IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

March 15, 2024

Outline

Executive Summary

- Summary of methodologies
    - Data Collection – SpaceX API
    - Data Collection – Scraping
    - Data Wrangling
    - EDA with Data Visualization
    - EDA with SQL
    - Build an Interactive Map with Folium
    - Build a Dashboard with Plotly Dash
    - Predictive Analysis (Classification)
- Summary of all results
    - Exploratory data analysis results
    - Interactive analytics demo in screenshots
    - Predictive analysis results

Introduction

- Project background and context

The commercial space industry has entered an exciting era, with companies like Virgin Galactic, Rocket Lab, Blue Origin, and SpaceX revolutionizing space travel. Among these, SpaceX stands out as a trailblazer, achieving remarkable milestones such as sending spacecraft to the International Space Station (ISS), deploying the Starlink satellite internet constellation, and conducting manned missions to space. One key factor contributing to SpaceX's success is its ability to make rocket launches more affordable.

- SpaceX advertises Falcon 9 rocket launches on its website with a cost of **62 million dollars**
- Other providers cost upwards of **165 million dollars**

Much of the savings is because SpaceX can reuse the first stage. Spaces X's Falcon 9 launch like regular rockets. We will be utilizing diagrams from Forest Katsch, at zlsadesign.com, A 3D artist and software engineer. He makes infographics on spaceflight and spacecraft art. He also makes software. The payload is enclosed in the fairings. Unlike other rocket providers, SpaceX's Falcon 9 Can recover the first stage. Other times, Space X will sacrifice the first stage due to the mission parameters like payload, orbit, and customer. We will determine if company Space Y, would like to compete with SpaceX.

Problems you want to find answers

- If we can determine if the first stage will land, we can determine the cost of launch
  - What factors determine if the first stage will land successfully?
- Based off of parameters, can the first stage be reused
  - What parameters will allow for the first stage to be reused?
- Using machine learning, predict if SpaceX will reuse the first stage
  - What predictions can be made on the different conditions for successful landing of the rocket?

Section 1

Methodology

Methodology

Executive Summary

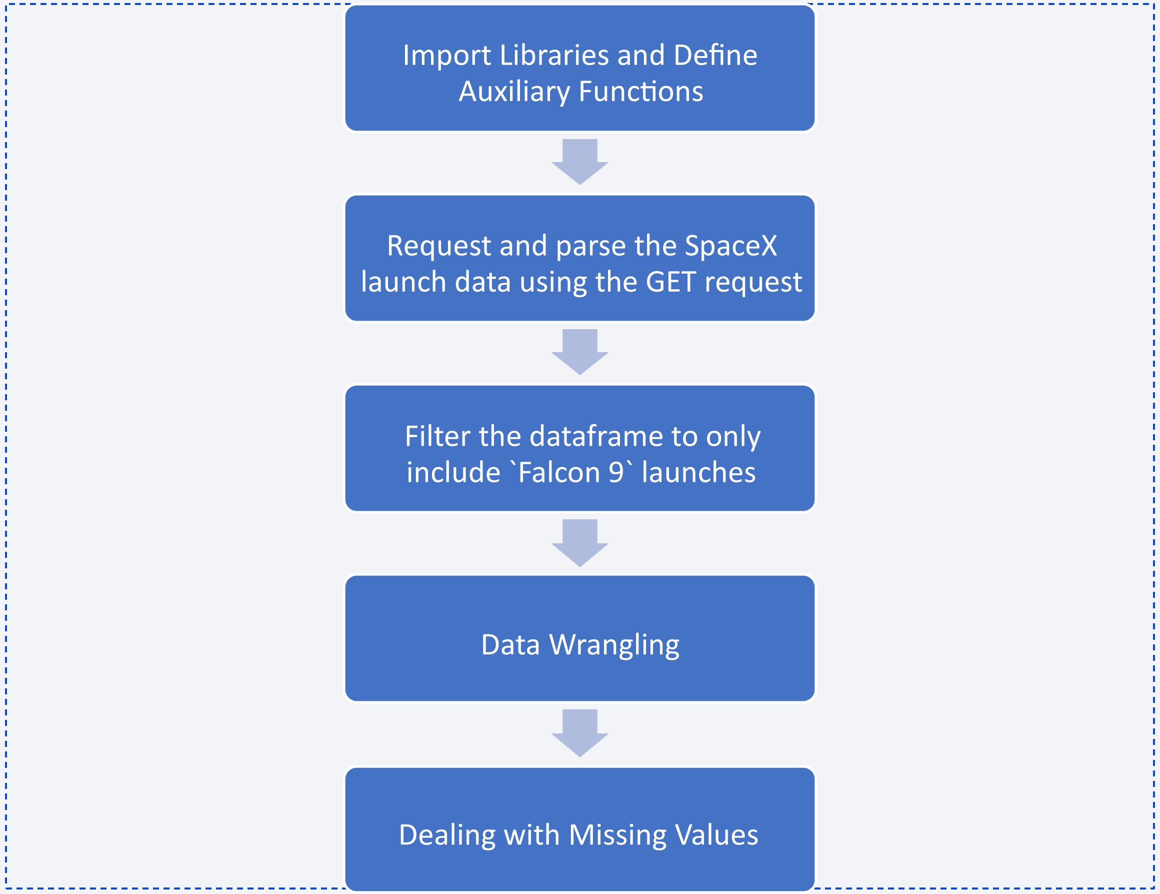
- Data collection methodology:
  - Data was collected from SpaceX API and web-scraping from sites like Wikipedia
- Perform data wrangling
  - Data was collected and cleaned using various coding practices including the use of Python's Pandas library and BeautifulSoup
- Perform exploratory data analysis (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

- Describe how data sets were collected.
  - Data collection was done using get requests to SpaceX API
  - Decoded the response as JSON and transformed it into a Pandas dataframe utilizing the function json_normalize()
  - Evaluated the data to discover any missing values within the dataframe and filled in the null values
  - Additionally, we included webscraping utilizing the Python library BeautifulSoup
  - From Wikipedia, we extracted the information to then convert the data into a dataframe for analysis

