IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Hanqing Zhao
August 15, 2022

Outline

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

Executive Summary

- Summary of methodologies
  - Collecting data through a Request method
  - Extracting data through web scrapping
  - Data wrangling through removing irrelevant data, handling null and binary values
  - Conducting exploratory data analysis by data visualization and SQL queries
  - Building classification models for prediction
- Summary of all results
  - Evaluating factors that affect the success rate
  - Retrieving valuable information from the database
  - Visualizing data with an interactive map and dashboard
  - Analyzing the best classifier

Introduction

- SpaceX is one of the most promising rocket companies
- SpaceX has two advantages:
  - Reusable rockets
  - Relatively inexpensive launch costs
- This project will analyze SpaceX data to determine if SpaceX will reuse the first stage
- This project will focus on the following questions:
  - What factors influence the success rate of rocket launches and landings?
  - To what degree are these factors related to the success rate?
  - What boosters are more likely to make a landing successful?
  - Which model performs the best in making predictions?
- Visualizing SpaceX data in different forms
- Analyzing other features in regard to launches and landings

Section 1

Methodology

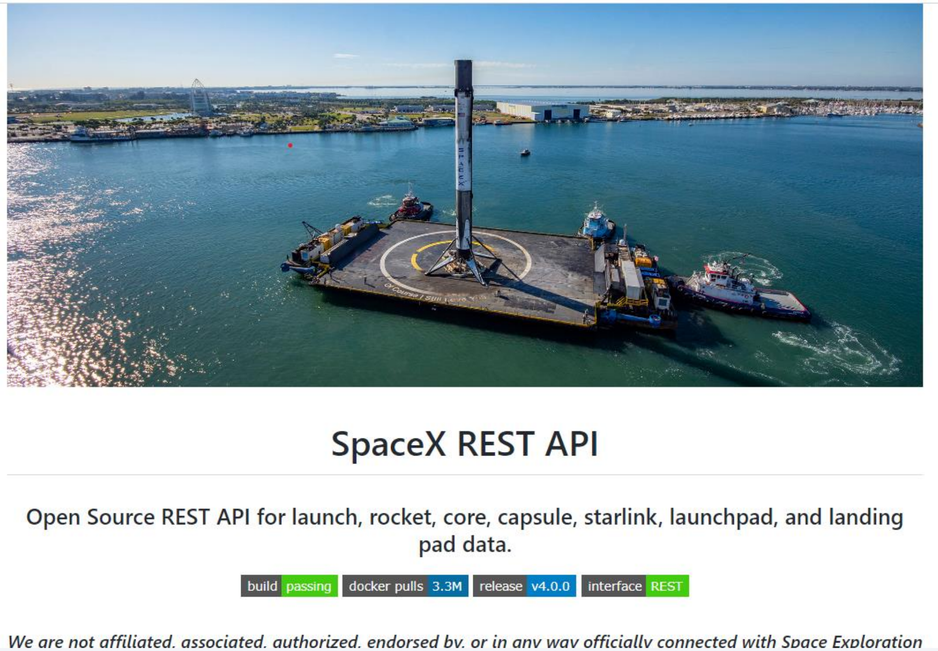
Methodology

Executive Summary

- Data collection methodology:
    - HTTP GET method, web scrapping and parsing & converting data
- Perform data wrangling
    - Removing irrelevant data, dealing with missing values & creating a binary column
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
    - Data preprocessing, train-test split, GridSearch, 4 classification algorithms & model evaluation

Data Collection

- Collect data from HTML pages (SpaceX REST API and Wikipedia)
- Then perform web scrapping to organize the data



2021

[hide] Flight No.	Date and time (UTC)	Version, Booster[b]	Launch site	Payload[c]	Payload mass	Orbit	Customer	Launch outcome	Booster landing
104	8 January 2021 02:15[604]	F9 B5 △ B1060.4	CCSFS, SLC-40	Türksat 5A[605]	3,500 kg (7,700 lb)	GTO	Türksat	Success	Success (drone ship)
	A 3,500 kg (7,700 lb) satellite intended to be stationed at 31.0° east.[605] This is the most powerful satellite in Türksat's fleet[606] and will provide Ku-band television broadcast services over Turkey, the Middle East, Europe and Africa. The satellite was injected in to a Super-synchronous transfer orbit of 280 km × 55,000 km (170 mi × 34,180 mi) with 17.6° inclination.[607]								
105	20 January 2021 13:02[608]	F9 B5 △ B1051.8[609]	KSC, LC-39A	Starlink 16 v1.0 (60 satellites)	15,600 kg (34,400 lb)	LEO	SpaceX	Success	Success (drone ship)
	The first booster to successfully launch and land eight times. Achieved a record turnaround time between two launches of the same booster of only 38 days and brought the total of launched Starlink satellites to over 1000.[610] SpaceX stated that the landing would occur during higher winds than usual; this test to expand the landing envelope was successfully passed by the booster.[611]								
106	24 January 2021 15:00[612]	F9 B5 △ B1058.5[613]	CCSFS, SLC-40	*Transporter-1* (143 smallsat rideshare)	~5,000 kg (11,000 lb)	SSO	Various	Success	Success (drone ship)
	First dedicated smallsat rideshare launch, targeting a 525 km (326 mi) altitude orbit.[614] The launch deployed a record 143 satellites, consisting of 120 CubeSats, 11 microsatellites, 10 Starlinks, and 2 transfer stages. In addition, 2 hosted payloads and 1 non-separating dummy satellite[615] were[*failed verification*] launched.[616] These include SpaceBEE (x 36), Lemur-2 (x 8), ICEYE (x 3), UVSQ-SAT,[617] ELaNa 35 (PTD-1),[381] and multiple Kepler nanosats.[618][619] D-Orbit flew their ION SCV LAURENTIUS, 10 Starlink satellites were placed in a polar orbit[620] and 2 of 15 payloads remained attached to SHERPA-FX1. Exolaunch deployed several small satellites and cubesats via their own deployment mechanisms. First flight of a Falcon 9 with a SHERPA-FX transfer stage called SHERPA-FX1.[621][622]								
107	4 February 2021 06:19[623]	F9 B5 △ B1060.5[624]	CCSFS, SLC-40	Starlink 18 v1.0 (60 satellites)	15,600 kg (34,400 lb)	LEO	SpaceX	Success	Success (drone ship)
	This marked the fastest turnaround to date, at 27 days, and the first time a Falcon 9 flies twice within a month.[625]								
108	16 February 2021 03:59:37[626]	F9 B5 △ B1059.6	CCSFS, SLC-40	Starlink 19 v1.0 (60 satellites)[627]	15,600 kg (34,400 lb)	LEO	SpaceX	Success	Failure (drone ship)
	A hole in a heat-shielding engine cover, which likely developed through fatigue, allowed recirculating hot exhaust gases to damage one of the Merlin 1D first-stage engines, causing it to shut down early during ascent. Engine-out capability of the Falcon 9 allowed the mission to continue and successfully deploy the 60 Starlink satellites to orbit.[628] The issue caused the booster to fail its landing attempt and miss the droneship *Of Course I Still Love You* (OCISLY) after its entry burn, breaking the longest streak of 24 landing successes.[629] During this mission, *GO Ms. Tree* and *GO Ms. Chief* were used for the last time to recover the fairings.[630][631] After this mission, both ships were retired because SpaceX no longer plans to catch the fairings.[632]								
109	4 March 2021 08:24[633]	F9 B5 △ B1049.8[634]	KSC, LC-39A	Starlink 17 v1.0 (60 satellites)	15,600 kg (34,400 lb)	LEO	SpaceX	Success	Success (drone ship)
	Launch had previously been postponed multiple times, causing the payload Starlink L17 to launch after the L18 and L19 missions. Featured for the first time, a fairing which was flying on its fourth flight.[635] The second-stage deorbit burn failed, causing an uncontrolled reentry on 26 March 2021 over the west coast of the United States.[636]								

Data Collection – SpaceX API

- Request the data from SpaceX REST API by performing a HTTP GET method

```
In [6]: spacex_url="https://api.spacexdata.com/v4/launches/past"

In [7]: response = requests.get(spacex_url)
```

