IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Ming Zhu
12.06.2023

Outline

- Executive Summary (page 3)
- Introduction (page 4)
- Methodology (page 5-15)
- Results
  - Exploratory data analysis results (page 17-34)
  - Launch site proximities analysis (page 35-38)
  - Interactive analytics demo in screenshots (page 39-42)
  - Predictive analysis results (page 43-45)
- Conclusion (page 46)
- Appendix (page 47)

Executive Summary

- Summary of methodologies
  - Import data from SpaceX API and Wikipedia
  - Getting data ready for analysis by data extracting, cleaning and removing nulls.
  - Data exploration using SQL, plots (scatter, bar, line), interactive maps and dashboard
  - Using machine learning model to predict the successful landing.
- Summary of all results
  - Flight number, payload mass, orbit and launch site all have an impact on the outcome
  - Four ML models we developed all have the same accuracy and same false positive rate, which may be improved when more data is available.

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.

- Problems you want to find answers
  - Which factors contribute to a successful Falcon 9 first stage landing?
  - Can we predict if a landing will be successful or not?

Section 1

Methodology

Methodology

Executive Summary

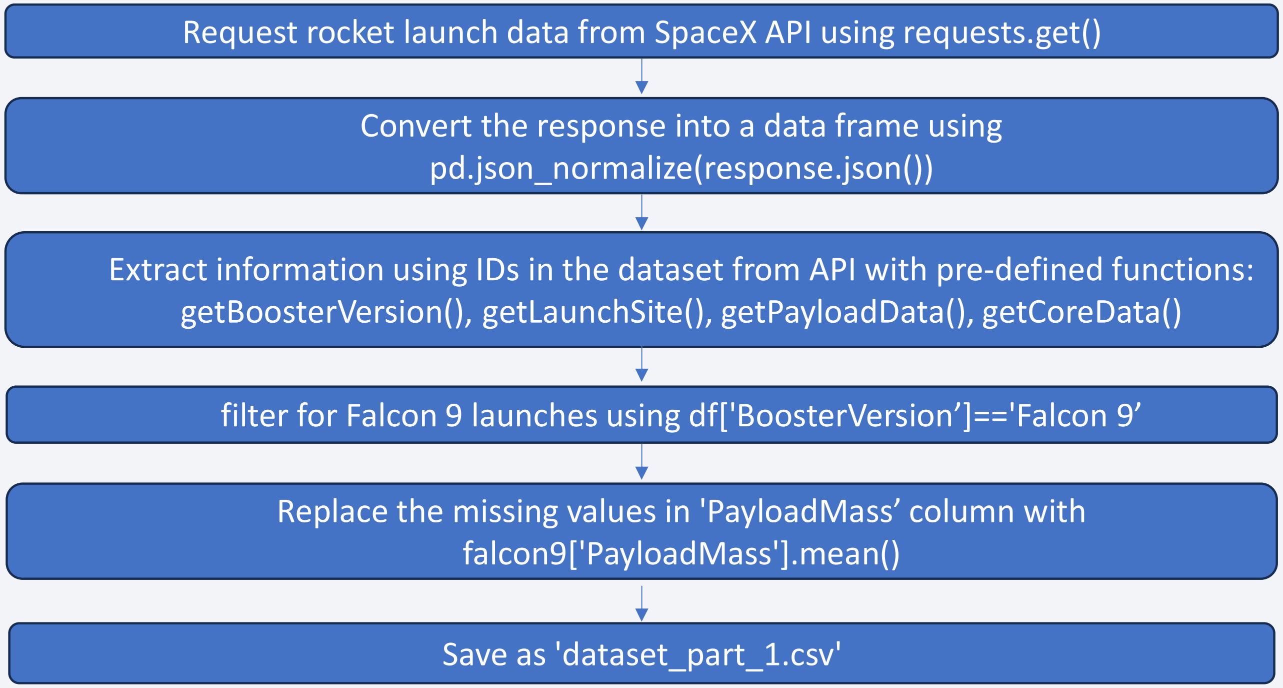
- Data collection methodology:
  - Collected data from SpaceX API and Wikipedia using request.get()
- Perform data wrangling
  - Extract, filter and replace NaN in the 'PayloadMass' column with the mean
  - Convert all categorical columns into numerical by creating landing_class for the outcomes and applying OneHotEncoder to the rest using get_dummies().
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - Standardize the data; split data into training and test sets; build the model using sklearn; check the accuracy using the test data.

Data Collection

- SpaceX API:
  - url: https://api.spacexdata.com/v4/launches/past
  - Request to the SpaceX API.
  - Convert the response JSON file to data frame.
  - Clean the requested data by extracting and filtering the data and replacing the null values using replace() and mean()
- Web scraping Falcon 9 and Falcon Heavy Launches Records from Wikipedia
  - url:https://en.wikipedia.org/w/index.php?title=List_of_Falcon_9_and_Falcon_Heavy_launches&oldid=1027686922
  - Request the Falcon9 Launch HTML response
  - Create a BeautifulSoup object.
  - Parsing and Extracting the data we need and converting the data into a data frame

Data Collection – SpaceX API

Flow Chart



Request rocket launch data from SpaceX API using requests.get()

Convert the response into a data frame using pd.json_normalize(response.json())

Extract information using IDs in the dataset from API with pre-defined functions: getBoosterVersion(), getLaunchSite(), getPayloadData(), getCoreData()

filter for Falcon 9 launches using df['BoosterVersion’]=='Falcon 9’

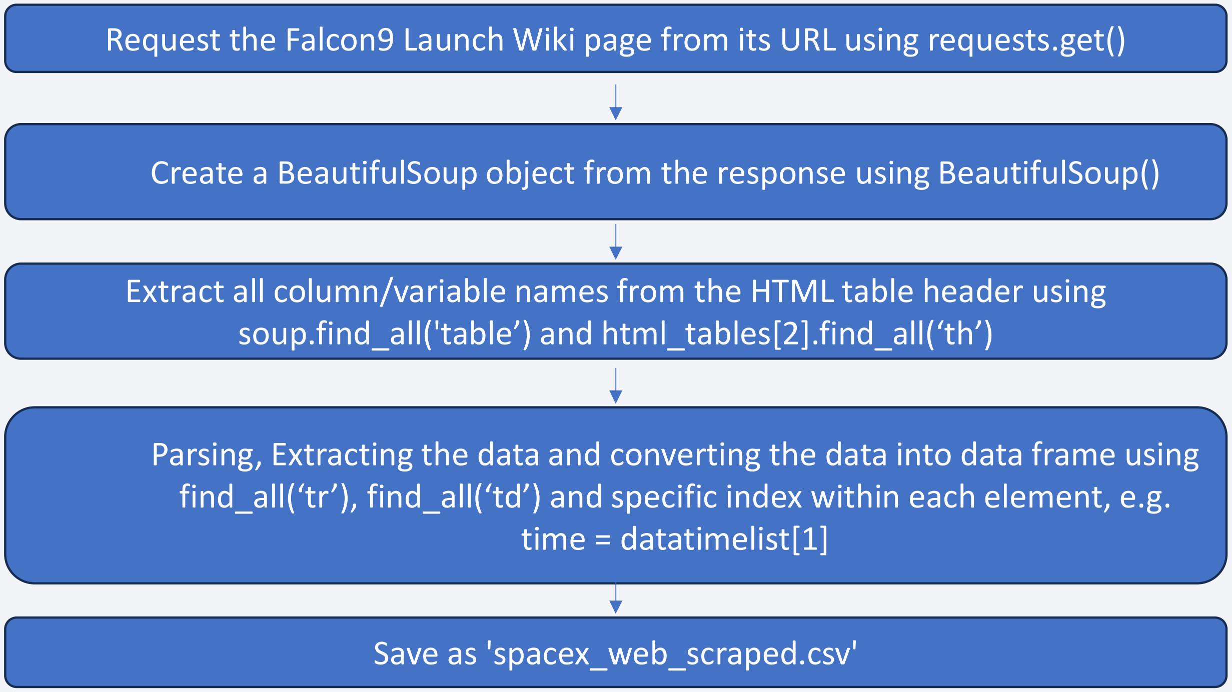
Replace the missing values in 'PayloadMass’ column with falcon9['PayloadMass'].mean()

Save as 'dataset_part_1.csv'

Github link to Data Collection from SpaceX API

Data Collection - Scraping

Flow Chart



Request the Falcon9 Launch Wiki page from its URL using requests.get()

Create a BeautifulSoup object from the response using BeautifulSoup()

Extract all column/variable names from the HTML table header using soup.find_all('table’) and html_tables[2].find_all(‘th’)

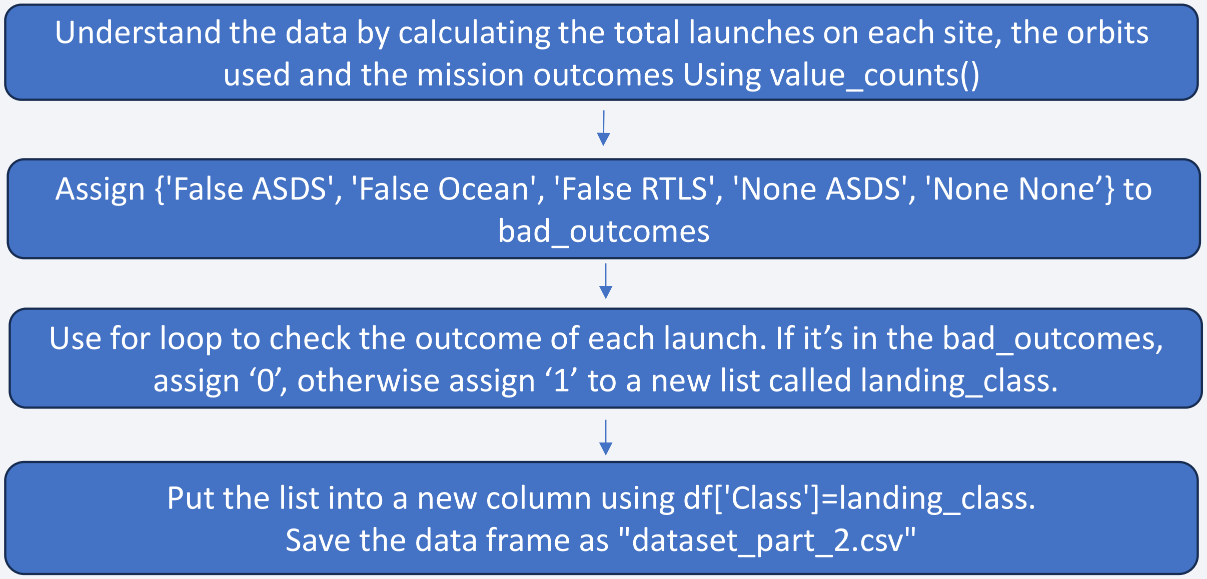
Parsing, Extracting the data and converting the data into data frame using find_all(‘tr’), find_all(‘td’) and specific index within each element, e.g. time = datatimelist[1]

Save as 'spacex_web_scraped.csv'

GitHub link to web scraping from Wikipedia

Data Wrangling

Flow Chart



Understand the data by calculating the total launches on each site, the orbits used and the mission outcomes Using value_counts()

Assign {'False ASDS', 'False Ocean', 'False RTLS', 'None ASDS', 'None None’} to bad_outcomes

Use for loop to check the outcome of each launch. If it’s in the bad_outcomes, assign ‘0’, otherwise assign ‘1’ to a new list called landing_class.

