IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Dmytro P
9 February 2024

Outline

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

Executive Summary

- Summary of methodologies
  A wide range of methodologies were used such as:
  - SpaceX Data Collection using SpaceX API
  - SpaceX Data Collection with Web Scraping
  - SpaceX Data Wrangling
  - SpaceX Exploratory Data Analysis using SQL
  - Space-X EDA DataViz Using Python Pandas and Matplotlib
  - Space-X Launch Sites Analysis with Folium-Interactive Visual Analytics and PlotyDash
  - SpaceX Machine Learning Landing Prediction
- Summary of all results
  The analysis reveals varying success rates across launch sites, orbits, and payload types. Notably, KSC LC-39A and VAFB SLC 4E exhibit higher success rates compared to CCAFS LC-40. Success rates tend to increase with flight number at each site, with some achieving perfect success rates after specific flight milestones. While certain orbits consistently achieve 100% success, distinguishing between positive and negative outcomes for GTO orbits remains challenging. Overall, the analysis underscores a steady increase in success rates since 2013, highlighting the advancements and reliability of space missions.

Introduction

- **Project background and context**

  Space X advertises Falcon 9 rocket launches on its website with a cost of 62 million dollars; other providers cost upward of 165 million dollars each, much of the savings is because Space X can reuse the first stage. Therefore if we can determine if the first stage will land, we can determine the cost of a launch. This information can be used if an alternate company wants to bid against space X for a rocket launch.

- **Problems we want to find answers to**

  SpaceY that would like to compete with SpaceX. In this project, we want to determine the price of each launch. We will do this by gathering information about SpaceX and creating dashboards for our team. We will also determine if SpaceX will reuse the first stage. We will train a machine learning model and use public information to predict if SpaceX will reuse the first stage.

Section 1

Methodology

Data Collection

- **How data sets were collected.**
  We will be working with SpaceX launch data that is gathered from an API, specifically the SpaceX REST API. This API will give us data about launches, including information about the rocket used, payload delivered, launch specifications, landing specifications, and landing outcome. Our goal is to use this data to predict whether SpaceX will attempt to land a rocket or not.

Data Collection – SpaceX API

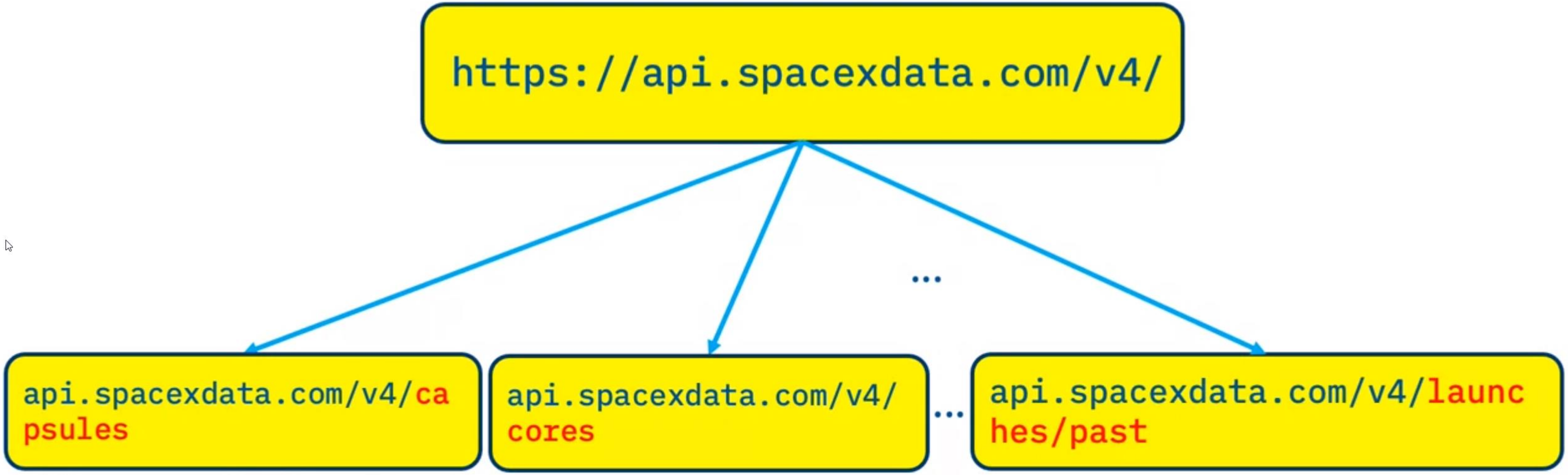
```python
url="https://api.spacexdata.com/v4/launches/past"

response =requests.get(url)

response.json()
```

```python
response.json()
```


```
[{'fairings': {'reused': False,
   'recovery_attempt': False,
   'recovered': False,
   'ships': []},
  'links': {'patch': {'small': 'https://images2.imgbox.com/3c/0e/T8iJcSN3_o.png',
    'large': 'https://images2.imgbox.com/40/e3/GypSkayF_o.png'},
   'reddit': {'campaign': None,
    'launch': None,
    'media': None,
    'recovery': None},
   'flickr': {'small': [], 'original': []},
   'presskit': None,
   'webcast': 'https://www.youtube.com/watch?v=0a_00nJ_Y88',
   'youtube_id': '0a_00nJ_Y88',
   'article': 'https://www.space.com/2196-spacex-inaugural-falcon-1-rocket-lost-launch.html',
   'wikipedia': 'https://en.wikipedia.org/wiki/DemoSat'},
  'static_fire_date_utc': '2006-03-17T00:00:00.000Z',
  'static_fire_date_unix': 1142553600,
  'tbd': False,
```


https://github.com/dpozharov/Testrepo/blob/main/labs/jupyter-labs-spacex-data-collection-api.ipynb

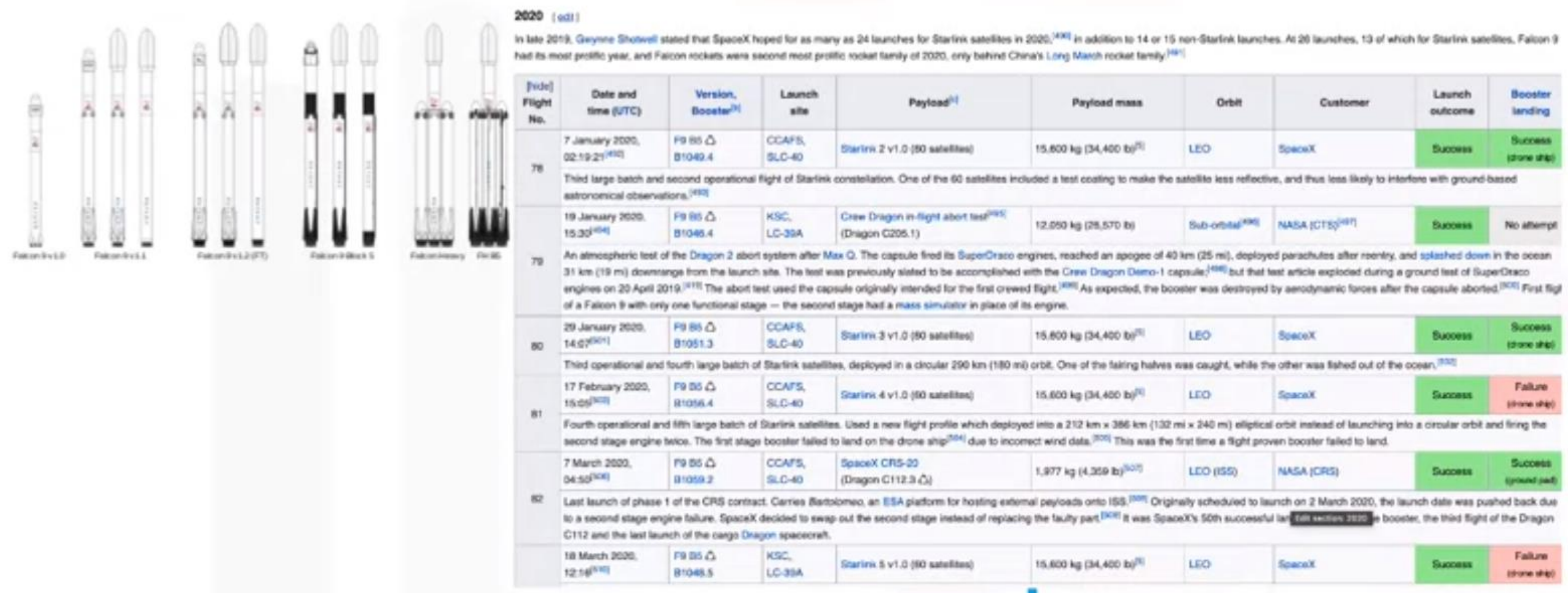
Flowchart of SpaceX API calls



'[{"reuse_count":0,"water_landings":1,"land_landings":0,"last_update":"Hanging in atrium at SpaceX HQ in Hawthorne","launches":["5eb87cdeffd86e000604b330"],"serial":"C101","status":"retired","type":"Dragon 1.0","id":"5e9e2c5bf35918ed873b2664"},{"reuse_count":0,"water_landings":1,"land_

'[{"block":null,"reuse_count":0,"rtls_attempts":0,"rtls_landings":0,"asds_attempts":0,"asds_landings":0,"last_update":"Engine failure at T+33 seconds resulted in loss of vehicle","launches":["5eb87cd9ffd86e000604b32a"],"serial":"Merlin1A","status":"lost","id":".."

