IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Gregory Hinds
04/03/24

Outline

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

Executive Summary

- Summary of methodologies
  - Data Collection via API
  - Data Collection with Web Scraping
  - Data Wrangling
  - Exploratory Data Analysis with Data Visualization
  - Exploratory Data Analysis with SQL
  - Interactive Visual Analytics
  - Predictive Analytics
- Summary of all results
  - Exploratory Data Analysis results
  - Interactive Analytics screen captures
  - Predictive Analytics results

Introduction

- Project background and context
  - 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. We endeavour to predict, using data science methodologies, if we can predict if the first stage booster will land successfully.
- Problems you want to find answers
  - What factors determine if the rocket will land successfully?
  - The interaction amongst various features that determine the success rate of a successful landing.
  - What operating conditions needs to be in place to ensure a successful landing program.

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - We collected data using the publicly available SpaceX API and from the Wikipedia page listing launches.
- Perform data wrangling
  - We checked for missing/incomplete data, engineered a feature to clearly identify successes and failures, and used one-hot encoding on categorical features.
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - Using a train/test split of 80%/20%, we trained a logistic regression, a SVM and a Decision Tree classifier on the data.

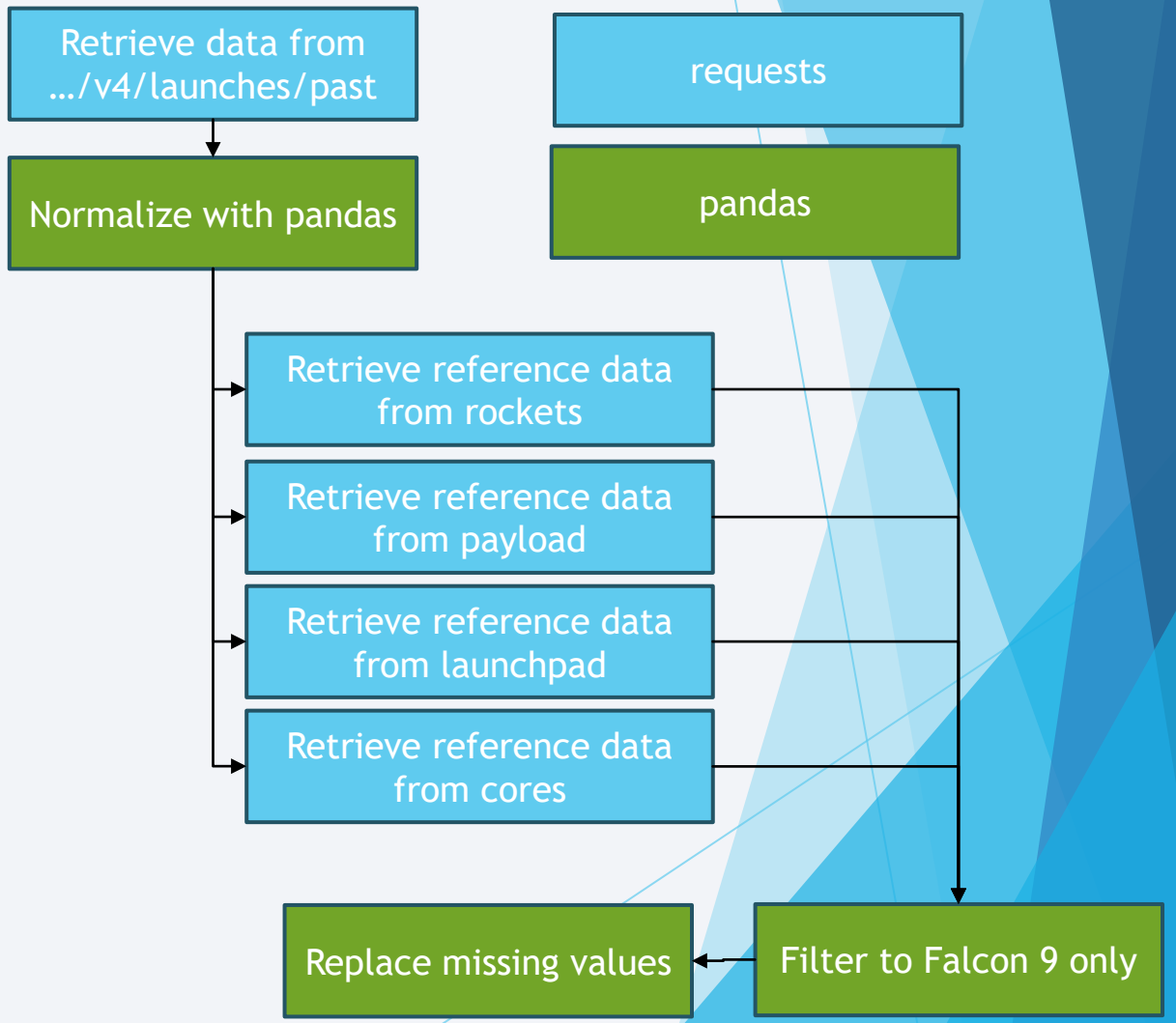
Data Collection – Requests and SpaceX API

We want to collect data from an external API and transform it into a well-structured dataframe.

- We collected data from the SpaceX API using the Python Requests package
- Using the https://api.spacexdata.com/v4/launches/past endpoint
- Using the Pandas json_normalize function we can extract data straight into a dataframe
- As many data records were internal references, we also needed to perform extra calls to get the names and values associated with those internal references
- We performed some filtering on this dataframe to show only Falcon 9 launches
- We replaced some missing values in the PayloadMass column with the column mean

Data Collection – Requests and SpaceX API

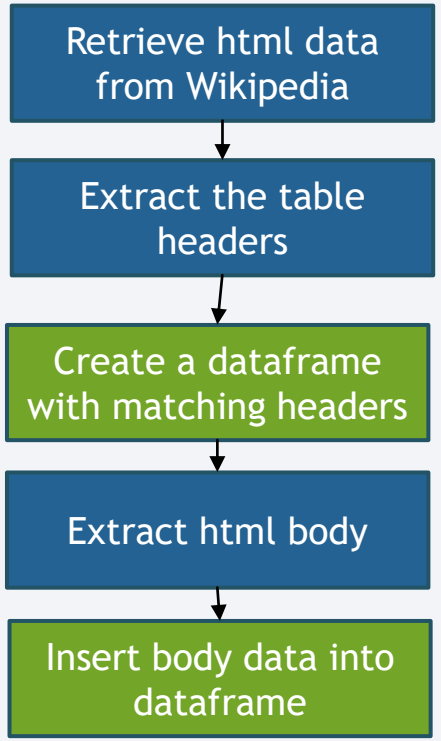
- Present your data collection with SpaceX REST calls using key phrases and flowcharts

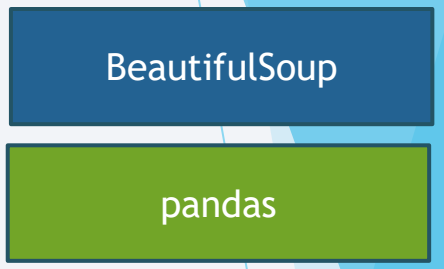


Notebook URL: https://github.com/Ysadore/SpaceX/blob/main/Data%20Collection%20API.ipynb

