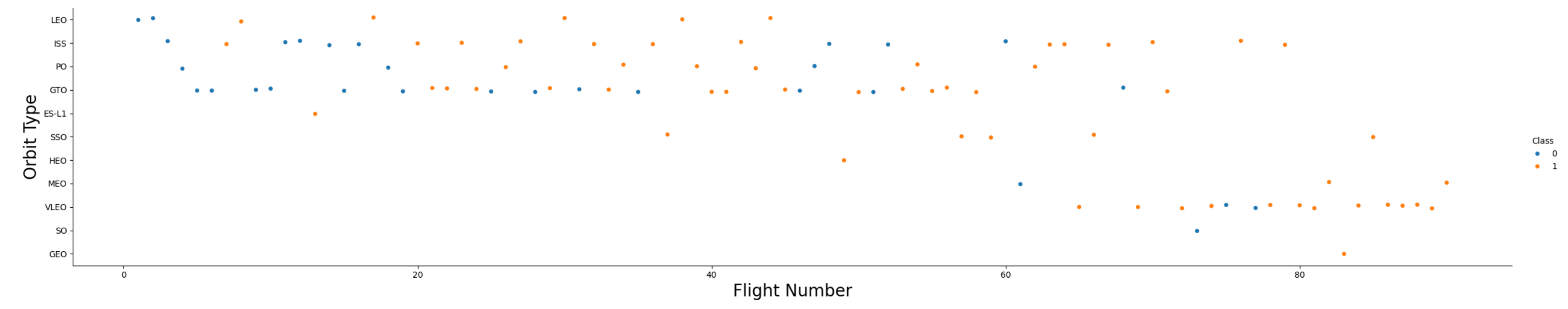
There is no discernable pattern to the orbit type and success holver. GTO is a high altitude orbit (approximately 35,786km) similar to ES-L1 and GEO but with a significantly worse success rate.

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

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



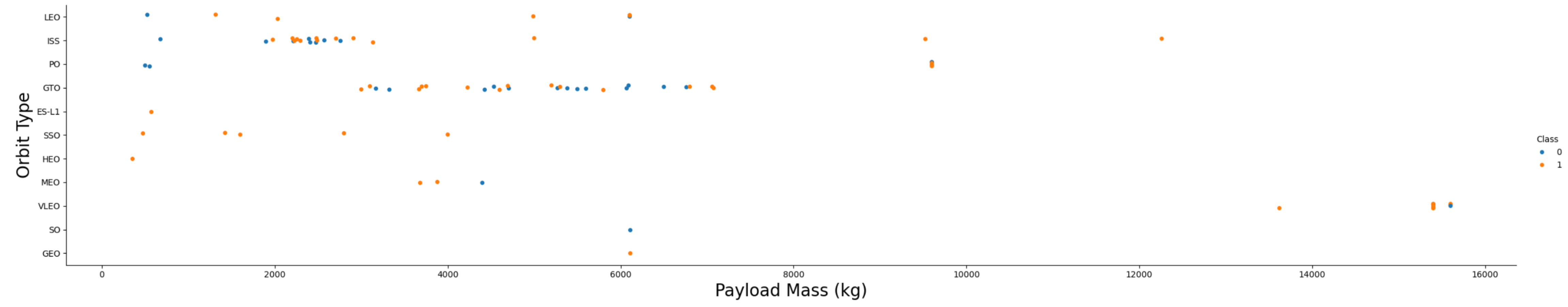
Flight Number vs. Orbit Type



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, are exclusively launched to VLEO orbit. ISS orbit has the largest spread in payloads, ranging from approximately 1,000 kg to 12,000 kg.

