



IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

James
October 2025



Outline



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

Executive Summary



- Methodologies

- Data collection
- Data Wrangling
- Exploratory Data Analysis
- Interactive Visuals
- Predictive Analytics

- Results

- EDA
- Geospatial Analysis
- Dashboard
- Classification Model

Introduction



- SpaceX advertises Falcon 9 rocket launches on its website with a cost of 62 million dollars; other providers cost upward of 165 million dollars.
 - Much of the savings is because SpaceX can reuse the first stage.
- This information can be used to bid against SpaceX for a rocket launch.
- We will predict the likelihood the Falcon 9 first stage will land successfully.
- Given this information, we can determine the cost of a launch which can be used to make a competitive bid.



Section 1

Methodology

Methodology



Executive Summary

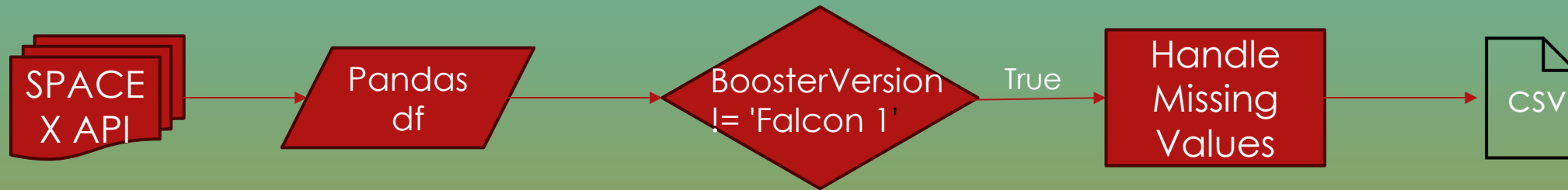
- Data collection methodology:
 - Data was collected using the SpaceX API and webscraping from Wikipedia
- Perform data wrangling
 - Describe how data was processed
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
 - Build, tune, evaluate classification models

Data Collection – SpaceX API



1. Data was collected using the SpaceX REST API
 1. Calls were made to <https://api.spacexdata.com/v4>
 1. Data on Rockets, launchpads, payloads and cores were retrieved
 2. Past data was used as the goal was to predict the cost of a launch.
2. Store in a Pandas Dataframe
3. Sampled to remove instances of Falcon 1 flights
4. Handle missing values of Payload mass with the overall mean
5. Write to CSV

Data Collection – SpaceX API



Github Link: <https://github.com/JPCarroll17/-IBM-Applied-Data-Science-Capstone/blob/main/jupyter-labs-spacex-data-collection-api-v2.ipynb>



Data Collection - Scraping

1. Scrape data using beautifulsoup from Wikipedia: [link](#)
2. Retrieve Past launches table
3. Parse into a dataframe
 1. Variables:
4. Export to csv



Data Collection - Scraping

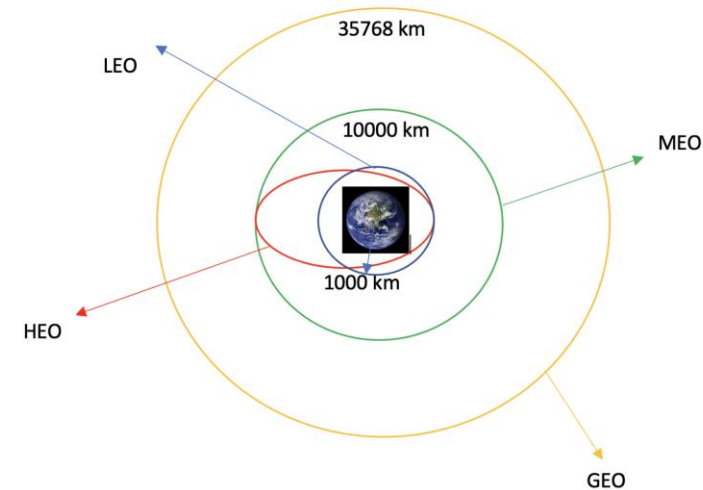


- ▶ Wikipedia tables were scraped using BeautifulSoup from:
https://en.wikipedia.org/wiki/List_of_Falcon_9_and_Falcon_Heavy_launches
- ▶ Github: <https://github.com/JPCarroll17/-IBM-Applied-Data-Science-Capstone/blob/main/jupyter-labs-webscraping.ipynb>

Data Wrangling



- ▶ Using the data from the SpaceX API
 - ▶ Investigate for missing data
 - ▶ Identify data types
 - ▶ Investigate launch sites, orbits and outcome by orbit
 - ▶ Added binomial classifier column "Class" representing
 - ▶ Use this to identify the success rate of landing
 - ▶ Export to CSV
- ▶ Github: <https://github.com/JPCarroll17/-IBM-Applic>
[Capstone/blob/main/labs-jupyter-spacex-Data%20v](https://github.com/JPCarroll17/-IBM-Applic-Capstone/blob/main/labs-jupyter-spacex-Data%20v)



EDA with Data Visualization



- ▶ Looking for relationships between variables, we visualized the following:
 - ▶ Flight number x (Pay load Mass, Launch site, Orbit type)
 - ▶ Pay load Mass x (Launch site, Orbit Type)
 - ▶ Bar chart of success rate by orbit type
 - ▶ Success rate yearly trend
- ▶ Based on these results the following features were chosen for use in our model:
- ▶ One-Hot

```
features = df[['FlightNumber', 'PayloadMass', 'Orbit', 'LaunchSite', 'Flights', 'GridFins', 'Reused', 'Legs', 'LandingPad', 'Block', 'ReusedCount', 'Serial']]
```
- ▶ Additionally, all numeric columns were cast to float64
- ▶ Github: <https://github.com/JPCarroll17/-IBM-Applied-Data-Science-Capstone/blob/main/jupyter-labs-eda-dataviz-v2.ipynb>

EDA with SQL



- ▶ SQL (sqlite) was used to assist in understanding the SpaceX Dataset by:
 - ▶ Identifying unique launch sites, viewing launch sites beginning with 'CCA'
 - ▶ Displaying total payload mass carried by NASA (CRS) boosters
 - ▶ Average Payload for booster F9 v1.1
 - ▶ Identify the first successful landing outcome in ground pad
 - ▶ Identify booster names having success in drone ship and have payload mass between 4000 and 6000
 - ▶ Total number of success and failure mission outcomes
 - ▶ List all booster versions that have carried the max payload mass
 - ▶ List 2015 failure landing outcomes in drone ship along with month, booster version and launch
 - ▶ Rank the count of outcomes between 2010-06-04 and 2017-03-20 in descending order
 - ▶ Github: https://github.com/JPCarroll17/-IBM-Applied-Data-Science-Capstone/blob/main/jupyter-labs-eda-sql-coursera_sqlite.ipynb

