Winning Space Race with Data Science

Francesco Bernasconi
01.05.2023

Outline

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

Executive Summary

- Supervised machine learning methods were employed to find the best model to estimate the success rate of SpaceX launches, depending on multiple variable such as location of the launch and payload mass
- A trained Decision Tree model algorithm showed accuracy of > 85%, false positive < 20% and false positive < 10% in estimating success of SpaceX launches on a selected test dataset
- We suggest using this Decision Tree model to predict the success rate of future SpaceX launches

Introduction

- Typical price for a single rocket launch value quote upwards of $165 millions
- SpaceX values one launch at $62 millions as it can reuse the first stage of its rockets
- The reuse of the first stage is based on a successful landing of the used first stage of the rocket
- We want to bid against SpaceX successful operation and we need a predictor for the successful landing of the first stage of the rocket
- We explore multiple machine learn algorithms and we train and test their predictive accuracy

Section 1

Methodology

Methodology

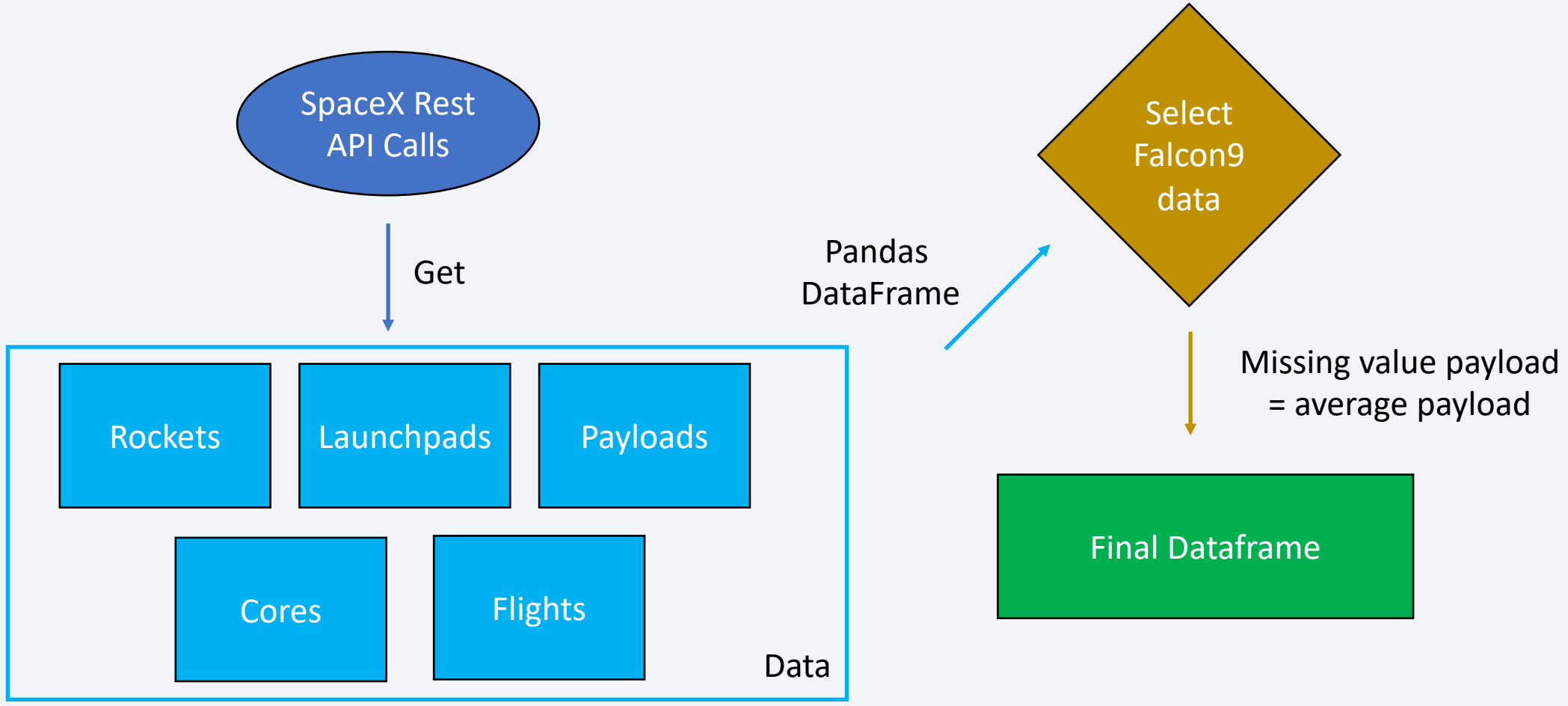
Executive Summary

- Data collection methodology:
  - Data were collect by calling SpaceX APIs
- Perform data wrangling
  - Data were selected, cleaned and prepared for machine learning use
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - Accuracy of the models were assessed by train-test splitting the data sets

Data Collection – SpaceX API

- Data collection by calls to SpaceX Rest API and creation of a Pandas Dataframe
- Data wrangling on the created Dataframe
- Python code for Data Collection can be found here: https://github.com/FrancescoBernasconi/IBM-Capston-Project-SpaceY/blob/main/1_jupyter-labs-spacex-data-collection-api.ipynb

