IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

William Mansell
March 2024

Outline

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

Executive Summary

- Summary of methodologies
  - Data Collection and Wrangling
  - Exploratory Data Analysis & Data Visualization
  - Exploratory Data Analysis using SQL
  - Building an interactive map with Folium
  - Building a Dashboard with Dash – Plotly
  - Predictive Analytics & Modeling
- Summary of all results
  - Collection, Wrangling, Exploration results
  - Predictive Analytics results

Introduction

- Project background and context
  - SpaceX is a successful corporation in the commercial space, with a focus on making space travel affordable. This company advertises Falcon 9 rocket launches costing $62M – the nearest competitor costing upwards of $165M. This is accomplished through reuse of the first stage of launch.
  - If we can determine if the first stage will land, we can then determine the cost of a launch using a variety of data science methods.
- Problems you want to find answers
  - How do variables such as payload mass, launch site, number of flights, orbits, and others impact the success rate of the first stage landing?
  - What is the best algorithm that can be created and leveraged for binary classification of success?

Section 1

Methodology

Methodology

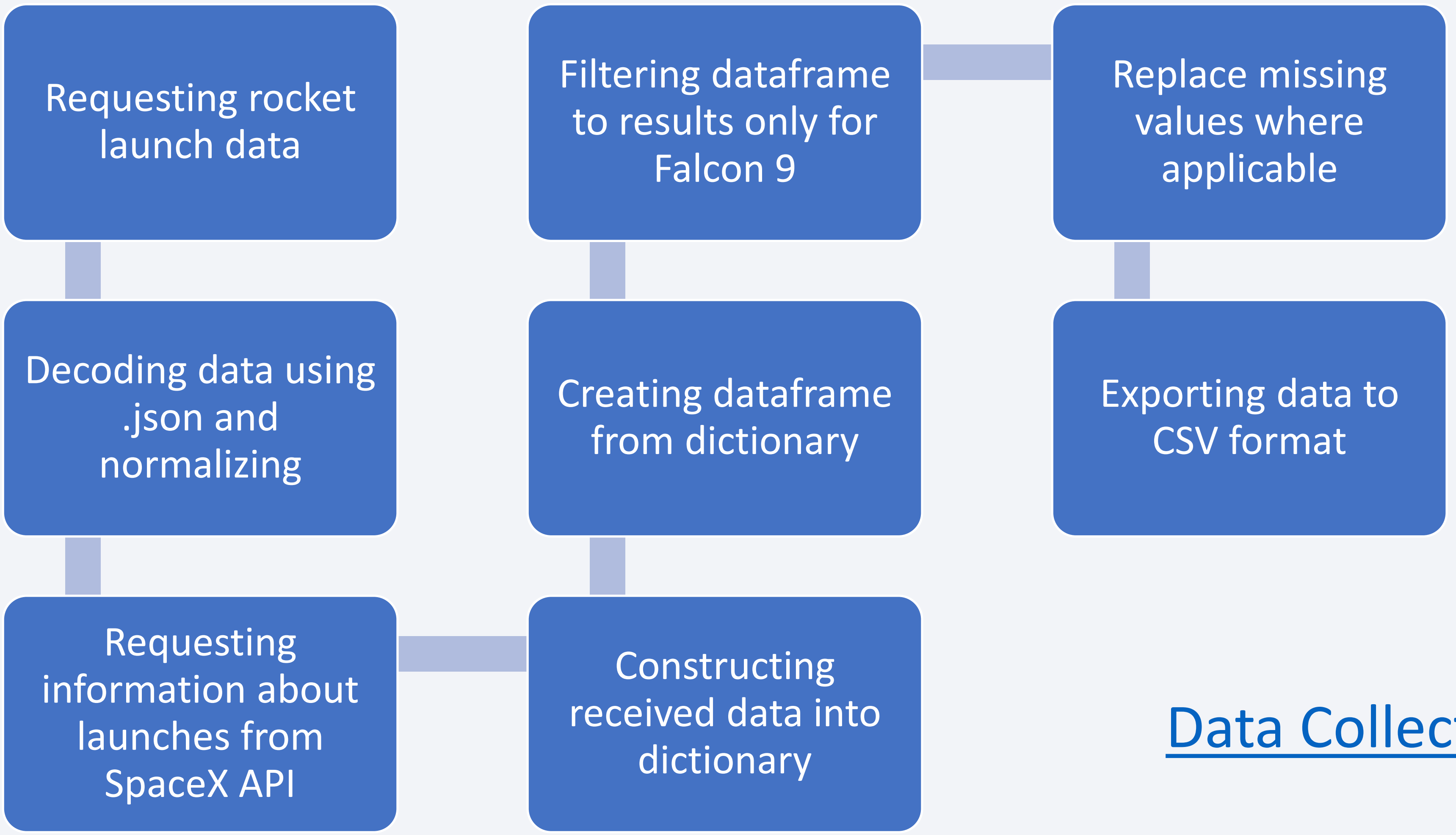
Executive Summary

- Data collection methodology:
  - We leveraged web scraping from Wikipedia combined with the SpaceX API
- Perform data wrangling
  - We filtered the data to relevant data points and eliminated missing values
  - Leveraged various techniques such as One Hot Encoding to prepare the data for binary classification
- 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 and evaluate the classification models that led to best results from our analysis

