IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Erik Tran

3 March 2024

Outline

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

Executive Summary

- The goal was to build a model to predict if the first stage of a given Falcon 9 rocket launch would successfully land in order to determine cost of launch.
- Data from publically available SpaceX Falcon 9 historical launches was collected, wrangled, explored and fitted using various Python sklearn machine learning classification models.
- The final selected model performs well to predict successful launches. However the model tends to be optimistic on successful launches, resulting in false positives. Accuracy score 0.83

Introduction

- Our goal is to predict if the SpaceX Falcon 9 first stage will land successfully. SpaceX 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 SpaceX 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 SpaceX for a rocket launch.
- The main question that we are trying to answer is, for a given set of features about a Falcon 9 rocket launch which include its payload mass, orbit type, launch site, and so on, will the first stage of the rocket land successfully?

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - Webscrape launch data from SpaceX website & Wikipedia
- Perform data wrangling
  - Clean up data to focus on whether the booster successfully landed or not
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - Standardised data, perform GridSearchCV to determine best parameters for various classification models: Log regression, SVM, Decision tree, K-Nearest Neighbours

Data Collection

- Data collected from following sources:
  - SpaceX API of past launches (URL: https://api.spacexdata.com/v4/launches/past)
  - Webscraping SpaceX Falcon 9 and Falcon Heavy historical launch data from Wikipedia (URL: https://en.wikipedia.org/wiki/List_of_Falcon_9_and_Falcon_Heavy_launches)

Data Collection – SpaceX API

Request and parse the SpaceX launch data using the GET request

Filter the dataframe to only include Falcon 9 launches

Dealing with Missing Values:

Payload Mass: Replace NaN with Mean

- Github URL: https://github.com/ektran2002/IBMdatascience/blob/347d09f03f4f5c03dab7107dad8912ca0f4e1e6c/jupyter-labs-spacex-data-collection-api.ipynb

Data Collection - Scraping

Request the Falcon9 Launch Wiki page from its URL

Extract all column/variable names from the HTML table header

Create a data frame by parsing the launch HTML tables

- Github URL: https://github.com/ektran2002/IBMdatascience/blob/347d09f03f4f5c03dab7107dad8912ca0f4e1e6c/jupyter-labs-webscraping.ipynb

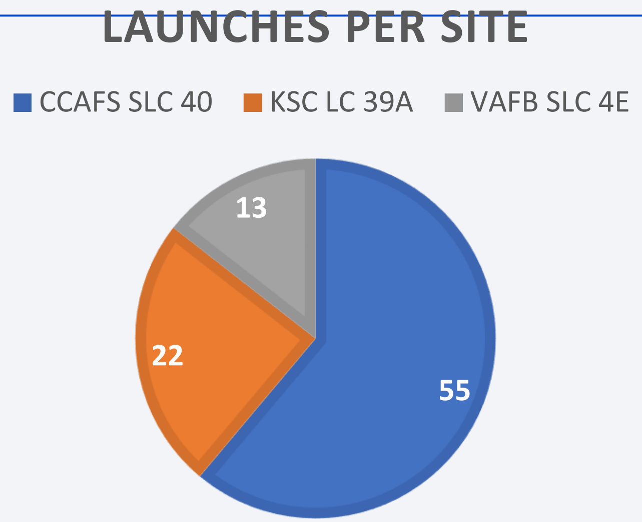
Data Wrangling

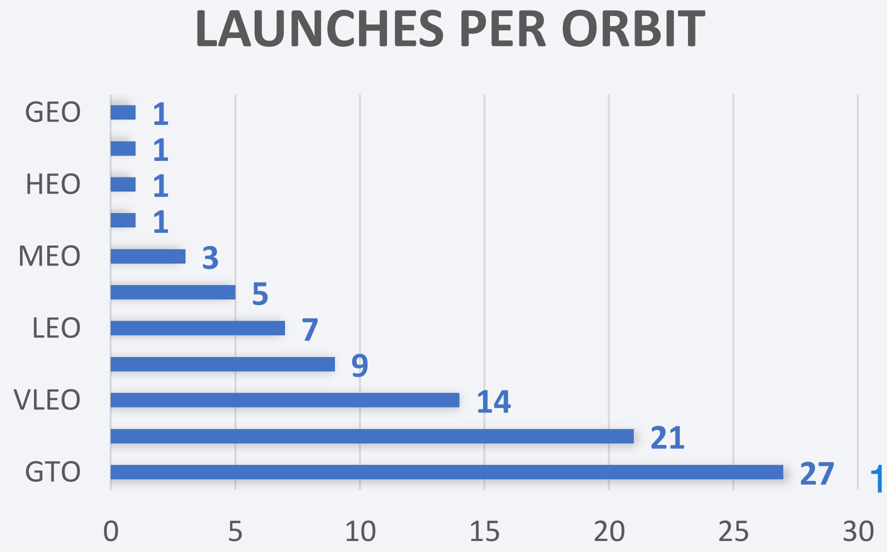
1. Calculate the number of launches on each site
2. Calculate the number and occurrence of each orbit
3. Calculate the number and occurence of mission outcome of the orbits:

    1: First stage successfully landed

    0: First stage landing was unsuccessful

- Github URL: https://github.com/ektran2002/IBMdatascience/blob/347d09f03f4f5c03dab7107dad8912ca0f4e1e6c/labs-jupyter-spacex-Data%20wrangling.ipynb





EDA with Data Visualization

- Scatter plot: Outcome class vs Flight Number & Launch Site
- Scatter plot: Outcome class vs Payload and Launch Site
- Bar plot: Outcome vs Orbit
- Cat plot: Outcome class vs Flight Number and Orbit
- Line plot: Success rate vs Year

- Github URL: https://github.com/ektran2002/IBMdatascience/blob/347d09f03f4f5c03dab7107dad8912ca0f4e1e6c/jupyter-labs-eda-dataviz.ipynb.jupyterlite.ipynb

