IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Daniel Gosselin
11 September 2023

Outline

Executive Summary

- Data was collected using an API and by webscraping, and wrangled to prepare for classification
- Data was explored to find relevant trends to investigate, using Pandas, SQL, and visualization using Folium maps and Plotly interactive dashboards
- Four machine learning models were run to determine the best performing approach to predicting mission success
- I determined the cost of each launch to be $62 million, by assuming the same cost as advertised by SpaceX
- I developed a model that can predict launch success with 83% accuracy

Introduction

- Project background and context:
  - The company Space Y, founded by Billionaire industrialist Allon Musk, would like to complete with SpaceX
  - As a data scientist at Space Y, I have been asked to determine the cost of each launch, by gathering information about Space X and creating dashboards for my team
  - I will also determine if SpaceX will reuse the first stage, making a prediction by using a machine learning model and public information.
- Problems to solve:
  - What is the cost of a launch
  - Will SpaceX reuse the first stage of a Falcon 9 rocket
- GitHub Link for all reference files: https://github.com/dgossghub46/final_project.git

Section 1

Methodology

Methodology

Executive Summary

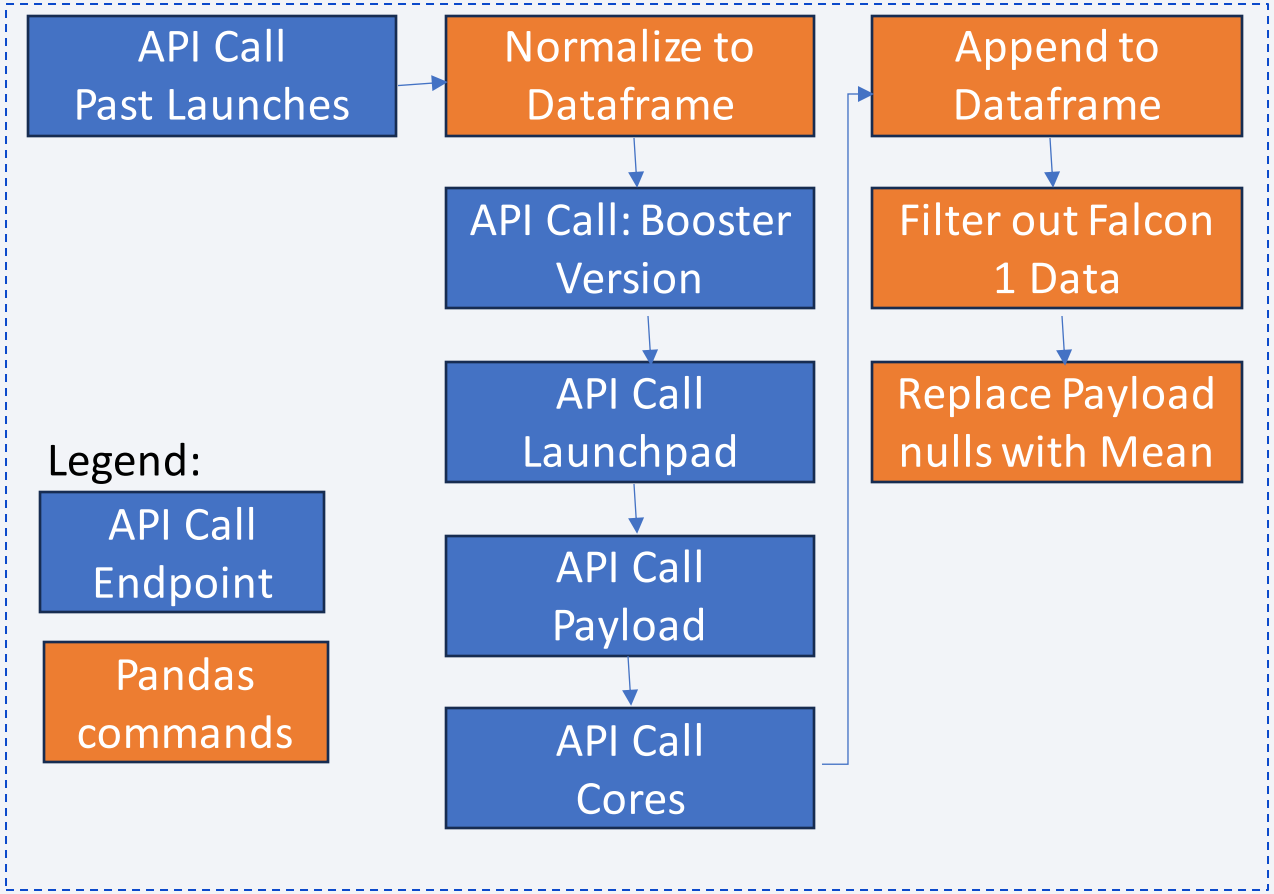
- Data collection methodology:
  - Data was collected using an API call and by webscraping
- Perform data wrangling
  - Describe how data was processed
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - How to build, tune, evaluate classification models

Data Collection

- The data sets were collected in two ways: using an API call, and by webscraping.
- These two methods are described on the following slides.

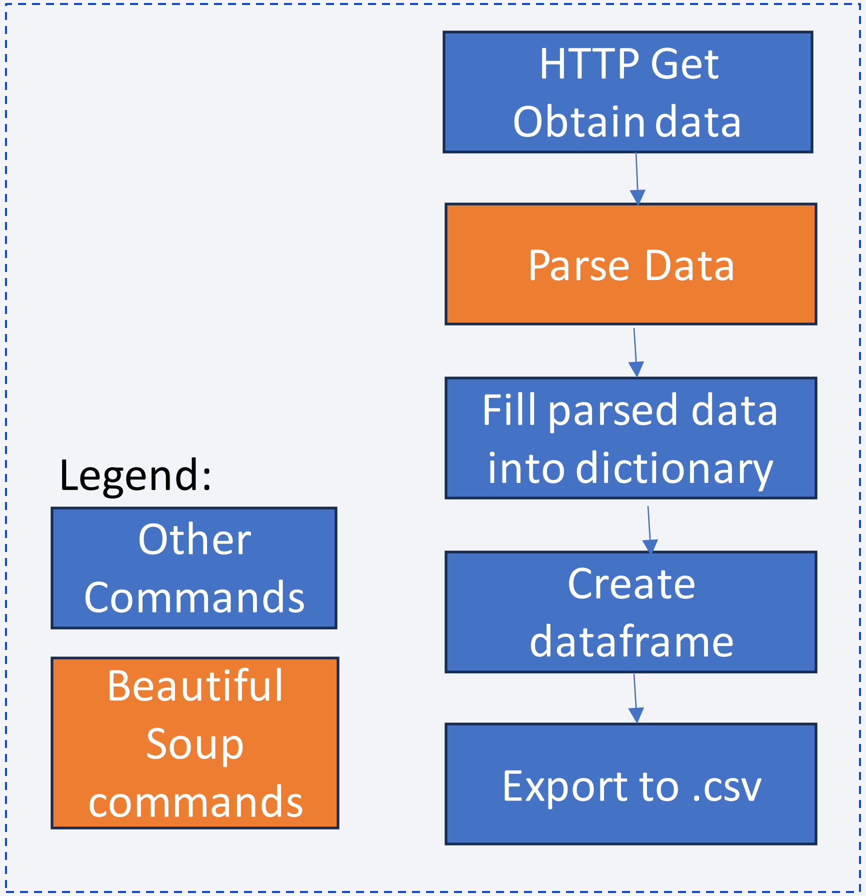
Data Collection – SpaceX API

- SpaceX REST API called five times to gather data:
  - First, /launches/past: past launches data. The response is a list of json objects, which are normalized into a Pandas dataframe.
  - Certain columns contain identification numbers instead of data. Subsequent API calls are made to other endpoints (/rockets, /launchpads, /payloads, /cores). The responses are appended to the dataframe.
- Falcon 1 data is removed from the response, leaving Falcon 9 data of interest. The Flight Number column is reset.
- Nulls:
  - 'Payload Mass' column: replaced with the mean; and
  - 'LandingPad' column: not changed, these are acceptable.
- GitHub URL of notebook: https://github.com/dgossghub46/final_project/blob/master/1-1%20Complete%20jupyter-labs-spacex-data-collection-api.ipynb



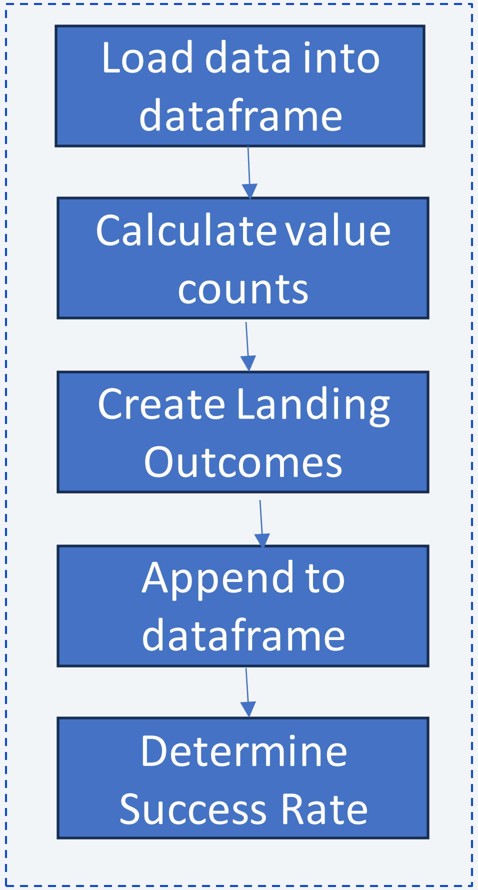
Data Collection – Web Scraping

- Webscrape the Wikipedia page for Falcon 9 launch records using Python Beautiful Soup package
- Parse, and convert to Pandas dataframe
- Links:
  - Wikipedia data: https://en.wikipedia.org/w/index.php?title=List_of_Falcon_9_and_Falcon_Heavy_launches&oldid=1027686922
  - Github notebook: https://github.com/dgossghub46/final_project/blob/master/1-2%20Complete%20jupyter-labs-webscraping.ipynb