Data Collection

- Data sets were collected using a combination of API requests from SpaceX and web scraping data from the Wikipedia site. This combination allowed for complete information about the launches, providing us with necessary knowledge for a more detailed analysis
- Data Columns utilized from SpaceX API:
  - FlightNumber, Date, BoosterVersion, PayloadMass, Orbit, LaunchSite, Outcome, Flgihts, GridFins, Resued, Legs, LandingPad, Block, ReusedCount, Serial, Longitude, Latitude
- Data Columns utilized from Wikipedia Page Web Scraping:
  - Flight No., Launch Site, Payload, PayloadMass, Orbit, Customer, Launch Outcome, Version Booster, Booster Landing, Date, Time

Data Collection – SpaceX API



Data Collection API

Data Collection - Scraping

1. Requesting Falcon 9 data from Wikipedia
2. Leveraged BeautifulSoup for HTML
3. Extracted all columns
4. Collected and analyzed the data
5. Input data into dictionary
6. Created a dataframe from dictionary
7. Exporting data to CSV format

[Data Collection via Web Scraping]()

Data Wrangling

- Performed exploratory analysis to derive training labels
- Calculate number of launches
- Calculate orbits
- Calculate mission outcome per orbit
- Create landing outcome label
- Export Data to csv

[Data Wrangling]

EDA with Data Visualization

- Several visualization charts were plotted to derive insights:
  - Flight Number vs Payload Mass
  - Flight Number vs Launch Site
  - Payload Mass vs Launch Site
  - Orbit Type vs Success Rate
  - Flight Number vs Orbit Type
  - Payload Mass vs Orbit Type
  - Success Rate Yearly Trend
- Scatter plots were used to show the relationship between variables
- Bar charts were used to show comparisons amongst categories
- Line charts were used to show trends over time

[EDA with Data Visualization]()

EDA with SQL

- Utilized several SQL queries for data exploration:
  - Display names of each distinct launch site
  - Displayed rows of data where launch sites began with ‘CCA’
  - Displayed total payload mass carried by boosters
  - Displayed average payload mass carried by booster ver. F9 v1.1
  - Date when the first successful landing outcome occurred
  - Listing names of boosters which have drone ship success between payload thresholds
  - Etc…

EDA with SQL

Build an Interactive Map with Folium

- Folium Map Markers
  - Added circle markers, pop-up labels, and text labels using latitude and longitude coordinates
  - Added circle markers, pop-up labels, and text labels using latitude and longitude coordinates to show geographical locations
- Launch Outcomes for each Site
  - Colored markers of success [green] and unsuccessful [red] launches using markerclusters to identify success rates for each site
- Distances Between Launch Sites
  - Colored lines to show distances between various launch sites

[Interactive Map with Folium]

Build a Dashboard with Plotly Dash

- Pie Chart showing successful launches
- Slider of payload mass range
- Scatter Chart of payload mass vs success rate for the different booster versions

- Unfortunately, this was the only part of the analysis that didn’t bear fruit, as the tools were not functioning properly

Predictive Analysis (Classification)

- Created numpy array for “class” data
- Standardized the data with StandardScaler
- Transformed the data and then “fit” it
- Split the data into a training set and a testing set
- Created a GridSearchCV to establish the best parameters
- Used LogReg, SVM, Decision Tree, and K-N models
- Calculating accuracy across various variables to determine best model
- Developed and applied confusion matrix for each model
- Found the best model based on Jaccard Score and F1 Score functions

[Machine Learning Prediction]()