Data Collection - Scraping

- We use the Python BeautifulSoup library to scrape structured data from contexts where the data is not formatted as we prefer





Notebook URL:
https://github.com/Ysadore/SpaceX/blob/main/Data%20Collection%20with%20Web%20Scraping.ipynb

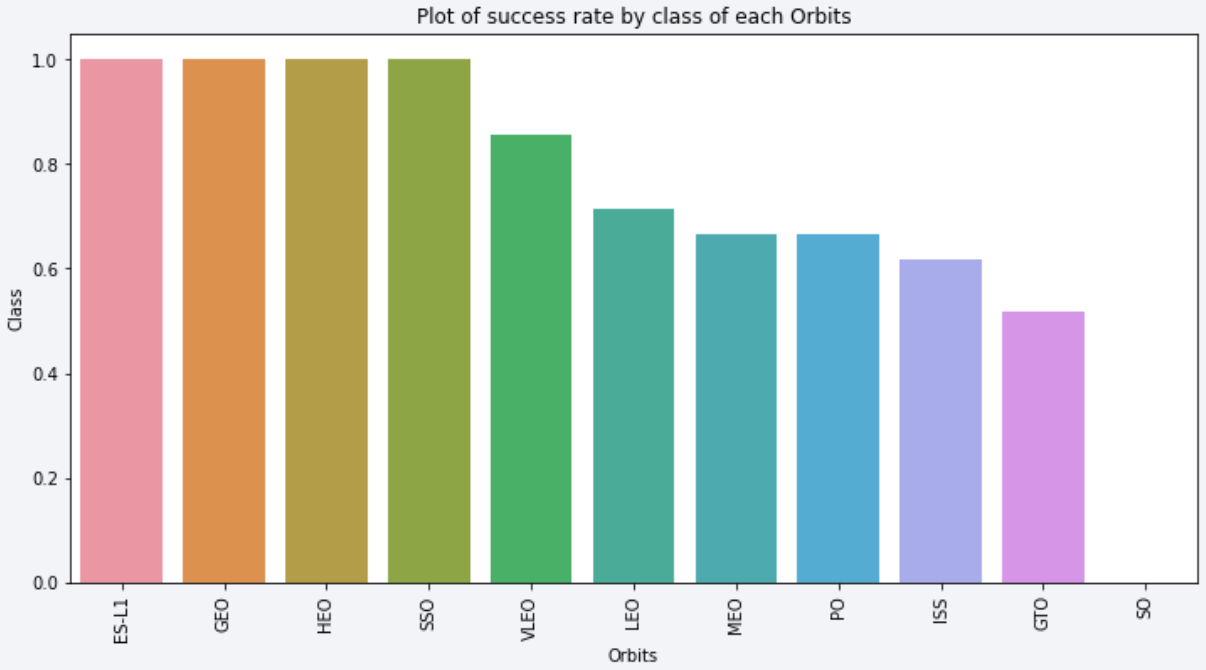
Data Wrangling

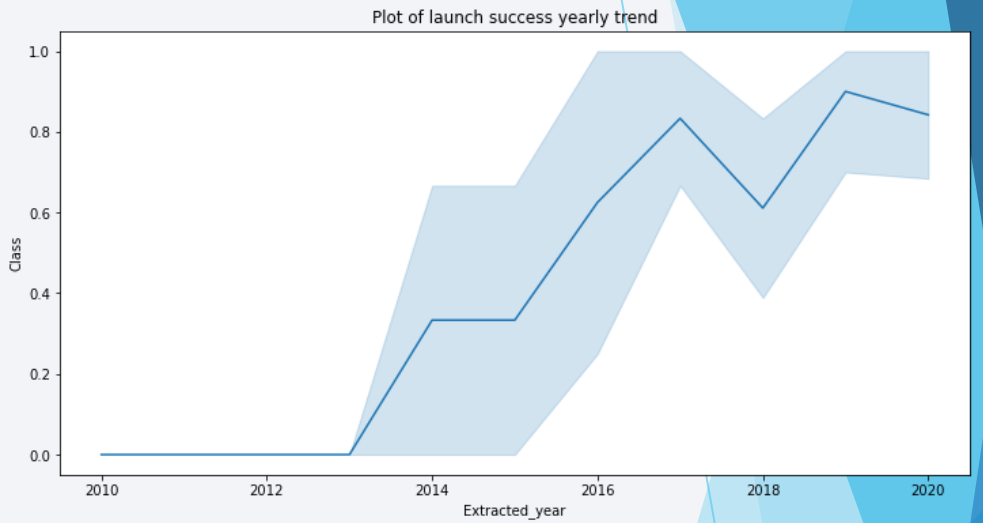
- We performed exploratory data analysis and determined which labels we would use for model training later
- We identified which columns had missing data
- We counted the number of launches from each site, the number of successful launches and calculated the percentage success rate
- We calculated the landing outcome per difference type of orbit
- We categorized various landing outcomes into a single success/failure feature

Notebook URL: https://github.com/Ysadore/SpaceX/blob/main/Data%20Wrangling.ipynb

EDA with Data Visualization

- We explored the relationship between Payload Mass, Flight Number (indicating progression over time), Orbit type and Success Rate





The success rate has mostly been increasing over time.

Some orbits have much lower success rates than others.

Notebook URL: https://github.com/Ysadore/SpaceX/blob/main/EDA%20with%20Data%20Visualization.ipynb

EDA with SQL

- SQL queries performed:
  - Display the names of unique launch sites
  - 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. Use a subquery
  - 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.

Notebook URL: https://github.com/Ysadore/SpaceX/blob/main/EDA%20with%20SQL.ipynb

Build an Interactive Map with Folium

- We marked all launch sites, and added map objects such as markers, circles, lines to mark the success or failure of launches for each site on the folium map.
- We mapped the feature launch outcomes to 0 for failure and 1 for success.
- Using the color-labeled marker clusters, we identified which launch sites have relatively high success rate.
- We calculated the distances between a launch site to its proximities and discovered some insights:
  - The launch sites are not near railways or highways, but are all close to the coast
  - The launch sites are all beyond a certain distance from cities

Notebook URL:
https://github.com/Ysadore/SpaceX/blob/main/Interactive%20Visual%20Analytics%20with%20Folium.ipynb

Build a Dashboard with Plotly Dash

- We built an interactive dashboard with Plotly dash
- We plotted pie charts showing the total launches by a certain sites
- We plotted scatter graph showing the relationship with Outcome and Payload Mass (Kg) for the different booster version.

