IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

<Name>
<Date>

Outline

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

Executive Summary

- **Methodologies**

- data collection with SpaceX API and web scraping from Wikipedia

- EDA with visualization

EDA with SQL

- Dashboard for interactive data analysis and geospacer analysis

- Machine learning classification models evaluation

- **Summary of all results**

- Relationship between variables

- Model selection

- Effective prediction

Introduction

- **Project background and context**

SpaceX is a private aerospace agency that provides services of transportation and launches using the Falcon 9 rocket; a rocket that can land its first stage for it to be reutilized, reducing cost in historical amounts.

- **Problem**

We want to know what determines if a landing is successful and create a model to make predictions of landing success.

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - The data was collected using web scraping from Wikipedia and with the SpaceX API
- Perform data wrangling
  - Categorical features were treated with One Hot Encoding
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - Sci Kit Learn was used to develop, optimize and test ML models

Data Collection

- With the SpaceX API and get request method the data was collected and later the json from the data was converted to a pandas dataframe.
- The data was cleaned and missing value were treated, taking care of the integrity of the data.
- Web Scraping from Wikipedia was performed in order to obtain launch records of the Falcon 9. Parsing the HTML tables and converting its information to a pandas dataframe.

Data Collection – SpaceX API

- Use of the SpaceX API

- [Complete Notebook in Git Hub.]()

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

```
response = requests.get(spacex_url)
```

Data Collection - Scraping

- Beautiful Soup was used for the web scraping of data from Wikipedia.

- [Complete Notebook in GitHub.]()

2020 [edit]

In late 2019, Gwynne Shotwell stated that SpaceX hoped for as many as 24 launches for Starlink satellites in 2020,[490] in addition to 14 or 15 non-Starlink launches. At 26 launches, 13 of which for Starlink satellites, Falcon 9 had its most prolific year, and Falcon rockets were second most prolific rocket family of 2020, only behind China's Long March rocket family.[491]

[hide] Flight No.	Date and time (UTC)	Version, Booster[b]	Launch site	Payload[c]	Payload mass	Orbit	Customer	Launch outcome	Booster landing
78	7 January 2020, 02:19:21[492]	F9 B5 ♺ B1049.4	CCAFS, SLC-40	Starlink 2 v1.0 (60 satellites)	15,600 kg (34,400 lb)[5]	LEO	SpaceX	Success	Success (drone ship)
	Third large batch and second operational flight of Starlink constellation. One of the 60 satellites included a test coating to make the satellite less reflective, and thus less likely to interfere with ground-based astronomical observations.[493]								
79	19 January 2020, 15:30[494]	F9 B5 ♺ B1046.4	KSC, LC-39A	Crew Dragon in-flight abort test[495] (Dragon C205.1)	12,050 kg (26,570 lb)	Sub-orbital[496]	NASA (CTS)[497]	Success	No attempt
	An atmospheric test of the Dragon 2 abort system after Max Q. The capsule fired its SuperDraco engines, reached an apogee of 40 km (25 mi), deployed parachutes after reentry, and splashed down in the ocean 31 km (19 mi) downrange from the launch site. The test was previously slated to be accomplished with the Crew Dragon Demo-1 capsule;[498] but that test article exploded during a ground test of SuperDraco engines on 20 April 2019.[419] The abort test used the capsule originally intended for the first crewed flight.[499] As expected, the booster was destroyed by aerodynamic forces after the capsule aborted.[500] First flight of a Falcon 9 with only one functional stage — the second stage had a mass simulator in place of its engine.								
80	29 January 2020, 14:07[501]	F9 B5 ♺ B1051.3	CCAFS, SLC-40	Starlink 3 v1.0 (60 satellites)	15,600 kg (34,400 lb)[5]	LEO	SpaceX	Success	Success (drone ship)
	Third operational and fourth large batch of Starlink satellites, deployed in a circular 290 km (180 mi) orbit. One of the fairing halves was caught, while the other was fished out of the ocean.[502]								
81	17 February 2020, 15:05[503]	F9 B5 ♺ B1056.4	CCAFS, SLC-40	Starlink 4 v1.0 (60 satellites)	15,600 kg (34,400 lb)[5]	LEO	SpaceX	Success	Failure (drone ship)
	Fourth operational and fifth large batch of Starlink satellites. Used a new flight profile which deployed into a 212 km × 386 km (132 mi × 240 mi) elliptical orbit instead of launching into a circular orbit and firing the second stage engine twice. The first stage booster failed to land on the drone ship[504] due to incorrect wind data.[505] This was the first time a flight proven booster failed to land.								
82	7 March 2020, 04:50[506]	F9 B5 ♺ B1059.2	CCAFS, SLC-40	SpaceX CRS-20 (Dragon C112.3 ♺)	1,977 kg (4,359 lb)[507]	LEO (ISS)	NASA (CRS)	Success	Success (ground pad)
	Last launch of phase 1 of the CRS contract. Carries *Bartolomeo*, an ESA platform for hosting external payloads onto ISS.[508] Originally scheduled to launch on 2 March 2020, the launch date was pushed back due to a second stage engine failure. SpaceX decided to swap out the second stage instead of replacing the faulty part.[509] It was SpaceX's 50th successful landing of a first stage booster, the third flight of the Dragon C112 and the last launch of the cargo Dragon spacecraft.								
83	18 March 2020, 12:16[510]	F9 B5 ♺ B1048.5	KSC, LC-39A	Starlink 5 v1.0 (60 satellites)	15,600 kg (34,400 lb)[5]	LEO	SpaceX	Success	Failure (drone ship)
	Fifth operational launch of Starlink satellites. It was the first time a first stage booster flew for a fifth time and the second time the fairings were reused (Starlink flight in May 2019).[511] Towards the end of the first stage burn, the booster suffered premature shut down of an engine, the first of a Merlin 1D variant and first since the CRS-1 mission in October 2012. However, the payload still reached the targeted orbit.[512] This was the second Starlink launch booster landing failure in a row, later revealed to be caused by residual cleaning fluid trapped inside a sensor.[513]								
84	22 April 2020, 19:30[514]	F9 B5 ♺ B1051.4	KSC, LC-39A	Starlink 6 v1.0 (60 satellites)	15,600 kg (34,400 lb)[5]	LEO	SpaceX	Success	Success (drone ship)

TASK 1: Request the Falcon9 Launch Wiki page from its URL

First, let's perform an HTTP GET method to request the Falcon9 Launch HTML page, as an HTTP response.

```python
# use requests.get() method with the provided static_url
# assign the response to a object
data = requests.get(static_url).text
```

Create a `BeautifulSoup` object from the HTML `response`

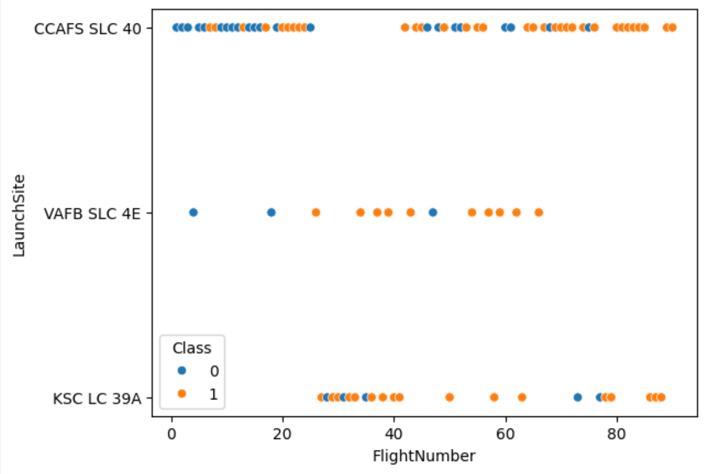
```python
# Use BeautifulSoup() to create a BeautifulSoup object from a response text content
soup = BeautifulSoup(data, 'html5lib')
```

Data Wrangling

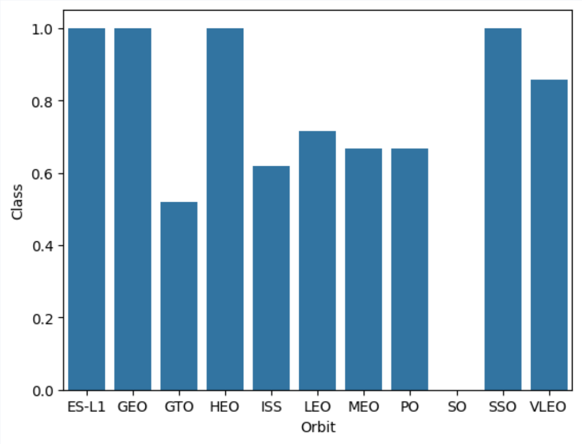
- First, we examined the missing values and data types.
- We differentiate between numerical and categorical data type in columns.
- Created a “class” column to make the launch success or fail a numerical value with 1 and 0.
- Complete Notebook in GitHub.

