IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Mauricio Campos Molina
September 2024

Outline

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

EXECUTIVE SUMMARY

- **SUMMARY OF METHODOLOGIES**
  - COLLECT DATA GET IN FROM API REQUEST AND WEB SCRAPING
  - STORE DATA INTO CSV FILE.
  - DATA WRANGLIN, AND STORE IN A NEW CSV FILE.
  - EXPLORE DATA, GRAPHING MAIN DATA
  - DRAW LAUNCH SITE ON A MAP TO SHOW SUCCESSFUL LANDING
  - BUILT DYNAMIC INTERFACE TO EXPLORE DATA
  - USE PREDICTIVE ANALYSIS TO BUILD MODELS TO DETERMINE SUCCESSFUL LANDING OUTCOMES
- **SUMMARY OF ALL RESULTS**
  - THE LAST 14 LAUNCH WERE 100% SUCCESSFUL LANDING
  - LAUNCH WITH MORE THAN PAYLOAD THAN 8,000 KG WERE 100% SUCCESSFUL LANDING
  - THE 100% OF LAUNCHES WAS NEAR THE COAST
  - ALL THE PREDICTIVE MODEL HAS HIGHT SUCCESSFUL OUTCOMES

INTRODUCTION

PROJECT BACKGROUND AND CONTEXT

SPACEX IS FIRST COMPANY IN REUSE THE FIRST STAGE IN THEIR ROCKETS AFTER THE LAUNCH, THIS IS THE BIG REASON THAT THE COMPANY SPEND LESS MONEY IN EVERY LAUNCH THAN OTHER COMPANIES.

PROBLEMS YOU WANT TO FIND ANSWERS

DETERMINE IF THE FIRST STAGE WILL LAND, WE CAN DETERMINE THE COST OF A LAUNCH. THIS INFORMATION CAN BE USED IF AN ALTERNATE COMPANY WANTS TO BID AGAINST SPACE X FOR A ROCKET LAUNCH.

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - Request to the SpaceX API
  - Extract a Falcon 9 launch records HTML table from Wikipedia
- Perform data wrangling
  - Exploratory Data Analysis (EDA)
  - Find some patterns in the data and determine what would be the label for training supervised models.
- Perform exploratory data analysis (EDA) using visualization and SQL
  - Execute SQL queries to answer assignment questions
- COMPARE OUTCOME OF PREDICTIVE MODELS

Executive Summary

- Perform interactive visual analytics using Folium and Plotly Dash
- Mark the site in the map where were the launchs
- Explore interative the payload mass relation on the success landing of the first stage
- Perform predictive analysis using classification models
- STANDARDIZE INFORMATION
- SPLIT INTO TRAINING DATA AND TEST DATA
- APPLY DIFERENTS PREDITIVE MODEL
- COMPARE OUTCOME OF PREDICTIVE MODELS

Data Collection

- USE SPACEX API AND WIKIPEDIA TABLE TO REQUEST AND PARSE DATA
- PUT DATA INTO DATA FRAME
- REVIEW THE DATA IN THE DATAFRAME
- CREATE NEW DATA FRAME WHIT IMPORTANT LABELS
- STORE DATA INTO FILE SCV
- CREATE NEW FILE SCV JOIN REVIEW DATA FROM SPACE API AND WIKIPEDIA TABLE.

Data Collection – SpaceX API

- PRESENT YOUR DATA COLLECTION WITH SPACEX REST CALLS USING KEY PHRASES AND FLOWCHARTS

Define helper functions
Extract request use SpaceX API
Store request data into dataframe
Use functions to get specify data
Filter to only include Falcon 9 launches
Dealing with missing values
Export filter data into a csv file.

- GITHUB URL: HTTPS://GITHUB.COM/MAURICIOCAMO/APPLIED-DATA-SCIENCE-CAPSTONE/BLOB/MAIN/WEEK%20%231%2C%202-4%20JUPYTER-LABS-SPACEX-DATA-COLLECTION-API.IPYNB

	FlightNumber	Date	BoosterVersion	PayloadMass	Orbit	LaunchSite	Outcome	Flights	GridFins	Reused	Legs	LandingPad	Block	ReusedCount	Serial	Longitude	Latitude
4	6	2010-06-04	Falcon 9	NaN	LEO	CCSFS SLC 40	None None	1	False	False	False	None	1.0	0	B0003	-80.577366	28.561857
5	8	2012-05-22	Falcon 9	525.0	LEO	CCSFS SLC 40	None None	1	False	False	False	None	1.0	0	B0005	-80.577366	28.561857
6	10	2013-03-01	Falcon 9	677.0	ISS	CCSFS SLC 40	None None	1	False	False	False	None	1.0	0	B0007	-80.577366	28.561857
7	11	2013-09-29	Falcon 9	500.0	PO	VAFB SLC 4E	False Ocean	1	False	False	False	None	1.0	0	B1003	-120.610829	34.632093

Data Collection - Scraping

PRESENT YOUR WEB SCRAPING PROCESS USING KEY PHRASES AND FLOWCHARTS

Write helper functions to process web scraped HTML table

Perform an HTTP GET method to request the Falcon9 Launch HTML page

Extract all column/variable names from the HTML table

Create a data frame by parsing the launch HTML table

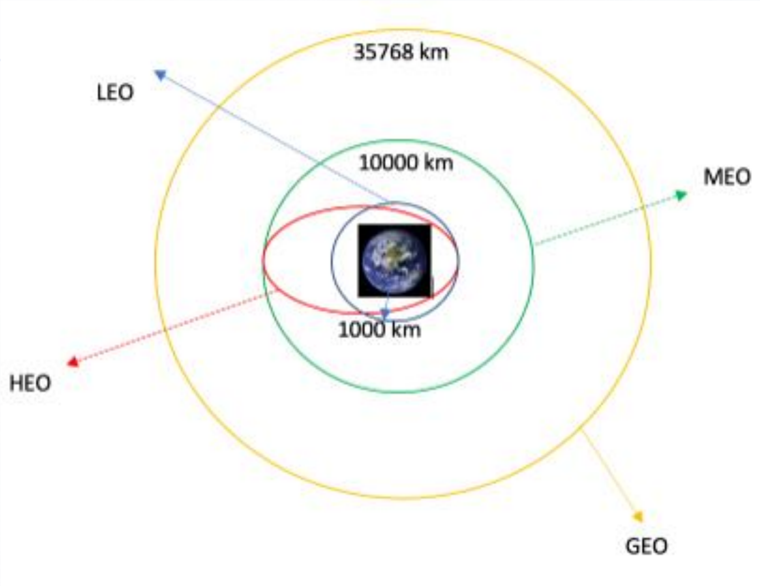
- GITHUB URL: HTTPS://GITHUB.COM/MAURICIOCAMO/APPLIED-DATA-SCIENCE-CAPSTONE/BLOB/MAIN/WEEK%20%231%2C%202-5%20JUPYTER-LABS-WEBSCRAPING.IPYNB

	Flight No.	Launch site	Payload	Payload mass	Orbit	Customer	Launch outcome	Version Booster	Booster landing	Date	Time
0	1	CCSFS	Transporter-1	~5,000 kg	SSO	SpaceX	Success\n	F9 B5B1058.5613	Failure	4 June 2010	15:00
1	2	CCAFS	Dragon Spacecraft Qualification Unit	0	LEO	NASA	Success	F9 v1.07B0003.18	Failure	8 December 2010	18:45
2	3	CCAFS	Dragon	0	LEO	NASA	Success	F9 v1.07B0004.18	No attempt\n	22 May 2012	15:43

Data Wrangling

DESCRIBE PROCESSES

Load data from store in cvs file into dataframe	○df=pd.read_csv("dataset_part_1.csv")
Identify which columns are numerical and categorical	○df.dtypes
Calculate the number of launches on each site	○df.value_counts("LaunchSite")
Calculate the number and occurrence of each orbit	○df.value_counts("Orbit")
Calculate the number and occurence of mission outcome of the orbits	○landing_outcomes = df.Outcome.value_counts()
Create a landing outcome label from Outcome column	○0 is bad outcome, 1 is successful outcome
Save data into scv file	• df.to_csv("dataset_part_2.csv", index=False)



- GITHUB URL: HTTPS://GITHUB.COM/MAURICIOCAMO/APPLIED-DATA-SCIENCE-CAPSTONE/BLOB/MAIN/WEEK%201%2C%203-2%20LABS-JUPYTER-SPACEX-DATA%20WRANGLING.IPYNB

	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	6123.547647	LEO	CCSFS SLC 40	None None	1	False	False	False	6123.547647058824	1.0	0	B0003	-80.577366	28.561857	0
1	2	2012-05-22	Falcon 9	525.000000	LEO	CCSFS SLC 40	None None	1	False	False	False	6123.547647058824	1.0	0	B0005	-80.577366	28.561857	0
2	3	2013-03-01	Falcon 9	677.000000	ISS	CCSFS SLC 40	None None	1	False	False	False	6123.547647058824	1.0	0	B0007	-80.577366	28.561857	0

EDA with Data Visualization

ScatterPlot

- Visualize the relationship betwen FlightNumber vs PayloadMass to successful landing
- Visualize the relationship betwen LaunchSite vs PayloadMass to successful landing
- **Visualize the relationship between Payload and Launch Site to**successful landing
- **Visualize the relationship between FlightNumber and Orbit type** to successful landing
- **Visualize the relationship between Payload and Orbit type** to successful landing

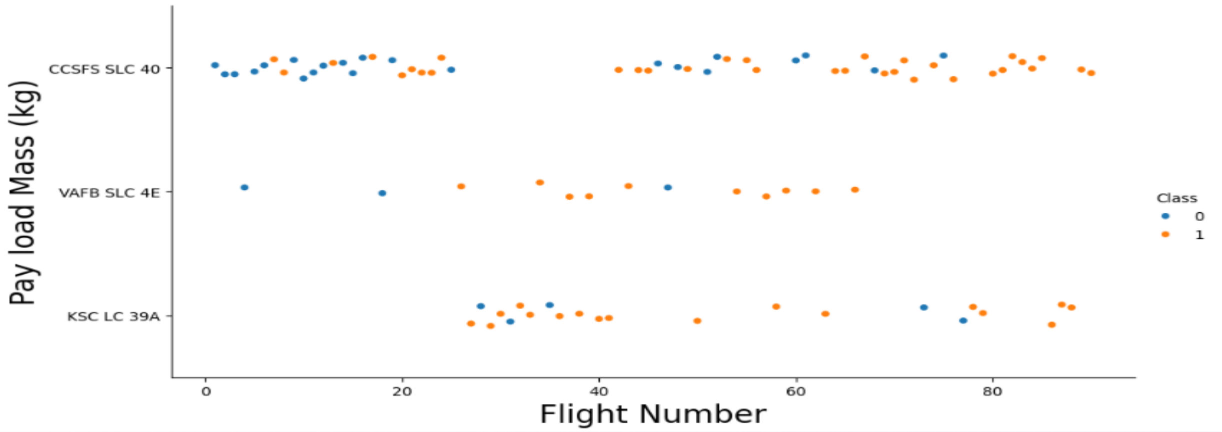
BarPlot

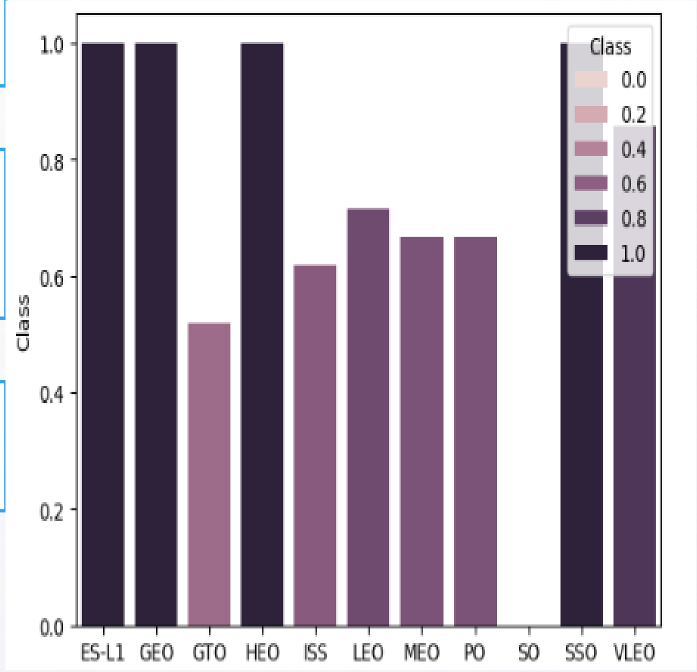
- **Visualize the relationship between success rate of each orbit type**

