IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

<Name>
<Date>

Outline

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

Executive Summary

- Summary of methodologies
  - NNK K-Nearst_neighbors
  - Decision Tree
  - SMV Support Vetor Machine
- Summary of all results
  - Best prediction accurace is 0.86
  - Best prdiction methodolog is Decision Tree

Introduction

- Project background and context
  - To make better success rate with data ananlysis

- Problems you want to find answers
  - What kind of datas infulence the result of success of the launchs

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - use get request : requests.get("url")
- Perform data wrangling
  - filter the data and replace the nan values df.fillna()
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - How to build, tune, evaluate classification models

Data Collection

Import Libraties

↓

Auxiliary Function
to
extract data

↓

requests.get(spacex_url)

↓

Decode
data = pd.json_normalize(response.json())

Data Collection – SpaceX API

- Present your data collection with SpaceX REST calls using key phrases and flowcharts
- Add the GitHub URL of the completed SpaceX API calls notebook (must include completed code cell and outcome cell), as an external reference and peer-review purpose

- spacex_url="https://api.spacexdata.com/v4/launches/past"
- response = requests.get(spacex_url)
- response.status_code
- data = pd.json_normalize(response.json())
- launch_df = pd.DataFrame.from_dict(launch_dict)
- data_falcon9 = launch_df[(launch_df['BoosterVersion']!='Falcon 1')]
- data_falcon9.loc[:,'FlightNumber'] = list(range(1, data_falcon9.shape[0]+1))

https://github.com/zjwxf/coursera/blob/main/jupyter-labs-spacex-data-collection-api.ipynb

Data Collection – Scraping

- Present your web scraping process using key phrases and flowcharts

- Add the GitHub URL of the completed web scraping notebook, as an external reference and peer-review purpose

- def getBoosterVersion(data):

  for x in data['rocket']:

  response = requests.get("https://api.spacexdata.com/v4/rockets/"+str(x)).json()

  BoosterVersion.append(response['name'])
- spacex_url="https://api.spacexdata.com/v4/launches/past"
- response = requests.get(spacex_url)
- URL:https://github.com/zjwxf/coursera/blob/main/jupyter-labs-spacex-data-collection-api.ipynb

Data Wrangling

1. Find out the missing values
2. replace the missing values
3. save the new dataset to csv

- data_falcon9.isnull().sum()
- mean_val = data_falcon9['PayloadMass'].mean()
- data_falcon9['PayloadMass'].fillna(value=mean_val, inplace=True)
- data_falcon9.isnull().sum()
- data_falcon9.to_csv('dataset_part_1.csv', index=False)
- URL:https://github.com/zjwxf/coursera/blob/main/jupyter-labs-spacex-data-collection-api.ipynb

EDA with Data Visualization

- Summarize what charts were plotted and why you used those charts
- Scatter piont plot,
- Bar chart,
- Line chart
- The charts show the relationship between the parameters which we interested in.

- URL:https://github.com/zjwxf/coursera/blob/main/jupyter-labs-eda-dataviz.ipynb

EDA with SQL

- SELECT TABSCHEMA, TABNAME, CREATE_TIME FROM SYSCAT.TABLES WHERE TABSCHEMA='XCG80731';
- SELECT DISTINCT LAUNCH_SITE FROM SPACEXTBL;
- SELECT COUNT(*) LAUNCH_SITE FROM SPACEXTBL WHERE LAUNCH_SITE = 'CCAFS LC-40';
- SELECT LAUNCH_SITE FROM SPACEXTBL WHERE LAUNCH_SITE LIKE 'CCA%' LIMIT 5;
- SELECT SUM(PAYLOAD_MASS__KG_) FROM SPACEXTBL WHERE Customer = 'NASA (CRS)';
- SELECT AVG(PAYLOAD_MASS__KG_) FROM SPACEXTBL WHERE Booster_Version LIKE 'F9 v1.0%';
- SELECT MIN(Date) FROM SPACEXTBL WHERE Landing__Outcome = 'Success (ground pad)'
- SELECT BOOSTER_VERSION FROM SPACEXTBL WHERE LANDING__OUTCOME = 'Success (drone ship)' AND 4000 < PAYLOAD_MASS__KG_ < 6000;
- SELECT MISSION_OUTCOME, COUNT(MISSION_OUTCOME) AS TOTAL_NUMBER FROM SPACEXTBLGROUP BY MISSION_OUTCOME;
- SELECT DISTINCT BOOSTER_VERSION FROM SPACEXTBL WHERE PAYLOAD_MASS__KG_ = (SELECT MAX(PAYLOAD_MASS__KG_) FROM SPACEXTBL);
- SELECT LANDING__OUTCOME, BOOSTER_VERSION, LAUNCH_SITE FROM SPACEXTBL WHERE Landing__Outcome = 'Failure (drone ship)' AND YEAR(DATE) = 201
- SELECT LANDING__OUTCOME, COUNT(LANDING__OUTCOME) AS TOTAL_NUMBER FROM SPACEXTBL WHERE DATE BETWEEN '2010-06-04' AND '2017-03-20' GROUP BY LANDING__OUTCOME ORDER BY TOTAL_NUMBER DESC
- URL:https://github.com/zjwxf/coursera/blob/main/Hands-on%20Lab%20Complete%20the%20EDA%20with%20SQL.ipynb