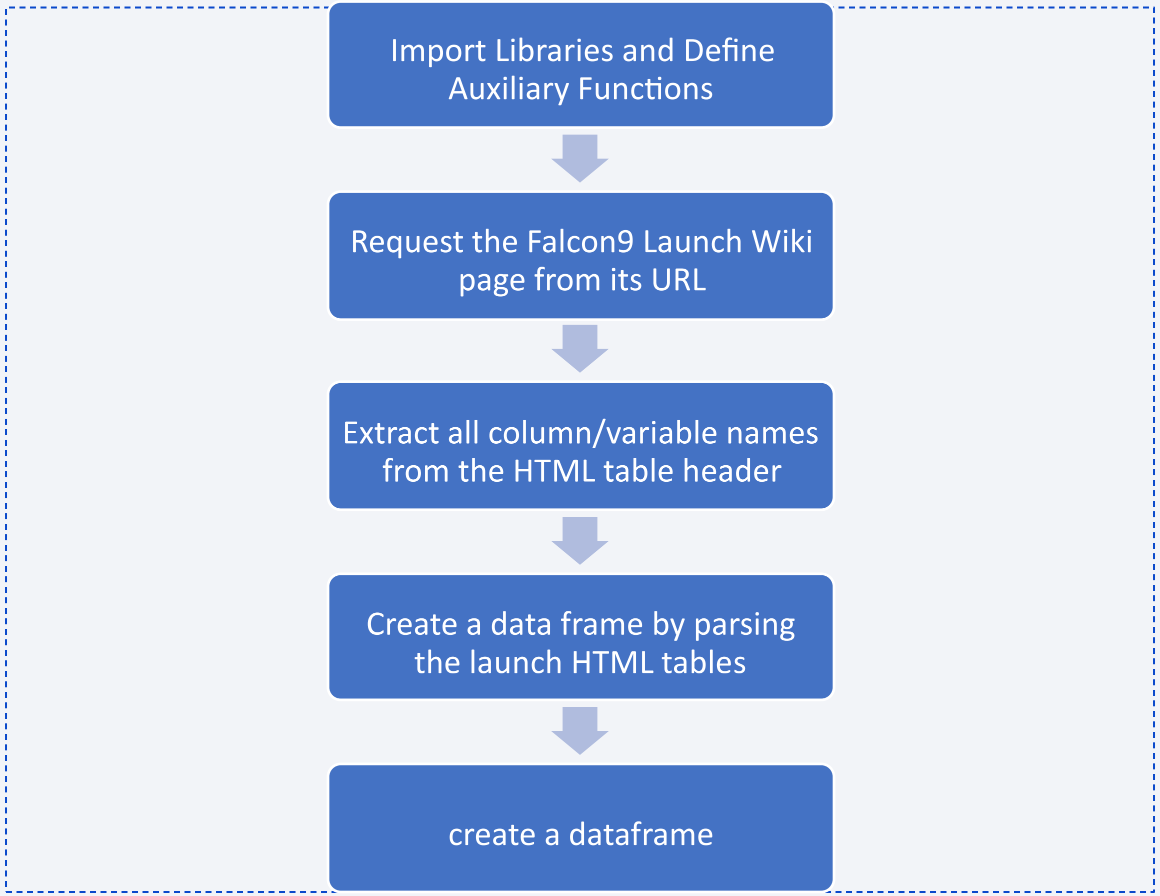
Data Collection – SpaceX API

- Data collection was done using get requests to SpaceX API.
- Decoded the response as JSON and transformed it into a Pandas dataframe utilizing the function json_normalize().
- Evaluated the data to discover any missing values within the dataframe and filled in the null values



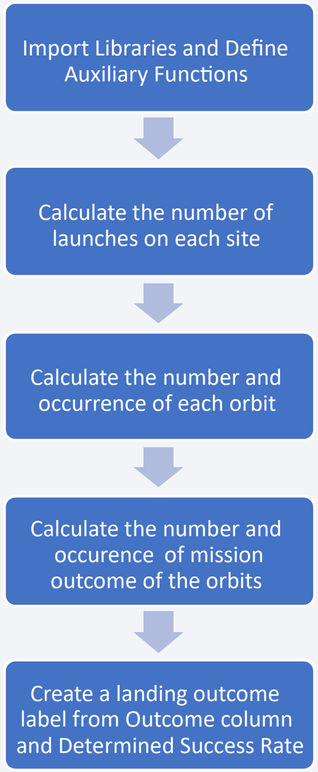
Data Collection - Scraping

- Applied Web Scraping to obtain data in regards to Falcon9 launch records utilizing BeautifulSoup
- Parsed the information into a Pandas dataframe



Data Wrangling

- We performed preliminary data analysis which helped determine training labels\
- Calculations were made to determine the number of launches that occurred on each launch site
- Additionally, calculated the occurrence, and mission outcomes of each orbit
- Determined success rate



EDA with Data Visualization

- Explored the data using Visualization tools. Some of the charts plotted were:
  - FlightNumbervs. PayloadMass
  - Flight Number and Launch Site
  - Payload and Launch Site
  - Success Rate per Orbit
  - FlightNumberand Orbit type
  - Payload and Orbit type
  - Yearly Evolution of Success Rate

  These charts were used as they conveyed a clear answer to the information asked for. Each one successfully tells a story about the comparison between two variables.

EDA with SQL

- Explored the database to extract pertinent information and create applicable charts. The charts that were plotted include:
    - names of the unique launch sites in the space mission including 5 records where launch sites begin with the string 'CCA'
    - total payload mass carried by boosters launched by NASA (CRS) also average payload mass carried by booster version F9 v1.1
    - date when the first succesful landing outcome in ground pad was acheived.
    - names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000
    - total number of successful and failure mission outcomes
    - names of the booster_versions which have carried the maximum payload mass. records which will display the month names, failure landing_outcomes in drone ship ,booster versions, launch_site for the months in year 2015
    - 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

Build an Interactive Map with Folium

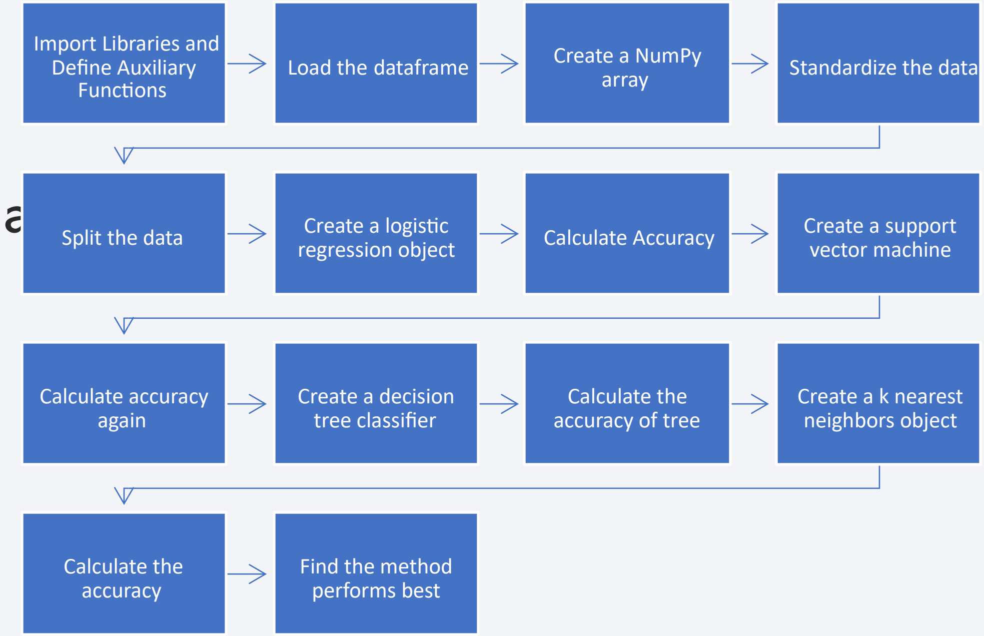
- We wanted to add Markers to all launch site locations on a map. From there we included markers for successful and failed launch sites. Which then we included relative distances with these points on a map.
- These different markings were included to provide relatively easy understanding of where launches occurred in comparison to the coast line nearest each launch site.

Build a Dashboard with Plotly Dash

- A Plotly Dashboard was created to provide visualizations to help summarize the total launches by certain sites using a pie graph
- Additionally, a scatter plot was created to represent Payload and Launch success for all sites.

Predictive Analysis (Classification)

- Data was loaded using Pandas andNumpy. Once loaded, the data was split into training and testing sets for better analysis. Several machine learning models were created to use accuracy as a metric and we were able to improve this model using feature engineering.
- In the end, the Decision tree model provided the highest training accuracy amongst our models.