Notebook URL: https://github.com/Ysadore/SpaceX/blob/main/app.py

Predictive Analysis (Classification)

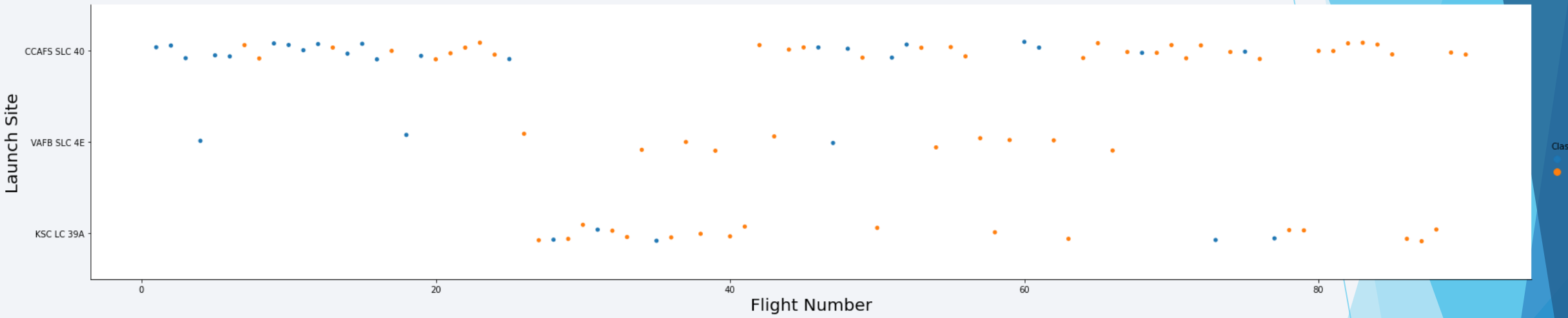
- We prepared the data with numpy, standardized/transformed it and split it into training and testing segments
- We built different machine learning models and tune different hyperparameters using GridSearchCV.
- We used accuracy as the metric for our model, improved the model using feature engineering and algorithm tuning.

Notebook URL: https://github.com/Ysadore/SpaceX/blob/main/Machine%20Learning%20Prediction.ipynb

Section 2

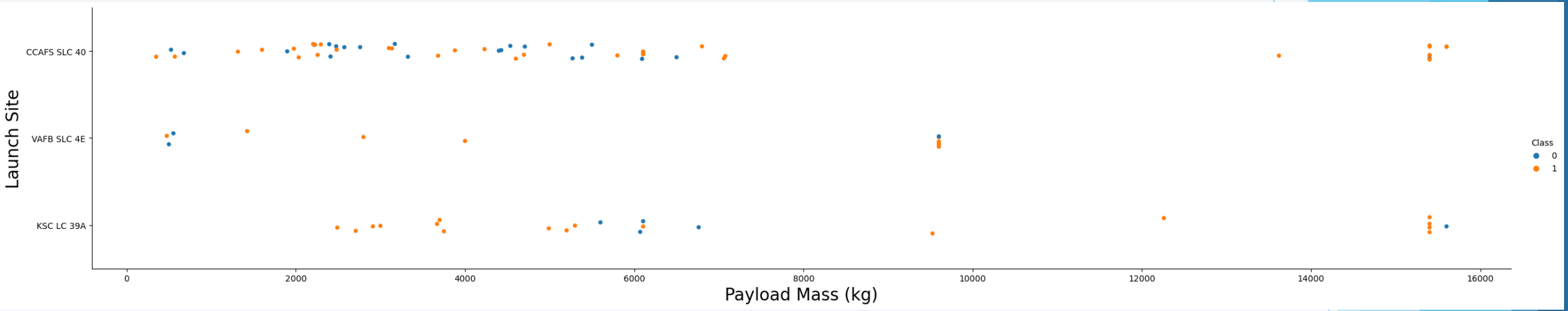
Insights drawn from EDA

Flight Number vs. Launch Site



- The more flights are launched from a site, the greater the success rate
- Site KSC LC 39A started launches later than the others
- Site VAFB SLC 4E is launched from less frequently than the others

Payload vs. Launch Site

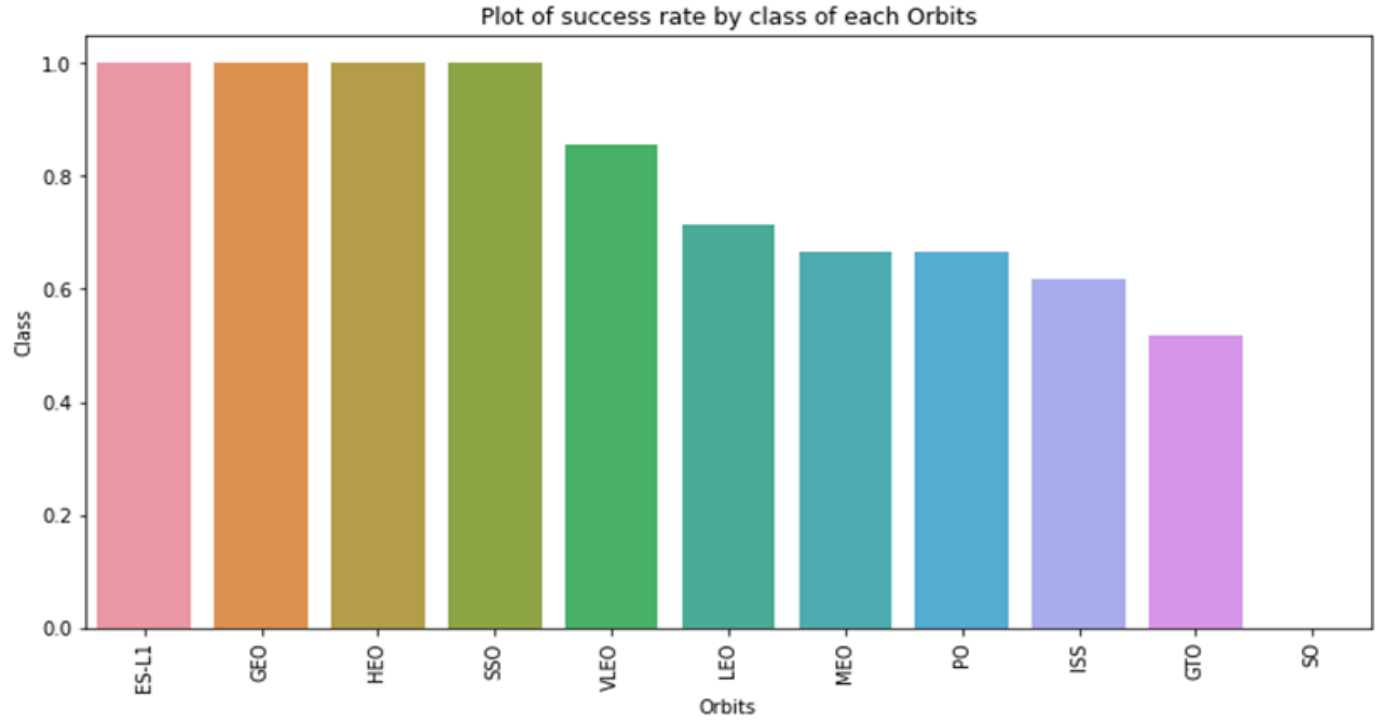


- Most launches have a payload of <8000kg
- KSC LC 39A has a very high success rate for low-mass payloads
- CCAFS and KSC has a very high success rate for very heavy payloads