Build an Interactive Map with Folium



- ▶ Begin by marking all launch sites on the map
- ▶ At each site, add a pointer to mark successful (green) or unsuccessful (red) launches
- ▶ Using these launch sites, add coordinates with line & distance to nearest coastline/railway/highway/city
- ▶ Notice that the sites tend to be close to railways and coastlines and highways while being in proximity to a city but a comfortable distance away.
- ▶ Github: <https://github.com/JPCarroll17/-IBM-Applied-Data-Science-Capstone/blob/main/lab-jupyter-launch-site-location-v2.ipynb>

Build a Dashboard with Plotly Dash



- ▶ Using plotly dash, generate a dashboard with helpful insights on launch success
- ▶ Allow for user selections:
 - ▶ A dropdown to choose an individual site or a combination of all - this choice will impact both visualizations
 - ▶ A slider filtering based on Payload Range – this choice will impact only the scatterplot (total success launches by site)
- ▶ Generate a pie chart with the following:
 - ▶ When all sites are chosen, each “slice” represents the proportion of successful launches from each site
 - ▶ When a single site is selected, let the pie represent the successful launches vs. failures
- ▶ Generate a scatterplot of payload mass by Class
 - ▶ 0: Failure
 - ▶ 1: Success
- ▶ Github: https://github.com/JPCarroll17/-IBM-Applied-Data-Science-Capstone/blob/main/spacex_dash_app.py

Predictive Analysis (Classification)



- ▶ To complete the predictive analysis, StandardScaler was applied to our data and data was split into training and testing datasets.
- ▶ A GridSearch object was created and fit to the data for the following methods
 - ▶ Logistic regression
 - ▶ SVM
 - ▶ Tree
 - ▶ knn
- ▶ The methods' scores were compared and the model with the highest score was chosen
- ▶ GitHub: <https://github.com/JPCarroll17/-IBM-Applied-Data-Science-Capstone/blob/main/SpaceX-Machine-Learning-Prediction-Part-5-v1.ipynb>



Results



Results will be broken into four groups:

1. Insights drawn from EDA
2. Launch Sites Proximities Analysis
3. Plotly – Dash
4. Predictive Analysis



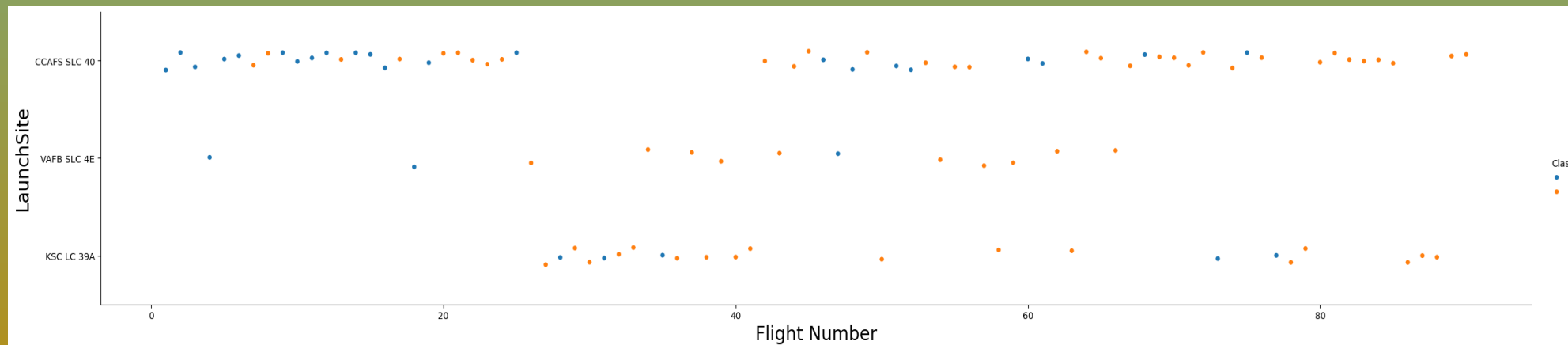
Section
2

Insights drawn from EDA

Flight Number vs. Launch Site



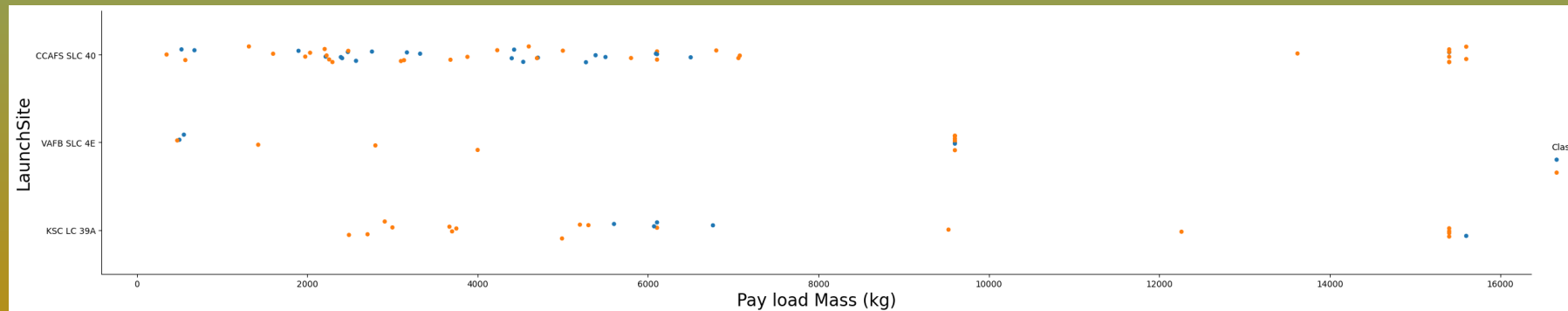
- ▶ Success rate for launches increases with time
- ▶ Initial flights were from the CCAFS SLC 40 site, with a couple from VAFB SLC 4E
- ▶ CCAFS SLC 40 was the most common site overall and VAFB SLC 4E having some spread throughout
- ▶ Coming on later was SKC LC 39A and there was a period where most flights left from here.