Data Wrangling

- Load dataset into Pandas dataframe
- Calculate using value counts:
  - Number of launches on each site
  - Number and occurrence of each orbit
  - Number and occurrence of mission outcome per orbit
- Create a set of outcomes where landing was not successful
- Create a landing outcome label depending on landing success (1) or failure (0), and add to dataframe
- Determine the success rate using the mean function on landing outcome – See Appendix A
- GitHub URL to notebook: https://github.com/dgossghub46/final_project/blob/master/1-3%20Complete%20labs-jupyter-spacex-data_wrangling_jupyterlite.jupyterlite.ipynb



EDA with Data Visualization

Chart (type)	Purpose	Findings
Payload vs Flight Number (colour by success) (Categorical)	See how flight number and payload affect launch outcomes.	As flight number increases, more likely to succeed. As payload increases, less likely to succeed.
Launch Site vs Flight Number (colour by success) (Categorical)	See how success rate varies by launch site	CCAFS (60%) has a lower success rate than KSC LC-39A and VAFB SLC 4E (both 77%)
Launch Site vs Payload Mass (colour by success) (Categorical)	Determine any relationships	VAFB-SLC site had no launches over 10,000 kg
Success rate vs Orbit (Bar Plot)	See if the orbit type has an influence on success rate	90-100%: ES-L1, GEO, HEO, SSO, VLEO 50-70%: GTO, ISS, LEO, MEO, PO 0%: SO (0/1)
Orbit Type vs Flight Number (colour by success) (Scatter)	Determine any relationships	LEO showed increased success over time, while GTO did not.

EDA with Data Visualization (../2)

Chart (type)	Purpose	Findings
Orbit vs Payload Mass (colour by success) (Scatter)	Look for relationships	Higher success for higher payloads for PO, ISS, LEO. Not the case for GTO.
Annual Success Rate vs Year	Determine trend	Generally increasing success by year.

- See graphs in Appendix B
- Get_dummies function used to apply one-hot encoding to prepare dataset for machine learning by converting all data to numbers, and casting all numeric columns to float64
- GitHub link to notebook: https://github.com/dgossghub46/final_project/blob/master/2-2%20Complete%20jupyter-labs-eda-dataviz.ipynb.jupyterlite.ipynb

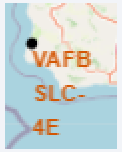
EDA with SQL

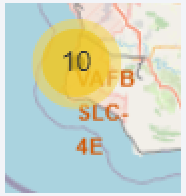
- Main SQL queries performed:
  - Identify distinct launch sites: SELECT DISTINCT "Launch_Site" FROM SPACEXTABLE
  - Determine the total payload mass (result: 45,596 kg): SELECT SUM(PAYLOAD_MASS__KG_) FROM SPACEXTABLE WHERE "Customer" = "NASA (CRS)"
  - Determine average payload mass carried by the Falcon 9 booster version 1.1 (result: 2928.4 kg): SELECT AVG(PAYLOAD_MASS__KG_) FROM SPACEXTABLE WHERE "Booster_Version" = "F9 v1.1"
  - Determine the date of the first successful landing on a ground pad (22 Dec 2015): SELECT MIN("Date") FROM SPACEXTABLE WHERE "Landing_Outcome" = "Success (ground pad)"
  - Identify the names of boosters having success landing on drone ships with payloads between 4000-6000 kg: SELECT DISTINCT("Booster_Version") FROM SPACEXTABLE WHERE "Landing_Outcome" = "Success (drone ship)" AND "PAYLOAD_MASS__KG_" BETWEEN 4000 AND 6000
  - Determine the number of successful and failed mission outcomes: SELECT "Mission_Outcome",COUNT("Mission_Outcome") FROM SPACEXTABLE GROUP BY "Mission_Outcome"
  - List the booster versions having carried the maxmimum payload mass: SELECT DISTINCT("Booster_Version") FROM SPACEXTABLE WHERE PAYLOAD_MASS__KG_ = (SELECT MAX("PAYLOAD_MASS__KG_") FROM SPACEXTABLE)
  - List certain data by month for launches in 2015: SELECT SUBSTR("Date",6,2) AS "Month","Landing_Outcome","Booster_Version","Launch_Site" FROM SPACEXTABLE WHERE SUBSTR("Date",1,4) = "2015"
  - Rank the count of landing outcomes for certain dates in descending order: SELECT "Landing_Outcome",COUNT(*) AS COUNTS FROM SPACEXTABLE WHERE "Date" BETWEEN "2010-06-04" AND "2017-03-20" GROUP BY "Landing_Outcome" ORDER BY COUNTS DESC

- GitHub URL for notebook: https://github.com/dgossghub46/final_project/blob/master/2-1%20Complete2%20jupyter-labs-eda-sql-coursera_sqllite.ipynb

Build an Interactive Map with Folium

- Objects added to map:
  - Circle and text label to show NASA Johnson Space Center location (to visualize location)
  - Markers added for launch sites, to confirm they are all near the equator and the coast.
  - Marker clusters to show launch success rates at each site
  - Line and line length to show distance between points





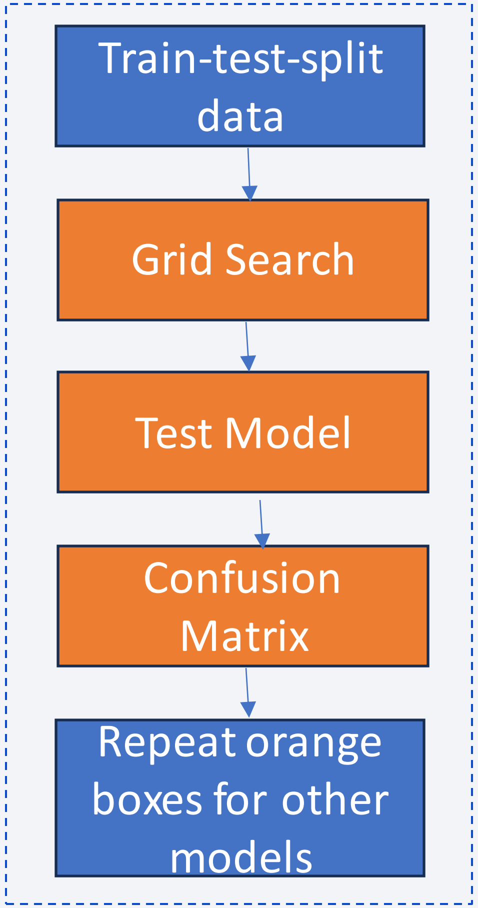
- GitHub URL of interactive map with Folium: https://github.com/dgossghub46/final_project/blob/master/3-1%20Complete2%20IBM-DS0321EN-SkillsNetwork_labs_module_3_lab_jupyter_launch_site_location.jupyterlite.ipynb

Build a Dashboard with Plotly Dash

- Pie Chart: to show the success rate by launch site
    - Drop-down: select launch site, to show success/failure rate for one launch site at a time
- Scatter Chart, coloured by booster version: to show the success rate by payload mass with insight into any booster version dependency
    - Payload mass slider: to narrow the range of payload mass shown
    - Drop-down: to narrow results by all or a selected launch site
- GitHub URL of Plotly Dash lab: https://github.com/dgossghub46/final_project/blob/master/3-1%20Complete%20spacex_dash_app.py

Predictive Analysis (Classification)

- Train-test-split was used to divide the dataset, 80% train/20% test
- Gridsearch was run on multiple models to determine the best hyperparameters
  - Models: Logistic Regression, Support Vector Machine, Decision Tree Classifier, k-Nearest Neighbours
- Each model was then run on the test data using the best hyperparameters, the accuracy was calculated, and the confusion matrix was generated
- GitHub URL of predictive analysis lab: https://github.com/dgossghub46/final_project/blob/master/4-1%20Complete%20SpaceX_Machine_Learning_Prediction_Part_5.jupyterlite.ipynb