EDA with Data Visualization

- We created mainly scatter plots colored by the outcome of the landing to understand the relationship between magnitudes like flight number and payload mass with other variables like launch site or orbit type.

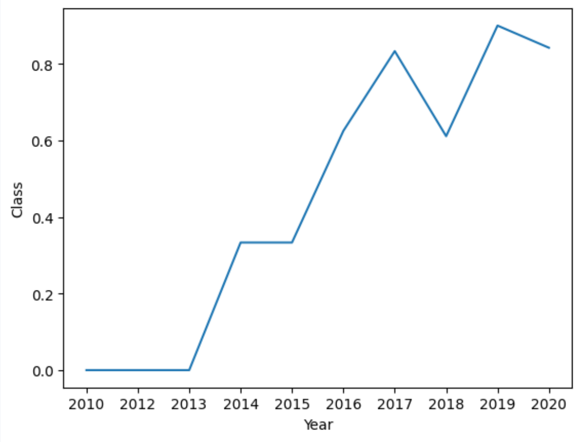


- Bar plot to see the success rate of each orbit type.



- Line plot to show a time series for the success of the landings.

- [Notebook to see the visualizations.]

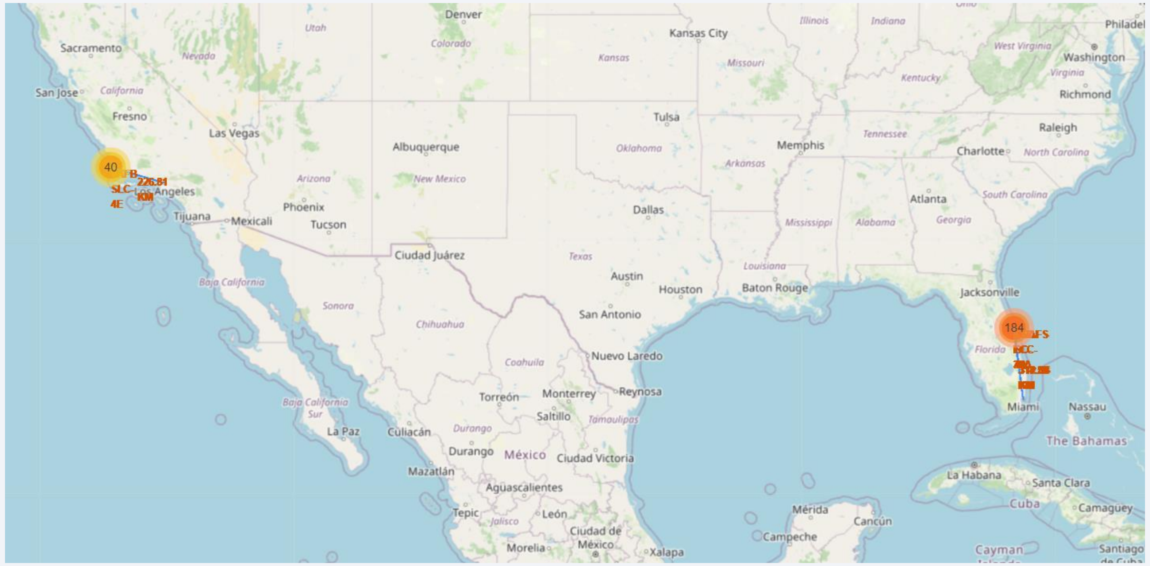


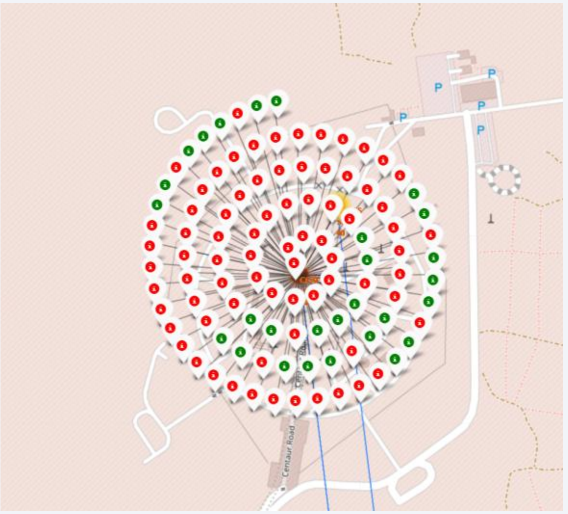
EDA with SQL

- Queries to find the name of all launch sites and the launch sites with “CCA” in its name.
- Query to find the total payload mass carried and launched by NASA
- Query to find mean payload mass carried by the Falcon 9 v1.1
- Query to find the date of the first successful landing
- Query to find the name of the boosters with a successful landing in drone ship and that are in a specific payload mass range.
- Query to find the number of success vs failure in missions
- Query to find the booster versions that have carried more mass
- Query to summarize the data for drone ship failures
- Query to rank the landing outcomes
- [Notebook with all the SQL queries]

Build an Interactive Map with Folium

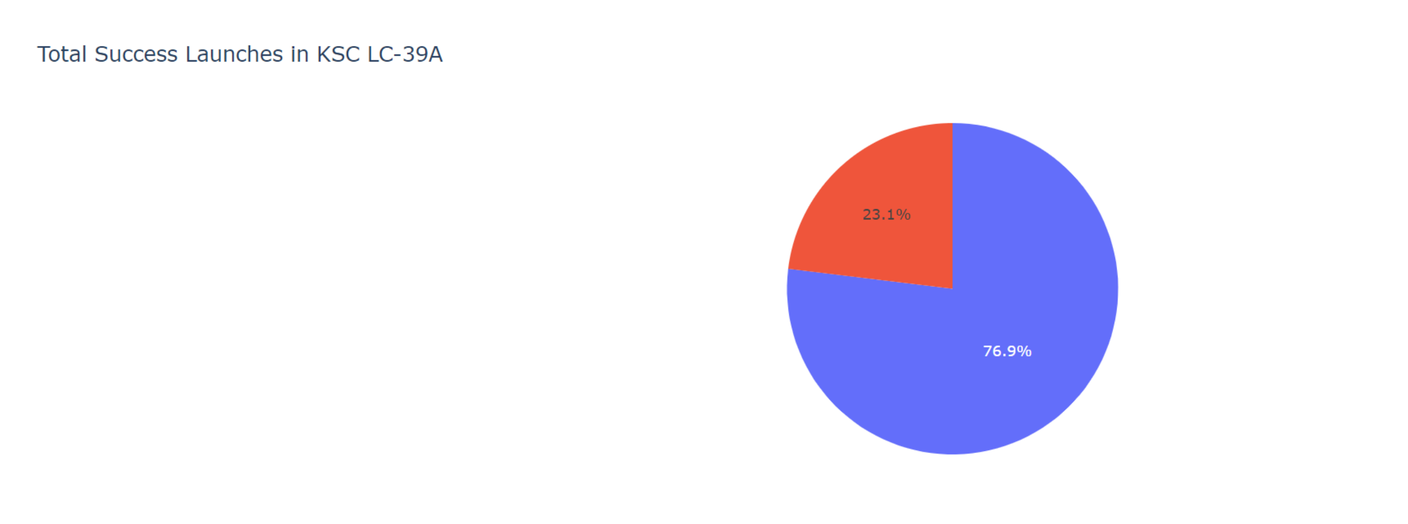
- We marked each launch site and its success and failure cases. We also measured its distance to coastlines, cities, and other important locations to understand if there was any relation between the outcome and the nature of the launch location
- [Complete Notebook in GitHub.]()





Build a Dashboard with Plotly Dash

- The dashboard lets the user select the between each or all launch site and then it shows a pie graph with the success vs failure outcomes of the selection and also a scatter plot of the payload mass vs the outcome, coloring the points by its booster version.
- With these characteristics it is easier to find relationships between the main variables in our analysis.
- Complete python code for the app in GitHub.