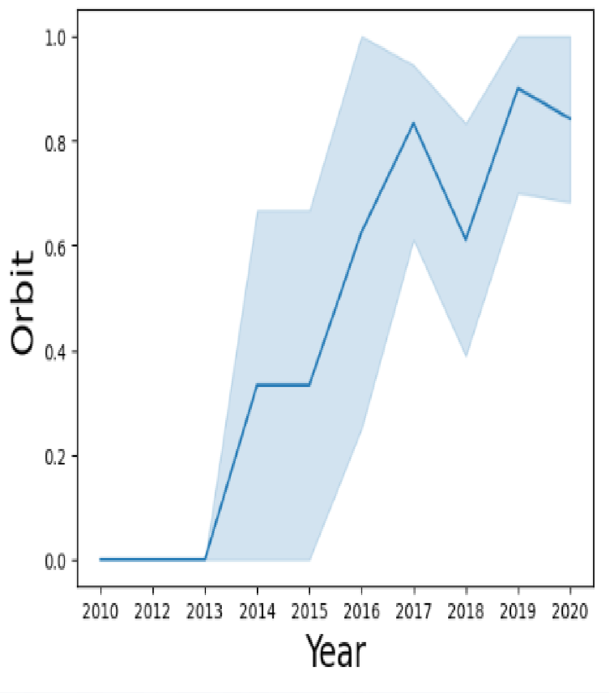
LinePlot

- **Visualize the launch success yearly trend**

- GITHUB URL: HTTPS://GITHUB.COM/MAURICIOCAMO/APPLIED-DATA-SCIENCE-CAPSTONE/BLOB/MAIN/WEEK%202%2C%202-1-JUPYTER-LABS-EDA-DATAVIZ.IPYNB







EDA with SQL

- RETRIEVE THE QUERY RESULTS INTO A PANDAS DATAFRAME
- DISPLAY 5 RECORDS WHERE LAUNCH SITES BEGIN WITH THE STRING 'CCA'
- DISPLAY THE TOTAL PAYLOAD MASS CARRIED BY BOOSTERS LAUNCHED BY NASA (CRS
- DISPLAY AVERAGE PAYLOAD MASS CARRIED BY BOOSTER VERSION F9 V1.1
- LIST THE DATE WHEN THE FIRST SUCCESFUL LANDING OUTCOME IN GROUND PAD WAS ACHEIVED
- LIST THE NAMES OF THE BOOSTERS WHICH HAVE SUCCESS IN DRONE SHIP AND HAVE PAYLOAD MASS GREATER THAN 4000 BUT LESS THAN 6000
- LIST THE TOTAL NUMBER OF SUCCESSFUL AND FAILURE MISSION OUTCOMES
- LIST THE NAMES OF THE BOOSTER_VERSIONS WHICH HAVE CARRIED THE MAXIMUM PAYLOAD MASS
- LIST THE RECORDS WHICH WILL DISPLAY THE MONTH NAMES, FAILURE LANDING_OUTCOMES IN DRONE SHIP ,BOOSTER VERSIONS, LAUNCH_SITE FOR THE MONTHS IN YEAR 2015
- RANK THE COUNT OF LANDING OUTCOMES (SUCH AS FAILURE (DRONE SHIP) OR SUCCESS (GROUND PAD)) BETWEEN THE DATE 2010-06-04 AND 2017-03-20, IN DESCENDING ORDER

- GITHUB URL:HTTPS://GITHUB.COM/MAURICIOCAMO/APPLIED-DATA-SCIENCE-CAPSTONE/BLOB/MAIN/WEEK%20%232%2C%201-2%20JUPYTER-LABS-EDA-SQL-COURSERA_SQLLITE.IPYNB

Build an Interactive Map with Folium

Circle

- Visualize the launch sites coordinates locations on a map.

Markers

- Use green markers for viazulize if launch was successful or use red marker if was failed

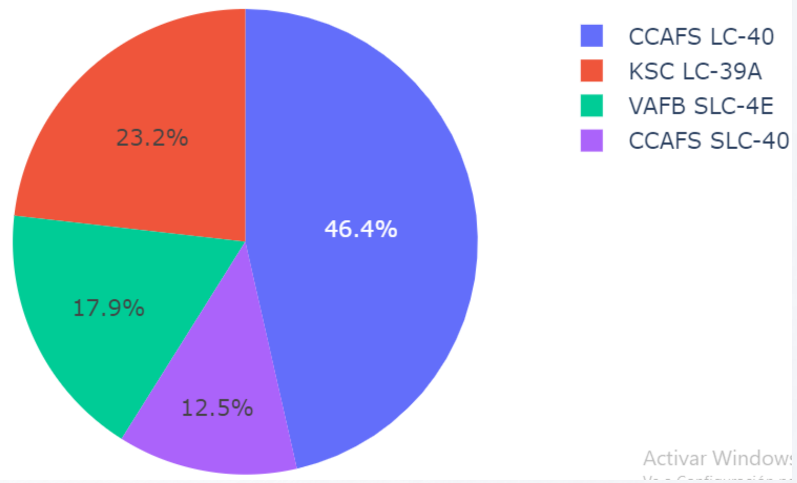
PolyLine

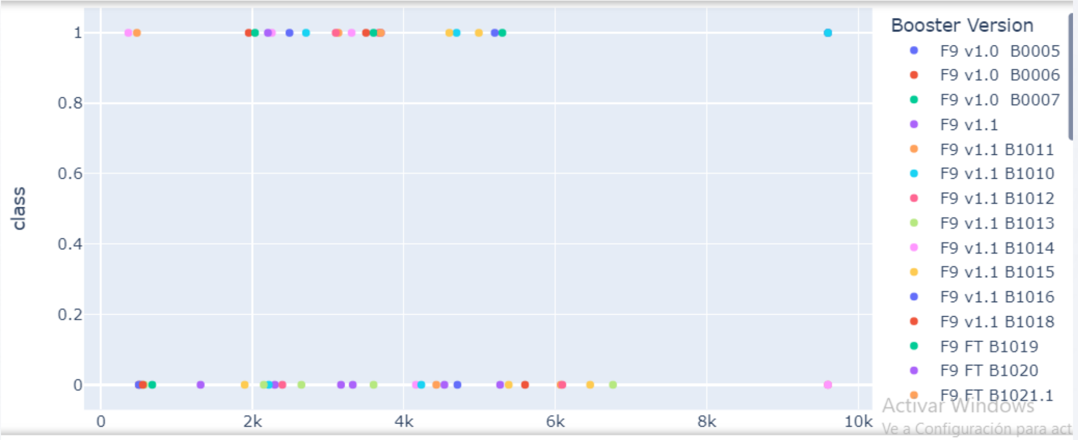
- Visualize distance betwen launch site to other point on the map.

- GITHUB URLHTTPS://GITHUB.COM/MAURICIOCAMO/APPLIED-DATA-SCIENCE-CAPSTONE/BLOB/MAIN/WEEK%203%2C%201-2%20LAB_JUPYTER_LAUNCH_SITE_LOCATION.IPYNB

Build a Dashboard with Plotly Dash

DropDown	○ Select differente Launch Site
RangeSlider	○ Select different payload range
Graph Pie Chart	○ Visualize launch success counts
Graph Scatter Chart	○ Visualize how payload may be correlated with mission outcomes for selected site(s).





- GITHUB URL: HTTPS://GITHUB.COM/MAURICIOCAMO/APPLIED-DATA-SCIENCE-CAPSTONE/BLOB/MAIN/WEEK%203%2C%201-2%20BUILD_A_DASHBOARD_APPLICATION_WITH_PLOTLY_DASH.IPYNB

Predictive Analysis (Classification)

SUMMARIZE

PREDICTIVE ANALYSIS TECHNIQUES USE THE DATA THAT WAS STORE IN IN PREVIOUS STAGES TO DEVELOP DIFERENT MODELS THAN CAN PREDICT SUCCESSFUL LANDING OF THE FIRST STAGE. IN THIS PROCESS SPLIT THE DATA INTO:

- TRAINING DATA: USED TO TRAINING OUR MODEL
- TEST DATA: USED TO VALIDATE ACCURACY OUR DEVELOPED MODEL

THESE MODELS HAVE GREAT VALUE, BECAUSE WE CAN PREDICT THE SUCCESSFUL RATE OF FUTURE LAUCHES

- GITHUB URL: HTTPS://GITHUB.COM/MAURICIOCAMO/APPLIED-DATA-SCIENCE-CAPSTONE/BLOB/MAIN/WEEK%203%2C%201-2%20BUILD_A_DASHBOARD_APPLICATION_WITH_PLOTLY_DASH.IPYNB

PROCESS

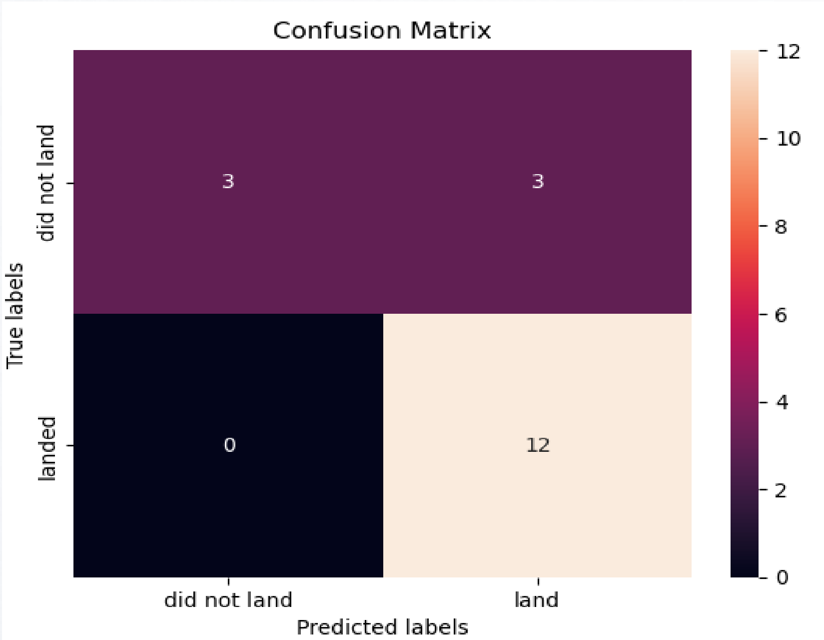
- LOAD THE DATA
- CREATE A NUMPY ARRAY FROM THE COLUMN CLASS IN DATA
- STANDARDIZE THE DATA
- SPLIT THE DATA INTO TRAINING AND TESTING DATA
- USE GRIDSEARCHCV TO FIND THE BEST PARAMETERS TO
  - LOGISTIC REGRESION MODEL
    - CALCULATE THE ACCURACY ON THE TEST DATA
    - DRAW THE CONFUSION MATRIX GRAPH
  - VECTOR MACHINE MODEL
    - CALCULATE THE ACCURACY ON THE TEST DATA
    - DRAW THE CONFUSION MATRIX GRAPH
  - DECISION TREE CLASSIFIER MODEL
    - CALCULATE THE ACCURACY ON THE TEST DATA
    - DRAW THE CONFUSION MATRIX GRAPH
  - K NEAREST NEIGHBORS MODEL
    - CALCULATE THE ACCURACY ON THE TEST DATA
    - DRAW THE CONFUSION MATRIX GRAPH
  - DECISION TREE CLASSIFIER MODEL
    - CALCULATE THE ACCURACY ON THE TEST DATA
    - DRAW THE CONFUSION MATRIX GRAPH
- COMPARE THE ACURRACY OF THE DIFFERENT MODELS USED

Results

Exploratory data analysis results

	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	28.561857	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	28.561857	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	28.561857	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	34.632093	0

Interactive analytics demo in screenshots



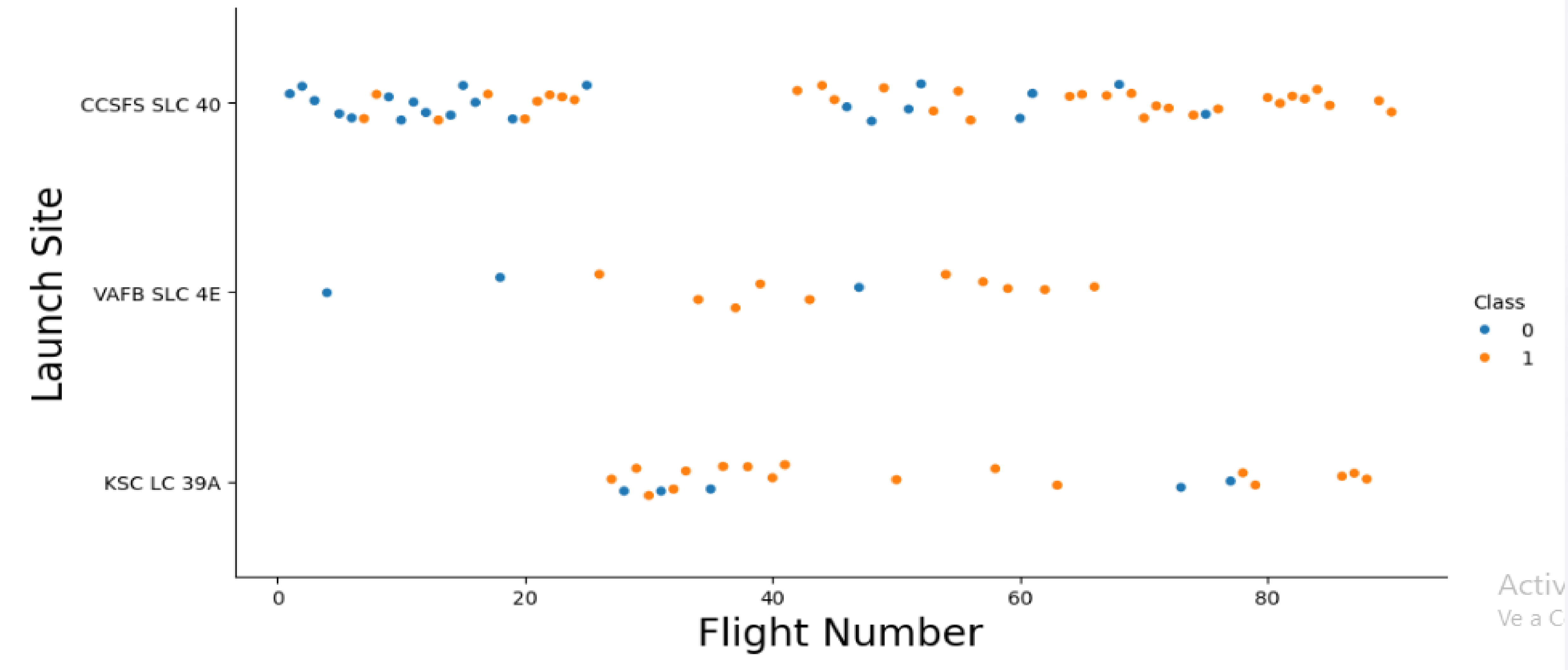
Predictive analysis results

Logistic Regresion : 0.8333333333333334
vector-machine : 0.8333333333333334
Trees : 0.8333333333333334
K-nearest : 0.8333333333333334