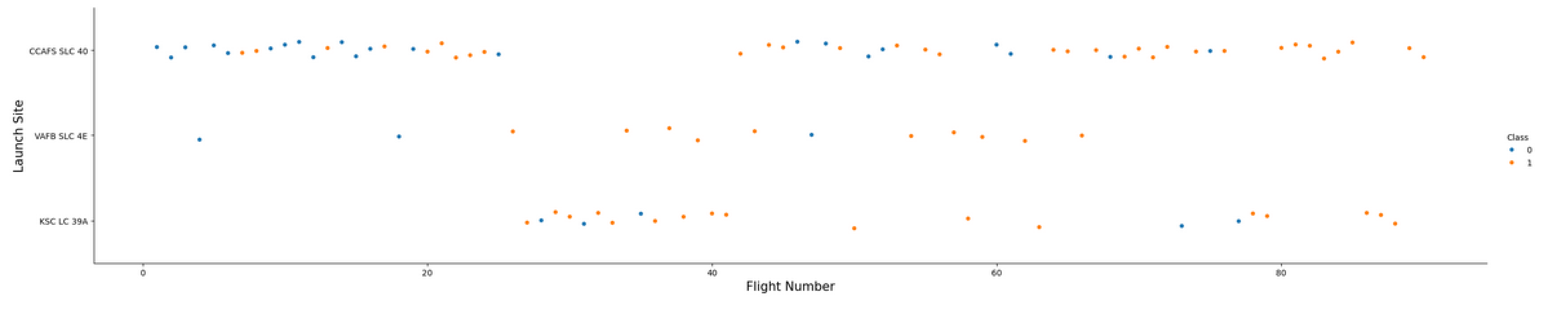
Results

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

Section 2

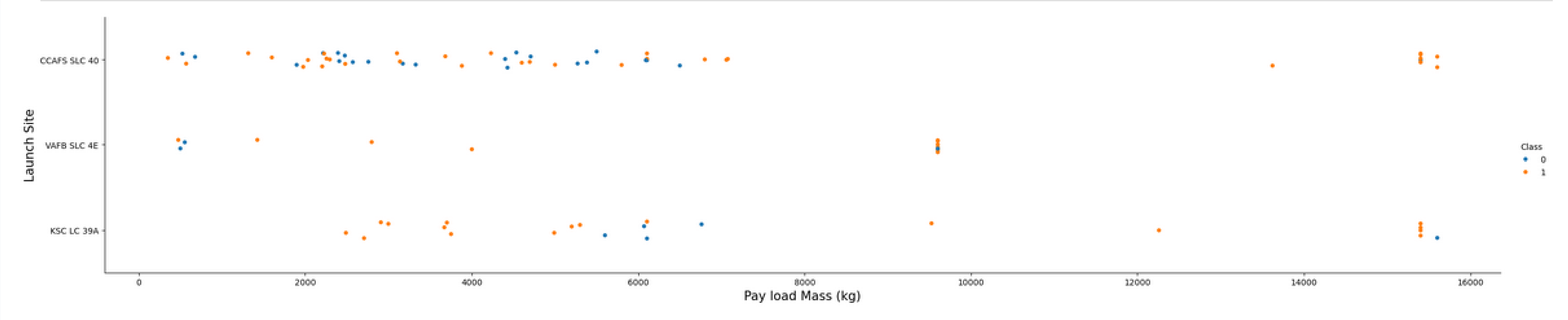
Insights drawn from EDA

Flight Number vs. Launch Site



- Earliest flights failed; later flights succeeded
- CCAFS SLC 40 launch site has majority of launches
- VAFB SLC 4E has a high success rate. KSC LC 39A is mediocre