Data Collection - Scraping



Web scraping with BeautifulSoup

	FlightNumber	Date	BoosterVersion	PayloadMass	Orbit	LaunchSite	Outcome	Flights	GridFins	Reused	Legs	LandingPad	Block	ReusedCount	Serial	Longitude	Latitude
0	1	2006-03-24	Falcon 1	20.0	LEO	Kwajalein Atoll	None None	1	False	False	False	None	NaN	0	Merlin1A	167.743129	9.047721
1	2	2007-03-21	Falcon 1	NaN	LEO	Kwajalein Atoll	None None	1	False	False	False	None	NaN	0	Merlin2A	167.743129	9.047721
2	4	2008-09-28	Falcon 1	165.0	LEO	Kwajalein Atoll	None None	1	False	False	False	None	NaN	0	Merlin2C	167.743129	9.047721
3	5	2009-07-13	Falcon 1	200.0	LEO	Kwajalein Atoll	None None	1	False	False	False	None	NaN	0	Merlin3C	167.743129	9.047721
4	6	2010-06-04	Falcon 9	NaN	LEO	CCAFS SLC 40	None None	1	False	False	False	None	1.0	0	B0003	-80.577366	28.561857

Data Collection - Scraping

- Web scraping to collect Falcon 9 historical launch records from a Wikipedia page titled List of Falcon 9 and Falcon Heavy launches
- https://en.wikipedia.org/wiki/List_of_Falcon_9_and_Falcon_Heavy_launches
- We will web scrap Falcon 9 launch records with BeautifulSoup:
  - Extract a Falcon 9 launch records HTML table from Wikipedia
  - Parse the table and convert it into a Pandas data frame

Data Wrangling

- Wrangling Data using an API

  In some of the columns, like rocket, we have an identification number, not actual data. We'll be targeting another endpoint to gather specific data for each ID number. The data will be stored in lists and will be used to create our dataset.

- Sampling Data

  The launch data we have includes data for the Falcon 1 booster whereas we only want falcon 9. We will filter/sample the data to remove Falcon 1 launches.

- Dealing with Nulls

  We will deal with the NULL values inside the PayloadMass by calculating the mean of the PayloadMass data and then replaceing the null values in PayloadMass with the mean. We will leave the column LandingPad with NULL values, as it is represented when a landing pad is not used. This will be dealt with using one hot encoding later on.

https://github.com/dpozharov/Testrepo/blob/main/labs/labs-jupyter-spacex-Data%20wrangling.ipynb

Wrangling Data using an API

Function	Targets	Endpoint
getBoosterVersion	→	Rockets URL: https://api.spacexdata.com/v4/rock
getLaunchSite	→	Launchpads URL: https://api.spacexdata.com/v4/laun
getPayloadData	→	Payloads URL: https://api.spacexdata.com/v4/payl
getCoreData	→	getCoreData URL: https://api.spacexdata.com/v4/core

EDA with Data Visualization

- Summarize what charts were plotted and why you used those charts
- Add the GitHub URL of your completed EDA with data visualization notebook, as an external reference and peer-review purpose

EDA with Data Visualization

- Objectives:
  - Obtain some preliminary insights about how each important variable would affect the success rate.
  - Select the features that will be used in success prediction.

To achieve this, we perform exploratory Data Analysis and Feature Engineering using Pandas and Matplotlib:

https://github.com/dpozharov/Testrepo/blob/main/labs/jupyter-labs-eda-dataviz.ipynb.jupyterlite.ipynb

EDA with SQL

Summary of the SQL queries performed:

- Display the names of the unique launch sites in the space mission
- Display 5 records where launch sites begin with the string 'CCA'
- Display the total payload mass carried by boosters launched by NASA (CRS)
- Display average payload mass carried by booster version F9 v1.1
- List the date when the first succesful landing outcome in ground pad was acheived.
- List the names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000
- List the total number of successful and failure mission outcomes
- List the names of the booster_versions which have carried the maximum payload mass.
- 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.
- 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.

https://github.com/dpozharov/Testrepo/blob/main/labs/jupyter-labs-eda-sql-coursera_sqllite.ipynb

Build an Interactive Map with Folium

- Objects we put on the map:
  - All launch sites
  - Mark the success/failed launches for each site
  - The distances between a launch site to its proximities
- Finding an optimal location for building a launch site certainly involves many factors and hopefully we could discover some of the factors by analyzing the existing launch site locations.

https://github.com/dpozharov/Testrepo/blob/main/labs/lab_jupyter_launch_site_location.jupyterlite.ipynb

Build an Interactive Map with Folium

Findings:

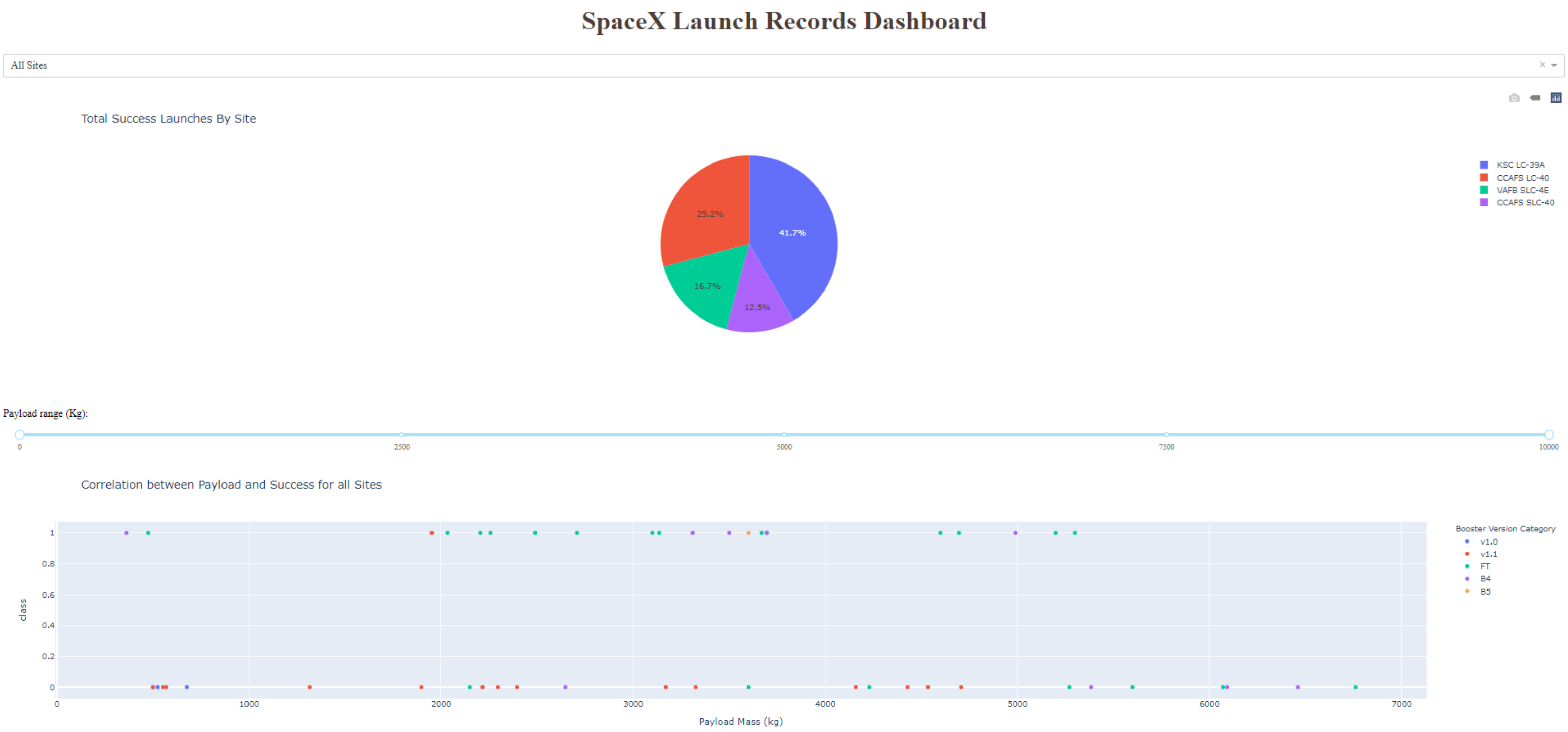
- Launch sites are in close proximity to railways
- Launch sites in close proximity to highways
- Launch sites in close proximity to coastline
- Launch sites keep certain distance away from cities

