IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Yen Po Liu
10/12/2022

Outline

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

Executive Summary

- Summary of methodologies
  - Data Collection through REST API
  - Data Collection with Web Scraping
  - Data Wrangling
  - Exploratory Data Analysis with SQL
  - Exploratory Data Analysis with Data Visualization
  - Interactive Visual Analytics with Folium
  - Machine Learning Prediction
- Summary of all results
  - Exploratory Data Analysis (EDA) result
  - Interactive analytics in screenshots
  - Predictive Analytics result

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.
  - The goal of this project is to develop a machine learning prediction pipeline to predict if the first stage will land successfully.
- Problems we want to find answers
  - What are the critical factors that effect landing outcome of first stage?
  - How to make the best prediction to first stage landing?
  - What operating conditions needs to be in place to improve success rate of landing?

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - REST API and Web scraping
- Perform data wrangling
  - One-hot encoding was applied to 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
  - Logistic Regression, SVM, Decision Tree, K-nearest neighbor.

Data Collection - API

- Data collected through API to connect endpoint. (https://api.spacexdata.com/v4/launches/past)

Request

- Request the endpoint to get launch data

Get

- Received raw data in .json format

Normalized

- Normalized json-formatted into pandas DataFrame

Data Collection – Web Scraping

- Data collected from Wikipedia (https://en.wikipedia.org/w/index.php?title=List_of_Falcon_9_and_Falcon_Heavy_launches&oldid=1027686922)

Request

- Requested webpage to collect text data in HTML format.

Parse

- Applied BeautifulSoup module as a parser to extract launch table.

Transform

- Transformed HTML data into data frame through pandas module.

Data Collection – SpaceX API

Request

- Request to endpoints.
- Extract launch data into lists through **API functions**.

```
spacex_url="https://api.spacexdata.com/v4/launches/past"
response = requests.get(spacex_url)
```

Wangling

- Normalize data structure, extract columns to lists respectively.
- Import lists to dictionary with functions, transform into data frame.

```
# Takes the dataset and uses the launchpad column to call the AP
def getLaunchSite(data):
    for x in data['launchpad']:
        if x:
            response = requests.get("https://api.spacexdata.com/v4
            Longitude.append(response['longitude'])
            Latitude.append(response['latitude'])
            LaunchSite.append(response['name'])
```

Cleaning

- Filter that data to only include “Falcon 9 ”launches.
- Replace null in “Payload column” with mean value.

```
# Calculate the mean value of PayloadMass column
meanPay = data_falcon9['PayloadMass'].mean()
meanPay
# Replace the np.nan values with its mean value
data_falcon9['PayloadMass'] =data_falcon9['PayloadMass'].repla
data_falcon9.head()
```

Data Collection - Scraping

Request

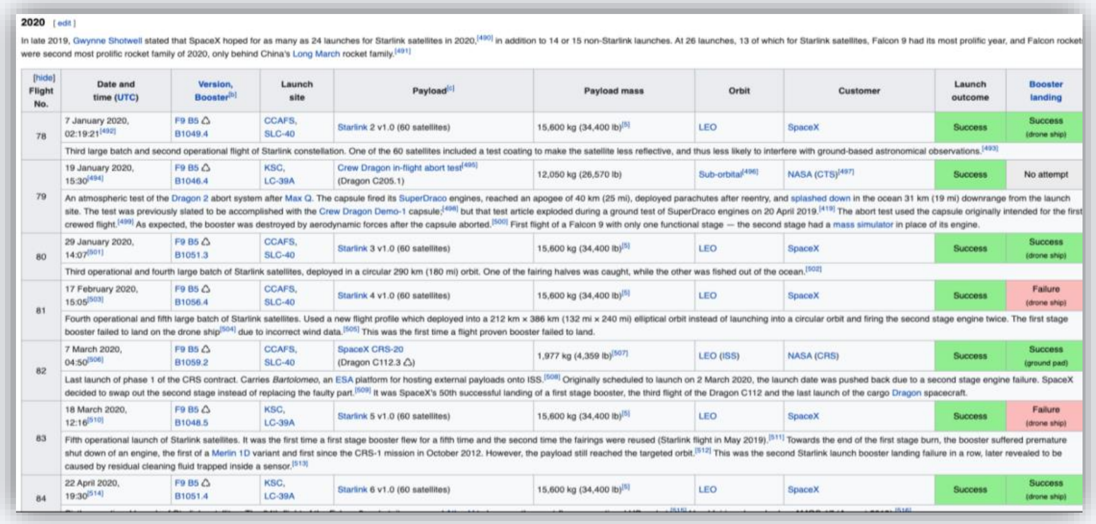
- Request to Falcon9 launch Wikipedia webpage.

Parse

- Parse the HTML with beautiful soup module, find out the Falcon 9 launch table.

Extract

- Implement "for loop", extract each cell from table, fill up new launch dictionary.



	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	Success	F9 v1.0B0004.1	Failure	8 December 2010	15:43
2	3	CCAFS	Dragon	525 kg	LEO	NASA	Success	F9 v1.0B0005.1	No attempt\n	22 May 2012	07:44
3	4	CCAFS	SpaceX CRS-1	4,700 kg	LEO	NASA	Success\n	F9 v1.0B0006.1	No attempt	8 October 2012	00:35
4	5	CCAFS	SpaceX CRS-2	4,877 kg	LEO	NASA	Success\n	F9 v1.0B0007.1	No attempt\n	1 March 2013	15:10
...	...	...	...	...	...	...	...	...	...	...	...
116	117	CCSFS	Starlink	15,600 kg	LEO	SpaceX	Success\n	F9 B5B1051.10	Success	9 May 2021	06:42
117	118	KSC	Starlink	~14,000 kg	LEO	SpaceX	Success\n	F9 B5B1058.8	Success	15 May 2021	22:56
118	119	CCSFS	Starlink	15,600 kg	LEO	SpaceX	Success\n	F9 B5B1063.2	Success	26 May 2021	18:59
119	120	KSC	SpaceX CRS-22	3,328 kg	LEO	NASA	Success\n	F9 B5B1067.1	Success	3 June 2021	17:29
120	121	CCSFS	SXM-8	7,000 kg	GTO	Sirius XM	Success\n	F9 B5	Success	6 June 2021	04:26

Data Wrangling

- Perform exploratory data analysis and determine training labels

Analysis

- Use “value_counts” to determine the number and occurrence of orbits and launch sites.

Identify

- Exploring landing outcomes feature, classify all outcome situation into binary result (1 for succeed ,0 for failed)

Labelling

- Create a landing outcome label from outcome column that is going to be predicted in the next step.