Payload vs. Launch Site



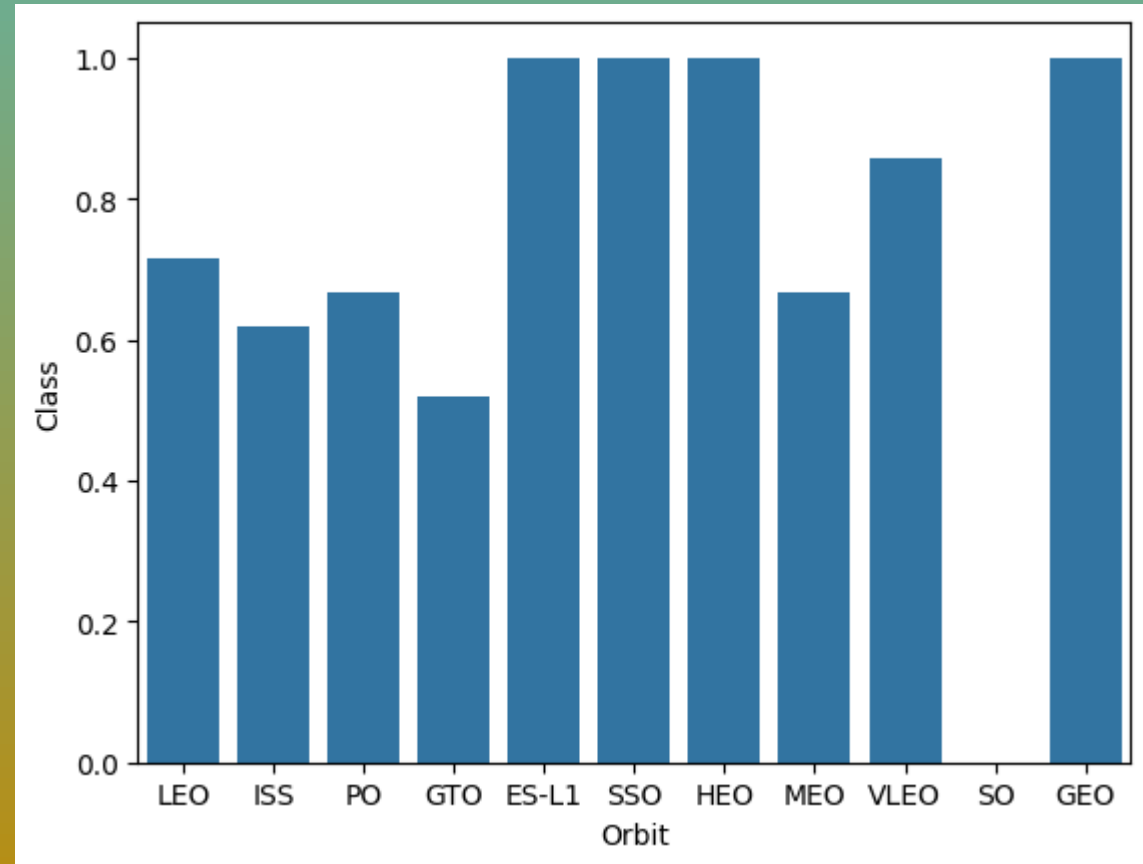
- ▶ Many lighter payload launches failed, heavier payloads ($>8,000$) were successful most of the time but were less frequent.
- ▶ Two sites (CCAFS SLC 40 and KSC LC 39A) have a max payload $\sim 15,500$
- ▶ VAFB SLC 4E has a max payload $\sim 9,500$



Success Rate vs. Orbit Type



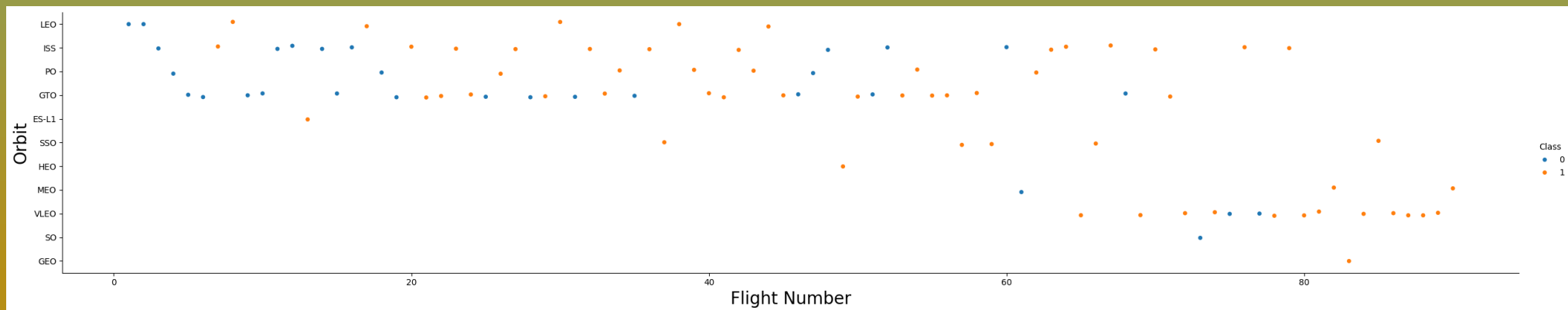
- ▶ Orbits show greater than .5 success rate for all but SO orbit which shows no successful launches
- ▶ ES-L, SSO, HEO and GEO all have successful launches for each launch



Flight Number vs. Orbit Type



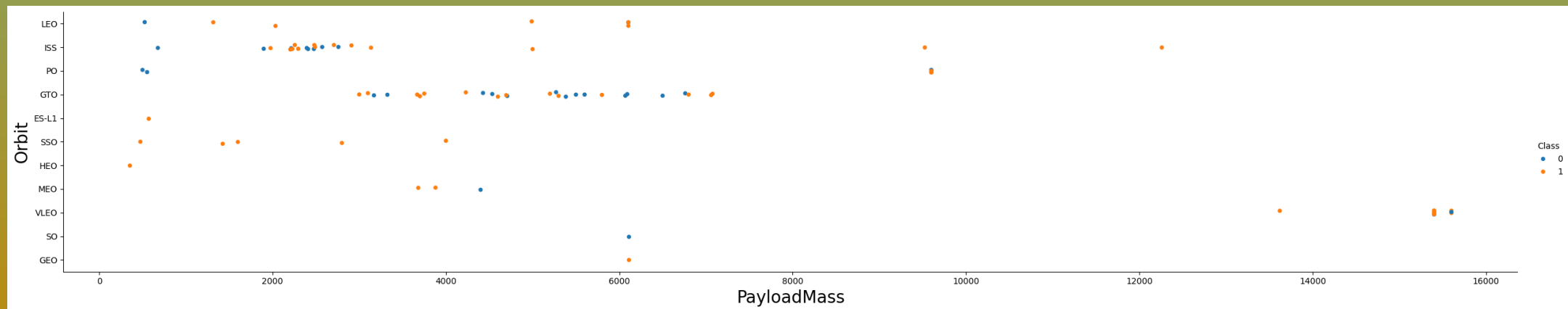
- ▶ Referencing the previous visual, we see there was only one SO orbit which failed.
- ▶ ES-L1, SSO, HEO, MEO, SO and GEO all have 5 or fewer launches.
- ▶ As time goes on the success rate for launches of orbit types tends to increase