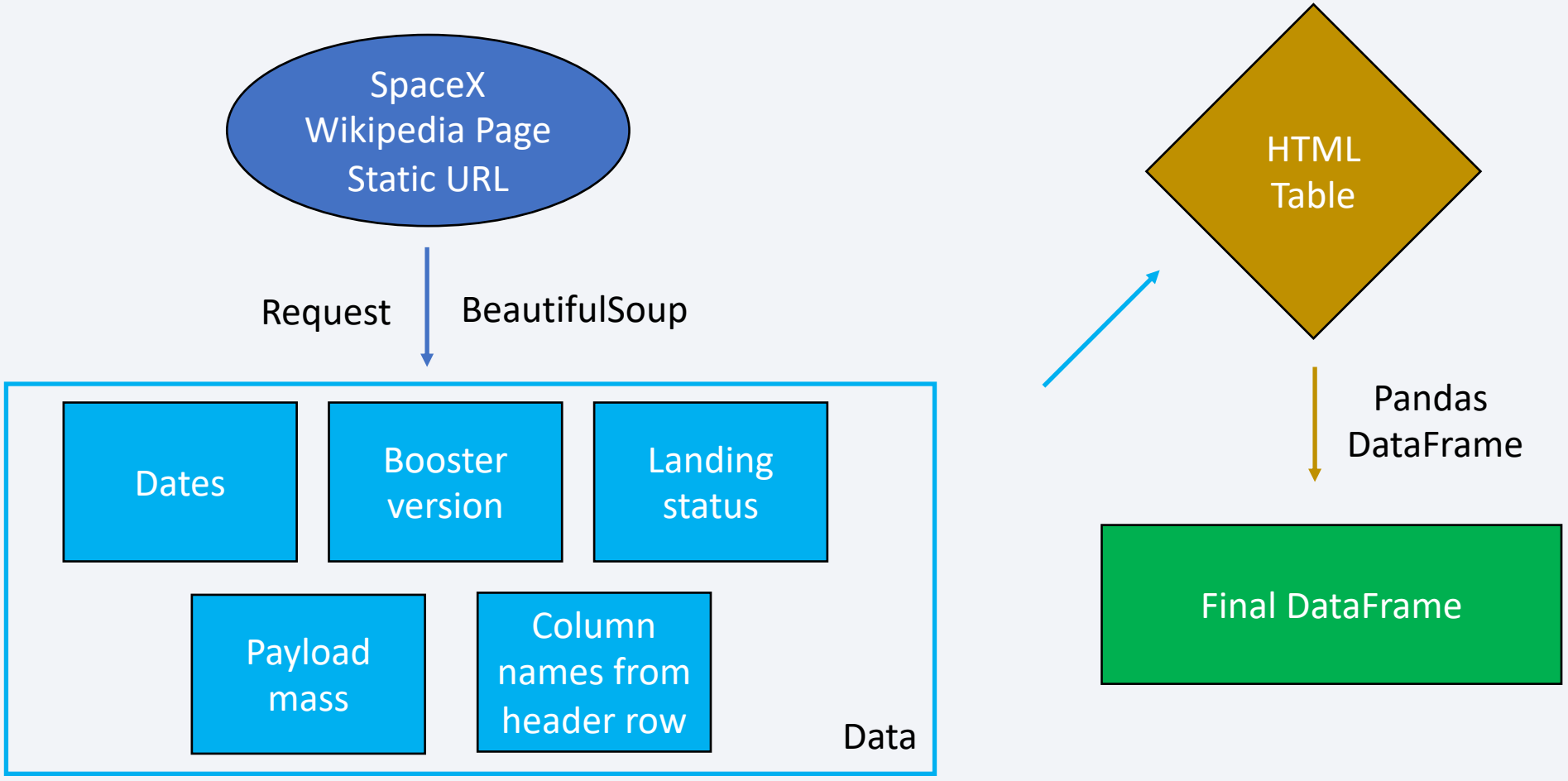
Data Collection – SpaceX API flowchart



Data Collection – Scraping

- The webscraping of SpaceX Wikipedia page was done through Requests and BeautifulSoup libraries.
- The static URL was converted to a Pandas DataFrame
- Python code for Data Scraping can be found here: https://github.com/FrancescoBernasconi/IBM-Capston-Project-SpaceY/blob/main/2_jupyter-labs-webscraping.ipynb

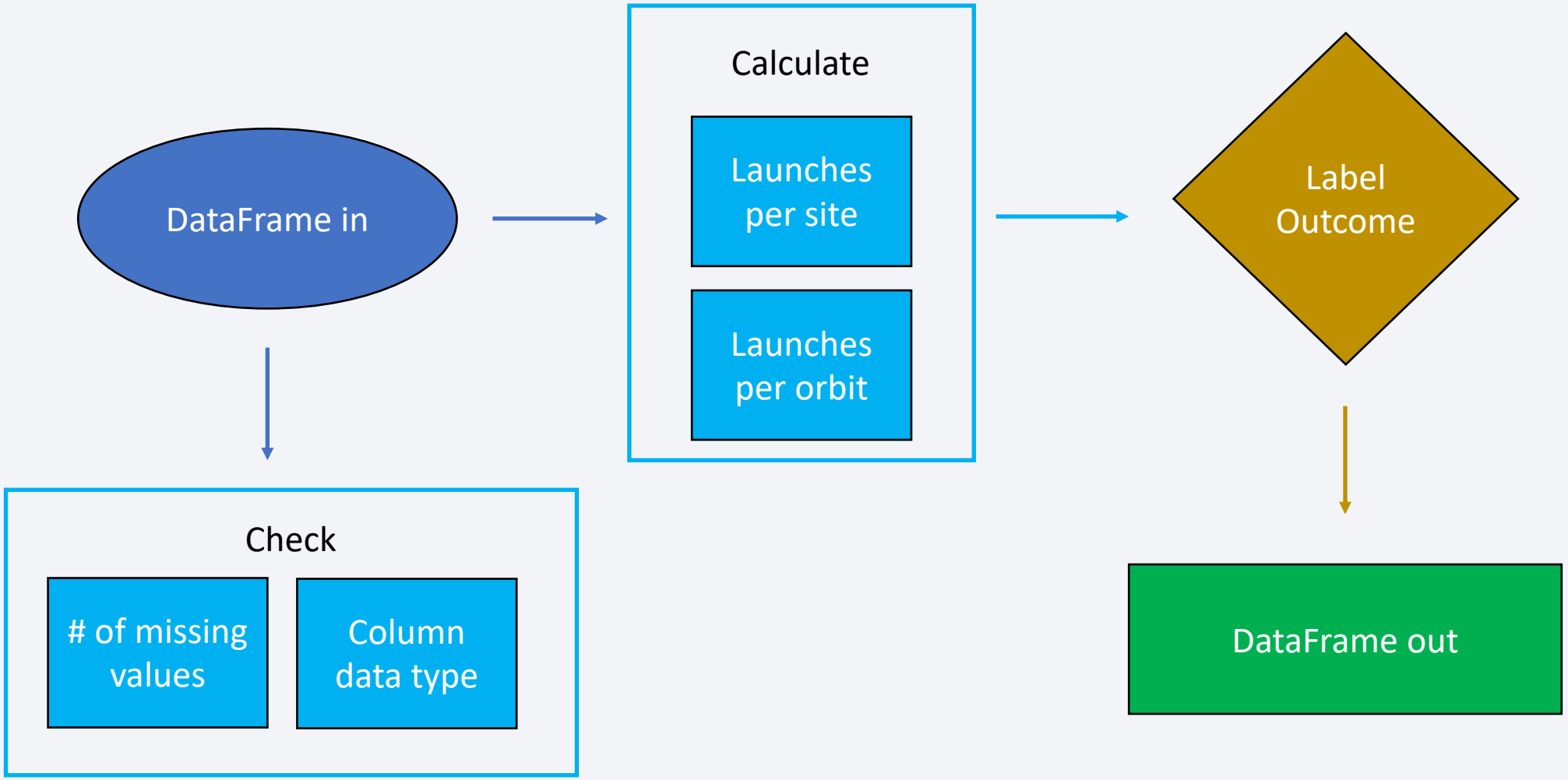
Data Collection – Scraping flowchart



Data Wrangling

- The data was processed by the following steps:
  - Identify missing value entries
  - Identify data type of each column of the DataFrame
  - Calculate number of launch on each site
  - Calculate number of and occurrence of each orbit
  - Add the mission outcome of each orbit type to the DataFrame as a column
- Python code for Data Wrangling can be found here: https://github.com/FrancescoBernasconi/IBM-Capston-Project-SpaceY/blob/main/3_labs-jupyter-spacex-data_wrangling_jupyterlite.jupyterlite.ipynb

Data Wrangling– Wrangling flowchart



EDA with Data Visualization

1. Flight number vs launch site: categorical plot fail/success
2. Payload vs launch site: categorical plot fail/success
3. Success rate vs orbit: identify most successful orbit target
4. Flight number vs orbit type: identify targeted orbit changed over development
5. Payload vs orbit type: identify which orbit is target when using a certain payload
6. Launch success yearly trend: observe evolution of success rate over development

- Python code for Data Visualization can be found here: https://github.com/FrancescoBernasconi/IBM-Capston-Project-SpaceY/blob/main/4_IBM-DS0321EN-SkillsNetwork_labs_module_2_jupyter-labs-eda-dataviz.ipynb.jupyterlite.ipynb

EDA with SQL

1. All launch site names
2. Launch site names begin with ‘KSL’ (5 record)
3. Total payload mass by boosters launches by NASA (CRS)
4. Average payload mass by F9 v1.1
5. First successful ground landing date
6. Success drone ship landing with payload between 4000 and 6000 kg

…

EDA with SQL

...