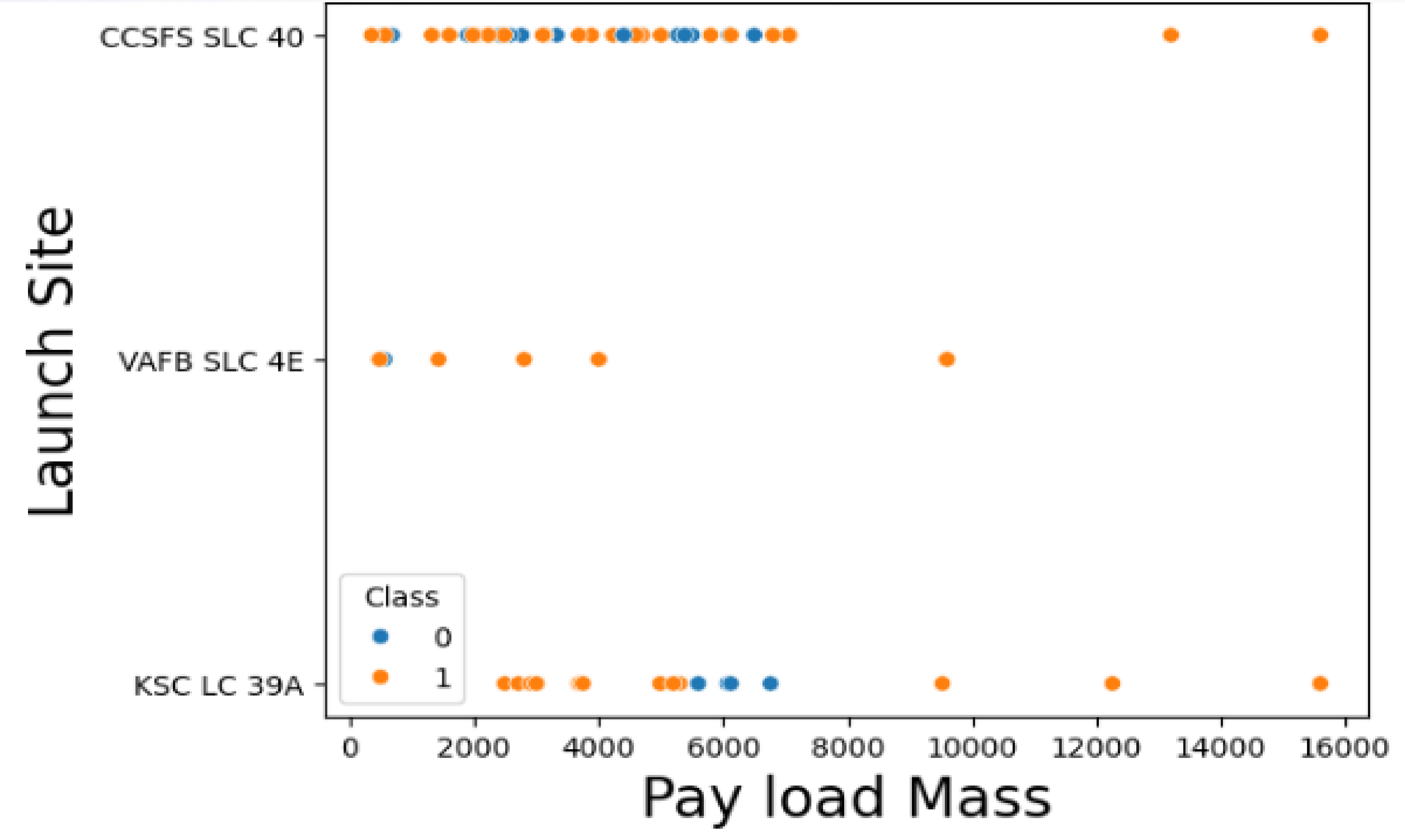
Section 2

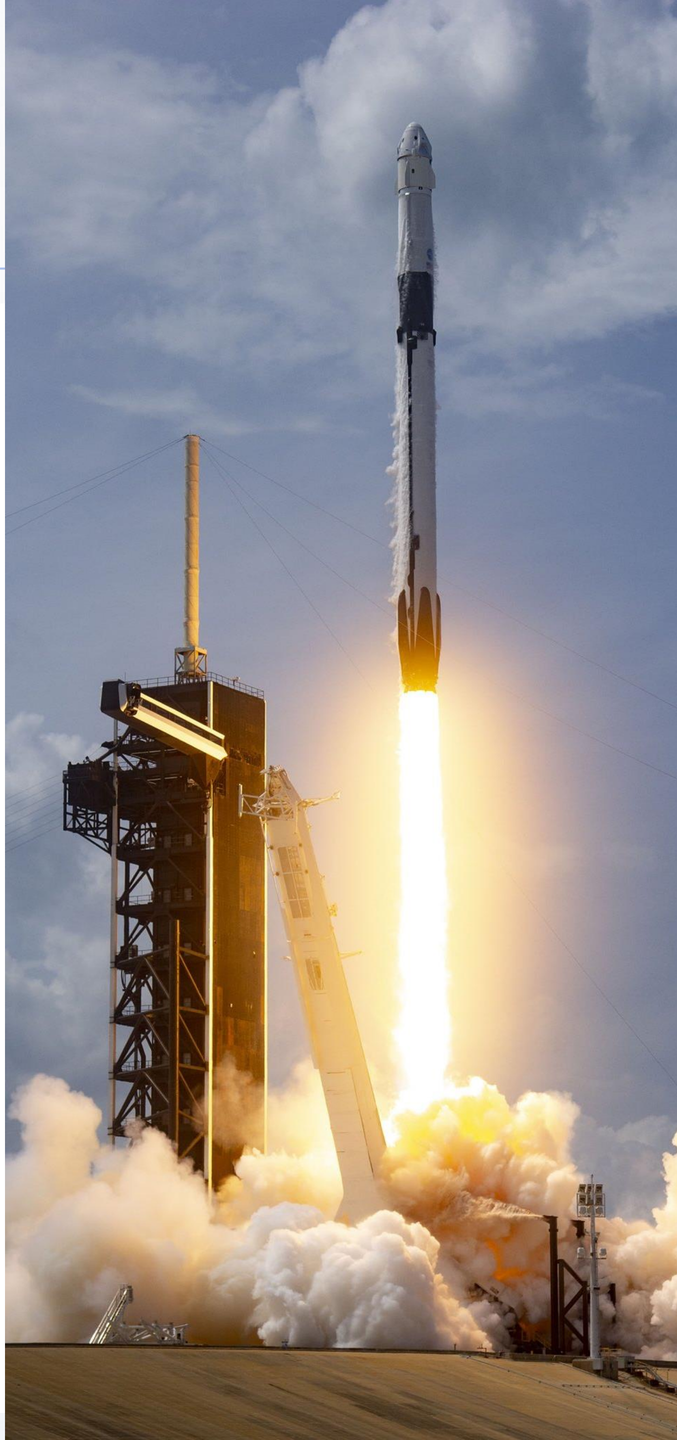
Insights drawn from EDA

Flight Number vs. Launch Site

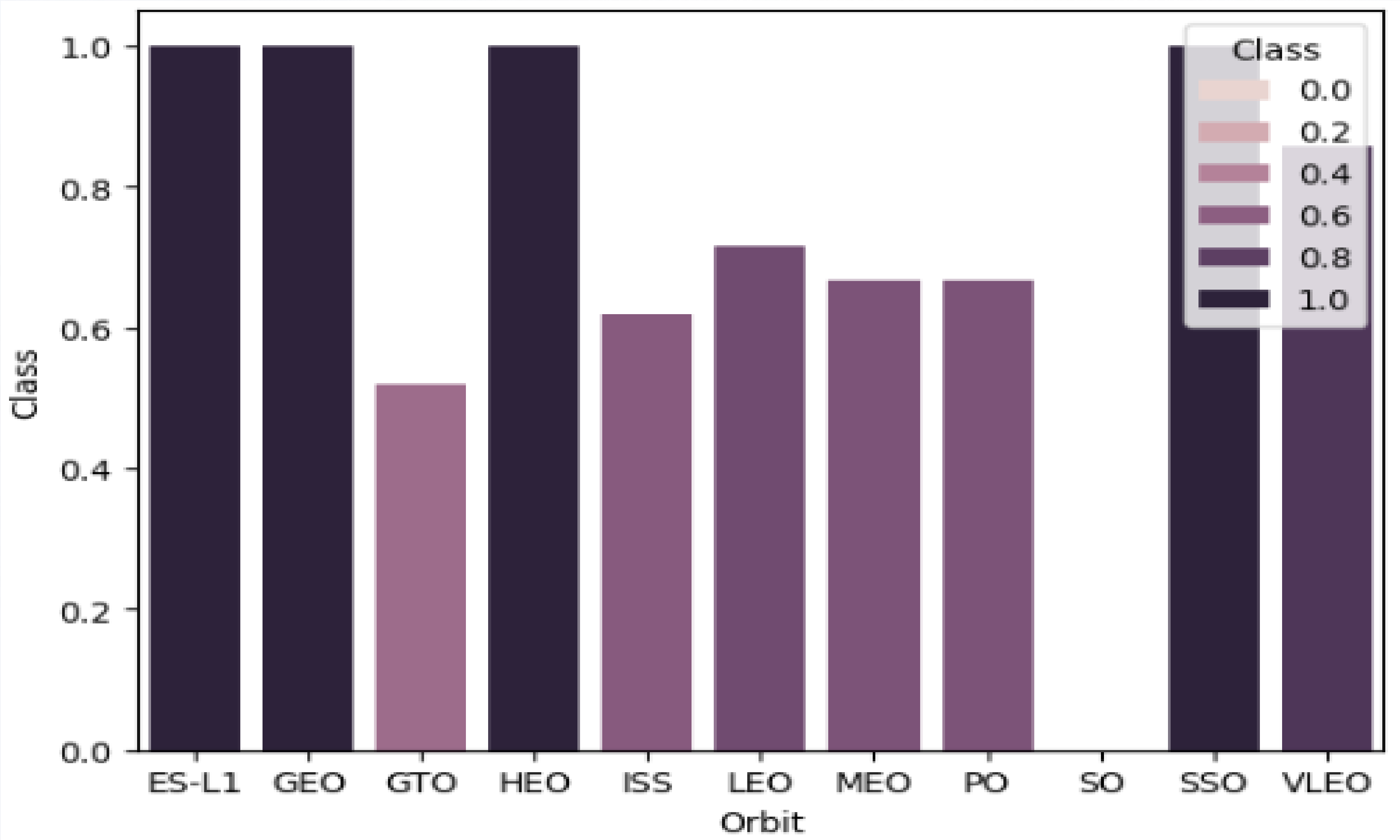


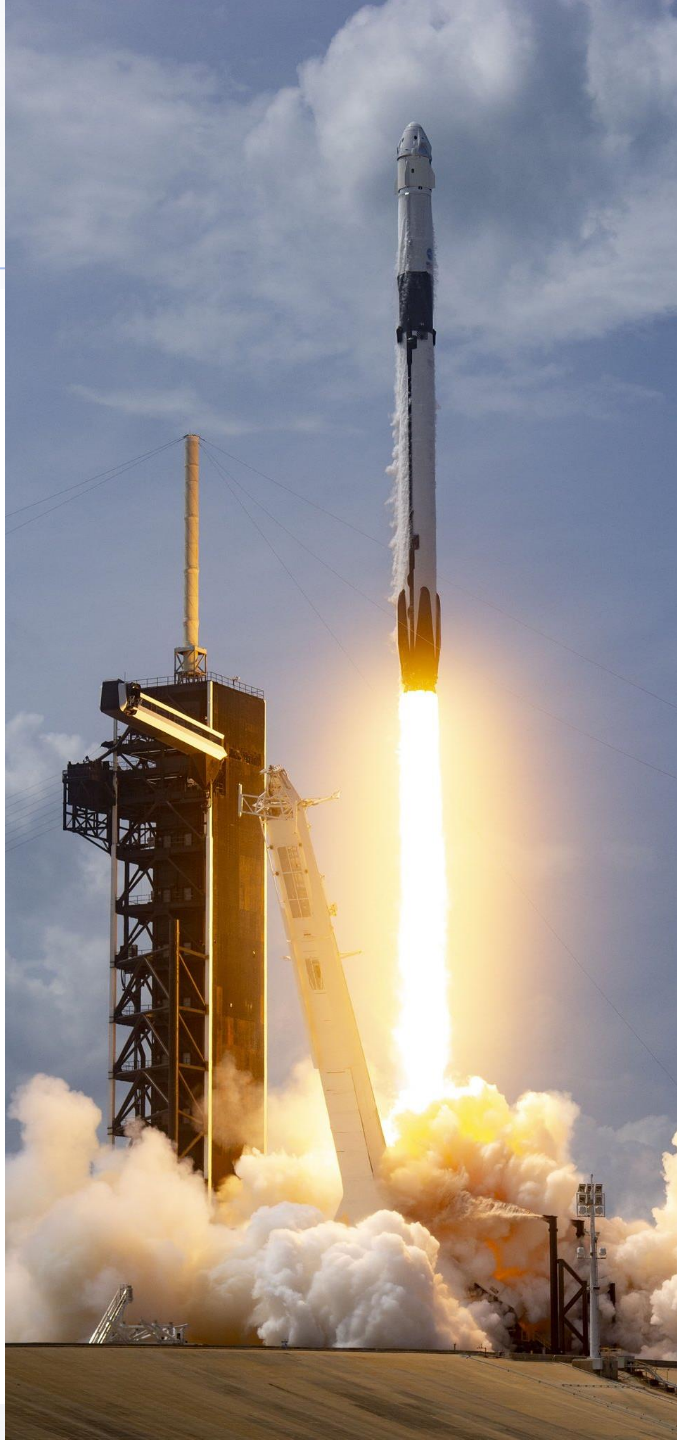
Payload vs. Launch Site



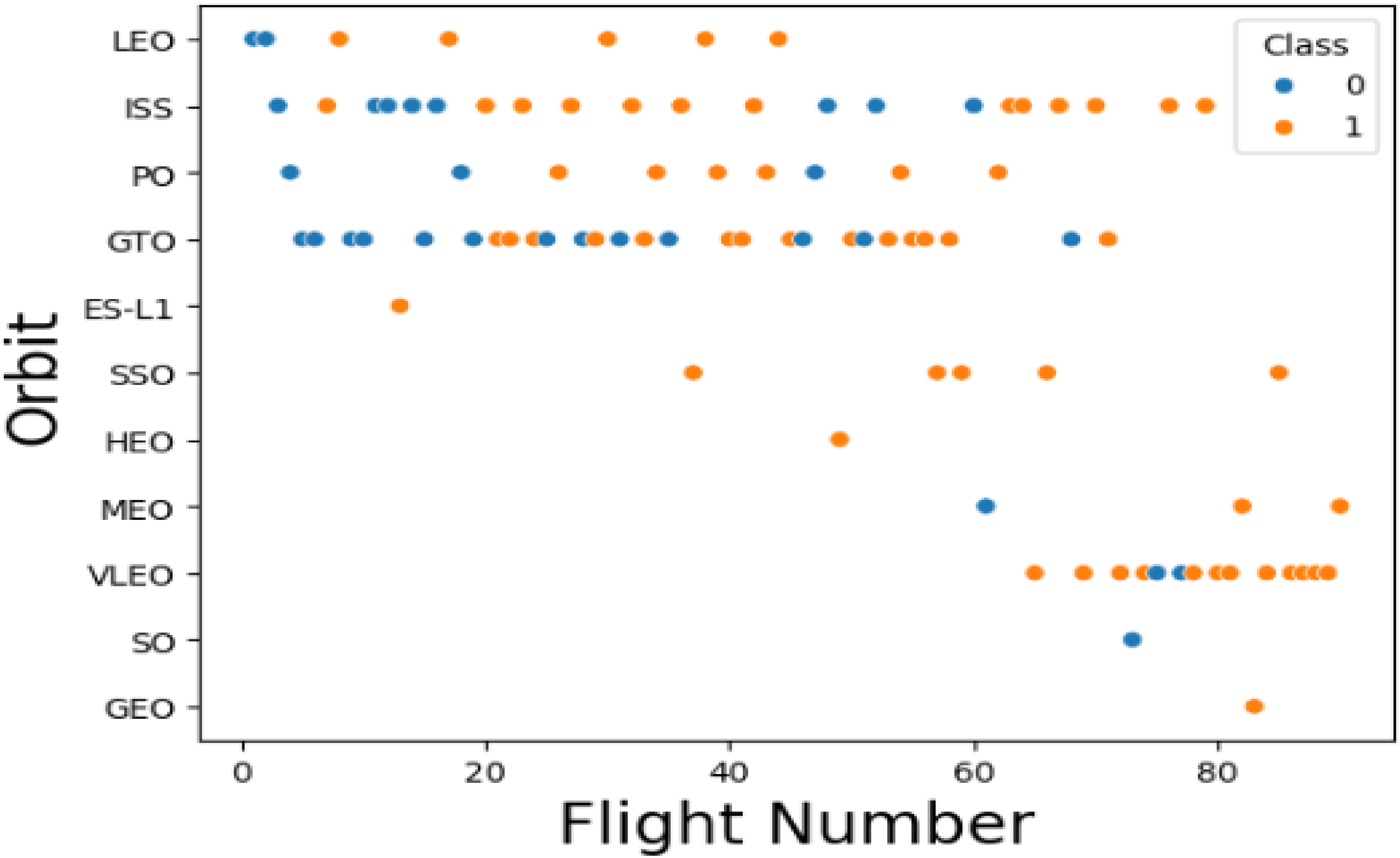


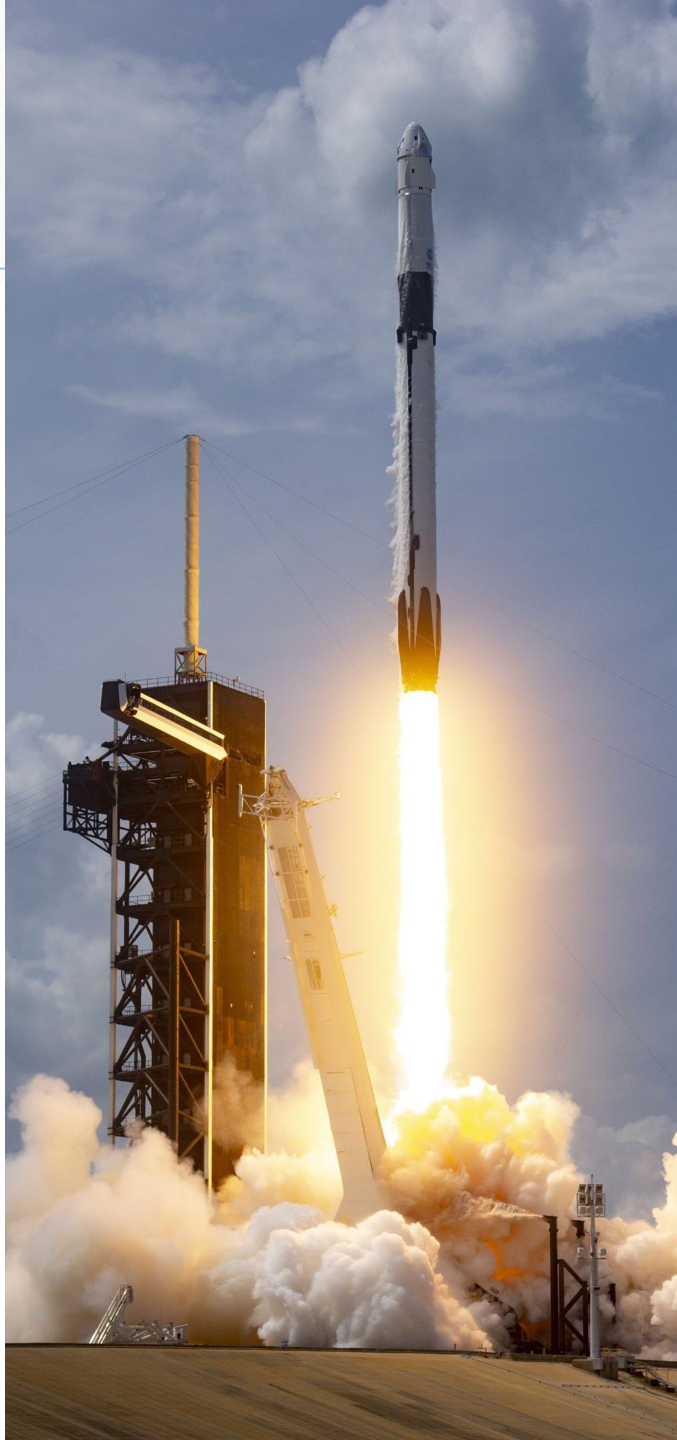
Success Rate vs. Orbit Type



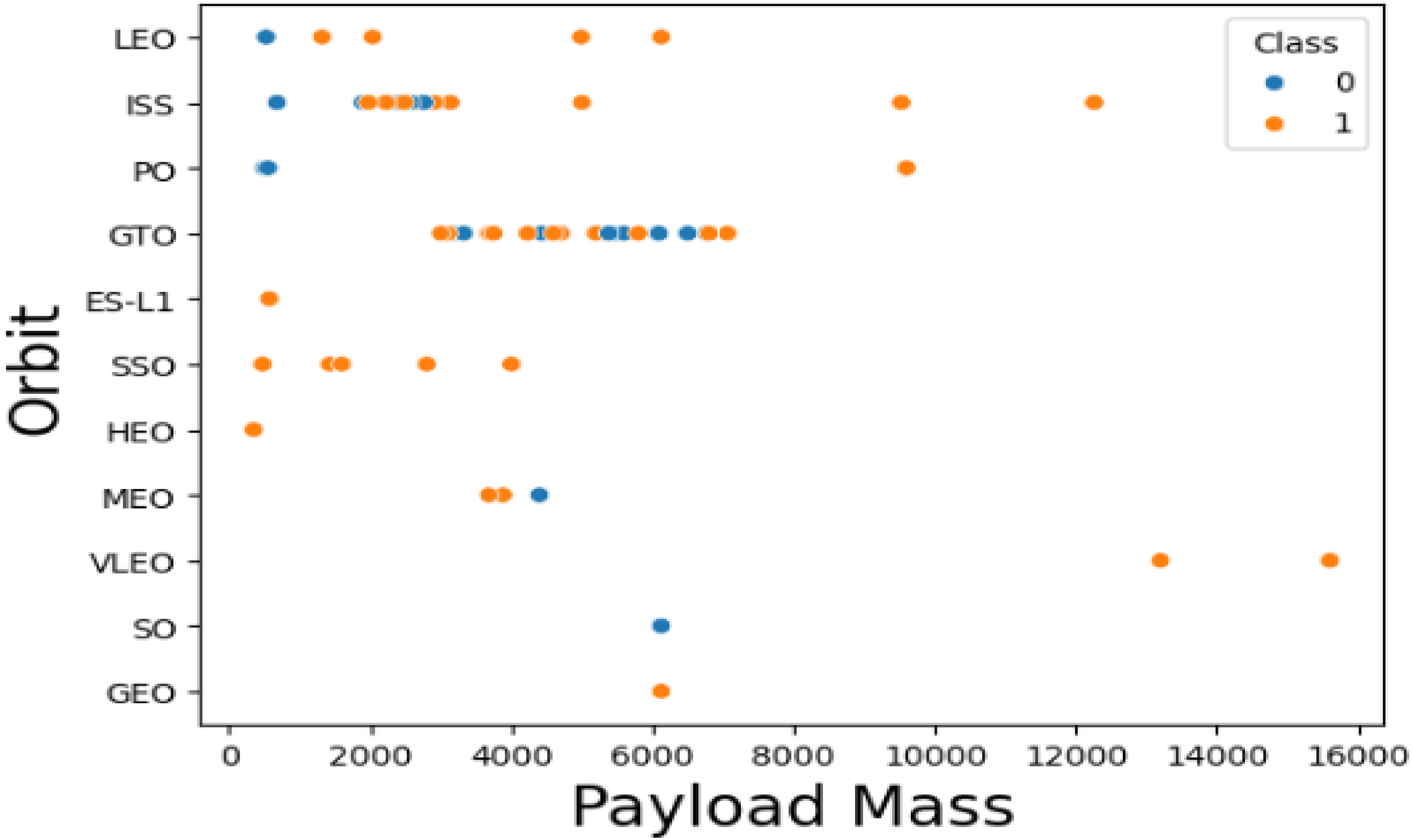


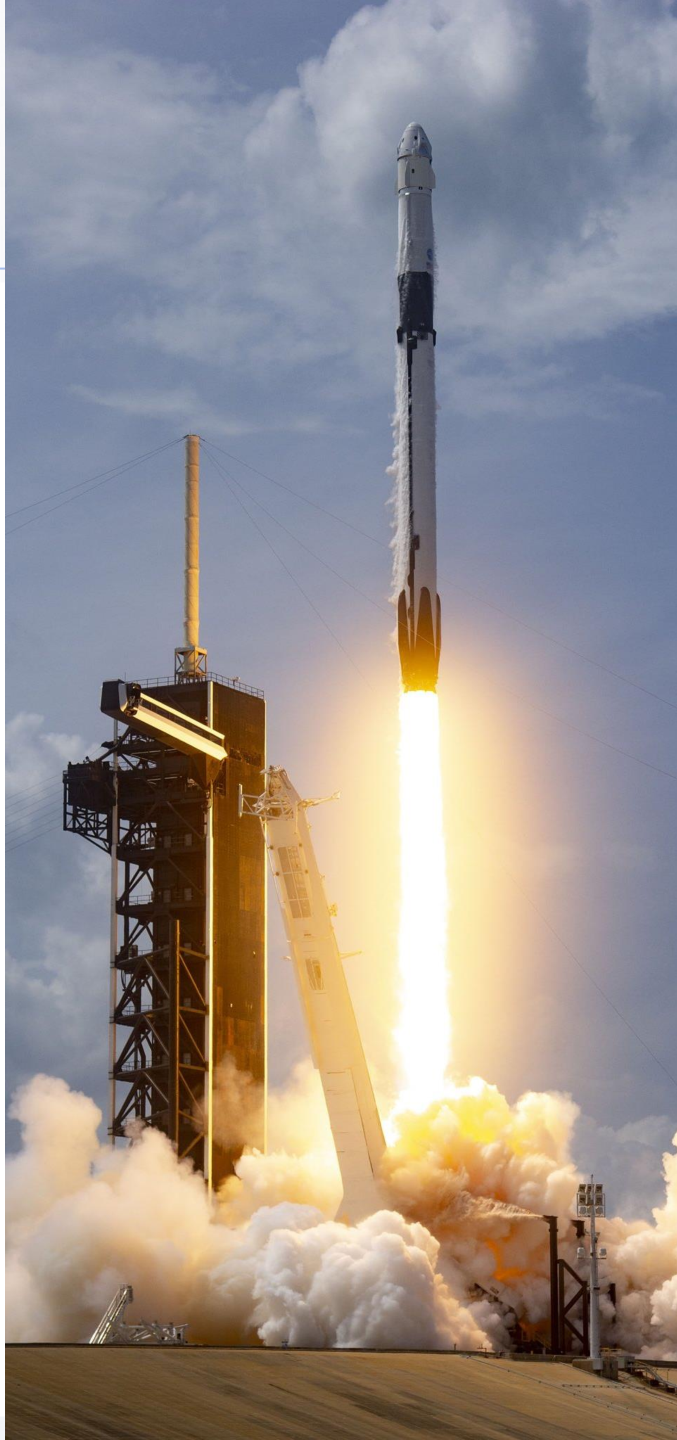
Flight Number vs. Orbit Type



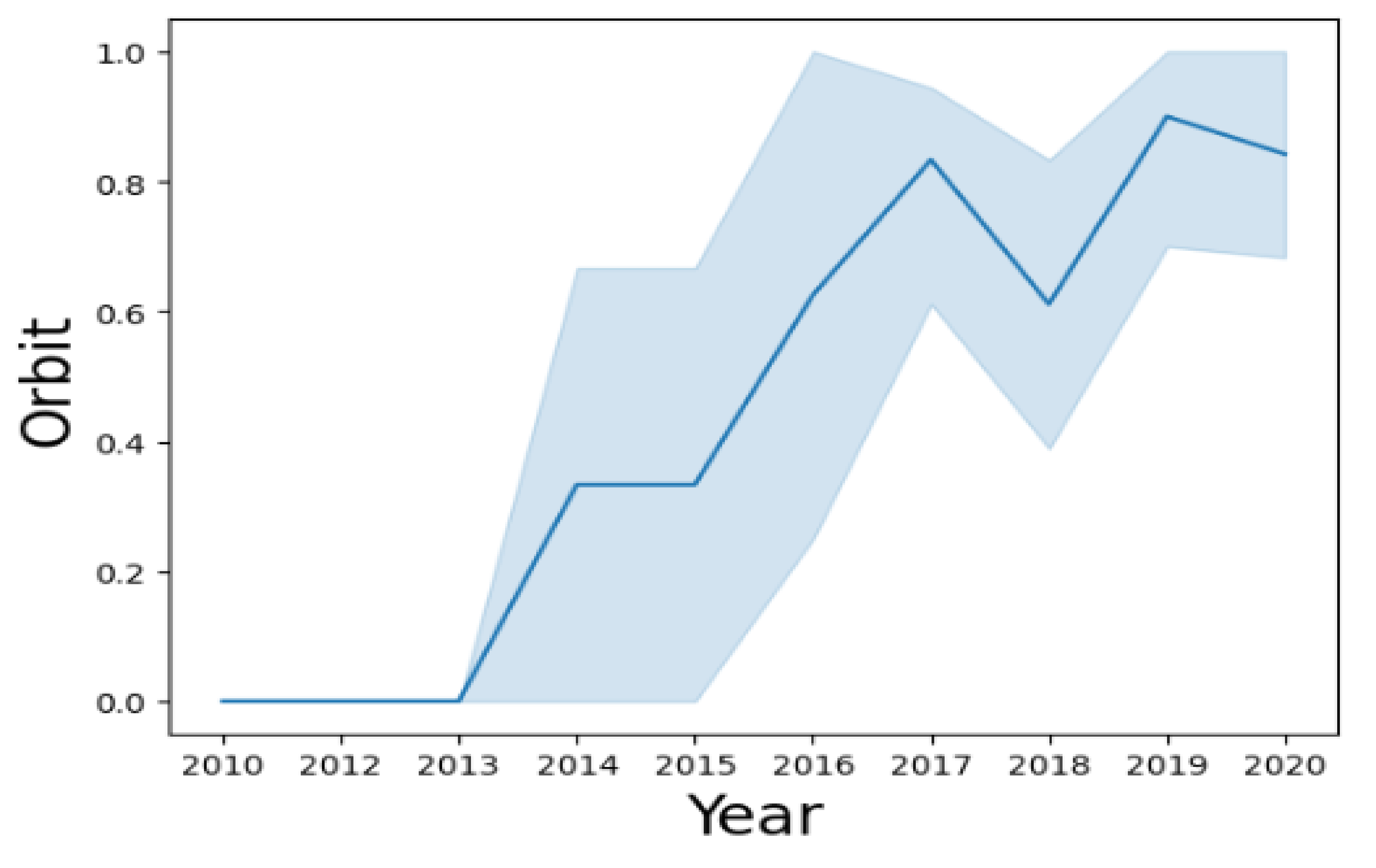


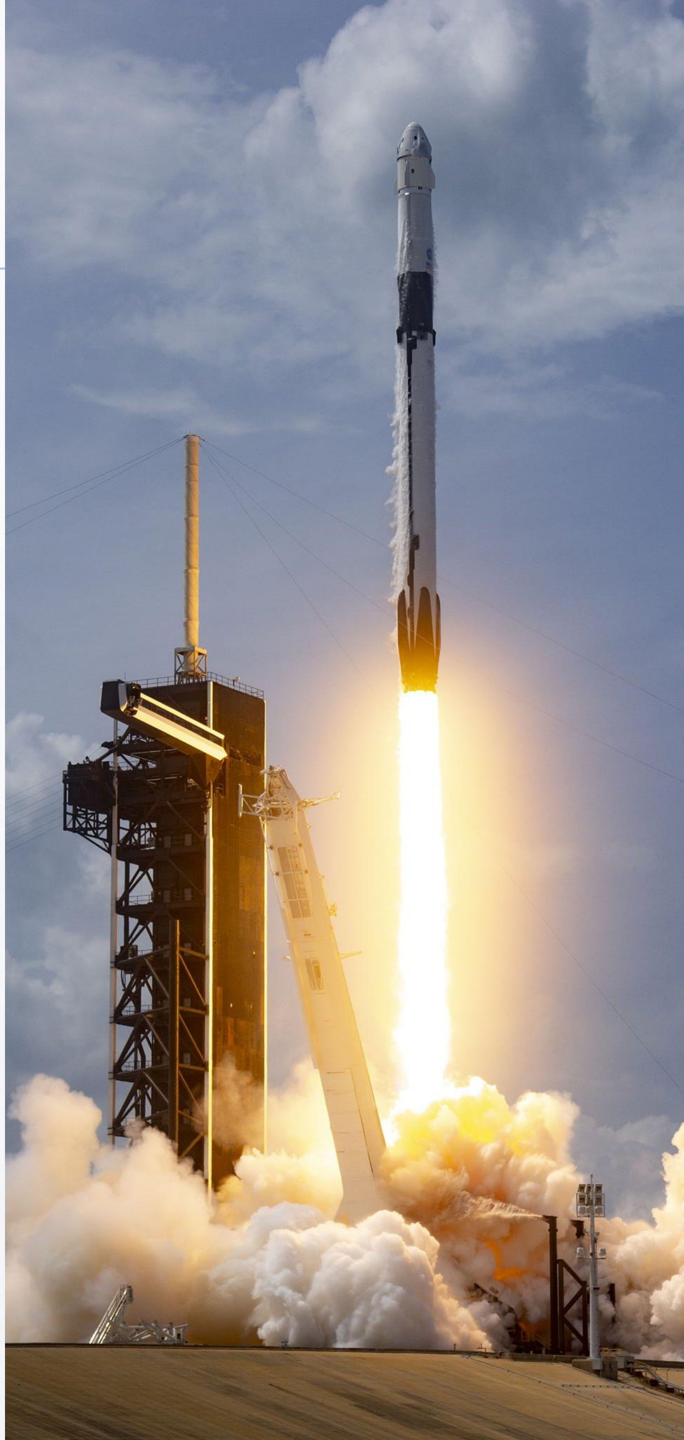
Payload vs. Orbit Type





Launch Success Yearly Trend





All Launch Site Names

THIS IS THE LIST OF LAUNCH SITE USED BY SPACE X

```
[5]: import pandas as pd
     #retrieve the query results into a pandas dataframe
     df = pd.read_sql_query("select DISTINCT(Launch_Site) from SPACEXTBL ", con)
     df
```

[5]:

	Launch_Site
0	CCAFS LC-40
1	VAFB SLC-4E