Payload vs. Orbit Type



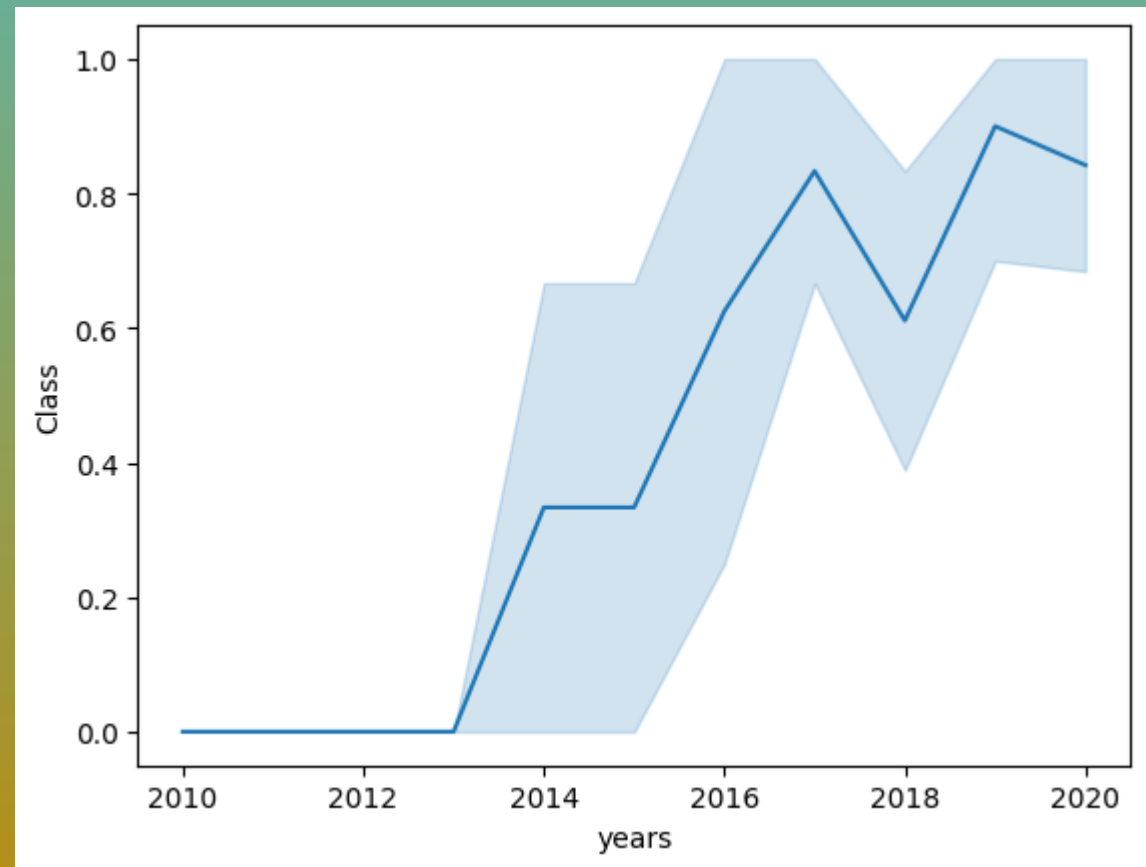
- ▶ ISS and LEO orbit types show increased likelihood of success at higher payload mass, but the number of launches is low.
- ▶ Overall payload mass does not seem to have much of a relationship with orbit type



Launch Success Yearly Trend



- ▶ For 2010-2013 launch success was 0
- ▶ 2014 shows an increase in likelihood for launch success which continues until 2017 where launch success begins to stabilize
- ▶ There are dips in 2018 and 2020 but these do not seem significant



All Launch Site Names



- ▶ By using the distinct clause you can obtain the unique values for “Launch_Site”
- ▶ There are 4 unique values (shown below)

```
%sql select distinct Launch_Site from SPACE_TABLE
```

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'
- ▶ Using the limit operator returns only 5 records
- ▶ Using the where clause with the like operator in conjunction with the % wildcard allows us to pull records which begin with 'CCA'
- ▶ All 5 names obtained here were the same

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

Launch_Site

CCAFS LC-40

CCAFS LC-40

CCAFS LC-40

CCAFS LC-40

CCAFS LC-40

Total Payload Mass



- ▶ Calculate the total payload carried by boosters from NASA (CRS)
- ▶ To obtain the sum of a value the keyword 'sum' can be used
- ▶ Use the where clause to keep only 'NASA (CRS)' records

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

total_Payload
45596

Average Payload Mass by F9 v1.1



- ▶ Calculate the average payload mass carried by booster version F9 v1.1
- ▶ Similar to the previous slide, avg can be used to obtain an average value while the where clause can be used as a filter

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

total_Payload

2928.4

First Successful Ground Landing Date

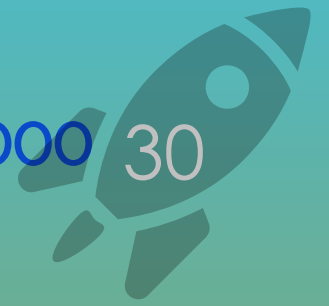


- ▶ Find the dates of the first successful landing outcome on ground pad
- ▶ You can use the Min function on the date variable to find the earliest date
- ▶ The where clause can be used to filter only successful launches

```
%sql select min(Date) from SPACEXTABLE where Landing_Outcome='Success (ground pad)'
```


min(Date)
2015-12-22

Successful Drone Ship Landing with Payload between 4000 and 6000 30



- ▶ List the names of boosters which have successfully landed on drone ship and had payload mass greater than 4000 but less than 6000
- ▶ Here we can use the distinct clause to obtain only the unique values for booster version
- ▶ But we must also filter using the between keyword in the where clause to obtain only records where Payload Mass is between 4000 and 6000
- ▶ We must also remember to use only 'Success (drone ship)' records

```
%sql select distinct Booster_Version from SPACEXTABLE where Landing_Outcome='Success (drone ship)' and PAYLOAD_MASS_KG_ between 4000 and 6000
```



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

Total Number of Successful and Failure Mission Outcomes 31



- ▶ Calculate the total number of successful and failure mission outcomes
- ▶ Using the count function and the group by clause you can obtain this information
- ▶ There is one 'Success' record which is not grouped with the others, this should be fixed before analysis

```
%sql select Mission_Outcome, count(Mission_Outcome) from SPACEXTABLE group by Mission_Outcome
```