7. Total number of successful and failure mission outcomes
8. Boosters carried maximum payload
9. 2017 launch records
10. Ranking landing outcomes between 2010-06-04 and 2017-03-20

- Python code for EDA with SQL can be found here: https://github.com/FrancescoBernasconi/IBM-Capston-Project-SpaceY/blob/main/5_jupyter-labs-eda-sql-edx_sqllite.ipynb

Build an Interactive Map with Folium

- Map 1: Identify launch sites with Folium Maker and circle objects
- Map 2: Display launch outcome for each sites with Folium MarkerCluster
- Map 3: Identify a potential observation point on the map (coastline) with Folium MousePosition and add a visual line from the closest launch site
- Python code for interactive Map with Folium can be found here: https://github.com/FrancescoBernasconi/IBM-Capston-Project-SpaceY/blob/main/6_IBM-DS0321EN-SkillsNetwork_labs_module_3_lab_jupyter_launch_site_location.jupyterlite.ipynb

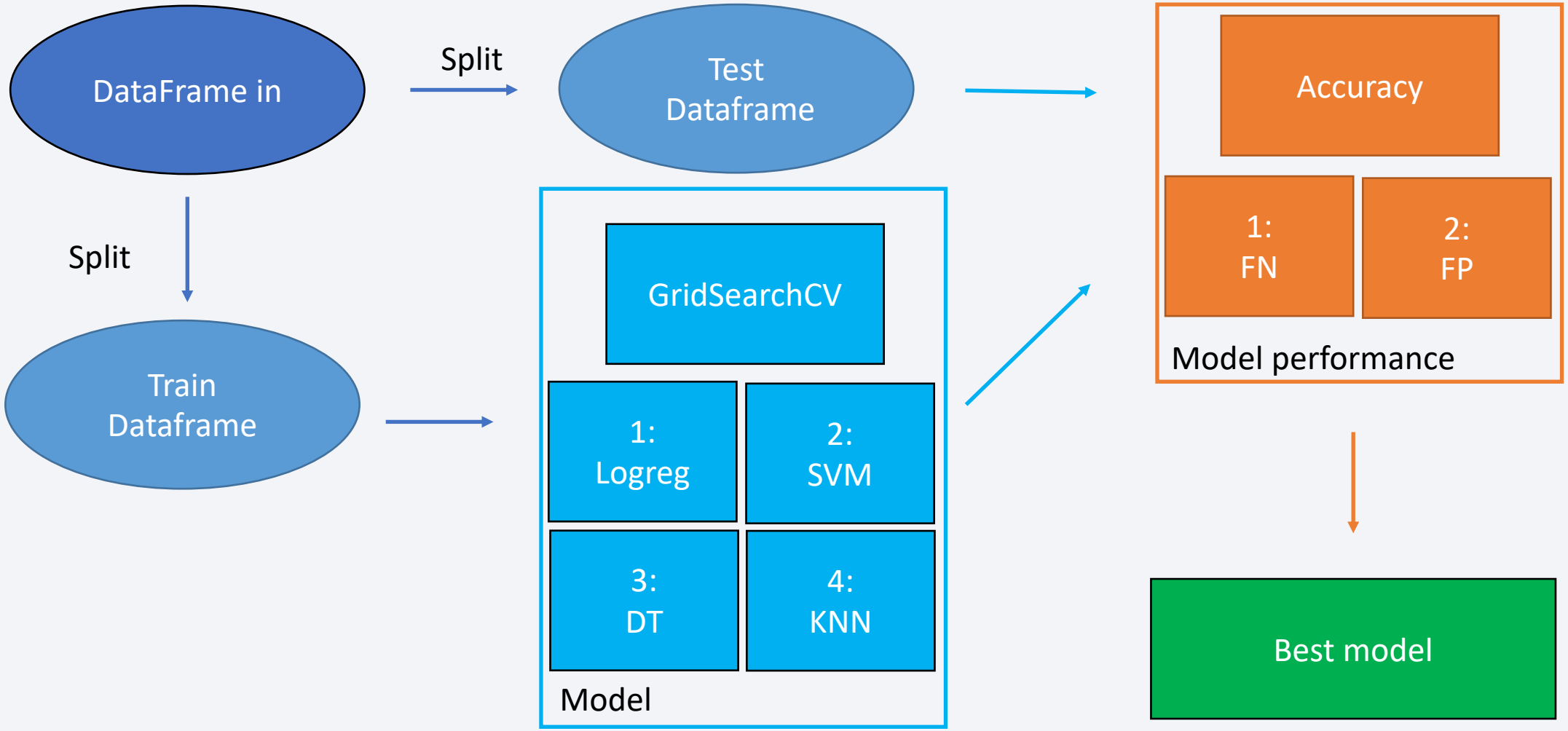
Build a Dashboard with Plotly Dash

- Pie Chart 1: Launch site identification for successful launches
- Pie Chart 2: Success ratio for most successful launch site
- Scatter plots 3-4: Booster type success ratio
- Python code for Dashboard with Plotly Dash can be found here: https://github.com/FrancescoBernasconi/IBM-Capston-Project-SpaceY/blob/main/7_spacex_dash_app.py

Predictive Analysis (Classification)

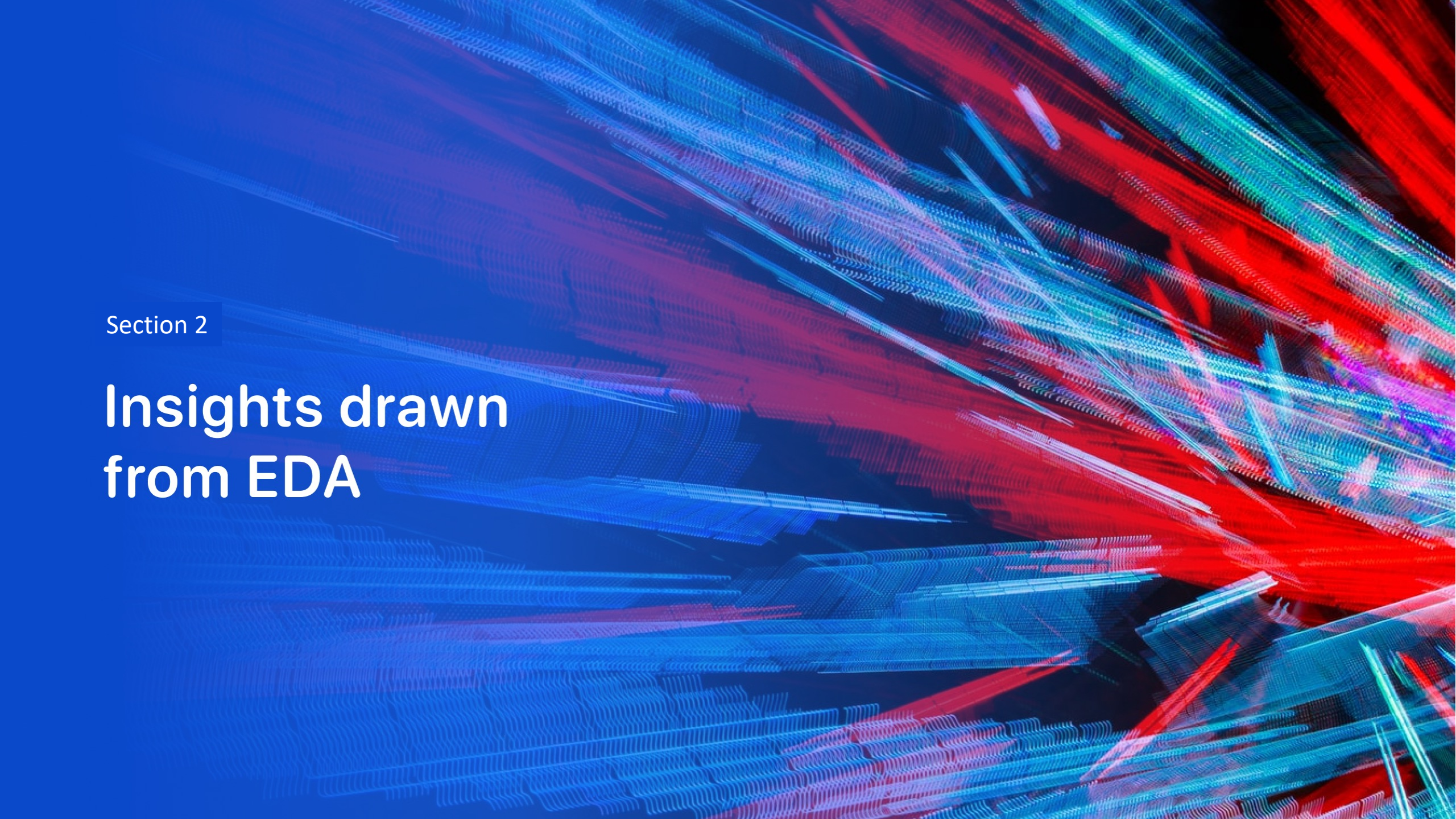
- Scale data and split it into train and test data subset
- Perform GridSearchCV for four machine learning models on train data subset
- Assess precision and errors of models on test data subset
- Python code for Predictive Analysis can be found here: https://github.com/FrancescoBernasconi/IBM-Capston-Project-SpaceY/blob/main/8_IBM-DS0321EN-SkillsNetwork_labs_module_4_SpaceX_Machine_Learning_Prediction_Part_5.jupyterlite.ipynb