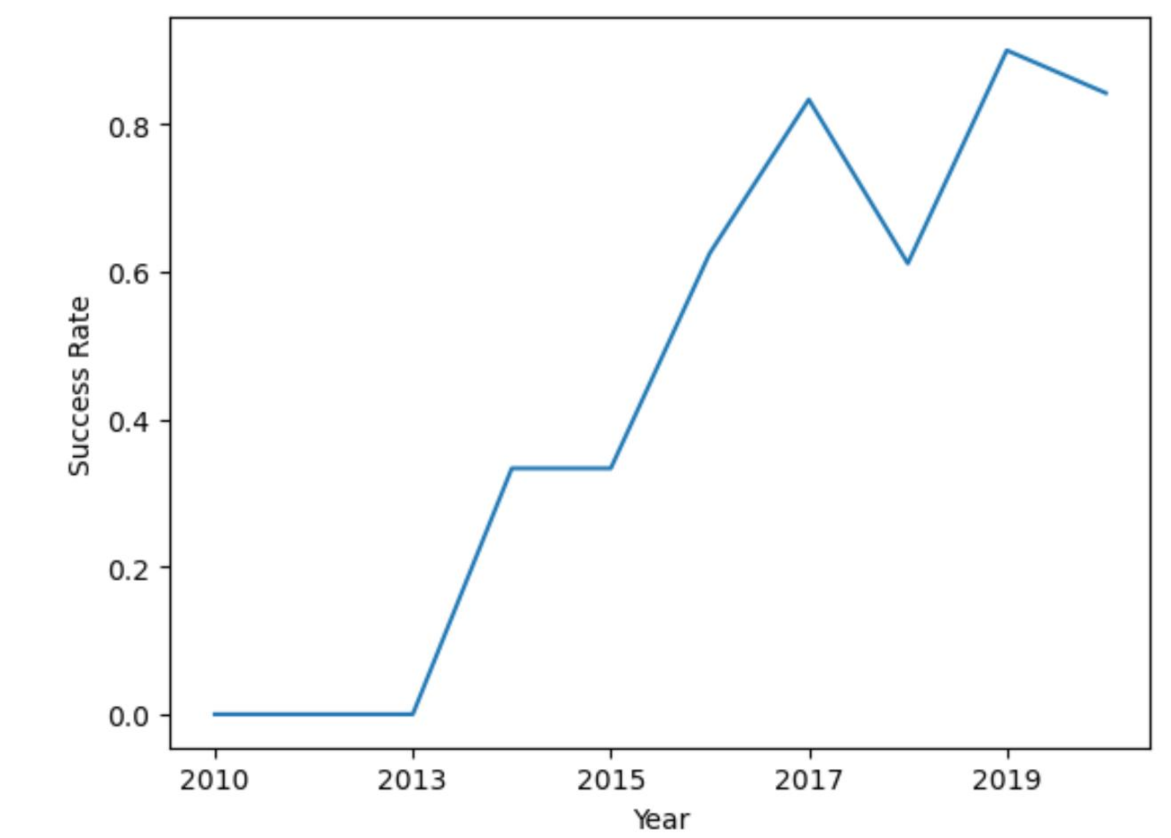
Ultimately there aren't enough data points to draw any solid conclusions about payload and orbit type.

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 I 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 other explanatory variables.

If I 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 the predictive model, so I 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 light 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 total count is 101 missions. An excerpt of the returned missions 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 light.