	FlightNumber	Date	BoosterVersion	PayloadMass	Orbit	LaunchSite	Outcome	Flights	GridFins	Reused	Legs	LandingPad	Block	ReusedCount	Serial	Longitude	Latitude	Class
0	1	2010-06-04	Falcon 9	6104.959412	LEO	CCAFS SLC 40	None None	1	False	False	False	NaN	1.0	0	B0003	-80.577366	28.5618[illegible]7	0
1	2	2012-05-22	Falcon 9	525.000000	LEO	CCAFS SLC 40	None None	1	False	False	False	NaN	1.0	0	B0005	-80.577366	28.561[illegible]57	0
2	3	2013-03-01	Falcon 9	677.000000	ISS	CCAFS SLC 40	None None	1	False	False	False	NaN	1.0	0	B0007	-80.577366	28.561[illegible]57	0
3	4	2013-09-29	Falcon 9	500.000000	PO	VAFB SLC 4E	False Ocean	1	False	False	False	NaN	1.0	0	B1003	-120.610829	34.6320[illegible]3	0
4	5	2013-12-03	Falcon 9	3170.000000	GTO	CCAFS SLC 40	None None	1	False	False	False	NaN	1.0	0	B1004	-80.577366	28.56185[illegible]	0

EDA with Data Visualization

➤**Objective**: To explore whether features would affect the landing outcome.

- **Scatter plots**: plot out relationship between “Flight number, Launch Site, Orbit Type and Payload Mass”, discover relationships that might lead to higher success landing rate.
- **Line chart**: display the trend of success rate.
- **Bar chart**: check if there are any relationship between success rate and orbit type.

EDA with SQL (query example)

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

```
%sql SELECT DISTINCT(LAUNCH_SITE) FROM SPACEXTBL
```

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

```
%sql SELECT SUM(PAYLOAD_MASS__KG_) FROM SPACEXTBL WHERE CUSTOMER  = 'NASA (CRS)'
```

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

```
%sql SELECT BOOSTER_VERSION,"Landing _Outcome", PAYLOAD_MASS__KG_  FROM SPACEXTBL
WHERE "Landing _Outcome" LIKE 'Success (drone ship)' and PAYLOAD_MASS__KG_ >4000 and PAYLOAD_MASS__KG_ < 6000
```

Build an Interactive Map with Folium

- **Objective**: Trying to observe in geographical view and explore implicit relationship that affect landing outcome.
  - Markers: Launch places and launch events (succeed or failed)
  - Lines: Display the nearest public facility or landmark and its distance from the launch sites.

Build a Dashboard with Plotly Dash

- **Objective**: To understand success rate of each site and landing result that base on different payload mass and booster versions
  - Display each site's successful landing rate in pie chart and payload mass within booster versions in scatter plot.

Predictive Analysis (Classification)

➢ **Objective**: Find out the best of the classification algorithms

- Import data with **numpy** and **pandas** module, use **fix_transform** function to normalized and then use **train test split** function to split data with 20% test size.
- Build machine learning models and test them with different hyperparameters using **GridSearchCV**.
- Display models' accuracy metric.