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
- ▶ Using a subquery in the where clause, you can obtain the maximum payload and use it in the statement.

```
%sql select booster_version, PAYLOAD_MASS_KG_ from SPACEXTABLE where PAYLOAD_MASS_KG_ = (select max(PAYLOAD_MASS_KG_) from SPACEXTABLE)
```




Booster_Version	PAYLOAD_MASS_KG_
F9 B5 B1048.4	15600
F9 B5 B1049.4	15600
F9 B5 B1051.3	15600
F9 B5 B1056.4	15600
F9 B5 B1048.5	15600
F9 B5 B1051.4	15600
F9 B5 B1049.5	15600
F9 B5 B1060.2	15600
F9 B5 B1058.3	15600
F9 B5 B1051.6	15600
F9 B5 B1060.3	15600
F9 B5 B1049.7	15600

2015 Launch Records



- ▶ List the failed landing_outcomes in drone ship, their booster versions, and launch site names for in year 2015
- ▶ Use substr function to obtain the year and month, and filter on the year 2015

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




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

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



- ▶ Rank the count of landing outcomes (such as Failure (drone ship) or Success (ground pad) between the date 2010-06-04 and 2017-03-20, in descending order
- ▶ Here we need to group by the landing outcome and order by count, but use desc to sort in descending order

```
%sql select Landing_Outcome, count(Landing_Outcome) as Landing_Count from SPACEXTABLE where Date between '2010-06-04' and '2017-03-20' group by Landing_Outcome order by count(Landing_Outcome) desc
```