Predictive Analysis (Classification)

- With our data, first we divided the variables between dependent and independent variables. Then the train and test set were selected with 80% and 20% respectively.
- Testing for logistic regression, support vector machine, decision tree and K-nearest neighbor, first its respective hyperparameters were optimized with GridSearchCV.
- We calculate the accuracy of each model and obtain a confusion matrix plot. Finally comparing its accuracy, we can select the better one.
- Complete Notebook with the development of models.

Results

- We found that the variables *Flight Number, Payload Mass, Launch Site* and *Orbit Type* are related to the outcome of the mission; and we can use it to develop a statistical model.
- A dashboard for gaining insight from the data was created
- The decision tree model had the highest accuracy

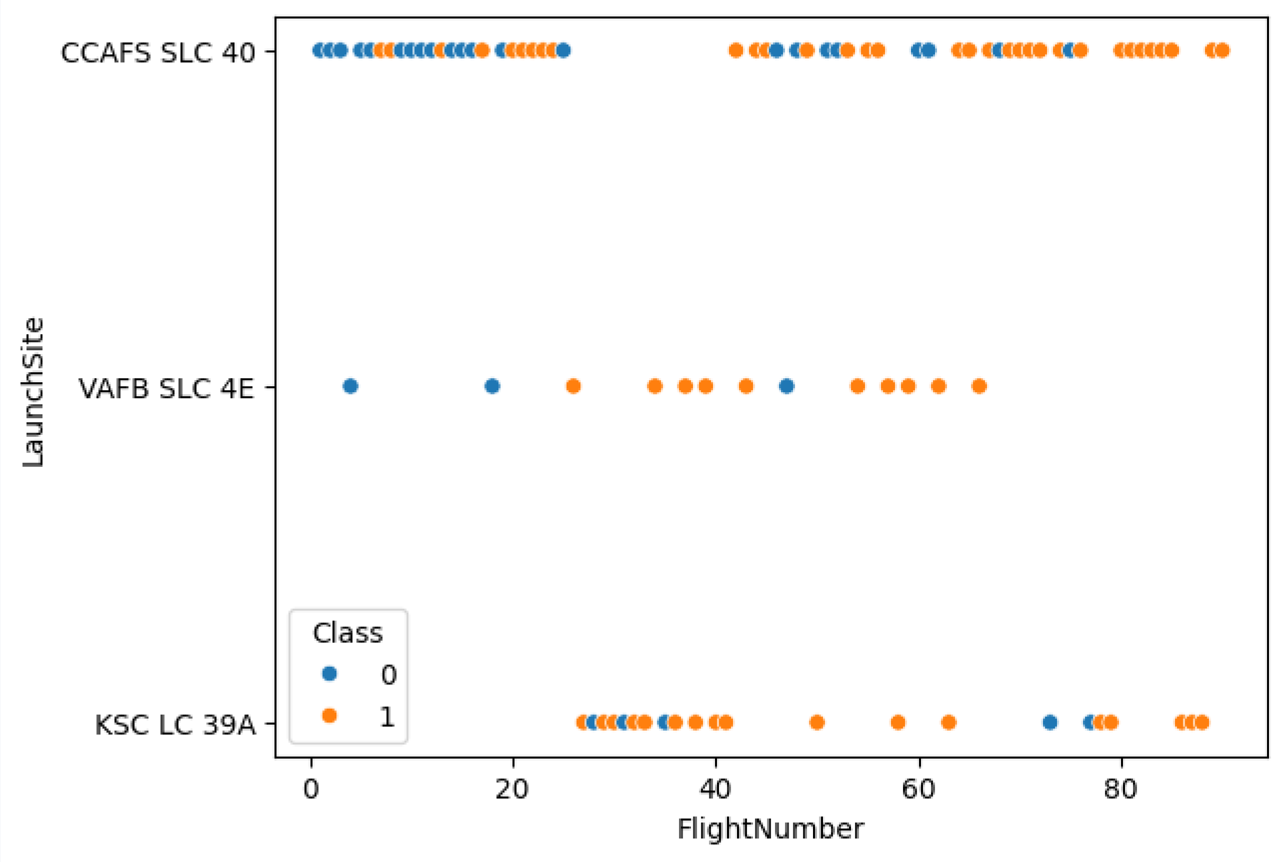
Section 2

Insights drawn from EDA

Flight Number vs. Launch Site

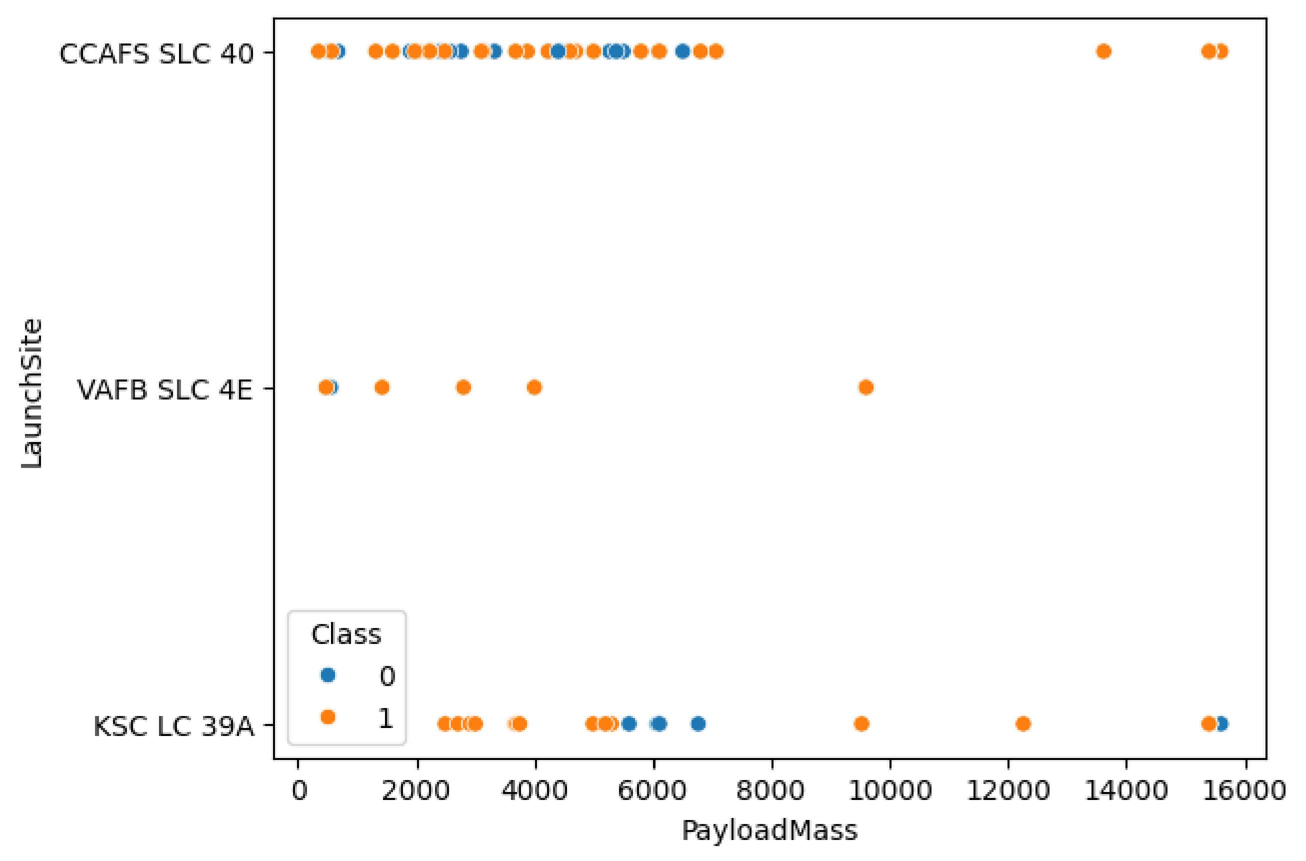
- Figure: Flight Number vs. Launch Site

- For the CCAFS SLC 40 and VAFB SLC 4E launch sites, we can see that after the flight number 20 there are more successful landings.
- While for the KSC LC 39A there aren't any launches before but has more successful landings regularly.