Build an Interactive Map with Folium

- Summarize what map objects such as markers, circles, lines, etc. you created and added to a folium map
- Mark all launch sites on a map with markers and circles
- Mark the success/failed launches for each site on the map with markers and circles
- Distance between a launch site to its proximities with lines.
- Explain why you added those objects
- To show the location of lauch sites

- URL:https://github.com/zjwxf/coursera/blob/main/lab_jupyter_launch_site_location.ipynb

Build a Dashboard with Plotly Dash

- Summarize what plots/graphs and interactions you have added to a dashboard
- piont chart
- pie chart (success-pie-chart, success-payload-scatter-chart)
- Explain why you added those plots and interactions
- To display the relation between success and paramerters

- URL: https://github.com/zjwxf/coursera/blob/main/SpaceX_Dashboard.py

Predictive Analysis (Classification)

Y= data['Class'].to_numpy()

↓

Standardlize and transform X
X.mean(axis=0)
X.std(axis=0)
transform = preprocessing.StandardScaler().fit(X)

↓

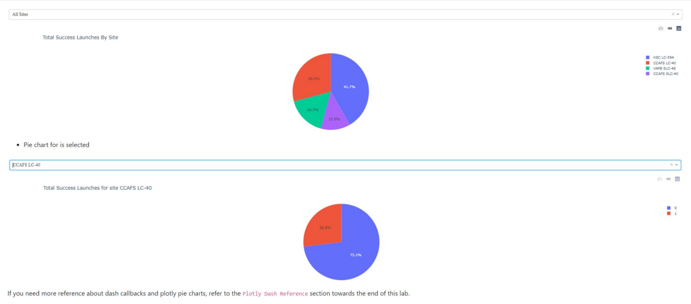
Train_Test_Split
X_train, X_test, Y_train, Y_test =
train_test_split(X,Y,test_size=0.2,random_state=2)

↓

Logistic regression
SVM
KNN
Decision tree

Results

- Exploratory data analysis results
- Success rate of SPACEXLauch is 0.66
- Interactive analytics demo in screenshots



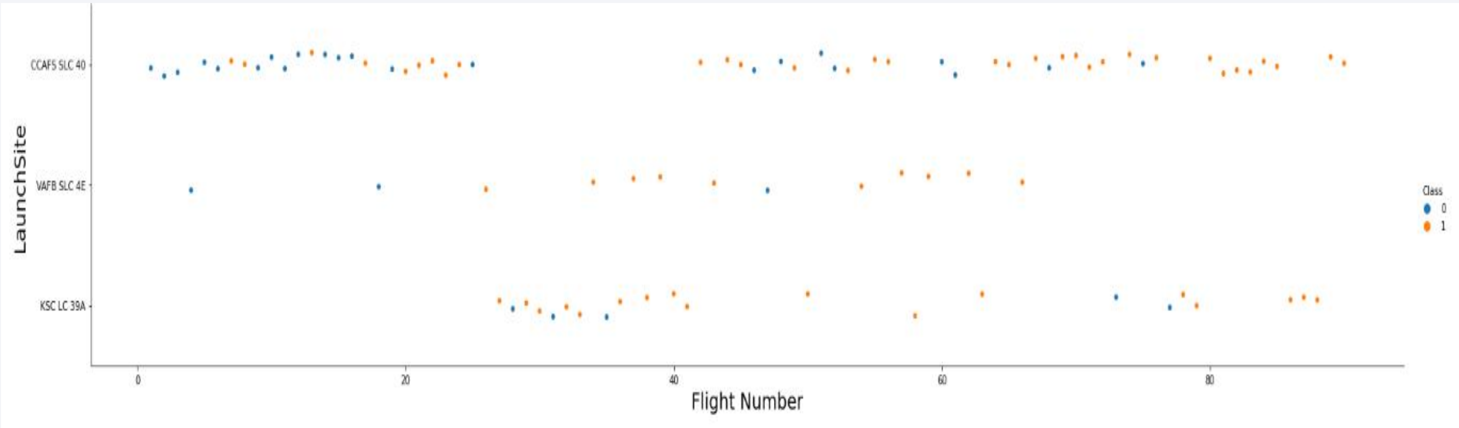
- Predictive analysis results
- KNN 0.83
- Decision tree 0.86
- SVM 0.84
- logistic regression 0.84