Put the list into a new column using df['Class']=landing_class.
Save the data frame as "dataset_part_2.csv"

GitHub link to data wrangling

EDA with Data Visualization

- I used scatter plot to visualize the relation between flight number, payload mass, orbit, and launch site, as well as their effect on the launch outcome.
- I used bar chart to compare the success rate of each orbit side-by-side.
- I used line chart to visualize the trend of success launches over the years.
- Input data file “dataset_part_2.csv”, output “dataset_part_3.csv”.

GitHub link to EDA dataviz

EDA with SQL

- Connect to db: %load_ext sql; con = sqlite3.connect("my_data1.db"); cur = con.cursor(); sqlite:///my_data1.db

1. select distinct Launch_Site from SPACEXTABLE
2. select * from SPACEXTABLE where Launch_Site like 'KSC%' limit 5
3. select sum(PAYLOAD_MASS__KG_) from SPACEXTABLE where Customer=='NASA (CRS)’
4. select avg (PAYLOAD_MASS__KG_) from SPACEXTABLE where Booster_Version like 'F9 v1.1%’
5. select Date, Landing_Outcome from SPACEXTABLE where Landing_Outcome=='Success (drone ship)’
6. select Booster_Version, Landing_Outcome, PAYLOAD_MASS__KG_ from SPACEXTABLE where Landing_Outcome=='Success (ground pad)' AND (PAYLOAD_MASS__KG_) between 4000 and 6000
7. select Mission_Outcome,count(*) from SPACEXTABLE where Mission_Outcome like 'Success%’ Union allselect Mission_Outcome,count(*) from SPACEXTABLE where Mission_Outcome like 'Failure%’
8. select Booster_Version, PAYLOAD_MASS__KG_ from SPACEXTABLE where PAYLOAD_MASS__KG_ =(select max(PAYLOAD_MASS__KG_) from SPACEXTABLE)
9. select substr(Date,0,5) as year, substr(Date,6,2) as month, Booster_Version, Launch_Site, Landing_Outcomefrom SPACEXTABLEwhere year ='2017' and Landing_Outcome=='Success (ground pad)’;
10. select Landing_Outcome, count(*) as counts from SPACEXTABLE where Date between '2010-06-04' and '2017-03-20'group by Landing_Outcomeorder by counts desc

GitHub link to EDA-SQL

Build an Interactive Map with Folium

- To mark all launch sites on a map, I added a circle with label for each site according to their latitude and longitude coordinates using folium.Circle(), folium.map.Marker() and add_child()
- To see which sites have high success rates, I marked the success/failed launches for each site on the map. I used MarkerCluster() to put all markers that have the same coordinate together and used green color for a successful launch and red for a failed one.
- I got the coordinates of the nearest coastline and railway using MousePosition() and calculated the distances between them and CCAFS LC-40 using pre-defined calculate_distance(). I marked these points on the map using folium.Marker() and drew lines between the launch site and these points using folium.PolyLine().

GitHub link

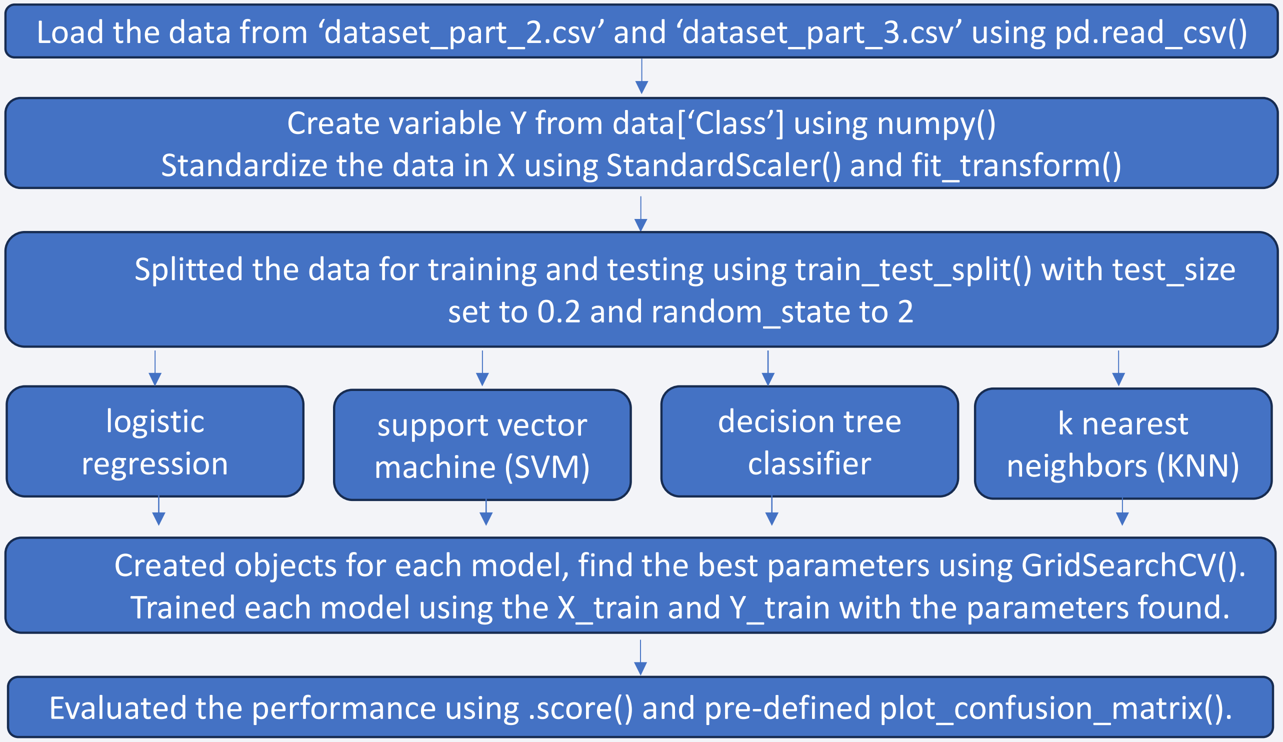
Build a Dashboard with Plotly Dash

- I added a launch site drop-down input component using dcc.Dropdown(). The options in the drop-down are ‘All’ for all site and the name of each site.
- I added a callback function to render success-pie-chart based on the site selected in the drop-down using @app.callback().
- I added a range slider to select payload using dcc.RangeSlider().
- I added a callback function to render the success-payload-scatter-chart using @app.callback() with selections in both drop-down and slider as the input.

GitHub link to the notebook for dashboard

Predictive Analysis (Classification)

Flow Chart



Load the data from ‘dataset_part_2.csv’ and ‘dataset_part_3.csv’ using pd.read_csv()

Create variable Y from data[‘Class’] using numpy()
Standardize the data in X using StandardScaler() and fit_transform()

Splitted the data for training and testing using train_test_split() with test_size set to 0.2 and random_state to 2

logistic regression

support vector machine (SVM)

decision tree classifier

k nearest neighbors (KNN)

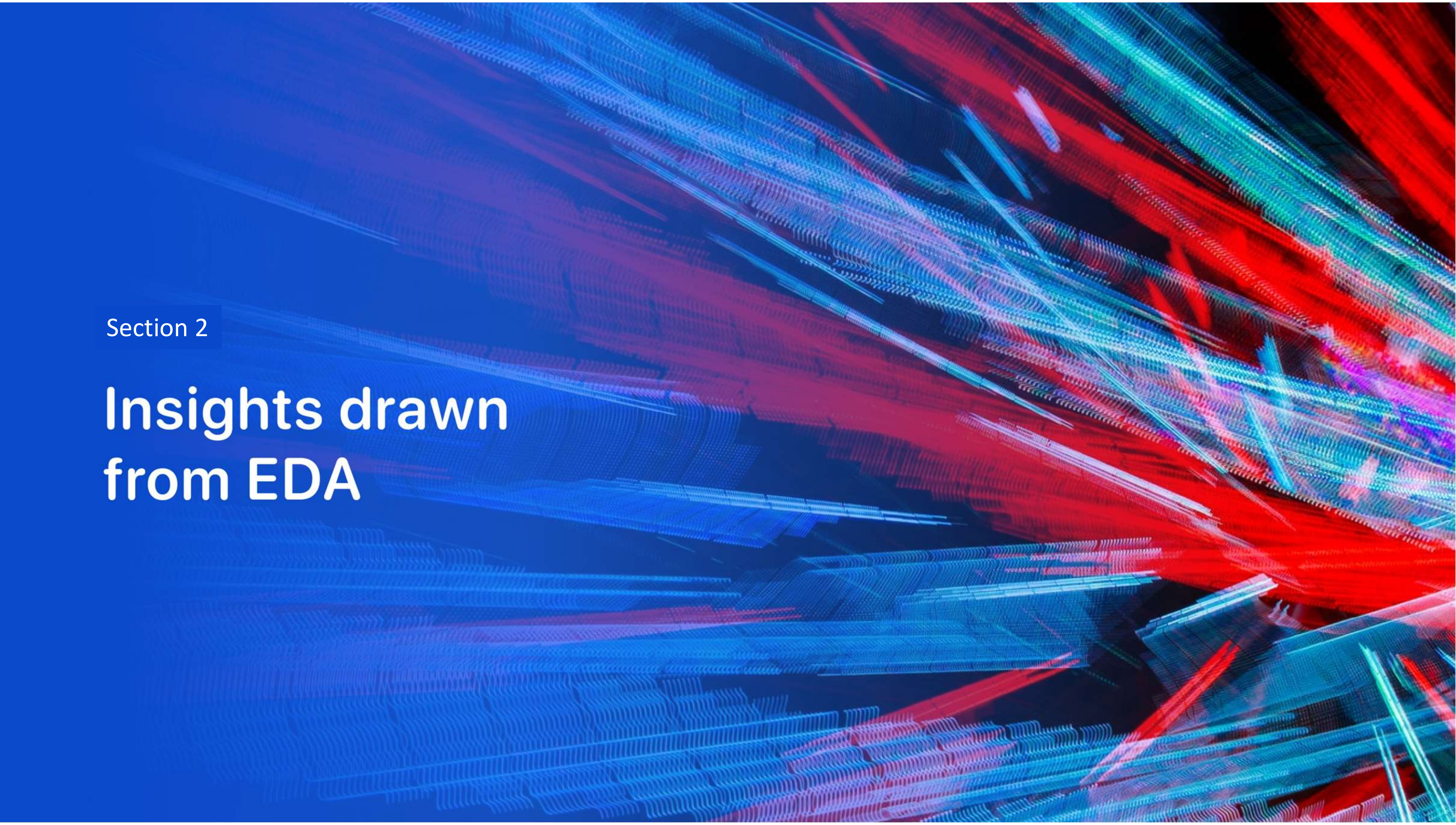
Created objects for each model, find the best parameters using GridSearchCV().
Trained each model using the X_train and Y_train with the parameters found.