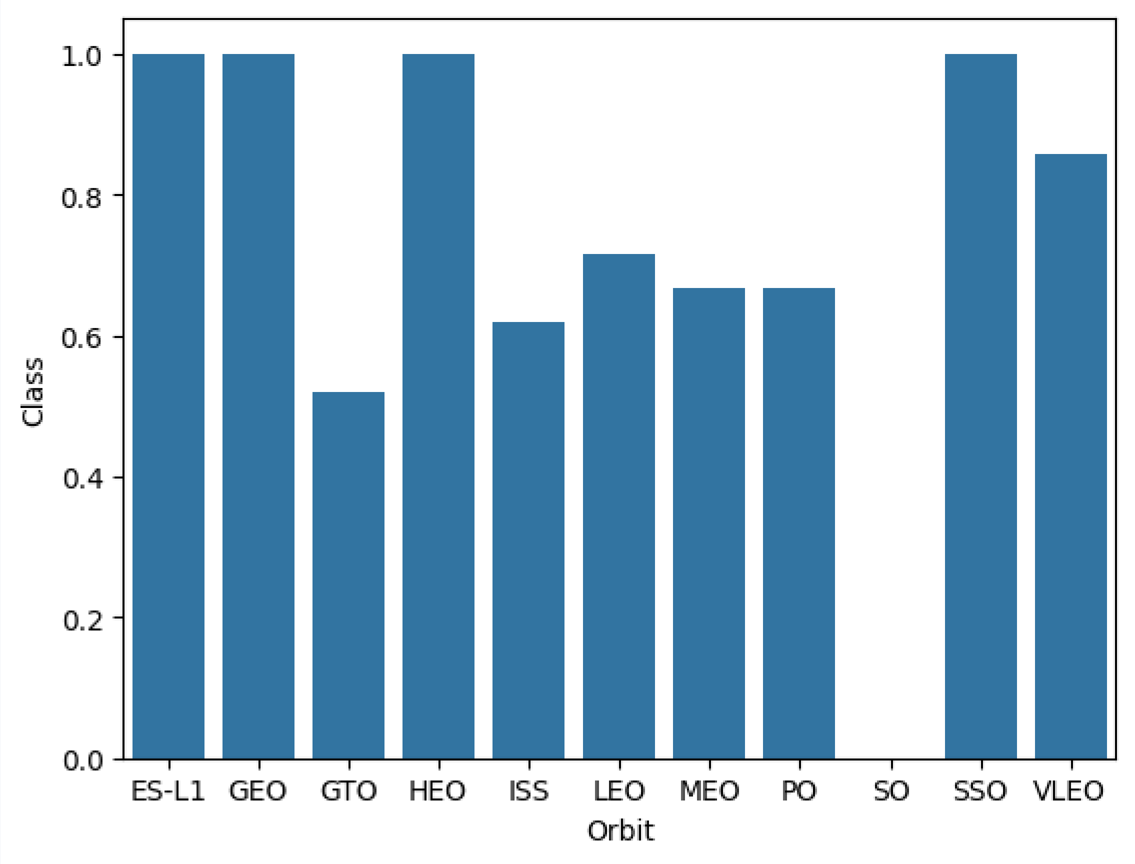
Payload vs. Launch Site

- Figure: Payload vs. Launch Site
- High payload mass launches are less common
- With very few launches, the VAFB SLC 4E launch site has mainly successful landings



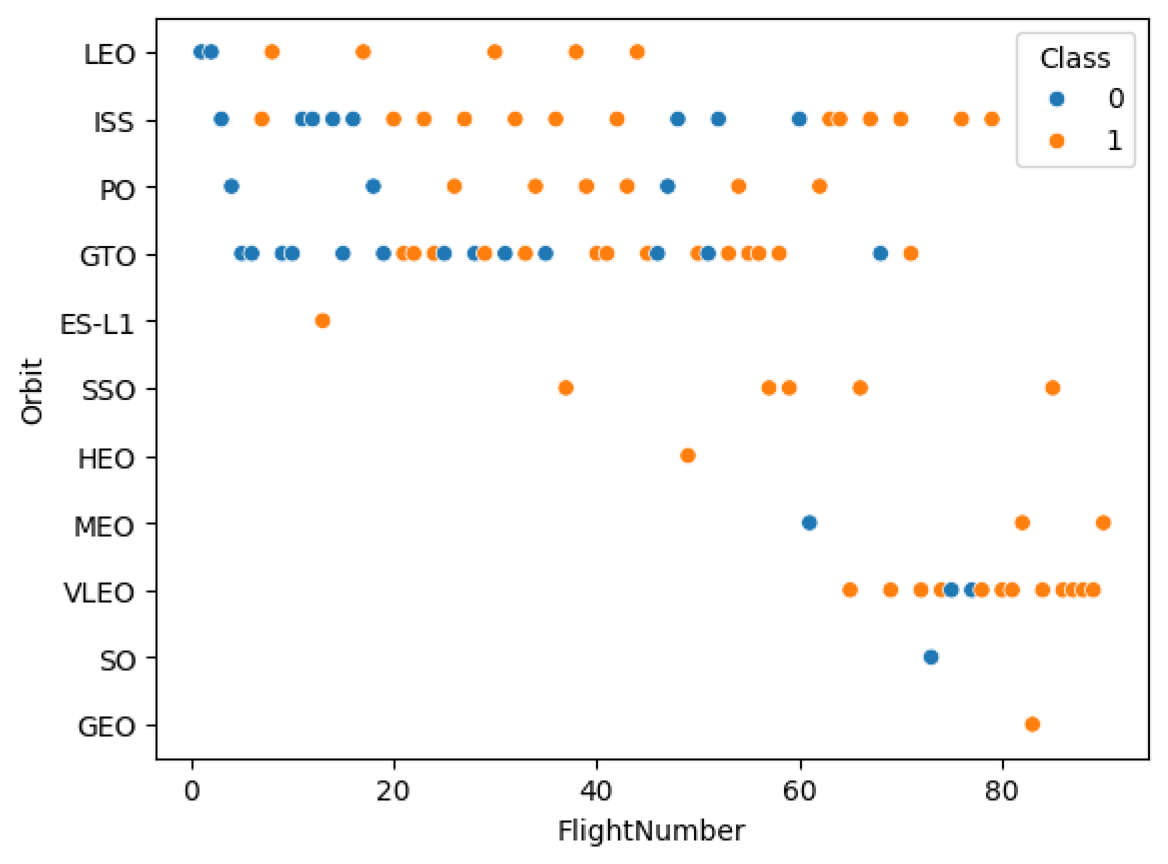
Success Rate vs. Orbit Type

- Figure: Bar plot with the success rate for each orbit type
- ES-L1, GEO, HEO and SSO have success rate of 100%, while SO of 0% however, we can't know from the plot how big or small is the population of each orbit type.