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 are 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
```

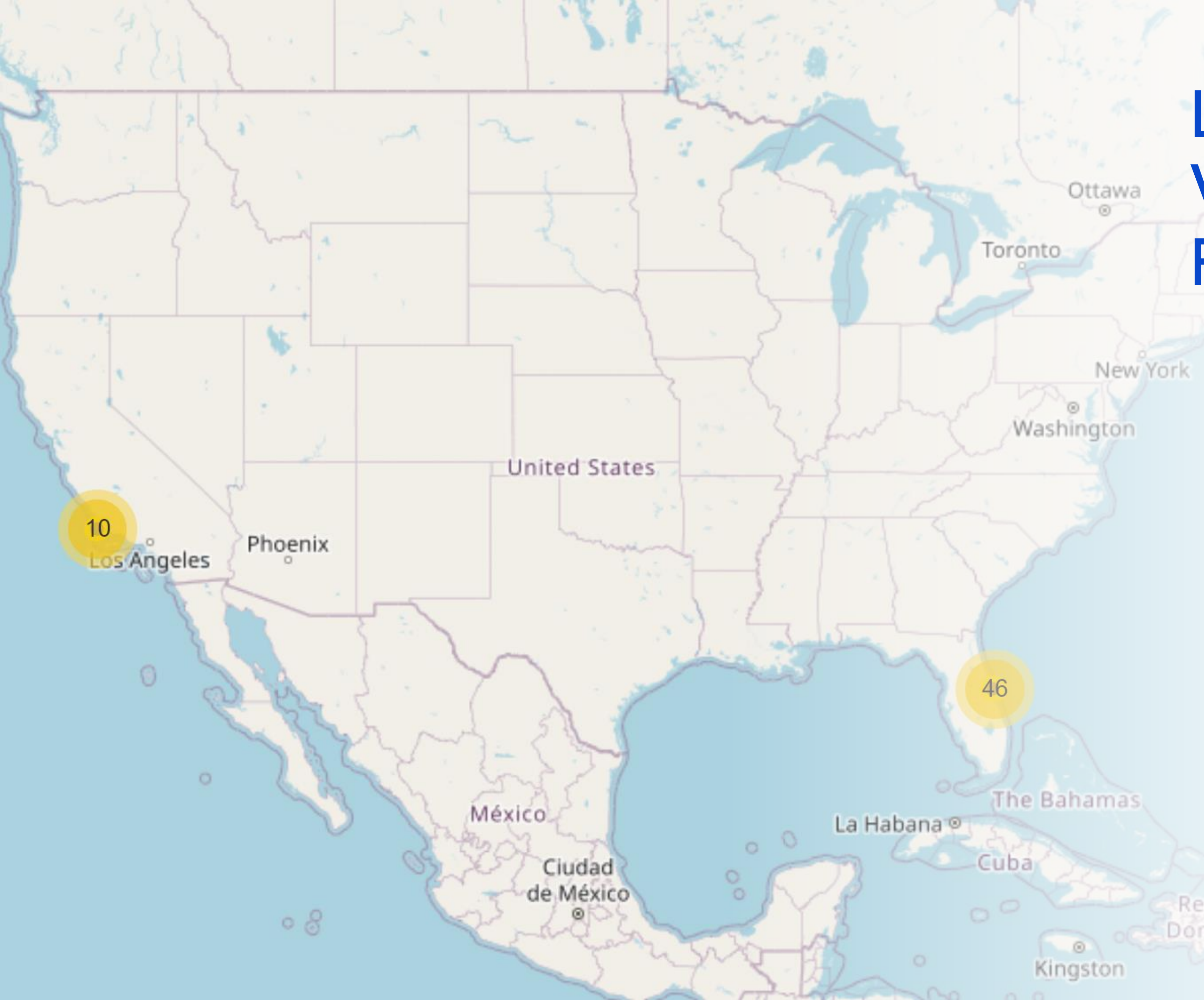
```
 * sqlite:///my_data1.db
Done.
```

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.

Section 5