Evaluated the performance using .score() and pre-defined plot_confusion_matrix().

GitHub to the ML results

Results

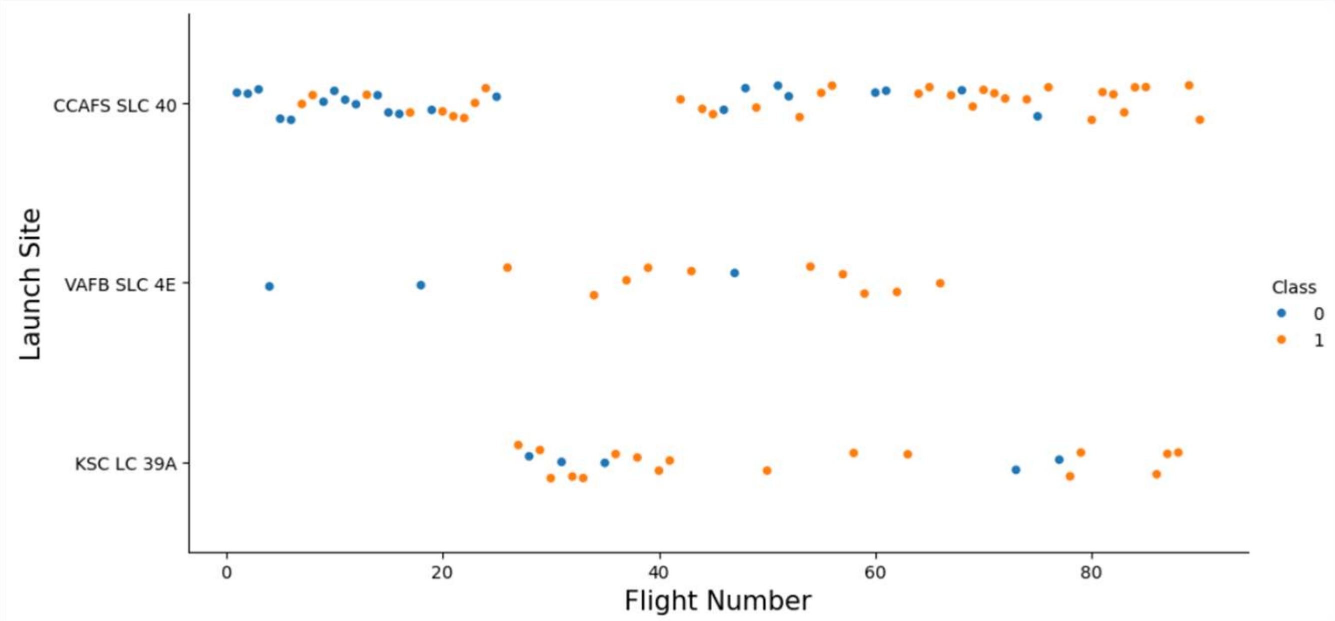
- Exploratory data analysis results
- Interactive analytics demo in screenshots
- Predictive analysis results



Section 2

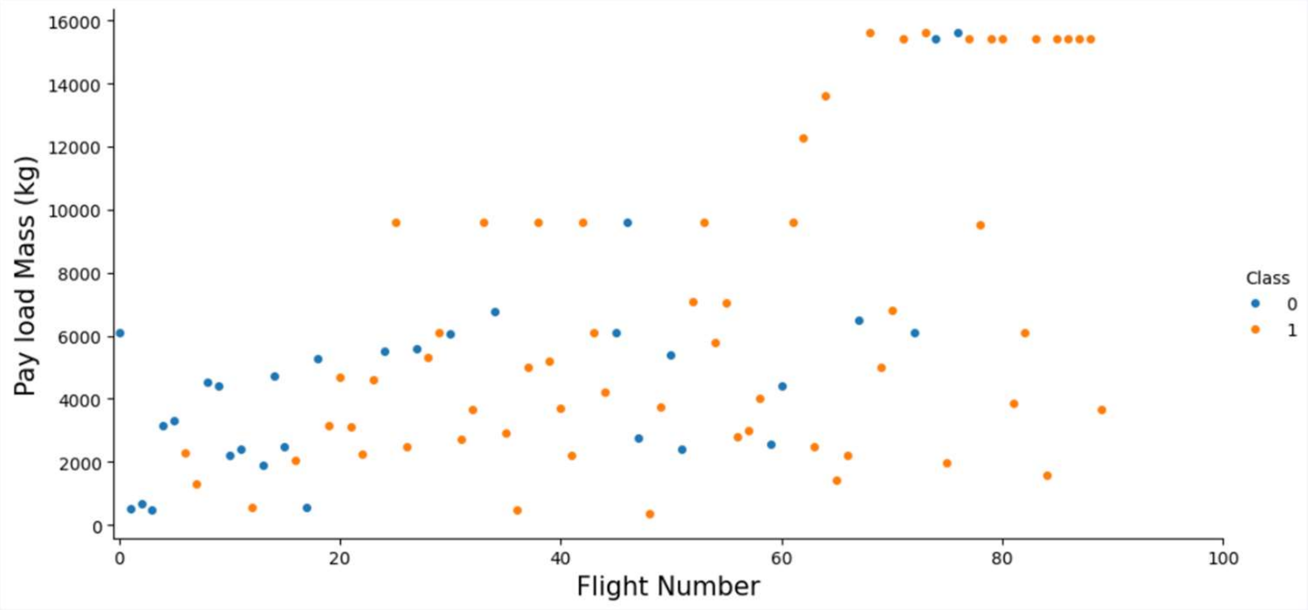
Insights drawn from EDA

Flight Number vs. Launch Site



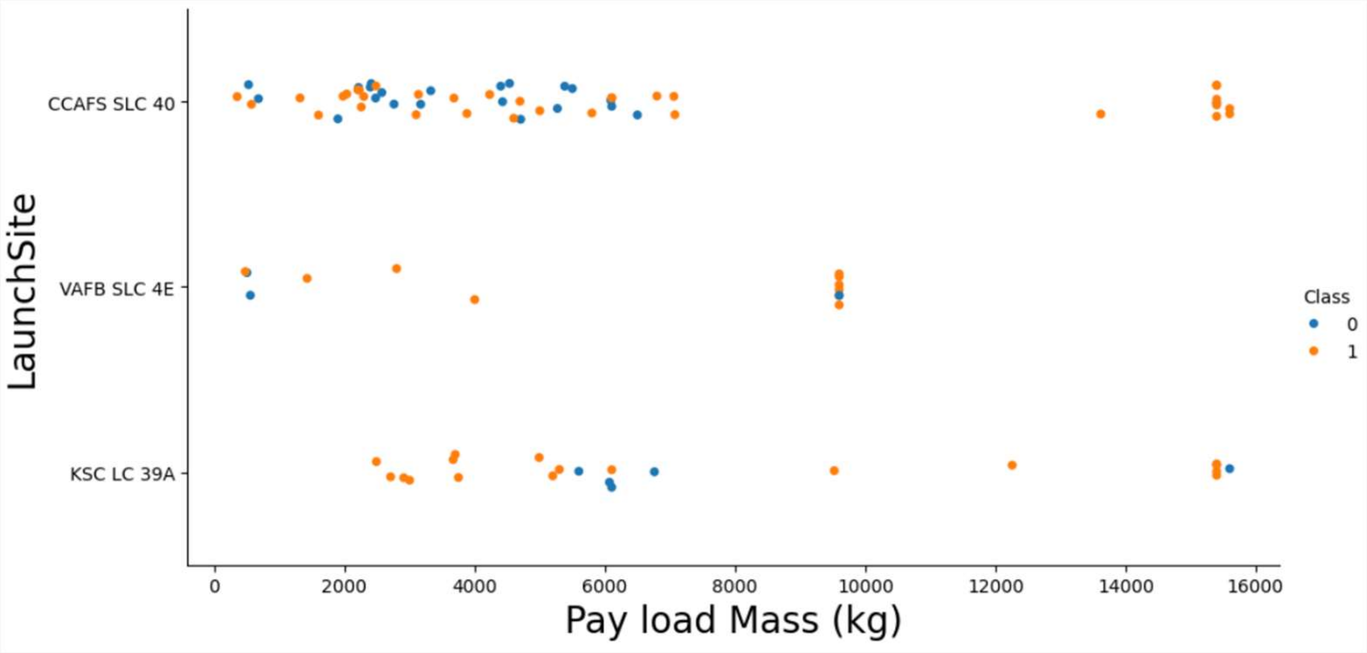
- Early flights mainly launched at the CCAFS SLC 40 site. VAFB SLC 4E also did a couple. Recent launches mainly happened at CCAFS SLC 40 and KSC LC 39A sites.
- As the flight number increases, the first stage is more likely to land successfully..

Flight Number vs. Payload Mass (kg)



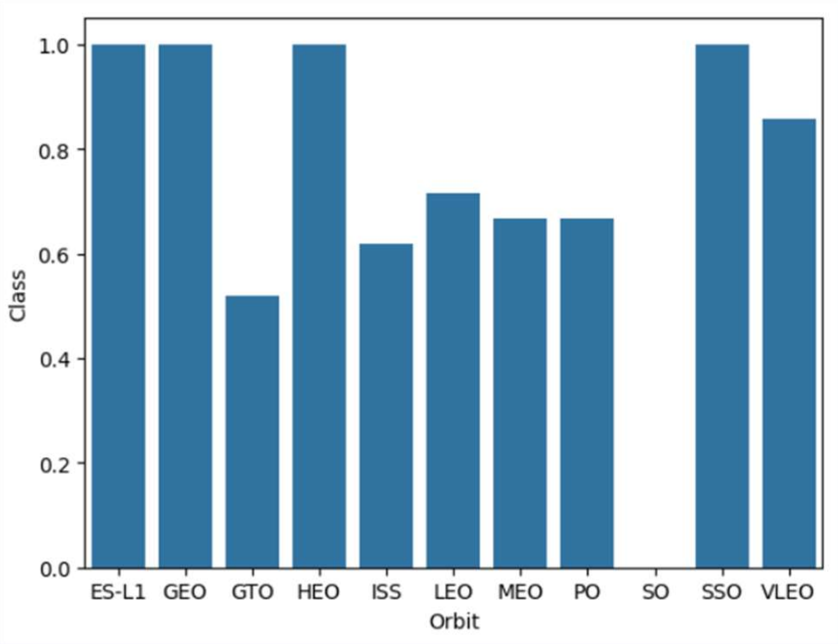
- Overall, there’s a trend that as the flight number increase, the payload mass increases, also more successful landing.