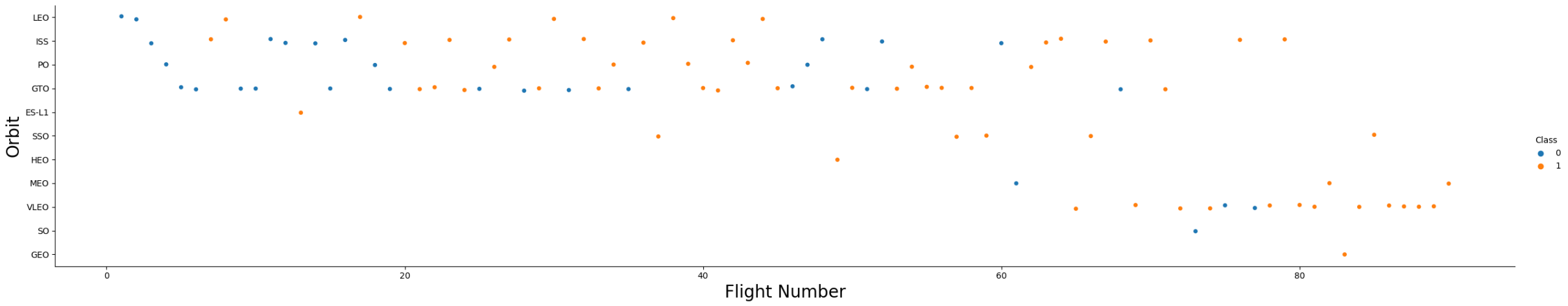
Success Rate vs. Orbit Type

- Some orbits have a much higher or flawless success rate: ES-L1, GEO, HEO and SSO

- Whereas launches to the ISS and into GTO are much less likely to succeed

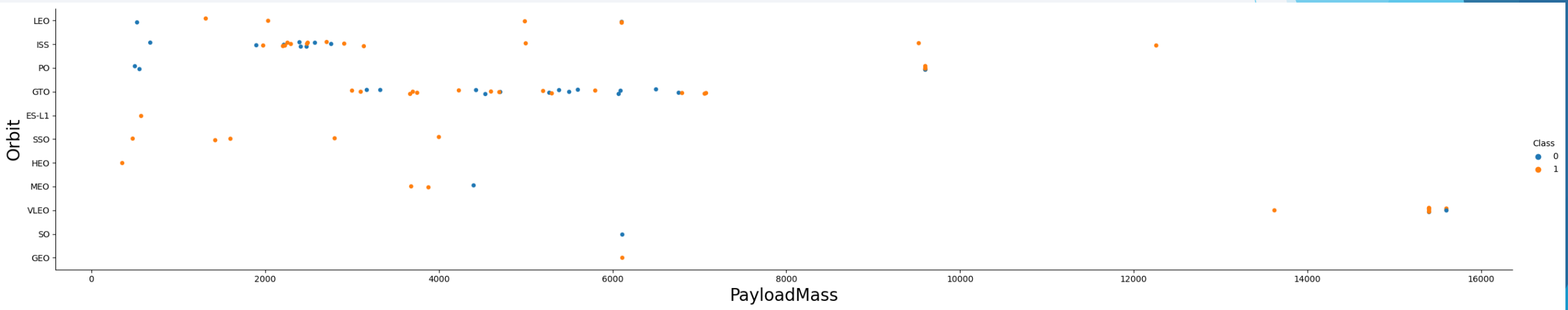


Flight Number vs. Orbit Type



- LEO launches are getting more reliable over time
- Some orbits are less frequent than others – MEO and GEO are particularly infrequent

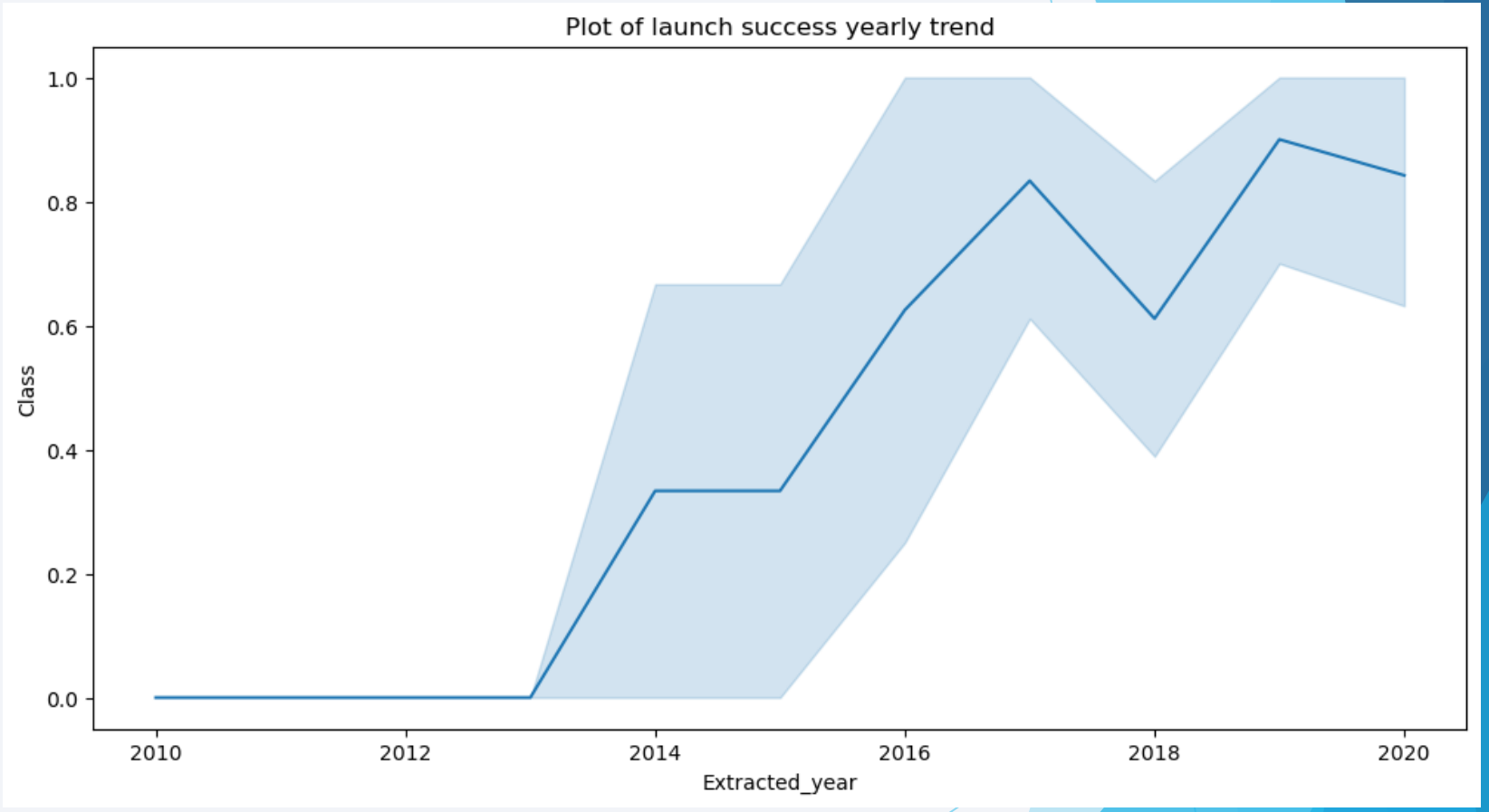
Payload vs. Orbit Type



- VLEO has a tendency towards very heavy payloads
- There isn't a clear relationship between payload mass and success for GTO

Launch Success Yearly Trend

- Success rate has been broadly increasing until 2017 before a dip in 2018
- Launch success rate in 2020 is around 80%