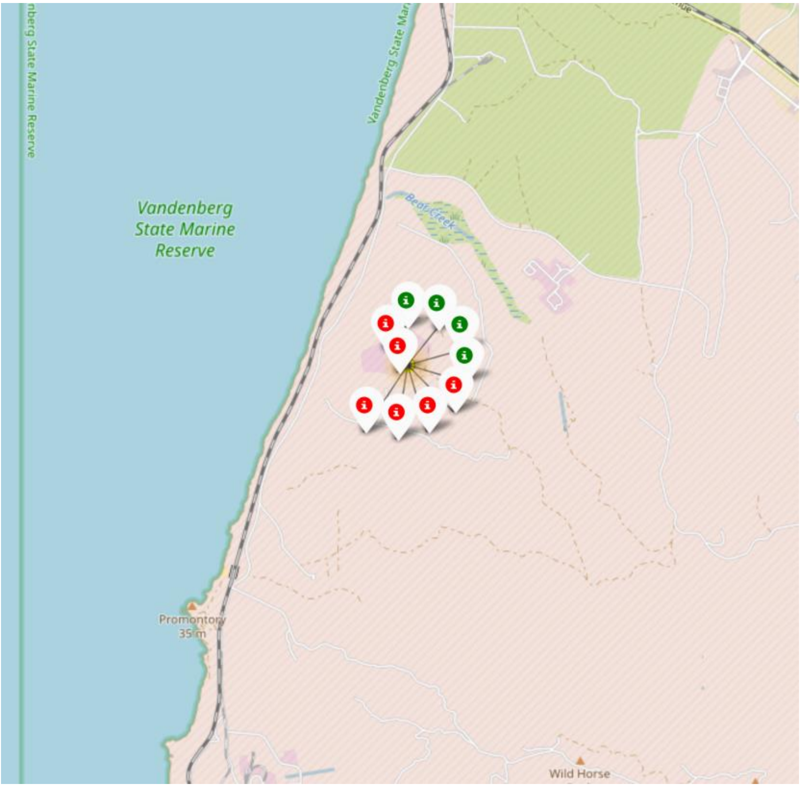
Launch Sites Proximities Analysis



Launch Site Visualization with Folium: Overview

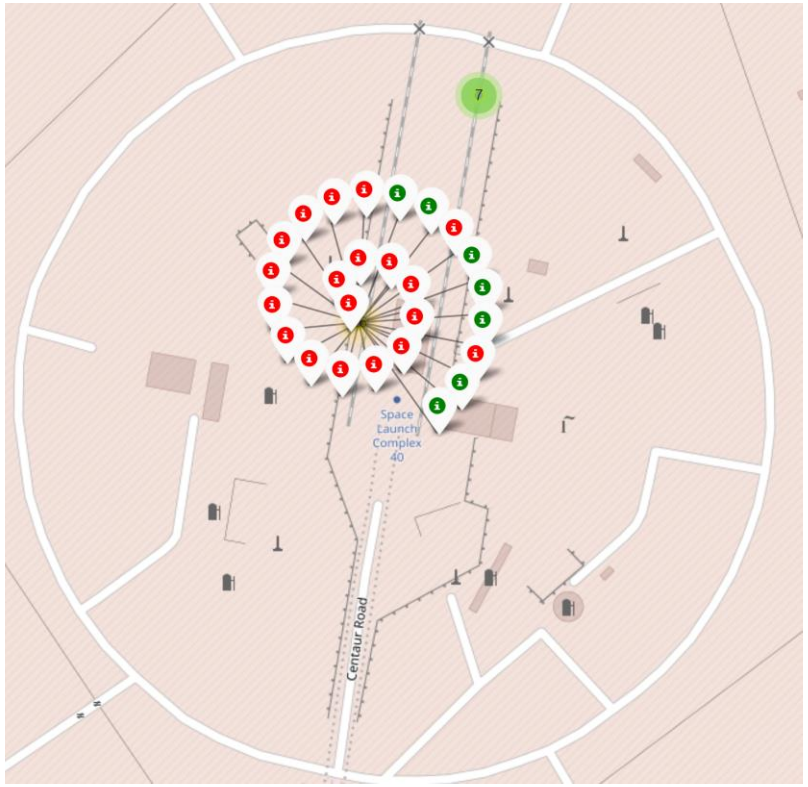
- Visually mapping out launch site locations provides additional insights to the dataset.