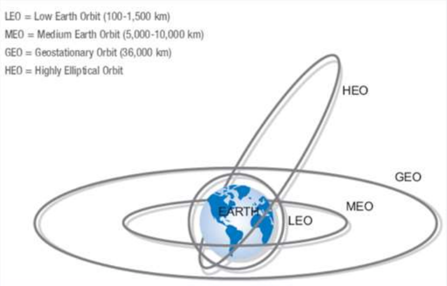
Payload vs. Launch Site



- VAFB SLC 4E had payload less than 10,000 kg. The other two sites had several launches with payload more than 10,000 kg, even though most of the payloads were less than 8,000 kg overall.
- Good success rate with payload more than 8.000 kg.

Success Rate vs. Orbit Type

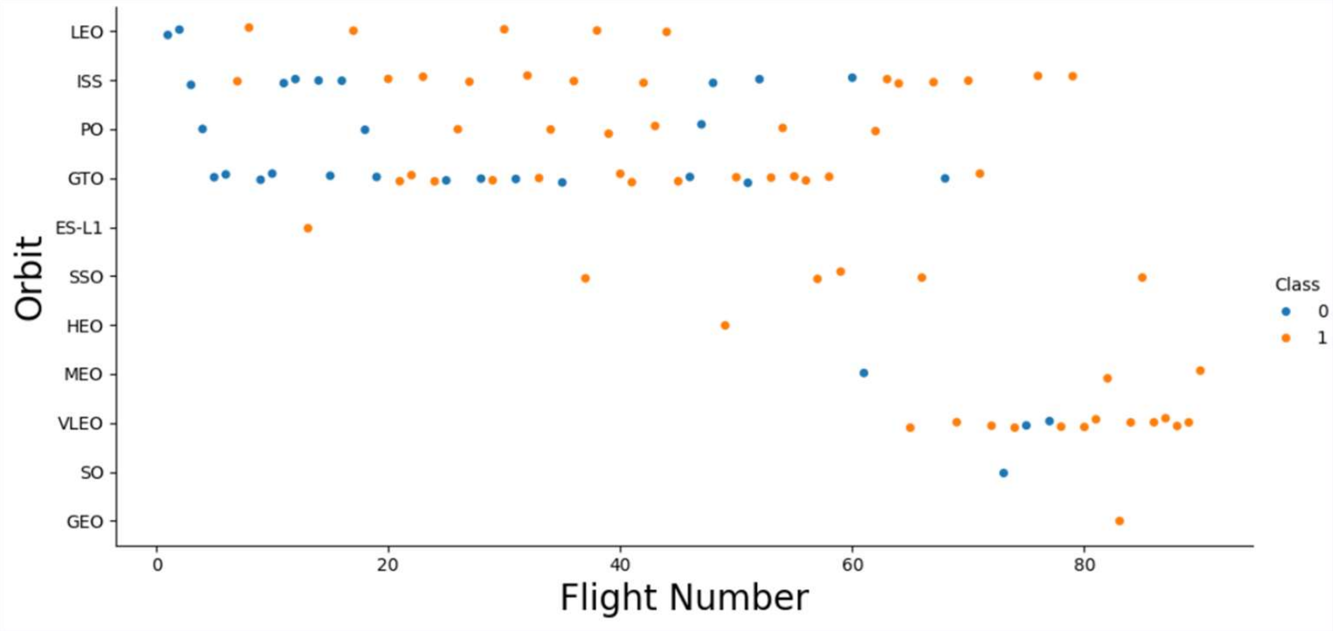



LEO = Low Earth Orbit (100-1,500 km)
MEO = Medium Earth Orbit (5,000-10,000 km)
GEO = Geostationary Orbit (36,000 km)
HEO = Highly Elliptical Orbit
HEO
GEO
MEO
EARTH
LEO

Types of Orbits - Space Foundation | www.spacefoundation.org

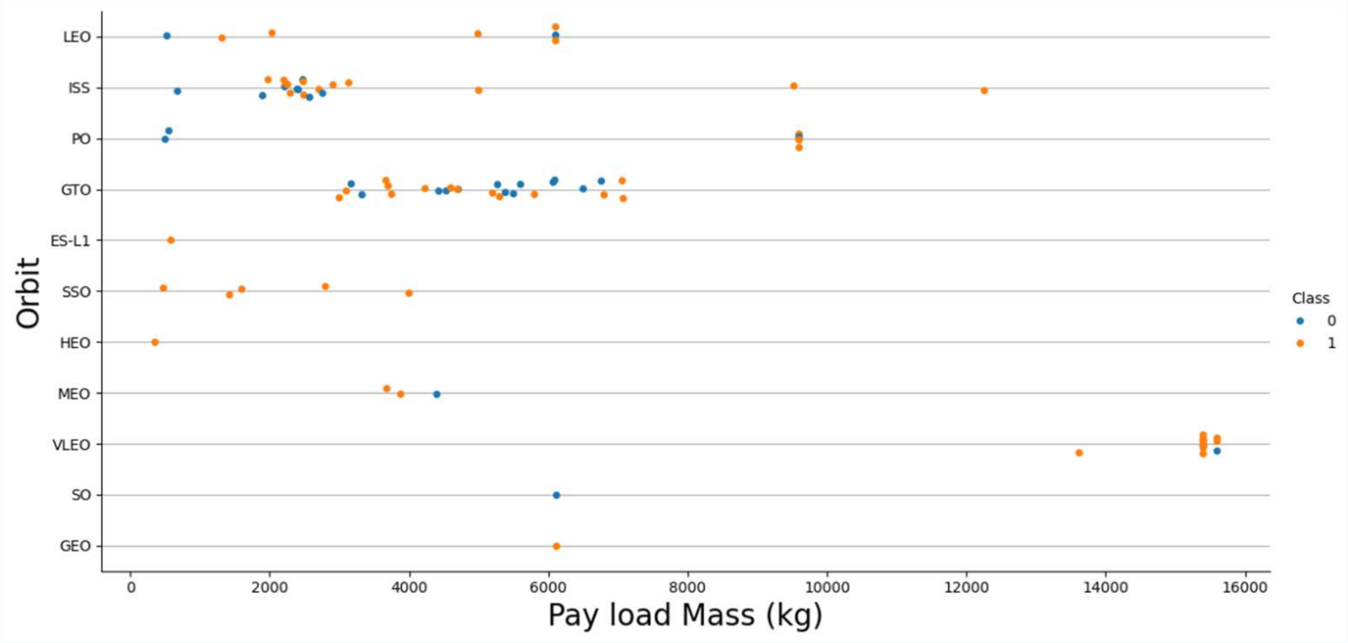
- High success rate with orbit ES-L1, GEO, HEO, SSO
- The success rate is above 50% overall.

Flight Number vs. Orbit Type



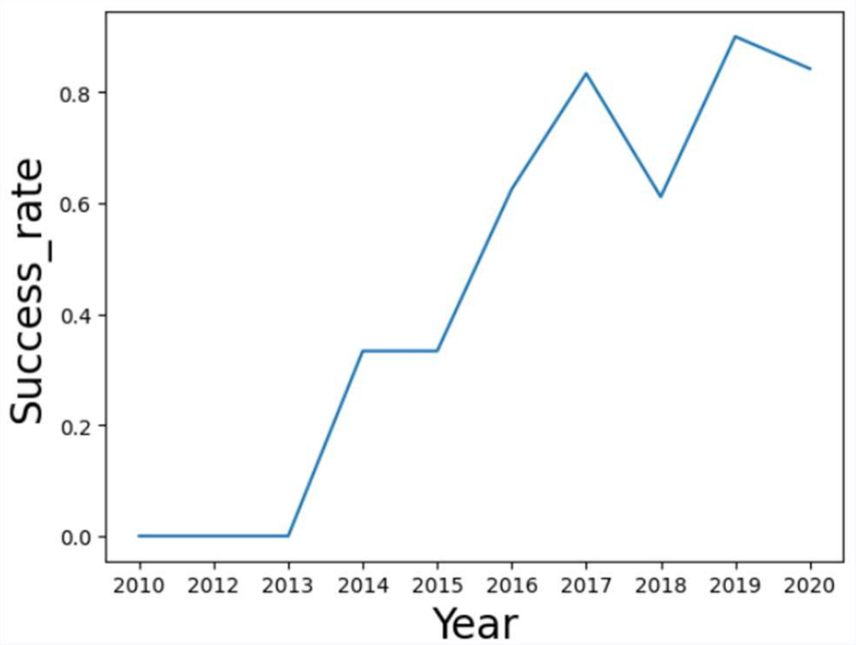
- More success in LEO and VLEO orbits with more recent flights.
- Looks like very low altitude orbit (VLEO) is getting more and more used lately.

Payload vs. Orbit Type



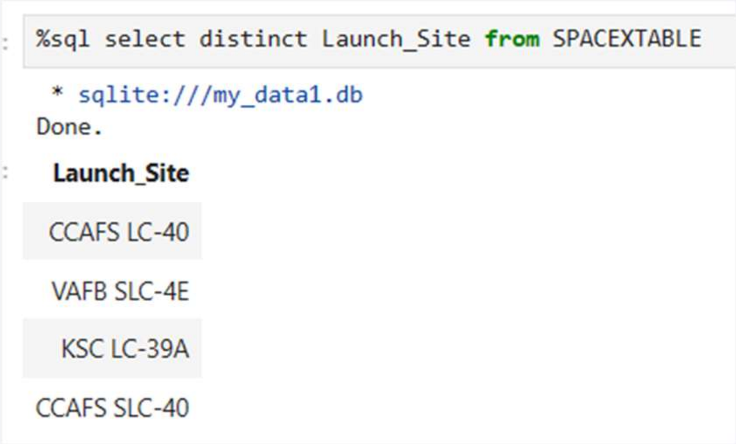
- Looks like VLEO has bigger payload capacity, and high success landing rates. This is probably because it’s closer to the earth surface than the other orbits.
- HEO, SSO and ES-L1 had 100 % success rate but they all had small payload (<4000 kg)

Launch Success Yearly Trend



- Success rate since 2013 kept increasing with a slight decrease in 2018. The highest success rate happened in 2019.