Results

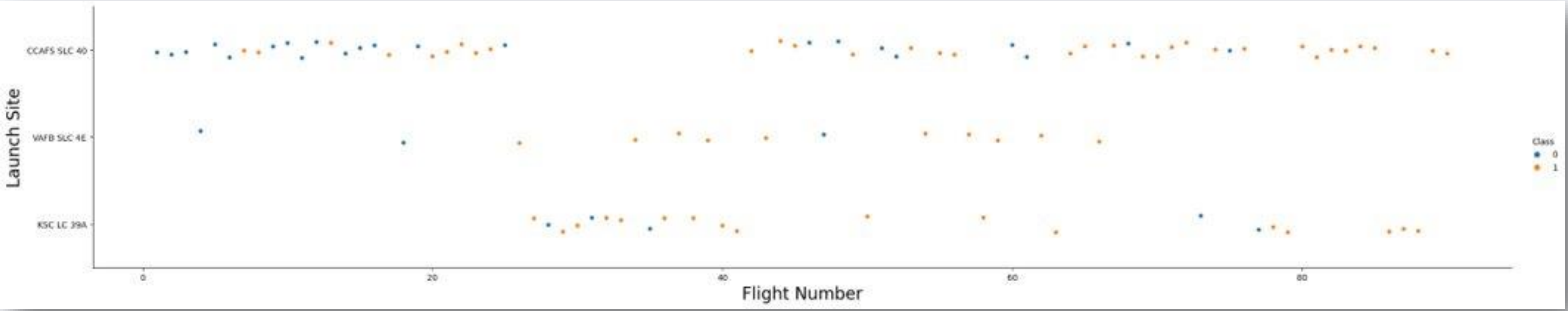
- Exploratory data analysis results
- Interactive analytics demo in screenshots
- Predictive analysis results

Section 2

Insights drawn from EDA

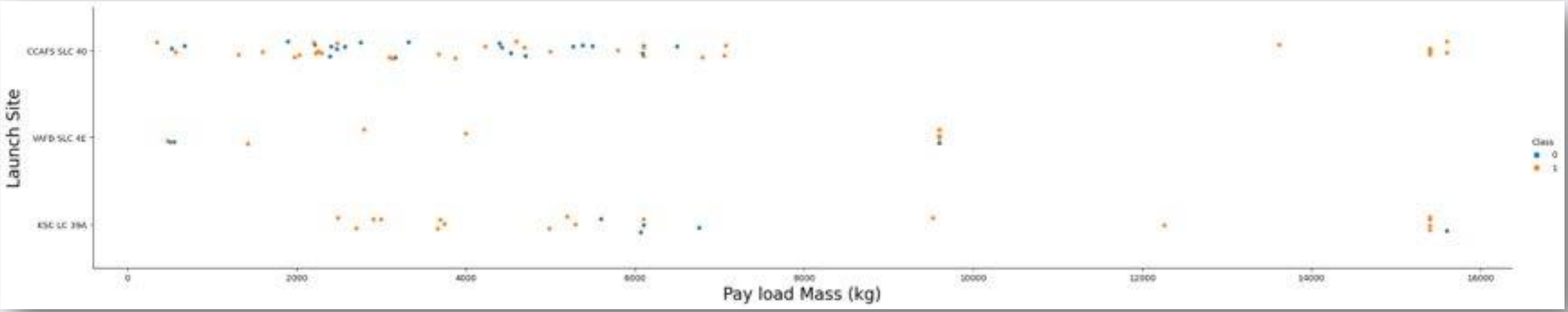
Flight Number vs. Launch Site

- The scatter plot of Flight Number vs. Launch Site, coloured by success (1 – red) or failure (0 – blue), shows:
  - Flights were concentrated at CCAFS SLC 40 until about flight 25, when the majority of flights were conducted at KSC LC 39A until flight 41
  - CCAFS SLC 40 is the busiest site, VAFB SLC 4E is the least busy and has not seen a flight since before flight 70
  - Success rates have generally been improving over time, particularly at CCAFS SLC 40



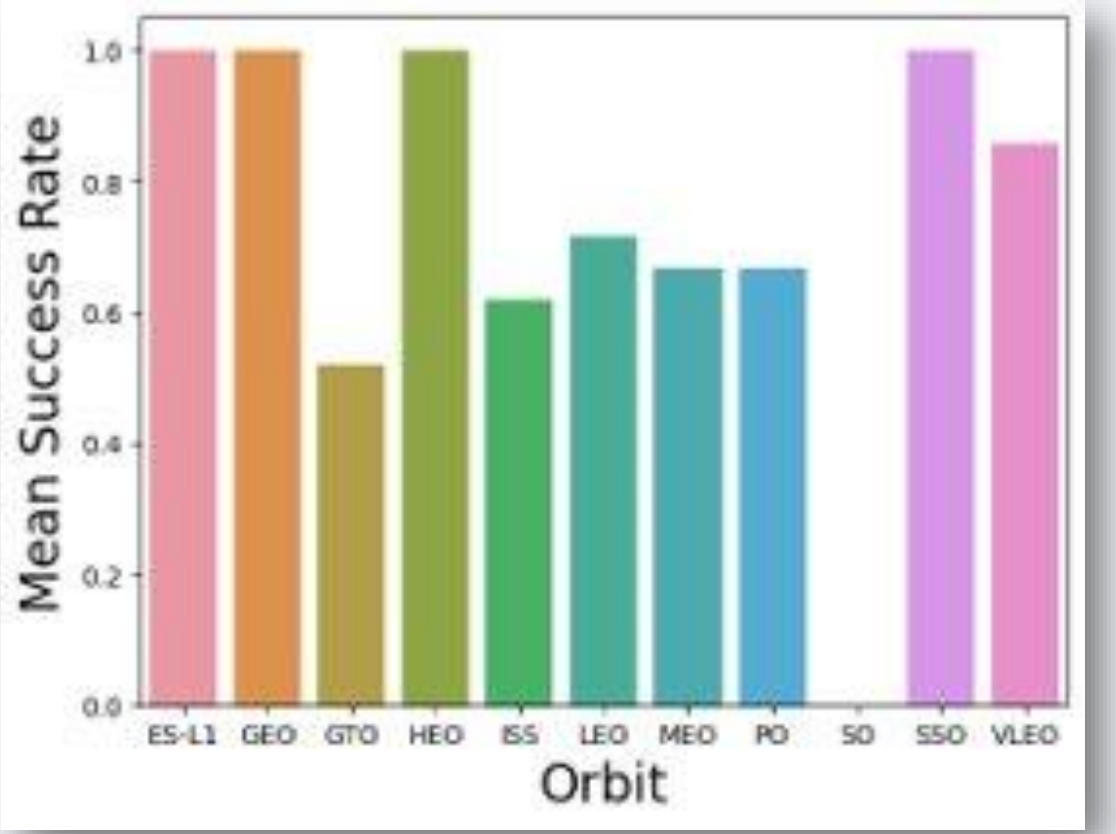
Payload vs. Launch Site

- Scatter plot of Payload vs. Launch Site, coloured by success (1 – red) or failure (0 – blue)
- Most flights had a payload of less than 8000 kg
- There is a high success rate for payloads exceeding 8000 kg
- The heaviest payloads are launched from CCAFS SLC 40 or KSC LC 39A, not from VAFB SLC 4E



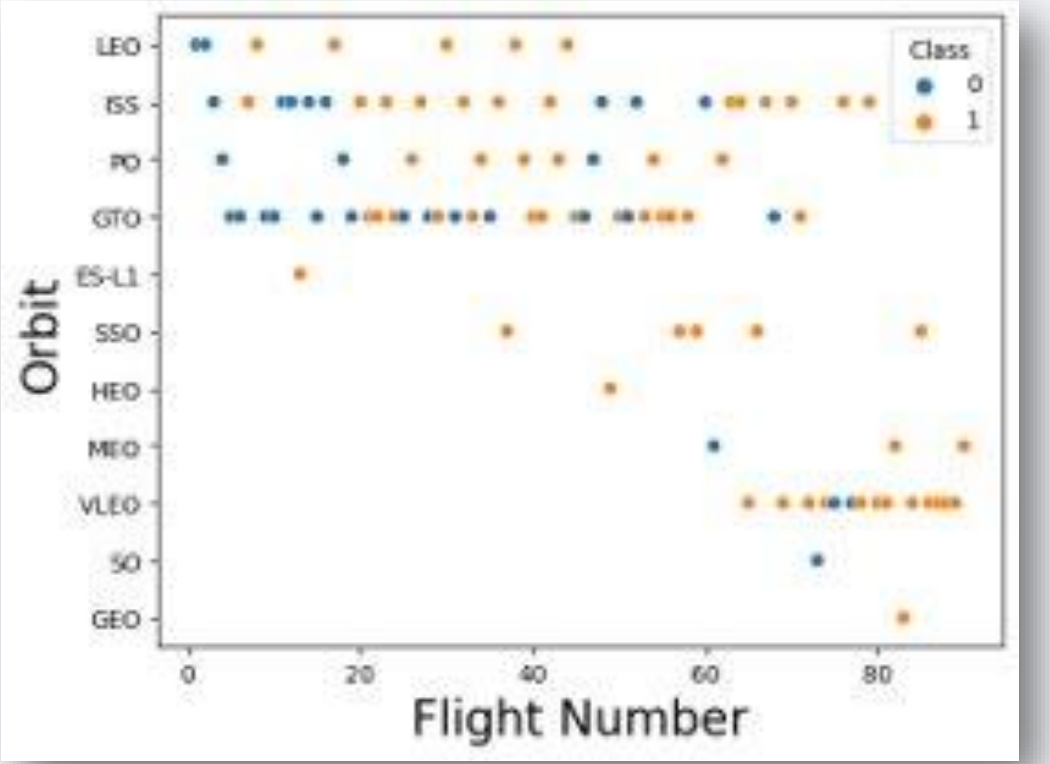
Success Rate vs. Orbit Type

- Bar chart for the success rate of each orbit type
- 90-100%: ES-L1, GEO, HEO, SSO, VLEO
- 50-70%: GTO, ISS, LEO, MEO, PO
- 0%: SO (0/1)