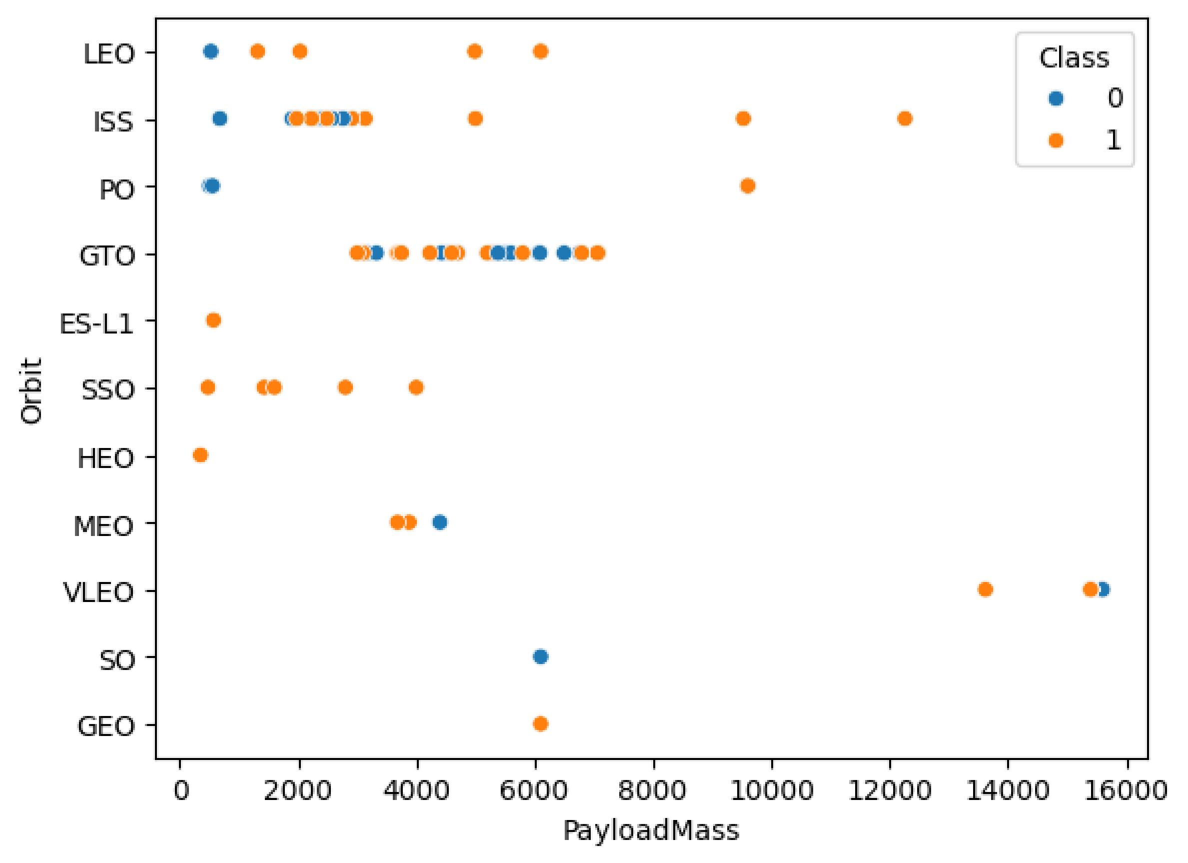
Flight Number vs. Orbit Type

- Flight number vs. Orbit type
- This plot solves some problems from the previous bar plot. We now can see that GEO, SO, HEO and ES-L1 have very low population and making conclusions from there might be problematic without caution. However, for SSO the 100% success rate is well populated
- LEO, ISS, PO and MEO seem to have upgraded their success frequency with flight number.



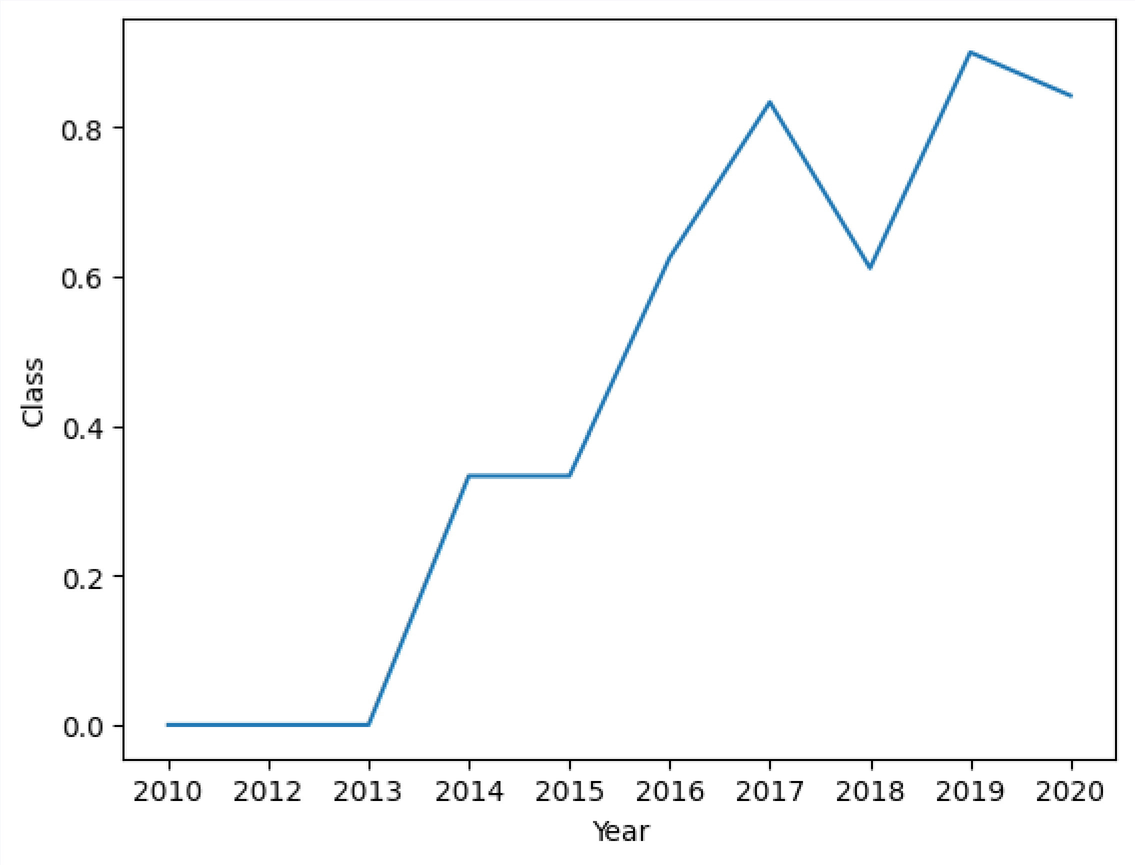
Payload vs. Orbit Type

- Figure payload vs. orbit type
- ISS and VLEO orbits use to have higher payload mass than others
- LEO, GTO, SSO and MEO use to have lower payload mass



Launch Success Yearly Trend

- Figure: yearly average success rate
- It is notorious that with time each year often has a higher success rate
- Between 2015 and 2017 there was a big acceleration in the success rate change
- 2015, 2018 and 2020 are the exception for the growth in the success rate



All Launch Site Names

- To not repeat the Launch_Site values, the use of DISTINCT(Launch_Site) is needed.

```
%%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'

- We used the WHERE clause with a LIKE 'CCA%' to select only the values for Launch_Site that start 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

- With a sum of the PAYLOAD_MASS_KG and selecting only the launches for NASA as Customer we can find the total payload mass for NASA launches to be of 48,213 kg

```
%%sql
SELECT
sum(PAYLOAD_MASS__KG_)
FROM SPACEXTABLE
WHERE Customer LIKE '%NASA (CRS)%';
```

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

sum(PAYLOAD_MASS__KG_)
48213

Average Payload Mass by F9 v1.1

- We can calculate the average payload mass carried by booster version F9 v1.1 using the function AVG(PAYLOAD_MASS_KG_) and selecting with the WHERE clause only the F9 v1.1 Booster_Version. Finding that it is of 2928.4 kg.

```
%%sql
SELECT
avg(PAYLOAD_MASS__KG_)
FROM SPACEXTABLE
WHERE Booster_Version LIKE 'F9 v1.1';
```

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

avg(PAYLOAD_MASS__KG_)
2928.4

First Successful Ground Landing Date

- With the WHERE clause we can find the Landing_Outcome ‘Success (ground pan)’ and select the date only, ordering with ORDER BY in ascendent order and selecting the first element, we find the date of the first success landing in ground pad to be December 22 of 2015.

```
%%sql
SELECT
"Date"
FROM SPACEXTABLE
WHERE Landing_Outcome LIKE 'Success (ground pad)'
ORDER BY "Date" ASC LIMIT 1;
```

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

Date
2015-12-22

Successful Drone Ship Landing with Payload between 4000 and 6000

- To list the names of boosters which have successfully landed on drone ship and had payload mass greater than 4000 but less than 6000, first we select the Booster_Version column from our table, and we used the WHERE clause with double statement, PAYLOAD_MASS_KG has to be in the given range, and also to have Landing_Outcome as ‘Success (drone ship)

```
%%sql
SELECT
Booster_Version
FROM SPACEXTABLE
WHERE PAYLOAD_MASS__KG_ BETWEEN 4000 AND 6000
AND Landing_Outcome LIKE  'Success (drone ship)';
```

```
 * 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

- To calculate the total number of successful and failure mission outcomes we used two subqueries where each one counts the elements of the selection success of failure.

```
%%sql
SELECT
(SELECT count(*) FROM SPACEXTABLE WHERE upper(Mission_Outcome) LIKE '%SUCCESS%') Success,
(SELECT count(*) FROM SPACEXTABLE WHERE upper(Mission_Outcome) LIKE '%FAILURE%') Failure;
```

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

Success	Failure
100	1

Boosters Carried Maximum Payload

- Selecting the Booster_Version column and with the clause WHERE to select only the PAYLOAD_MASS__KG with the same value of max(PAYLOAD_MASS__KG_) obtained from a subquery, we list the names of the booster which have carried the maximum payload mass

```
%%sql
SELECT
Booster_Version
FROM SPACEXTABLE
WHERE PAYLOAD_MASS__KG_ IN (SELECT max(PAYLOAD_MASS__KG_) FROM SPACEXTABLE);
```

```
* 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

- We list the failed landing_outcomes in drone ship, their booster versions, and launch site names for in year 2015 by selecting the mentioned columns alongside date, and with the clause WHERE select only the ‘Failure (drone ship) in Landing_Outcome and the range of 2015 for date.