Check the content of the response

```
In [8]: print(response.content)
```


```
b'[{"fairings":{"reused":false,"recovery_attempt":false,"recovered":false,"ships":[]},
e":"https://images2.imgbox.com/40/e3/GypSkayF_o.png"},"reddit":{"campaign":null,"launc
t":null,"webcast":"https://www.youtube.com/watch?v=0a_00nJ_Y88","youtube_id":"0a_00nJ_
aunch.html","wikipedia":"https://en.wikipedia.org/wiki/DemoSat"},"static_fire_date_utc
w":0,"rocket":"5e9d0d95eda69955f709d1eb","success":false,"failures":[{"time":33,"altit
ds and loss of vehicle","crew":[],"ships":[],"capsules":[],"payloads":["5eb0e4b5b6c3bb
onSat","date_utc":"2006-03-24T22:30:00.000Z","date_unix":1143239400,"date_local":"2006
e":"5e9e289df35918033d3b2623","flight":1,"gridfins":false,"legs":false,"reused":false,
1}],"auto_update":true,"tbd":false,"launch_library_id":null,"id":"5eb87cd9ffd86e000604
ps":[]},"links":{"patch":{"small":"https://images2.imgbox.com/4f/e3/I0lkuJ2e_o.png","1
1,"launch":null,"media":null,"recovery":null},"flickr":{"small":[],"original":[]},"pre
d":"Lk4zQ2wP-Nc","article":"https://www.space.com/3590-spacex-falcon-1-rocket-fails-re
```


- GitHub URL:
- Lab 1: Collecting the data
- Lab 2: Web Scraping

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

```
In [5]: # use requests.get() method with the provided static_url
        # assign the response to a object
        response = requests.get(static_url).text
```

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

```
In [6]: # Use BeautifulSoup() to create a BeautifulSoup object from a response text content
        soup = BeautifulSoup(response,"html.parser")
```

Print the page title to verify if the `BeautifulSoup` object was created properly

```
In [7]: # Use soup.title attribute
        soup.title

Out[7]: <title>List of Falcon 9 and Falcon Heavy launches - Wikipedia</title>
```

Data Collection - Scraping

- Store the data in the form of a JSON after the data is extracted from Wiki pages
- Use BeautifulSoup to parse the data
- Convert the data into a Pandas data frame
- GitHub URL:
- [Lab 1: Collecting the data]
- [Lab 2: Web Scraping]

In [11]:
```
# Use json_normalize meethod to convert the json result into a dataframe
data = response.json()
data = pd.json_normalize(data)
```

Using the dataframe `data` print the first 5 rows

In [12]:
```
# Get the head of the dataframe
data.head()
```

Out[12]:

	static_fire_date_utc	static_fire_date_unix	net	window	rocket	success	failures
0	2006-03-17T00:00:00.000Z	1.142554e+09	False	0.0	5e9d0d95eda69955f709d1eb	False	[{'time': 33, 'altitude': None, 'reason': 'merlin engine failure'}]

In [15]:
```
df.head()
```

Out[15]:

	Flight No.	Launch site	Payload	Payload mass	Orbit	Customer	Launch outcome	Version Booster	Booster landing	Date	Time
0	1	CCAFS	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success\n	F9 v1.0B0003.1	Failure	4 June 2010	18:45
1	2	CCAFS	Dragon	0	LEO	NASA (COTS)\nNRO	Success	F9 v1.0B0004.1	Failure	8 December 2010	15:43
2	3	CCAFS	Dragon	525 kg	LEO	NASA (COTS)	Success	F9 v1.0B0005.1	No attempt\n	22 May 2012	07:44
3	4	CCAFS	SpaceX CRS-1	4,700 kg	LEO	NASA (CRS)	Success\n	F9 v1.0B0006.1	No attempt	8 October 2012	00:35
4	5	CCAFS	SpaceX CRS-2	4,877 kg	LEO	NASA (CRS)	Success\n	F9 v1.0B0007.1	No attempt\n	1 March 2013	15:10

Data Wrangling

- Remove irrelevant data (Falcon 1) from the data frame
- Replace the missing values with the mean
- Create a new column ‘Class’ to show the outcome of each launch
- GitHub URL:
- [Lab 1: Collecting the data]()
- [Lab 3: Data Wrangling]()

```
[29]: data_falcon9['PayloadMass'].mean()

[29]: 6123.547647058824

[30]: # Calculate the mean value of PayloadMass column
      mean = data_falcon9['PayloadMass'].mean()
      # Replace the np.nan values with its mean value
      data_falcon9['PayloadMass'] = data_falcon9['PayloadMass'].replace(np.nan, mean)

      /home/jupyterlab/conda/envs/python/lib/python3.7/site-packages/ipykernel_launche
      A value is trying to be set on a copy of a slice from a DataFrame.
      Try using .loc[row_indexer,col_indexer] = value instead

      See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stab
      y
        after removing the cwd from sys.path.

[31]: data_falcon9.isnull().sum()

[31]: FlightNumber      0
      Date              0
      BoosterVersion    0
      PayloadMass       0
      Orbit             0
      LaunchSite        0
      Outcome           0
      Flights           0
      GridFins          0
```

```
In [26]: # Hint data['BoosterVersion']!='Falcon 1'
         data_falcon9 = data[data['BoosterVersion']!='Falcon 1']
```

Now that we have removed some values we should reset the FlgihtNumber column

```
In [27]: data_falcon9.loc[:,'FlightNumber'] = list(range(1, data_falcon9.shape[0]+1))
         data_falcon9

/home/jupyterlab/conda/envs/python/lib/python3.7/site-packages/pandas/core/indexing.py:1773: SettingW
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexin
  self._setitem_single_column(ilocs[0], value, pi)
```

Out[27]:

	FlightNumber	Date	BoosterVersion	PayloadMass	Orbit	LaunchSite	Outcome	Flights	GridFins	Reused	Legs
4	1	2010-06-04	Falcon 9	NaN	LEO	CCSFS SLC 40	None None	1	False	False	False
5	2	2012-05-22	Falcon 9	525.0	LEO	CCSFS SLC 40	None None	1	False	False	False
6	3	2013-03-01	Falcon 9	677.0	ISS	CCSFS SLC 40	None None	1	False	False	False
7	4	2013-09-29	Falcon 9	500.0	PO	VAFB SLC 4E	False Ocean	1	False	False	False
8	5	2013-12-03	Falcon 9	3170.0	GTO	CCSFS SLC 40	None None	1	False	False	False
...	...	...	...	...	...	...	...	...	...	...	...

```
In [11]: df['Class']=landing_class
         df[['Class']].head(8)
```

Out[11]:

	Class
0	0
1	0
2	0
3	0
4	0
5	0
6	1
7	1

EDA with Data Visualization

- In this part, scatter plot, bar chart and line chart are used for EDA.
- By visualizing data, we can have a clear picture that shows how independent variables influence the target variable
- Scatter plot: determines if any correlation between the two variables exists
- Bar chart: compares the values of a variable at a given point in time
- Line chart: shows a variable that changes over a period of time

- GitHub URL: Lab 5: EDA with Visualization

EDA with SQL

Use SQL Magic function to access the database and perform the following queries:

- GROUP BY: finds the names of the unique launch sites
- LIKE: finds launch sites beginning with 'CCA'
- SUM(): calculates total payload mass
- AVG(): calculates average payload mass
- MIN(): lists the date of the earliest successful landing outcome
- BETWEEN: limits the results within a certain range of values (between 4000 and 6000)
- COUNT(): lists the total number of successful and failure mission outcomes
- MAX(): lists the booster versions having the maximum payload mass
- SUBSTR(): finds the records in the months of 2015
- DESC: ranks successful landing outcomes by date in descending order

GitHub URL: [Lab 4: EDA with SQL notebook]

Build an Interactive Map with Folium

- Map objects including markers, circles, MarkerCluster, MousePosition and lines are used to build the interactive map
- Markers: labels the names of launch sites and show the distance
- Circles: highlights the launch sites on the map
- MarkerCluster: shows the number of markers having the same coordinate on the map
- MousePosition: finds coordinates on the map to calculate the distance
- Lines: highlights the proximity between launch sites and coastlines/highways/railways