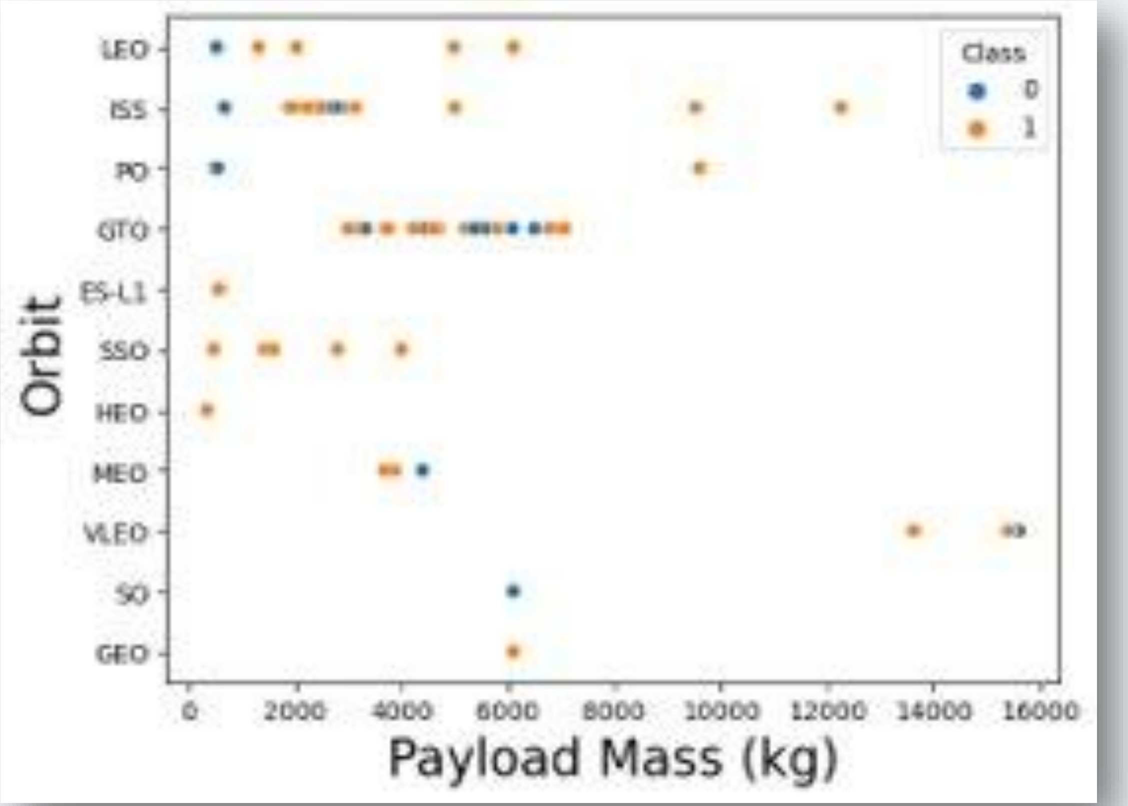
Flight Number vs. Orbit Type

- A scatter plot of Flight number vs. Orbit type
- Early flights arrived at orbits: LEO, ISS, PO and GTO
- Later flights were concentrated at VLEO
- A small number of SSO, HEO, MEO, SO and GEO flights wre also conducted, later in time



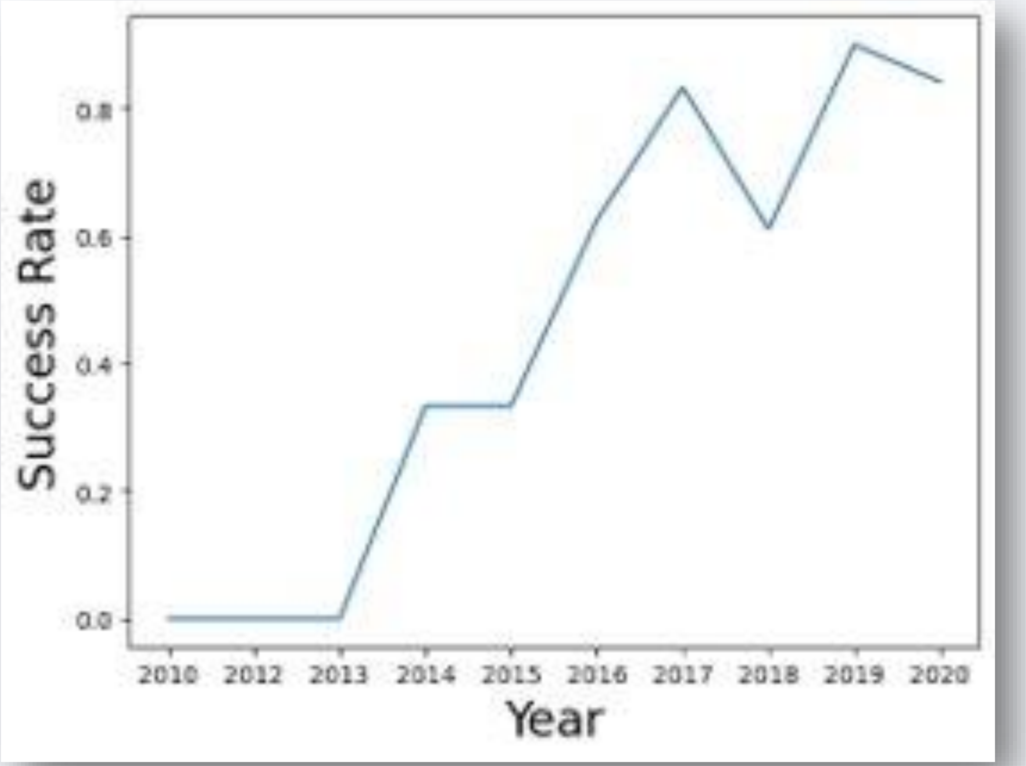
Payload vs. Orbit Type

- A scatter plot of payload vs. orbit type
- Heavier payloads are destined for VLEO
- GTO has a relatively small range of payload
- ISS is generally small payloads but has some outliers at heavier masses



Launch Success Yearly Trend

- A line chart of yearly average success rate
- Success rate was zero for the first years, 2010-2013
- The success rate increased monotonically until 2017, and reached an all-time peak in 2019



All Launch Site Names

- A query returning the launch site names is shown
- The DISTINCT qualifier ensures that duplicate values are not returned.

Task 1

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

```
%sql SELECT DISTINCT "Launch_Site" FROM SPACEXTABLE
```

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

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

Launch Site Names Begin with 'CCA'

- Find 5 records where launch sites begin with `CCA`
- "LIKE" was used to find the letters "CCA" in the "LAUNCH SITE". The % wildcard was used to ensure the CCA was at the beginning of the site name
- LIMIT 5 was used to ensure only the first 5 results were returned

Task 2

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

```
%sql SELECT * FROM SPACEXTABLE WHERE "Launch_Site" LIKE "CCA%" LIMIT 5

### Task 3



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

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

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	No attempt
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

- Calculate the total payload carried by boosters from NASA
- This query returned the sum of all payloads where the customer was NASA

```
%sql SELECT SUM(PAYLOAD_MASS__KG_) FROM SPACEXTABLE WHERE "Customer" = "NASA (CRS)"
```

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

SUM(PAYLOAD_MASS__KG_)
45596

Average Payload Mass by F9 v1.1

- Calculate the average payload mass carried by booster version F9 v1.1
- This query found all payload masses for which the booster version was F9 v1.1, and returned the average

Task 4

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

```
%sql SELECT AVG(PAYLOAD_MASS__KG_) FROM SPACEXTABLE WHERE "Booster_Version" = "F9 v1.1"
```

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

AVG(PAYLOAD_MASS__KG_)
2928.4

First Successful Ground Landing Date

- Find the dates of the first successful landing outcome on ground pad
- This query found all dates where the landing outcome was success (ground pad), then returned the minimum of those dates (e.g. the earliest date)

Task 5

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

Hint:Use min function

```
%sql SELECT MIN("Date") FROM SPACEXTABLE WHERE "Landing_Outcome" = "Success (ground pad)"
```

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

MIN("Date")
2015-12-22

Successful Drone Ship Landing with Payload between 4000 and 6000

- List the names of boosters which have successfully landed on drone ship and had payload mass greater than 4000 but less than 6000
- This query found all records where the landing outcome was Success (Drone Ship) and the payload mass was between 4000 and 6000. It then returned the associated booster versions, using "distinct" to remove duplicate values.

Task 6

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

```
%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.
```

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

Total Number of Successful and Failure Mission Outcomes

- Calculate the total number of successful and failure mission outcomes
- This query counted the mission outcomes, grouped by mission outcome

Task 7

List the total number of successful and failure mission outcomes

```
%sql SELECT "Mission_Outcome",COUNT("Mission_Outcome") FROM SPACEXTABLE GROUP BY "Mission_Outcome"
```

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

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

Boosters Carried Maximum Payload

- List the names of the booster which have carried the maximum payload mass
- A subquery was used to find the maximum payload mass in the table
- The main query found the booster versions associated with that maximum payload mass, using DISTINCT to remove duplicate values

Task 8

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

```
%sql SELECT DISTINCT("Booster_Version") FROM SPACEXTABLE WHERE PAYLOAD_MASS__KG_ = (SELECT MAX("PAYLOAD_MASS__KG_") FROM SPACEXTABLE)
```

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

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

2015 Launch Records

- List the failed landing_outcomes in drone ship, their booster versions, and launch site names for in year 2015
- The query selected a number of columns, renaming part of the date column to "Month", where the part of the date column indicating the year was equal to 2015.

Task 9

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

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.