Predictive Analysis (Classification) - flowchart

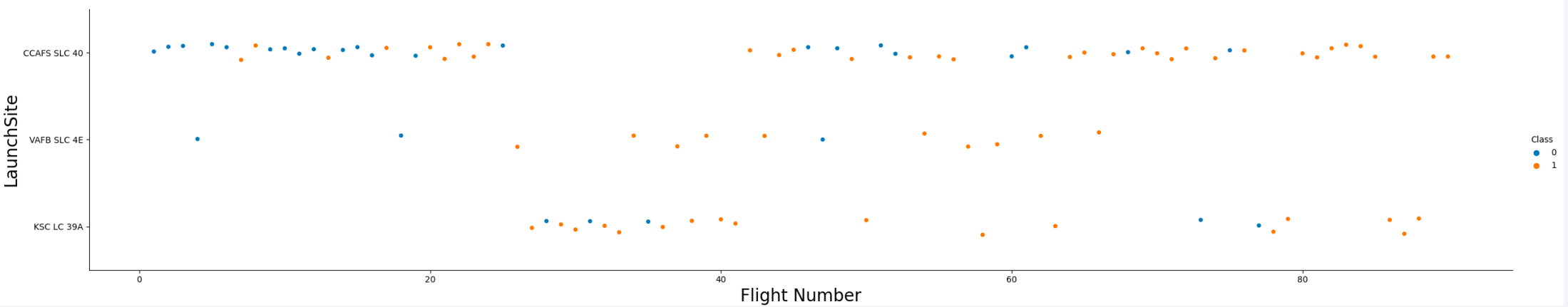


Results

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

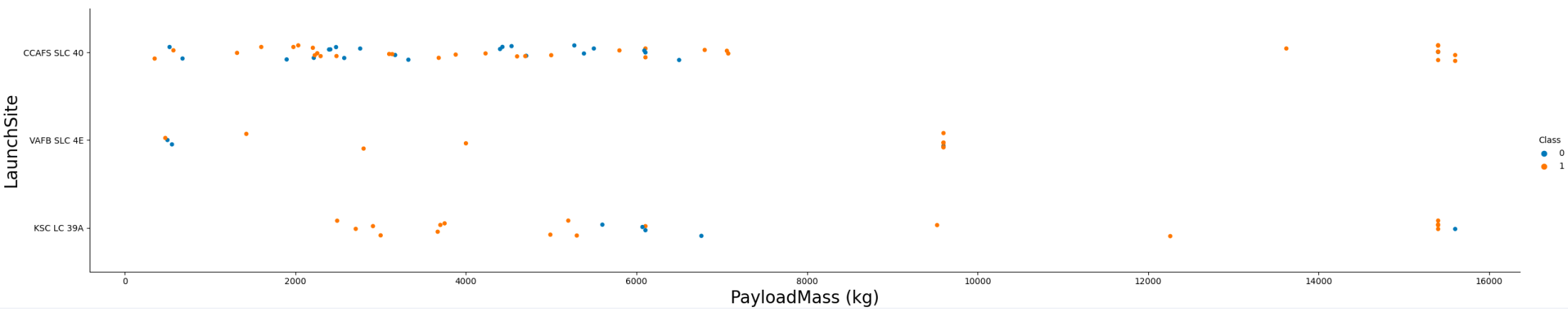

Section 2
Insights drawn from EDA

Flight Number vs. Launch Site



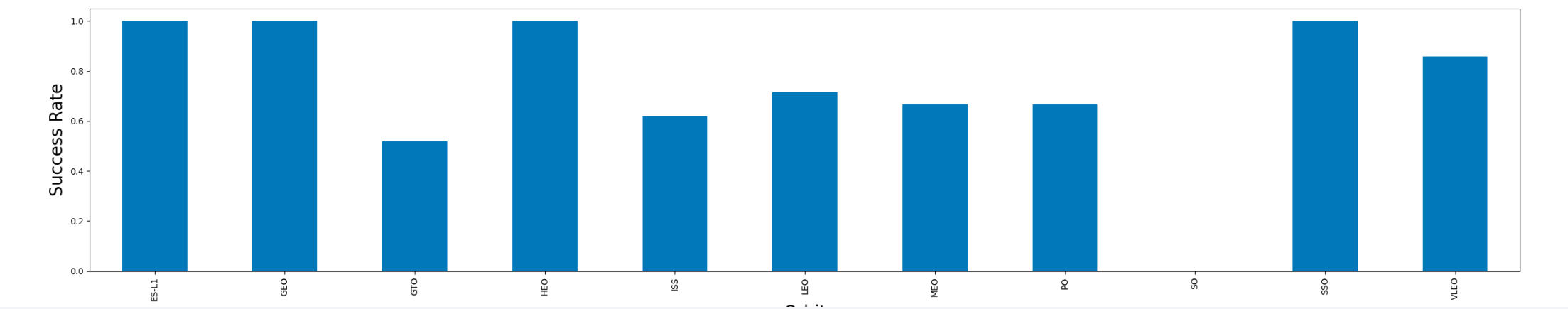
- Launch location mainly CCAFS SLC-40 and less frequently KSC LC 39A
- Outcome has become more and more successful

Payload vs. Launch Site



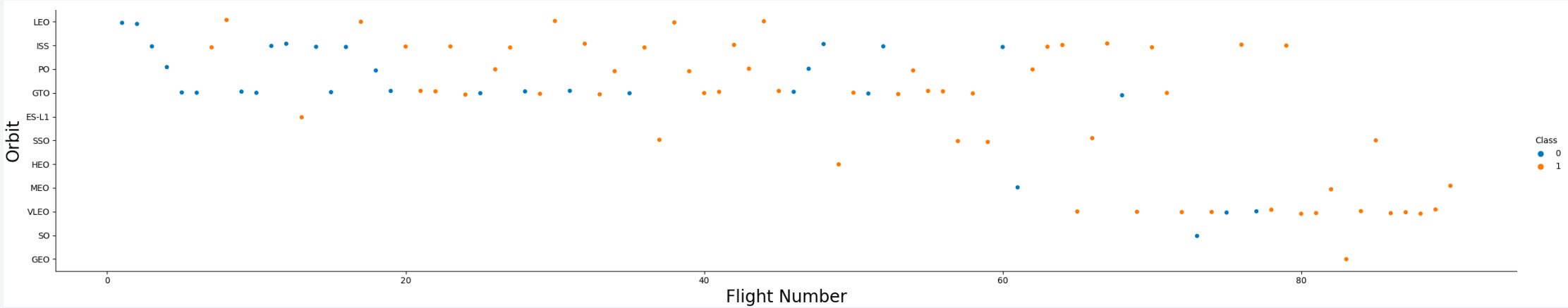
- CCAFS SLC-40 and KSC LC 39A for heavy payloads
- Heavy payloads launches usually have successful outcome