Section 2

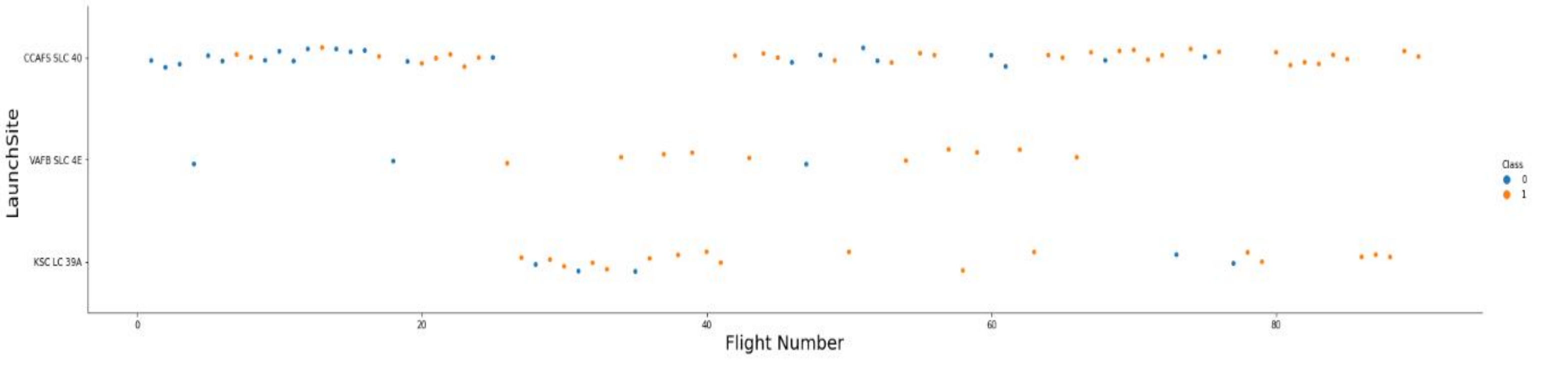
Insights drawn from EDA

Flight Number vs. Launch Site

- Show a scatter plot of Flight Number vs. Launch Site

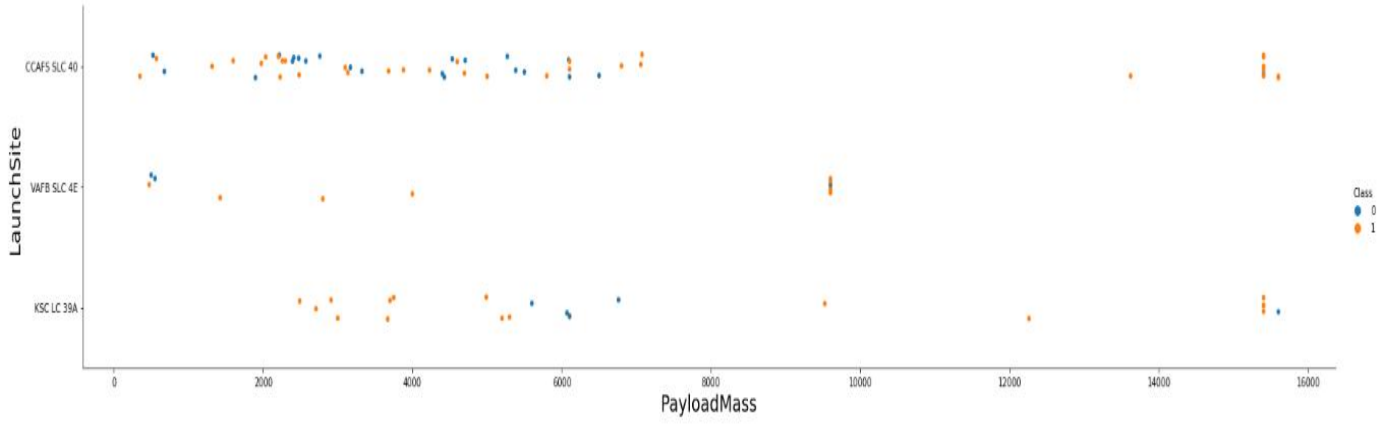


- Show the screenshot of the scatter plot with explanations