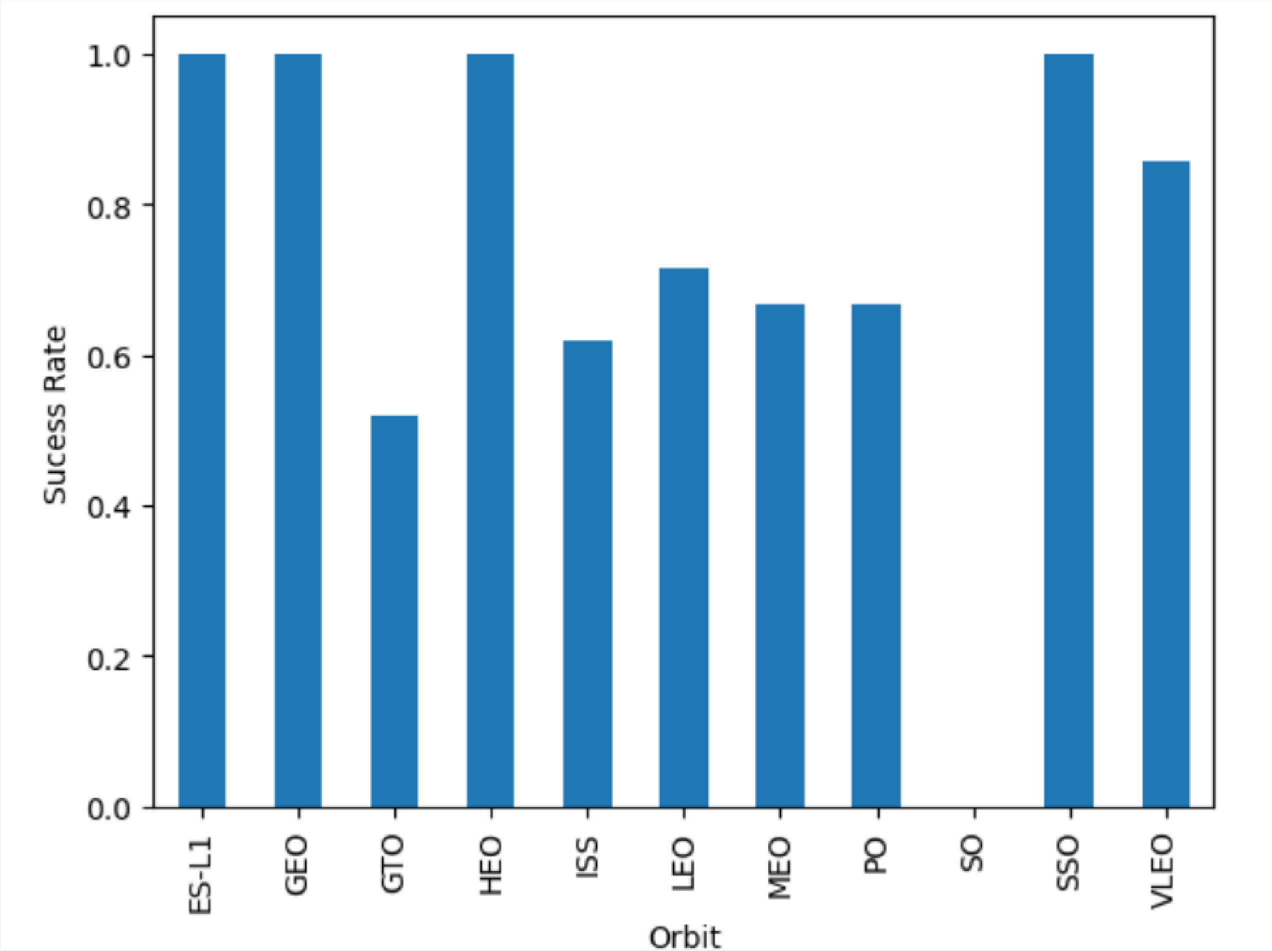
Payload vs. Launch Site



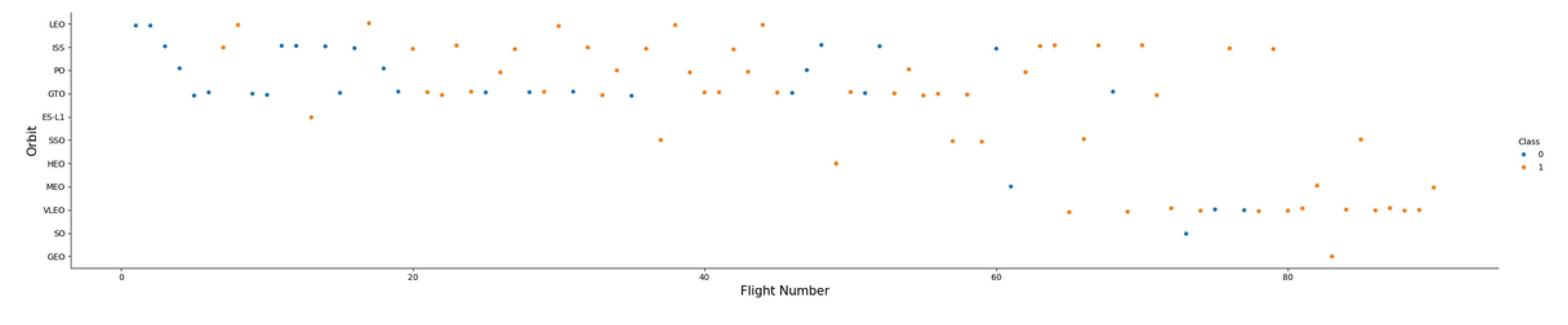
- Launch sites with a larger payload mass have a higher success rate
- Majority of launches with payloads >= 6500 were successful, with few outliers
- KSC LC 39A almost 100% successful after 6500 payload

Success Rate vs. Orbit Type

- ES-L1, GEO, HEO have 100% success rates.
  - SSO is very close
- SO has a 0% success rate – recommend removing orbit
- Remainder have greater than 50%