All Launch Site Names



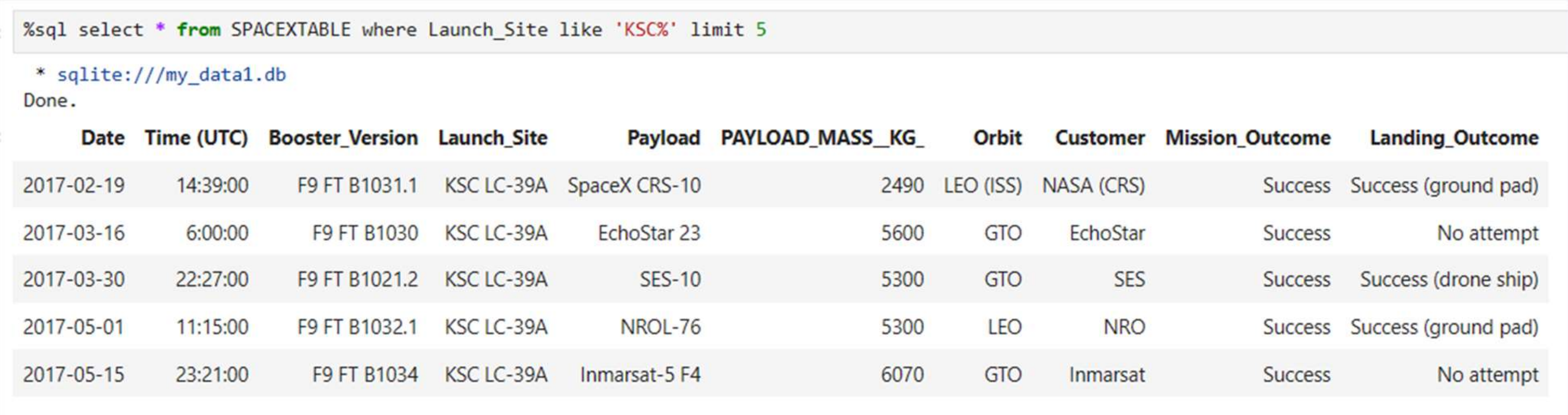
```
%sql select distinct Launch_Site from SPACEXTABLE
```

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

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

- There are four launch sites in the SPACEXTABLE dataset.

Launch Site Names Begin with 'KSC'



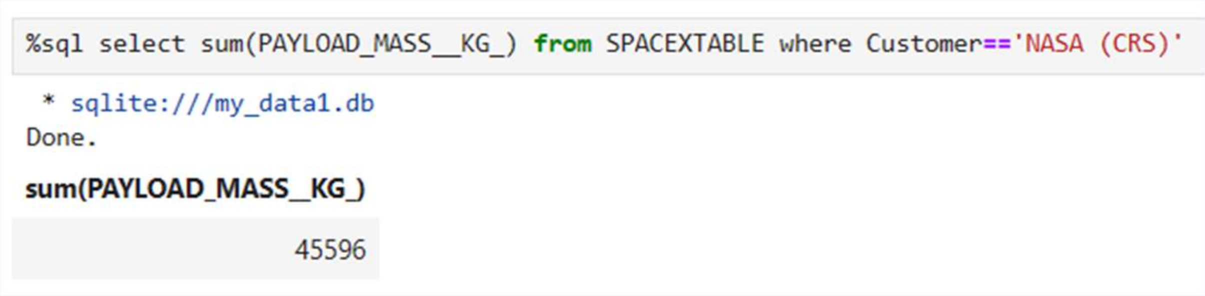
```
%sql select * from SPACEXTABLE where Launch_Site like 'KSC%' limit 5
```

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

Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Orbit	Customer	Mission_Outcome	Landing_Outcome
2017-02-19	14:39:00	F9 FT B1031.1	KSC LC-39A	SpaceX CRS-10	2490	LEO (ISS)	NASA (CRS)	Success	Success (ground pad)
2017-03-16	6:00:00	F9 FT B1030	KSC LC-39A	EchoStar 23	5600	GTO	EchoStar	Success	No attempt
2017-03-30	22:27:00	F9 FT B1021.2	KSC LC-39A	SES-10	5300	GTO	SES	Success	Success (drone ship)
2017-05-01	11:15:00	F9 FT B1032.1	KSC LC-39A	NROL-76	5300	LEO	NRO	Success	Success (ground pad)
2017-05-15	23:21:00	F9 FT B1034	KSC LC-39A	Inmarsat-5 F4	6070	GTO	Inmarsat	Success	No attempt

- Returned 5 records where launch sites’ names start with `KSC`

Total Payload Mass

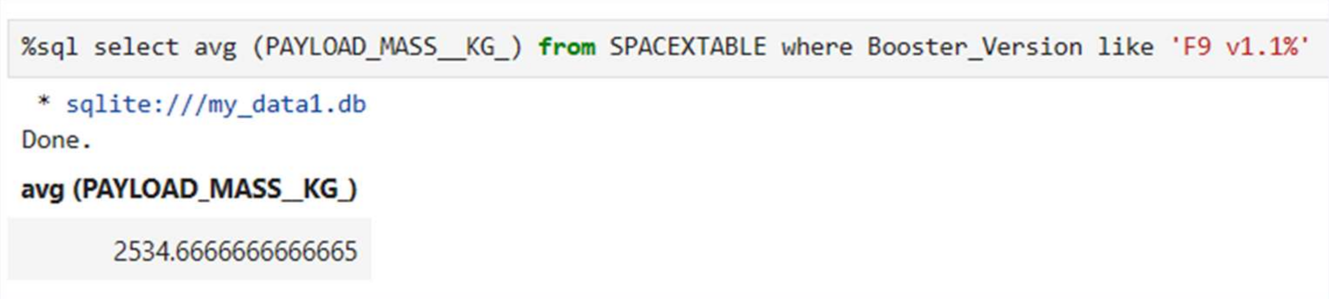


```
%sql select sum(PAYLOAD_MASS__KG_) from SPACEXTABLE where Customer=='NASA (CRS)'
```

```
 * sqlite:///my_data1.db
Done.
sum(PAYLOAD_MASS__KG_)
                 45596
```

- Total payload carried by boosters from NASA is 45596 Kg.

Average Payload Mass by F9 v1.1



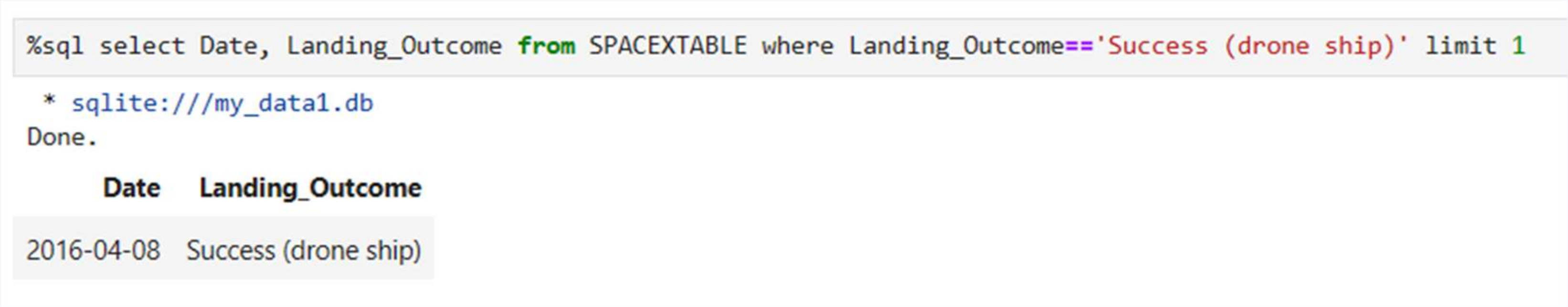
```
%sql select avg (PAYLOAD_MASS__KG_) from SPACEXTABLE where Booster_Version like 'F9 v1.1%'
```

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

avg (PAYLOAD_MASS__KG_)
2534.6666666666665

- Average payload mass carried by booster version F9 v1.1 is 2534.667 Kg

First Successful Ground Landing Date



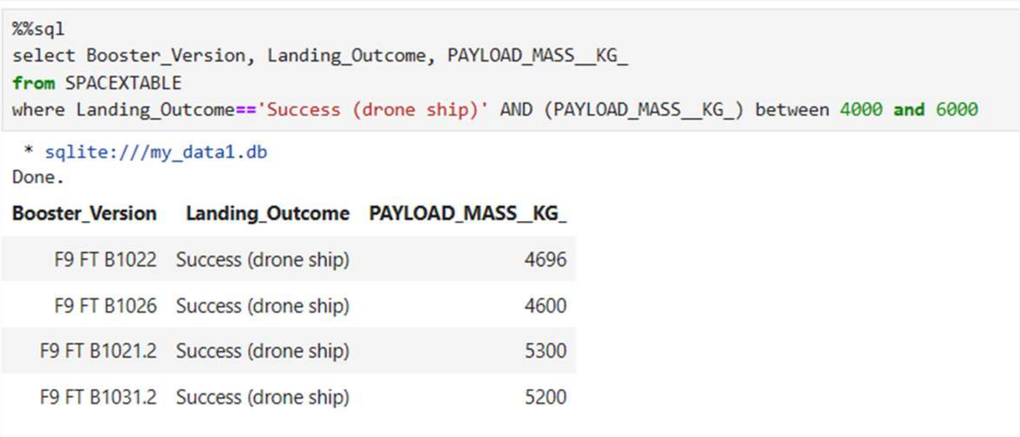
```
%sql select Date, Landing_Outcome from SPACEXTABLE where Landing_Outcome=='Success (drone ship)' limit 1
```

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

Date	Landing_Outcome
2016-04-08	Success (drone ship)

- The first successful drone ship landing was on 2016-04-08.