Results

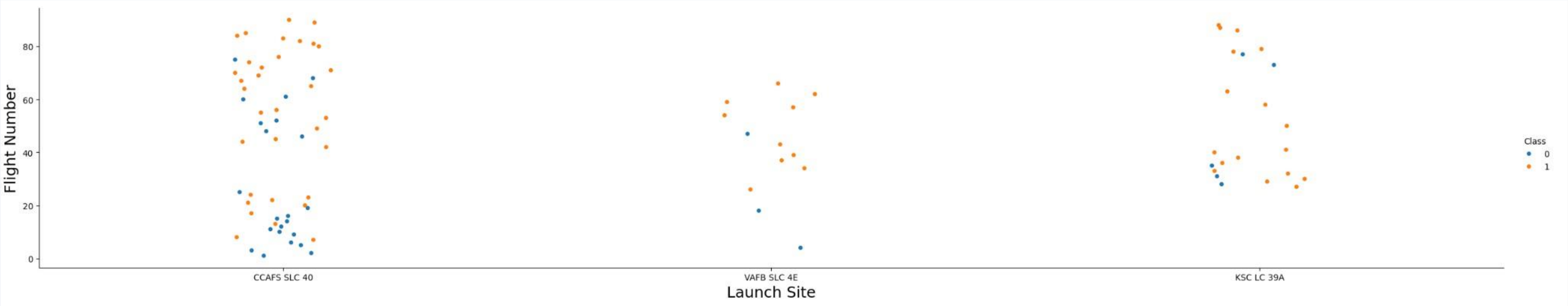
- Exploratory data analysis results
- Based solely on the accuracy score, the decision tree is the method that performs best, with an accuracy score of 0.8889.
- Interactive analytics demo in screenshots
- Predictive analysis results

Section 2

Insights drawn from EDA

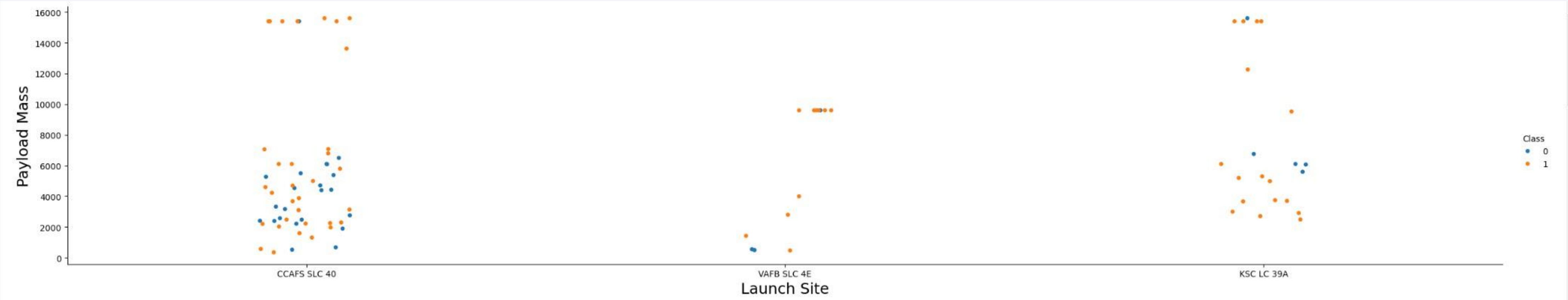
Flight Number vs. Launch Site

- We find that the larger amount of flights, at a particular launch site, the greater the success rate. CCAFS-SLC 40 has the maximum success rate



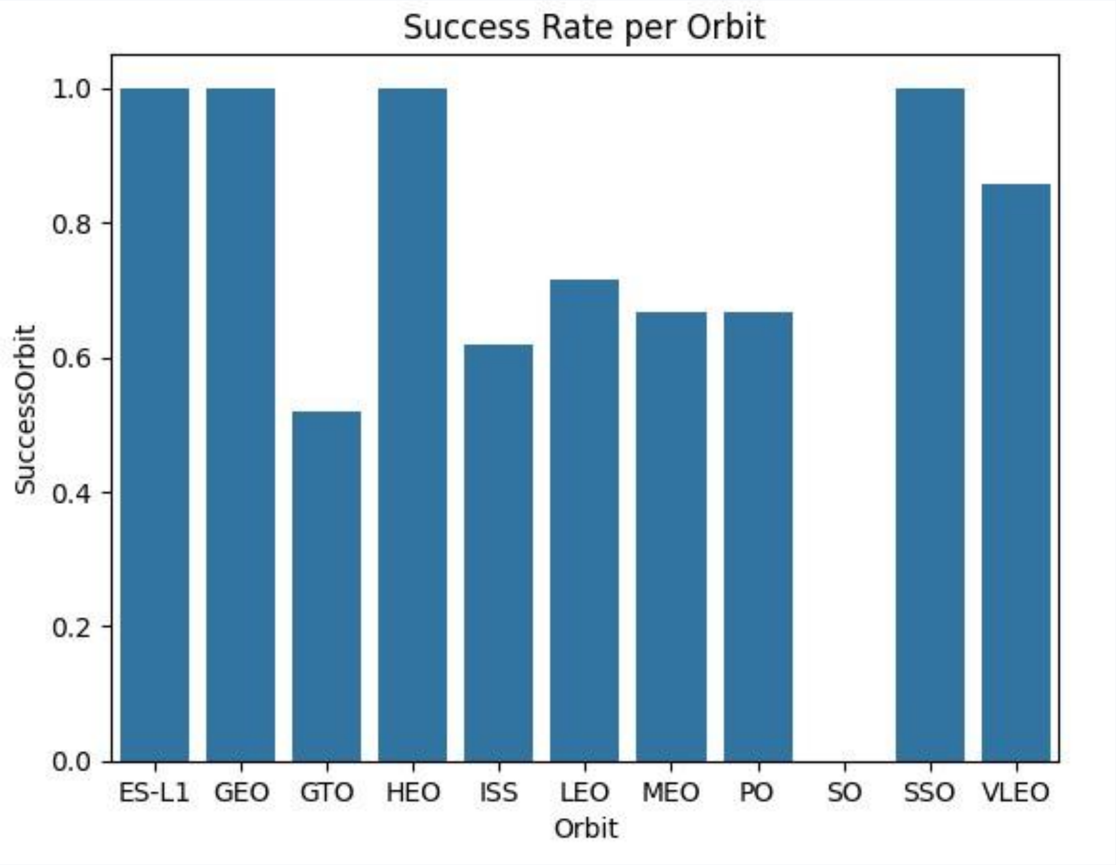
Payload vs. Launch Site

- VAFB SLC 4E launch site has no rockets of a heavy payload mass greater than 10000
- CCAFS SLC 40 has high success rate for high payload mass around 15000