All Launch Site Names

Task 1

Display the names of the unique launch sites in the space mission

In [11]:
```
%sql select distinct "Launch_Site" from SPACEXTABLE
```

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

Out[11]:

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

- The DISTINCT keyword allows us to select only unique values

Launch Site Names Begin with 'CCA'

- The LIMIT keywords selects only that many rows
- The % symbol in the LIKE clause acts as a wildcard for any number of characters

Task 2

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

In [12]:
```
%sql select * from SPACEXTABLE where "Launch_Site" like 'CCA%' Limit 5
```

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

Out[12]:

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

Task 3

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

In [14]:
```
%sql select sum("PAYLOAD_MASS__KG_") from SPACEXTABLE where Customer like 'NASA (CRS)'
```

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

Out[14]:

sum("PAYLOAD_MASS__KG_")
45596

- SUM is an aggregate function, returning a single value calculated from all the values in the column

Average Payload Mass by F9 v1.1

Task 4

Display average payload mass carried by booster version F9 v1.1

In [17]:
```
%sql select avg("PAYLOAD_MASS__KG_") from SPACEXTABLE where "Booster_Version" like 'F9 v1.1'
```

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

Out[17]:

avg("PAYLOAD_MASS__KG_")
2928.4

- AVG is also an aggregate function, performing the arithmetic mean

First Successful Ground Landing Date

Task 5

List the date when the first succesful landing outcome in ground pad was acheived.

Hint:Use min function

In [21]:
```
%sql select min(Date) from SPACEXTABLE where Landing_Outcome like 'Success (ground pad)'
```

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

Out[21]:

min(Date)
2015-12-22

- MIN works on numbers and dates

Successful Drone Ship Landing with Payload between 4000 and 6000

```
select distinct "Booster_Version", PAYLOAD_MASS__KG_, Landing_Outcome from SPACEXTABLE
where Landing_Outcome like 'Success (drone ship)' and PAYLOAD_MASS__KG_ < 6000 and
PAYLOAD_MASS__KG_ > 4000
```

Done.

Out[36]:

Booster_Version	PAYLOAD_MASS__KG_	Landing_Outcome
F9 FT B1022	4696	Success (drone ship)
F9 FT B1026	4600	Success (drone ship)
F9 FT B1021.2	5300	Success (drone ship)
F9 FT B1031.2	5200	Success (drone ship)

- We can use Boolean operators like AND, OR, NOT and so forth to join and modify WHERE clause items

Total Number of Successful and Failure Mission Outcomes

Task 7

List the total number of successful and failure mission outcomes

In [38]:
```
%sql select Mission_Outcome, count(Mission_Outcome) from SPACEXTABLE group by Mission_Outcome
```

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

Out[38]:

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

- We can combine aggregate functions and the GROUP BY keyword to perform the aggregation only over unique values from another column

Boosters Carried Maximum Payload

```
select distinct booster_version from SPACEXTABLE where PAYLOAD_MASS__KG_ = (select max(PAYLOAD_MASS__KG_) from SPACEXTABLE)
```

Done.

Out[42]:

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

- This query uses a subquery to pre-calculate another value; in this case the mass of the heaviest payload

2015 Launch Records

```
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)'
```

Out[44]:

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

- We needed to use substr to handle date comprehension here as this implementation SQL lacks native functions

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

```
select landing_outcome, count(landing_outcome) from SPACEXTABLE group by landing_outcome where "date" > 2010-06-04 and "date" < 2017-03-20
```

- The ORDER BY … DESC keyword would be used to order the results

Out[48]:

Landing_Outcome	count(landing_outcome)
Controlled (ocean)	5
Failure	3
Failure (drone ship)	5
Failure (parachute)	2
No attempt	21
No attempt	1
Precluded (drone ship)	1
Success	38
Success (drone ship)	14
Success (ground pad)	9
Uncontrolled (ocean)	2