- GitHub URL: Lab 6: Interactive Visual Analytics with Folium

Build a Dashboard with Plotly Dash

- An interactive dashboard encompasses a dropdown list, a pie chart, a range slider and a scatter plot
- Dropdown list: selects certain data to be visualized
- Pie chart: compares the proportion of different data on a circular statistical graphic (e.g. success count vs failure count)
- Range slider: selects different payload ranges to see launches with different payload mass on scatter plot
- Scatter plot: presents the relationship between the payload mass and class in a dataset

- GitHub URL: Lab 7: Build an Interactive Dashboard with Ploty Dash

Predictive Analysis (Classification)

- The process of predictive analysis demonstrates as follows:

1. Preprocessing data: standardizes the data of predictor variables (X)
2. Train-test split: uses 80% of data as the training set and 20% of data as the test set
3. GridSearch: finds the hyperparameters that allow algorithms to perform best
4. Algorithms: fit the data in Logistic Regression, Support Vector Machines, Decision Tree Classifier, and K-Nearest Neighbors respectively
5. Model evaluation: compares the best accuracy score of these algorithms and find the algorithm with the highest accuracy score

- GitHub URL: Lab 8: SpaceX Machine Learning Prediction

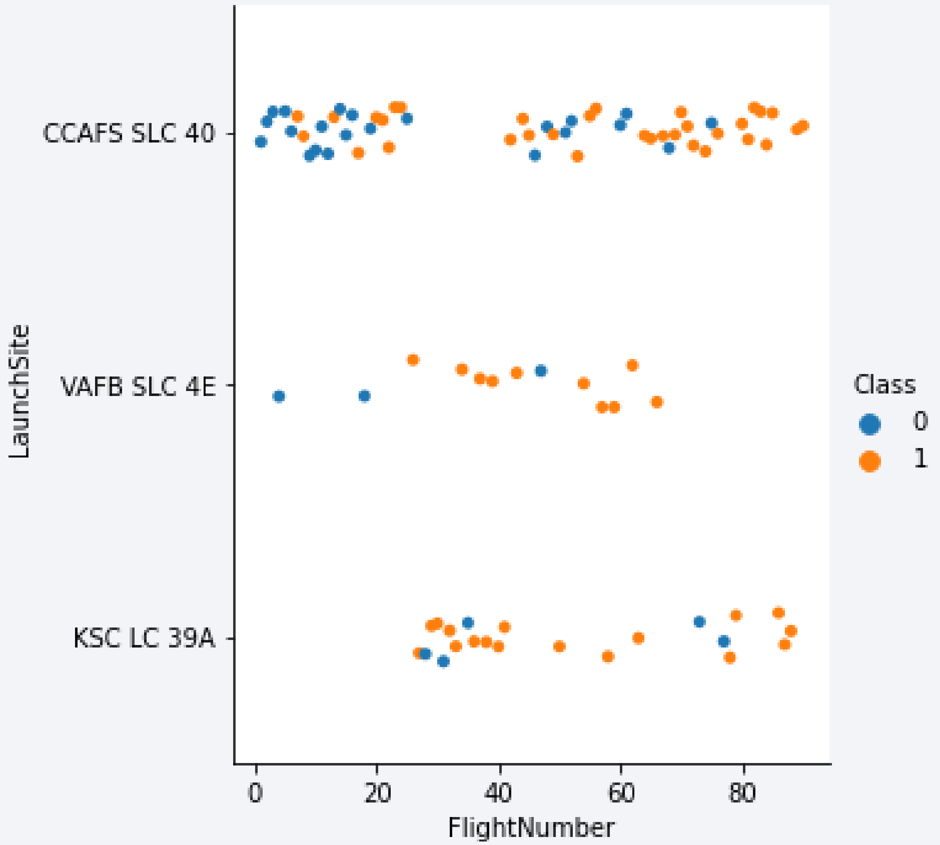
Results

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

Section 2

Insights drawn from EDA

Flight Number vs. Launch Site

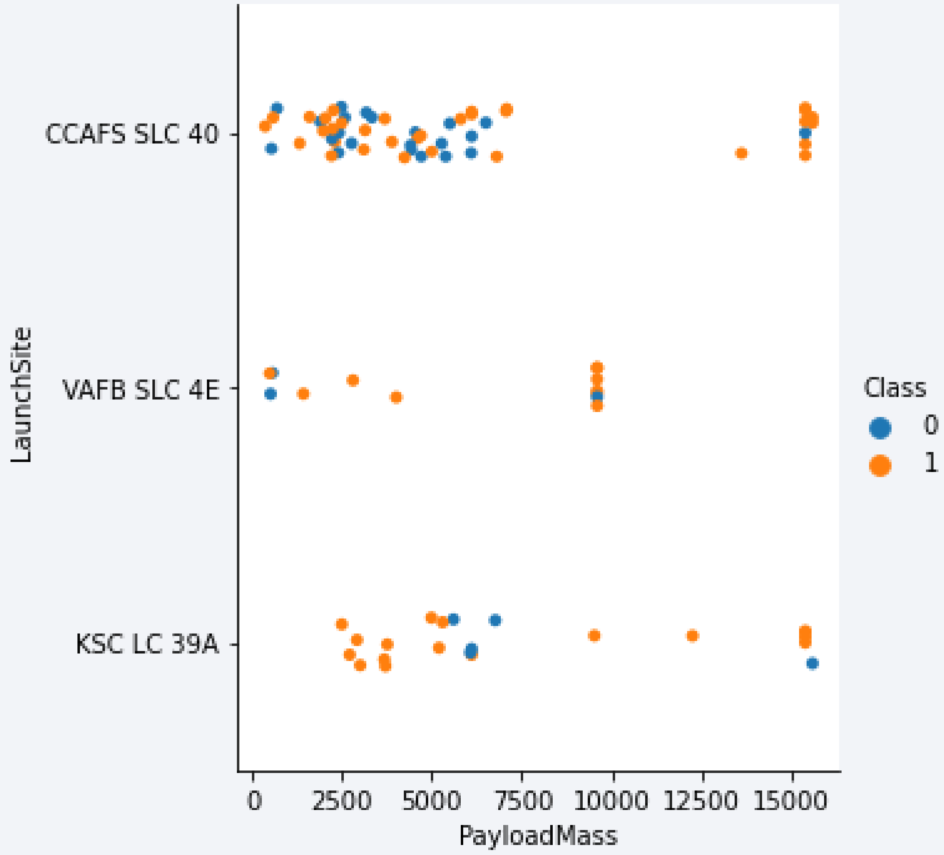


Explanations:
For CCAFS SLC launchsite, when the flight number increases, the success rate increases accordingly.

For VAFB-SLC launchsite, the trend is similar to CCAFS SLC launchsite but no flight number is greater than 70.

For KSC LC launchsite, no flight number is less than 20 and there is a weak relationship between flight number and success rate.