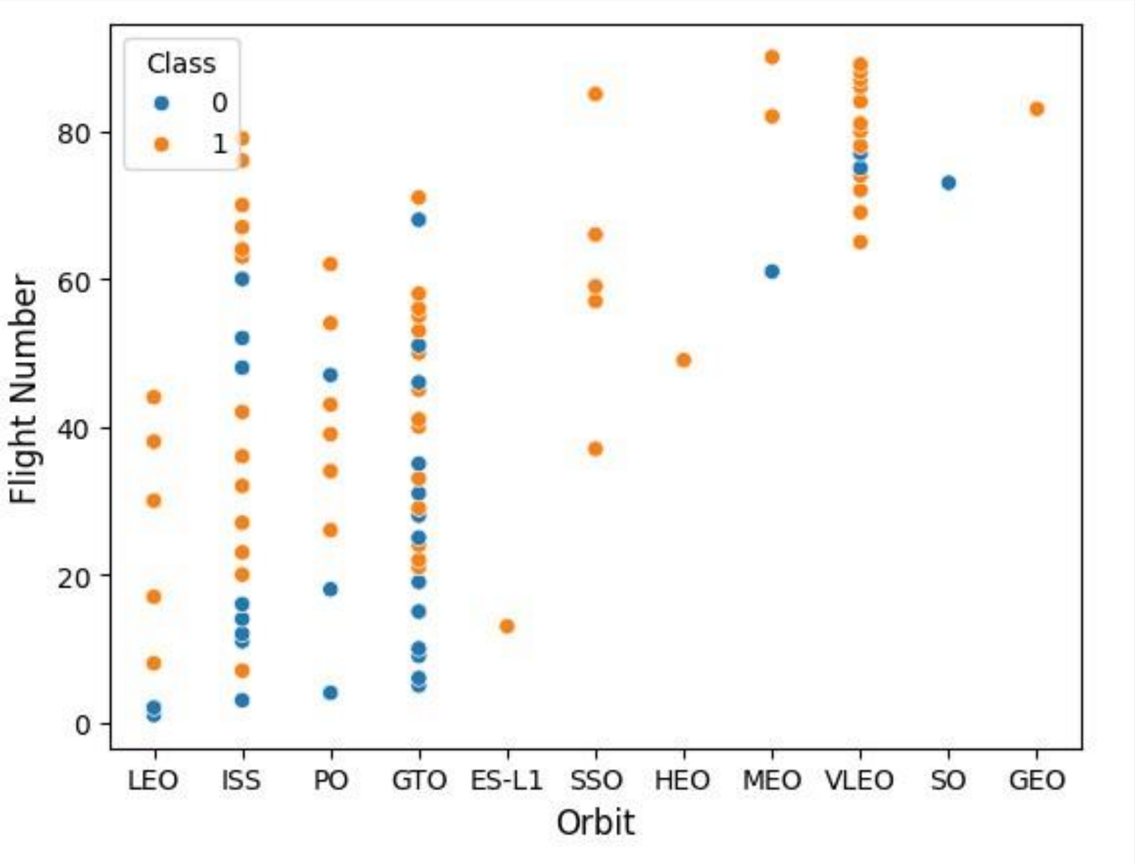
Success Rate vs. Orbit Type

- Within the bargraph, ES-L1, GEO, HEO and SSO have 100% success rate while SO has a 0% success rate



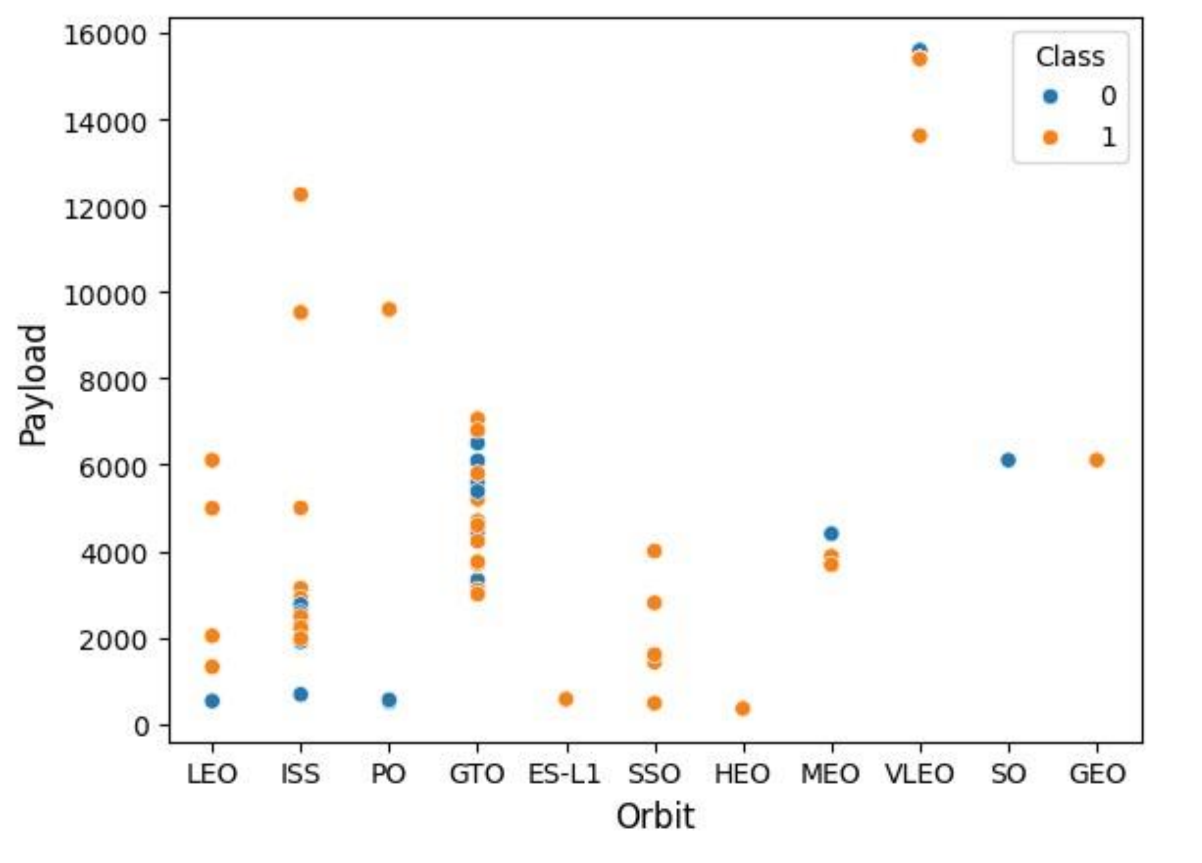
Flight Number vs. Orbit Type

- Here there are very few flights from the GEO orbit. However all the flights are successful. This is followed by ES-L1, SSO and HEO showing that they are 100% successful



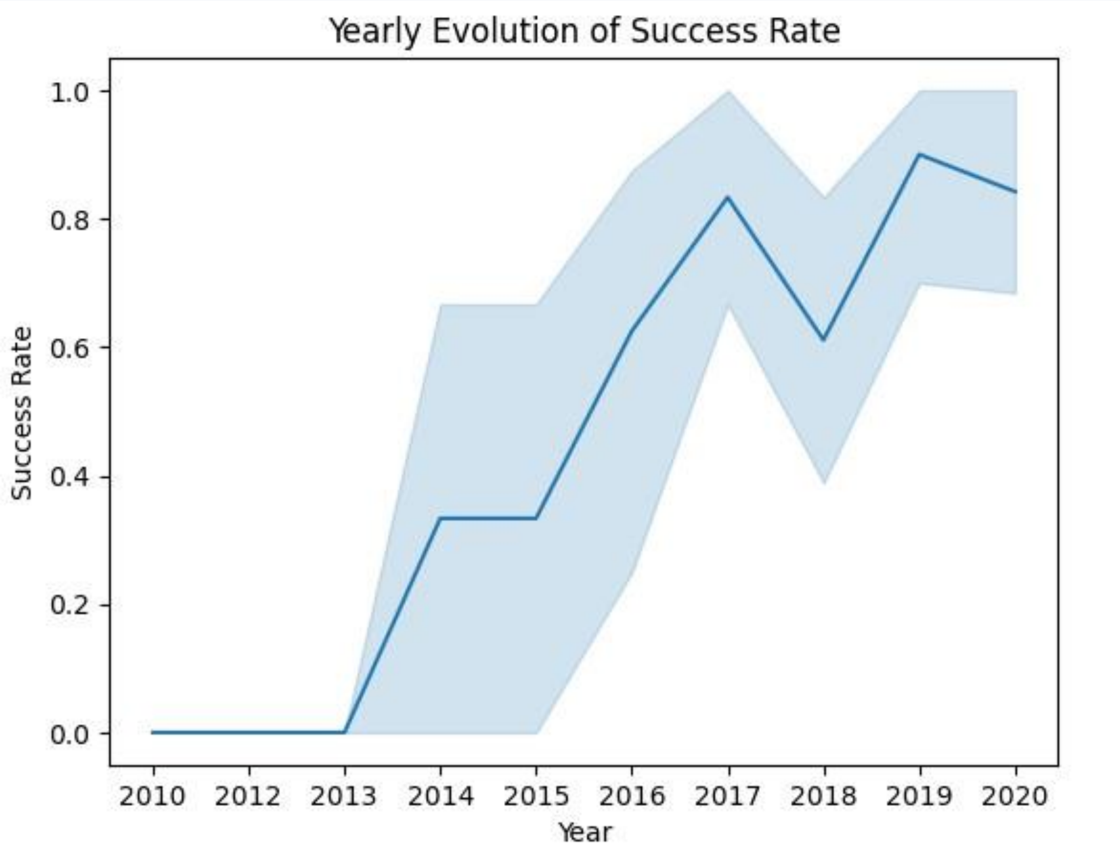
Payload vs. Orbit Type

- We see that heavy payloads and successful landing rates are for PO, LEO and ISS
- GTO we cannot distinguish positive or negative landings