```
#%sql SELECT * FROM SPACEXTABLE
%sql SELECT SUBSTR("Date",6,2) AS "Month","Landing_Outcome","Booster_Version","Launch_Site" FROM SPACEXTABLE WHERE SUBSTR("Date",1,4) = "2015"
```

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

Month	Landing_Outcome	Booster_Version	Launch_Site
10	Failure (drone ship)	F9 v1.1 B1012	CCAFS LC-40
11	Controlled (ocean)	F9 v1.1 B1013	CCAFS LC-40
02	No attempt	F9 v1.1 B1014	CCAFS LC-40
04	Failure (drone ship)	F9 v1.1 B1015	CCAFS LC-40
04	No attempt	F9 v1.1 B1016	CCAFS LC-40
06	Precluded (drone ship)	F9 v1.1 B1018	CCAFS LC-40
12	Success (ground pad)	F9 FT B1019	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
- The COUNT function counted the entries, GROUP BY to total by Landing_Outcome, where the date was within the range of interest

Task 10

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

```
%%sql SELECT "Landing_Outcome",COUNT(*) AS COUNTS FROM SPACEXTABLE
WHERE "Date" BETWEEN "2010-06-04" AND "2017-03-20"
GROUP BY "Landing_Outcome" ORDER BY COUNTS DESC
```

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

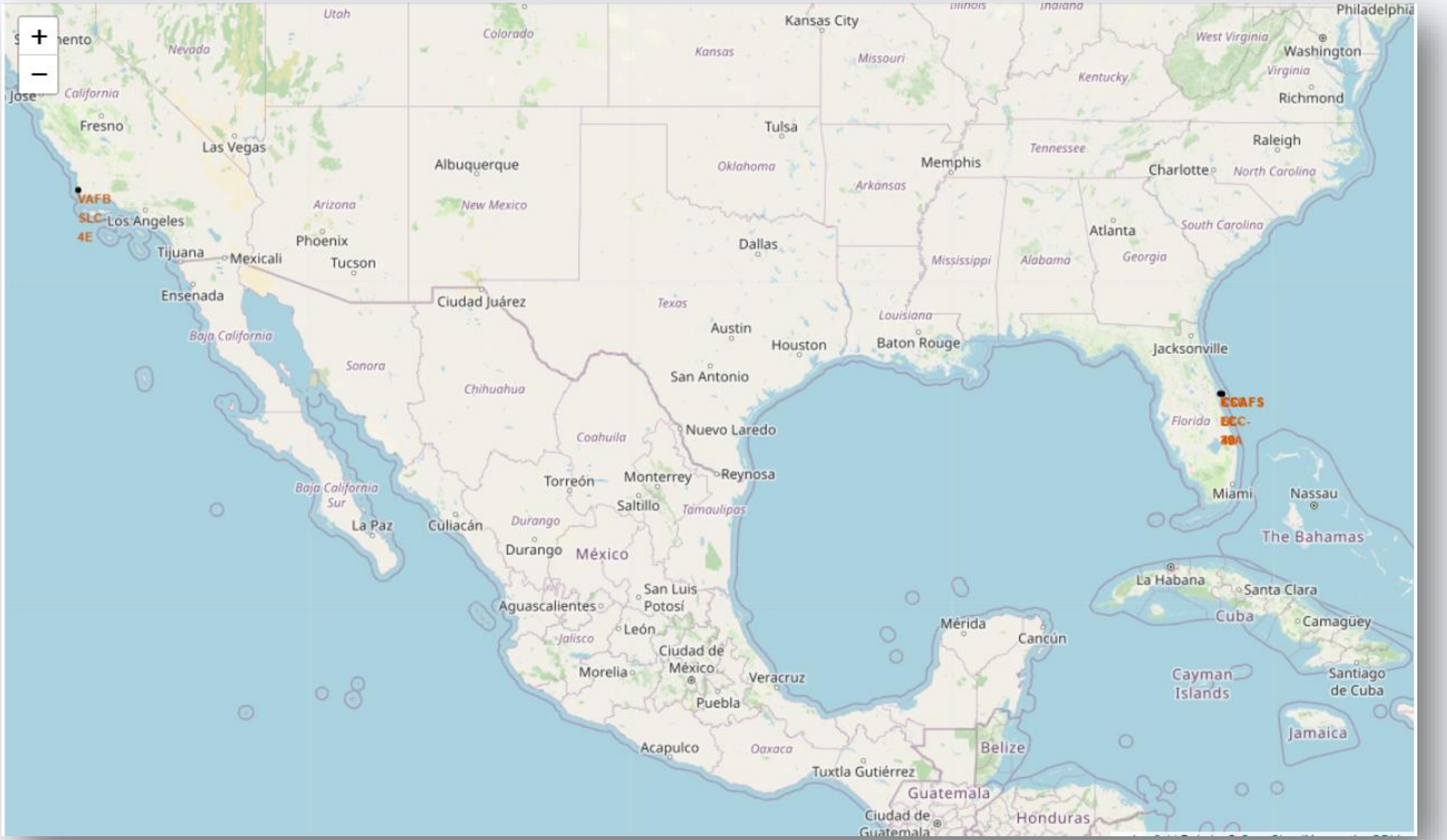
Landing_Outcome	COUNTS
No attempt	10
Success (ground pad)	5
Success (drone ship)	5
Failure (drone ship)	5
Controlled (ocean)	3
Uncontrolled (ocean)	2
Precluded (drone ship)	1
Failure (parachute)	1

Section 3

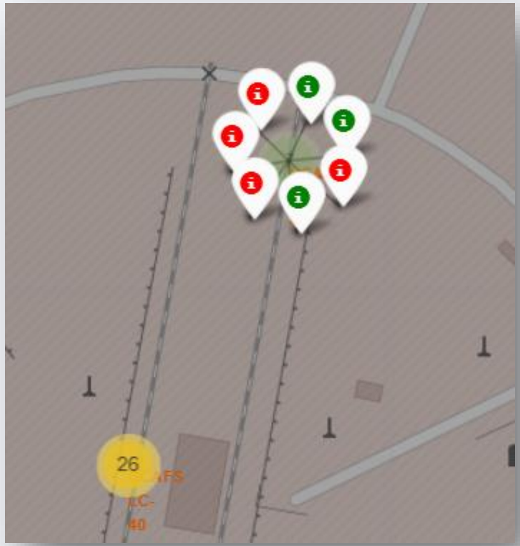
Launch Sites Proximities Analysis

Map of Launch Site Locations

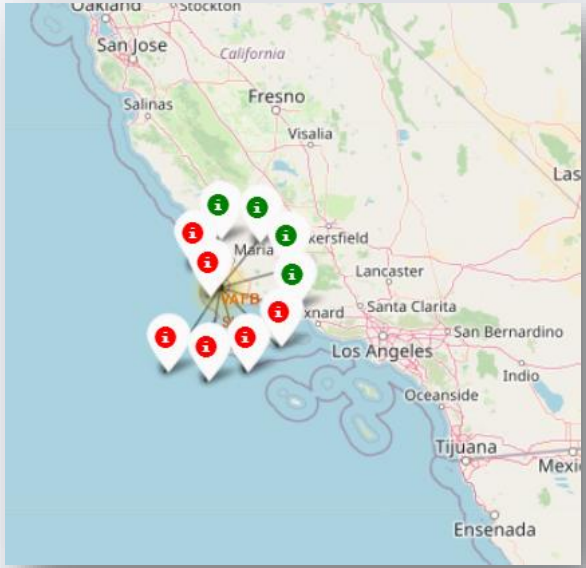
- Launch sites are near the equator, to take advantage of the Earth's rotational speed