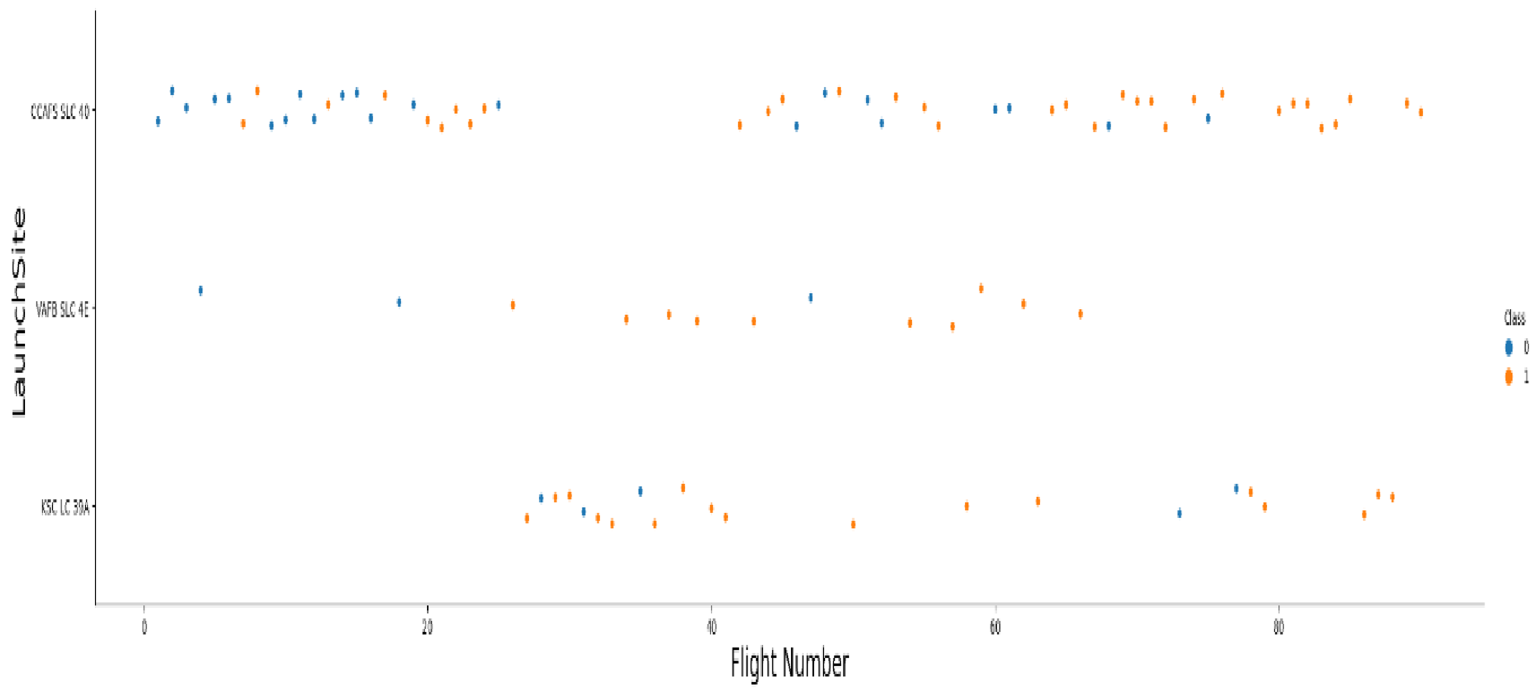
Results

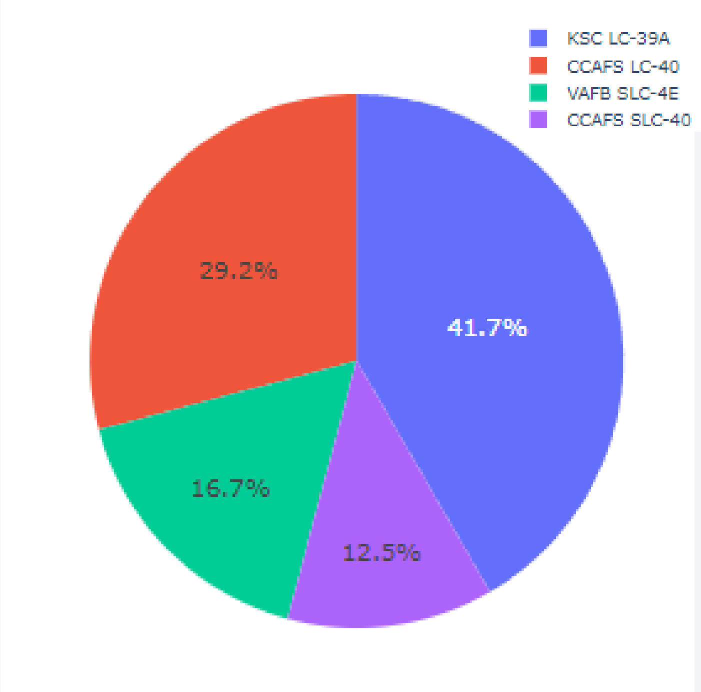
- Exploratory data analysis results
- Interactive analytics demo in screenshots
- Predictive analysis results

Section 2

Insights drawn from EDA

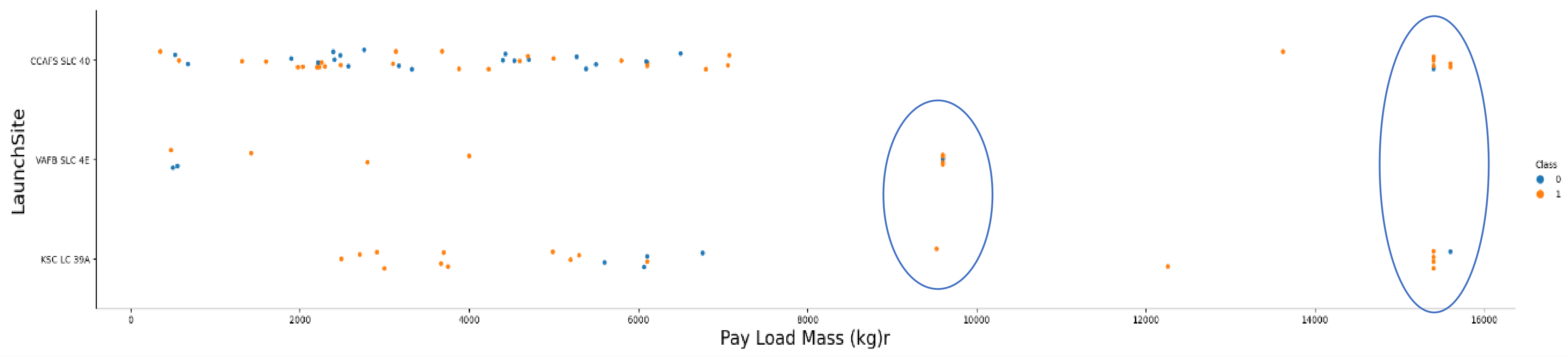
Flight Number vs. Launch Site





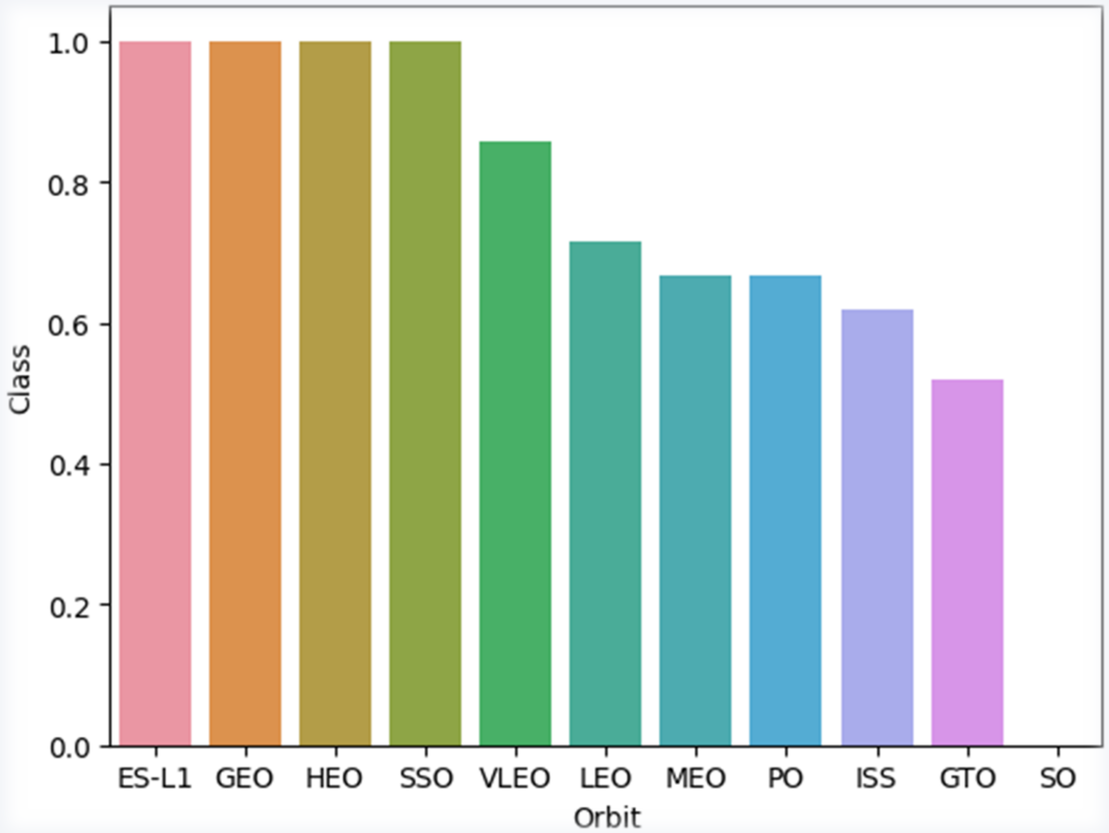
- In recent launches, it seems to have the most successful events and highest success rate at **CCAFS SLC 40**.