Launch Success Yearly Trend

- Success rate really "Launched" around 2013. Towards the end of 2013 it flatlined. Around 2015 it began to increase again with a small decline around 2017.



All Launch Site Names

- The keyword DISTINCT was essential for finding unique launch site names

```
[23]: %sql SELECT DISTINCT "Launch_Site" FROM SPACEXTABLE

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

[23]:

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

Launch Site Names Begin with 'CCA'

- Utilizing the LIKE keyword to find launch site names starting with CCA and using LIMIT to 5 helped find 5 launch site names with CCA in the name

```
[11]: %sql select "Date", "Time (UTC)", "Launch_Site", "Payload" from SPACEXTABLE where "Launch_Site" like "CCA%" limit 5;
 * sqlite:///my_data1.db
Done.
```

[11]:

Date	Time (UTC)	Launch_Site	Payload
2010-06-04	18:45:00	CCAFS LC-40	Dragon Spacecraft Qualification Unit
2010-12-08	15:43:00	CCAFS LC-40	Dragon demo flight C1, two CubeSats, barrel of Brouere cheese
2012-05-22	7:44:00	CCAFS LC-40	Dragon demo flight C2
2012-10-08	0:35:00	CCAFS LC-40	SpaceX CRS-1
2013-03-01	15:10:00	CCAFS LC-40	SpaceX CRS-2

Total Payload Mass

- To find the sum of the Payload mass, SUM allows for adding all of the Payload Mass together WHERE filters to Customer.
- The total Payload Mass is 48213

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

```
[12]: %sql select sum(PAYLOAD_MASS__KG_) from SPACEXTABLE where "Customer" like "%NASA (CRS)%";

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

[12]:

sum(PAYLOAD_MASS__KG_)
48213

Average Payload Mass by F9 v1.1

- Using AVG computes the average payload mass while IS helps to narrow down the entries to just F9 rockets.
- The average payload mass is 2928.4

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

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

avg(PAYLOAD_MASS__KG_)
2928.4

First Successful Ground Landing Date

- Using the MIN keyword, we are able to obtain the first successful Ground landing date which is in the year 2015

```
[13]: %sql select min(Date) from SPACEXTABLE where "Landing_Outcome" is "Success (ground pad)";

 * sqlite:///my_data1.db
Done.

[13]: min(Date)

      2015-12-22
```

Successful Drone Ship Landing with Payload between 4000 and 6000

- Using the BETWEEN keyword allows us to narrow our query to values between 4000 and 6000

```
[15]: %sql select "Booster_Version", "Payload" from SPACEXTABLE where "Landing_Outcome" is "Success (drone ship)" and 4000 < "PAYLOAD_MASS__KG_" < 6000;

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

[15]:

Booster_Version	Payload
F9 FT B1021.1	SpaceX CRS-8
F9 FT B1022	JCSAT-14
F9 FT B1023.1	Thaicom 8
F9 FT B1026	JCSAT-16
F9 FT B1029.1	Iridium NEXT 1
F9 FT B1021.2	SES-10
F9 FT B1029.2	BulgariaSat-1
F9 FT B1036.1	Iridium NEXT 2
F9 FT B1038.1	Formosat-5
F9 B4 B1041.1	Iridium NEXT 3
F9 FT B1031.2	SES-11 / EchoStar 105
F9 B4 B1042.1	Koreasat 5A
F9 B4 B1045.1	Transiting Exoplanet Survey Satellite (TESS)
F9 B5 B1046.1	Bangabandhu-1

Total Number of Successful and Failure Mission Outcomes

- The COUNT keyword allows us to count the total successes and failure missions
- The GROUPBY puts everything together

[17]:
```
%%sql select
    case
        when "Mission_Outcome" like "%uccess%" then "Success"
        else "Mission_Outcome"
    end as "Binary_Mission_Outcome",
    count(*) as count
from SPACEXTABLE
group by "Binary_Mission_Outcome";
```

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

[17]:

Binary_Mission_Outcome	count
Failure (in flight)	1
Success	100

Boosters Carried Maximum Payload

- Using a Subquery we are able to obtain the Boosters Versions that carried Max payloads

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

```
[18]: %sql select "Booster_Version", "Payload", "PAYLOAD_MASS__KG_" from SPACEXTABLE where "PAYLOAD_MASS__KG_" = (select max(PAYLOAD_MASS__KG_) from SPACEXTABLE);
```

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

[18]:

Booster_Version	Payload	PAYLOAD_MASS__KG_
F9 B5 B1048.4	Starlink 1 v1.0, SpaceX CRS-19	15600
F9 B5 B1049.4	Starlink 2 v1.0, Crew Dragon in-flight abort test	15600
F9 B5 B1051.3	Starlink 3 v1.0, Starlink 4 v1.0	15600
F9 B5 B1056.4	Starlink 4 v1.0, SpaceX CRS-20	15600
F9 B5 B1048.5	Starlink 5 v1.0, Starlink 6 v1.0	15600
F9 B5 B1051.4	Starlink 6 v1.0, Crew Dragon Demo-2	15600
F9 B5 B1049.5	Starlink 7 v1.0, Starlink 8 v1.0	15600
F9 B5 B1060.2	Starlink 11 v1.0, Starlink 12 v1.0	15600
F9 B5 B1058.3	Starlink 12 v1.0, Starlink 13 v1.0	15600
F9 B5 B1051.6	Starlink 13 v1.0, Starlink 14 v1.0	15600
F9 B5 B1060.3	Starlink 14 v1.0, GPS III-04	15600
F9 B5 B1049.7	Starlink 15 v1.0, SpaceX CRS-21	15600

2015 Launch Records

- Using a substring we can extract the 2015 launch records

```
[19]: %%sql
select
    substr(Date,6,2) as "Month",
    substr(Date,0,5) as "Year",
    "Landing_Outcome",
    "Booster_Version",
    "Launch_Site"
from
    SPACEXTABLE
where
    substr(Date,0,5)='2015' and
    "Landing_Outcome" = "Failure (drone ship)"
;
```

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

[19]:

Month	Year	Landing_Outcome	Booster_Version	Launch_Site
01	2015	Failure (drone ship)	F9 v1.1 B1012	CCAFS LC-40
04	2015	Failure (drone ship)	F9 v1.1 B1015	CCAFS LC-40

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

- Utilizing GROUPBY & ORDERBY, we can group together all the records and sort them.
- Using DESC will sort these values in descending order.