- Launch sites are near the coast, likely for safety to avoid unpopulated areas

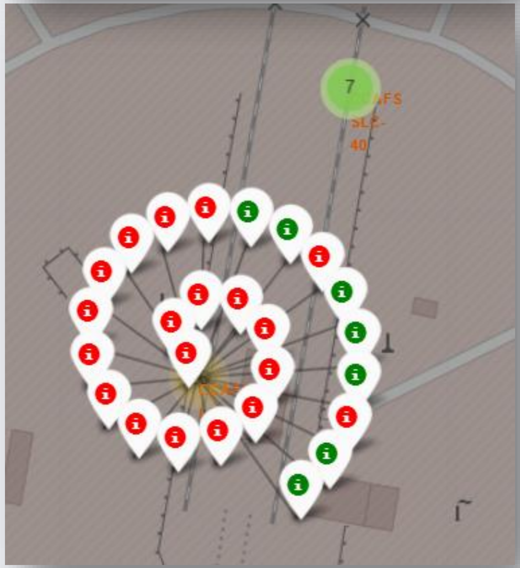


Launch Outcomes by Launch Site



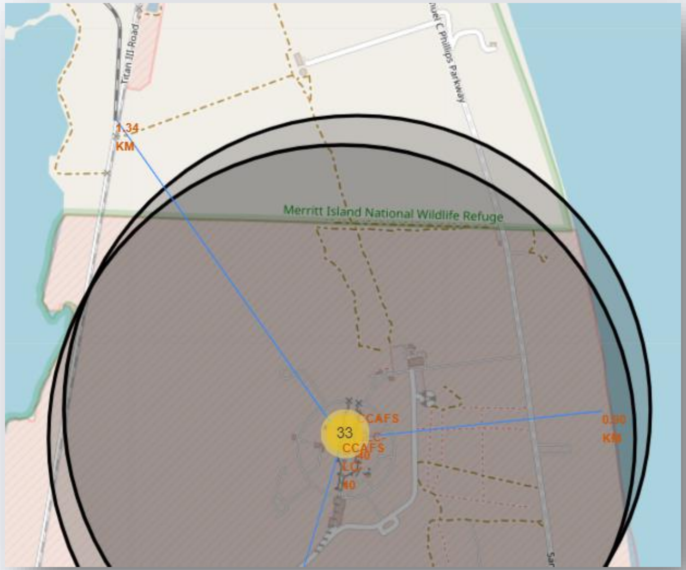
- Each marker indicates a launch: green markers indicate launch success; red markers indicate launch failure.
- Findings:
  - More launch sites on the East coast (3) than the West coast (1)
  - CCAFS LC-40 is the busiest launch site (26), followed by KSC LC-39A (13), VAFB SLC-4E (10) and CCAFS SLC-40 (7)
  - KSC LC-39A has the best success rate, the others are less than 50%

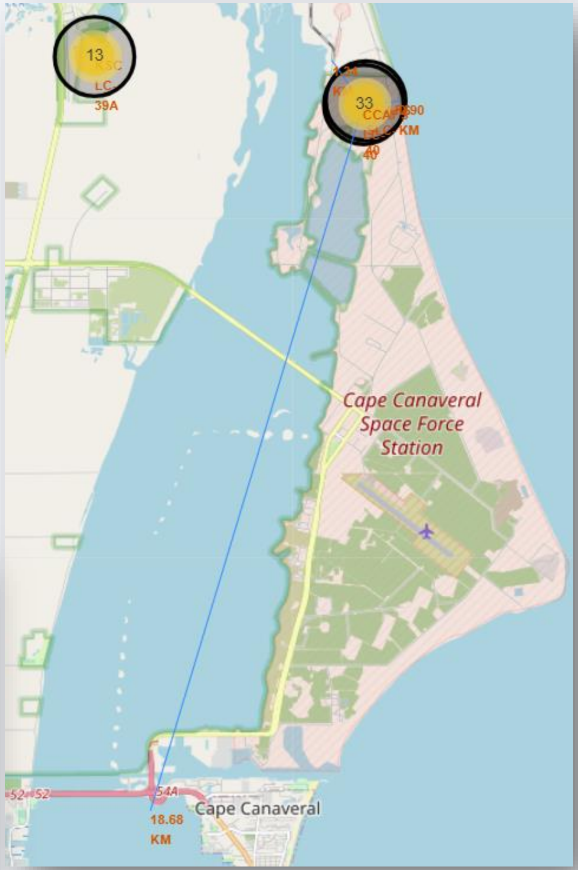




Nearby Points of Interest: Launch Site CCAFS LC-40

- Measurements:
    - Distance to coastline: 0.9 km
    - Distance to rail: 1.3 km
    - Distance to highway and nearest city: 18.7 km
- Observations:
    - Launch sites are near the coast, likely for safety
    - Launch sites are near rail, likely for logistics reasons
    - Launch sites are further from cities, likely for safety



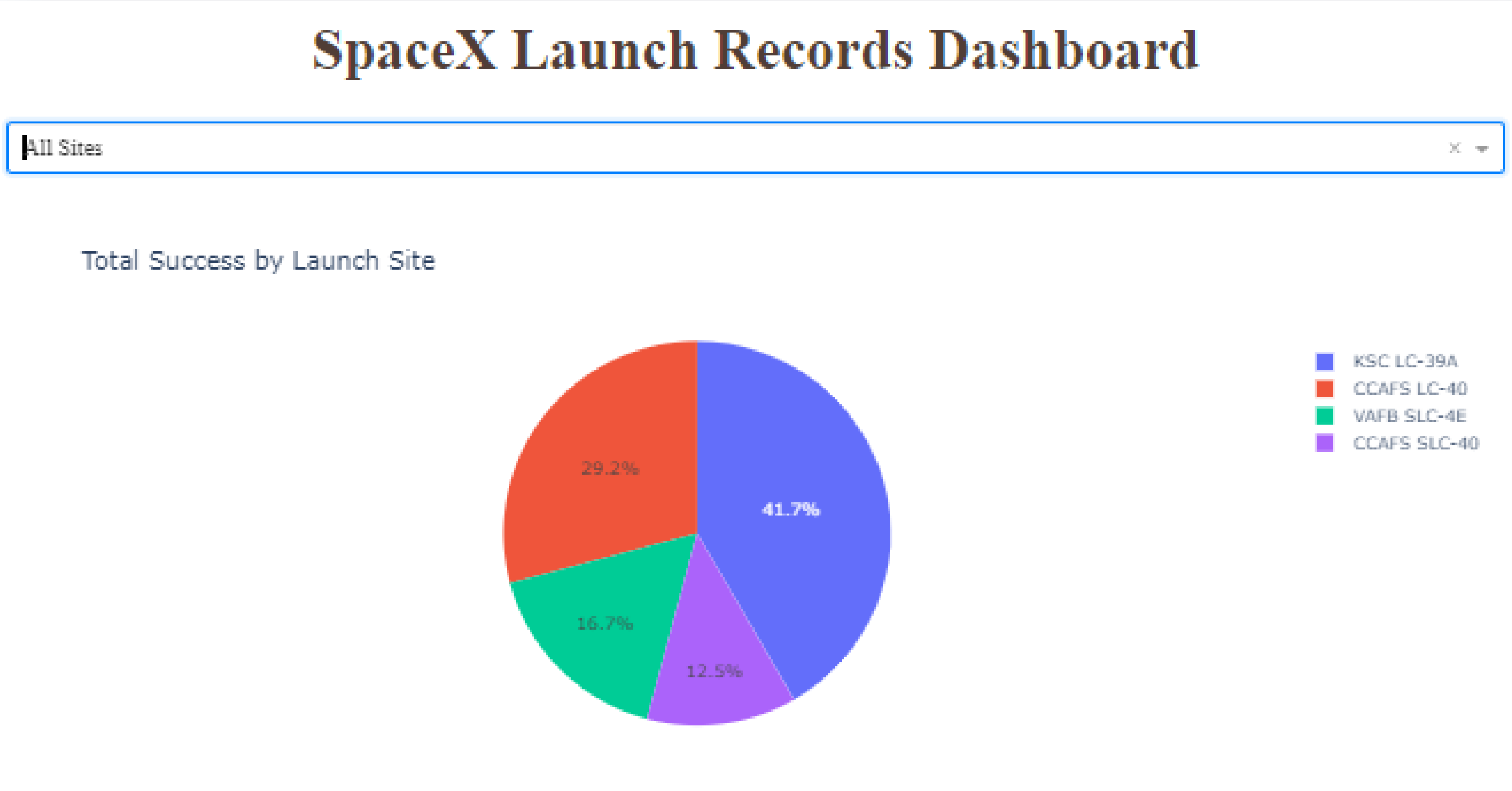


Section 4

Build a Dashboard with Plotly Dash

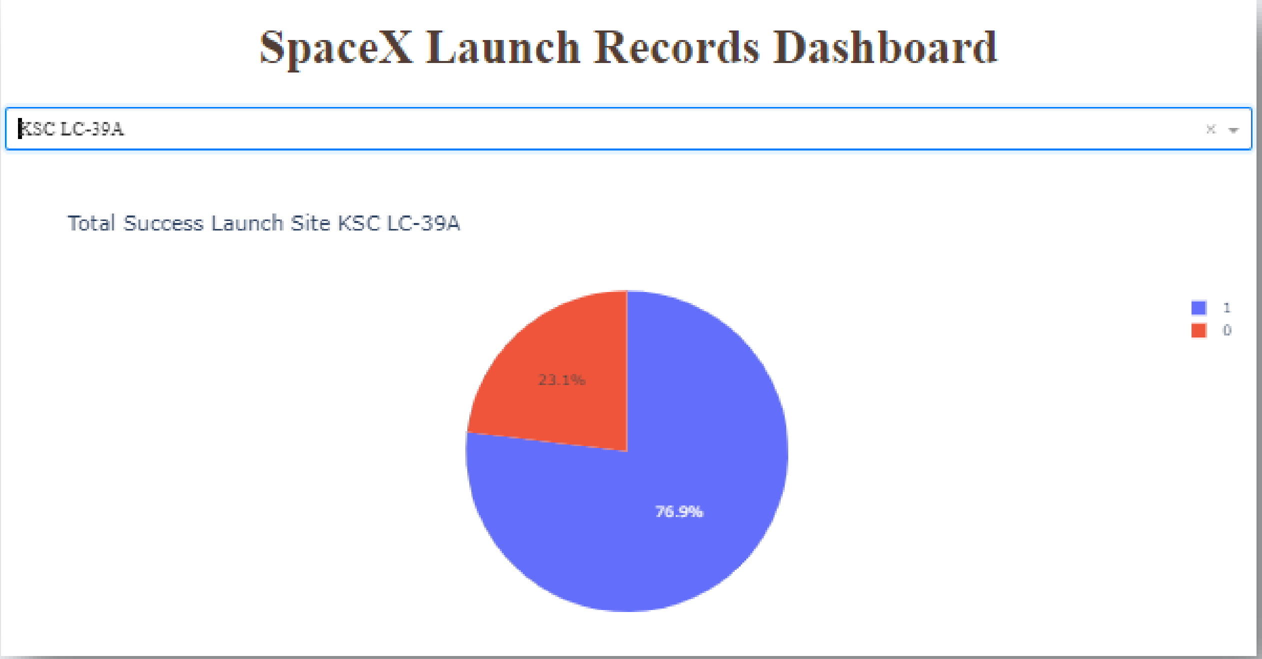
Launch Success Count for All Sites

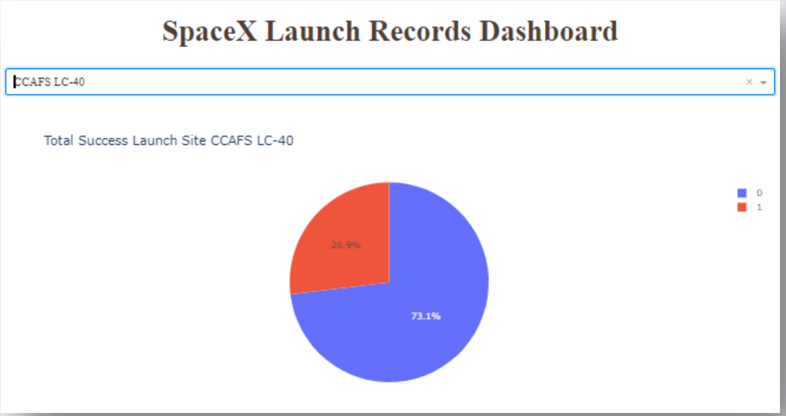
- This chart shows that the most successful launches come from, in descending order:
  - KSC LC-39A
  - CCADS LC-40
  - VAFB SLC-4E
  - CCAFS SLC-40
- However, this does not give an indication of each site's success rate

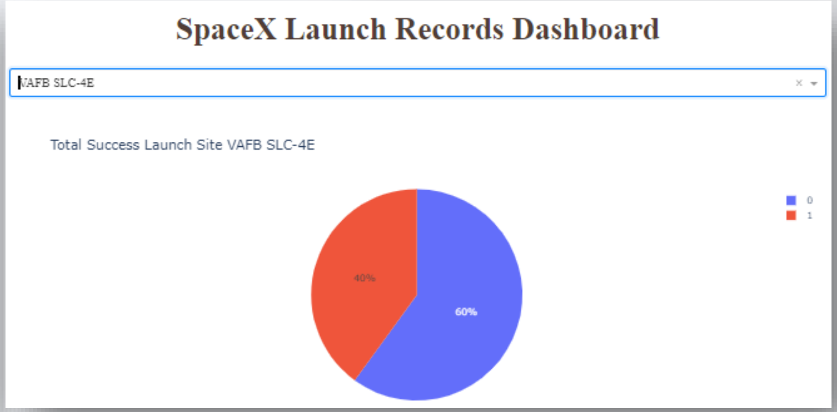


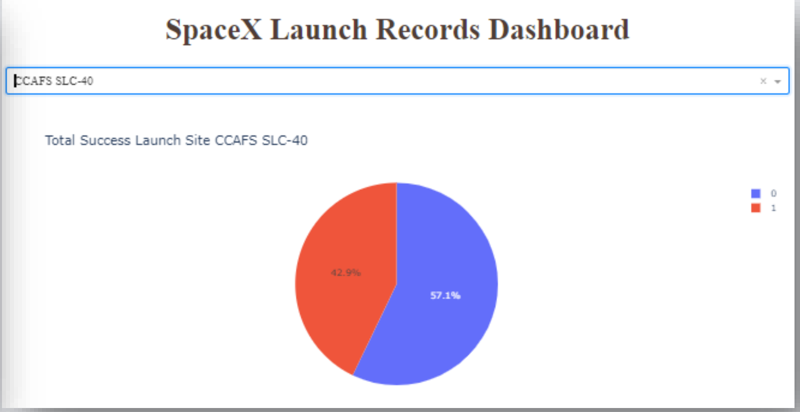