Build a Dashboard with Plotly Dash

- The SpaceX Launch Records Dashboard components:
  - The interactive dropdown component allows to select All Sites or a specific site. The selection affects the graphs below.
  - Pie chart: Which launch site which one has the largest success count.
  - Pie chart: For a specific launch site, check its detailed success rate (class=0 vs. class=1)
  - Scatter chart allows to analyse the payload mass vs launch outcome for the selected site. The interactive slider allows to select the payload range in kg.
- This analysis should help us figure out the best launch sites for a particular payload mass.

https://github.com/dpozharov/Testrepo/blob/main/labs/spacex_dash_app.py

Build a Dashboard with Plotly Dash



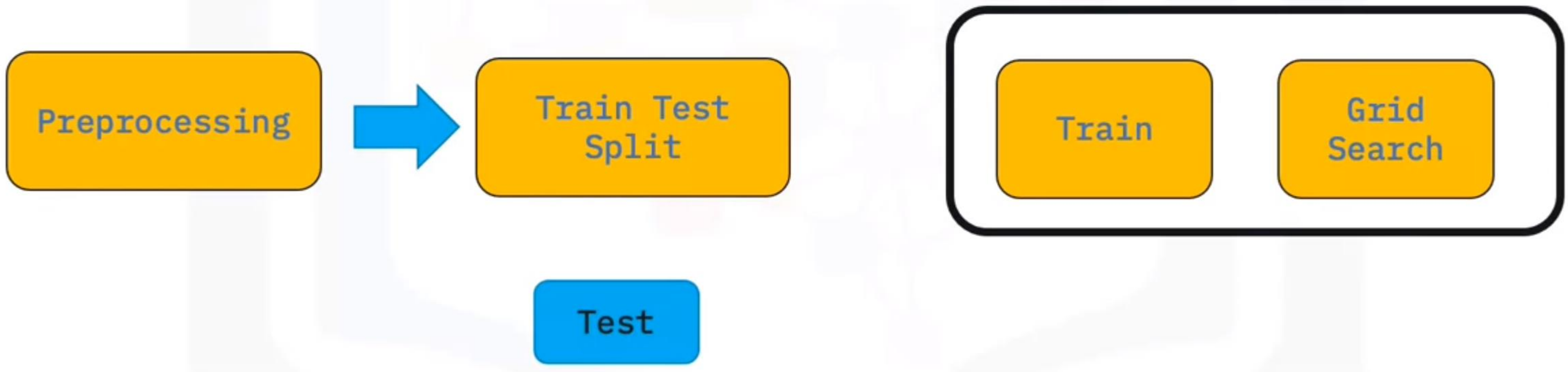
Predictive Analysis (Classification)

We will build a machine learning pipeline to predict if the first stage of the Falcon 9 lands successfully. This will include:

- Preprocessing, allowing us to standardize our data
- Train_test_split, allowing us to split our data into training and testing data,
- We will train the model and perform Grid Search, allowing us to find the hyperparameters that allow a given algorithm to perform best.
- Using the best hyperparameter values, we will determine the model with the best accuracy using the training data.
- You will test Logistic Regression, Support Vector machines, Decision Tree Classifier, and K-nearest neighbors.
- Finally, we will output the confusion matrix.

https://github.com/dpozharov/Testrepo/blob/main/labs/SpaceX_Machine_Learning_Prediction_Part_5.jupyterlite.ipynb

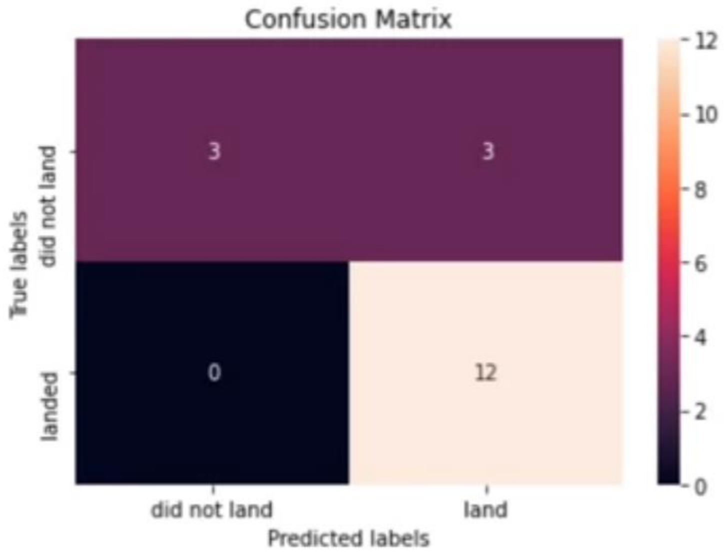
Predictive Analysis (Classification) workflow:



Predictive Analysis (Classification): Determine Model with Best Accuracy

Test

Model
Logistic Regression
Support Vector Machine
Decision Tree Classifier
K-nearest Neighbors

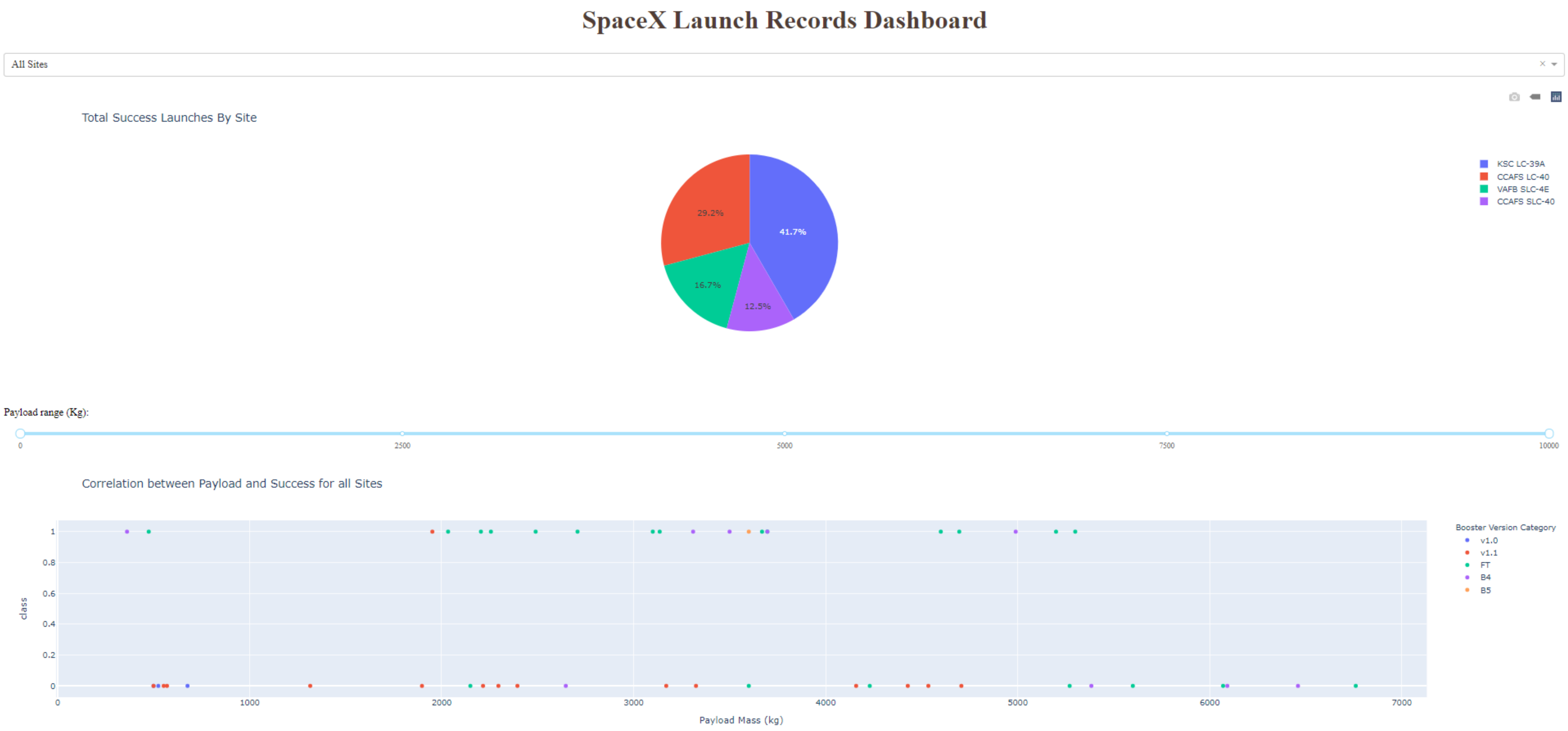


Results

- **Exploratory data analysis results:**
  Having obtained the preliminary insights about how each important variable would affect the success rate, we selected the features that will be used in success prediction:
  features = df[['FlightNumber', 'PayloadMass', 'Orbit', 'LaunchSite', 'Flights', 'GridFins', 'Reused', 'Legs', 'LandingPad', 'Block', 'ReusedCount', 'Serial']]
- Interactive analytics demo in screenshots
- Predictive analysis results