Successful Drone Ship Landing with Payload between 4000 and 6000



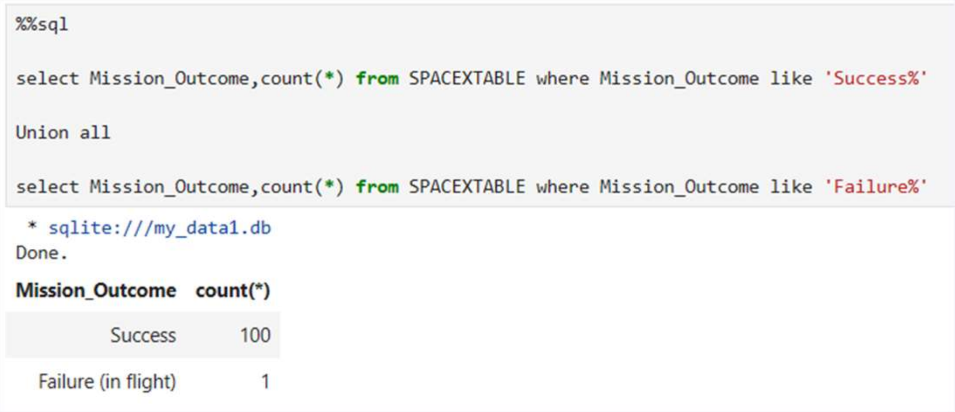
```
%%sql
select Booster_Version, Landing_Outcome, PAYLOAD_MASS__KG_
from SPACEXTABLE
where Landing_Outcome=='Success (drone ship)' AND (PAYLOAD_MASS__KG_) between 4000 and 6000
```

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

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

- Four Falcon 9 boosters have successfully landed on drone ship and had payload mass greater than 4000 but less than 6000

Total Number of Successful and Failure Mission Outcomes



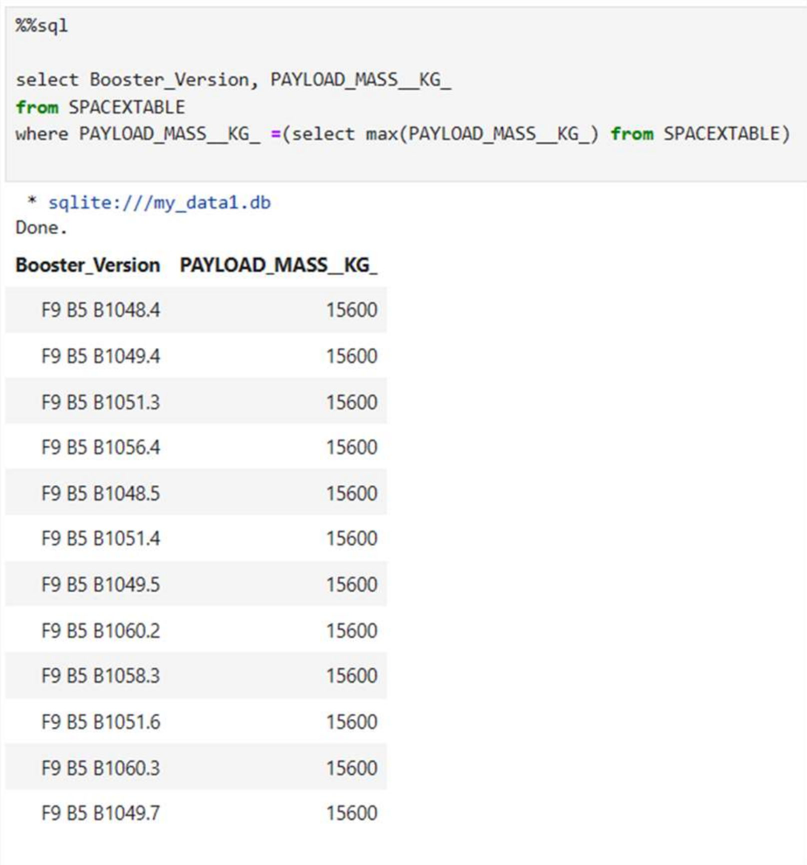
```
%%sql

select Mission_Outcome,count(*) from SPACEXTABLE where Mission_Outcome like 'Success%'

Union all

select Mission_Outcome,count(*) from SPACEXTABLE where Mission_Outcome like 'Failure%'
```

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

Mission_Outcome	count(*)
Success	100
Failure (in flight)	1

- In this dataset, there’re 100 successful mission outcomes and 1 failure.

Boosters Carried Maximum Payload



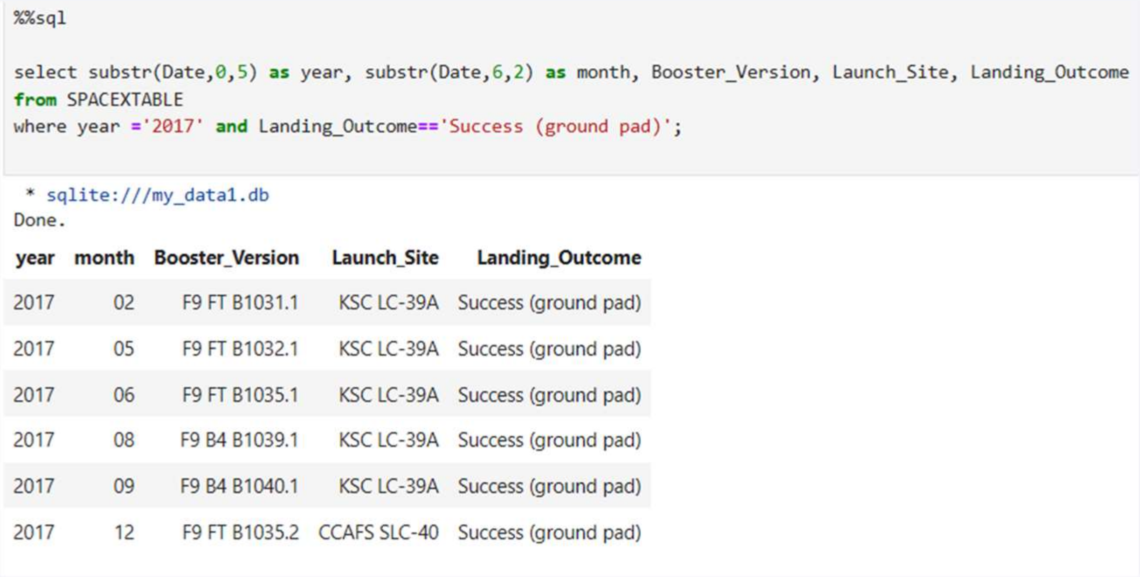
```
%%sql

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

- A couple of Falcon 9 boosters have carried the maximum payload mass

2015 Launch Records



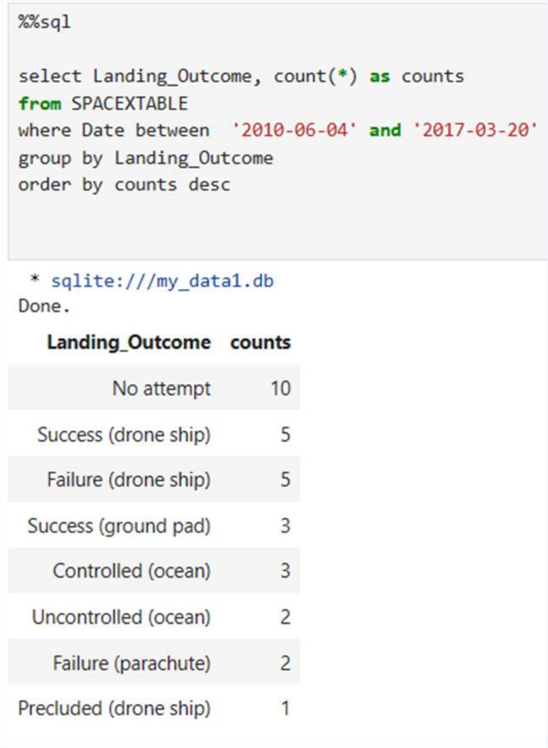
```
%%sql

select substr(Date,0,5) as year, substr(Date,6,2) as month, Booster_Version, Launch_Site, Landing_Outcome
from SPACEXTABLE
where year ='2017' and Landing_Outcome=='Success (ground pad)';
```

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

year	month	Booster_Version	Launch_Site	Landing_Outcome
2017	02	F9 FT B1031.1	KSC LC-39A	Success (ground pad)
2017	05	F9 FT B1032.1	KSC LC-39A	Success (ground pad)
2017	06	F9 FT B1035.1	KSC LC-39A	Success (ground pad)
2017	08	F9 B4 B1039.1	KSC LC-39A	Success (ground pad)
2017	09	F9 B4 B1040.1	KSC LC-39A	Success (ground pad)
2017	12	F9 FT B1035.2	CCAFS SLC-40	Success (ground pad)

- There were 6 successful ground pad landing that were launched from KSC LC-39A and CCAFS SLC-40 sites and were carried by different Falcon 9 boosters in 2017.

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