2	KSC LC-39A
3	CCAFS SLC-40

Launch Site Names Begin with 'CCA'

5 RECORDS WHERE LAUNCH SITES BEGIN WITH `CCA`

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

```
[23]: df = pd.read_sql_query("select *  from SPACEXTBL where Launch_Site like 'CCA%' LIMIT 10", con)
      df
```

[23]:

	Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Orbit	Customer	Mission_Outcome	Landing_Outcome
0	2010-06-04	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success	Failure (parachute)
1	2010-12-08	15:43:00	F9 v1.0 B0004	CCAFS LC-40	Dragon demo flight C1, two CubeSats, barrel of...	0	LEO (ISS)	NASA (COTS) NRO	Success	Failure (parachute)
2	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
3	2012-10-08	0:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500	LEO (ISS)	NASA (CRS)	Success	No attempt
4	2013-03-01	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677	LEO (ISS)	NASA (CRS)	Success	No attempt
5	2013-12-03	22:41:00	F9 v1.1	CCAFS LC-40	SES-8	3170	GTO	SES	Success	No attempt
6	2014-01-06	22:06:00	F9 v1.1	CCAFS LC-40	Thaicom 6	3325	GTO	Thaicom	Success	No attempt
7	2014-04-18	19:25:00	F9 v1.1	CCAFS LC-40	SpaceX CRS-3	2296	LEO (ISS)	NASA (CRS)	Success	Controlled (ocean)
8	2014-07-14	15:15:00	F9 v1.1	CCAFS LC-40	OG2 Mission 1 6 Orbcomm-OG2 satellites	1316	LEO	Orbcomm	Success	Controlled (ocean)
9	2014-08-05	8:00:00	F9 v1.1	CCAFS LC-40	AsiaSat 8	4535	GTO	AsiaSat	Success	No attempt

Total Payload Mass

TOTAL PAYLOAD CARRIED BY BOOSTERS FROM NASA

```
[14]: df = pd.read_sql_query("select sum(PAYLOAD_MASS__KG_) as suma from SPACEXTBL where customer ='NASA (CRS)'", con)
      df
```

[14]:

	suma
0	45596

Average Payload Mass by F9 v1.1

THE AVERAGE PAYLOAD MASS CARRIED BY BOOSTER VERSION F9 V1.1

```
[17]: df = pd.read_sql_query("select avg(PAYLOAD_MASS__KG_)  from SPACEXTBL where Booster_Version ='F9 v1.1'", con)
      df
```