```
%%sql
SELECT
"Date",
Landing_Outcome,
Booster_Version,
Launch_Site
FROM SPACEXTABLE
WHERE
Landing_Outcome LIKE 'Failure (drone ship)'
AND "Date" BETWEEN '2015-01-01' AND '2015-12-31';
```

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

Date	Landing_Outcome	Booster_Version	Launch_Site
2015-01-10	Failure (drone ship)	F9 v1.1 B1012	CCAFS LC-40
2015-04-14	Failure (drone ship)	F9 v1.1 B1015	CCAFS LC-40

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

- We 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 by selecting the Landing_Outcome column and counting its elements from our table, also with the WHERE clause we select only the data range mentioned, and we group our elements for the sum by Langind_Outcome with GROUP BY, finally to order we use ORDER BY DESC.

```
%%sql
SELECT
Landing_Outcome,
count(Landing_Outcome)
FROM SPACEXTABLE
WHERE "Date" BETWEEN '2010-06-04' AND '2017-03-20'
GROUP BY Landing_Outcome;
```

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

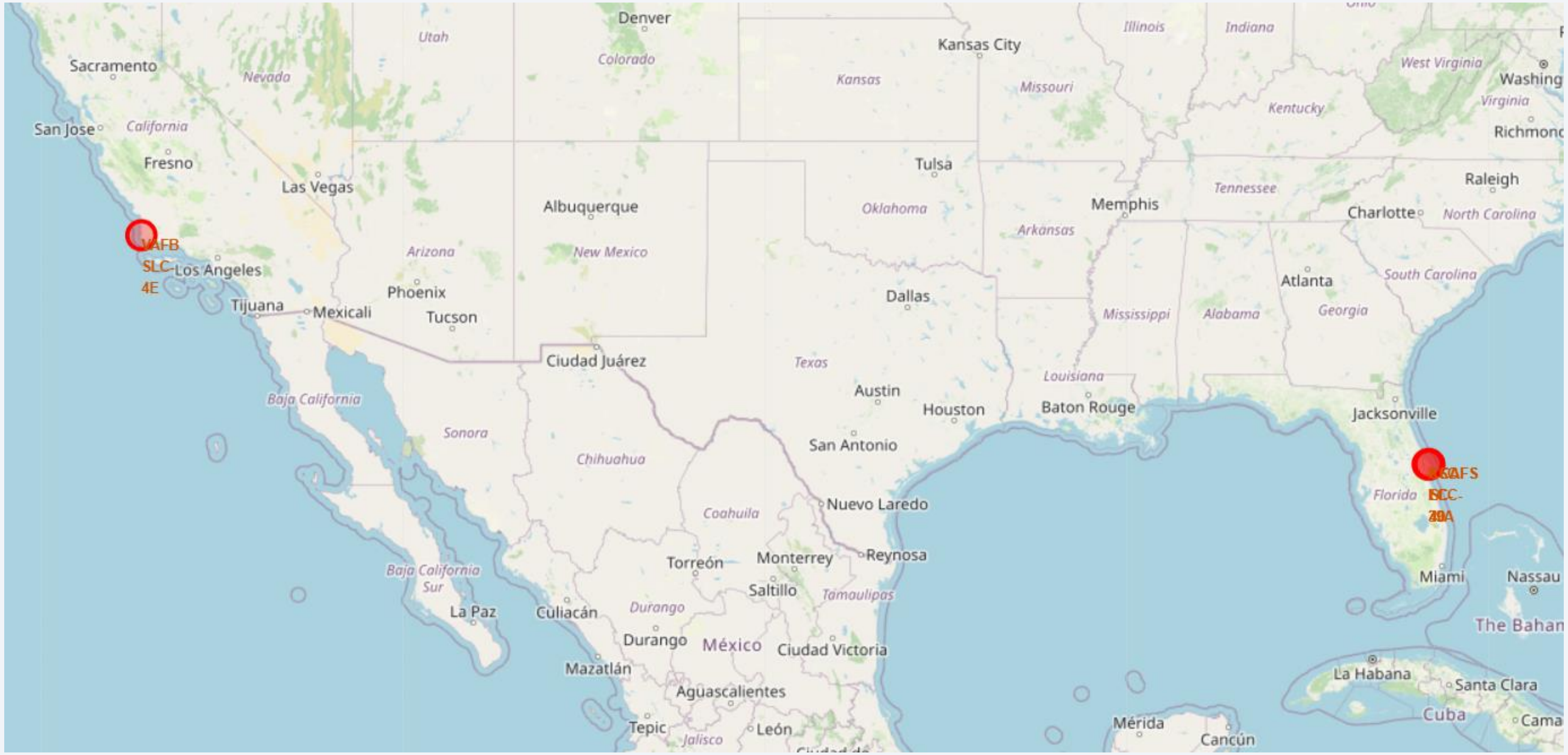
Landing_Outcome	count(Landing_Outcome)
Controlled (ocean)	3
Failure (drone ship)	5
Failure (parachute)	2
No attempt	10
Precluded (drone ship)	1
Success (drone ship)	5
Success (ground pad)	3
Uncontrolled (ocean)	2

Section 3

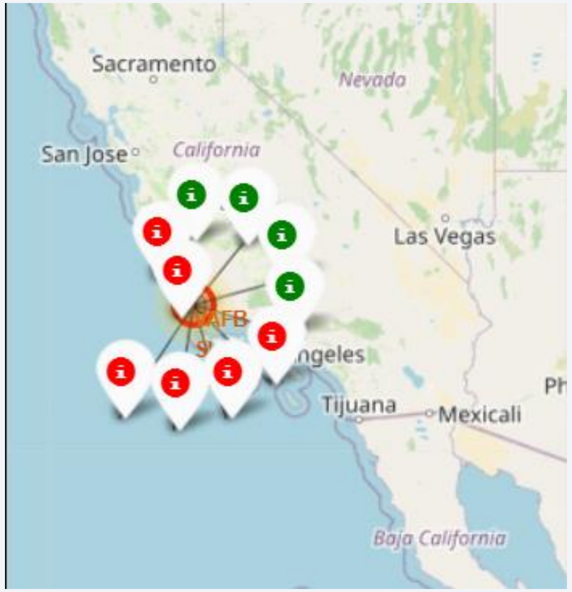