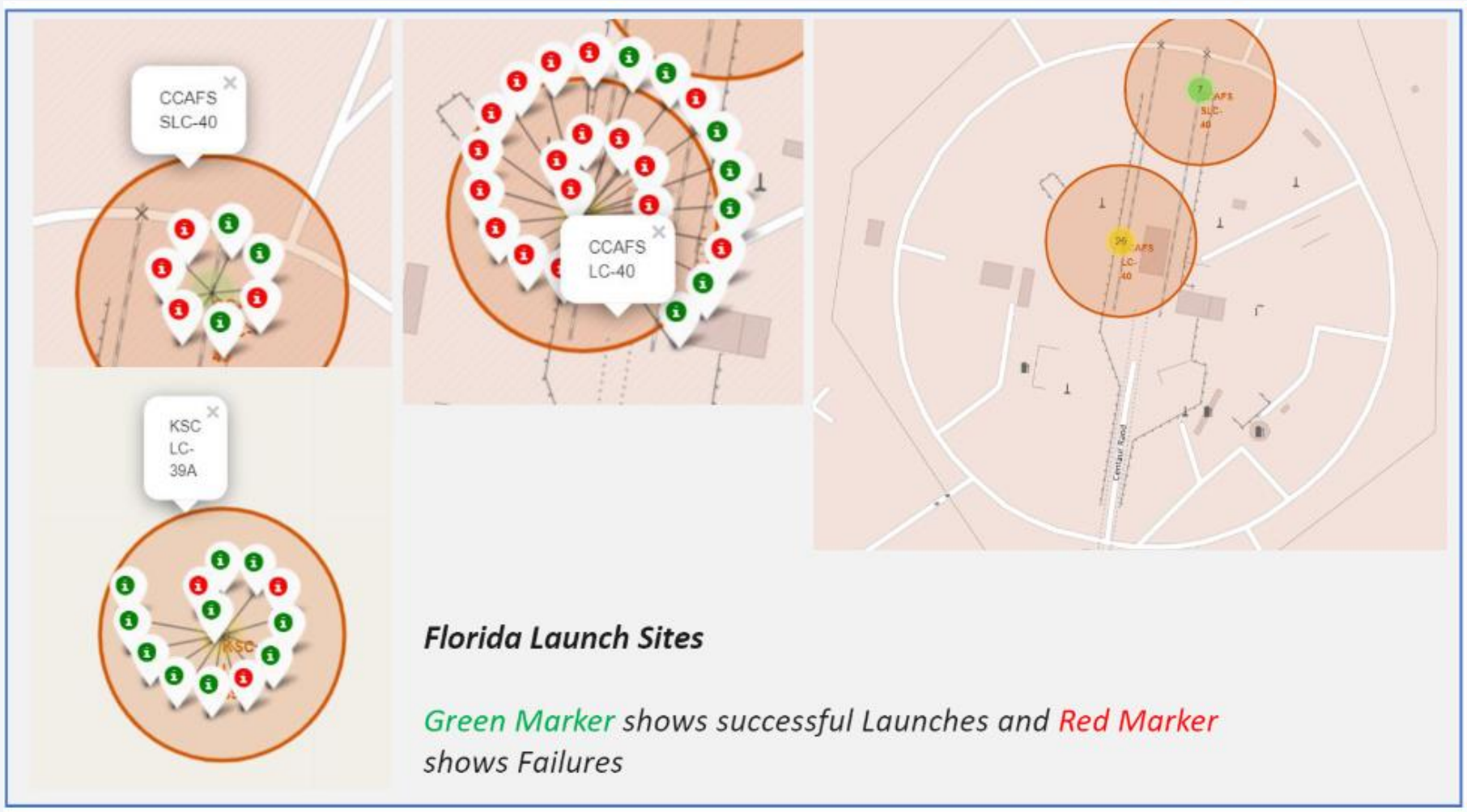
Section 3

Launch Sites Proximities Analysis

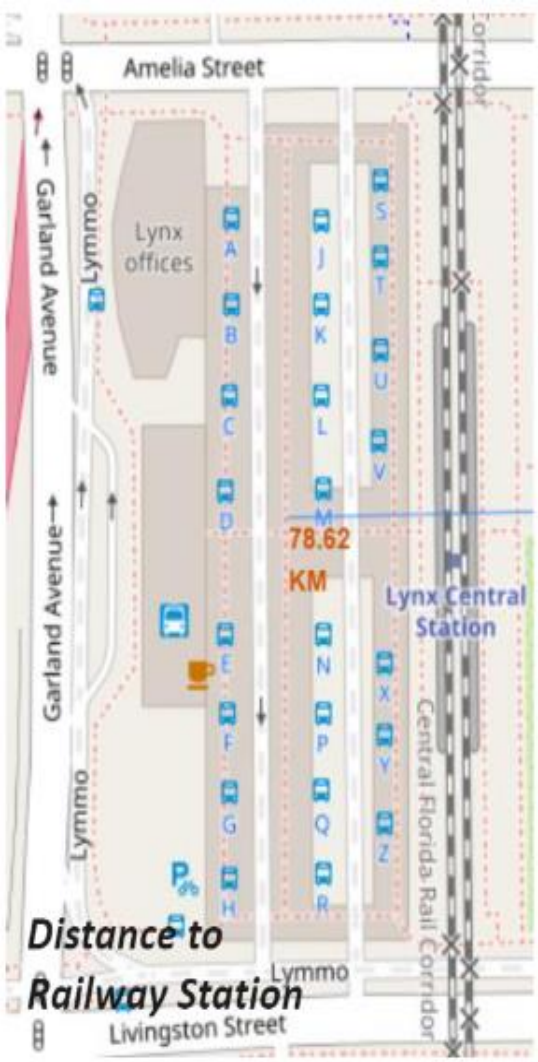
Launch Sites Global Map

We can see that the SpaceX launch sites are in the United States of America coasts. Florida and California