Payload vs. Launch Site



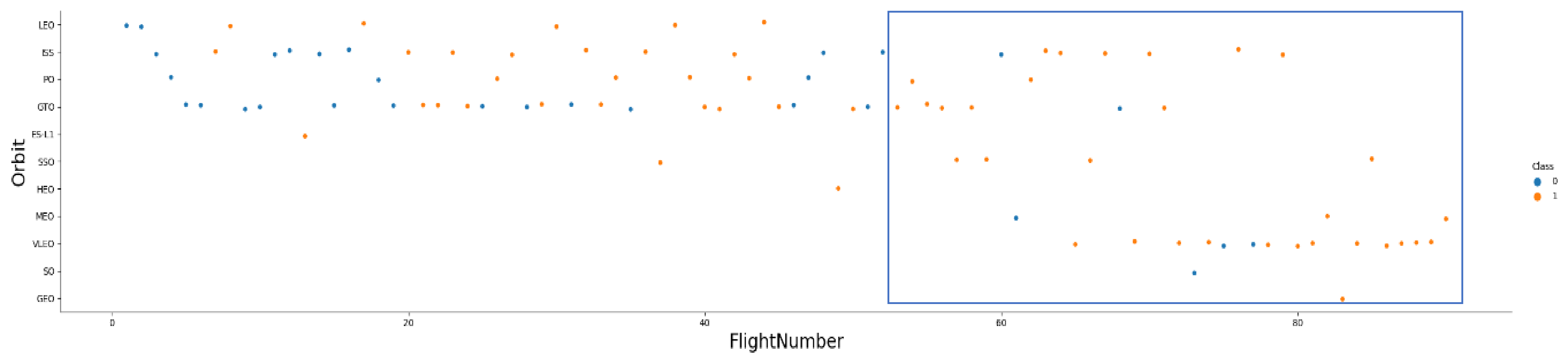
- The greater payload mass the higher success rate for launches.

Success Rate vs. Orbit Type



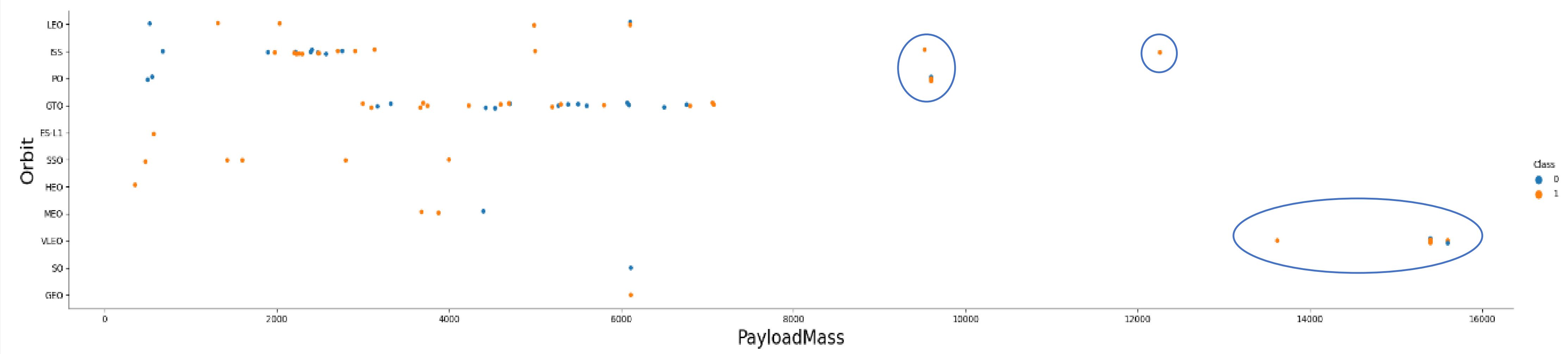
- "ES-L1, GEO, HEO, SSO" orbit types have the highest success rate.

Flight Number vs. Orbit Type



- We can tell that according to the blue box, it seems that last 5 of orbit type had significantly launched recently, and most of them have higher success rate.

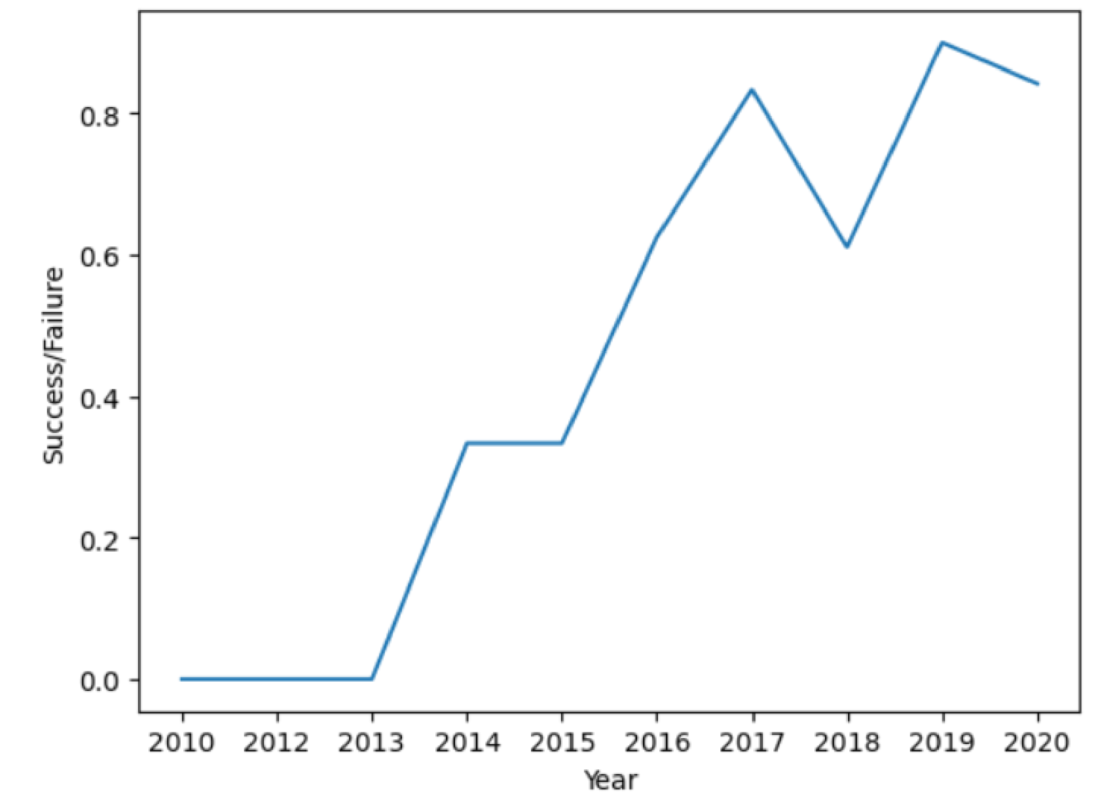
Payload vs. Orbit Type



- With heavy payloads the successful landing or positive landing rate are more for PO, VLEO and ISS.