```
[21]: %%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
    "Landing_Count" desc
;

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

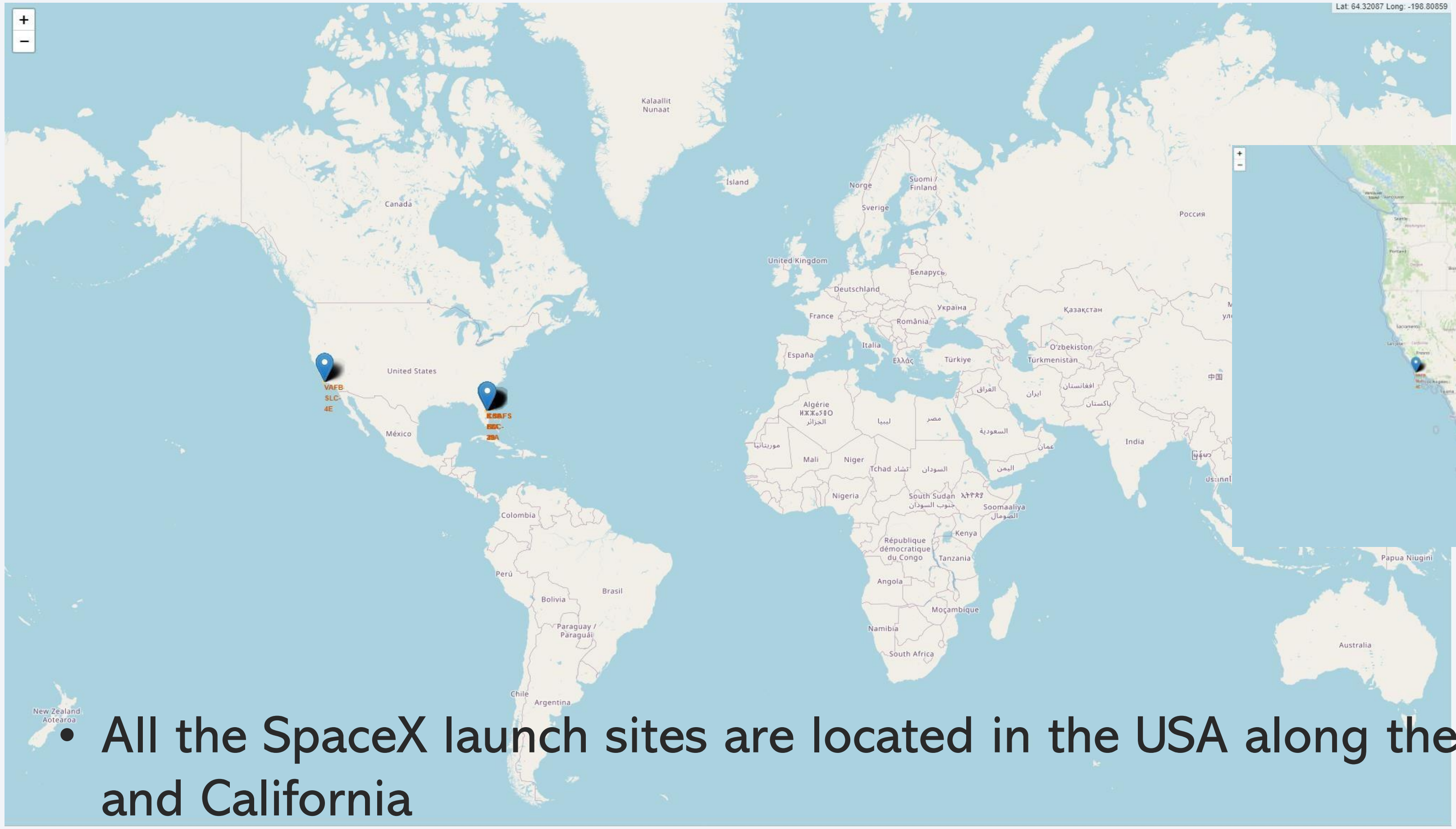
[21]:

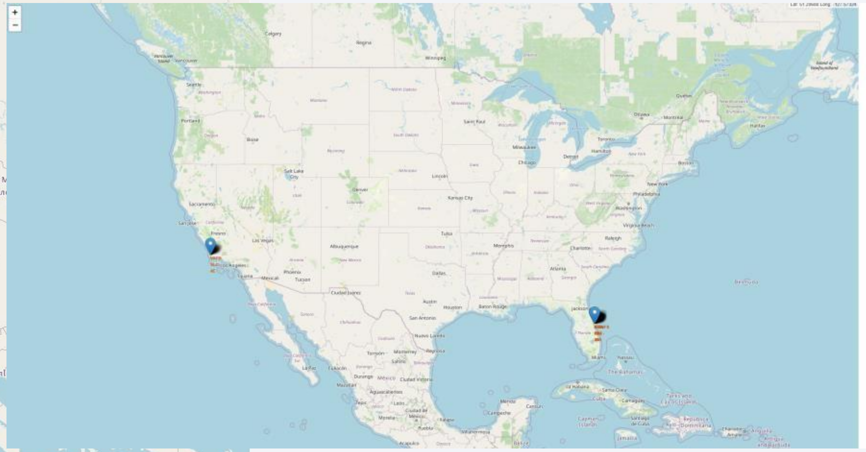
Landing_Outcome	Landing_Count
No attempt	10
Success (drone ship)	5
Failure (drone ship)	5
Success (ground pad)	3
Controlled (ocean)	3
Uncontrolled (ocean)	2
Failure (parachute)	2
Precluded (drone ship)	1

Section 4

Launch Sites Proximities Analysis

Launch Sites on a Global Map

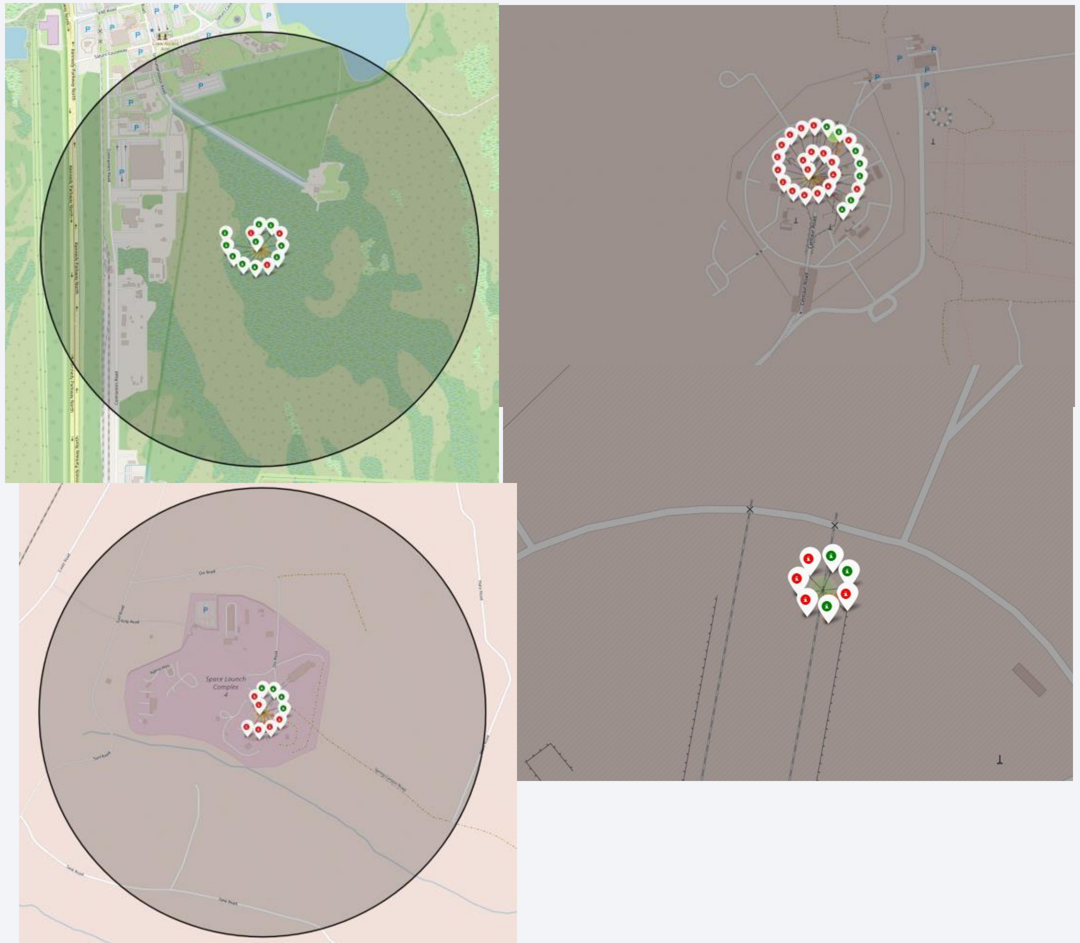




- All the SpaceX launch sites are located in the USA along the Coasts of Florida and California