[17]:

	avg(PAYLOAD_MASS__KG_)
0	2928.4

First Successful Ground Landing Date

DISPLAY DATES OF THE FIRST SUCCESSFUL LANDING OUTCOME ON GROUND PAD

```
28]: df = pd.read_sql_query("select  min(Date) from SPACEXTBL where Landing_Outcome ='Success (ground pad)' ", con)
     df
```

28]:

	min(Date)
0	2015-12-22

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

[32]:
```
df = pd.read_sql_query("select Booster_Version, PAYLOAD_MASS__KG_ from SPACEXTBL where Mission_Outcome ='Success' and  (PAYLOAD_MASS__KG_ > 4000 and PAYLOAD_MASS__
df
```

[32]:

	Booster_Version	PAYLOAD_MASS__KG_
0	F9 v1.1	4535
1	F9 v1.1 B1011	4428
2	F9 v1.1 B1014	4159
3	F9 v1.1 B1016	4707
4	F9 FT B1020	5271
5	F9 FT B1022	4696
6	F9 FT B1026	4600
7	F9 FT B1030	5600
8	F9 FT B1021.2	5300
9	F9 FT B1032.1	5300
10	F9 B4 B1040.1	4990
11	F9 FT B1031.2	5200
12	F9 FT B1032.2	4230
13	F9 B4 B1040.2	5384
14	F9 B5 B1046.2	5800
15	F9 B5 B1047.2	5300
16	F9 B5 B1048.3	4850
17	F9 B5 B1051.2	4200
18	F9 B5B1060.1	4311
19	F9 B5 B1058.2	5500
20	F9 B5B1062.1	4311

Total Number of Successful and Failure Mission Outcomes

TOTAL NUMBER OF SUCCESSFUL AND FAILURE MISSION OUTCOMES

```
[25]: #Mission_Outcome
      df = pd.read_sql_query("select count(Mission_Outcome) as 'Mission Success' from SPACEXTBL where Mission_Outcome == 'Success'", con)
      df
```

[25]:

	Mission Success
0	98

```
[26]: df = pd.read_sql_query("select count(Mission_Outcome) as 'Mission Fail' from SPACEXTBL where Mission_Outcome != 'Success'", con)
      df
```

[26]:

	Mission Fail
0	3

Boosters Carried Maximum Payload

LIST THE NAMES OF THE BOOSTER WHICH HAVE CARRIED THE MAXIMUM PAYLOAD MASS

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

```
sql = "select substr(Date, 6,2) as month, Landing_Outcome,Booster_Version, Launch_Site  from SPACEXTBL WHERE substr(Date,0,5)='2015' and Landing_Outcome = 'Failure (drone ship)'"
df = pd.read_sql_query(sql, con)
df
```

	month	Landing_Outcome	Booster_Version	Launch_Site
0	01	Failure (drone ship)	F9 v1.1 B1012	CCAFS LC-40
1	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

	Landing_Outcome	Date
0	Failure (parachute)	2010-06-04
1	Failure (parachute)	2010-12-08
2	No attempt	2012-05-22
3	No attempt	2012-10-08
4	No attempt	2013-03-01
5	Uncontrolled (ocean)	2013-09-29
6	No attempt	2013-12-03
7	No attempt	2014-01-06
8	Controlled (ocean)	2014-04-18
9	Controlled (ocean)	2014-07-14
10	No attempt	2014-08-05
11	No attempt	2014-09-07
12	Uncontrolled (ocean)	2014-09-21
13	Failure (drone ship)	2015-01-10
14	Controlled (ocean)	2015-02-11
15	No attempt	2015-03-02
16	Failure (drone ship)	2015-04-14

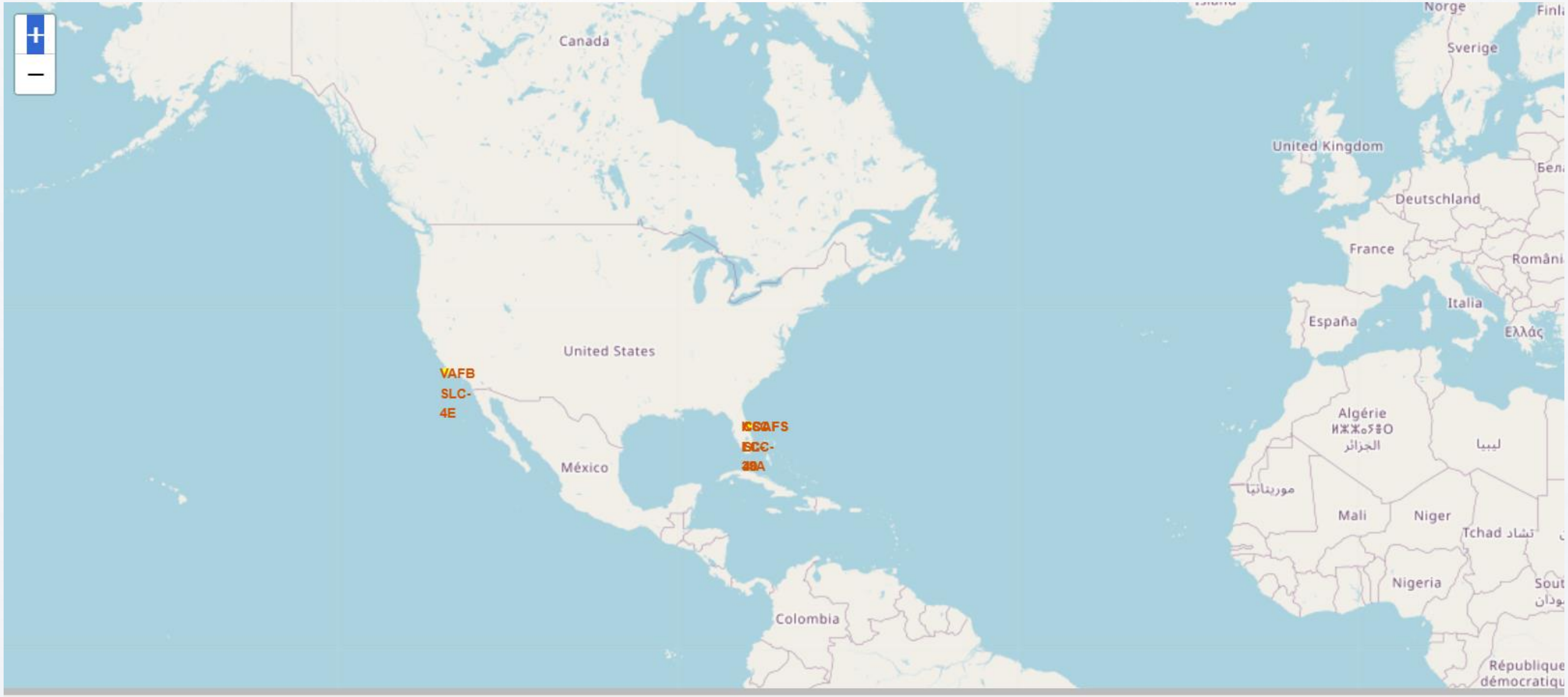
	Landing_Outcome	Date
16	Failure (drone ship)	2015-04-14
17	No attempt	2015-04-27
18	Precluded (drone ship)	2015-06-28
19	Success (ground pad)	2015-12-22
20	Failure (drone ship)	2016-01-17
21	Failure (drone ship)	2016-03-04
22	Success (drone ship)	2016-04-08
23	Success (drone ship)	2016-05-06
24	Success (drone ship)	2016-05-27
25	Failure (drone ship)	2016-06-15
26	Success (ground pad)	2016-07-18
27	Success (drone ship)	2016-08-14
28	Success (drone ship)	2017-01-14
29	Success (ground pad)	2017-02-19
30	No attempt	2017-03-16

Section 3

Launch Sites Proximities Analysis

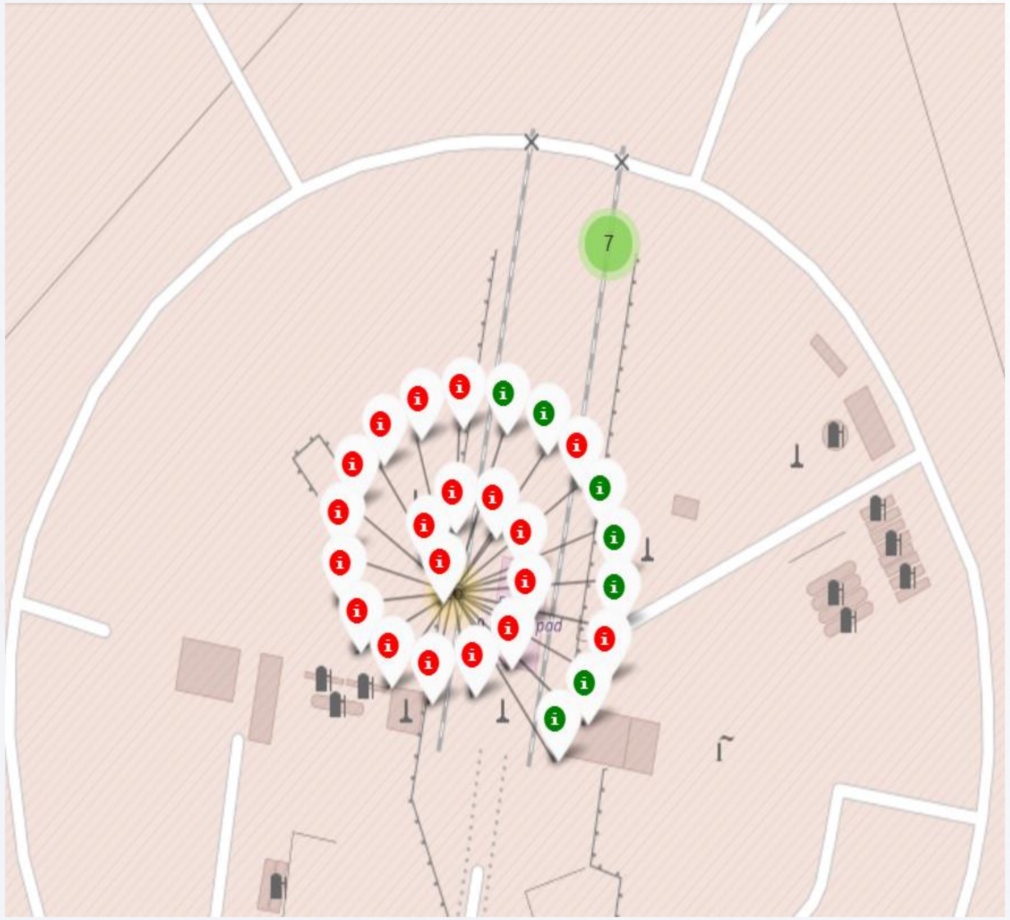
SPACE X LAUNCH SITE

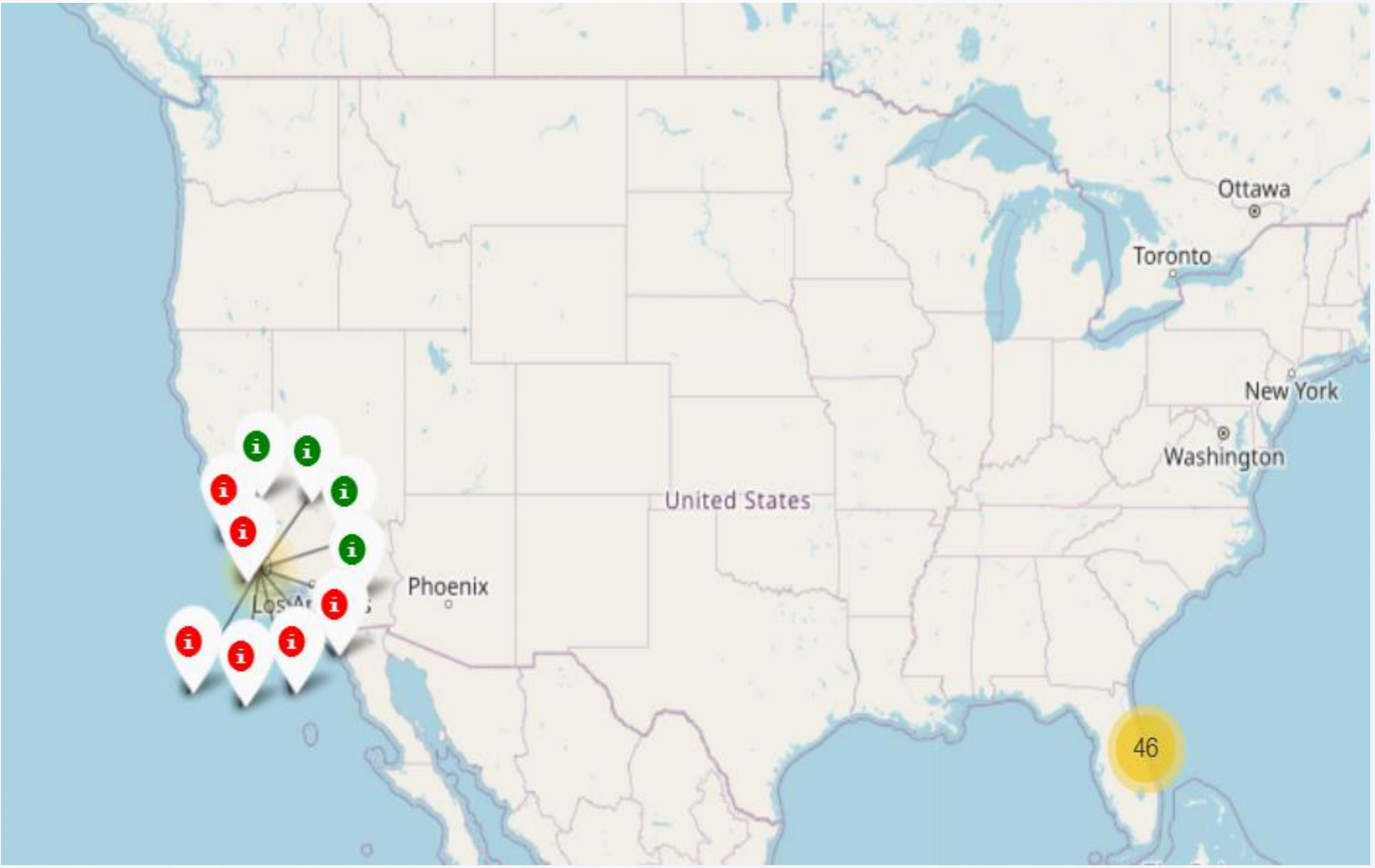
THE LAUNCH SITE IS IMPORTANT BECAUSE THE SUCCESS RATE MAY ALSO DEPEND OF THE LOCATION. IN THIS CASE WE HAVE TWO LOCATIONS: LOS ANGELES AND MIAMI