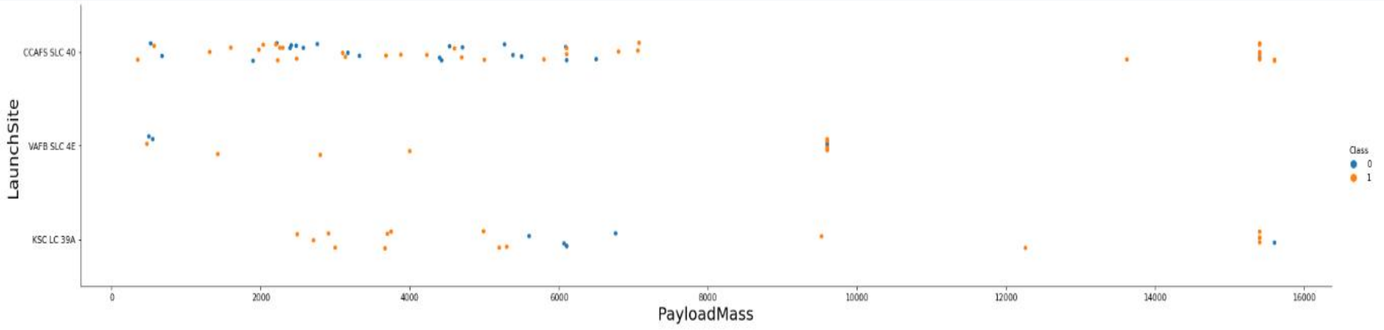


Payload vs. Launch Site

- Show a scatter plot of Payload vs. Launch Site



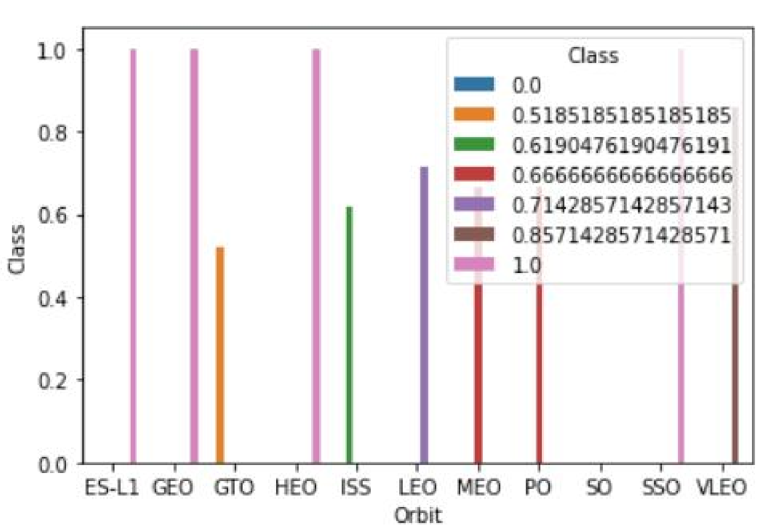
- Show the screenshot of the scatter plot with explanations