Payload vs. Launch Site



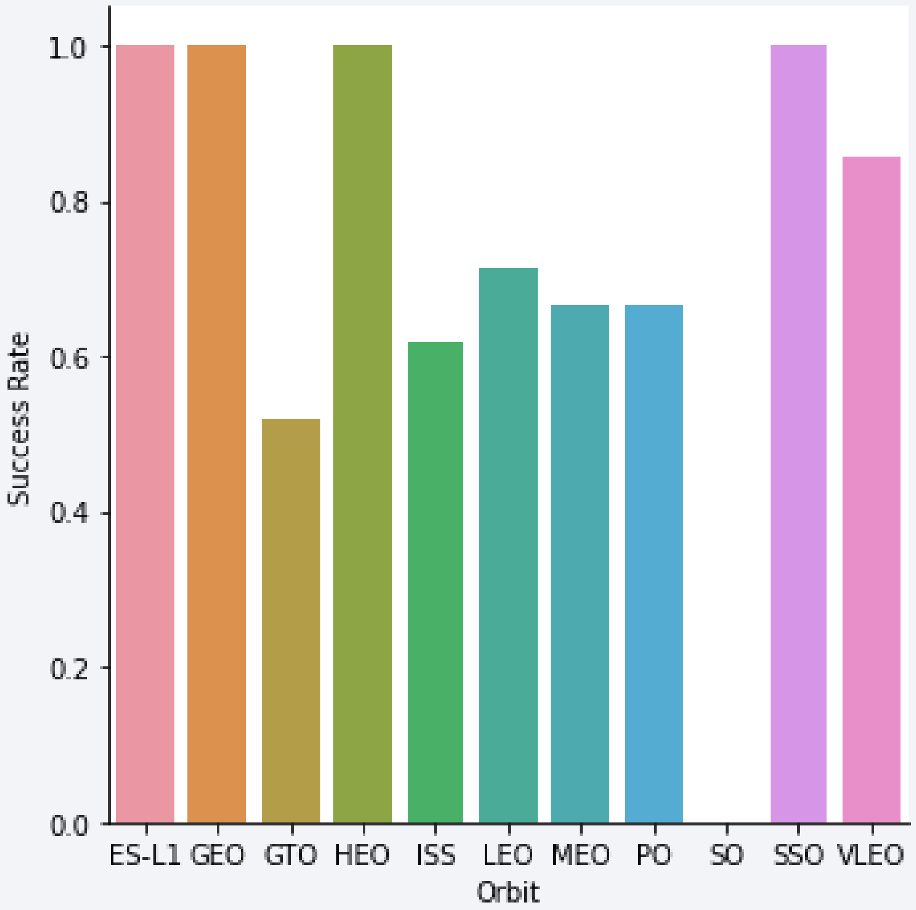
Explanations:

For CCAFS SLC launchsite, when the payload mass is larger than 10,000, the success rate is higher.

For VAFB-SLC launchsite, no rocket is launched for heavy payload mass (greater than 10,000).

For KSC LC launchsite, no rocket is launched for light payload mass (less than 2,500) and there is a weak relationship between the payload mass and success rate.

Success Rate vs. Orbit Type



Explanations:

ES-L1, GEO, HEO and SSO have the highest success rate at 1 meaning landings are very likely to be successful.

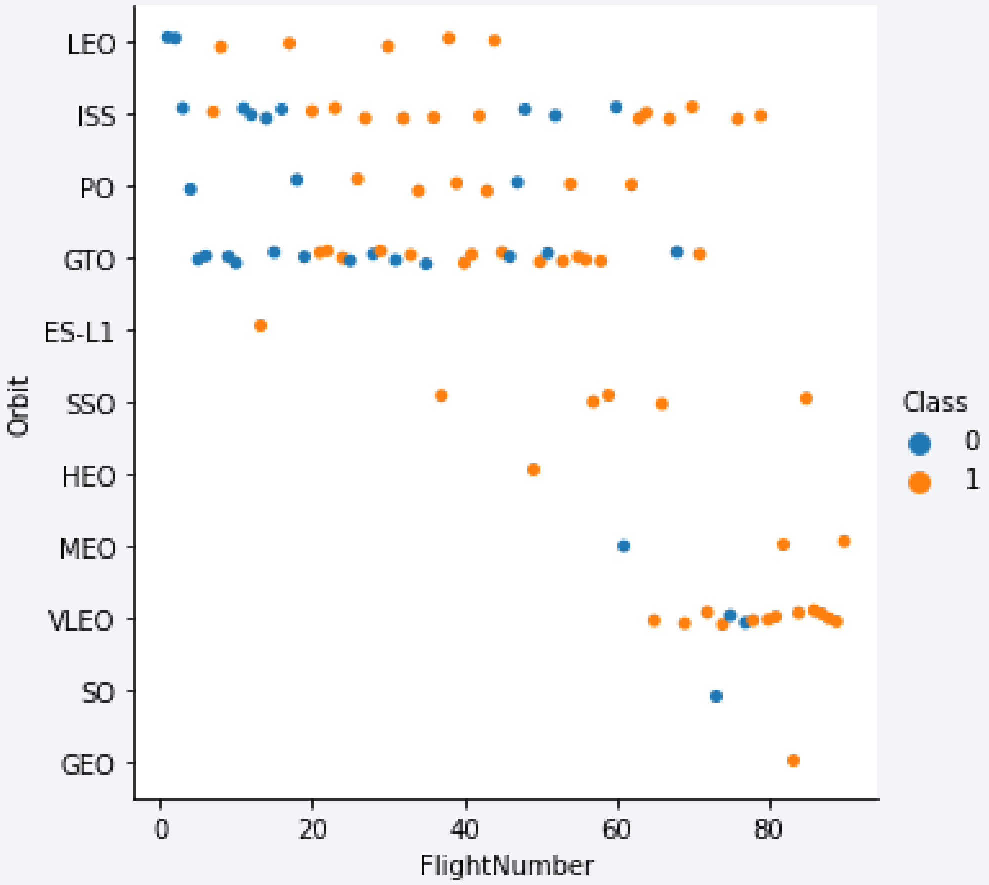
VLEO has the second highest success rate and LEO has the third highest success rate.

MEO, PO and ISS have a moderate success rate between 0.6 and 0.7.

In GTO orbit, only a half of launches are successful.

SO has the least success rate at 0.

Flight Number vs. Orbit Type



Explanations:
In LEO, ISS and PO orbits, the success rate is associated with the number of flights. When the flight number becomes bigger, the landing is more likely to succeed.

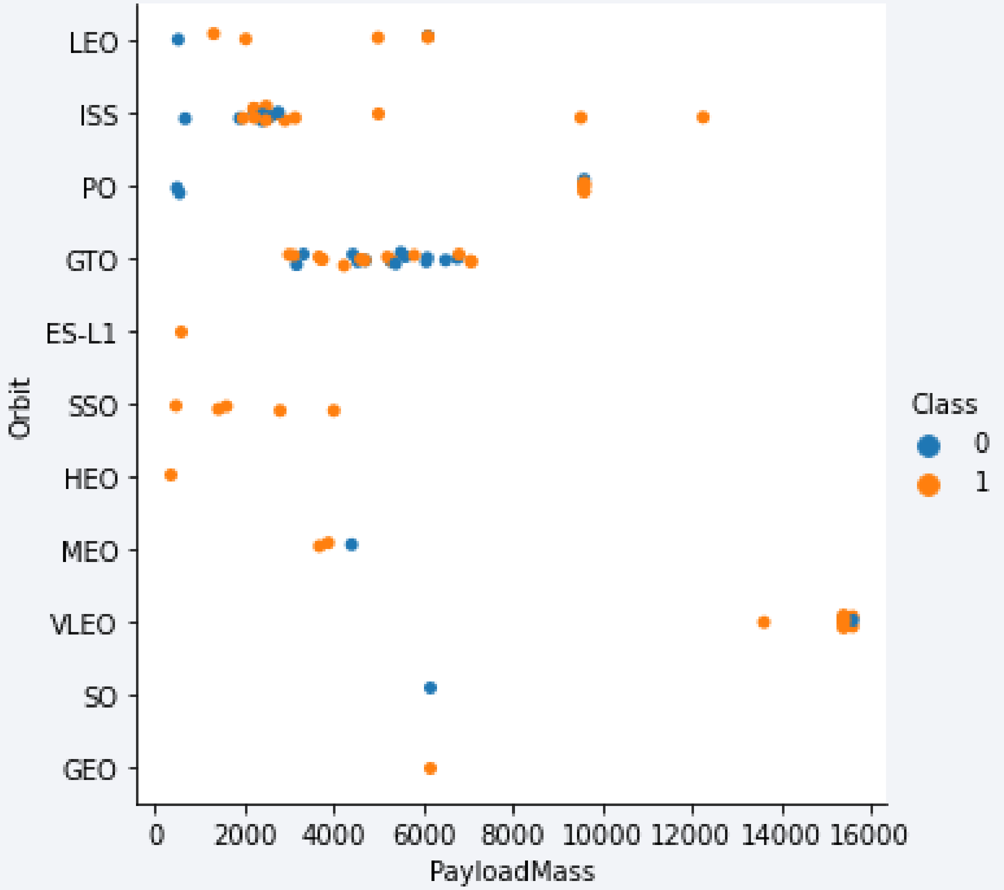
When in GTO orbit, there seems to be no relationship between the flight number and success rate.