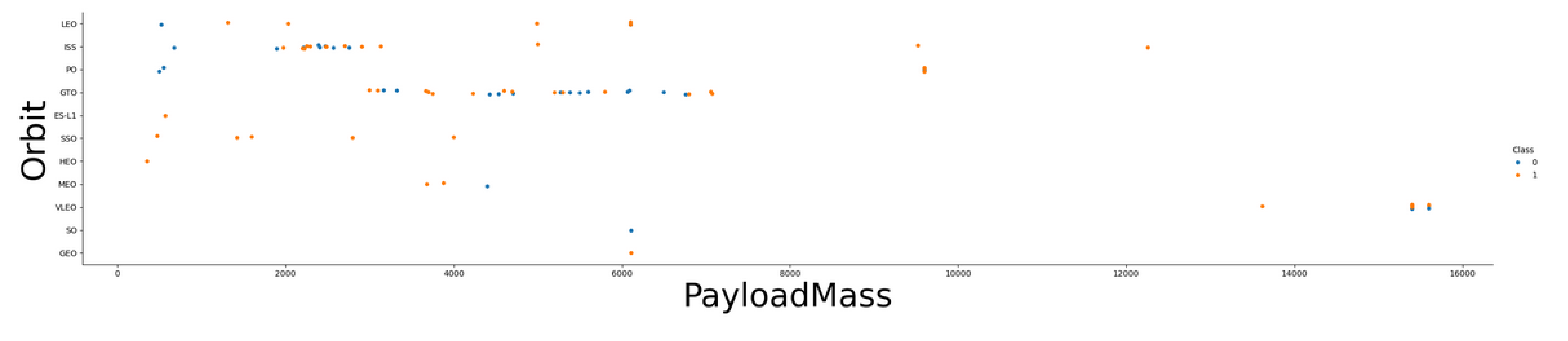


Flight Number vs. Orbit Type



- LEO Orbit success appears related to the number of flights
- No relationship gleamed from GTO Orbit

Payload vs. Orbit Type



- Heavy payloads impact the successful landing and positive landing rates for POLAR, LEO and ISS
- GTO has both positive and negative landing rates
- I used bigger font for the axes because I felt like it I guess

Launch Success Yearly Trend

- Notated that the success rate since 2013 kept increasing until 2020
- Unable to develop chart to reflect these findings due to error

All Launch Site Names

```
%sql SELECT DISTINCT "LAUNCH_SITE" FROM SPACEXTBL
```

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

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

- SQL query that selects distinctly unique launch site names from the database

Launch Site Names Begin with 'CCA'

```
%sql SELECT * FROM SPACEXTBL WHERE "LAUNCH_SITE" LIKE '%CCA%' LIMIT 5

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

Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Orbit	Customer	Mission_Outcome	Landing_Outc
2010-06-04	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success	Failure (paracl
2010-12-08	15:43:00	F9 v1.0 B0004	CCAFS LC-40	Dragon demo flight C1, two CubeSats, barrel of Brouere cheese	0	LEO (ISS)	NASA (COTS) NRO	Success	Failure (paracl
2012-05-22	7:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success	No atte
2012-10-08	0:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500	LEO (ISS)	NASA (CRS)	Success	No atte
2013-03-01	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677	LEO (ISS)	NASA (CRS)	Success	No atte

- SQL query selects 5 rows from the database where the launch sites begin with the character string ‘CCA’

Total Payload Mass

```
%sql SELECT SUM("PAYLOAD_MASS__KG_") FROM SPACEXTBL WHERE "CUSTOMER" = 'NASA (CRS)'
```

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

SUM("PAYLOAD_MASS__KG_")
45596

- SQL query calculates the sum of the payload mass column where the customer is NASA (CRS)

Average Payload Mass by F9 v1.1

- SQL query calculates the average of the payload mass from the database where the booster version is F9 v1.1

```
%sql SELECT AVG("PAYLOAD_MASS__KG_") FROM SPACEXTBL WHERE "BOOSTER_VERSION" LIKE '%F9 v1.1%'
```

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

AVG("PAYLOAD_MASS__KG_")
2534.6666666666665

First Successful Ground Landing Date

- SQL query derives the first successful landing date by leveraging the "min" function

```
%sql SELECT MIN("DATE") FROM SPACEXTBL WHERE "Landing_Outcome" LIKE '%Success%'
```

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

MIN("DATE")
2015-12-22

Successful Drone Ship Landing with Payload between 4000 and 6000

- SQL query selects the booster version column from the database where the landing outcome column reflects “success (drone ship), but only if the Payload column is between 4000 and 6000

```
%sql SELECT "BOOSTER_VERSION" FROM SPACEXTBL WHERE "Landing_Outcome" = 'Success (drone ship)' AND "PAYLOAD_MASS__KG_" > 400(
```

```
 * 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

- SQL query selects the count of how many successful mission outcomes there were, along with how many failures

Task 7

List the total number of successful and failure mission outcomes

```
%sql SELECT (SELECT COUNT("MISSION_OUTCOME") FROM SPACEXTBL WHERE "MISSION_OUTCOME" LIKE '%Success%') AS SUCCESS, (SELECT CC
```

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

SUCCESS	FAILURE
100	1

Boosters Carried Maximum Payload

- SQL query selects unique booster version rows from the database where the payload column is the maximum value

```
%sql SELECT DISTINCT "BOOSTER_VERSION" FROM SPACEXTBL WHERE "PAYLOAD_MASS__KG_" = (SELECT max("PAYLOAD_MASS__KG_") FROM SPAC
```

```
 * 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

```
%sql SELECT substr("DATE", 6, 2) AS MONTH, "BOOSTER_VERSION", "LAUNCH_SITE" FROM SPACEXTBL WHERE "LANDING_OUTCOME" = 'Failure (drone ship)' AND substr("DATE", 0, 5) = '2015'
```

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

MONTH	Booster_Version	Launch_Site
01	F9 v1.1 B1012	CCAFS LC-40
04	F9 v1.1 B1015	CCAFS LC-40

- SQL query shows the month, booster version and launch sites with failed landings in 2015

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