EDA with SQL

Display the names of the unique launch sites in the space mission	select distinct Launch_Site from SPACEXTABLE
Display 5 records where launch sites begin with the string 'CCA'	select * from SPACEXTABLE where Launch_Site like "CCA%" limit 5
Display the total payload mass carried by boosters launched by NASA (CRS)	select sum(PAYLOAD_MASS__KG_) from SPACEXTABLE where Customer == "NASA (CRS)“
Display average payload mass carried by booster version F9 v1.1	**select avg**(PAYLOAD_MASS__KG_) **from** SPACEXTABLE **where** Booster_Version **like** "F9 v1.1%"
List the date when the first succesful landing outcome in ground pad was acheived.	select min(Date) from SPACEXTABLE where Landing_Outcome == "Success (ground pad)"
List the names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000	select distinct Booster_Version from SPACEXTABLE where Landing_Outcome == "Success (drone ship)" and PAYLOAD_MASS__KG_ > 4000 and PAYLOAD_MASS__KG_ < 6000
List the total number of successful and failure mission outcomes	select Mission_Outcome, count(*) from SPACEXTABLE group by Mission_Outcome
List the names of the booster_versions which have carried the maximum payload mass. Use a subquery	select Booster_Version, PAYLOAD_MASS__KG_ from SPACEXTABLE where PAYLOAD_MASS__KG_ == (select max(PAYLOAD_MASS__KG_) from SPACEXTABLE)
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.	select substr(Date,0,5) as year, substr(Date, 6,2) as month, Booster_Version, Launch_Site, Landing_Outcome from SPACEXTABLE where Landing_Outcome == "Failure (drone ship)" and substr(Date,0,5)='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.	**select** Landing_Outcome, **count**(*) **from** SPACEXTABLE **group by** Landing_Outcome **order by count**(*) **desc**

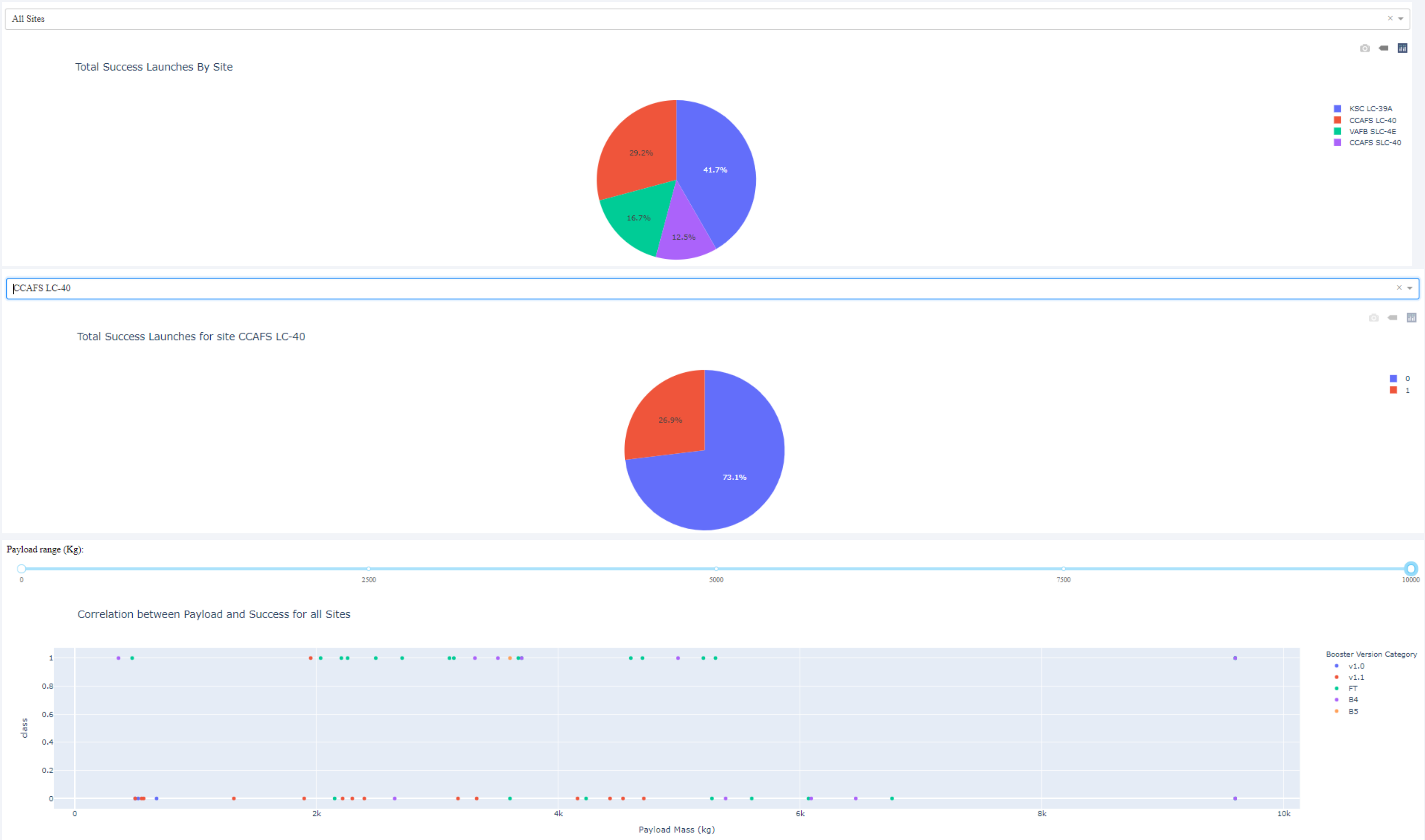
- Github URL: https://github.com/ektran2002/IBMdatascience/blob/cec52c827b465d31c1d9b38b348d381f8a560804/jupyter-labs-eda-sql-coursera_sqllite.ipynb

Build an Interactive Map with Folium

- TASK 1: Mark all launch sites on a map using Circle for Launch Site
- TASK 2: Mark the success/failed launches for each site on the map using Green Marker for Success and Red Marker for failed
- TASK 3: Calculate the distances between a launch site to its proximities using Polyline to indicate distance to nearest City, Highway, Rail and Coast

- Github URL: https://github.com/ektran2002/IBMdatascience/blob/347d09f03f4f5c03dab7107dad8912ca0f4e1e6c/lab_jupyter_launch_site_location.jupyterlite.ipynb

Build a Dashboard with Plotly Dash



- Github URL: https://github.com/ektran2002/IBMdatascience/blob/10a94197f4dbdd73b330584f4e09f07b577a948e/%5B7%5D%20spacex%20dash%20app.py

Predictive Analysis (Classification)

Assign Outcome Class to variable Y

StandardScaler fit to standardise data as X

Split into training and test datasets using train_test_split

Using Train data, Fit and find the best parameters using GridSearchCV for the following classifiers:

- Log regression
- SVM
- Decision Tree
- KNN

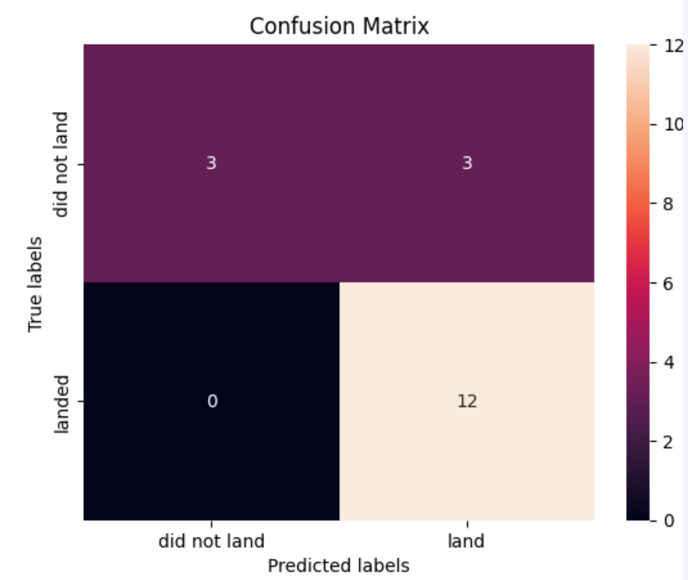
For each classifier, calculate accuracy score of the fit for Test data and visual with Confusion Matrix

Determine best classifier based on accuracy score

- Github URL: https://github.com/ektran2002/IBMdatascience/blob/347d09f03f4f5c03dab7107dad8912ca0f4e1e6c/SpaceX_Machine_Learning_Prediction_Part_5.jupyterlite.ipynb

Results

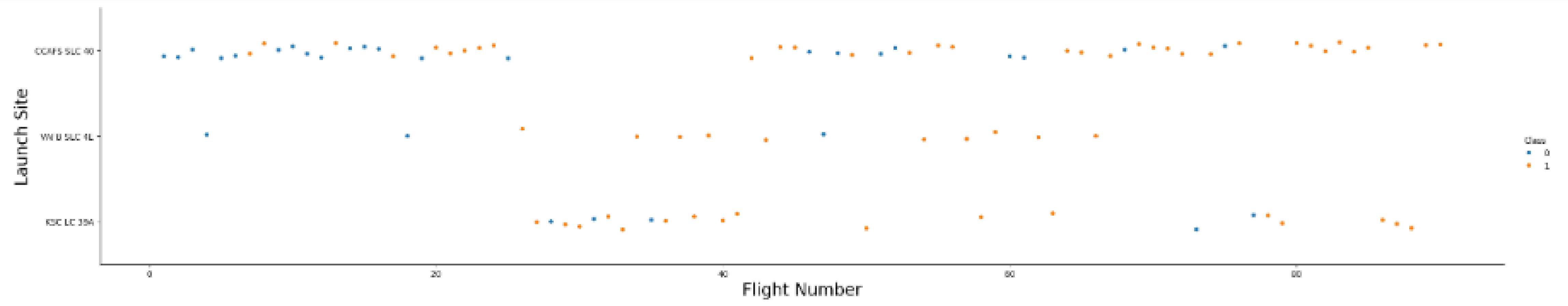
- Exploratory data analysis results:
  - as the flight number increases, the first stage is more likely to land successfully.
  - the more massive the payload, the less likely the first stage will return.
  - different launch sites have different success rates. CCAFS LC-40, has a success rate of 60 %, while KSC LC-39A and VAFB SLC 4E has a success rate of 77%.
  - for the VAFB-SLC launchsite there are no rockets launched for heavypayload mass(greater than 10000)
  - ES-L1, SSO, HEO, GEO orbits have the highest success rate (100% success)
  - 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
  - heavy payloads the successful landing or positive landing rate are more for Polar,LEO and ISSHowever for GTO we cannot distinguish this well as both positive landing rate and negative landing(unsuccessful mission) are both there here.
  - sucess rate since 2013 kept increasing till 2020
- Interactive analytics demo in screenshots
- Predictive analysis results:
  - All classifier methods (Log regression, SVM, Decision tree, KNN) gave equal score and predictions, therefore all performed equally



Section 2

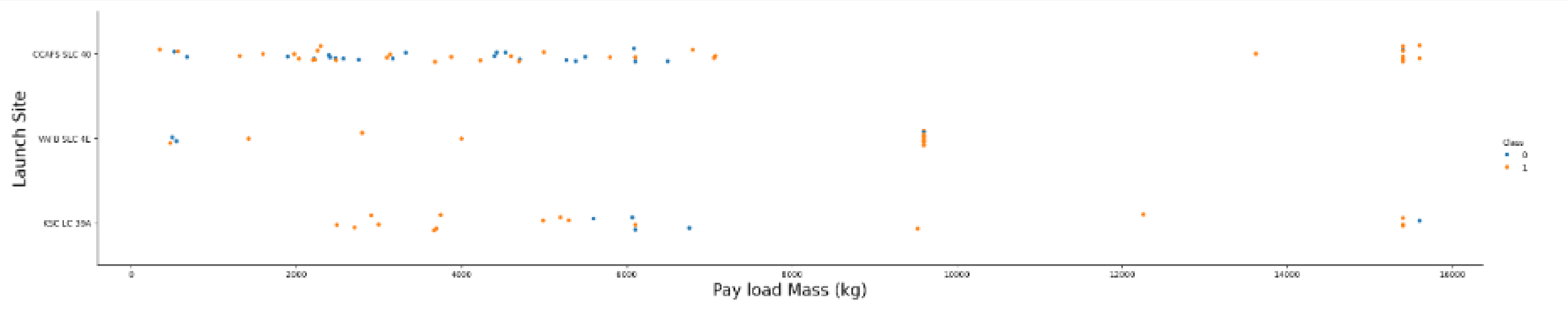
Insights drawn from EDA

Flight Number vs. Launch Site



- CCAFS LC-40 was the first launch site and the most frequent
- No recent launches at VAFB-SLC 4E
- KSC LC-39A launches started later