Launch Sites Proximities Analysis

Launch Sites with Folium

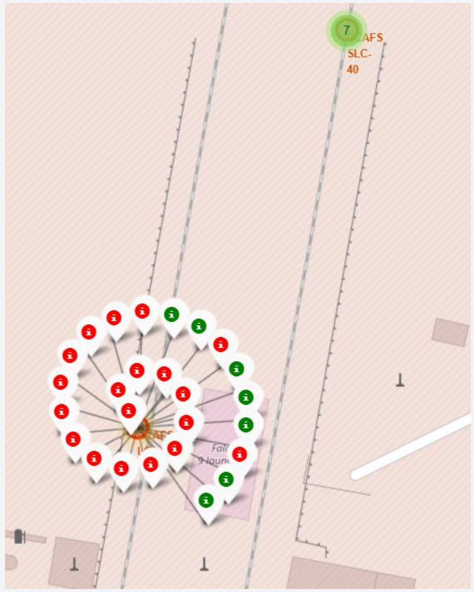
- All launch sites are in the USA, specifically in California and Florida.

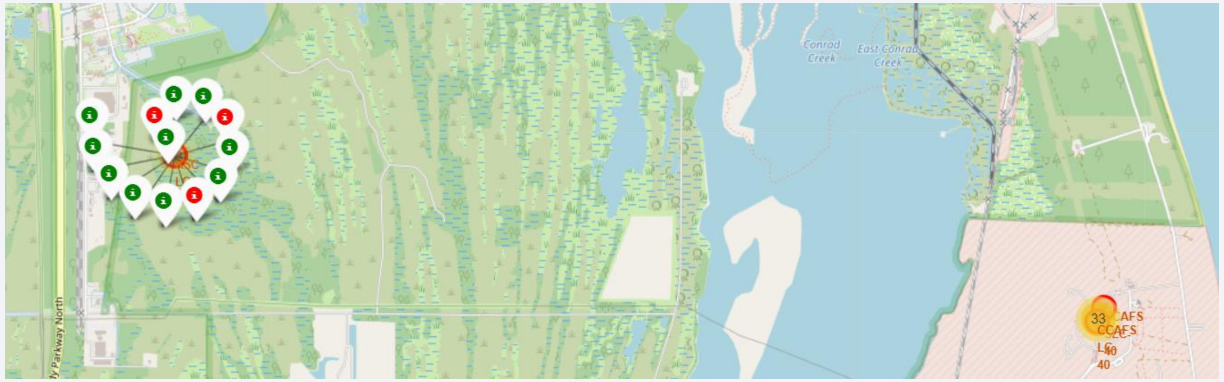


Launch Site by Outcome



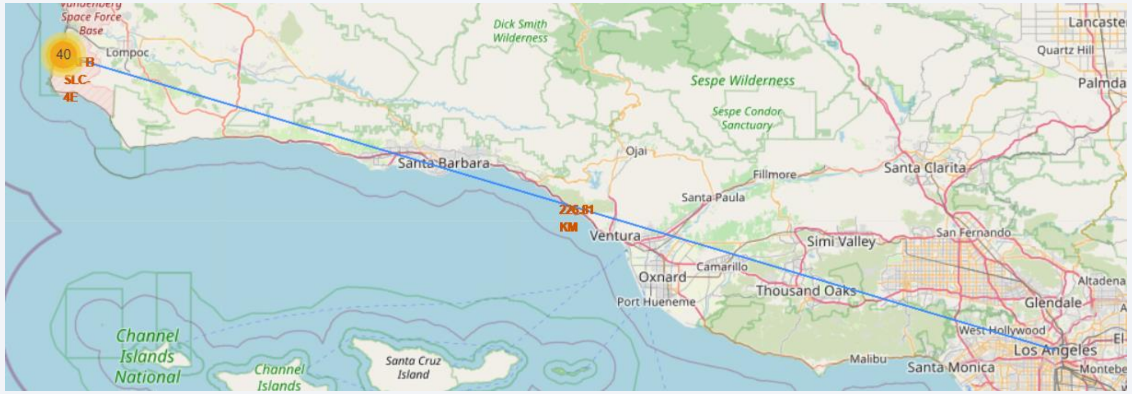
- We can see the landing outcome of each launch by launch site by zooming in and clicking on the site
- Also each circle is colored for easy evaluation of the success rate





Evaluating Distances

- We can evaluate the distance from the launch sites to different places of interest.
- For example, coastlines or cities

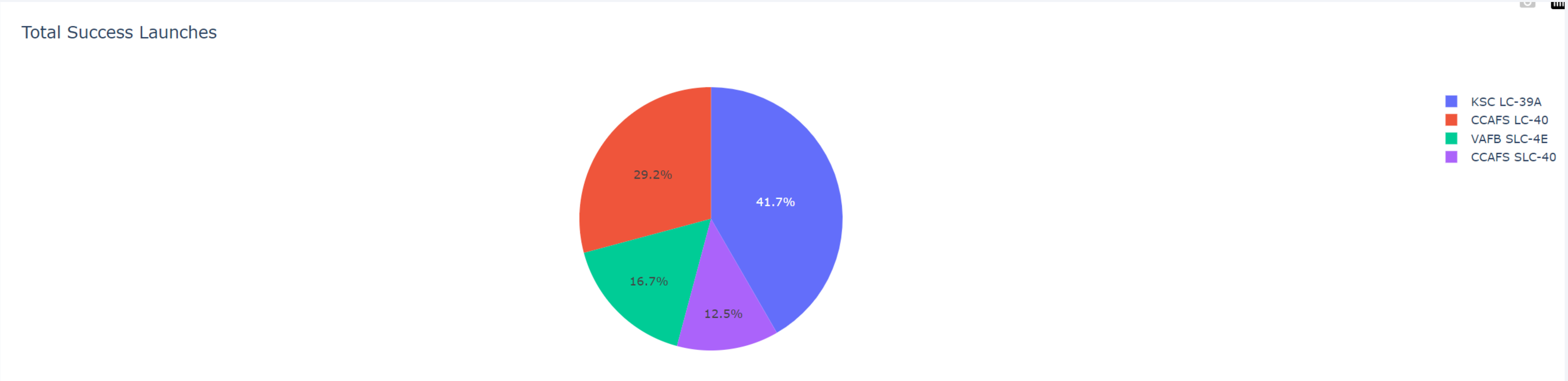


Section 4

Build a Dashboard with Plotly Dash

Success Count by Launch Site

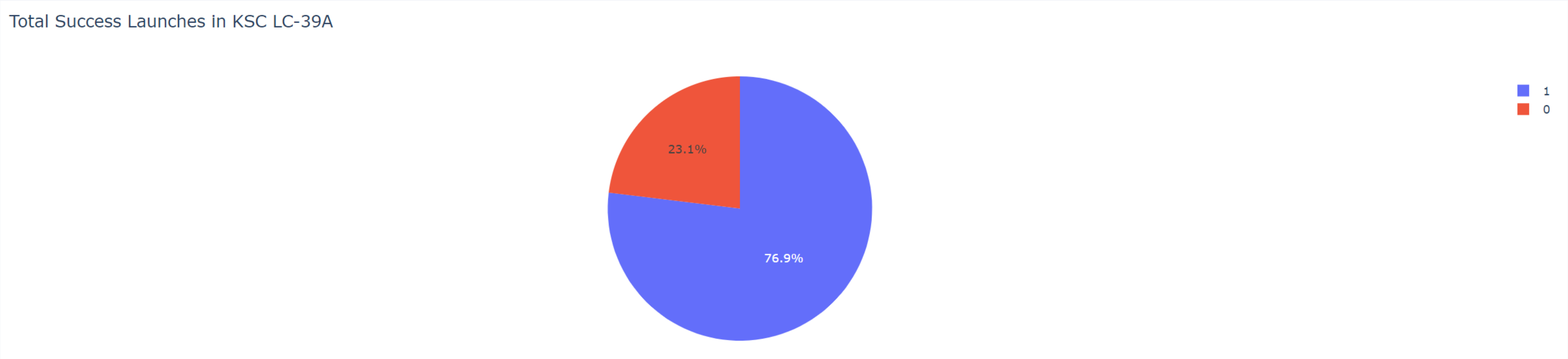
- Figure: Pie chart with the success rate for each launch site



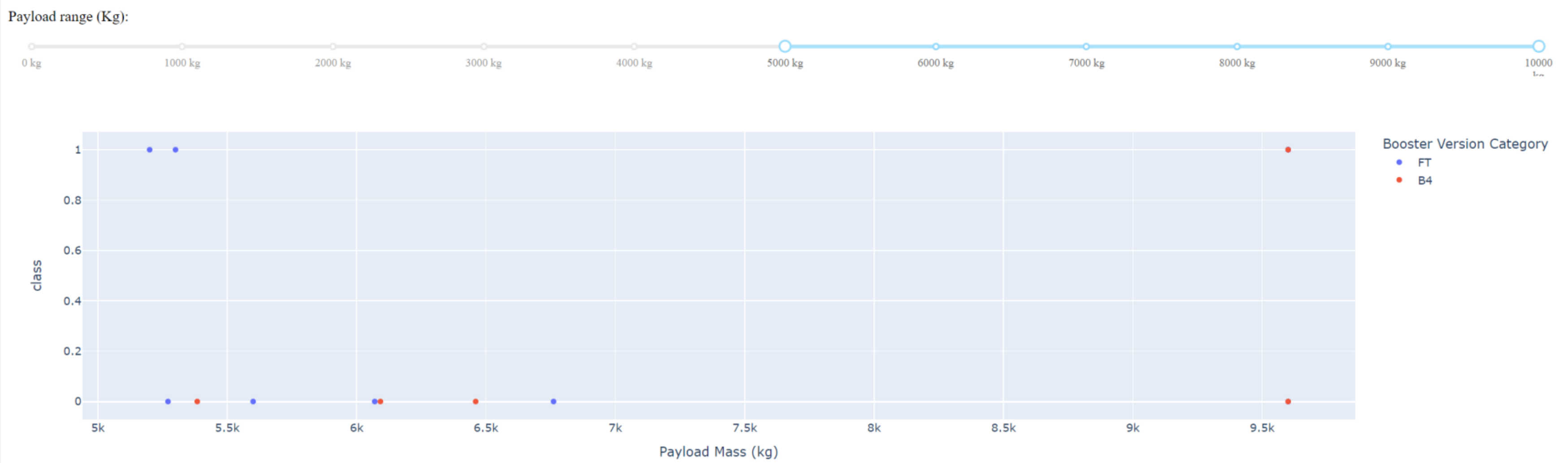
- We can see that the launch site with highest success is KSC LC-39A, with 41.7% of all success

KSC LC-39A success rate

- The success rate of this launch site is 77.9%, with only 23.1% of its landings being classified as failure.



Success and Payload mass



- For heavier payload masses, only FT and B4 booster versions are used
- The success is more common in the lower payload masses