```
%sql SELECT "LANDING_OUTCOME", COUNT("LANDING_OUTCOME") FROM SPACEXTBL WHERE "DATE" >= '2010-06-04' AND "DATE" <= '2017-03-20' AND "LANDING_OUTCOME" LIKE '%Success%' GROUP BY "LANDING_OUTCOME" ORDER BY COUNT("LANDING_OUTCOME") DESC
```

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

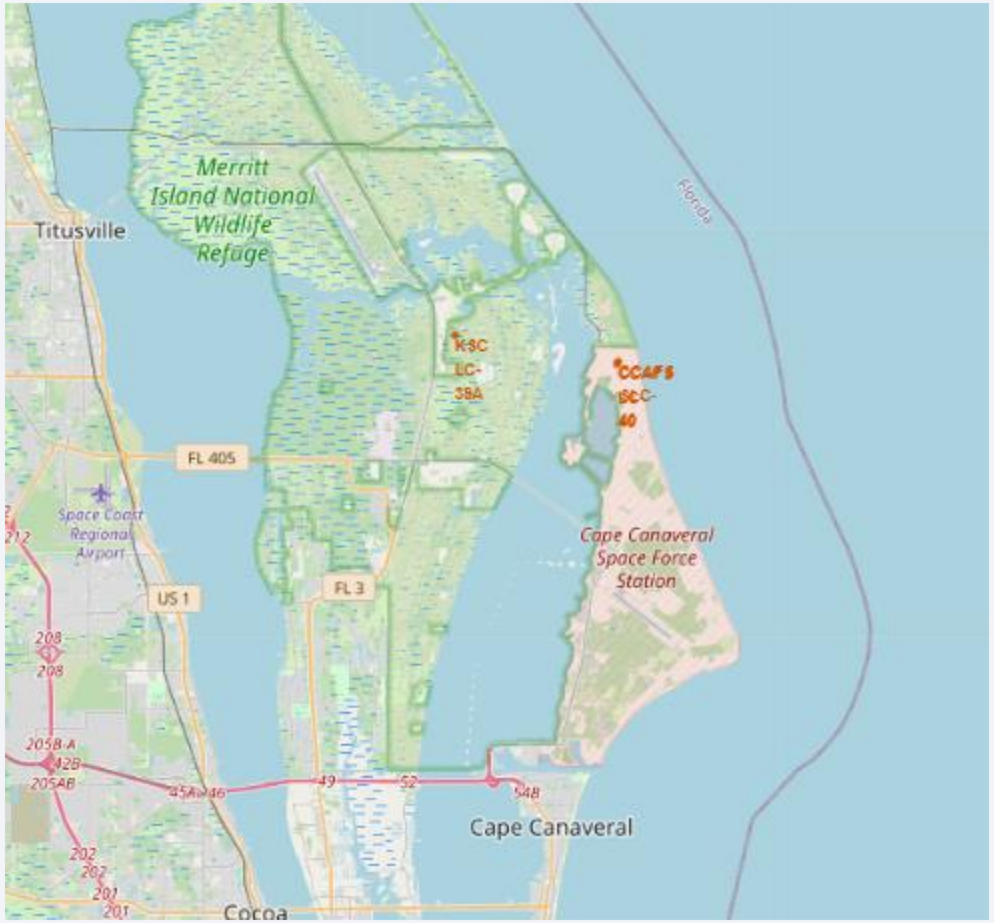
Landing_Outcome	COUNT("LANDING_OUTCOME")
Success (drone ship)	5
Success (ground pad)	3

- 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

Section 3

Launch Sites Proximities Analysis

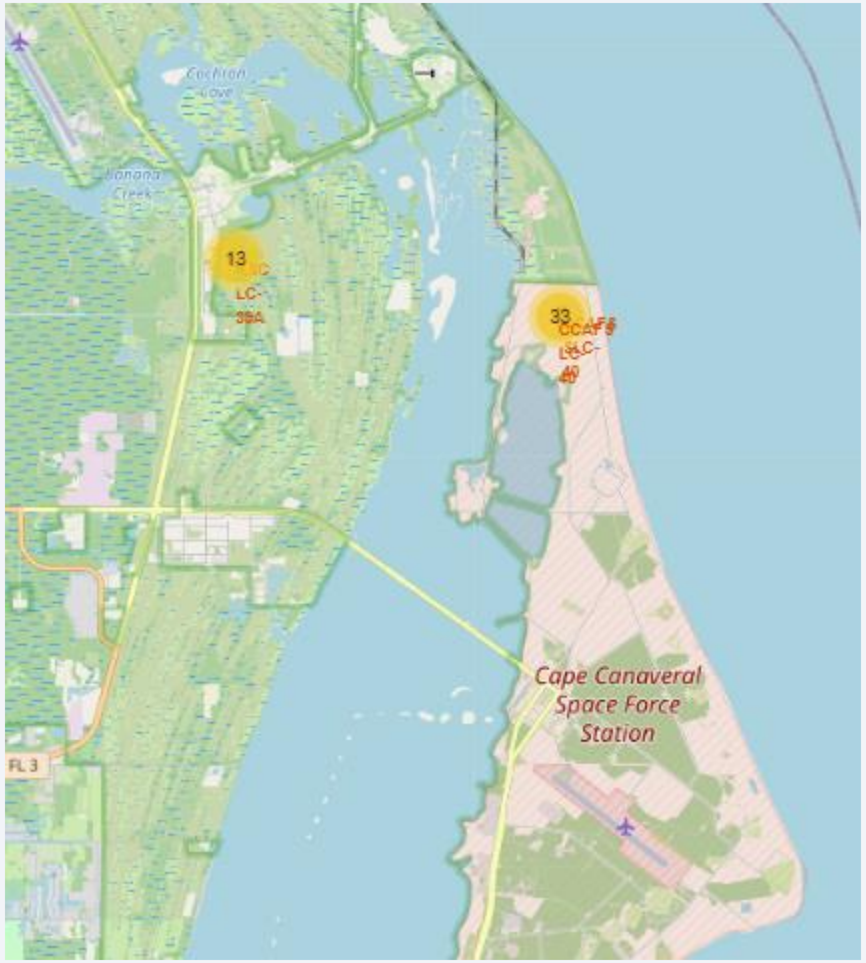
Folium Map 1