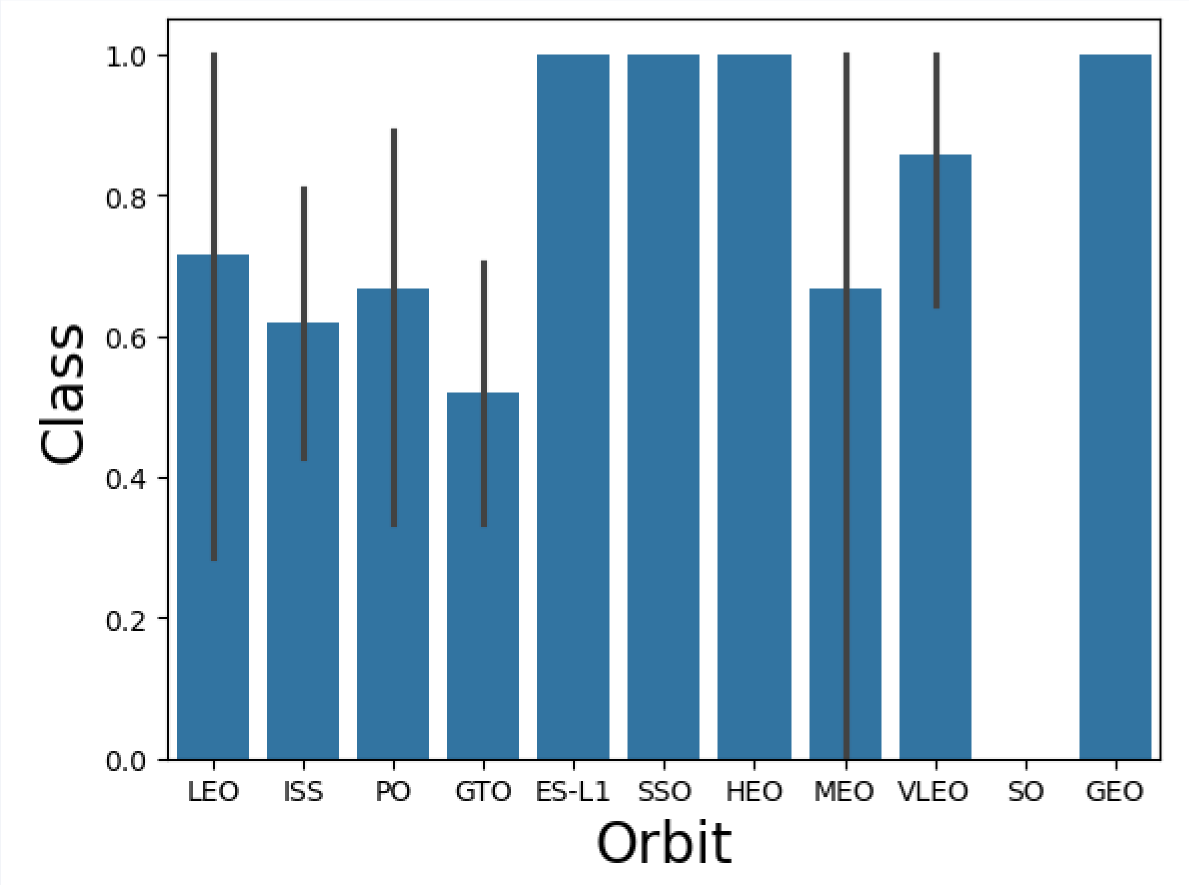
Payload vs. Launch Site



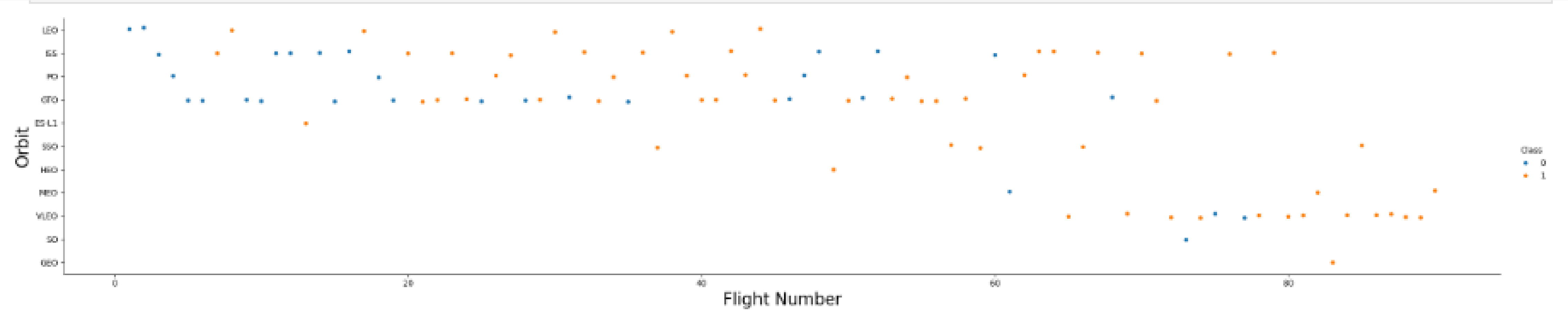
- VAFB-SLC launchsite there are no rockets launched for heavypayload mass(greater than 10000)

Success Rate vs. Orbit Type

- ES-L1, SSO, HEO, GEO has the highest (perfect) success rate
- GTO has the lowest success rate ~50%