When in VLEO orbit, no flight number is smaller than 60, there also seems to be no relationship between the flight number and success rate.

In ES-L1, SSO, HEO and GEO orbits, all launches are successful but there is only a small number of flights for prediction.

Similarly, in MEO and SO orbits, although the larger flight number has a higher success rate, the number of flights is too small.

Payload vs. Orbit Type



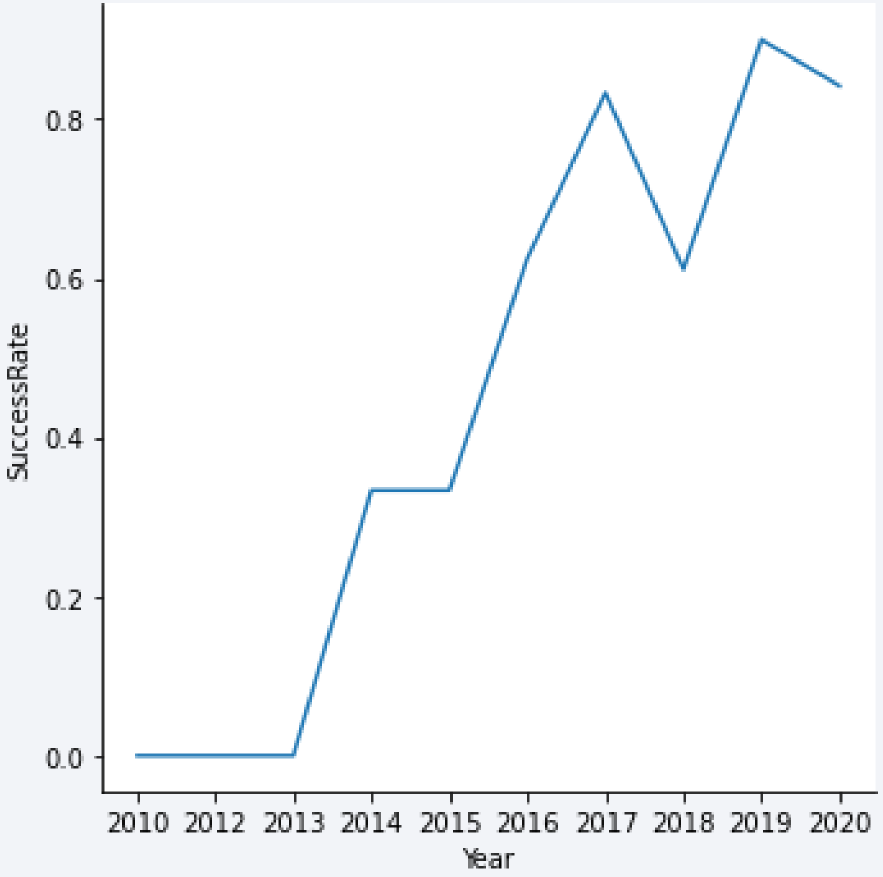
Explanations:
With heavy payloads the successful landing or positive landing rate are more for Polar, LEO and ISS.

By contrast, with light payloads (no more than 4,000) the successful landing or positive landing rate can be higher for ES-L1, SSO, HEO and MEO.

There seems to be no relationship between the payload mass and success rate for GTO.

For other orbits, because there is a very small number of launches, it lacks robust evidence to conclude whether a relationship exists.

Launch Success Yearly Trend




```
Explanations:
Generally speaking, the success rate has increased
significantly from 2014 onward. Especially, the success
rate increases two-fold in 2017 compared to it in 2014.
After experiencing a decrease in 2018, the success rate
rebounds and reaches the peak in 2019.
```

All Launch Site Names

- Find the names of the unique launch sites
- There are 4 unique launch sites including
  - CCAFS LC-40
  - CCAFS SLC-40
  - KSC LC-39A
  - VAFB SLC-4E

Task 1

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

In [7]:
```
%%sql

SELECT Launch_Site From SPACEXTBL
GROUP BY Launch_Site
```

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

Out[7]:

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

Launch Site Names Begin with 'CCA'

- Find 5 records where launch sites begin with `CCA`
- The result is displayed in chronological order by default

Task 2

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

In [8]:

```
%%sql

SELECT * From SPACEXTBL
WHERE Launch_Site LIKE 'CCA%'
LIMIT 5
```

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

Out[8]:

Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Orbit	Customer	Mission_Outcome	Landing _Outcome
04-06-2010	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success	Failure (parachute)
08-12-2010	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)
22-05-2012	07:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success	No attempt
08-10-2012	00:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500	LEO (ISS)	NASA (CRS)	Success	No attempt
01-03-2013	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677	LEO (ISS)	NASA (CRS)	Success	No attempt

Total Payload Mass

- Calculate the total payload carried by boosters from NASA
- The total payload mass is calculated as 45596

Task 3

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

In [9]:

```
%%sql

SELECT SUM(PAYLOAD_MASS__KG_) AS 'Total Payload Mass' FROM SPACEXTBL
WHERE Customer = 'NASA (CRS)'
```

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

Out[9]:

Total Payload Mass
45596

Average Payload Mass by F9 v1.1

- Calculate the average payload mass carried by booster version F9 v1.1
- The average payload mass is calculated as 2928.4

Task 4

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

In [10]:
```
%%sql

SELECT AVG(PAYLOAD_MASS__KG_) AS "Average Payload Mass" FROM SPACEXTBL
WHERE Booster_Version = 'F9 v1.1'
```

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

Out[10]:

Average Payload Mass
2928.4

First Successful Ground Landing Date

- Find the dates of the first successful landing outcome on ground pad
- The result is 22 December 2015

In [11]:

```
%%sql

SELECT MIN(substr(Date,7, 4)), Date FROM SPACEXTBL
WHERE "Landing _Outcome" = 'Success (ground pad)'
```

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

Out[11]:

MIN(substr(Date,7, 4))	Date
2015	22-12-2015

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
- There are 4 types of boosters which meet the condition showing as follows

Task 6

List the names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000

In [12]:

```
%%sql
SELECT Booster_Version FROM SPACEXTBL
WHERE "Landing _Outcome" = 'Success (drone ship)'
AND PAYLOAD_MASS__KG_ BETWEEN 4000 AND 6000
```

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

Out[12]:

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
- The total successful mission outcome is actually 100, the total failure mission outcome is just 1.

Task 7

List the total number of successful and failure mission outcomes

In [13]:

```
%%sql
SELECT Mission_Outcome, COUNT(Mission_Outcome) FROM SPACEXTBL
GROUP BY Mission_Outcome
```

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

Out[13]:

Mission_Outcome	COUNT(Mission_Outcome)
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
- The maximum payload mass is 15600
- There are 12 boosters which have carried the maximum payload mass

Task 8

List the names of the booster_versions which have carried the maximum payload mass. Use a subquery

In [14]:

```
%%sql
SELECT Booster_Version, PAYLOAD_MASS__KG_ FROM SPACEXTBL
WHERE PAYLOAD_MASS__KG_ IN (SELECT MAX(PAYLOAD_MASS__KG_) FROM SPACEXTBL)
```

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

Out[14]:

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
- There are two failed landings which happened in January and April respectively, the specific information shows as follows

In [15]:

```
%%sql
SELECT substr(Date, 4, 2) AS Month, "Landing _Outcome", Booster_Version, Launch_Site
FROM SPACEXTBL WHERE substr(Date, 7, 4) = '2015' AND "Landing _Outcome" = 'Failure (drone ship)'
```

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

Out[15]:

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

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

Task 10

Rank the count of successful landing_outcomes between the date 04-06-2010 and 20-03-2017 in descending order.