Success Rate vs. Orbit Type



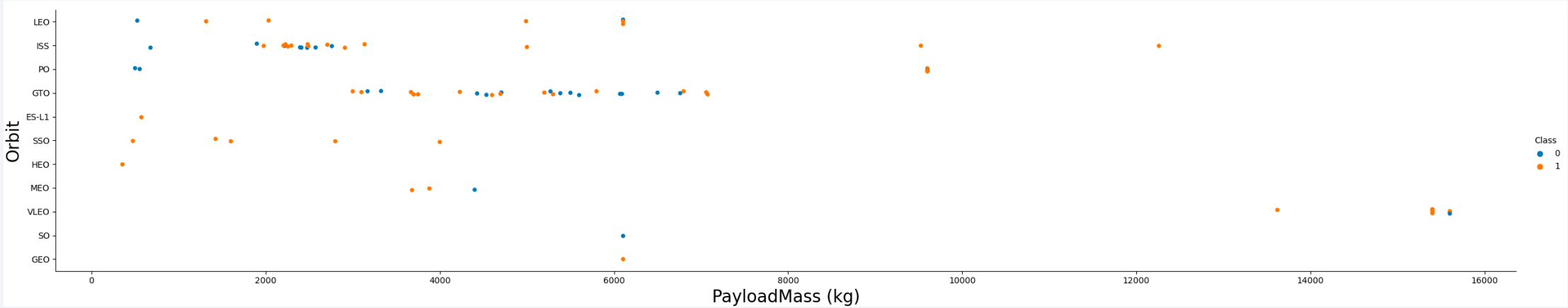
- Launches to ES-LI, GEO, HEO and SSO orbits have a perfect outcome
- Launches to GTO and ISS have the poorest outcome probability

Flight Number vs. Orbit Type



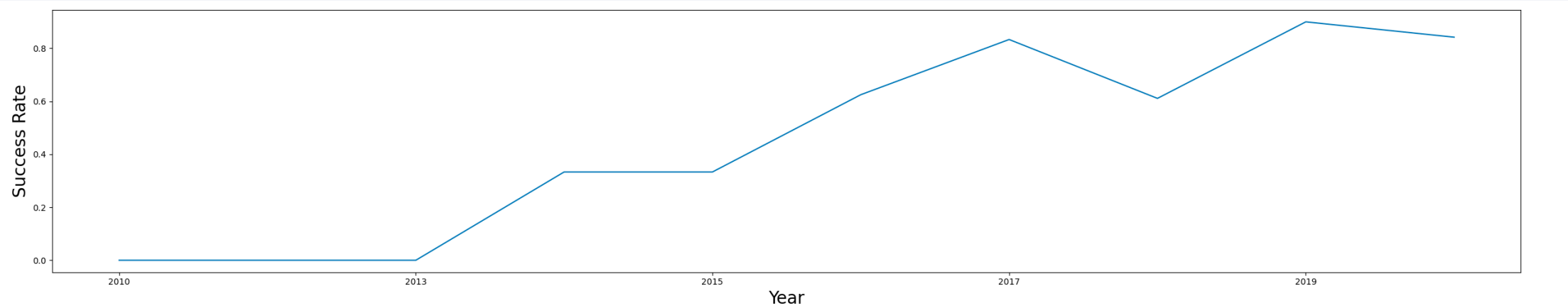
- Launches to ES-LI, GEO, HEO and SSO orbits have limited sample size
- Clear shift of targeted orbit types over the development of the operation

Payload vs. Orbit Type



- Positive correlation between ISS, GTO and VLEO, and their payload mass
- For the others orbit types the sample size is rather small

Launch Success Yearly Trend



- Outcome of the launches has become more and more successful over the years
- Setback experienced in 2020

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

- Select DISTINCT to pull unique launch sites

Launch Site Names Begin with 'KSC'

Display 5 records where launch sites begin with the string 'KSC'

```
%sql SELECT * from SPACEXTBL where (LAUNCH_SITE) LIKE 'KSC%' LIMIT 5;
```

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

Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Orbit	Customer	Mission_Outcome	Landing _Outcome
19-02-2017	14:39:00	F9 FT B1031.1	KSC LC-39A	SpaceX CRS-10	2490	LEO (ISS)	NASA (CRS)	Success	Success (ground pad)
16-03-2017	06:00:00	F9 FT B1030	KSC LC-39A	EchoStar 23	5600	GTO	EchoStar	Success	No attempt
30-03-2017	22:27:00	F9 FT B1021.2	KSC LC-39A	SES-10	5300	GTO	SES	Success	Success (drone ship)
01-05-2017	11:15:00	F9 FT B1032.1	KSC LC-39A	NROL-76	5300	LEO	NRO	Success	Success (ground pad)
15-05-2017	23:21:00	F9 FT B1034	KSC LC-39A	Inmarsat-5 F4	6070	GTO	Inmarsat	Success	No attempt

- Use LIKE to filter launch site KSC
- Limit the output to 5 launch sites

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

- Select SUM (Payload) to pull the sum of the payload
- Use WHERE CUSTOMER to select uniquely ‘NASA (CRS)’ as customer

Average Payload Mass by 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

- Select AVG (Payload) to pull the sum of the payload
- Use WHERE booster version to select uniquely ‘F9 v1.1’ as booster

First Successful Ground Landing Date

```
%sql SELECT min(DATE) FROM SPACEXTBL WHERE "LANDING _OUTCOME" like '%Success (drone ship)%'
```

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

min(DATE)
06-05-2016

- Select min (DATE) to pull the minimum of data
- Use WHERE landing outcome to select uniquely ‘Success (done ship)’ landing

Successful Drone Ship Landing with Payload between 4000 and 6000

```
%sql SELECT DISTINCT BOOSTER_VERSION FROM SPACEXTBL WHERE PAYLOAD_MASS__KG_ BETWEEN 4000 AND 6000 AND "LANDING _OUTCOME" = 'Success (ground pad)';
```

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

Booster_Version
F9 FT B1032.1
F9 B4 B1040.1
F9 B4 B1043.1

- Use DISTINC to uniquely select booster version
- Use WHERE payload to select uniquely value between 4000 and 6000 kg payload and ‘Success (done ship)’ landing

Total Number of Successful and Failure Mission Outcomes

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

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

SUCCESS	FAILURE
100	1

- Use SELECT COUNT to count mission outcomes
- Create two columns with number of successes and failures

Boosters Carried Maximum Payload

```
%sql SELECT BOOSTER_VERSION,PAYLOAD_MASS__KG_ FROM SPACEXTBL WHERE PAYLOAD_MASS__KG_ = (SELECT MAX(PAYLOAD_MASS__KG_) FROM SPACEXTBL)
```

```
 * 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

- Use WHERE payload = SELECT (MAX(payload)) to select maximum payload
- Create two columns with booster version and payload mass

2017 Launch Records