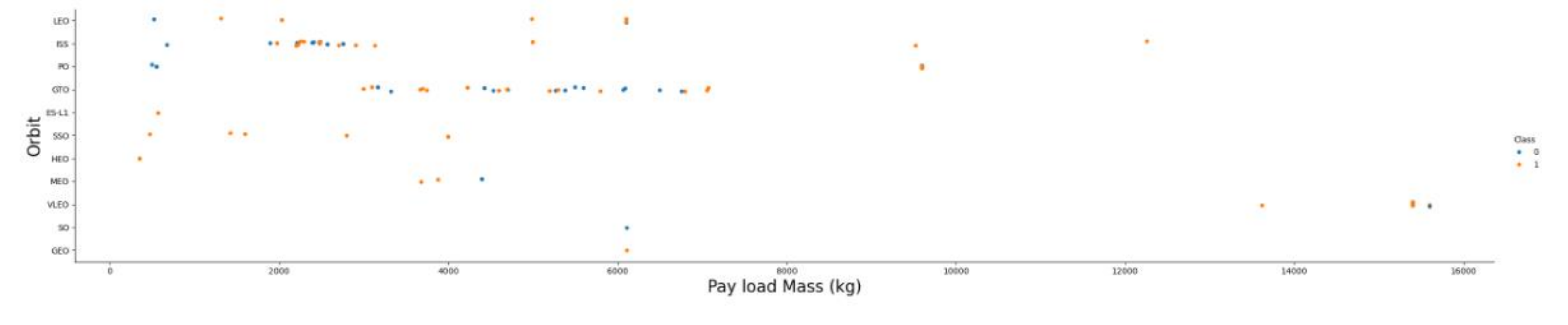


Flight Number vs. Orbit Type



- 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

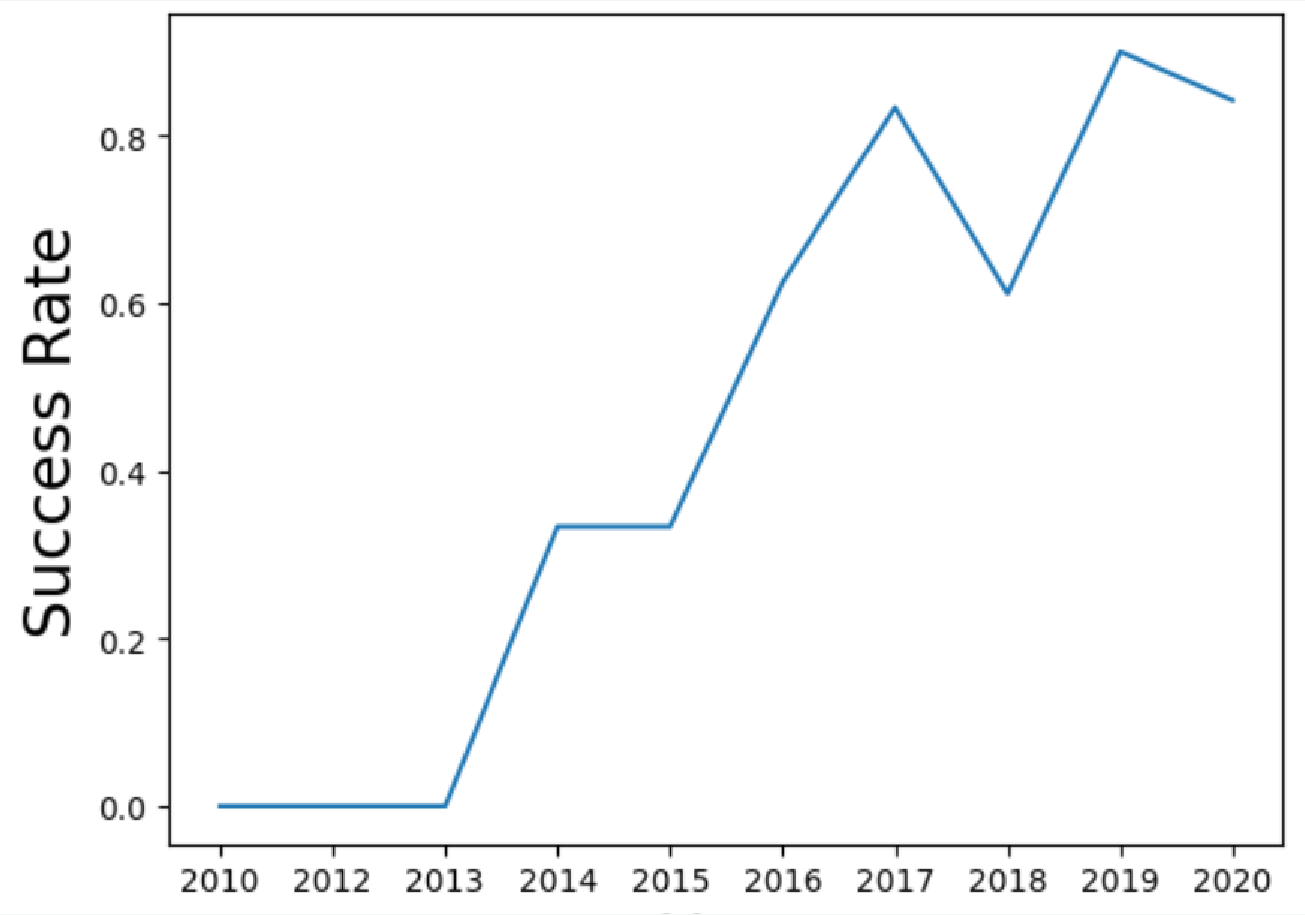
Payload vs. Orbit Type



- 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

- sucess rate since 2013 kept increasing till 2020



All Launch Site Names

```
%%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'

```
%%sql

select * from SPACEXTABLE 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_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

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

```
%%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

```
%%sql

select avg(PAYLOAD_MASS__KG_) from SPACEXTABLE where Booster_Version like "F9 v1.1%"
```

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

avg(PAYLOAD_MASS__KG_)
2534.6666666666665

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)"
```

```
 * 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

```
%%sql

select distinct Booster_Version from SPACEXTABLE
where Landing_Outcome == "Success (drone ship)" and PAYLOAD_MASS__KG_ > 4000 and  PAYLOAD_MASS__KG_ < 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

```
%%sql

select Mission_Outcome, count(*) from SPACEXTABLE group by Mission_Outcome
```

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

Mission_Outcome	count(*)
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

```
%%sql
select Booster_Version, PAYLOAD_MASS__KG_ from SPACEXTABLE
where PAYLOAD_MASS__KG_ == (select max(PAYLOAD_MASS__KG_) from SPACEXTABLE)
```

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

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

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

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

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

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

```
%%sql
select Landing_Outcome, count(*) from SPACEXTABLE where Date between "2010-06-04" and "2017-03-20" group by Landing_Outcome
```