Launch Success Yearly Trend

- Success rate has generally increased in the last decade, however, the rate performed in 2018 and 2020 decreased, further investigation is recommended to find out reasons that caused the result.



All Launch Site Names

- DISTINCT was used when querying unique launch site name from SpaceX.

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

```
%sql SELECT DISTINCT(LAUNCH_SITE) FROM SPACEXTBL
```

```
 * ibm_db_sa://ygk14970:***@21fecfd8-47b7-4937-840d-d791d02
Done.
```

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

Launch Site Names Begin with 'CCA'

- We used the query below to display 5 records where launch sites begin with `CCA`

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

```
%sql SELECT * FROM SPACEXTBL WHERE LAUNCH_SITE LIKE 'CCA%' LIMIT 5
```

```
 * ibm_db_sa://ygk14970:***@21fecfd8-47b7-4937-840d-d791d0218660.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:31864/bludb
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	07:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success	No attempt
2012-10-08	00:35:00	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

- Calculate the total payload carried by boosters from NASA as 45596 using the query below

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

```
%sql SELECT SUM(PAYLOAD_MASS__KG_) FROM SPACEXTBL WHERE CUSTOMER  = 'NASA (CRS)'
```

```
 * ibm_db_sa://ygk14970:***@21fecfd8-47b7-4937-840d-d791d0218660.bs2io90l08kqb1od
Done.
```

1
45596

Average Payload Mass by F9 v1.1

- Calculate the average payload mass carried by booster version F9 v1.1

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

```
%sql SELECT AVG(PAYLOAD_MASS__KG_) FROM SPACEXTBL WHERE BOOSTER_VERSION LIKE 'F9 v1.1%'
```

```
 * ibm_db_sa://ygk14970:***@21fecfd8-47b7-4937-840d-d791d0218660.bs2io90l08kqb1od8lcg.dat
Done.
```

1
2534

First Successful Ground Landing Date

- Found the dates of the first successful landing outcome on ground pad was 2015-12-22

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

Hint:Use min function

```
%sql SELECT MIN(DATE) FROM SPACEXTBL WHERE "Landing _Outcome" LIKE 'Success (ground pad)'
```

```
 * ibm_db_sa://ygk14970:***@21fecfd8-47b7-4937-840d-d791d0218660.bs2io90l08kqb1od8lcg.datal
Done.
```

1
2015-12-22

Successful Drone Ship Landing with Payload between 4000 and 6000

- We list the names of boosters which have successfully landed on drone ship and had payload mass greater than 4000 but less than 6000

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

```
%sql SELECT BOOSTER_VERSION,"Landing _Outcome", PAYLOAD_MASS__KG_  FROM SPACEXTBL WHERE "Landing _Outcome" LIKE 'Success (drone ship)' and PAYLOAD_MAS
```

```
 * ibm_db_sa://ygk14970:***@21fecfd8-47b7-4937-840d-d791d0218660.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:31864/bludb
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

Total Number of Successful and Failure Mission Outcomes

- Calculated the total number of successful and failure mission outcomes by using count(MISSION_OUTCOME)

List the total number of successful and failure mission outcomes

```
%sql SELECT count(MISSION_OUTCOME) FROM SPACEXTBL GROUP BY MISSION_OUTCOME
```

```
 * ibm_db_sa://ygk14970:***@21fecfd8-47b7-4937-840d-d791d0218660.bs2io90l08k
Done.
```

1
1
99
1

Boosters Carried Maximum Payload

- We extracted the names of the booster which have carried the maximum payload mass by using subquery.

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

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

```
 * ibm_db_sa://ygk14970:***@21fecfd8-47b7-4937-840d-d791d0218660.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:31864/bludb
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

2015 Launch Records

- List the failed landing_outcomes in drone ship, their booster versions, and launch site names for in year 2015, set query date between 2015/01/01 and 2015/12/31

List the failed landing_outcomes in drone ship, their booster versions, and launch site names for in year 2015

```
%sql SELECT BOOSTER_VERSION,LAUNCH_SITE FROM SPACEXTBL WHERE "Landing _Outcome" = 'Failure (drone ship)' AND DATE BETWEEN '2015-01-01' AND '2015-12-31
```

```
 * ibm_db_sa://ygk14970:***@21fecfd8-47b7-4937-840d-d791d0218660.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:31864/bludb
Done.
```

booster_version	launch_site
F9 v1.1 B1012	CCAFS LC-40
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

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