SUCCESS LANDING

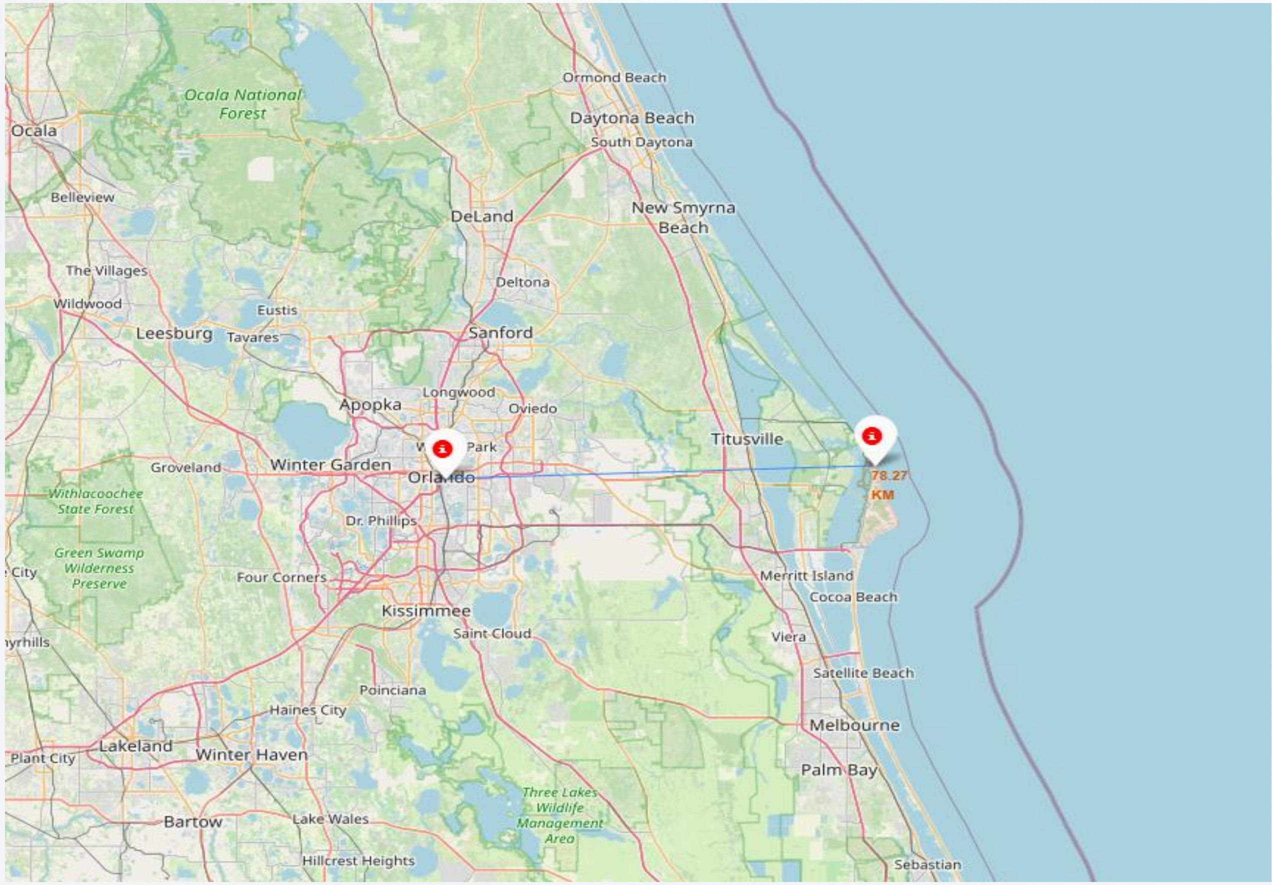
GREEN MARKS IDENTIFY SUCCESS LANDING AND RED MARKS IDENTIFY FALIL LANDING. WE SHOULD BE ABLE TO EASILY IDENTIFY WHICH LAUNCH SITES HAVE RELATIVELY HIGH SUCCESS RATES.