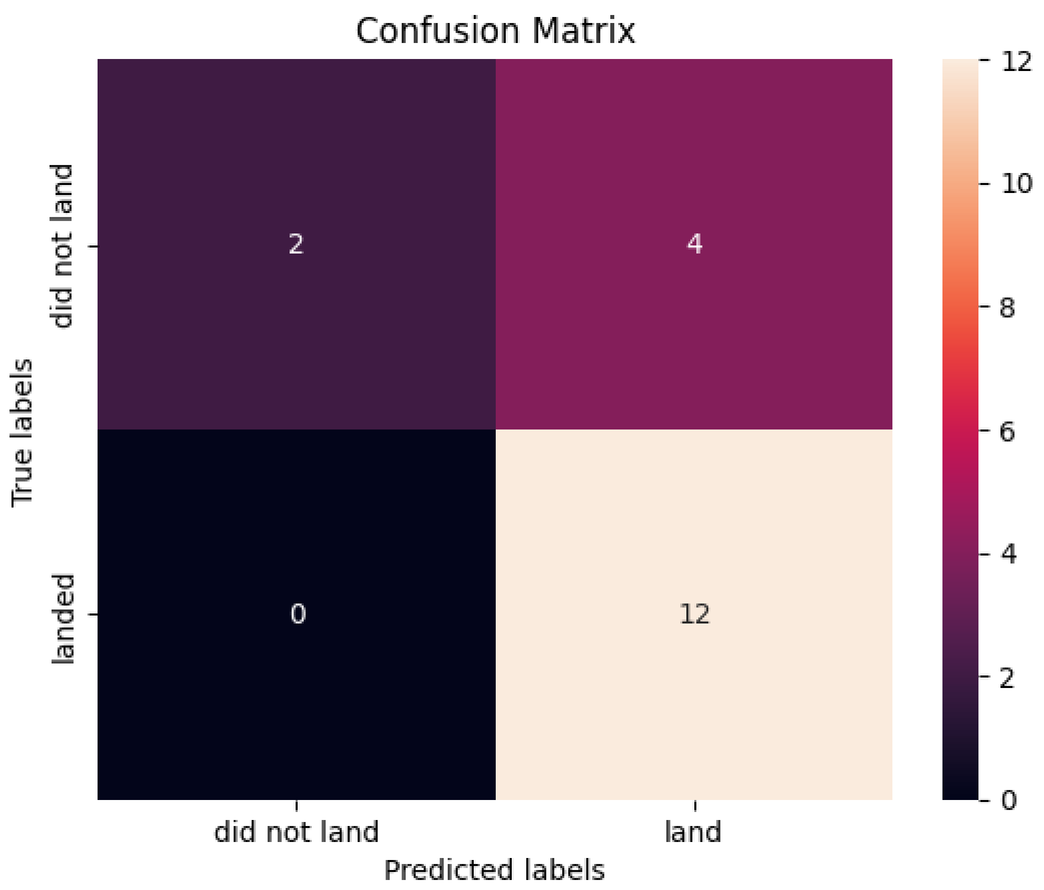
Results

- Interactive analytics demo in screenshots



Results

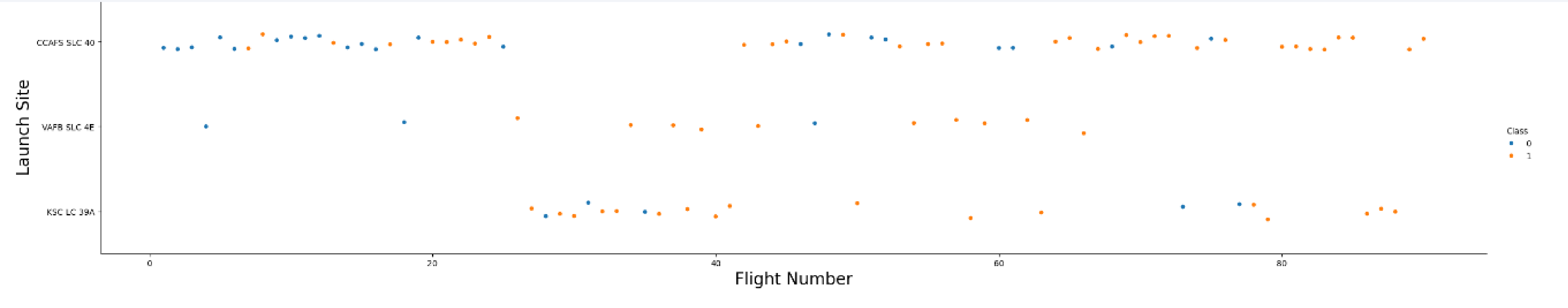
- **Predictive analysis results: decision tree classifier provides the best prediction**