Landing_Outcome	Landing_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



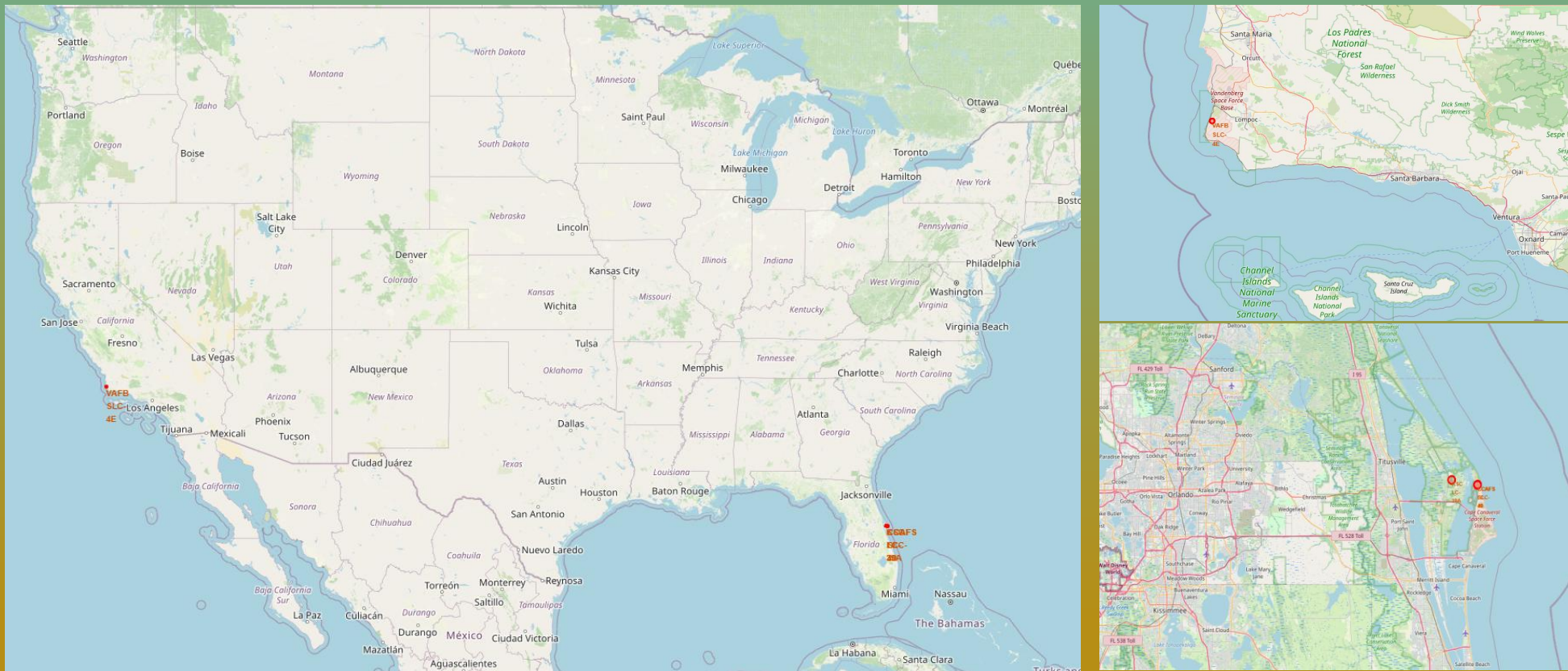
Section 3

Launch Sites Proximities Analysis

Falcon 9 Map with SpaceX launch sites



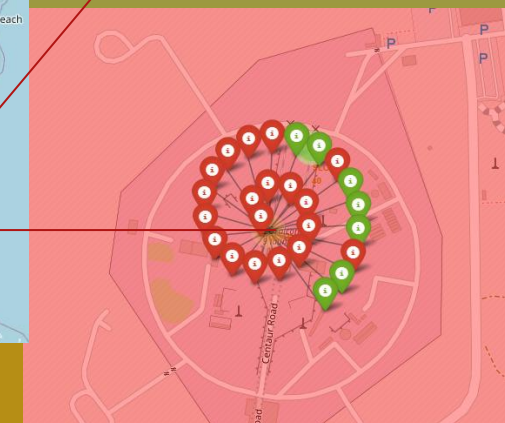
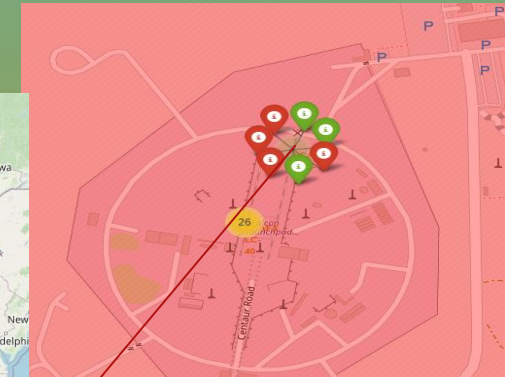
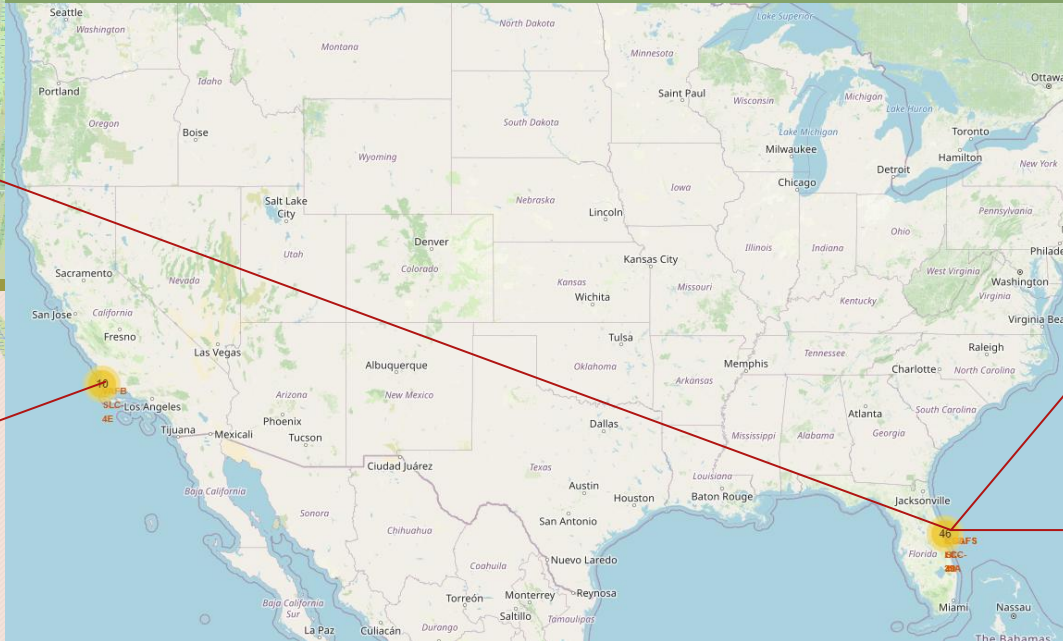
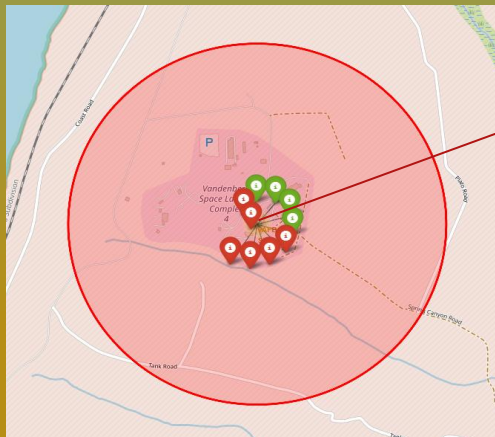
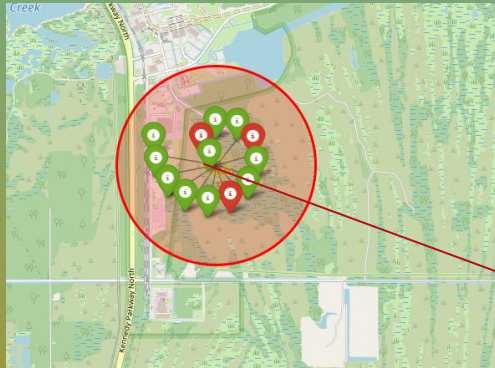
► All Launch sites are in the US, one in CA (top right) three in FL (bottom right)



Success and Failure of launches at each site



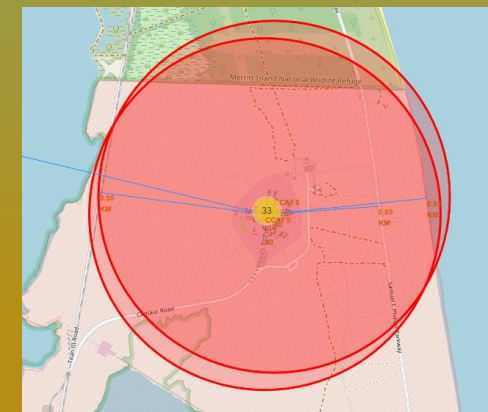
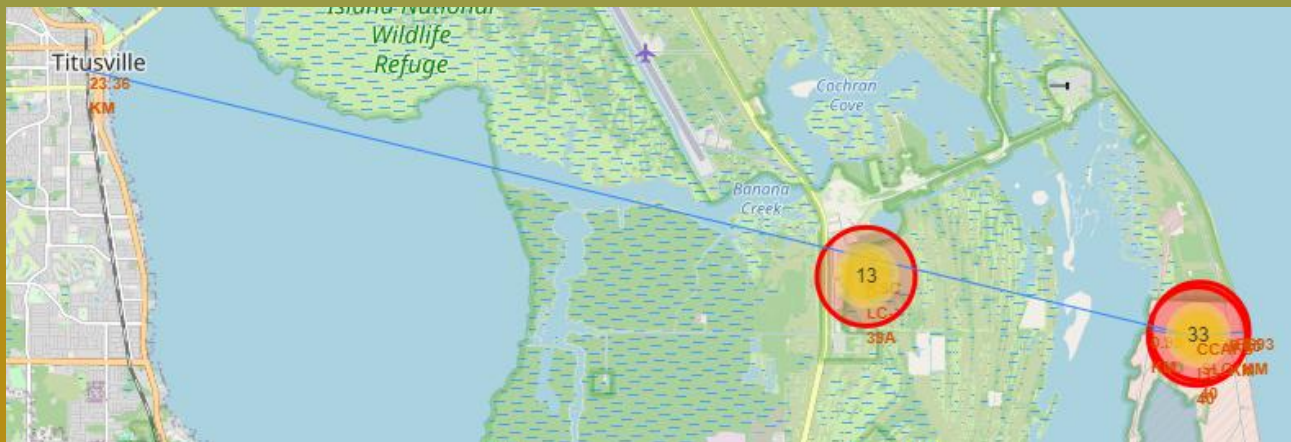
- ▶ Successful launches are in green unsuccessful are in red
- ▶ KSC LC-39A (top left) has the best success ratio



Proximity of Points of Interest CCAFS SLC-40



- ▶ Using CCAFS SLC-40 there are many points of interest nearby to the site
- ▶ Within 1 KM is shoreline, highway and railway
- ▶ Under 25 KM away is A city
- ▶ Not shown: There are also no less than three airports within 40 KM