```
%sql SELECT "Landing _Outcome", COUNT("Landing _Outcome") AS FREQ FROM SPACEXTBL WHERE DATE BETWEEN '2010-06-04' AND '2017-03-20' GROUP BY "Landing _O
```

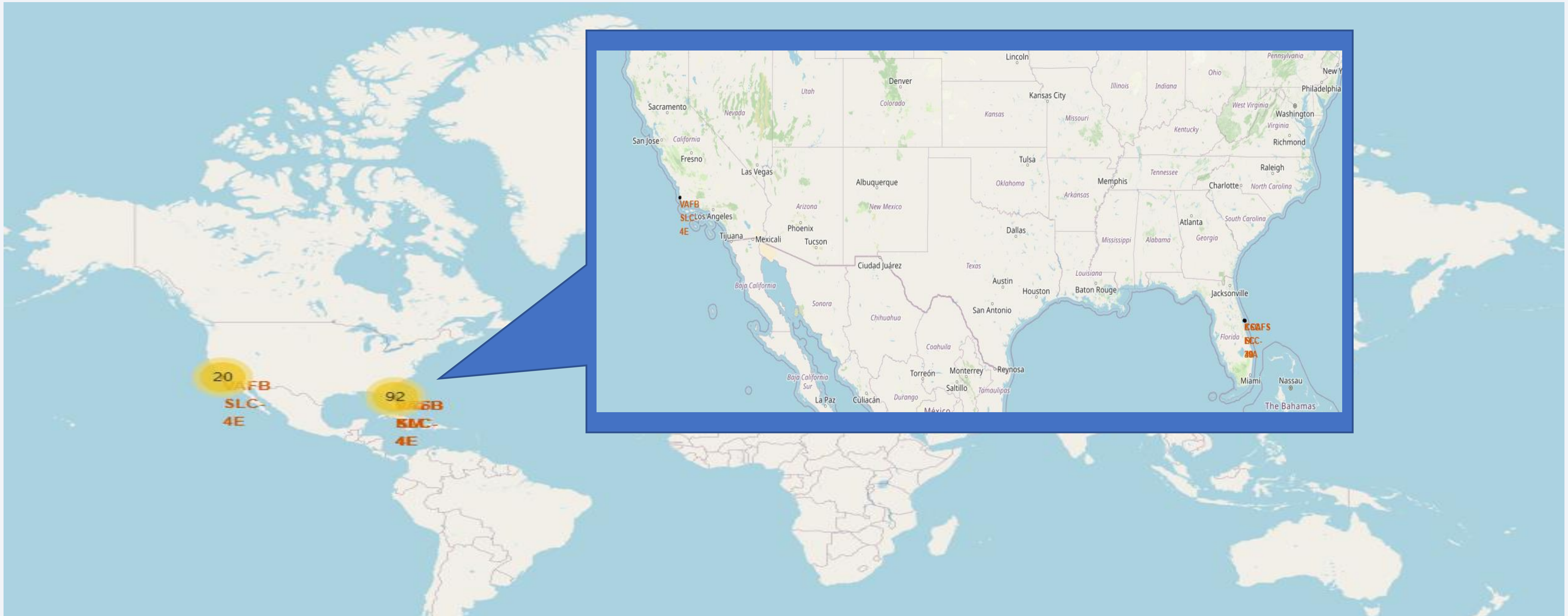
```
 * ibm_db_sa://ygk14970:***@21fecfd8-47b7-4937-840d-d791d0218660.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:31864/bludb
Done.
```

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

Section 3

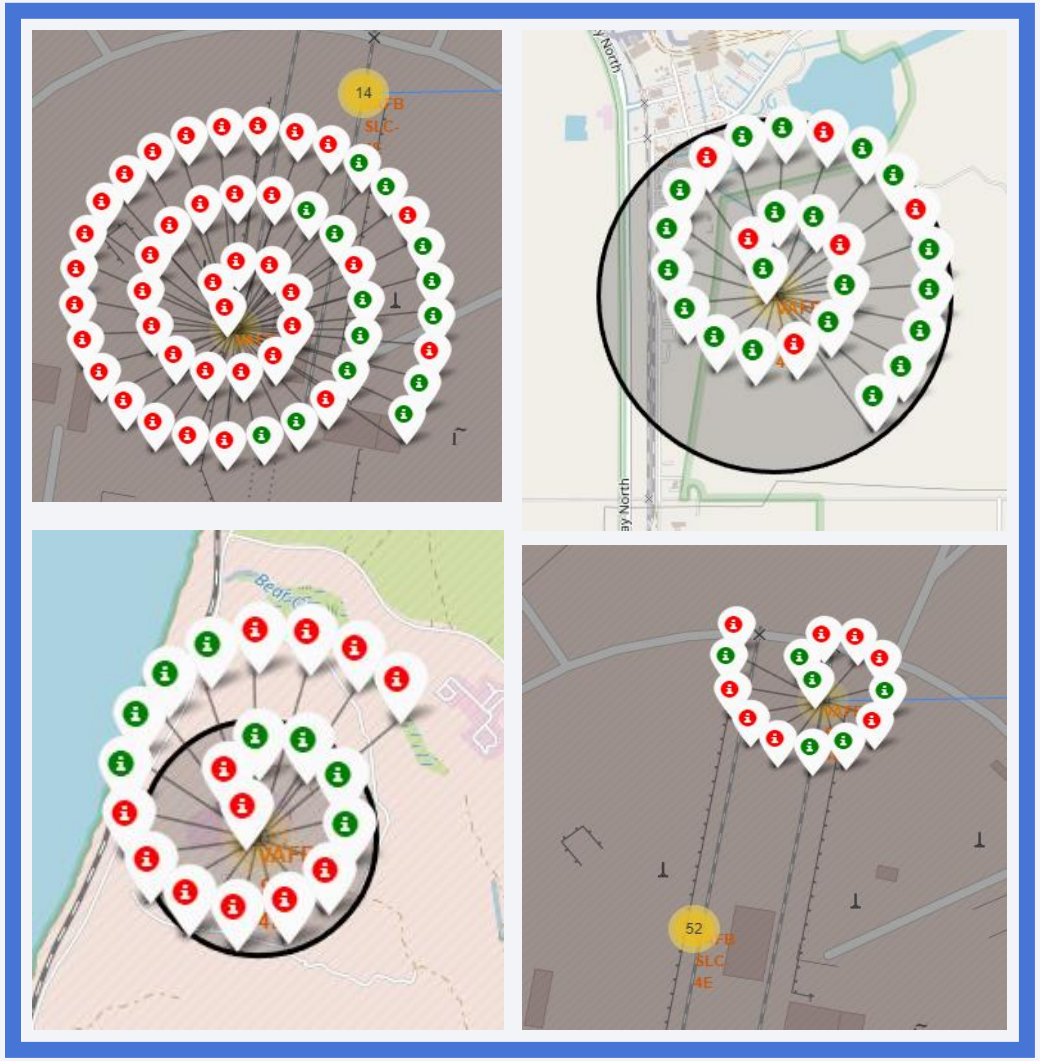
Launch Sites Proximities Analysis

Global map markers – all launch sites



- We can discover that all SpaceX launch sites are basically located in coast of United States

Marker Clusters – launch events



- Marker clusters show every launch at each launch site.

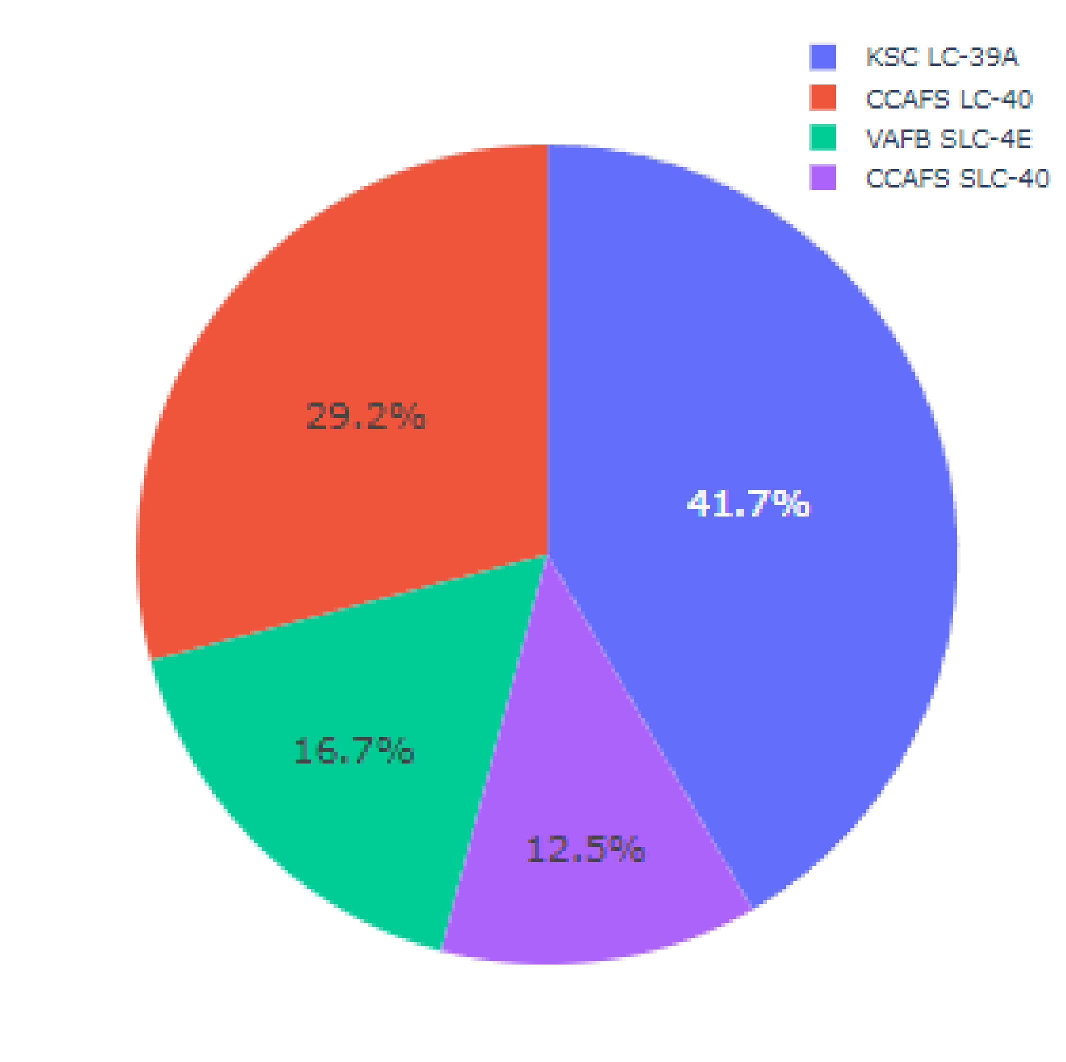