- SpaceX launch sites are located no further than 10km away from the nearest oceans. No launch site is further North than -120.6 longitude. Launch sites are located either within wildlife reserves or non-private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

Outcomes by Launch Site



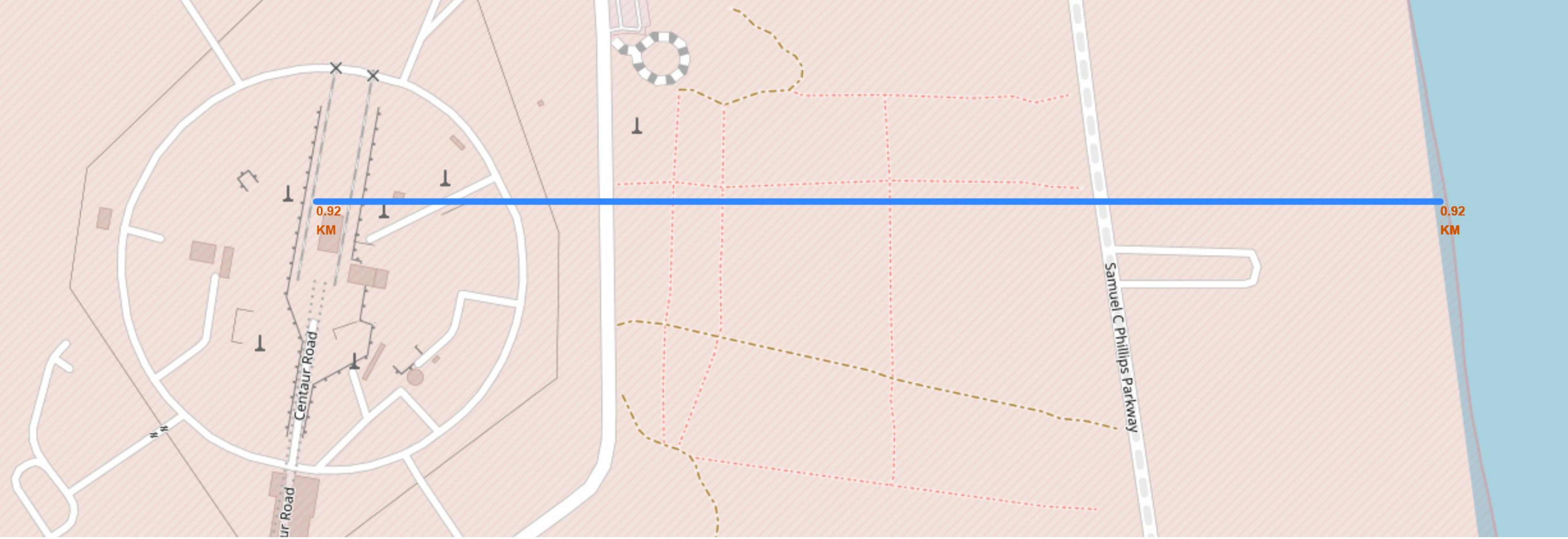
VAFB SLC-4E - California

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 outcomes as a count (19) and percentage (73.1%) of total launches at that site.