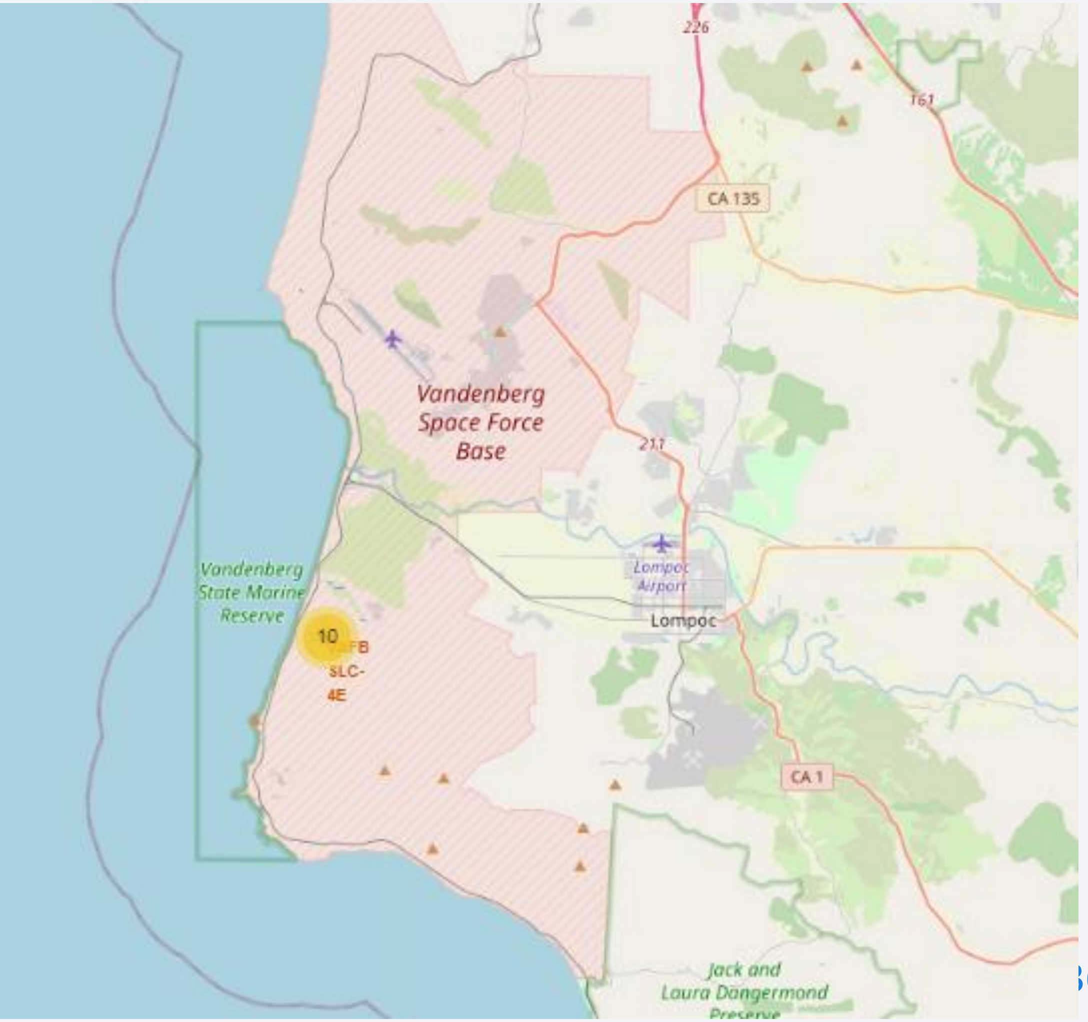


Merritt Island National Wildlife Refuge
Titusville
KSC LC-39A
CCAFS SLC-40
FL 405
Space Coast Regional Airport
Cape Canaveral Space Force Station
US 1
FL 3
Cape Canaveral
Cocoa

Vandenberg Space Force Base
Vandenberg State Marine Reserve
Lompoc Airport
Lompoc
VAFB SLC-4E
Jack and

Folium Map 2


13
LC-
39A
33
Cape Canaveral
Space Force
Station
FL 3


Vandenberg
Space Force
Base
Vandenberg
State Marine
Reserve
10
SLC-
4E
Lompoc
Airport
Lompoc
CA 135
CA 1
226
161
Jack and
Laura Dangermond

Folium Map 3

Section 4