- Green marker: landing success
- Red marker: landing failed

Distance to landmarks

Section 4

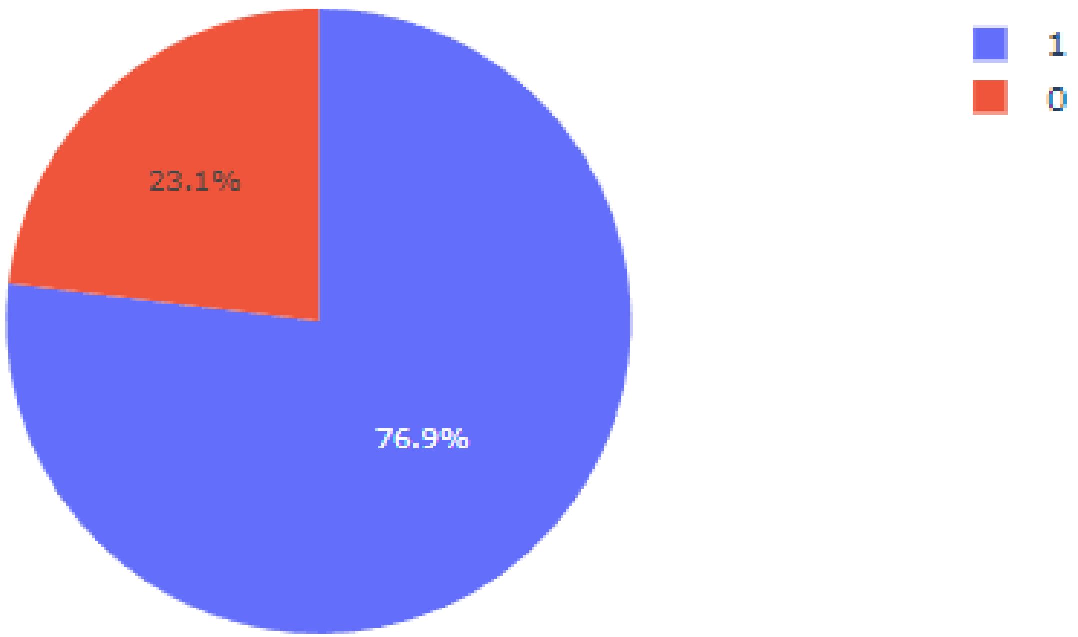
Build a Dashboard with Plotly Dash

Success count for all launch sites



- We can see that launch site KSC LC-39A has the most successful launches.

Total success launches for KSC LC-39A

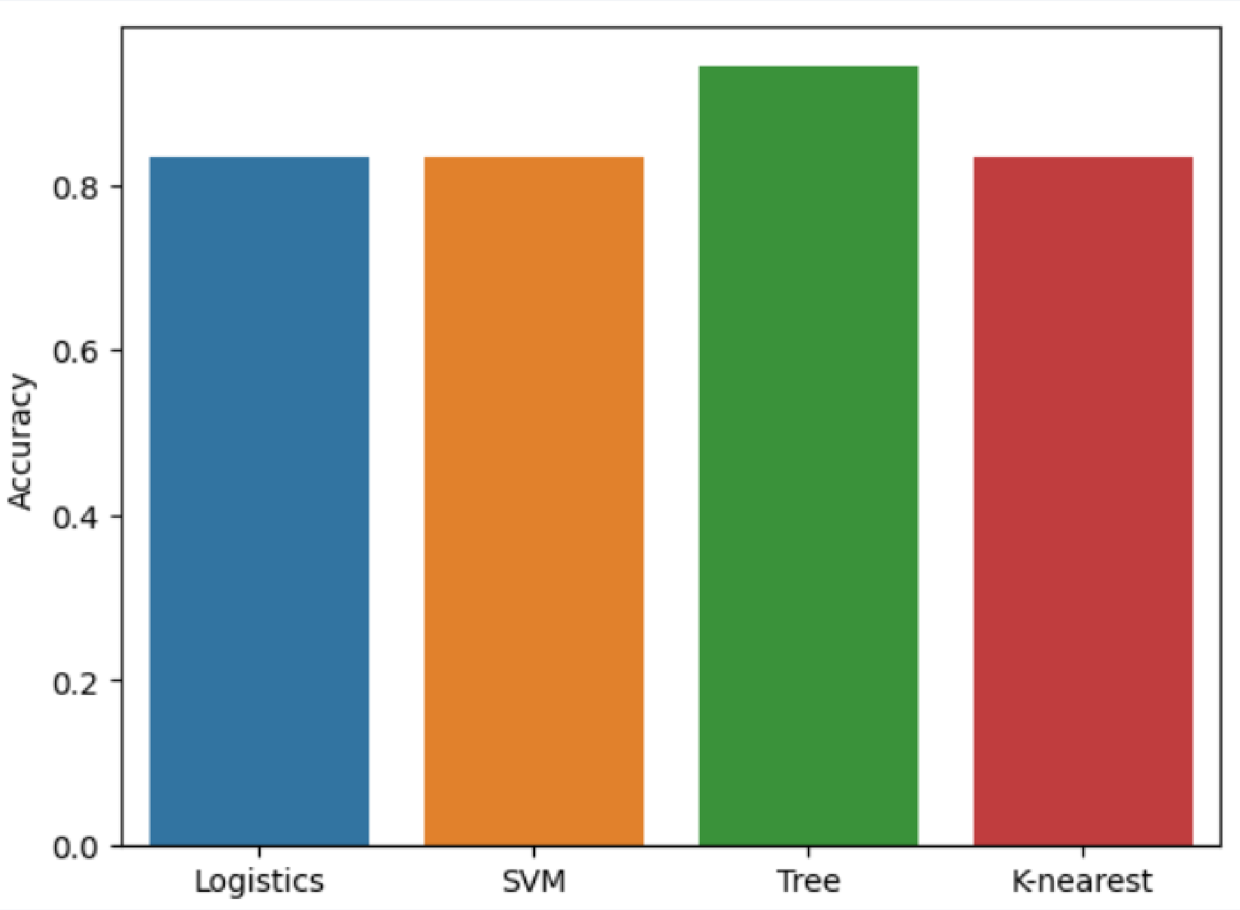


- KSC LC-39A achieved a 76.9% success rate while getting a 23.1% failure rate.

Section 5

Predictive Analysis (Classification)

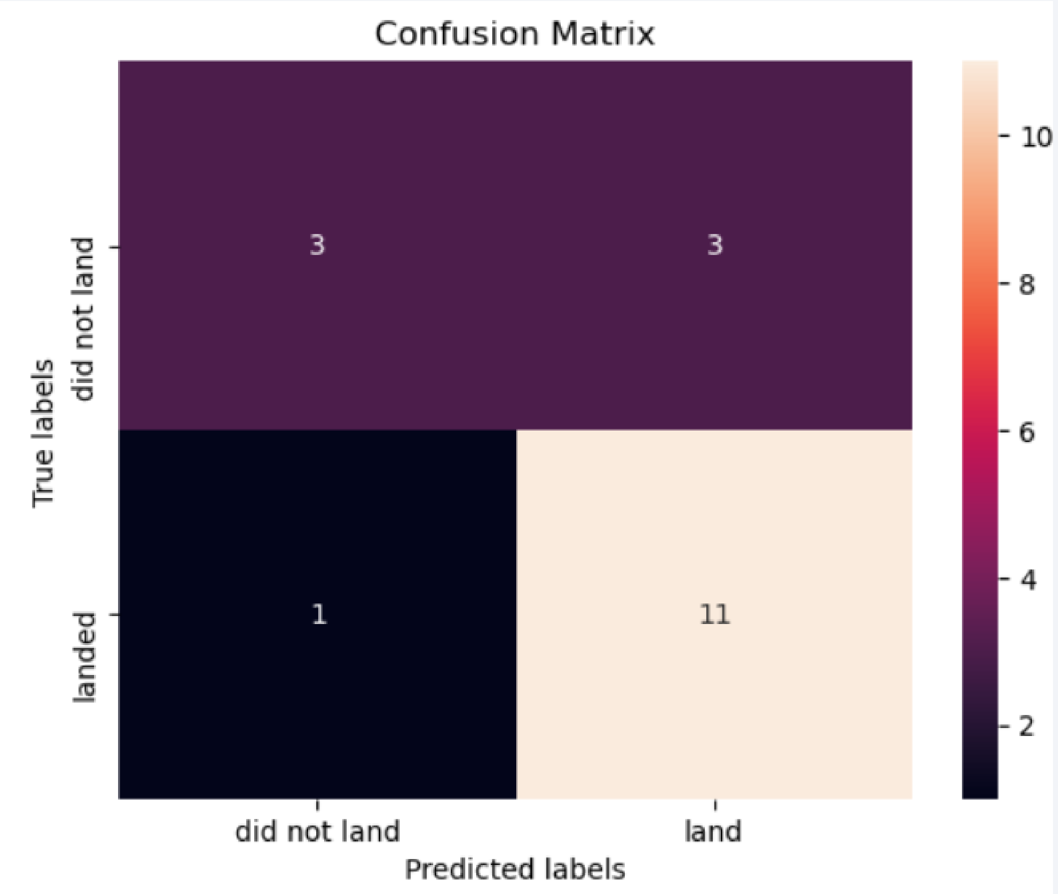
Classification Accuracy



- Applied GridSearchCV to algorithms finding optimized parameters.
- Split the data to be trained and tested (train_test_split function with test size 20%)
- Compared R scores and confusion matrics within algorithms

➢ The result that prediction model with decision tree performed the highest accuracy.

Confusion Matrix – Decision Tree



- The confusion matrix shows that the classifier can distinguish between the different classes.
- The major problem is the false positives .i.e., unsuccessful landing marked as successful landing by the classifier.

Conclusions

We can conclude that:

1. The larger the flight amount at a launch site, the greater the success rate at a launch site.
2. Generally, launch success rate increased in the last decade.
3. Orbits ES-L1, GEO, HEO, SSO had the most success rate.
4. KSC LC-39A had the most successful launches of any sites.
5. The Decision tree classifier is the best machine learning algorithm for this task.

Thank you!