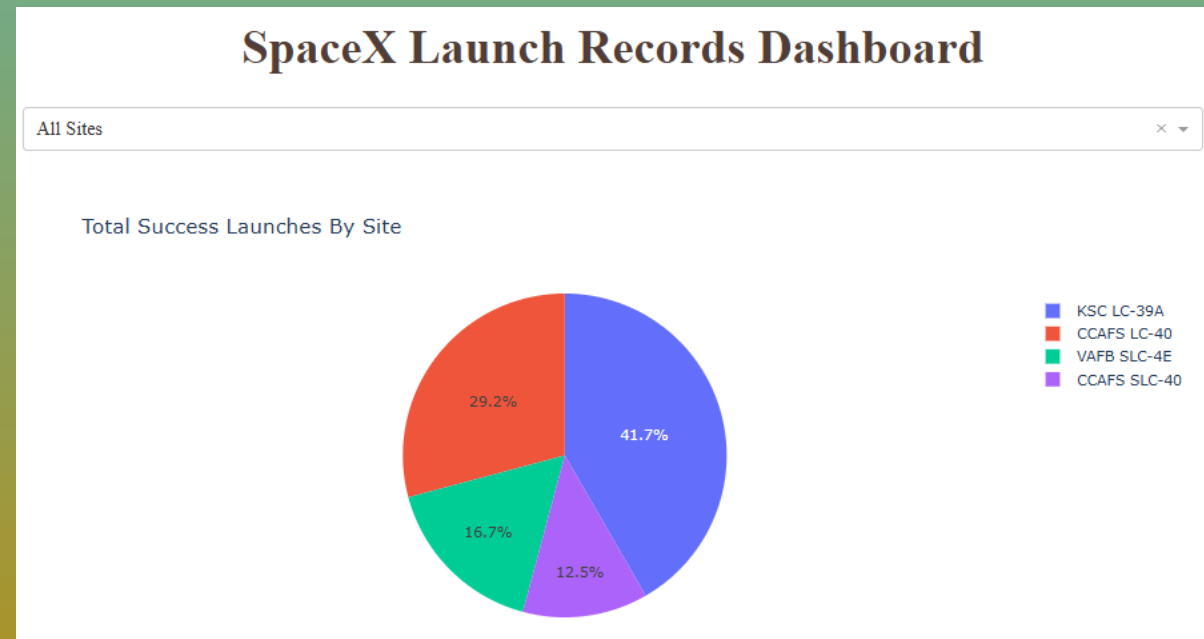
Section 4

Build a Dashboard with Plotly Dash

Dashboard: Launch Success for all sites



- ▶ Proportion of successful launches is not evenly distributed
- ▶ KSC LC-39A has the most successful launches
- ▶ CCAFS SLC-40 has the fewest successful launches



Dashboard: Launch Success Ratio



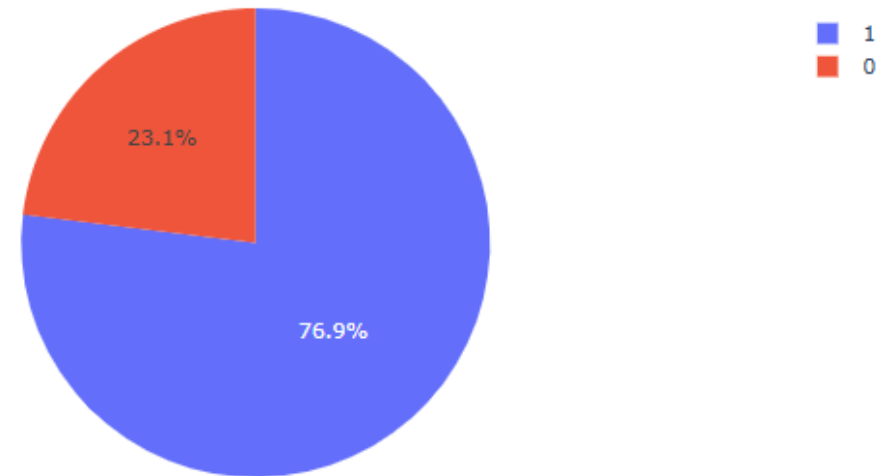
▶ Not surprisingly, KSC LC-39A with the most successful launches also has the highest success ratio at 76.9%

SpaceX Launch Records Dashboard

KSC LC-39A



Pie Chart

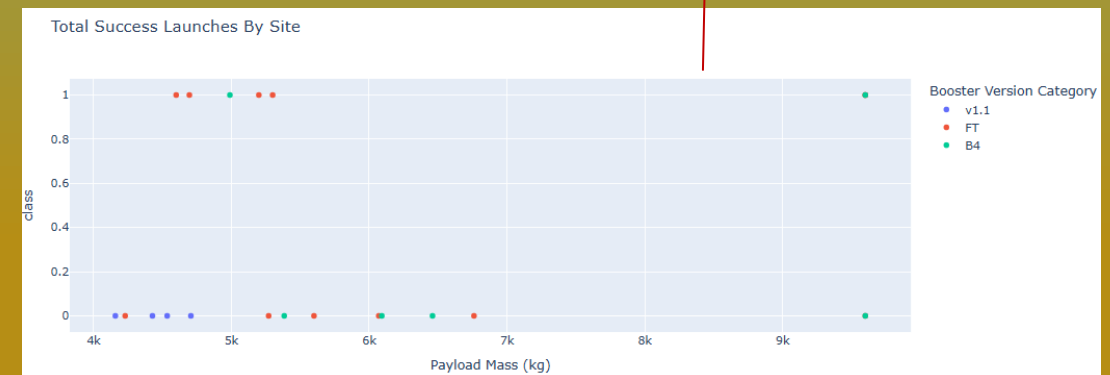


Dashboard: Payload vs. Launch Outcome all sites



► The payload range has a split around 4,000 kg, we will use this to separate the plot into low and high payload

- Notice the lower payload has a higher success rate even more so on the higher end of this range
- v1.0 and B5 are not included in the high payload



The background of the slide is an abstract composition. The left half is a solid blue field. The right half features a series of concentric, curved white and light blue lines that create a sense of depth and motion, resembling a tunnel or a stylized architectural structure. A solid red rectangle is positioned in the upper right corner.