Build a Dashboard with Plotly Dash

Plotly Code

- Provided software did not work – link below to code

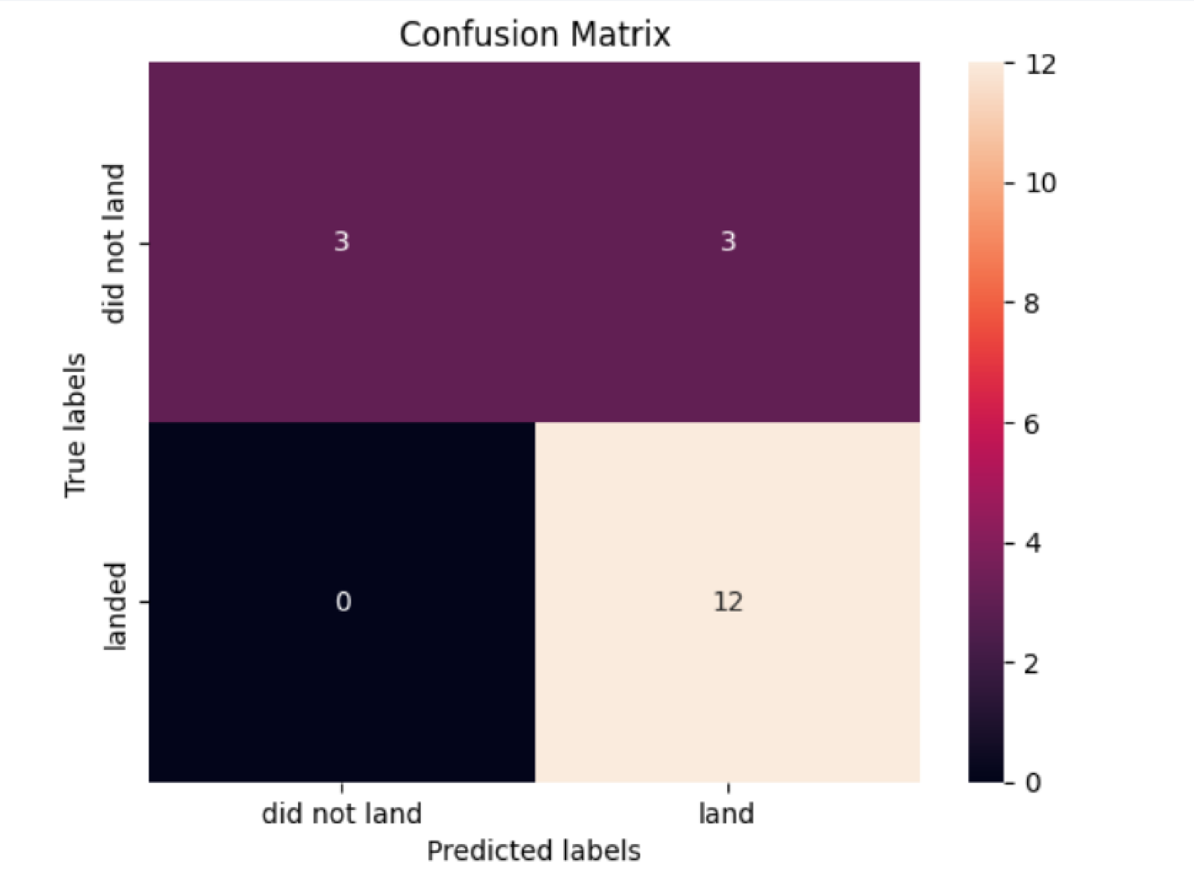
Link to code

Section 5

Predictive Analysis (Classification)

Confusion Matrix

- SVM model had the most accurate confusion matrix, with an accuracy of 83.33%



Conclusions

- We’ve leveraged many data science techniques to delve deeper into the relevant data to determine how to approach our competition with a price that cannot be beaten, while ensuring our data backs up our claims pertaining to first stage success of launch.

Thank you!