Launch Site Success Rates

- Launch site KSC LC-39A had the highest success rate at 76.9%
- The success rate graphs for the other launch sites are included for comparison



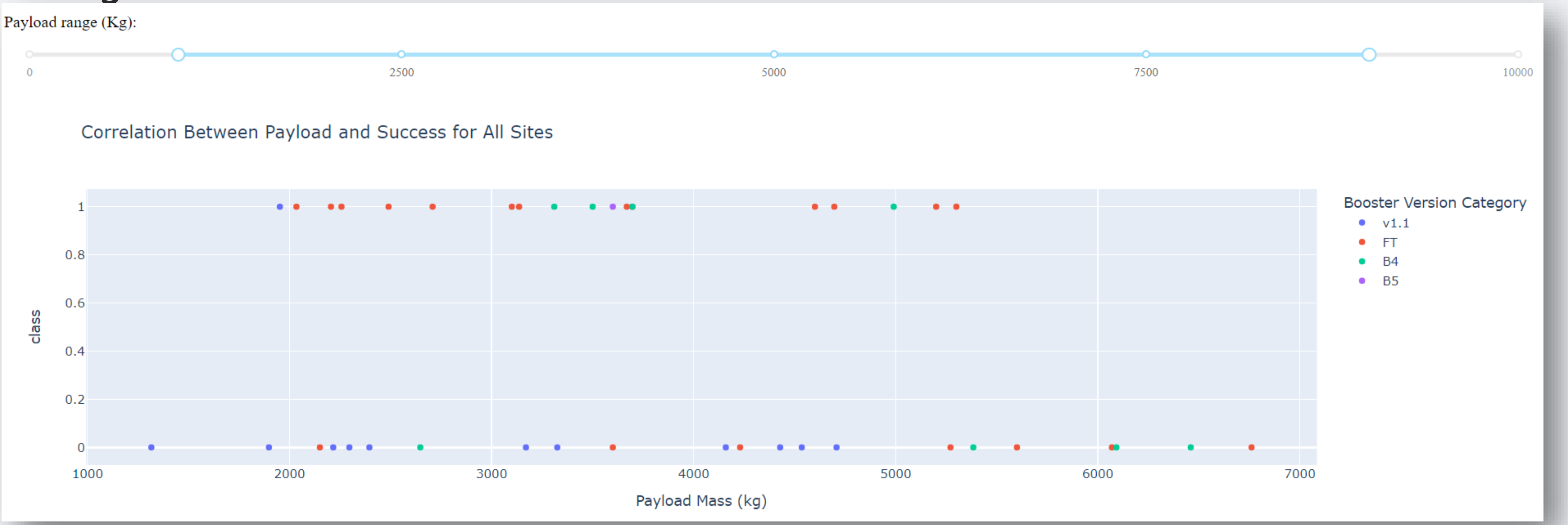






Influence of Payload on Launch Outcome

- Payloads above 5500 kg have a 0% success rate
- Booster version FT has the highest success rate, version v1.1 has the lowest success rate
- Booster versions B5 and v1.1 are typically used for payloads with a mass not exceeding 4800 kg

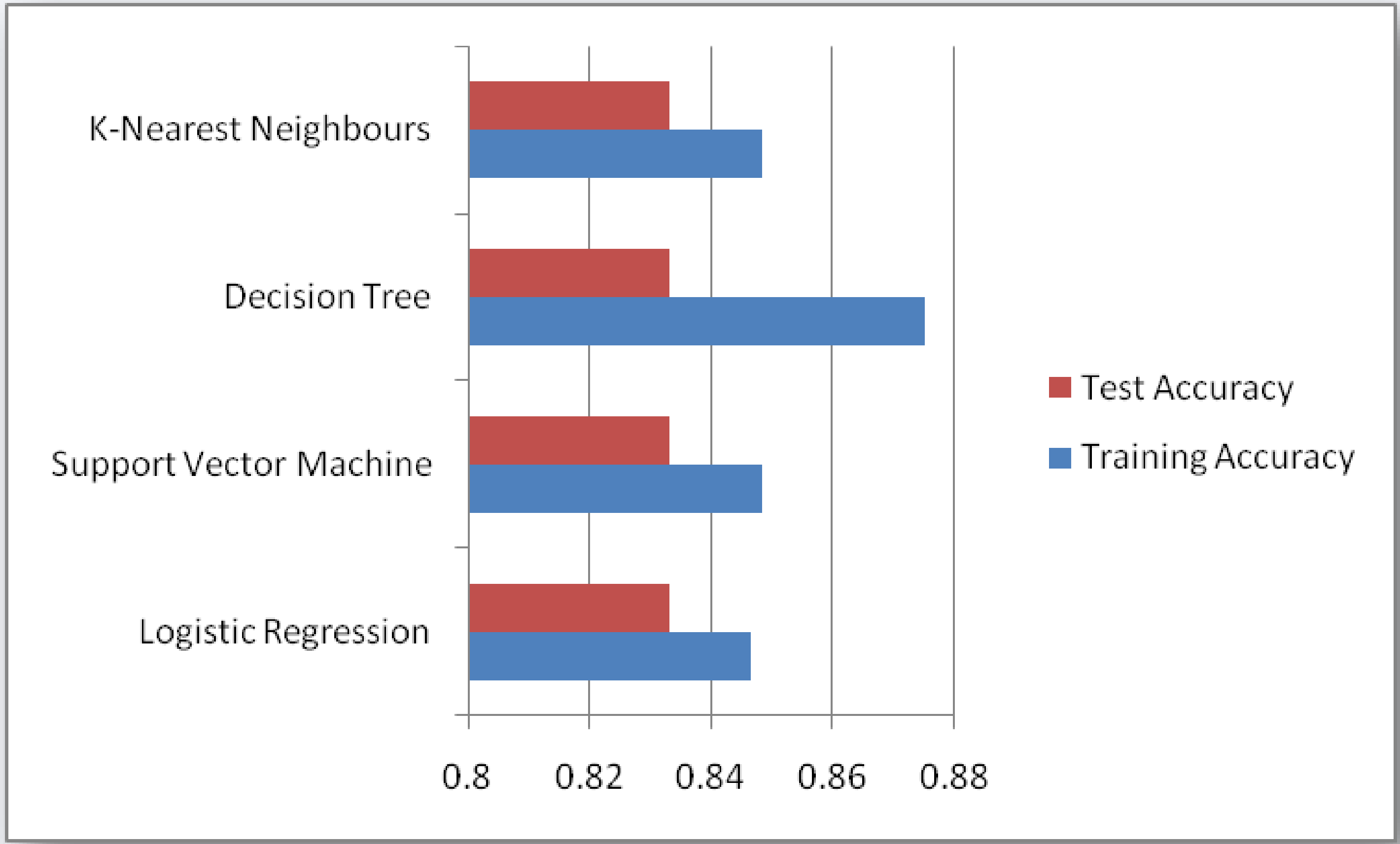


Section 5

Predictive Analysis (Classification)

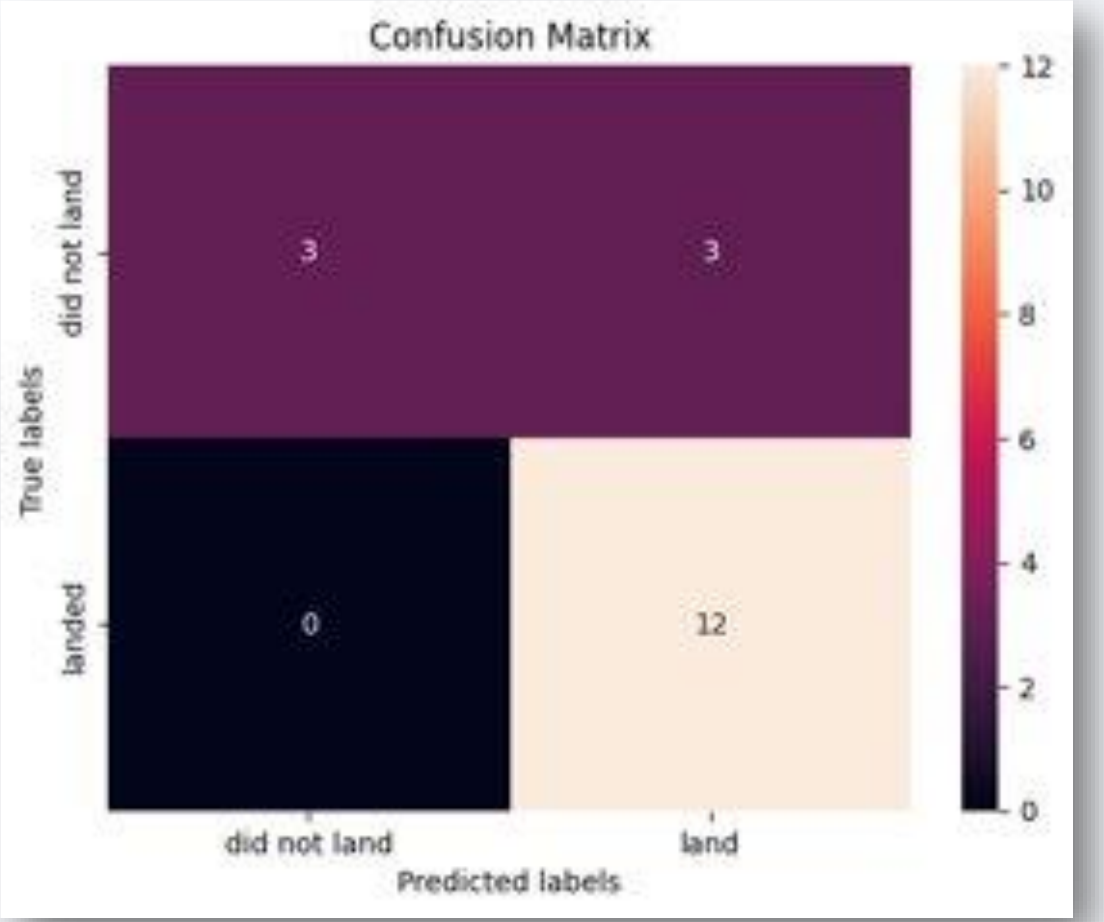
Classification Accuracy

- Visualize the built model accuracy for all built classification models, in a bar chart
- All models had the same accuracy on the test set, however the decision tree performed the best on the training set



Confusion Matrix

- The confusion matrix of the best performing model
- Since all models performed the same on the test data set, this matrix is representative of all of them
- The major problem is false positives, where the model predicts a landing but the flight did not land

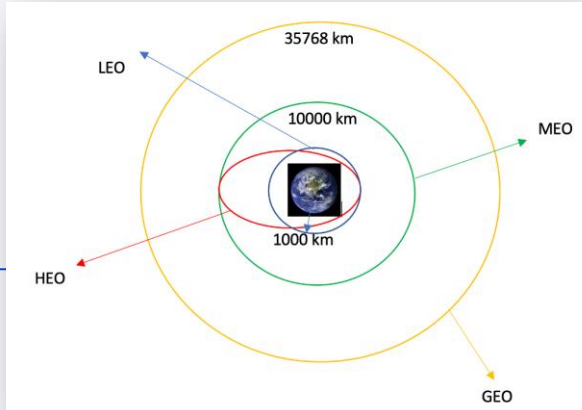


Conclusions

- I determined the cost of each launch to be $62 million, by assuming the same cost as advertised by SpaceX
- I developed a model that can predict launch success with 83% accuracy
- The launch success rate increased over time, likely indicating that SpaceX learned from their mistakes

Appendix

Appendix A: Data Wrangling