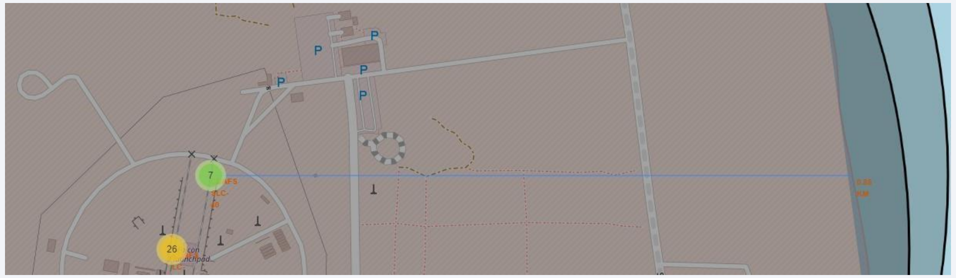
Color Labeled Markers for Success and Failed Launch

- Green Markers indicate successful launches
- Red Markers indicate failure launches



Launch Site proximities to Launch Sites

- We are able to evaluate the distances the launch sites are to surrounding areas including highways, populated areas, and railways
- The nearest city is Orlando to the launch sites on the East Coast

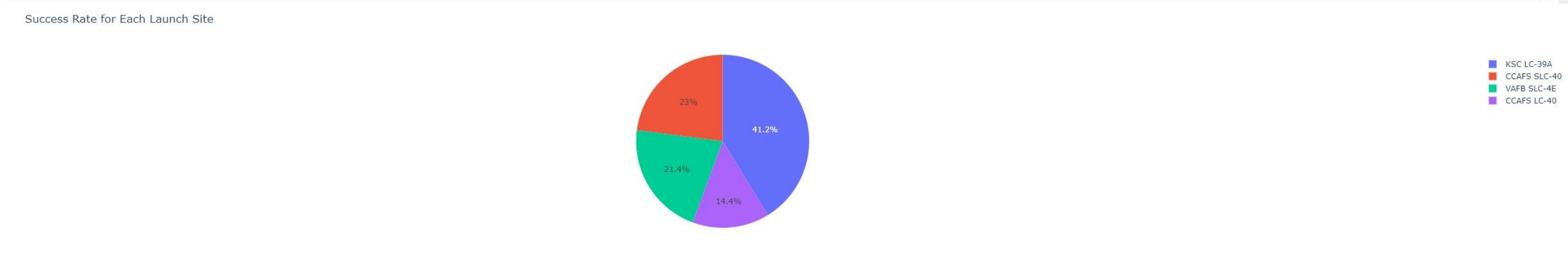


Section 5

Build a Dashboard with Plotly Dash

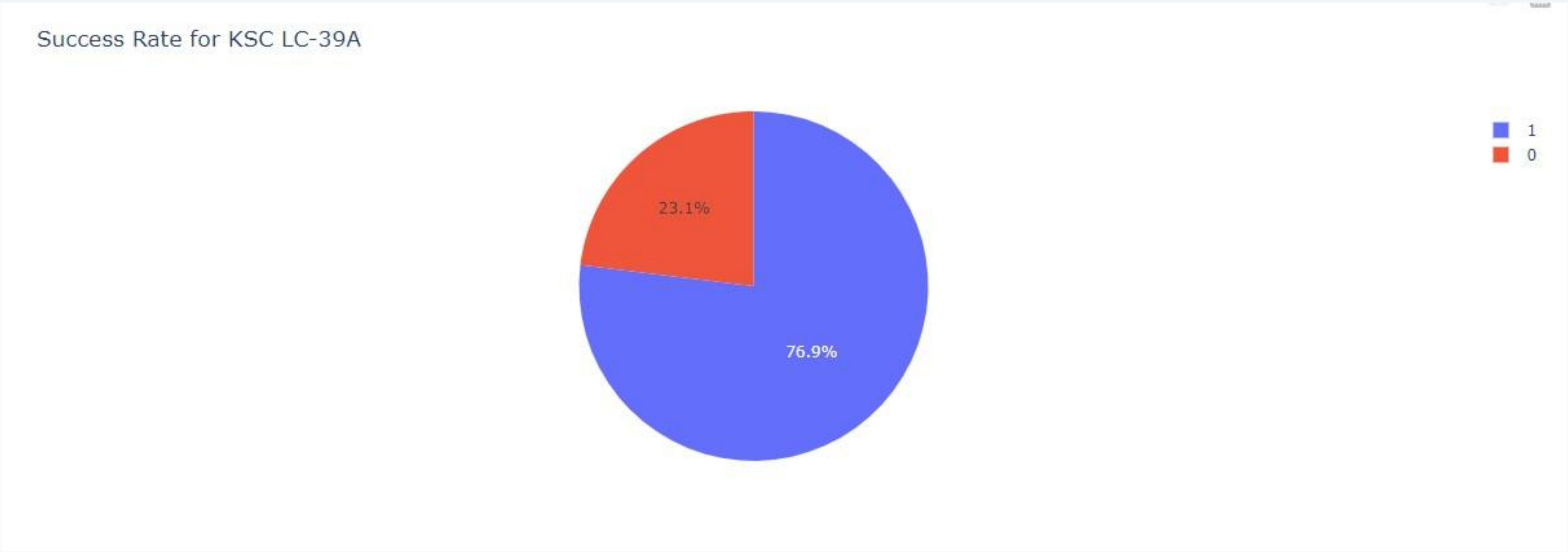
Plotly Dashboard: Overall Pie Chart

- In the piechart we can visually see that KSC LC-39A had the most successful launches out of all of the Lauch sites.