Section 5

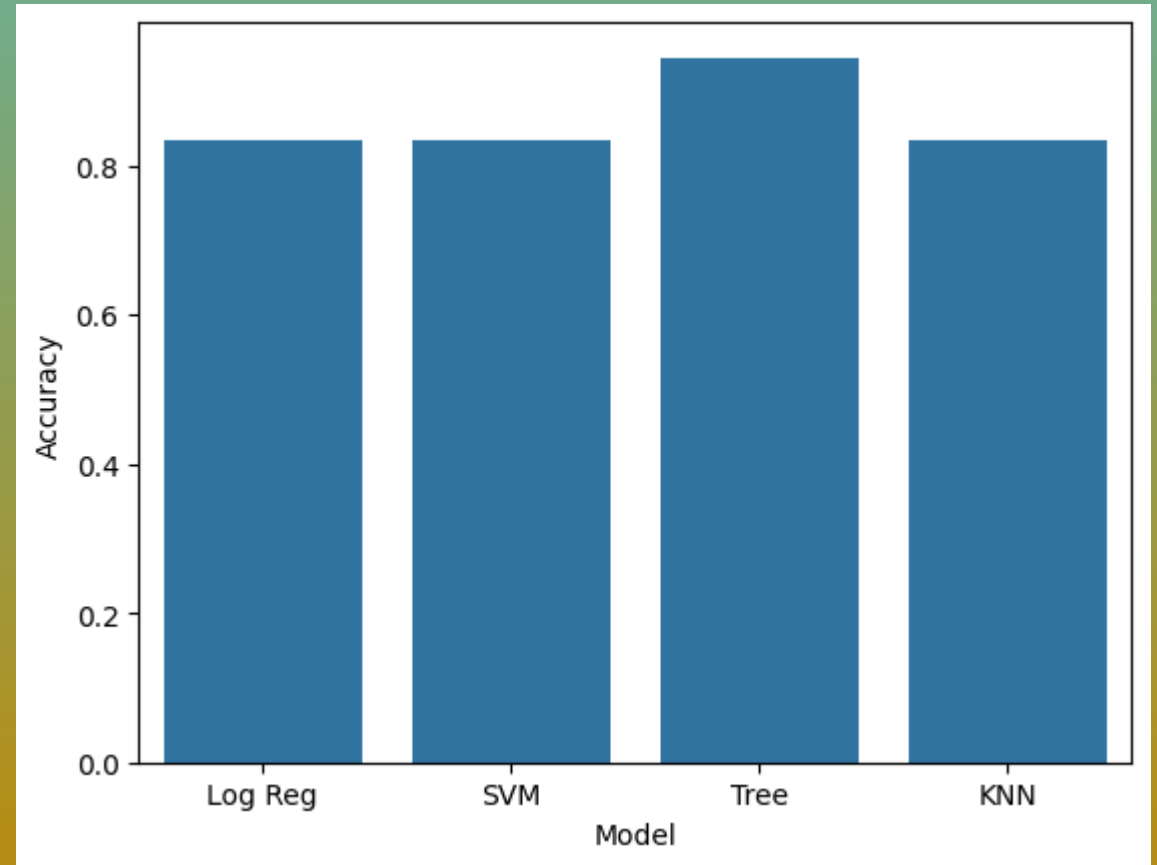
Predictive Analysis (Classification)

Classification Accuracy



- ▶ The Tree method has the highest accuracy at 94.4

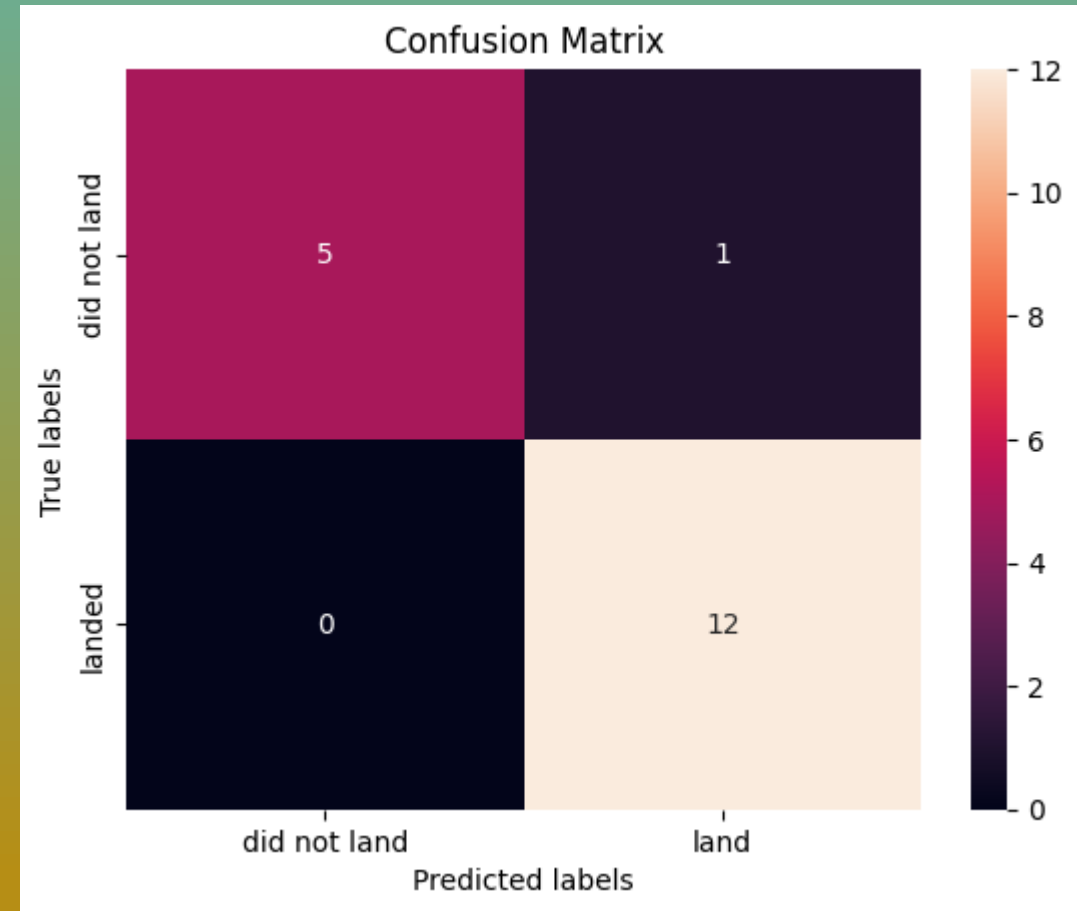
Model	Accuracy
Log Reg	0.833333
SVM	0.833333
Tree	0.944444
KNN	0.833333



Confusion Matrix



- ▶ The confusion matrix for the tree method shows one incorrect classification where a landing was predicted but the landing failed
- ▶ 12 records were predicted to land and did
- ▶ 5 records were predicted to fail and did



Conclusions



- ▶ Using ML methods landing success or failure can be predicted by using decision trees
- ▶ Certain launch sites have a higher probability of success
- ▶ The success rate drops with higher payloads
- ▶ After a few years, the likelihood of a successful launch has increased significantly from 0 in 2010 to over 80% by 2020

Appendix



- ▶ Include any relevant assets like Python code snippets, SQL queries, charts, Notebook outputs, or data sets that you may have created during this project

Thank you!