```
%sql SELECT substr("DATE", 4, 2) AS MONTH, "LANDING _OUTCOME" , "BOOSTER_VERSION", "LAUNCH_SITE" FROM SPACEXTBL\
WHERE "LANDING _OUTCOME" = 'Success (ground pad)' and substr("DATE",7,4) = '2017'
```

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

MONTH	Landing _Outcome	Booster_Version	Launch_Site
02	Success (ground pad)	F9 FT B1031.1	KSC LC-39A
05	Success (ground pad)	F9 FT B1032.1	KSC LC-39A
06	Success (ground pad)	F9 FT B1035.1	KSC LC-39A
08	Success (ground pad)	F9 B4 B1039.1	KSC LC-39A
09	Success (ground pad)	F9 B4 B1040.1	KSC LC-39A
12	Success (ground pad)	F9 FT B1035.2	CCAFS SLC-40

- Use substr(”DATE”, 4, 2) to select the months of 2017
- Create four columns with month, landing outcome, booster version and launch site

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

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

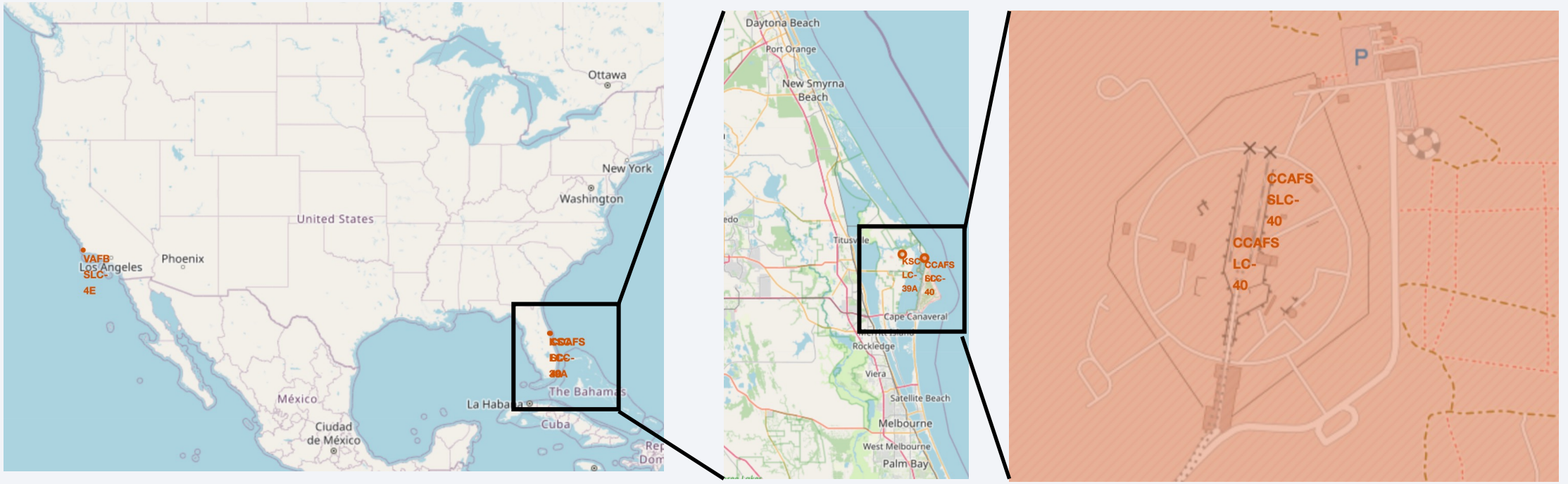
```
 * sqlite:///my_data1.db
Done.
```

Landing _Outcome	COUNT("LANDING _OUTCOME")
Success	20
Success (drone ship)	8
Success (ground pad)	6

- Use WHERE to select dates between 2010-06-04 and 2017-03-20