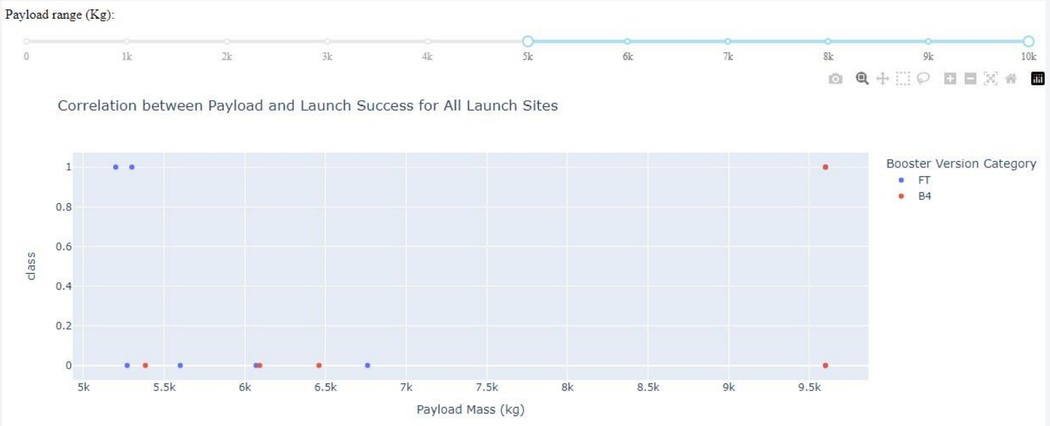
Plotly Dashboard: KSC LC-39A Pie Chart

- Evaluating just launch site KSC LC-39A, we determined that out of all the launches at this particular site, 76.9% were successful and 23.1% were failures



Plotly Dashboard: Scatter Plots for Payload Range

- We can see from the two different scatter plots that lower payload range had more successful launches than the higher payload range launches

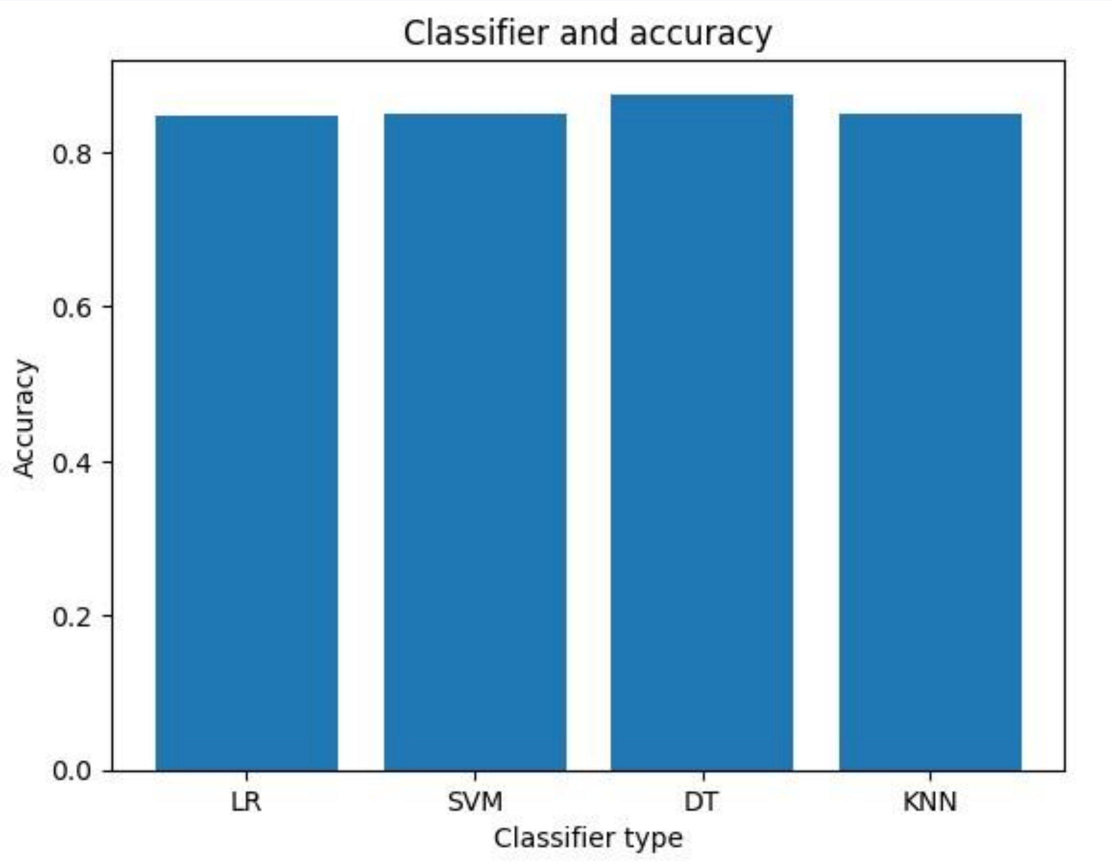


Section 6

Predictive Analysis (Classification)

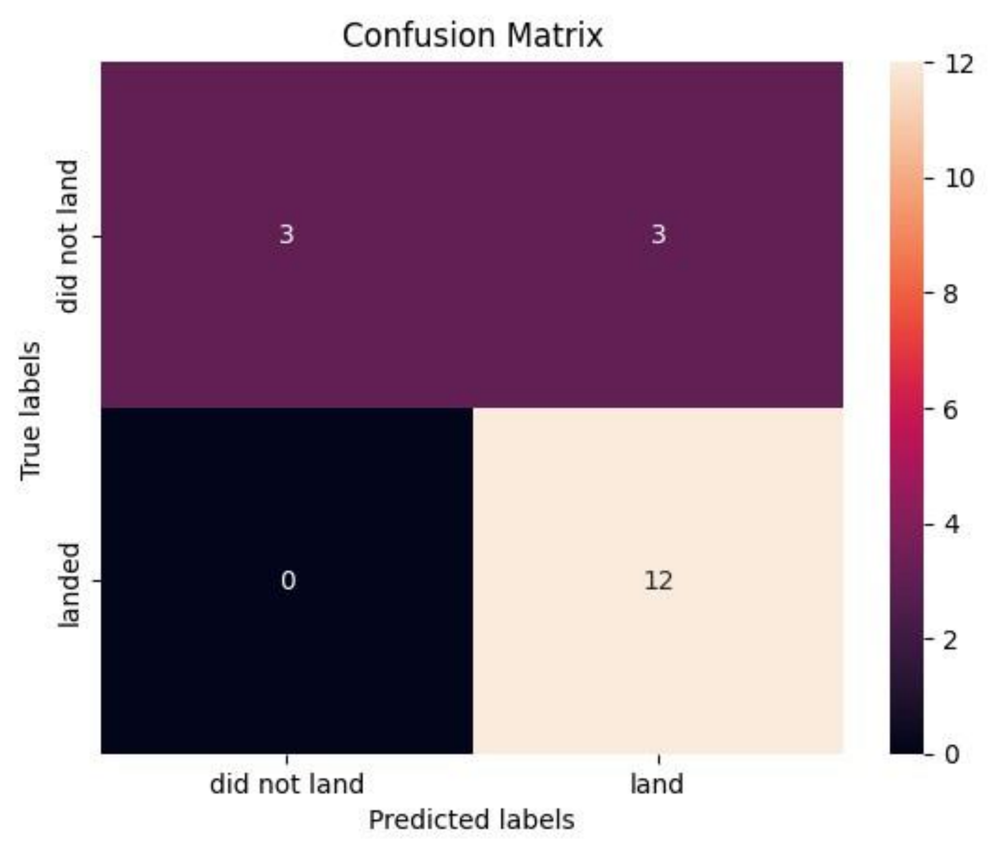
Classification Accuracy

- We can determine in the Bar plots that the accuracy of all the models is almost the same around 83%, however the accuracy of the Decision Tree model is higher than the others.
- This indicates that the Decision Tree model is the most accurate classifier



Confusion Matrix

- The confusion matrix for the Decision tree model shows that the classifier can distinguish between different classes.
- The false postive is a problem, ie., the chance of successful landings marked as successful by the classifier.



Conclusions

- The Decision Tree model is the best algorithm for this Dataset
- Ther larger the flight amount at the launch site, the greater the success rate at a launch site
- Lauch success for low payload mass is better for all orbits
- Launch success rate has steadily increased since 2015
- Orbits ES L1, GEO, HEO, SO, VLEO had the highest success rates
- KSC LC-39A had the most successful launches of any sites
- Launch sites are placed near coastal areas and away from populated areas

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!