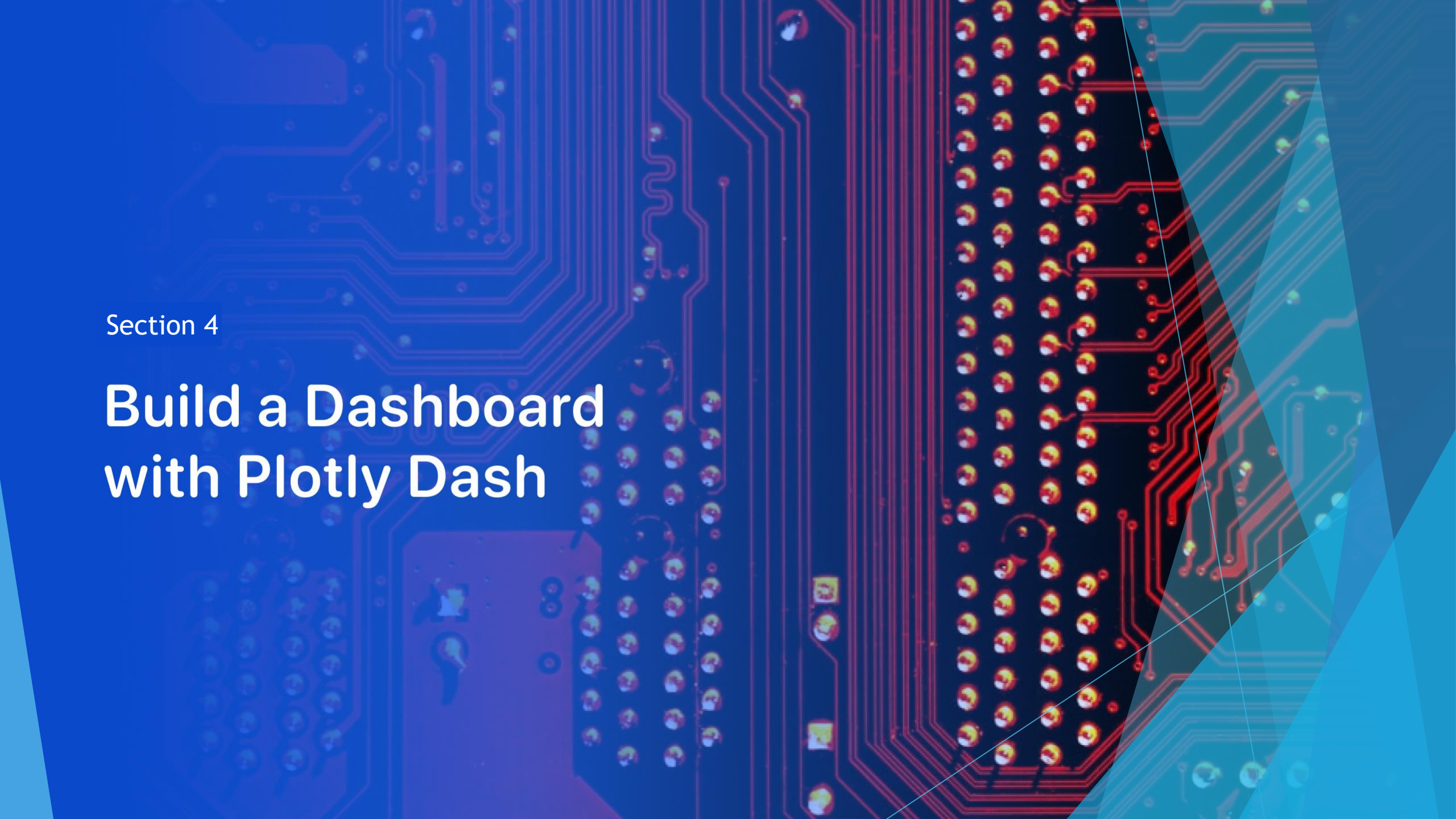
Markers showing Launch Sites



Launch Site proximity to landmarks



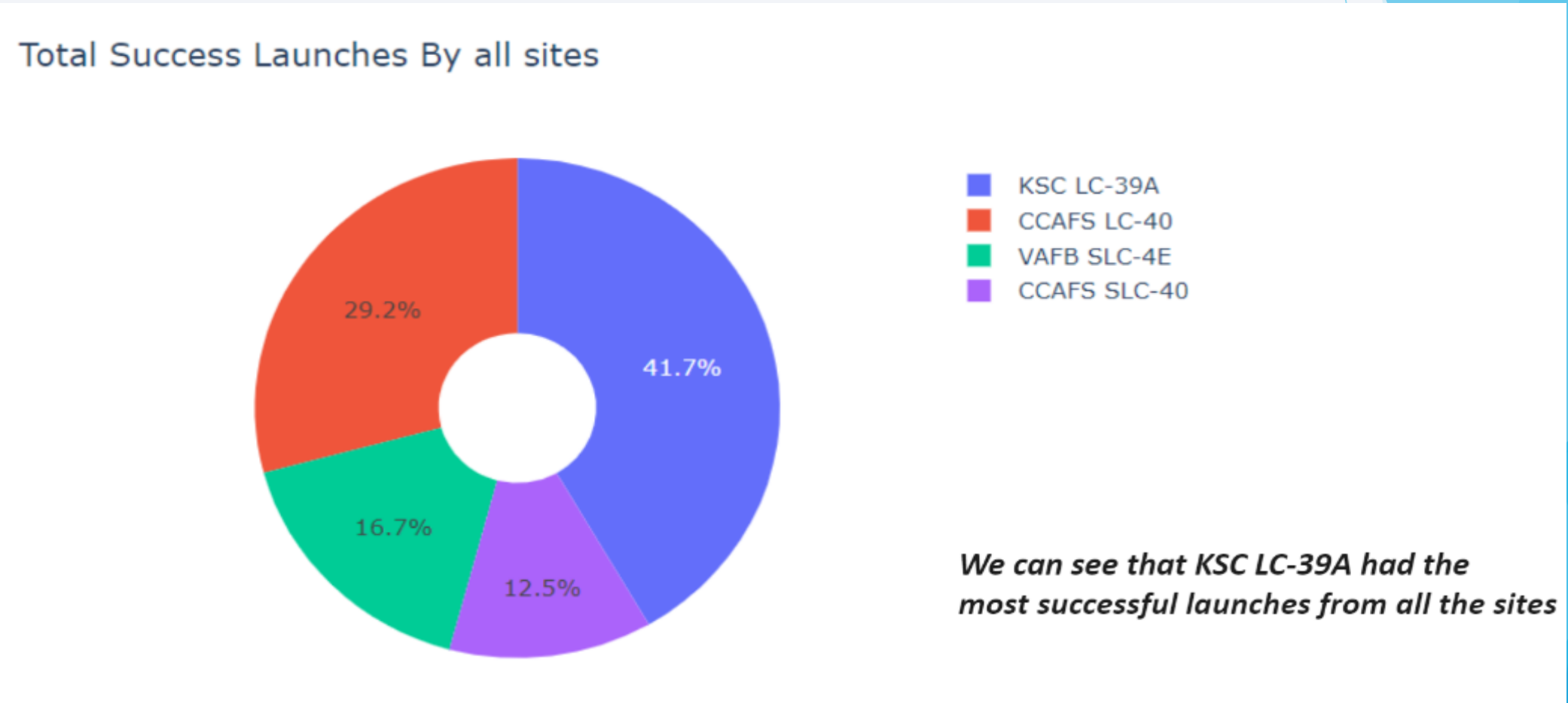
- Are launch sites in close proximity to railways? No
- Are launch sites in close proximity to highways? No
- Are launch sites in close proximity to coastline? Yes
- Do launch sites keep certain distance away from cities? Yes