Section 2

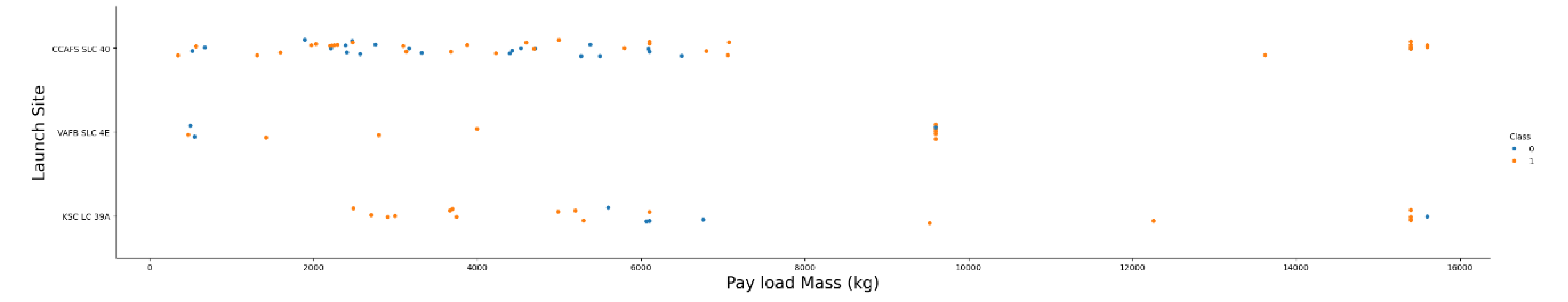
Insights drawn from EDA

Flight Number vs. Launch Site



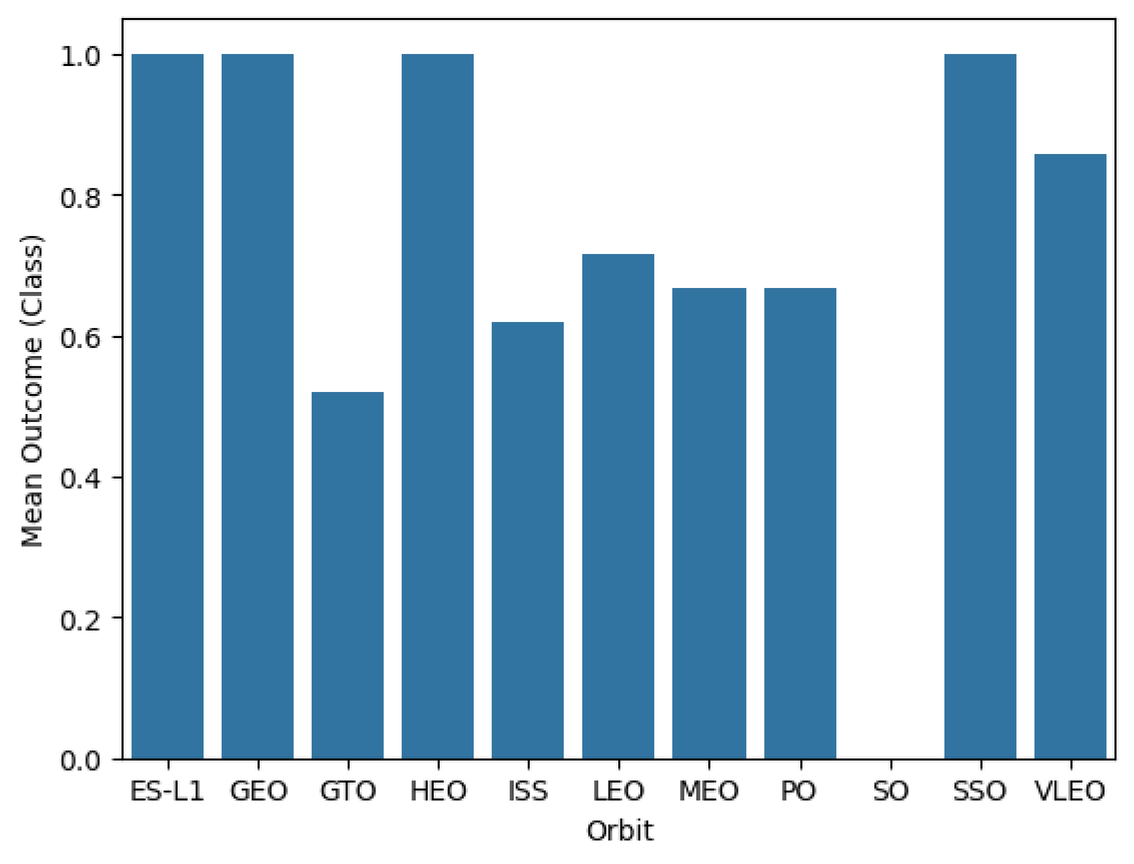
- Success rate in general increase with the flight number

Payload vs. Launch Site



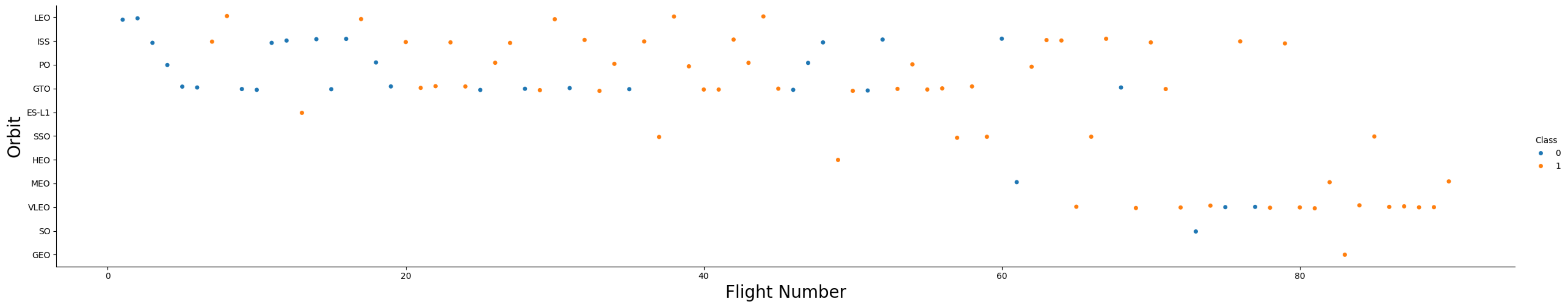
Observation: for the VAFB-SLC launchsite there are no rockets launched for heavypayload mass (greater than 10000).

Success Rate vs. Orbit Type



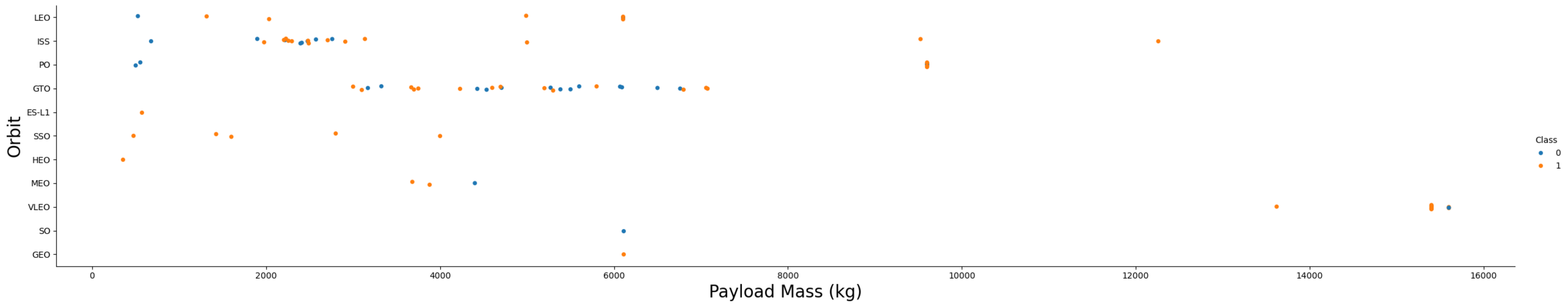
Orbits ES-L1, GEO, HEO and SSO have very high success rate (mean outcome = 1)

Flight Number vs. Orbit Type



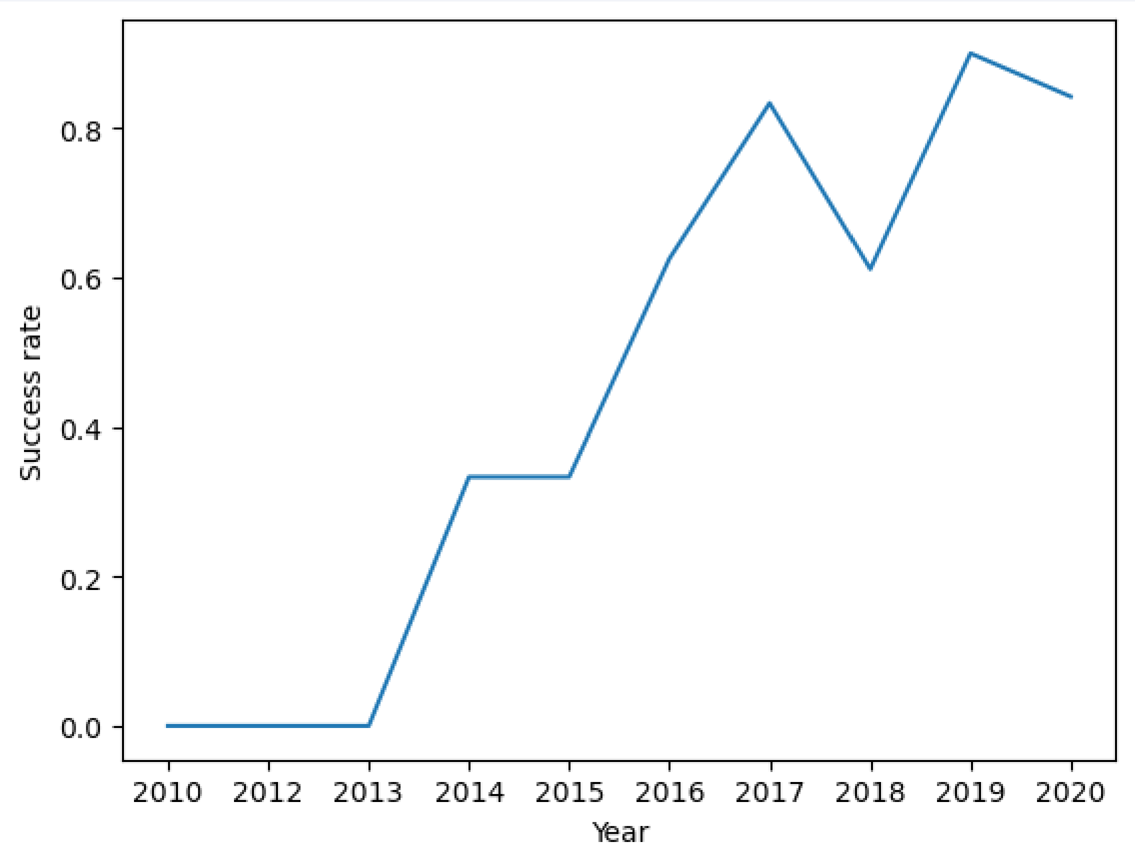
Observation: in the LEO orbit the Success appears related to the number of flights; on the other hand, there seems to be no relationship between flight number when in GTO orbit.

Payload vs. Orbit Type



Observation: with heavy payloads the successful landing or positive landing rate are more for Polar, LEO and ISS.
However for GTO we cannot distinguish this well as both positive landing rate and negative landing (unsuccessful mission) are both there here.