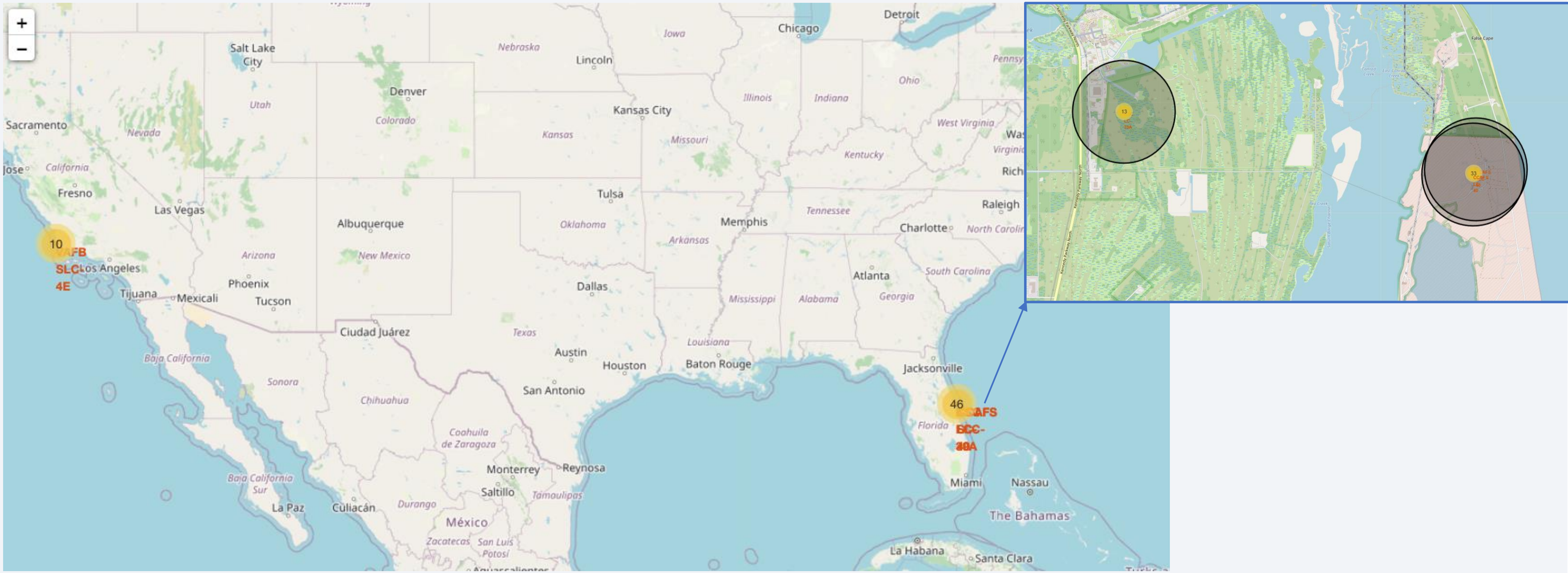
```
 * sqlite:///my_data1.db
Done.
```

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

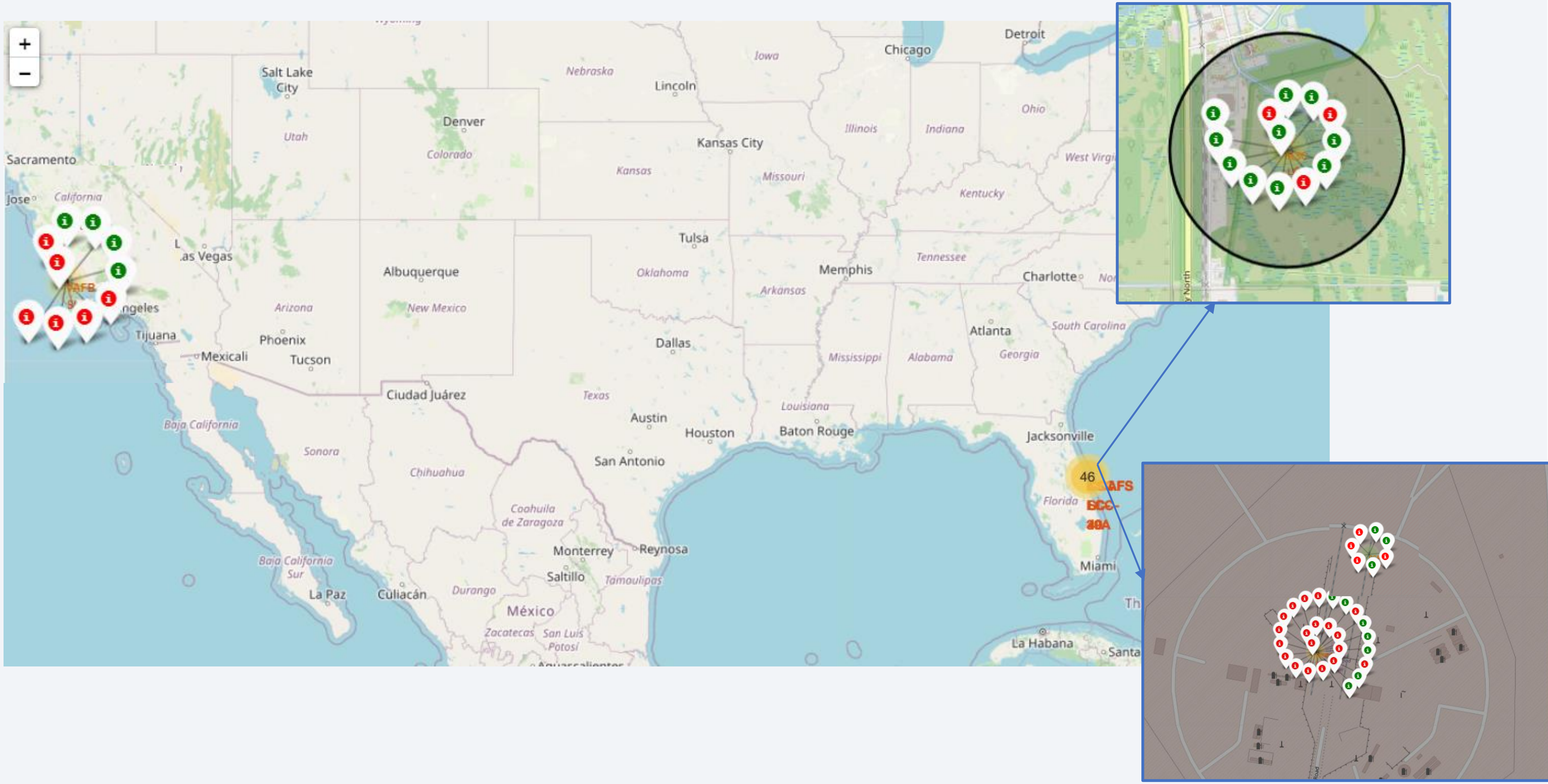
Section 3

Launch Sites Proximities Analysis

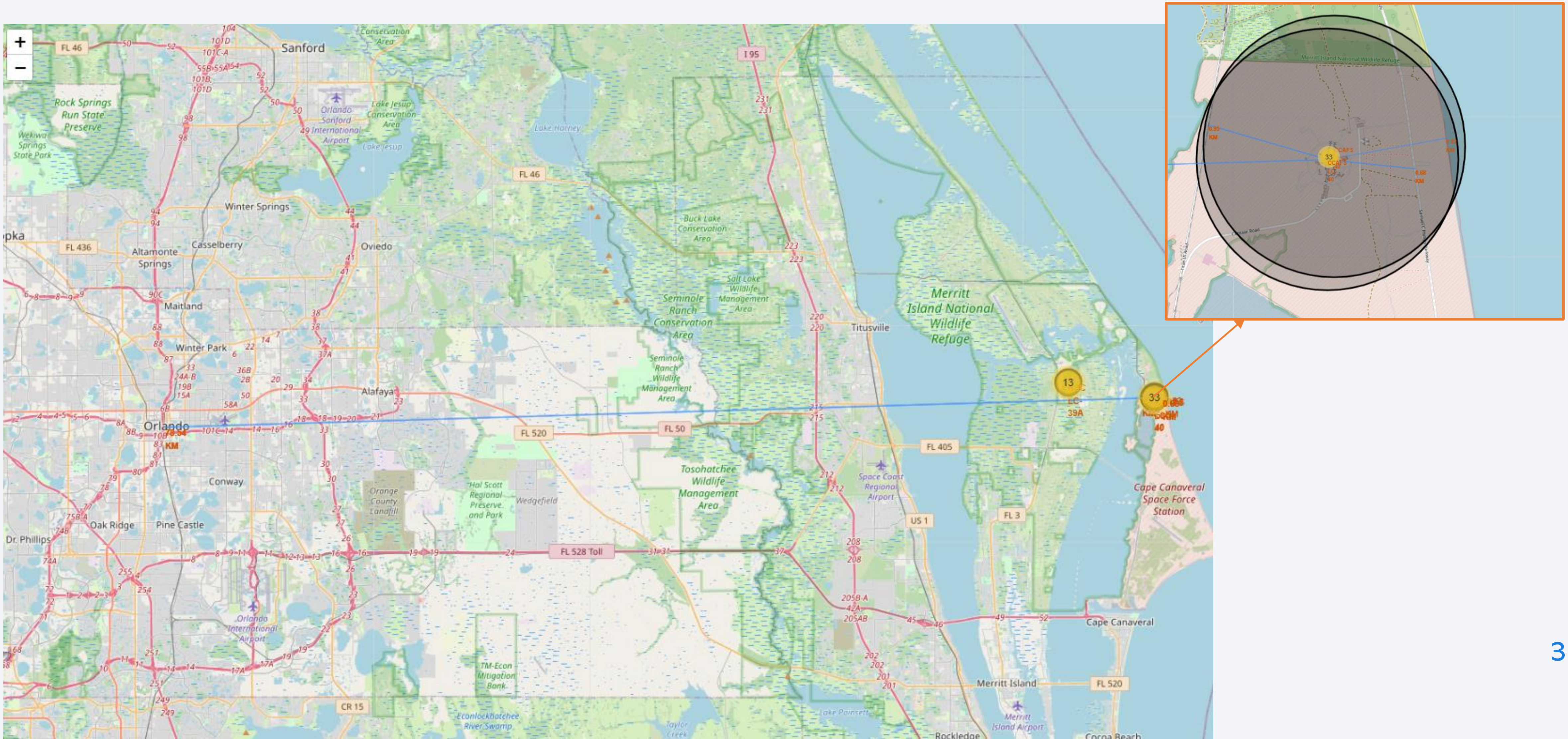
Map: Launch Sites



Launch Outcome per Site



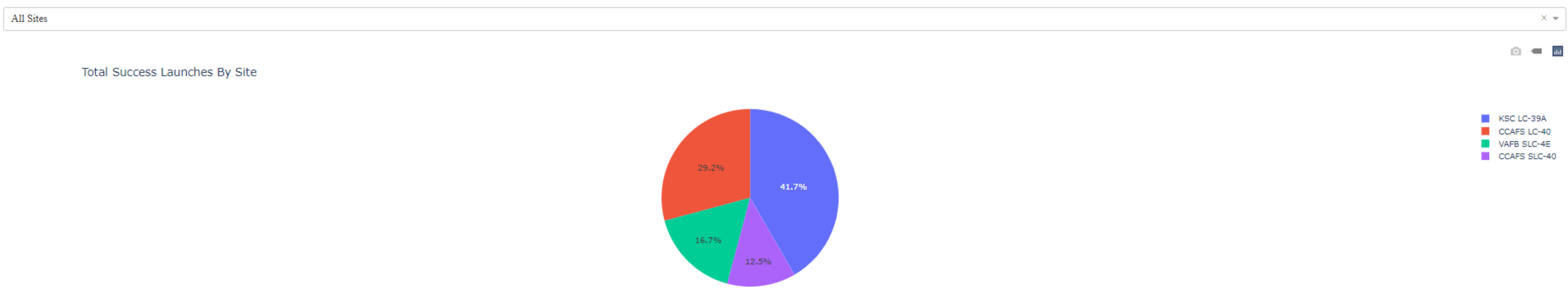