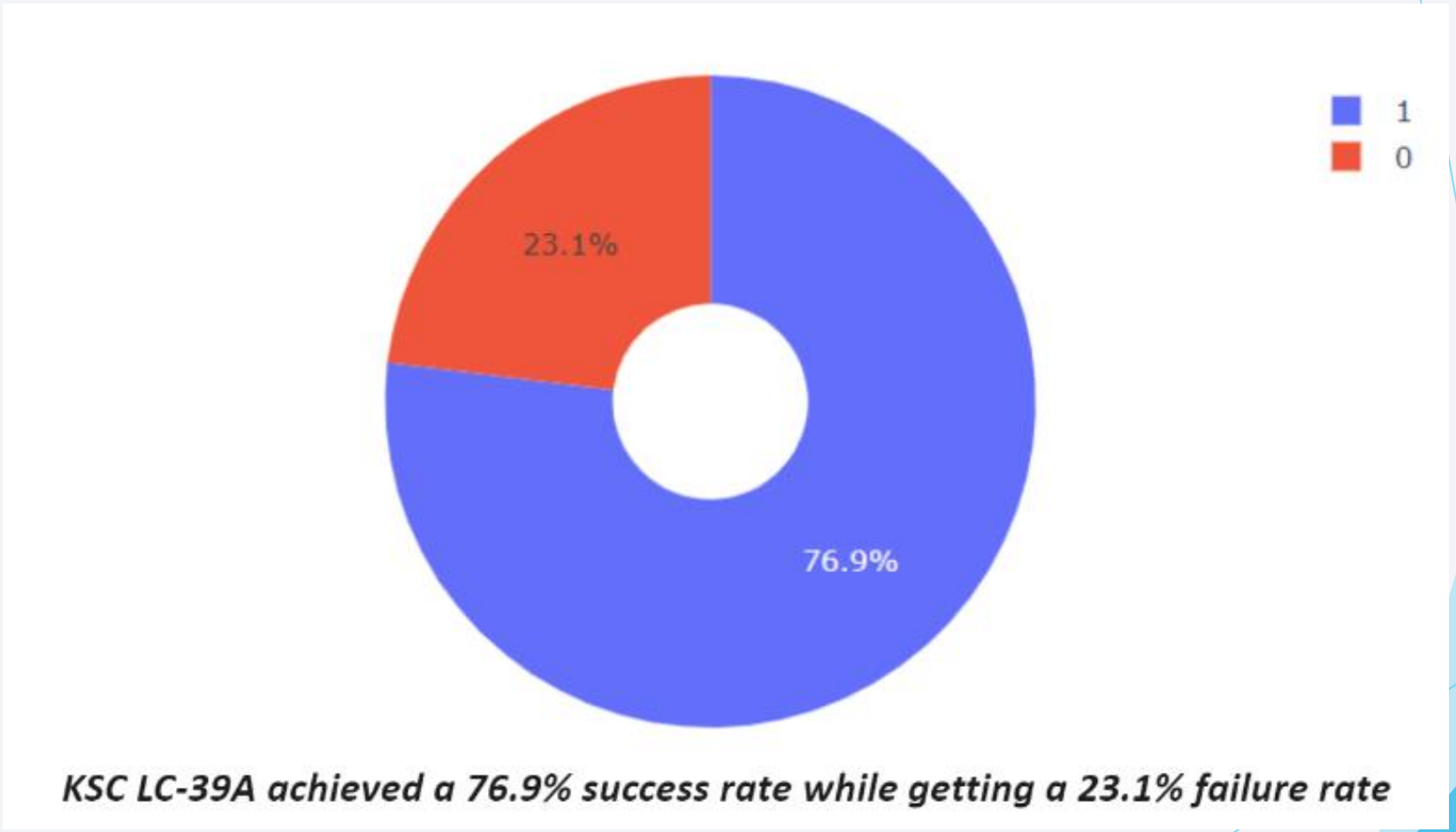
Section 4

Build a Dashboard with Plotly Dash

Success Rate by Launch Site

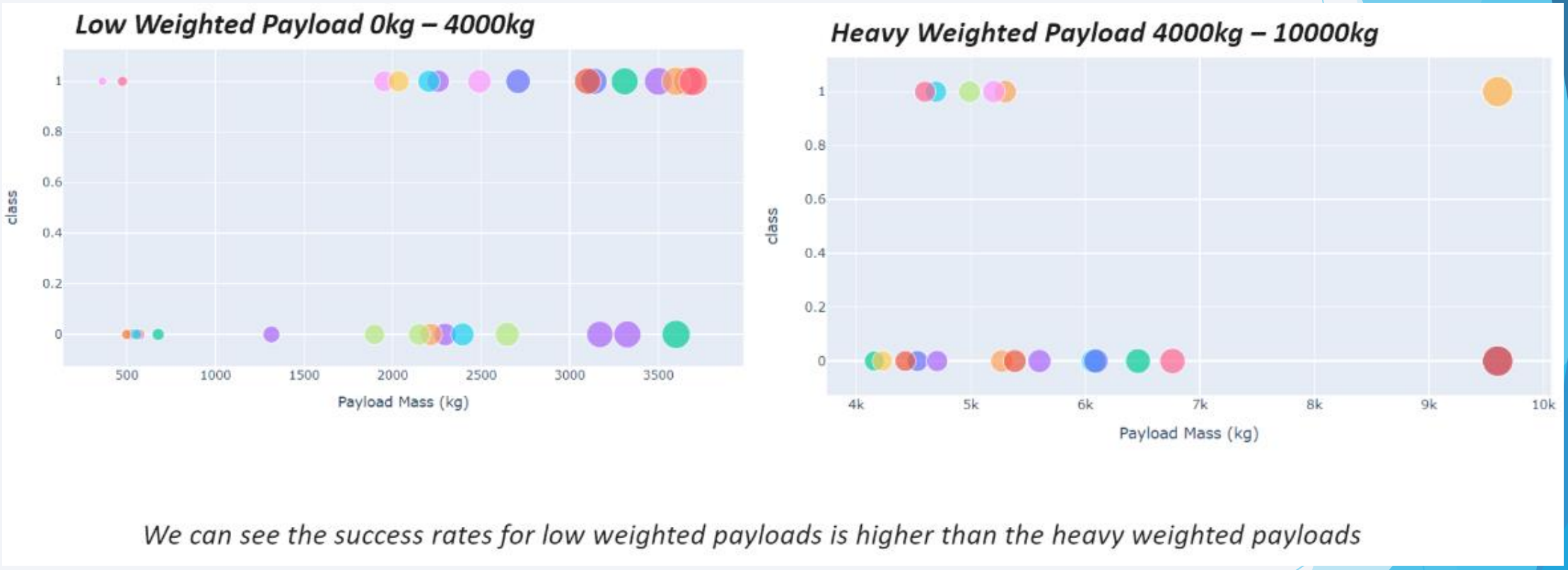


Success Ratio of the most successful Launch Site



KSC LC-39A achieved a 76.9% success rate while getting a 23.1% failure rate

Payload Mass and Launch Outcome variants



We can see the success rates for low weighted payloads is higher than the heavy weighted payloads

Section 5

Predictive Analysis (Classification)

Classification Accuracy

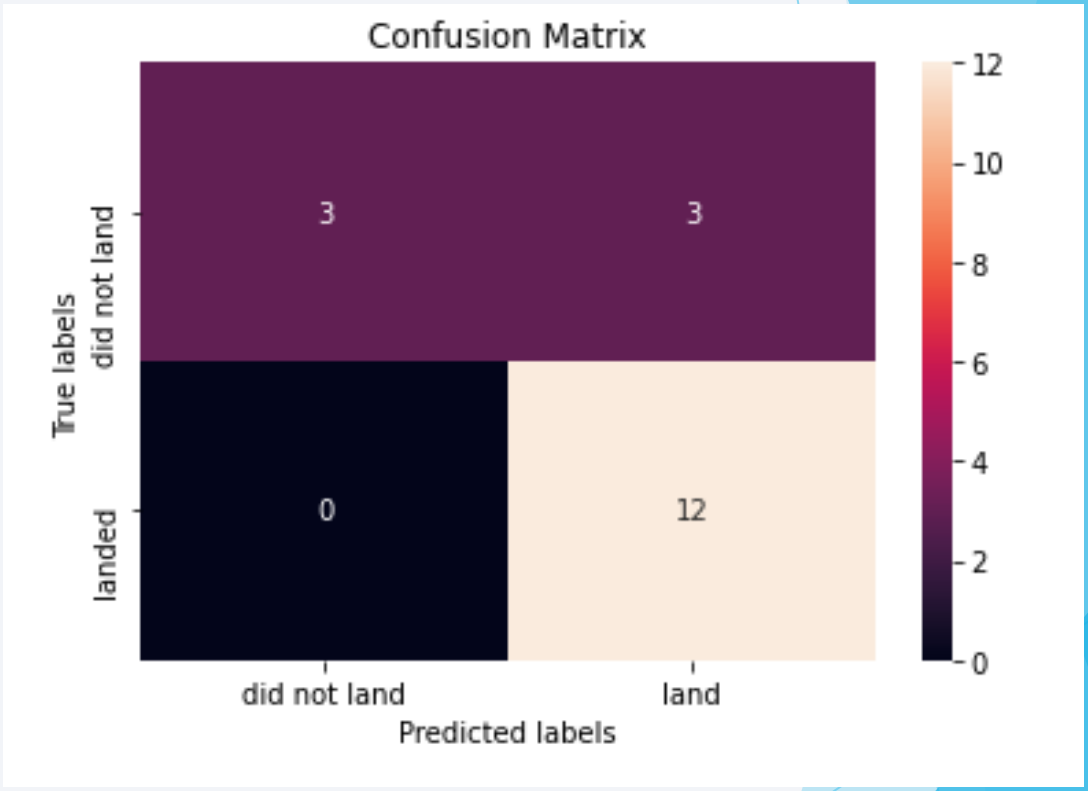
```
models = {'KNeighbors':knn_cv.best_score_,
              'DecisionTree':tree_cv.best_score_,
              'LogisticRegression':logreg_cv.best_score_,
              'SupportVector': svm_cv.best_score_}

bestalgorithm = max(models, key=models.get)
print('Best model is', bestalgorithm,'with a score of', models[bestalgorithm])
if bestalgorithm == 'DecisionTree':
    print('Best params is :', tree_cv.best_params_)
if bestalgorithm == 'KNeighbors':
    print('Best params is :', knn_cv.best_params_)
if bestalgorithm == 'LogisticRegression':
    print('Best params is :', logreg_cv.best_params_)
if bestalgorithm == 'SupportVector':
    print('Best params is :', svm_cv.best_params_)
```

```
Best model is DecisionTree with a score of 0.8732142857142856
Best params is : {'criterion': 'gini', 'max_depth': 6, 'max_features': 'auto', 'min_samples_leaf': 2, 'min_samples_split': 5, 'splitter': 'random'}
```

- We are seeking the model with the best model accuracy.
- In this case it was a Decision Tree

Confusion Matrix

- This is the Confusion Matrix for the Decision Tree model
- The model is highly accurate for Successful Landings, but significantly less so for Unsuccessful Landings



Conclusions

We can conclude that:

- The larger the flight amount at a launch site, the greater the success rate at a launch site.
- Launch success rate has increased steadily between 2013 and 2020.
- Launches to ES-L1, GEO, HEO, SSO, VLEO orbits had the highest success rates.
- KSC LC-39A had the most successful launches of any sites.
- The Decision Tree classifier is the best machine learning algorithm for this task.

Thank you!