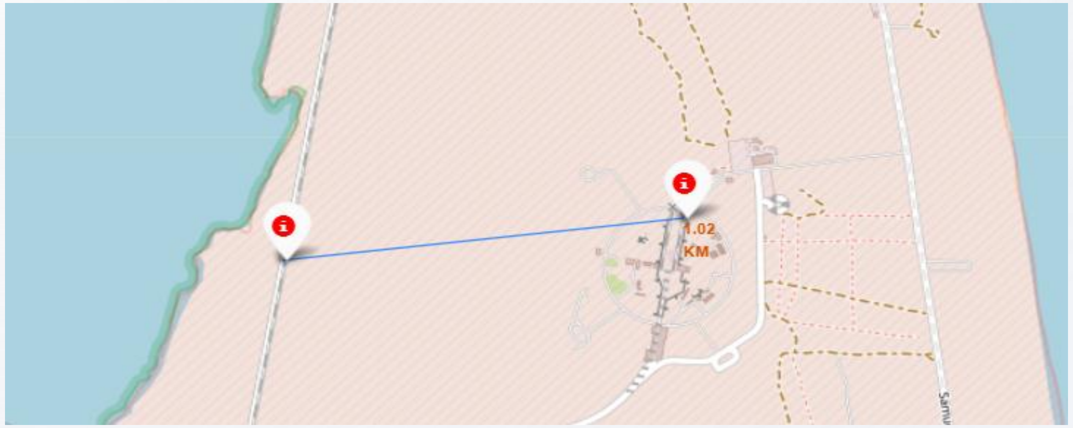




DISPLAY DISTANCES BETWEEN A LAUNCH SITE

- DISPLAY DISTANCE BETWEEN A LAUNCH SITE TO: CLOSEST CITY, RAIL ROAD AND HIGHWAY



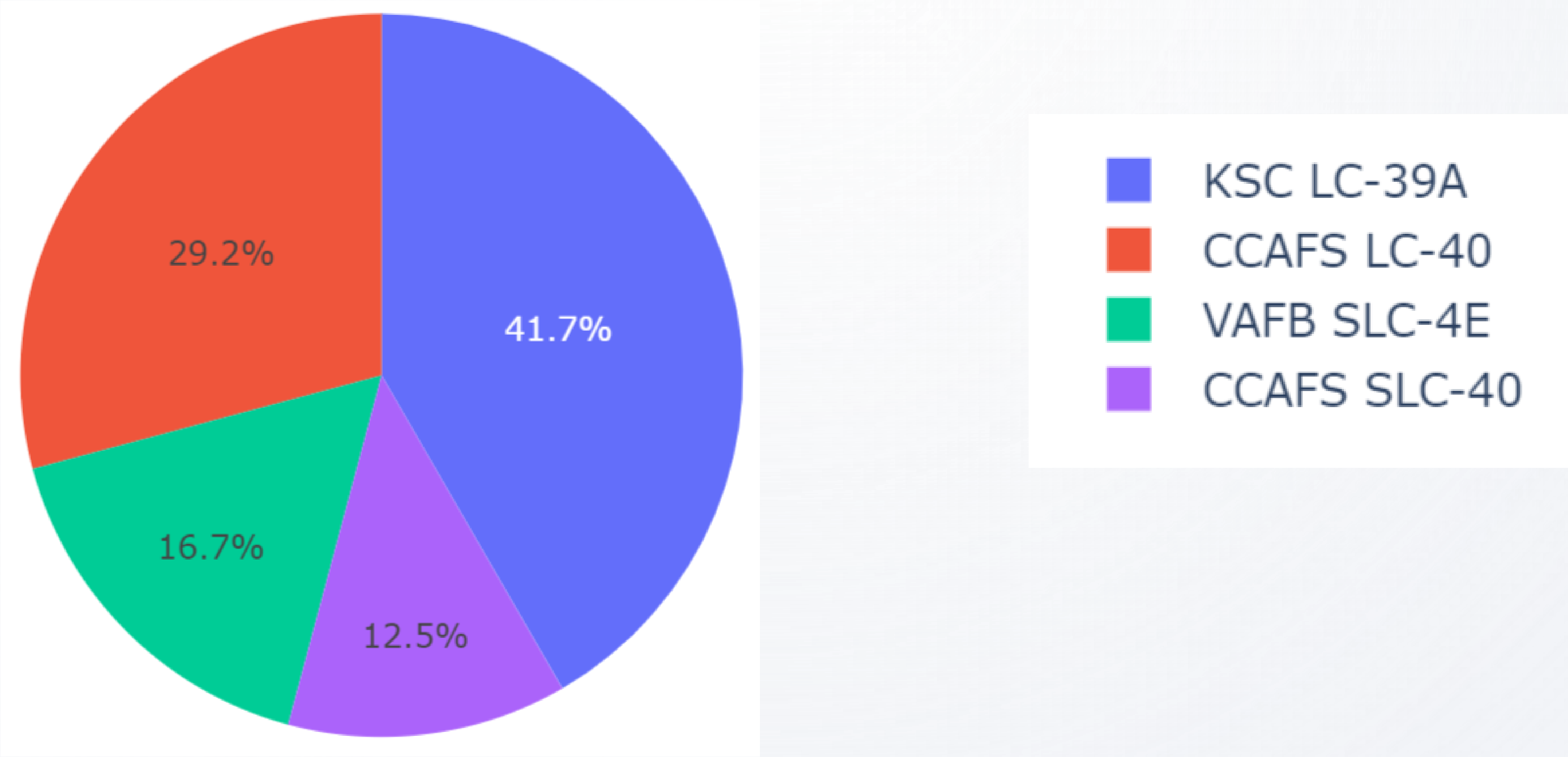


Section 4

Build a Dashboard with Plotly Dash

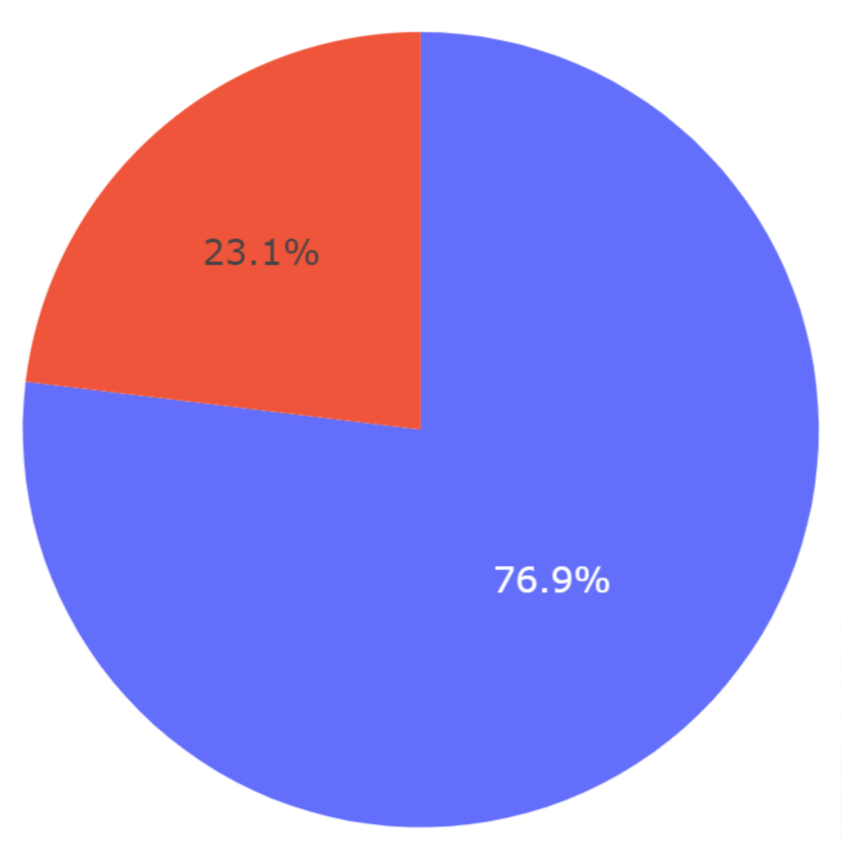
LANDING SUCCESS BY LAUNCH SITE

THE SITE KSC LC-39A HAS THE MOST LANDING SUCCESS WHIT A 41% RATE AND CCFC LC-40 HAS THE SECOND PLACE WITH A 29% RATE



KSC LC-39A LANDING SUCCESS RATE

DISPLAY THE SUCCESS AND FAIL LANDING OF THE LAUNCH SITE WITH THE HIGHEST SUCCESS LANDING RATE

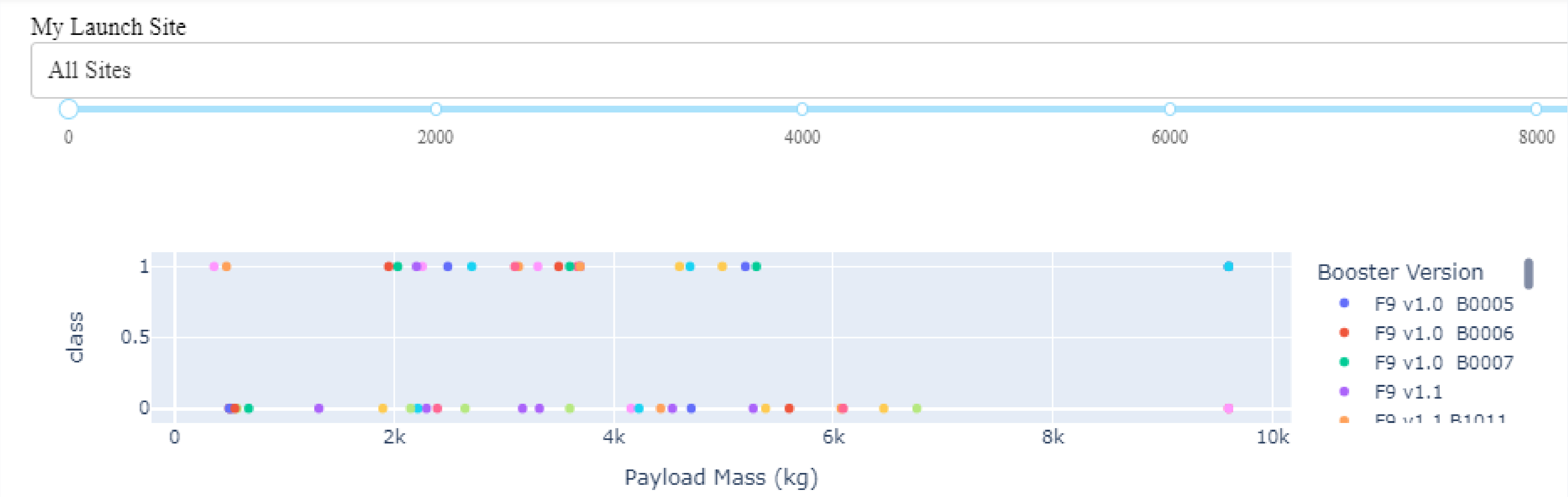


1 SUCCESS

0 FAIL

PAYLOADA RANGE SUCCESS

- THE GRAPH DISPLAY
  - THE HIGHEST RATE SUCCESS PAYLOAD IS BETWEEN 2000KG AND 5000KG
  - AND THE HIGHEST PAYLOAD SUCCESS IS 9000KG.



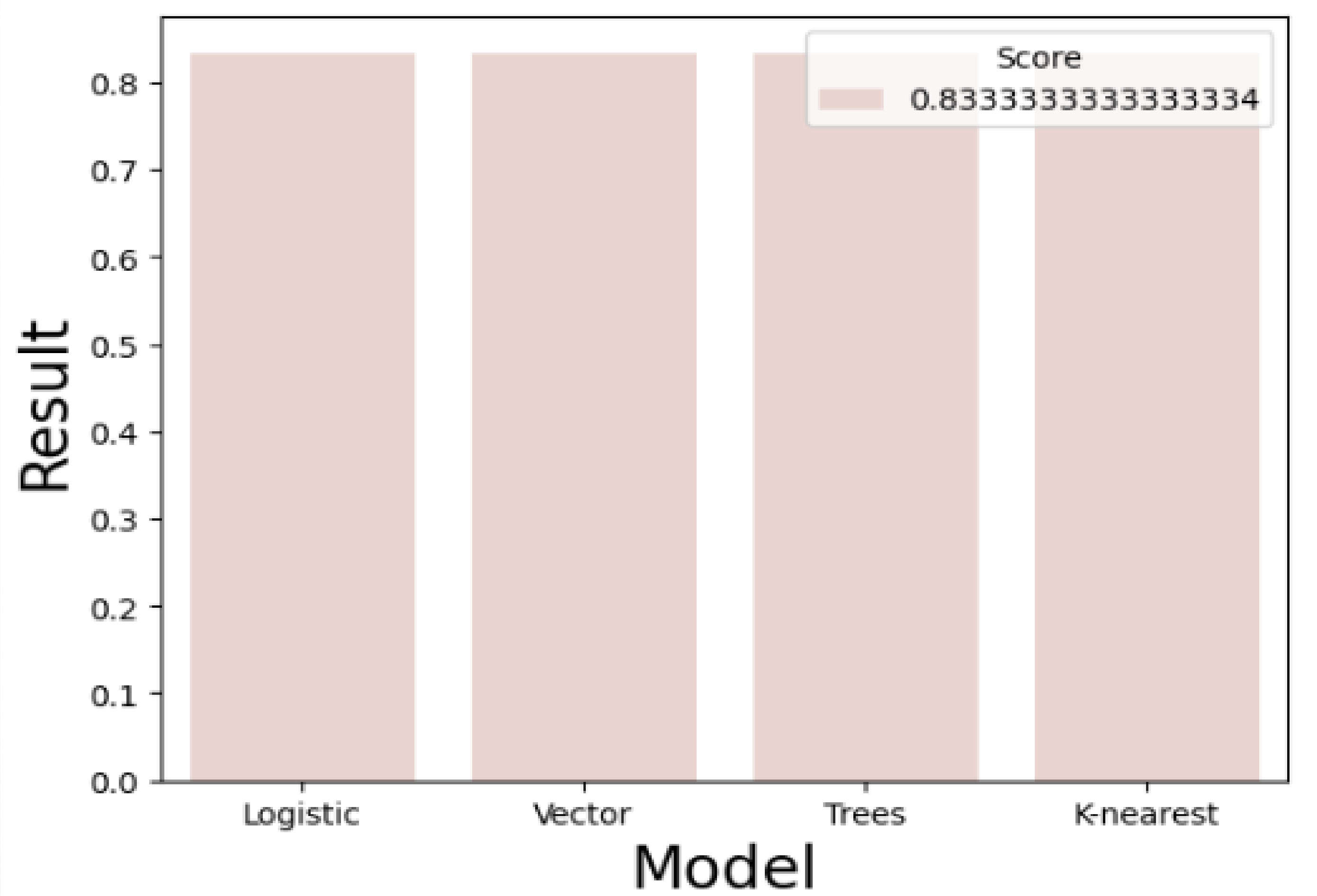
Section 5

Predictive Analysis (Classification)

Classification Accuracy

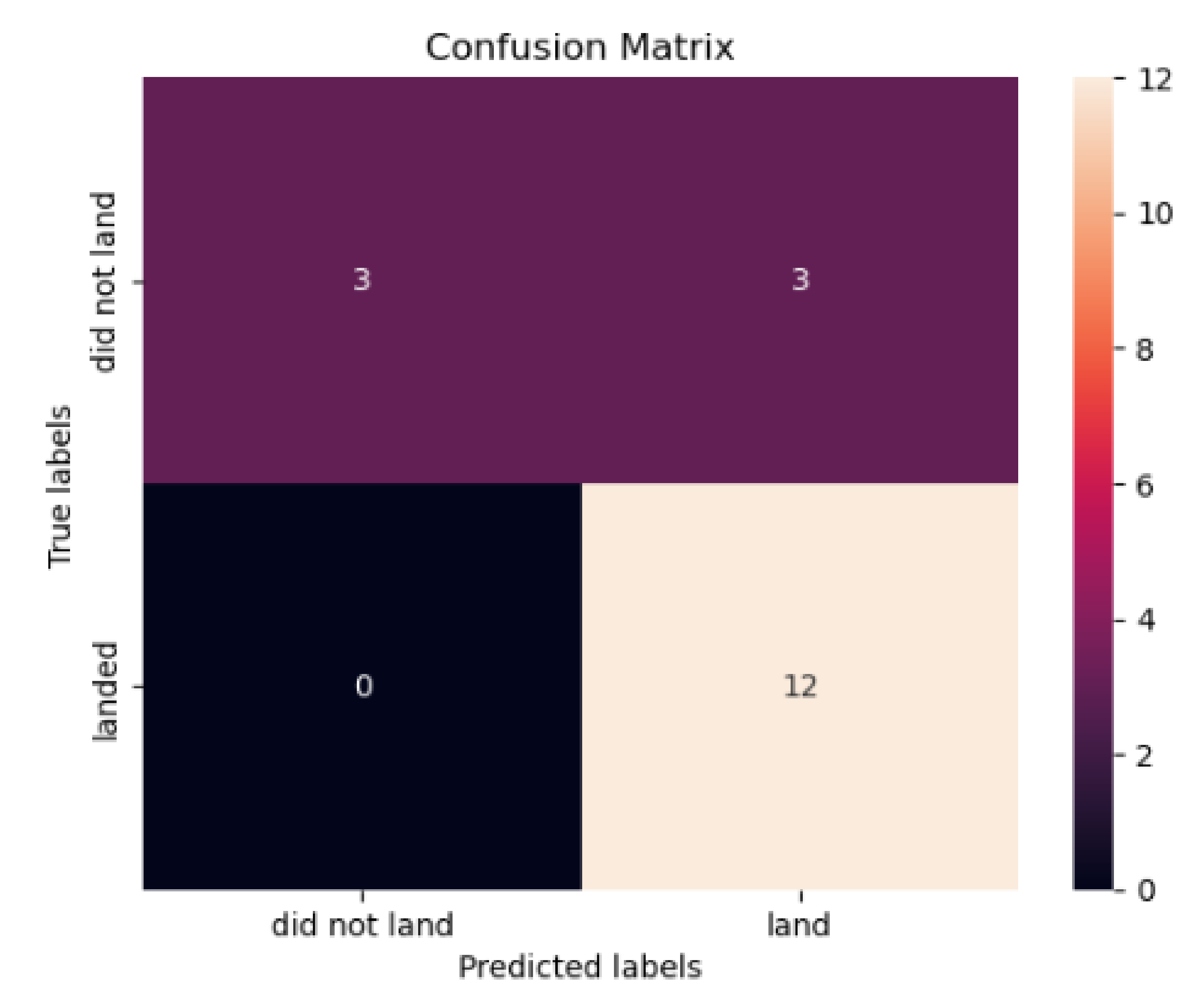
THE GRAPH DISPLAY THAT ALL MODELS HAVE THE SAME CLASSIFICATIONS ACCURACY

	Model	Score
0	Logistic	0.833333
1	Vector	0.833333
2	Trees	0.833333
3	K-nearest	0.833333



Confusion Matrix

THE GRAPH DISPLAY THE MODEL HAS HIGH ACCURACY TO PREDICT SUCCESSFUL LANDING, WHILE ONLY THE 50% OF FAIL LANDING WAS PREDICT SUCCESSFUL



Conclusions

- THE LAST 13 LAUNCH HAS 100% LANDING SUCCESS
- THE MOST PAYLOAD IS 15600KG
- THE LAUNCH TO ORBIT ES-L1, GEO, HEO, SSO, HAS 100% LANDING SUCCESS
- ORBIT VLEO, PO, ISS ARE THE ONLY ONE THAT HAVE MORE THAN 8000KG OF PAYLOAD
- KSK LC-39A LAUNCH SITE HAS THE HIGHEST LANDING SUCCESS
- ALL LAUNCHES WERE NEAR THE COAST
- THE PREDICTIVE MODELS HAS HIGH ACCURACY DETERMINING SUCCESS LANDINGS, BUT HAS POOR RESULTS WHIT FAILED LANDINGS, JUST 50%

Appendix

- FILE CSV USED TO STORAGE DATA
- FILES OF JUPYTER NOTEBOOK, USED TO PROCESS THE DATA AND SHOW RESULT.

Thank you!