Launch Success Yearly Trend



The sucess rate since 2013 kept increasing till 2020

All Launch Site Names

%sql select distinct Launch_Site from SPACEXTABLE

These are all the launch sites we have in the dataset:

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

Launch Site Names Begin with 'CCA'

%sql select * from SPACEXTABLE where Launch_Site like "CCA%" limit 5

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
2012-10-08	0:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500	LEO (ISS)	NASA (CRS)	Success	No attempt
2013-03-01	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:
  %sql select sum(PAYLOAD_MASS__KG_) from SPACEXTABLE where Customer='NASA (CRS)'
- The Total Payload Mass is 45,596 kg.

sum(PAYLOAD_MASS__KG_)
45596

Average Payload Mass by F9 v1.1

- Calculate the average payload mass carried by booster version F9 v1.1
  %sql select avg(PAYLOAD_MASS__KG_) from SPACEXTABLE where Booster_Version='F9 v1.1'
- The average payload mass carried by booster version F9 v1.1 is 2,928.4 kg

avg(PAYLOAD_MASS__KG_)
2928.4

First Successful Ground Landing Date

- Find the dates of the first successful landing outcome on ground pad
  %sql select min(Date) from SPACEXTABLE where Landing_Outcome='Success (ground pad)'
- The first successful landing outcome on ground pad was on 22 Dec 2015

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
  %sql select 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

- Calculate the total number of successful and failure mission outcomes
```
SELECT Mission_Outcome as Outcome, COUNT(*) as count
FROM SPACEXTABLE
GROUP BY Mission_Outcome
```
- Most missions are successful

Outcome	count
Failure (in flight)	1
Success	98
Success	1
Success (payload status unclear)	1

Boosters Carried Maximum Payload

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

- List the names of the booster which have carried the maximum payload mass
  SELECT distinct b.Booster_Version
- FROM SPACEXTABLE b
  WHERE b.PAYLOAD_MASS__KG_ IN (
      SELECT MAX(PAYLOAD_MASS__KG_)
      FROM SPACEXTABLE
)
- The max payload max is 15600 kg and these are the boosters that were able to carry it

2015 Launch Records

- List the failed landing_outcomes in drone ship, their booster versions, and launch site names for in year 2015
  SELECT substr(Date, 6,2) as month, Landing_Outcome, Booster_Version, Launch_Site
  FROM SPACEXTABLE
  WHERE substr(Date,0,5)='2015'
  AND Landing_Outcome = 'Failure (drone ship)'
- Months when these failures accured:

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

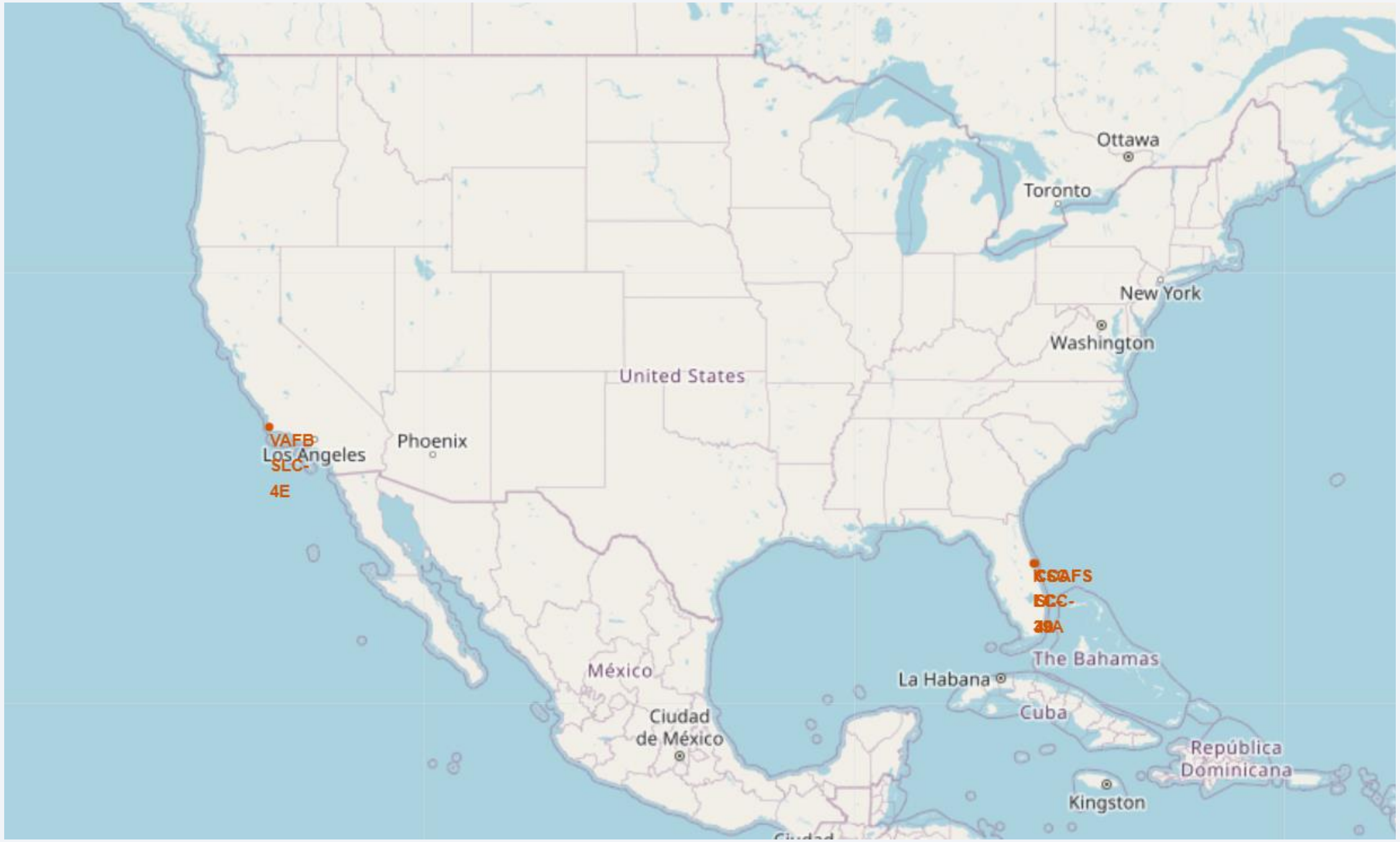
Landing_Outcome	Count
No attempt	10
Success (drone ship)	5
Failure (drone ship)	5
Success (ground pad)	3
Controlled (ocean)	3
Uncontrolled (ocean)	2
Failure (parachute)	2
Precluded (drone ship)	1

- 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 most frequent outcome is No attempt. Then success and failure are almost equally distributed between different landing modes