In [16]:

```
%%sql

SELECT "Landing _Outcome", Booster_Version, Launch_Site, Date
FROM SPACEXTBL WHERE substr(Date, 7, 4) BETWEEN '2010' AND '2017'
ORDER BY substr(Date, 7, 4) DESC, substr(Date, 4, 2) DESC, substr(Date, 1, 2) DESC
LIMIT 50 OFFSET 15
```

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

Out[16]:

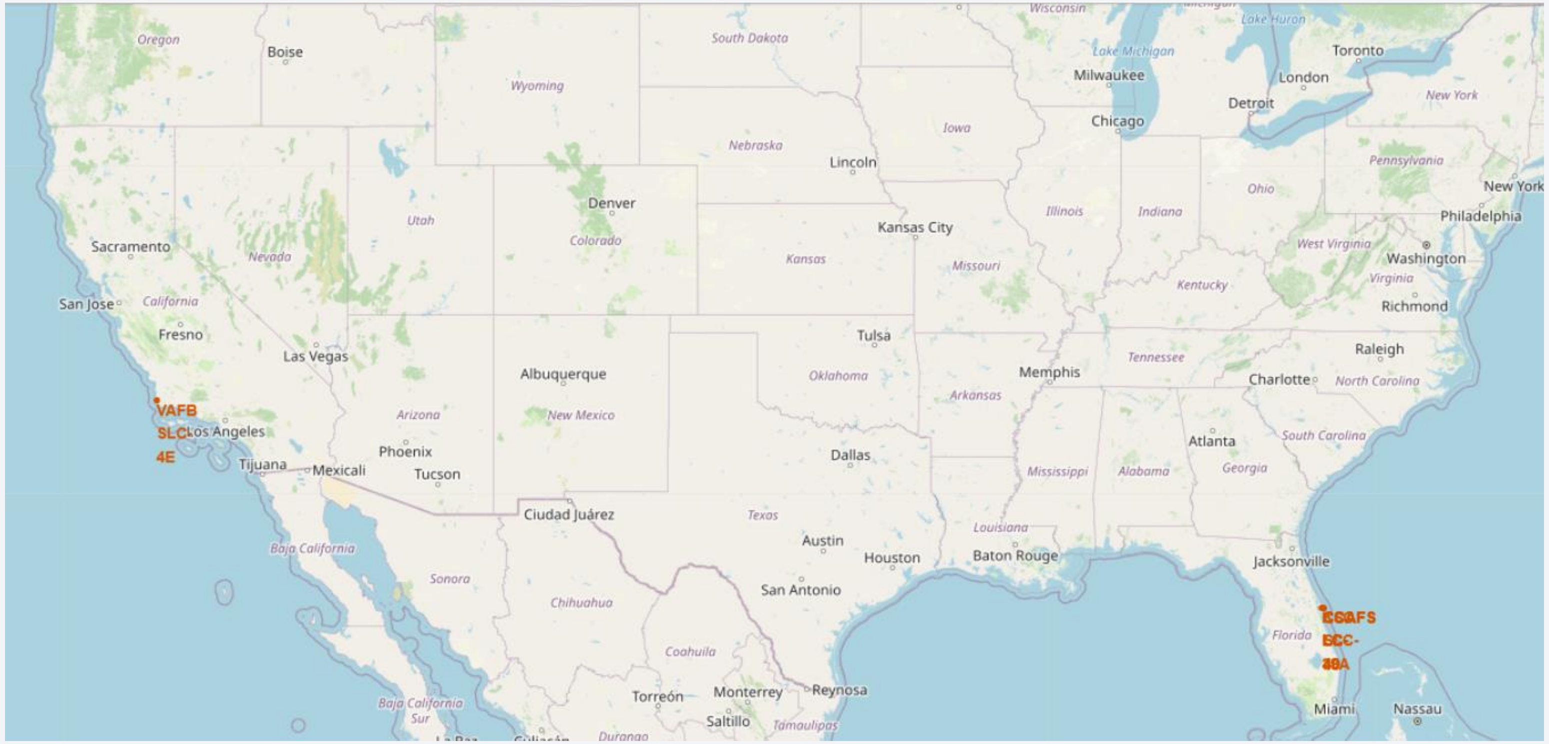
Landing _Outcome	Booster_Version	Launch_Site	Date
No attempt	F9 FT B1030	KSC LC-39A	16-03-2017
Success (ground pad)	F9 FT B1031.1	KSC LC-39A	19-02-2017
Success (drone ship)	F9 FT B1029.1	VAFB SLC-4E	14-01-2017
Success (drone ship)	F9 FT B1026	CCAFS LC-40	14-08-2016
Success (ground pad)	F9 FT B1025.1	CCAFS LC-40	18-07-2016
Failure (drone ship)	F9 FT B1024	CCAFS LC-40	15-06-2016
Success (drone ship)	F9 FT B1023.1	CCAFS LC-40	27-05-2016
Success (drone ship)	F9 FT B1022	CCAFS LC-40	06-05-2016
Success (drone ship)	F9 FT B1021.1	CCAFS LC-40	08-04-2016
Failure (drone ship)	F9 FT B1020	CCAFS LC-40	04-03-2016
Failure (drone ship)	F9 v1.1 B1017	VAFB SLC-4E	17-01-2016
Success (ground pad)	F9 FT B1019	CCAFS LC-40	22-12-2015
Precluded (drone ship)	F9 v1.1 B1018	CCAFS LC-40	28-06-2015
No attempt	F9 v1.1 B1016	CCAFS LC-40	27-04-2015
Failure (drone ship)	F9 v1.1 B1015	CCAFS LC-40	14-04-2015
No attempt	F9 v1.1 B1014	CCAFS LC-40	02-03-2015
Controlled (ocean)	F9 v1.1 B1013	CCAFS LC-40	11-02-2015
Failure (drone ship)	F9 v1.1 B1012	CCAFS LC-40	10-01-2015
Uncontrolled (ocean)	F9 v1.1 B1010	CCAFS LC-40	21-09-2014
No attempt	F9 v1.1 B1011	CCAFS LC-40	07-09-2014
No attempt	F9 v1.1	CCAFS LC-40	05-08-2014
Controlled (ocean)	F9 v1.1	CCAFS LC-40	14-07-2014
Controlled (ocean)	F9 v1.1	CCAFS LC-40	18-04-2014
No attempt	F9 v1.1	CCAFS LC-40	06-01-2014
No attempt	F9 v1.1	CCAFS LC-40	03-12-2013
Uncontrolled (ocean)	F9 v1.1 B1003	VAFB SLC-4E	29-09-2013
No attempt	F9 v1.0 B0007	CCAFS LC-40	01-03-2013
No attempt	F9 v1.0 B0006	CCAFS LC-40	08-10-2012
No attempt	F9 v1.0 B0005	CCAFS LC-40	22-05-2012
Failure (parachute)	F9 v1.0 B0004	CCAFS LC-40	08-12-2010
Failure (parachute)	F9 v1.0 B0003	CCAFS LC-40	04-06-2010