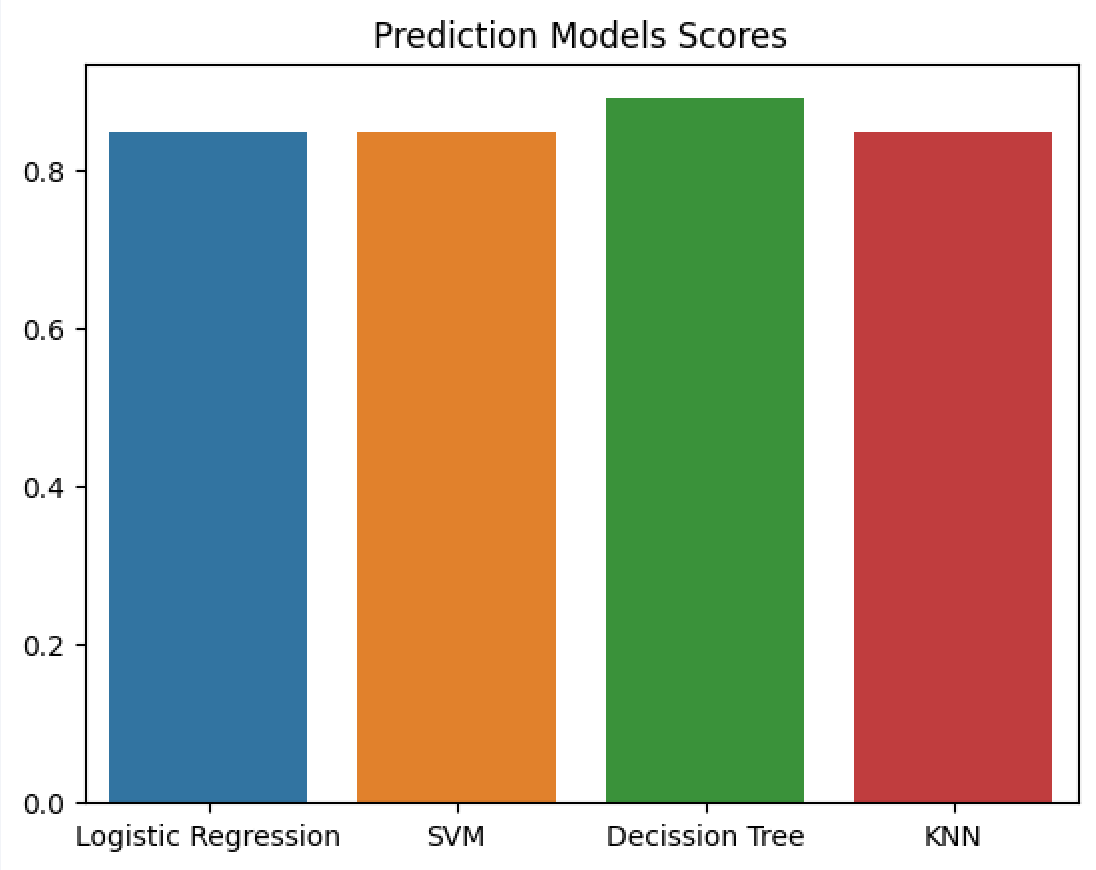
Section 5

Predictive Analysis (Classification)

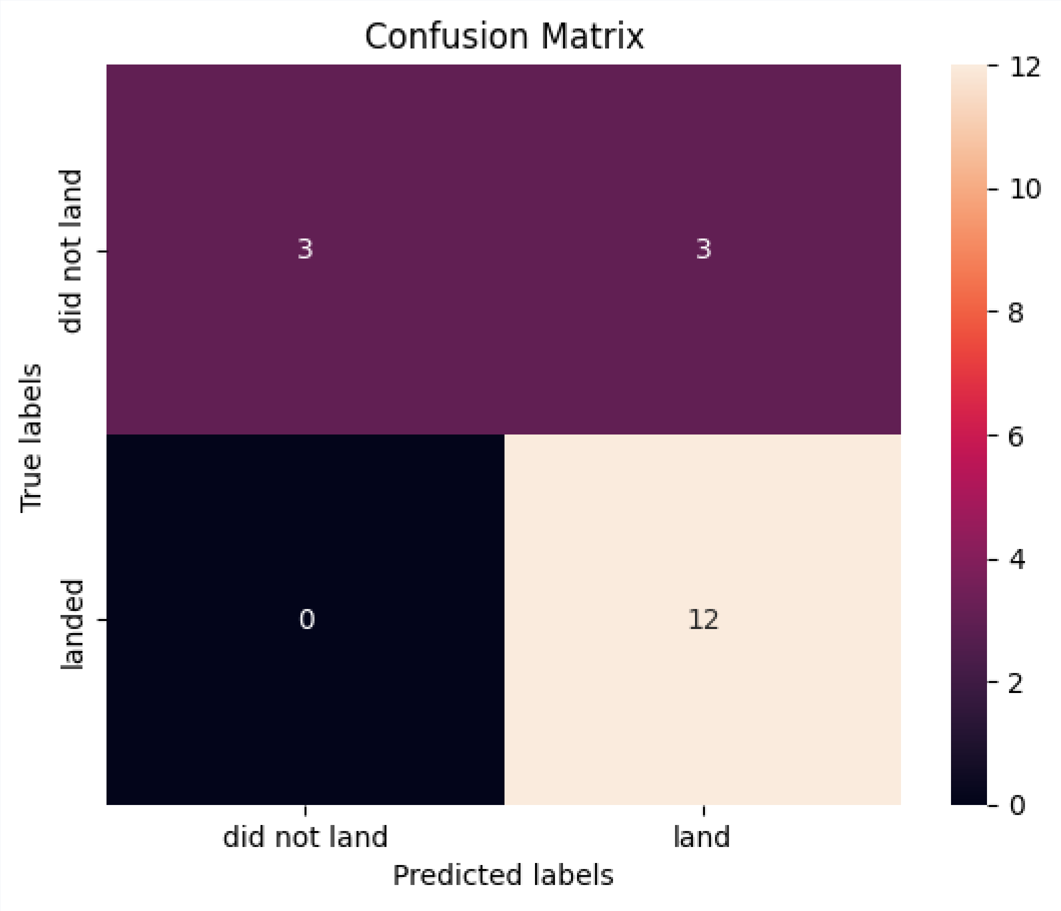
Classification Accuracy

- Figure: bar chart with the accuracy scores for each one of the models evaluated

- The highest accuracy is found in the decision tree model



Confusion Matrix



- Figure: confusion matrix for the decision tree model
- The model predicted effectively all the 12 launches that landed
- While for the not successful landing cases, the model predicted effectively that 3 wouldn’t land, but failed at other 3 flights that didn’t landed and labeled as successful
- The model is better at predicting successful landings than unsuccessful landings

Conclusions

- We found that multiple variables are related between them
- The success rate for the landings have been generally upgrading with time
- The payload mass cargo has been increasing with time, and is significantly influential in the success on a landing
- Multiple classification models were evaluated, with a best fit for a simple decision tree
- We can effectively predict the outcome of a launch mission based on the public data obtained, and therefore use it for a better planning of future missions

Appendix

- Complete GitHub with all relevant files here

Thank you!