Section 3

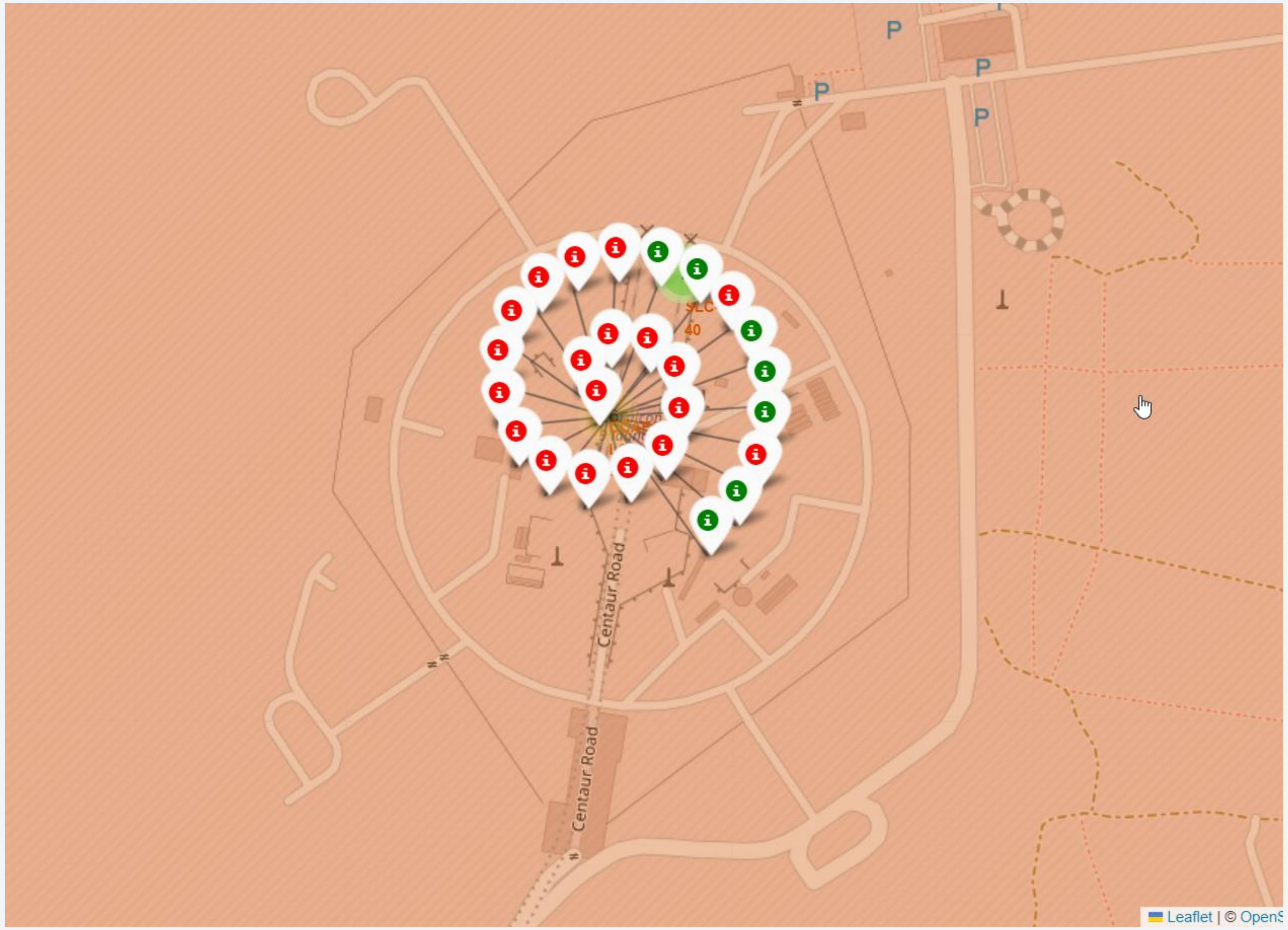
Launch Sites Proximities Analysis

Launch sites on the map



- All launch sites are in the US in close proximity to the Equator line
- All launch sites ae in very close proximity to the coast

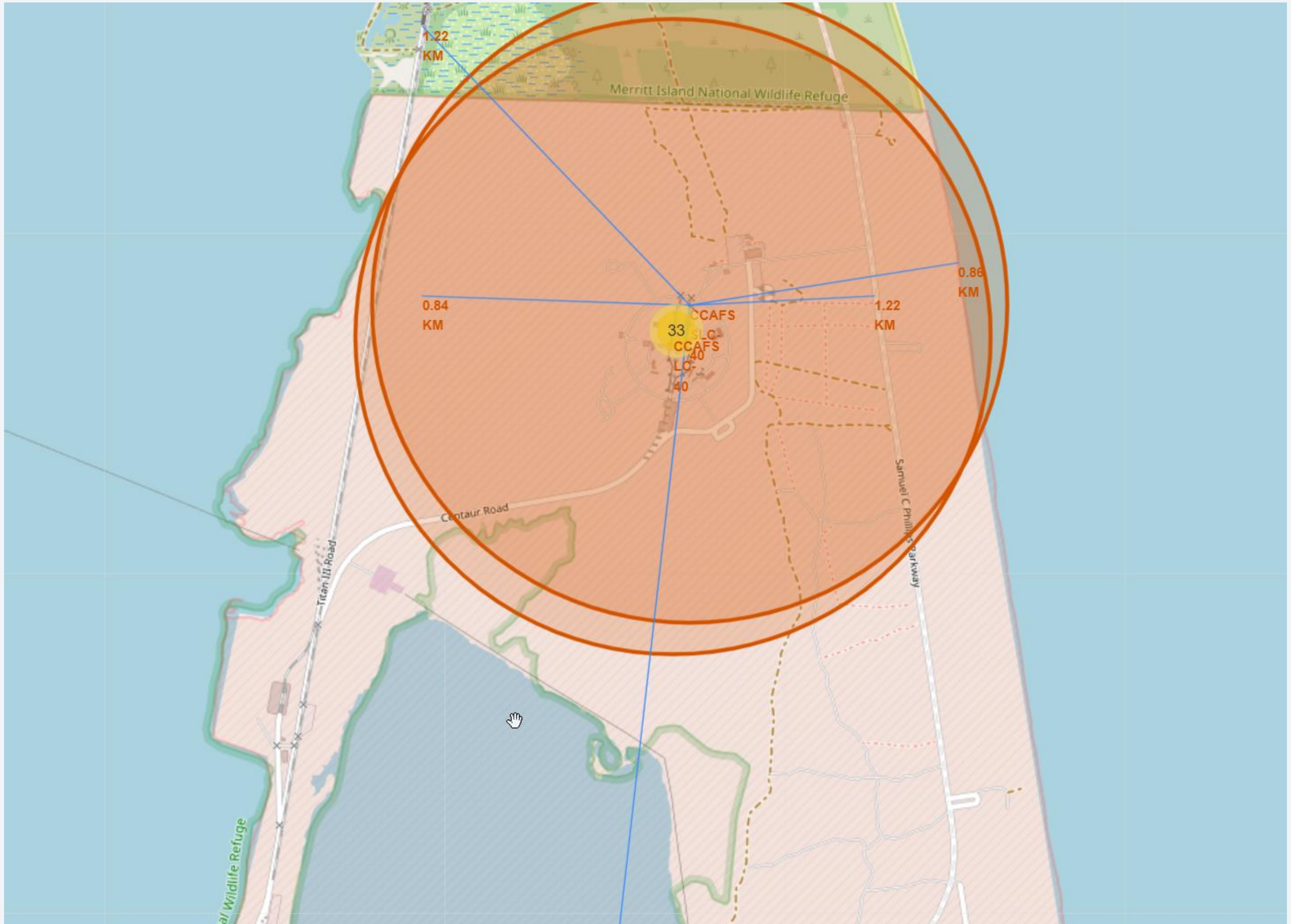
Success/failed launches for each site on the map



The color-labeled markers in marker clusters allow to easily identify which launch sites have relatively high success rates.

CCAFS LC-40 is shown here.

Distance lines to the proximities from a launch site



Findings:

- Launch sites are in close proximity to railways
- Launch sites in close proximity to highways
- Launch sites in close proximity to coastline
- Launch sites keep certain distance away from cities

Section 4

Build a Dashboard with Plotly Dash

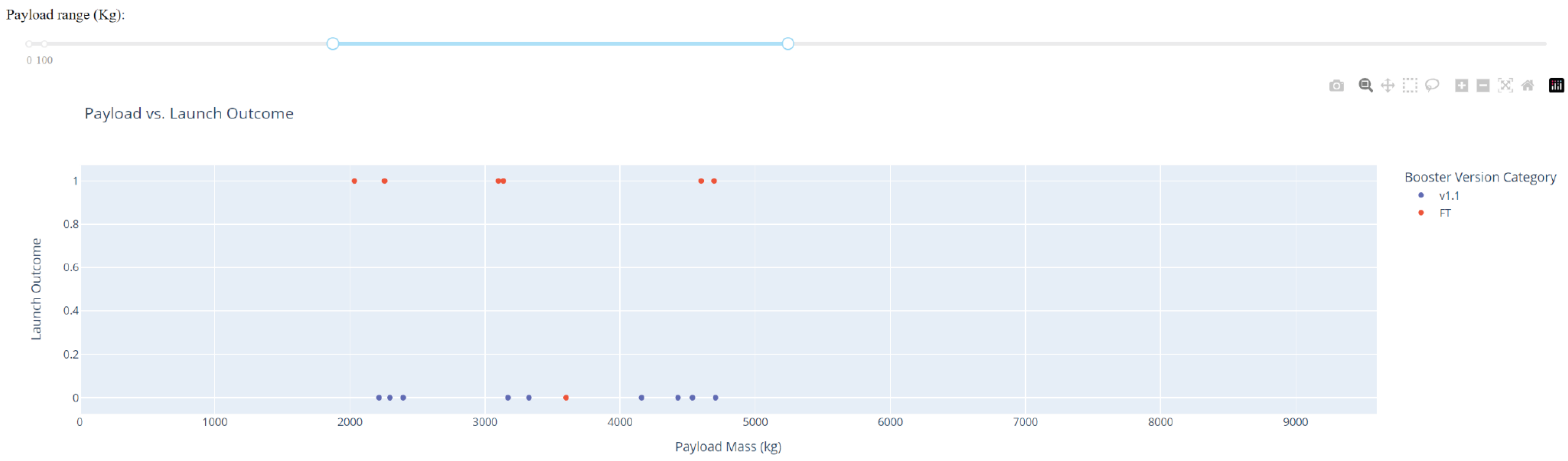
Total Success Launches by Site

KSC LC-39A and CCAFS LC-40 have the highest number of success launches.