Point of Interest Proximity

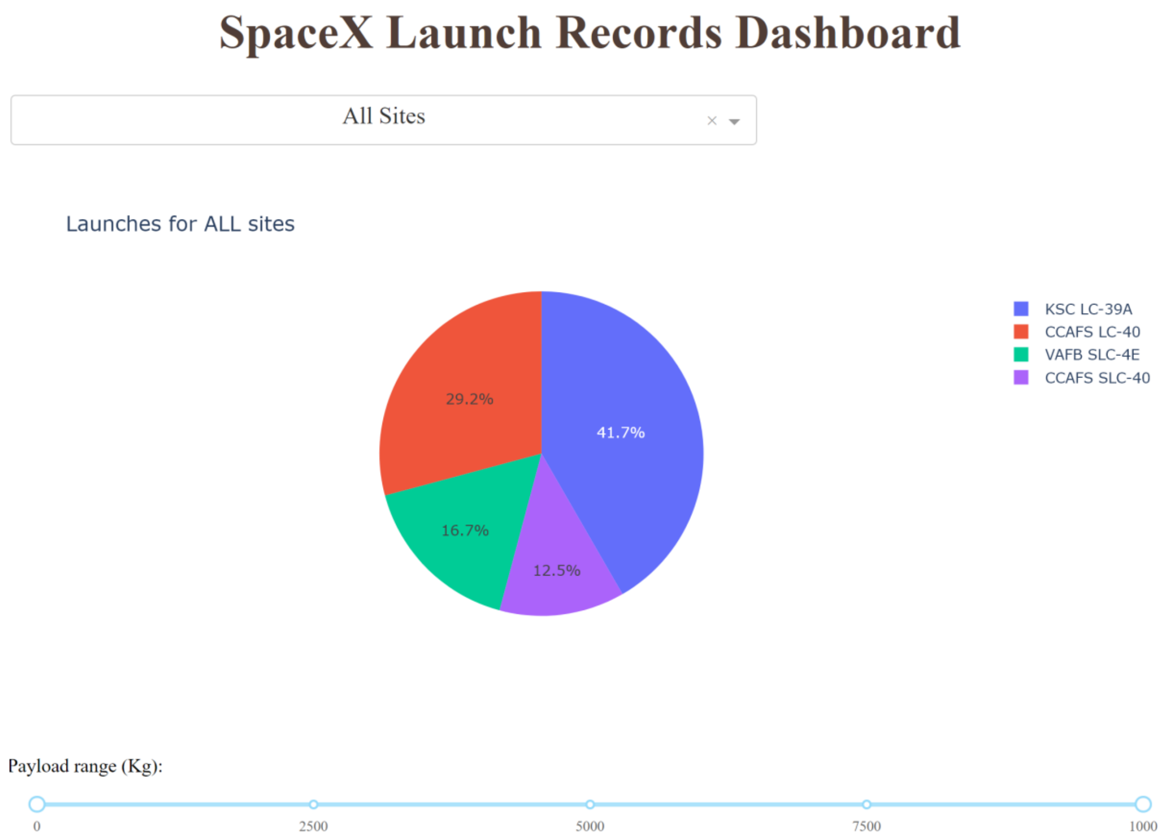
- Distance to nearby points of interest are plotted and measured. Shown above, launch site CCAFS LC-40 is approximately 0.92 km away from the Atlantic Ocean. Other proximities such as highways, rail lines, and residential zones are examined for potential correlation with outcome success.

Section 6

Build a Dashboard with Plotly Dash

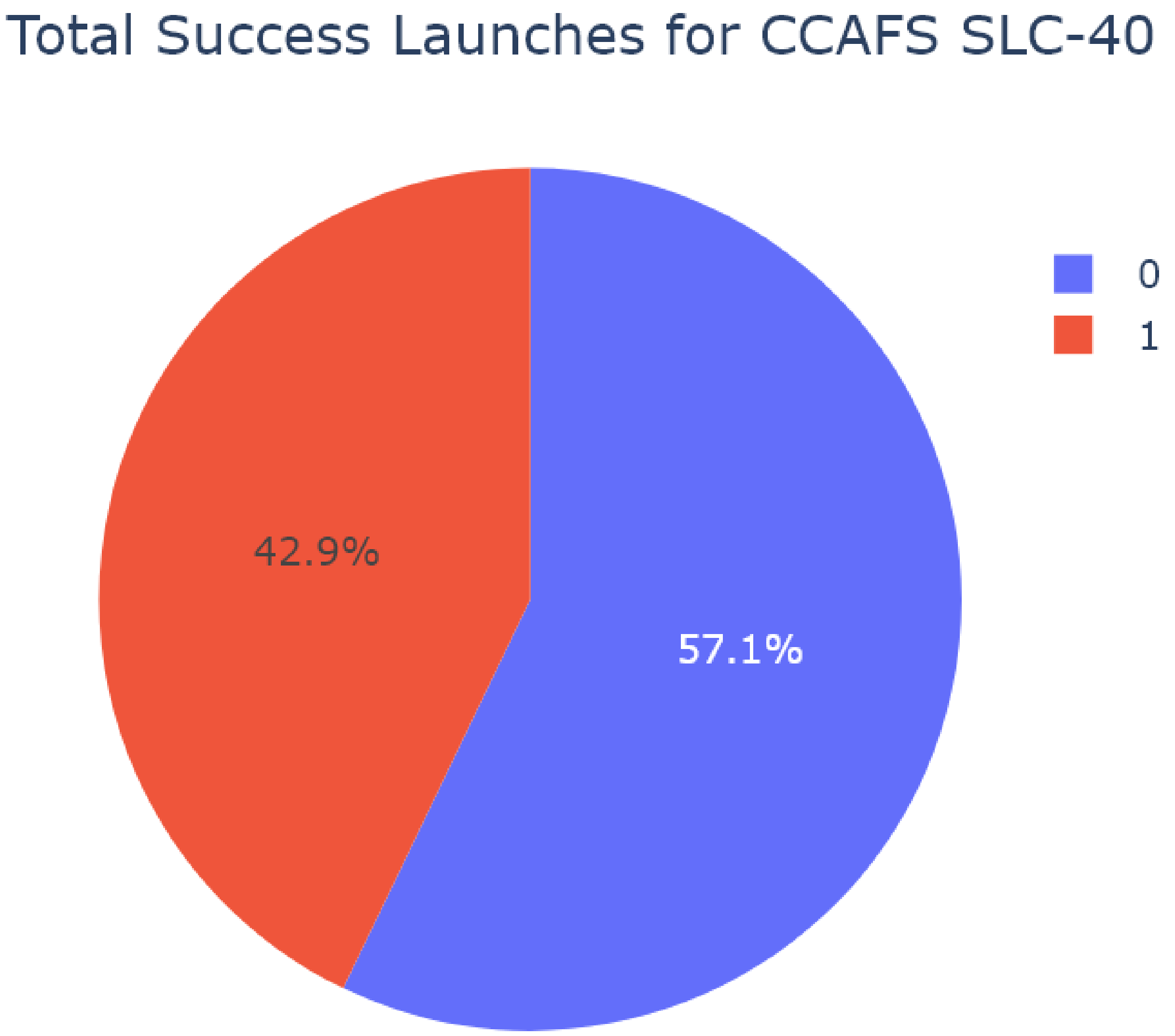
Plotly Dashboard: All Launch Sites