```python
# Apply value_counts() on column LaunchSite
df['LaunchSite'].value_counts()
```

```
CCAFS SLC 40    55
KSC LC 39A      22
VAFB SLC 4E     13
Name: LaunchSite, dtype: int64
```

```python
# Apply value_counts on Orbit column
df['Orbit'].value_counts()
```

```
GTO      27
ISS      21
VLEO     14
PO        9
LEO       7
SSO       5
MEO       3
ES-L1     1
HEO       1
SO        1
GEO       1
Name: Orbit, dtype: int64
```

```python
df["Class"].mean()

#df.to_csv("dataset_part_2.csv", index=False)
```

```
0.6666666666666666
```

```python
# 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
```

Successful

Failure

```python
for i,outcome in enumerate(landing_outcomes.keys()):
    print(i,outcome)
```

```
0 True ASDS
1 None None
2 True RTLS
3 False ASDS
4 True Ocean
5 False Ocean
6 None ASDS
7 False RTLS
```

We create a set of outcomes where the second stage did not land successfully:

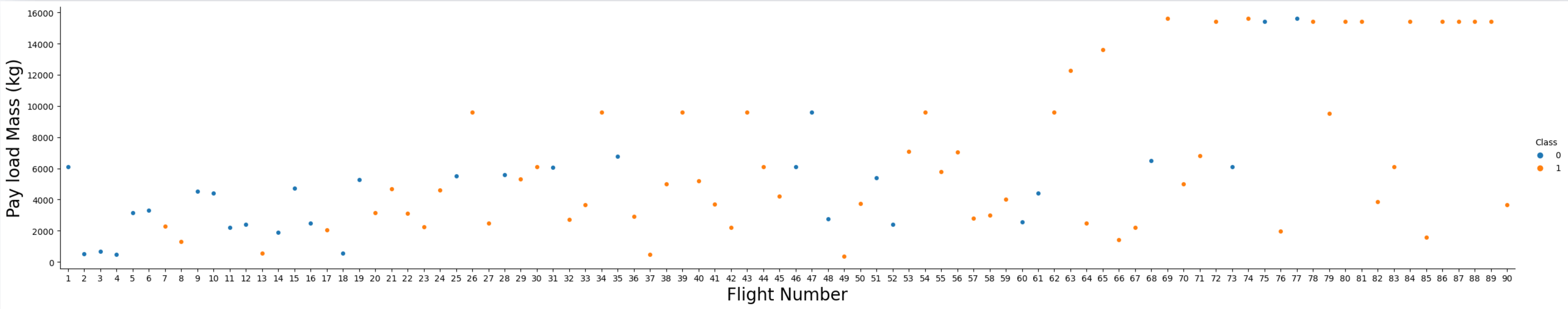
```python
bad_outcomes=set(landing_outcomes.keys()[[1,3,5,6,7]])
bad_outcomes
```

```
{'False ASDS', 'False Ocean', 'False RTLS', 'None ASDS', 'None None'}
```

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

Appendix B: EDA With Data Visualization

- This graph of payload mass vs flight number shows that early flights used lower payload masses, which more recent flights used some of the highest payloads (10000 kg)



Thank you!