CCAFS LC-40 proximity to cities, highways, coast, rail



Section 4

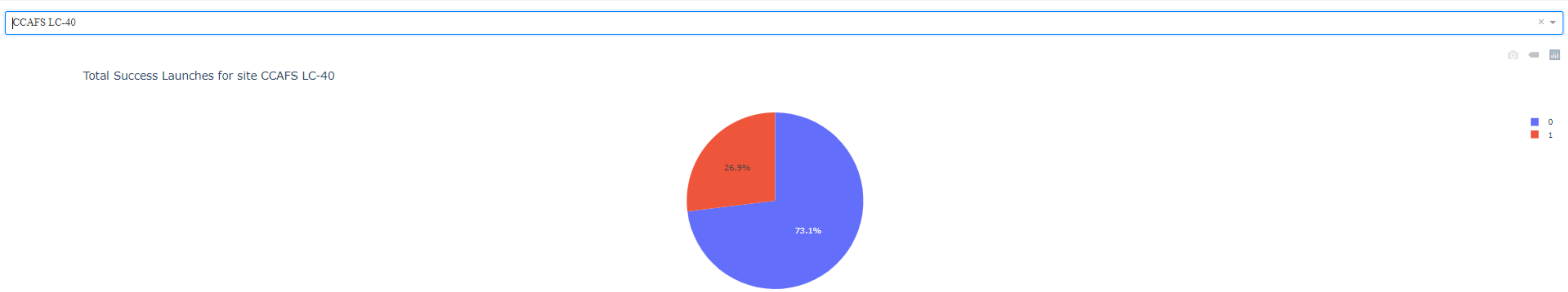
Build a Dashboard with Plotly Dash

Successful stage 1 landing for all sites

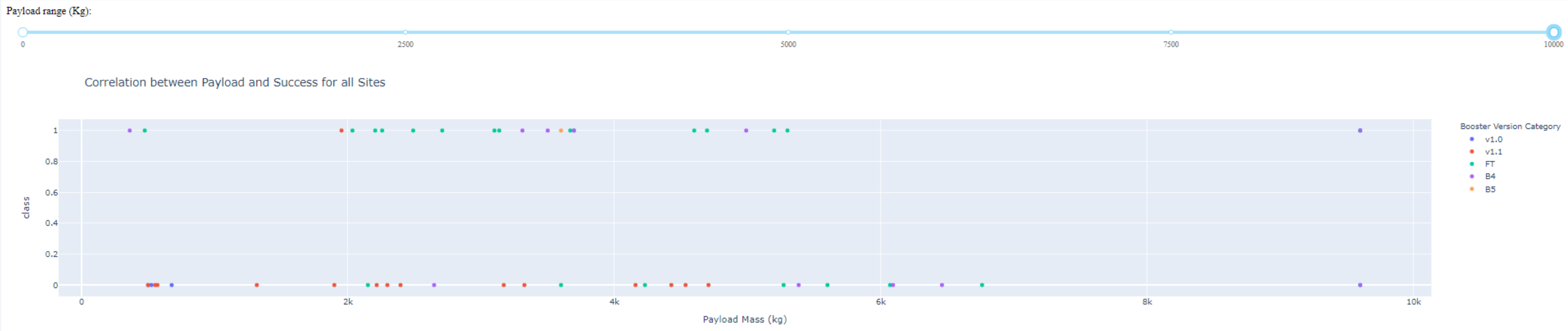


Success vs Failed stage 1 landings per site

For example, CCAFS LC-40 Launch site:



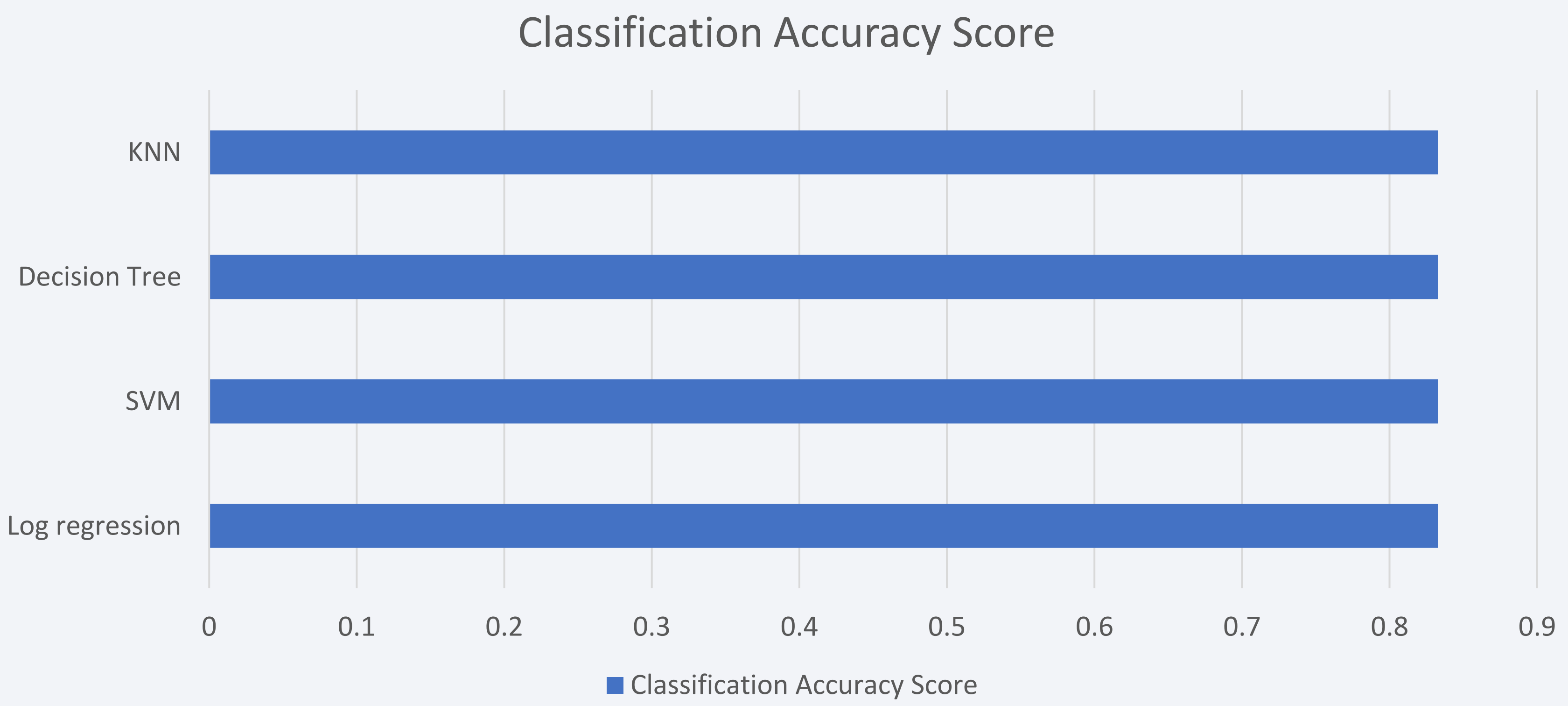
Payload vs. Launch Outcome scatter plot for all sites



Section 5

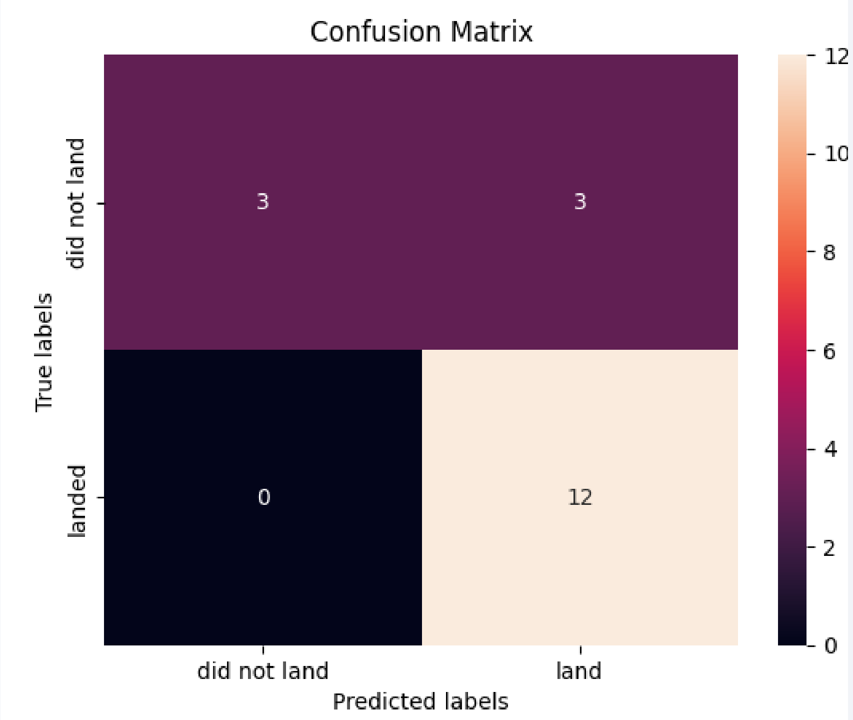
Predictive Analysis (Classification)

Classification Accuracy



- All 4 classification models had the same accuracy score 0.833, therefore can conclude that all models perform equally and any can be selected as the final model.

Confusion Matrix



- All models had the same confusion matrix
- The models correctly predicted all landed (successful flights)
- The models tends to be optimistic in predicting successful flights and results in false positives (predicted successful flights which actually failed)

Conclusions

- The final model performs well to predict successful launches
- However the model tends to be optimistic on successful launches, resulting in false positives. Accuracy score 0.83

Thank you!