Section 3

Launch Sites Proximities Analysis

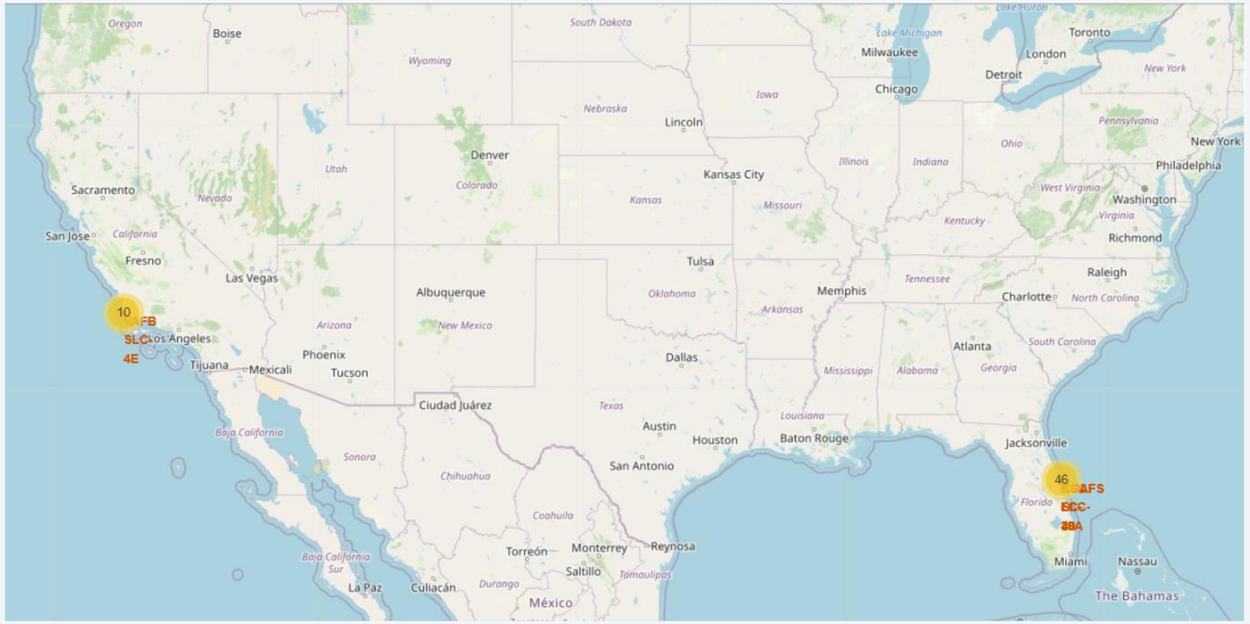
Launch sites' location on a Folium map

- All launch sites are very close to the coast
- No launch site is close to the Equator line



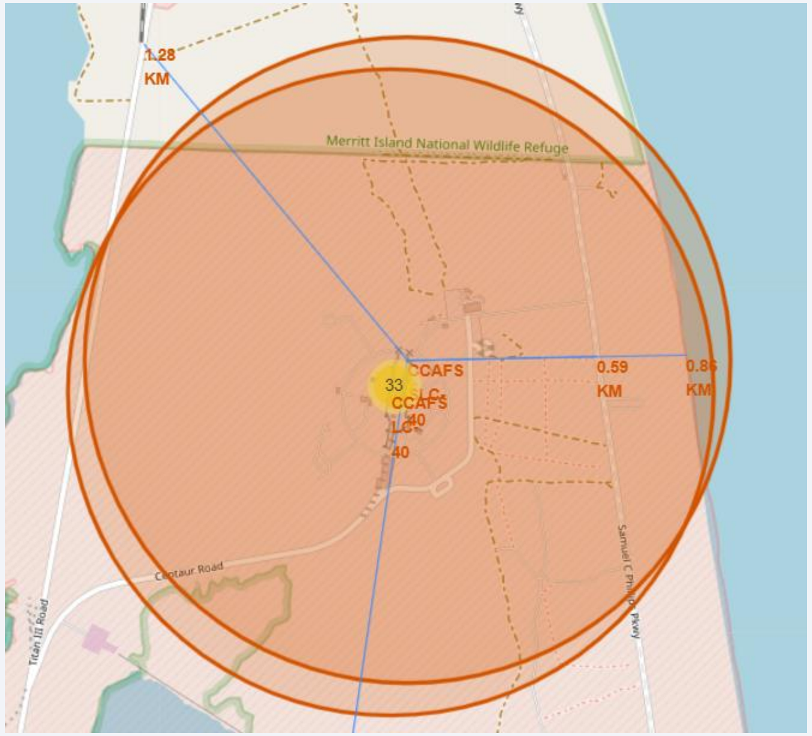
The success/failed launches for each site on the map

- Create a MarkCluster that counts the number of sites having the same coordinate
- Visualize launches with different colors on the map
- Label the success launches as ‘green’ and the failed launches as ‘red’ that helps identify which launch sites have relatively high success rates



The launch site's proximities to the coastline, highway, railway and city

- The distance to the nearest coastline is about 0.86 km
- The distance to the nearest highway is about 0.59 km
- The distance to the nearest railway is about 1.28 km
- The launch site is in close proximity with these infrastructure



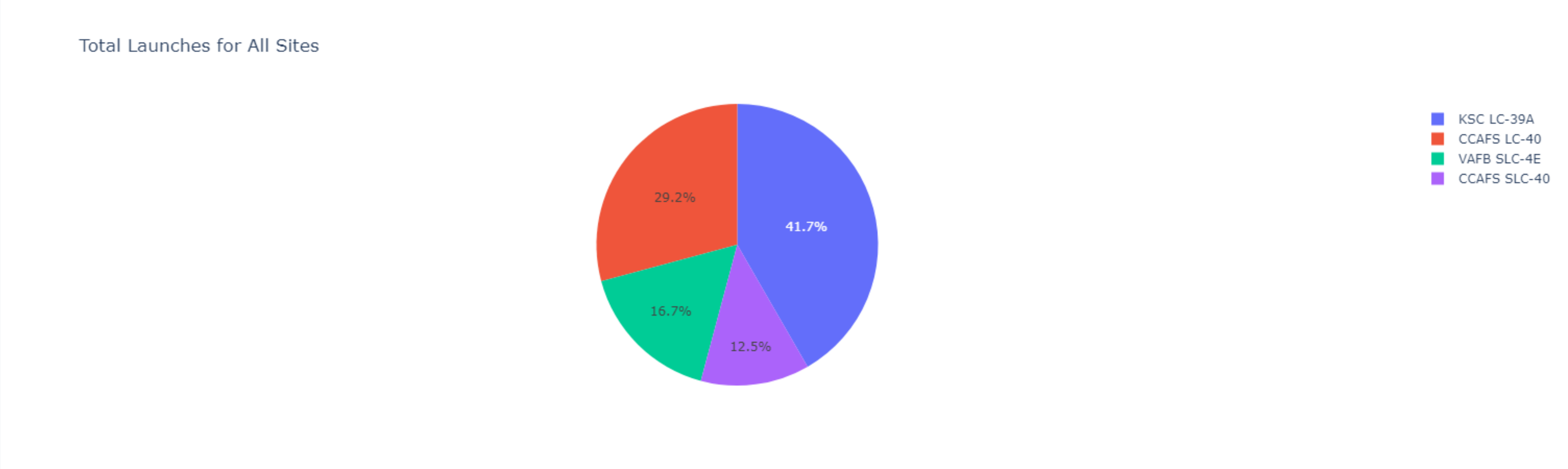
- The distance to the nearest city is about 18.35 km
- The launch site keeps away from the city

Section 4

Build a Dashboard with Plotly Dash

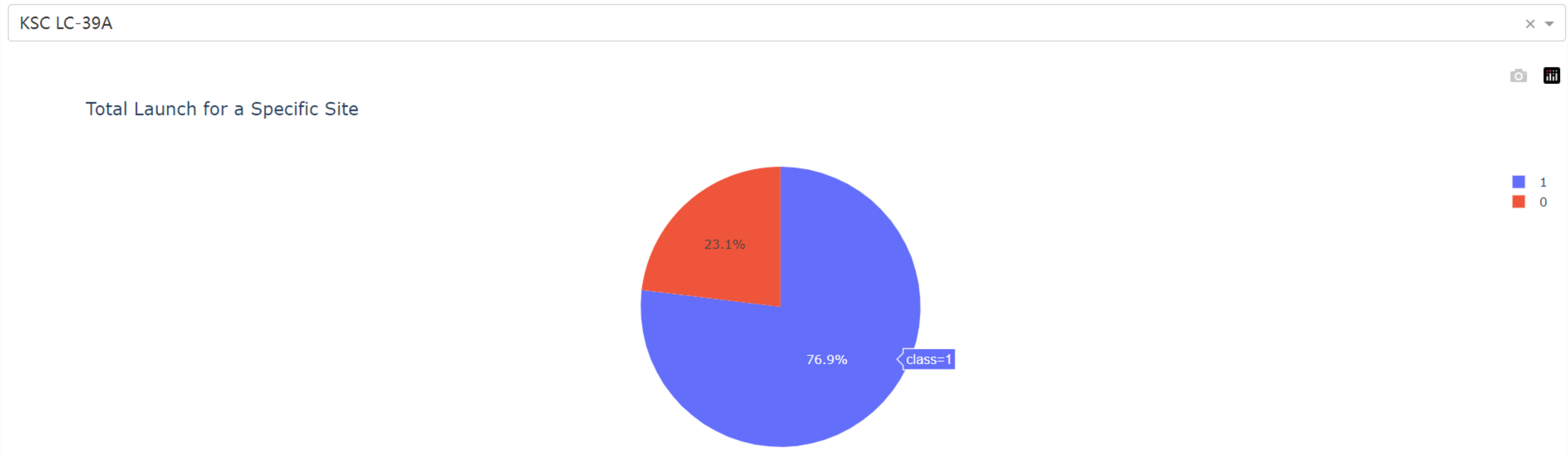
Total launches for all sites

- Total launches for all sites show in a pie chart
- Most success launches take place in KSC LC-39A and CCAFS LC-40 with the approximate proportion of 70%