Success vs. Failed Launches for CCAFS LC-40

About 27% of failures and 73% of successes

Payload vs. Launch Outcome for CCAFS LC-40

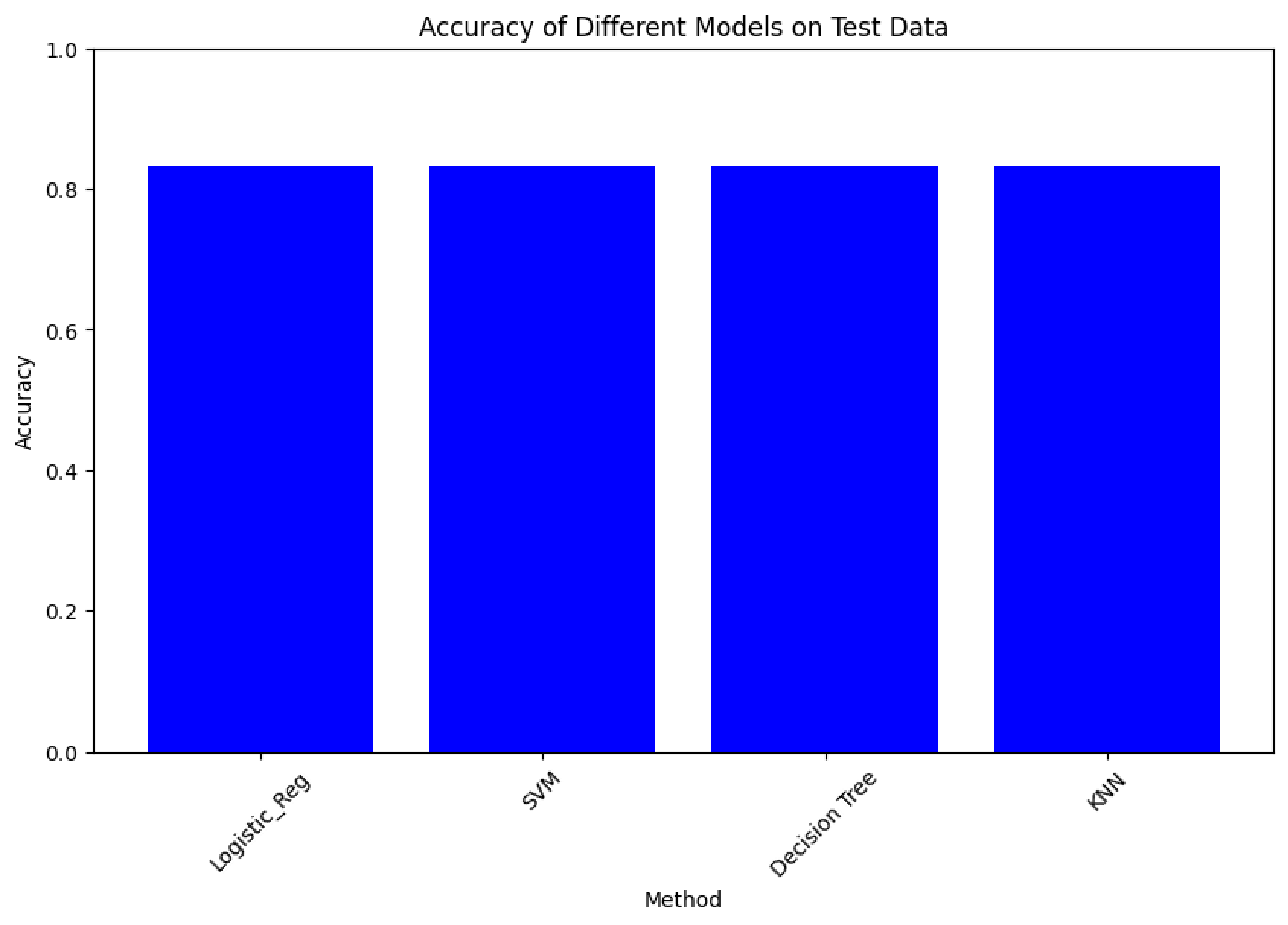


The outcome for this particular site and this particular payload mass range heavily depends on the buster version: v1.1 is mostly failures and FT is always success

Section 5

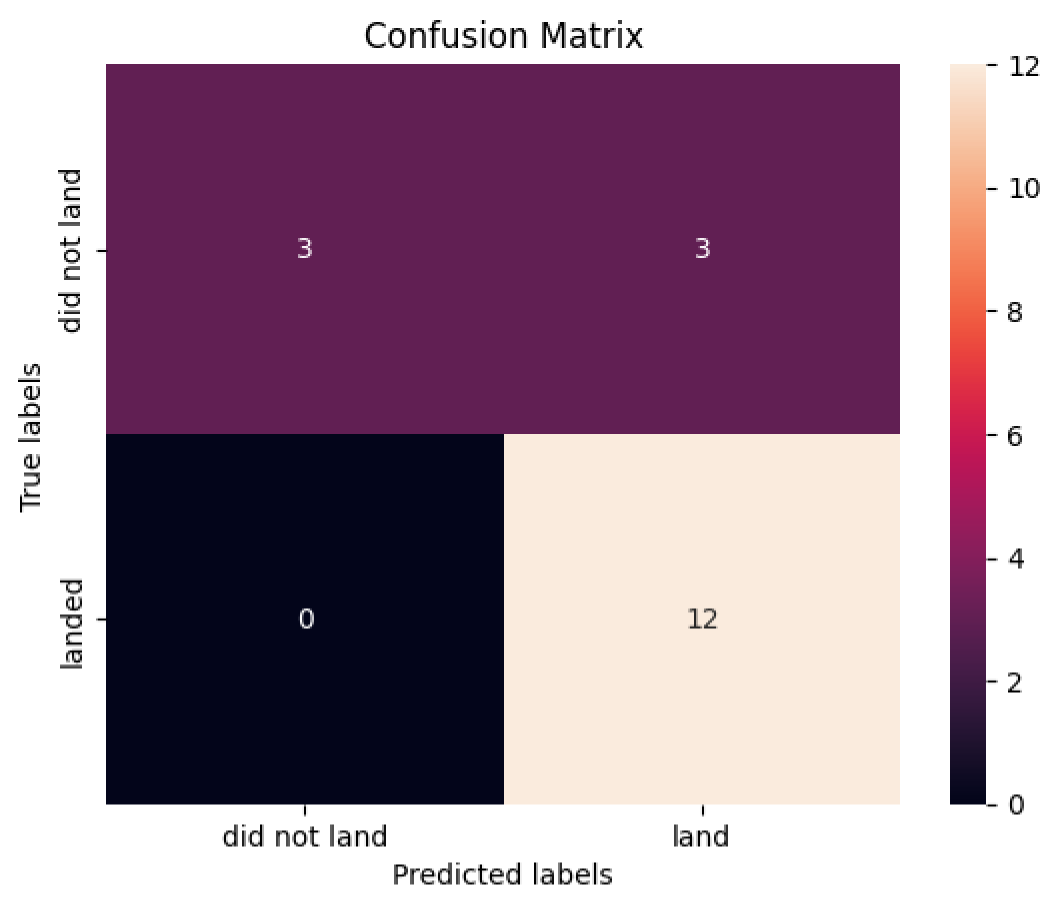
Predictive Analysis (Classification)

Classification Accuracy



- All models have the same classification accuracy

Confusion Matrix



All the classification model have the same confusion matrices. The major problem is false positives for all the models.

Conclusions

- The success rates vary across different launch sites. Specifically, the success rate at CCAFS LC-40 is 60%, while KSC LC-39A and VAFB SLC 4E have success rates of 77%. It's notable that as the flight number increases at each launch site, the success rate tends to increase as well. For instance, the success rate for VAFB SLC 4E reaches 100% after the 50th flight, while both KSC LC 39A and CCAFS SLC 40 achieve 100% success rates after the 80th flight.
- Regarding payload versus launch site, there is an observation that for the VAFB-SLC launch site, no rockets have been launched for heavy payload masses greater than 10,000.
- In terms of orbits, ES-L1, GEO, HEO, and SSO orbits have the highest success rates, all at 100%. However, the SO orbit stands out with a success rate of approximately 50%, and the orbit SO has a 0% success rate.
- For LEO orbit, success appears to be related to the number of flights, while for GTO orbit, there seems to be no clear relationship between flight number and success rate.
- When considering heavy payloads, successful landings or positive landing rates are more common for Polar, LEO, and ISS orbits. However, distinguishing between positive and negative landing rates for GTO orbit is challenging, as both positive and negative outcomes occur.
- Finally, the success rate has been steadily increasing since 2013 and continued to do so until 2020.

Appendix

- All Jupiter Notebooks and labs are available here:

  https://github.com/dpozharov/Testrepo/tree/main/labs

Thank you!