```
%%sql

select Landing_Outcome, count(*) as counts
from SPACEXTABLE
where Date between  '2010-06-04' and '2017-03-20'
group by Landing_Outcome
order by counts desc
```

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

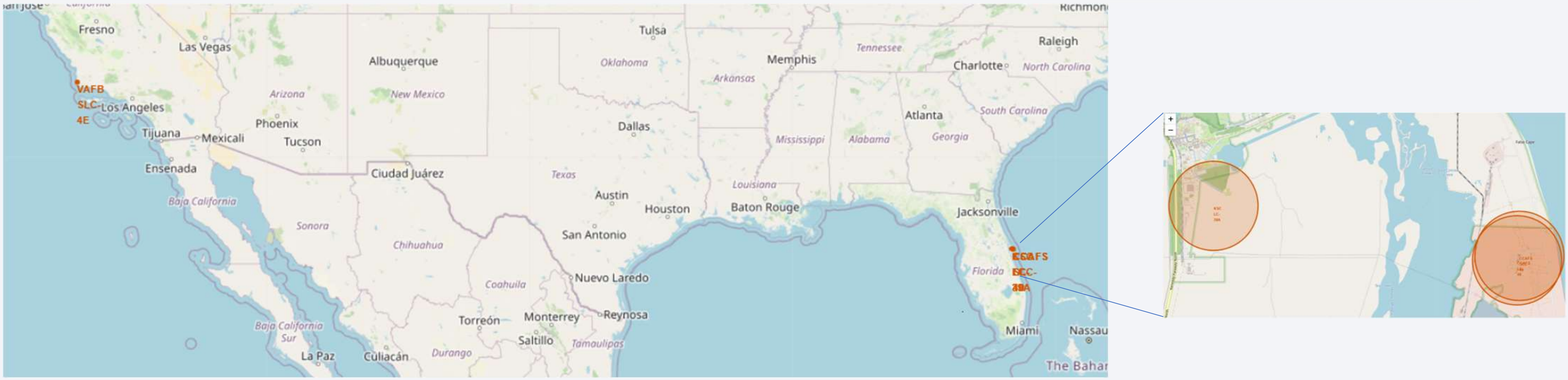
Landing_Outcome	counts
No attempt	10
Success (drone ship)	5
Failure (drone ship)	5
Success (ground pad)	3
Controlled (ocean)	3
Uncontrolled (ocean)	2
Failure (parachute)	2
Precluded (drone ship)	1

- Counts of landing outcomes (such as Failure (drone ship) or Success (ground pad)) between the date 2010-06-04 and 2017-03-20 have been ranked and listed in descending order

Section 3

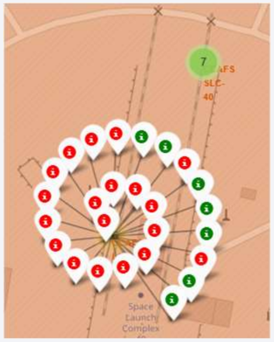
Launch Sites Proximities Analysis

Location of the most commonly used rocket launch sites in the United States

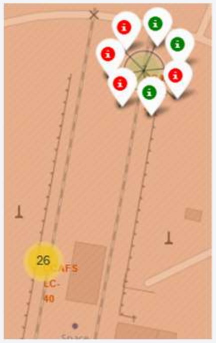


1. **Kennedy Space Center (KSC)**: NASA's primary launch center of human spaceflight, where Apollo, Skylab, and Space Shuttle have been launched.
2. **Cape Canaveral Space Launch Complex (CCSFS or CCAFS)**: Adjacent to the KSC. Major site for commercial, military, and scientific missions. It has been used for SpaceX, United Launch Alliance (ULA), and other commercial launches.
3. **Vandenberg Space Force Base (VSFB)**: Vandenberg is crucial for launches into polar orbits. Its geographical location on the West Coast makes it ideal for launching satellites into north-south orbits around the Earth, a trajectory that is not achievable from the eastern launch sites.

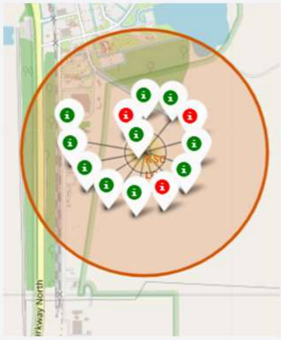
Launch Outcome at Each Site



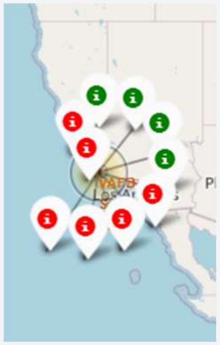
CCAFS LC-40



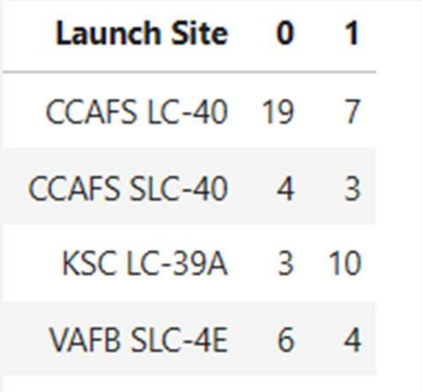
CCAFS SLC-40



KSC LC-39A



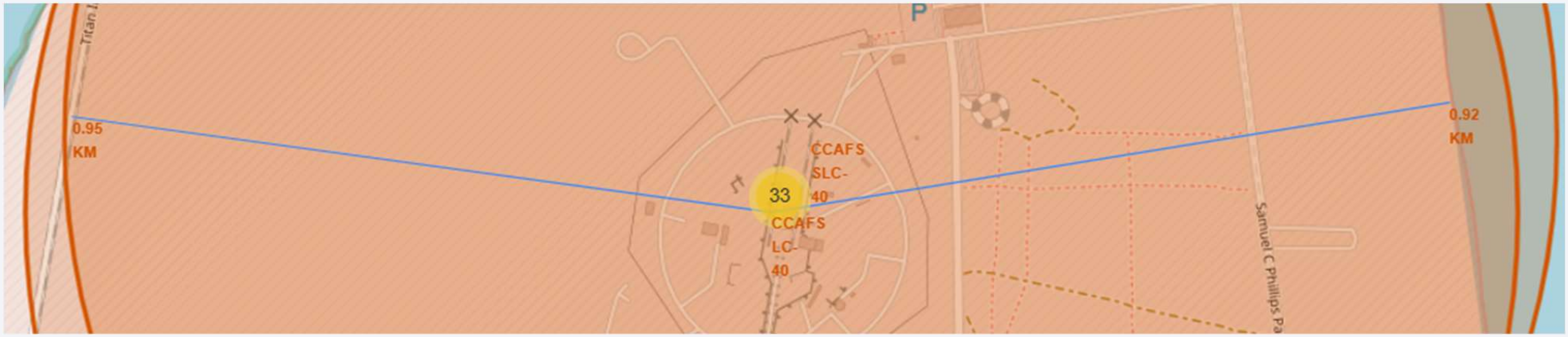
VAFB-SLC-4E



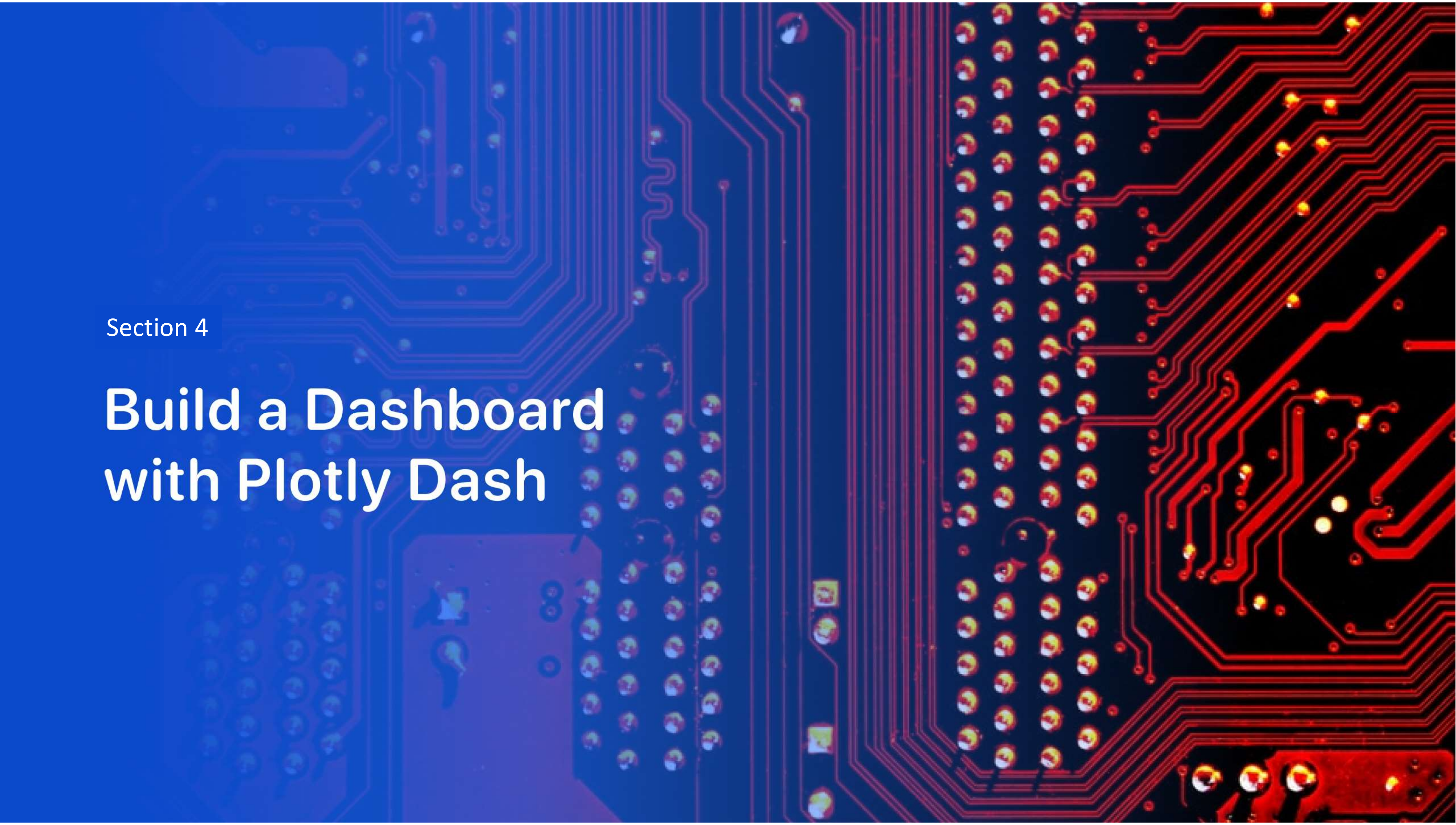
Launch Site	0	1
CCAFS LC-40	19	7
CCAFS SLC-40	4	3
KSC LC-39A	3	10
VAFB SLC-4E	6	4

- KSC LC-39A has been the most successful one, as almost all of the markers are green, which means success.
- CCAFS LC-40 has the lowest success rate, probably because early launches happened there when there wasn't much experience yet.
- The other two sites have about 40% success rate.