- Create two columns with landing outcome and their number

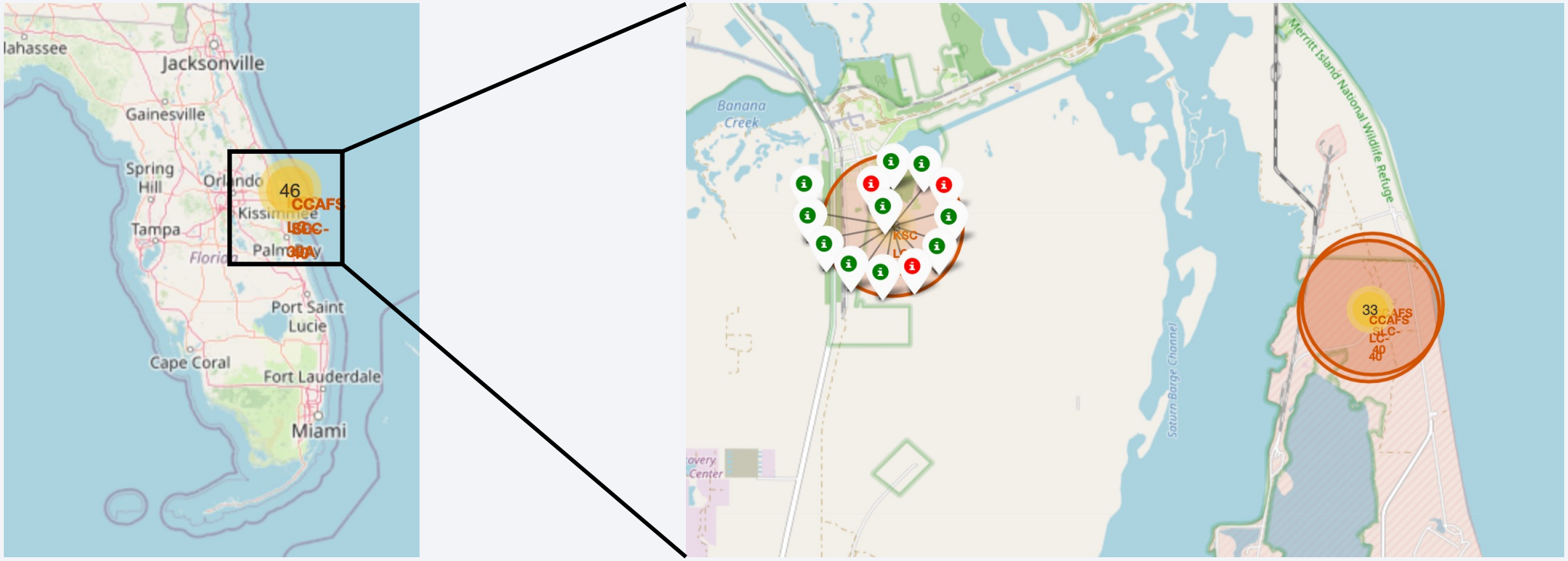

Section 3
Launch Sites
Proximities Analysis

Launch Site position visualization



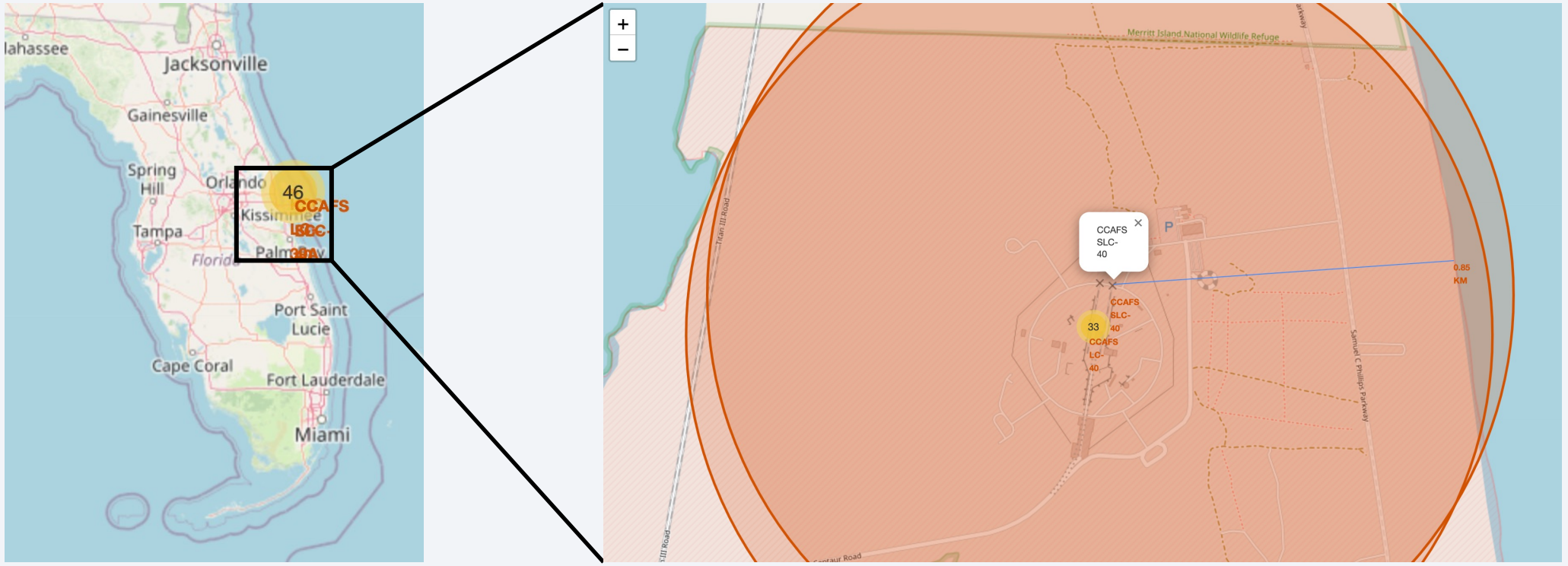
- Launch site VAFB SLC 4E in south California
- Launch sites KSC LC 39A, CCAFS SLC-40 and CCAFS LC-40 in south Florida

Successful and failed launches per each site

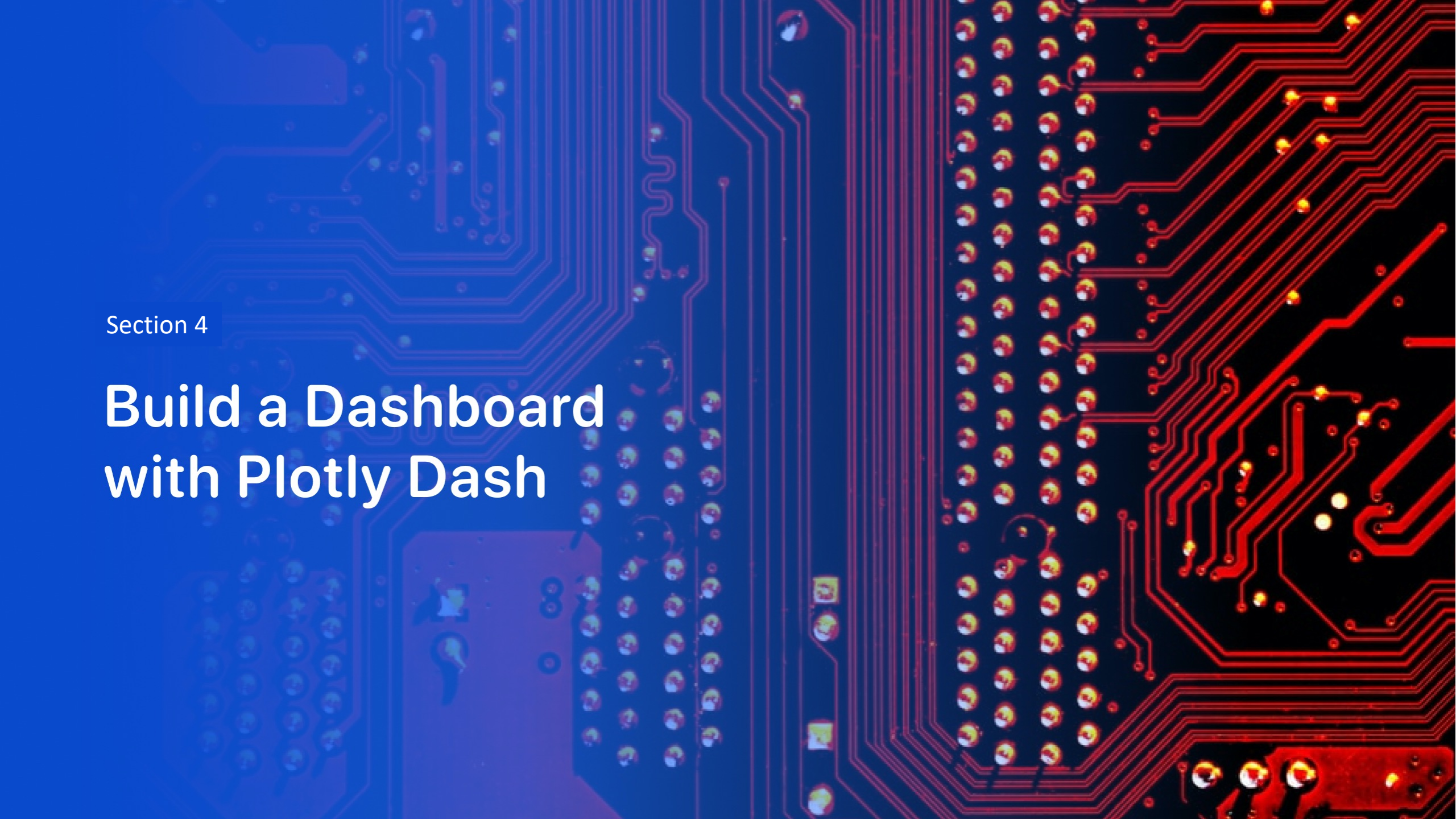


- Focus on launch site in south Florida
- Interactive map to explore successful and failed launches for each site

Distance CCAFS SLC-40 - Coastline



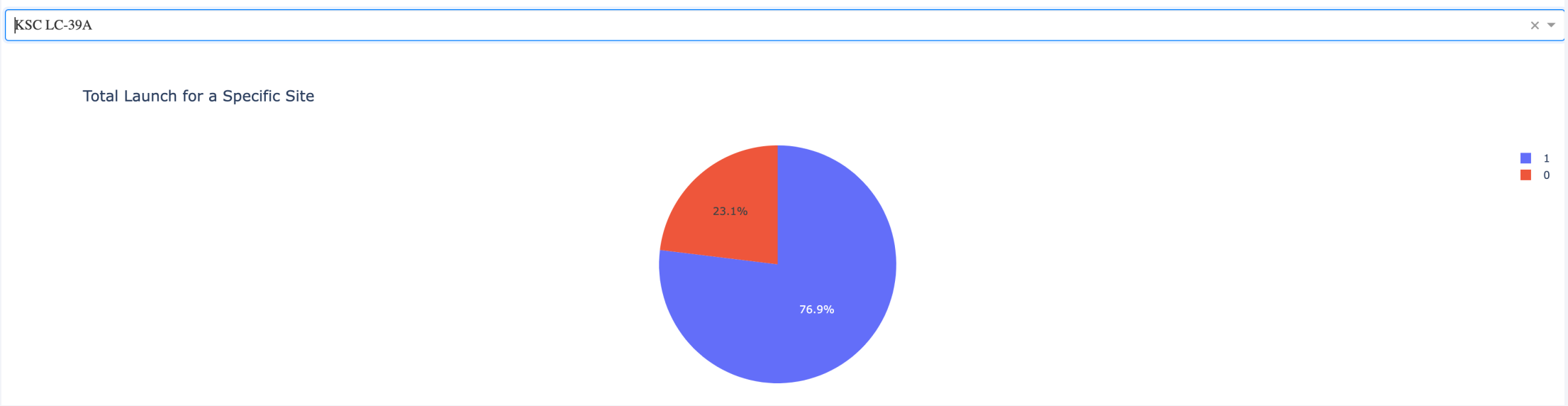
- Focus on launch site CCAFS SLC-40
- Distance from coastline marked by the blue line (0.85 km)


Section 4
Build a Dashboard with Plotly Dash

Launch site identification for successful launches

- Most successful launches carried out at KSC LC-39A

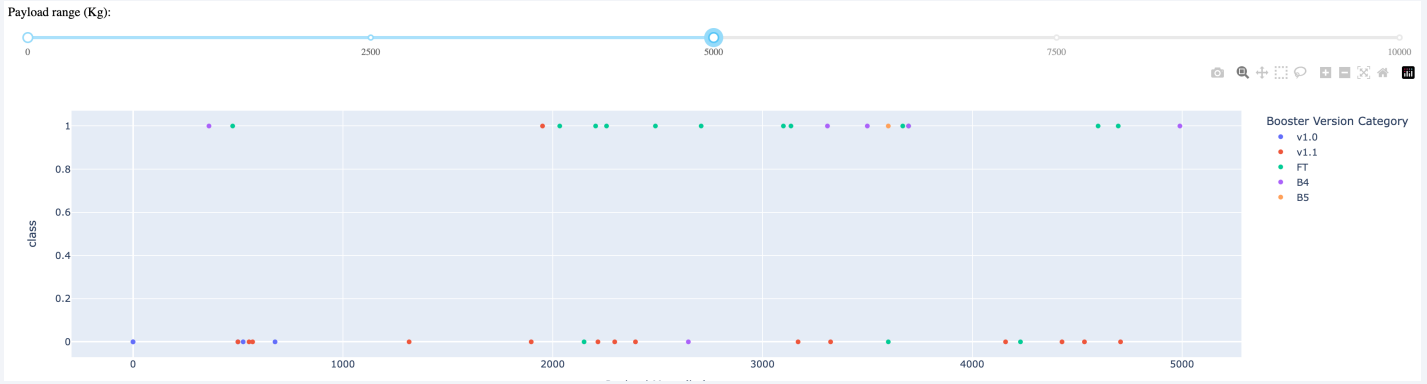
Success ratio for most successful launch site



- Success ratio for KSC LC-39A is >75%

Booster type success ratio

- FT and B4 most successful boosters for light (< 5000 kg) payload mass