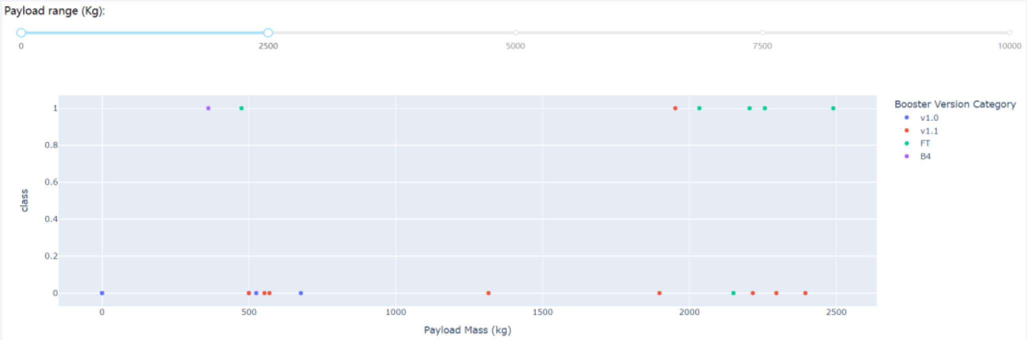
The launch site with highest launch success ratio

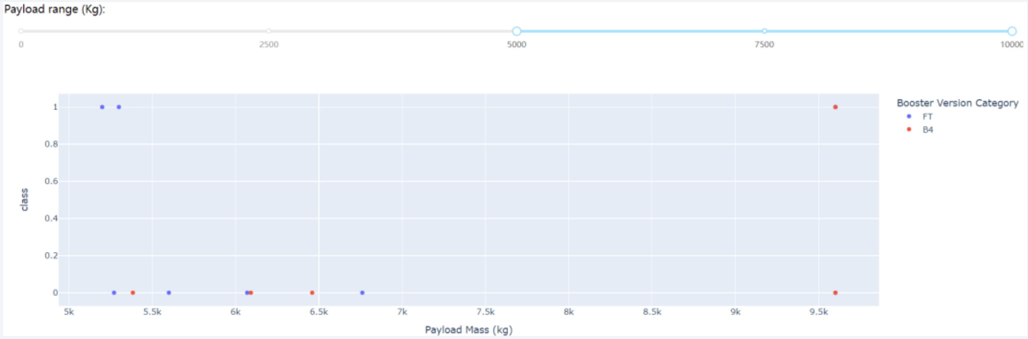
- KSC LC-39A has the highest launch success ratio at 76.9%



Payload vs. Launch Outcome with different payloads

- When the payload is less than 2500, FT booster version has the largest success rate.
- When the payload lies between 2500 and 5000, B4 booster version has the largest success rate. (Given that B5 booster version just has one successful sample, that may not be representative).
- Meanwhile, FT booster version still has a high success rate in this range.
- When the payload is greater than 5000, most of the launches fails. Only FT and B4 booster versions carries this heavy payload.
- Launches with v1.0 and v1.1 booster versions have a low success rate. All Launches with v1.0 fails.



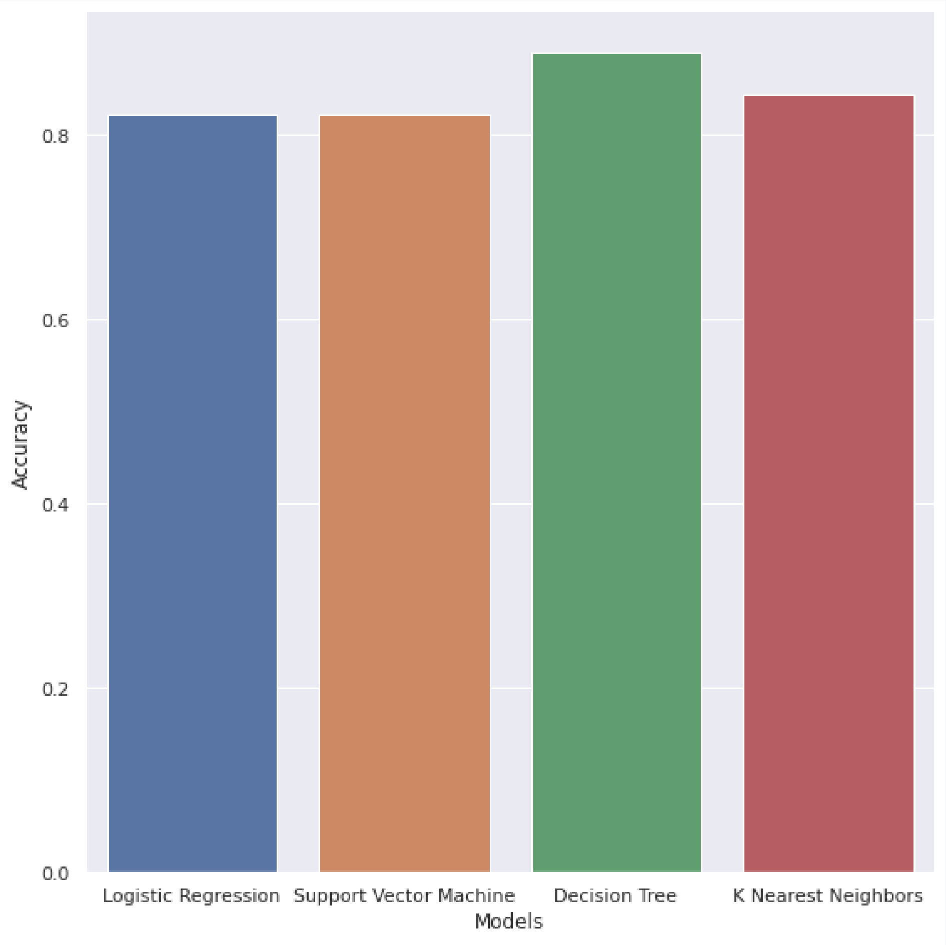


Section 5

Predictive Analysis (Classification)

Classification Accuracy

- Bar chart compares the classification accuracy amongst Logistic Regression, Support Vector Machine, Decision Tree and K-Nearest Neighbors models
- Decision Tree has the highest classification accuracy (~0.8777)



*See appendix for accuracy score of Logistic Regression, Support Vector Machine, and K-Nearest Neighbors models

Confusion Matrix

- Confusion Matrix of the decision tree (the best performing model)
- 3 cases are misclassified in fitting the test set. They should have been labeled as ‘land’, but they have been labeled as ‘did not land’

*See appendix for confusion matrix of Logistic Regression, Support Vector Machine, and K-Nearest Neighbors models

Conclusions

- Flight number and payload have a positive correlation with the success rate (except for KSC LC launch site).
- There seem to be a weak relationship between orbit types and the success rate.
- The launch success rate has increased gradually since 2014.
- The maximum payload mass is 15600 and the overwhelming majority (~98%) of mission outcomes are successful.
- All launch sites are close to the coastline and some infrastructure. Launch sites keep away from the city.
- KSC LC-39A and CCAFS LC-40 are main launch sites. KSC LC-39A has the highest success ratio.
- FT and B4 boosters have a higher success rate.
- Decision tree performs the best in make predictions if hyperparameters are correctly chosen.

Appendix

- Confusion matrix and accuracy score of Logistic Regression, Support Vector Machine and K-Nearest Neighbors.

Logistic Regression

0.8222

Support Vector Machine

0.8222

K-Nearest Neighbors

0.8444

Thank you!