CCAFS LC-40 is about 0.9 KM away from the nearest coastline and railroad



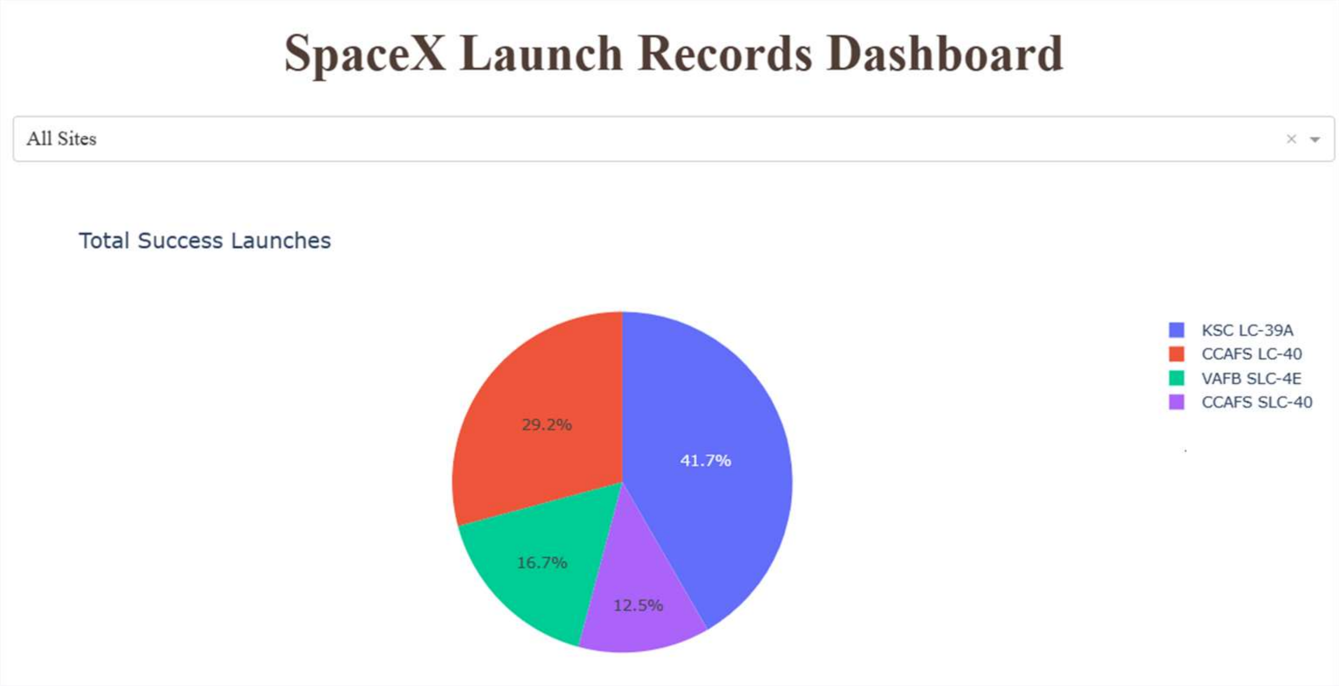
- CCAFS LC-40 is within 1 KM away from the nearest coastline and railroad



Section 4

Build a Dashboard with Plotly Dash

Overview of Successful Launches at all Sites



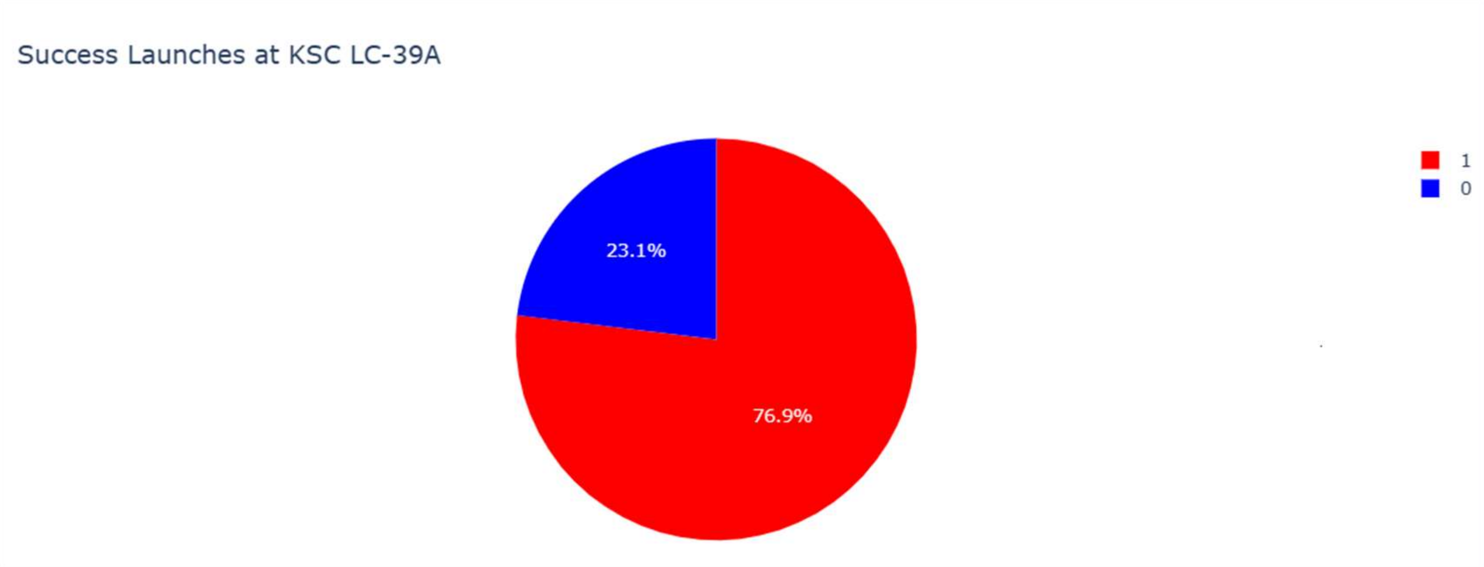
SpaceX Launch Records Dashboard

All Sites

Total Success Launches

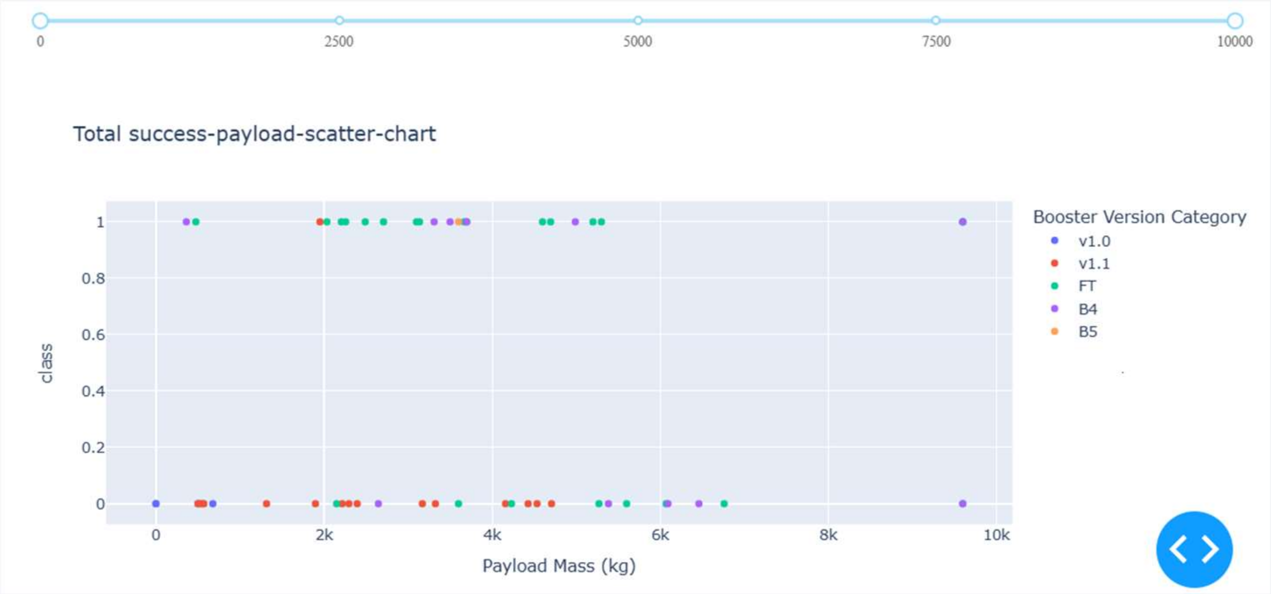
- KSC LC-39A had the most success, while CCAFS SLC-40 had the least.

KSC LC-39A had the Highest Success Rate



- The success rate at KSC LC-39A is over 75%, which is the highest among all four sites.

Effect of Payload Mass and Booster Category on the Launch Outcome



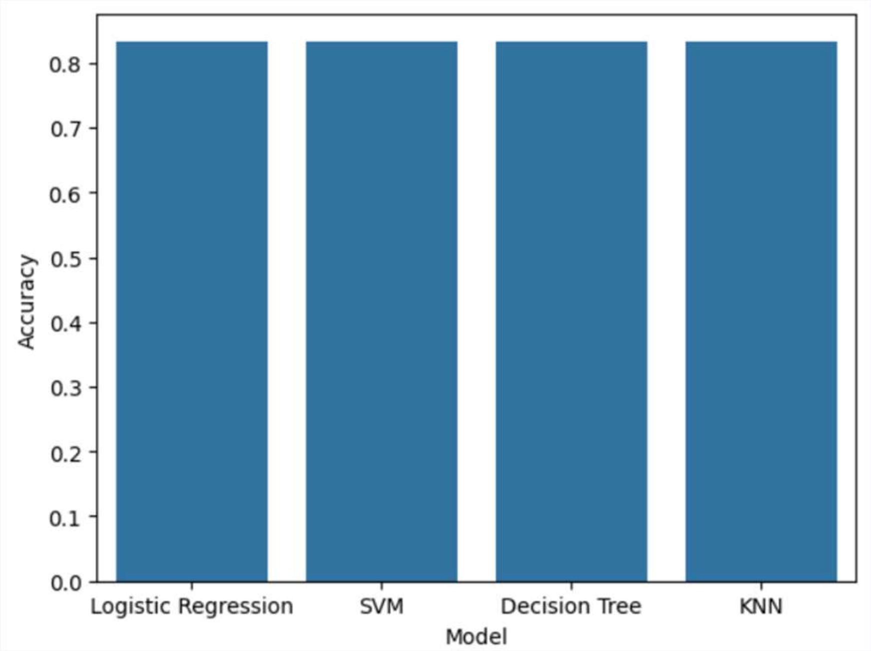
- FT category of boosters had the highest success rate.
- Looks like most success happened with payload less than 6000 kg, in this dataset.

Section 5

Predictive Analysis (Classification)

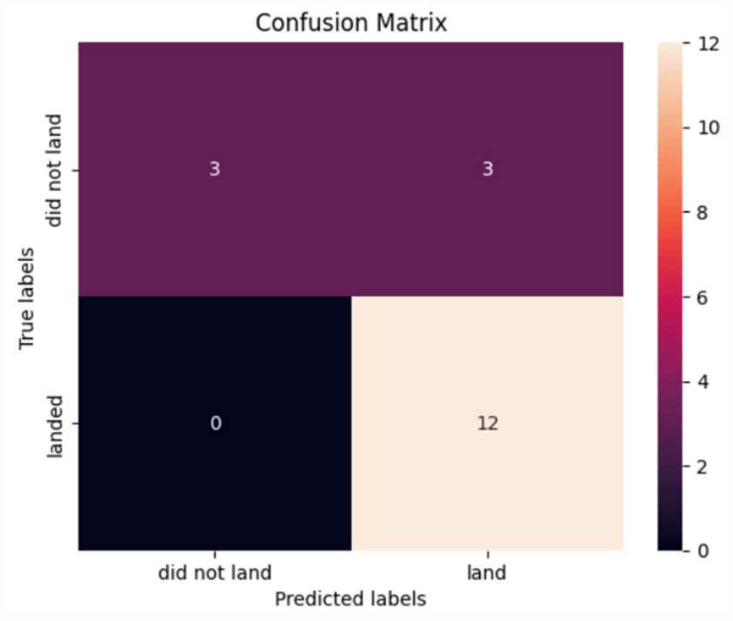
using the method score.

Classification Accuracy



- Same accuracy for all built classification models using the method .score(). This is likely because of the small dataset size.

Confusion Matrix



- All models generated the same confusion matrix. They can predict the true positive, true negative and false negative well. But they over predicted the success landing and got some false positives. This is likely due to the small dataset size.

Conclusions

- Landing success rate is increasing overtime.
- Good success rate with payload more than 8.000 kg.
- The success rate at KSC LC-39A is over 75%, which is the highest among four sites.
- ES-L1, GEO, HEO, SSO orbits have 100 % success rate.
- FT boosters had the highest success rate among all the boosters.
- Launch sites are located in proximity to coastline.
- Four machine learning models we developed have the same accuracy. They all over-predict successful landings. This can be improved when more data is available.

Appendix

- GitHub link
- Thanks to the instructors for carefully preparing this project!
- Thanks to my virtual classmates for the discussions they posted!
- Thanks to my peer who will be reviewing my work!

Thank you!