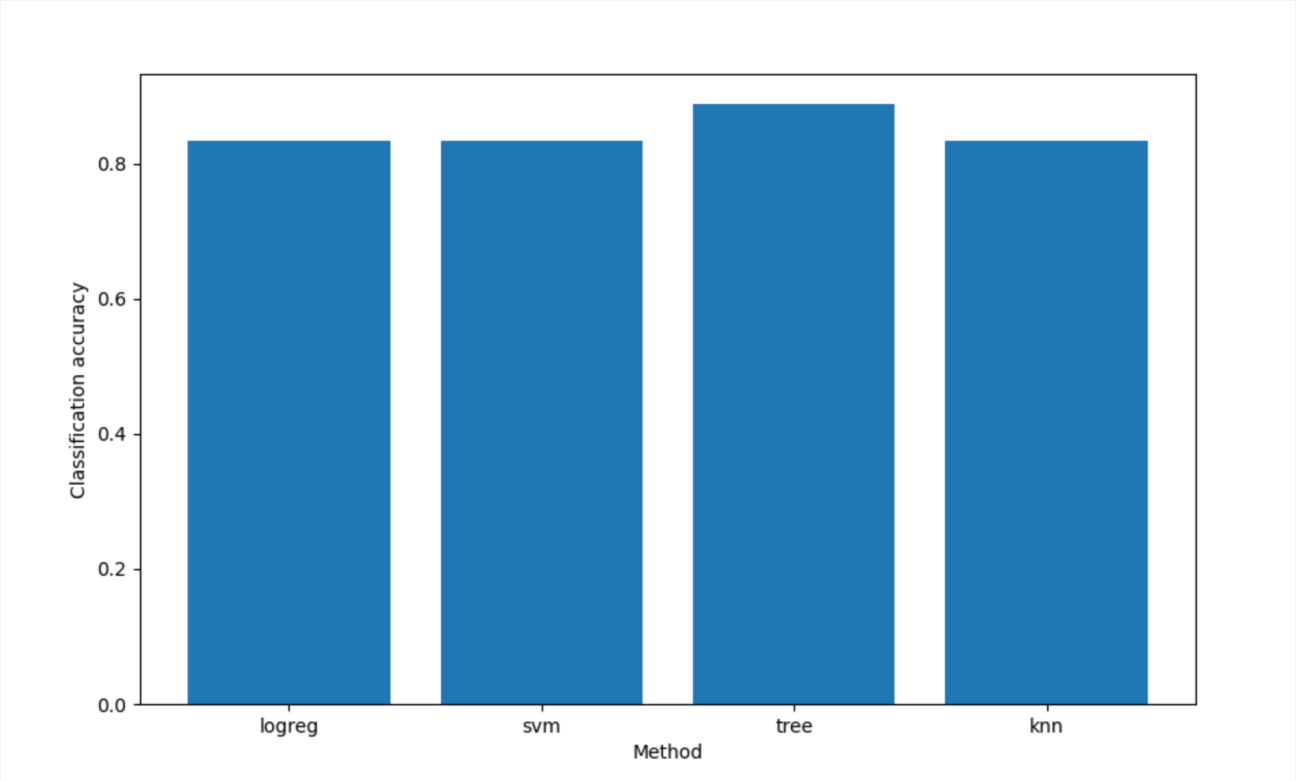


- Launches carrying heavy payload mass (> 5000 kg) have scarce success regardless of the booster type

Section 5

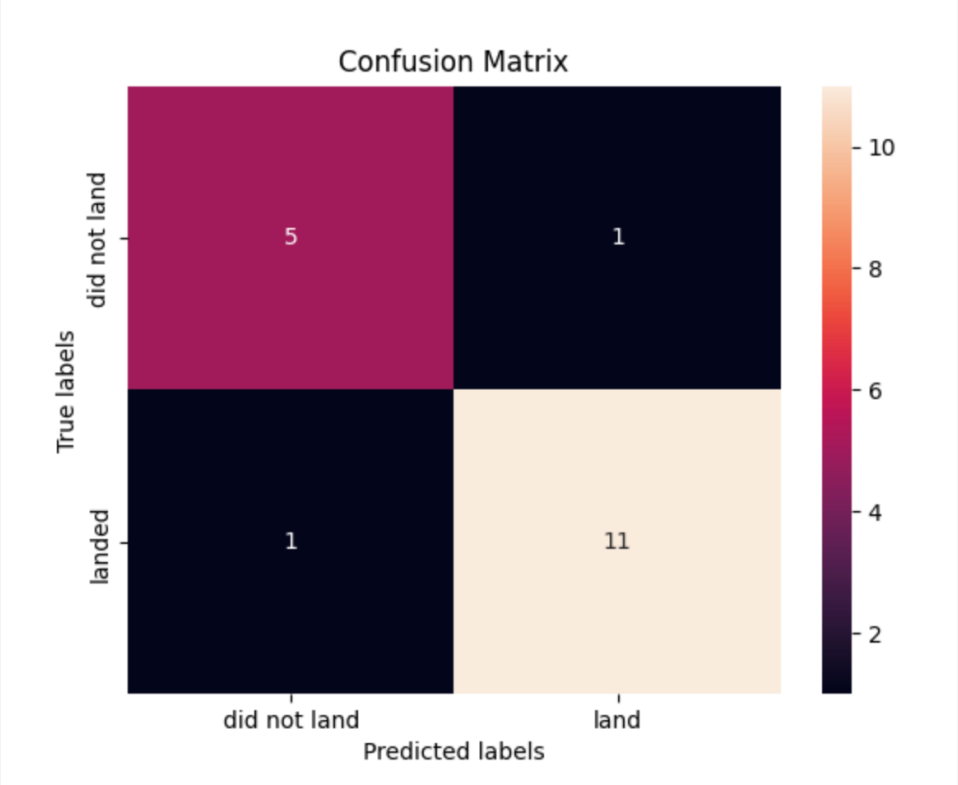
Predictive Analysis (Classification)

Classification Accuracy



- All methods have a classification accuracy > 80%
- Decision tree method is the best performing with 88%

Confusion Matrix



- False positive for Decision Tree < 20% by far the best among the various methods
- False negative for Decision Tree < 10%

Conclusions

- We successfully retrieve SpaceX API and we visualize launch location
- Trained decision tree machine learning algorithm delivers predictive accuracy of >85%, the most accurate model at disposal
- False negative and positive of decision tree model are the lowest of the explored training models (< 10% and < 20%, respectively)
- We suggest employing this trained decision tree model to predict future success launches of SpaceX

Appendix

- All the Python and SQL codes can be found on the following GitHub link: https://github.com/FrancescoBernasconi/IBM-Capston-Project-SpaceY

Thank you!