- 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 outcomes 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%.

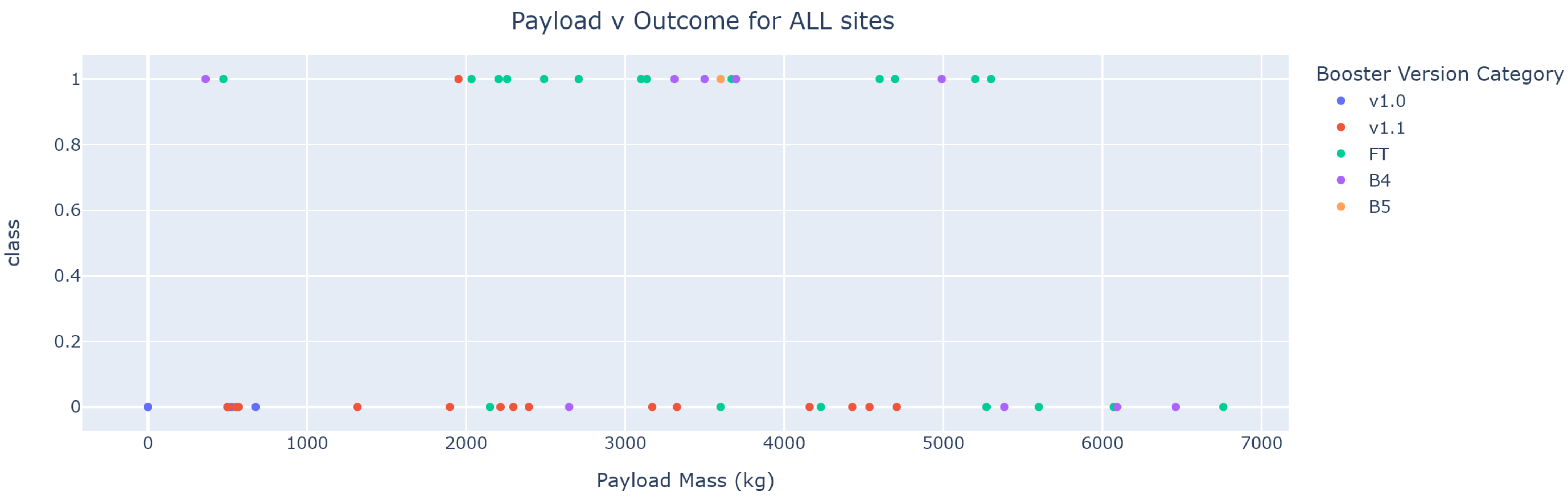


Plotly Dashboard: Most Successful Launch Site

- Site CCAFS SLC-40 was the most successful launch site, with 42.9% of launches ending successfully.
- 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 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 the 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.

Thank you!