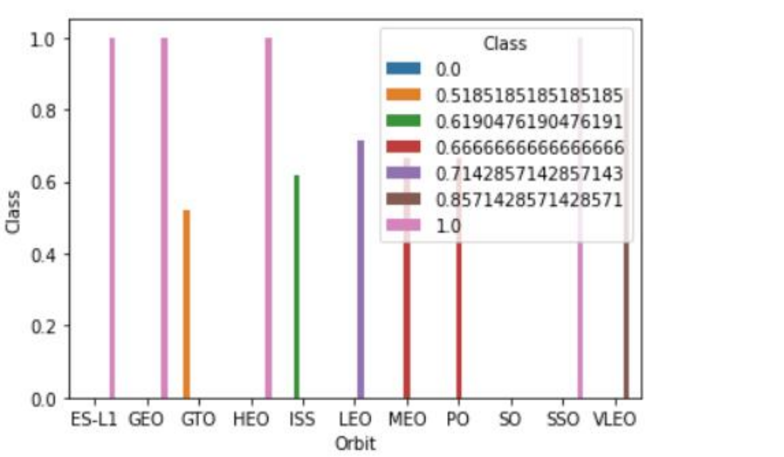
Now if you observe Payload Vs. Launch Site scatter point chart you will find for the VAFB-SLC launchsite there are no rockets launched for heavypayload mass(greater than 10000).

Success Rate vs. Orbit Type

- Show a bar chart for the success rate of each orbit type



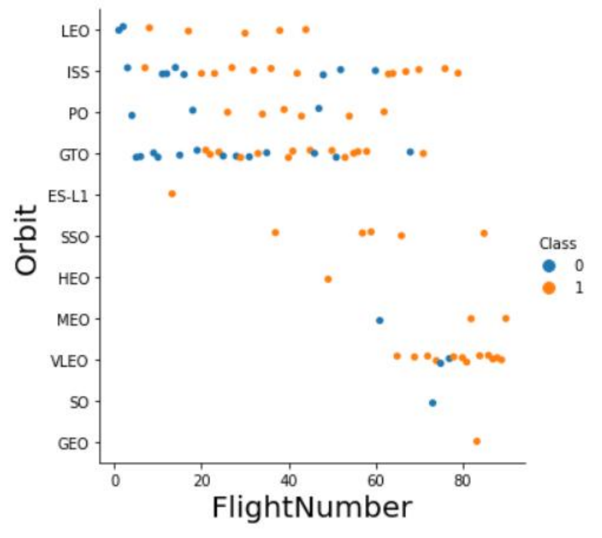
- Show the screenshot of the scatter plot with explanations



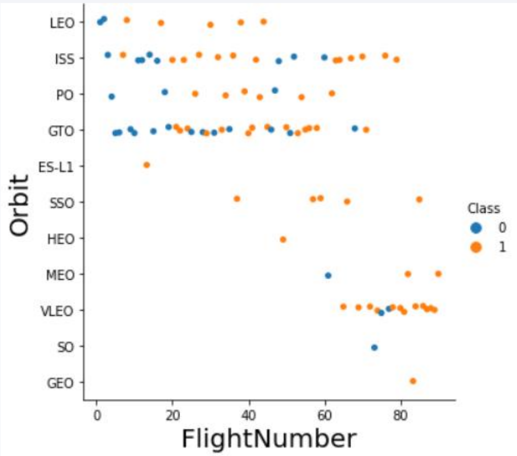
Analyze the ploted bar chart try to find which orbits have high sucess rate.

Flight Number vs. Orbit Type

- Show a scatter point of Flight number vs. Orbit type



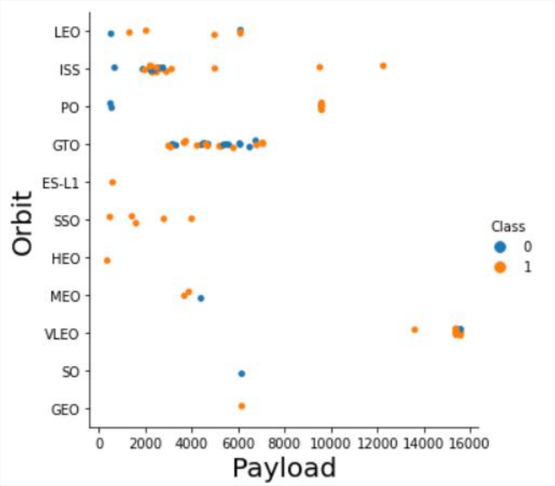
- Show the screenshot of the scatter plot with explanations



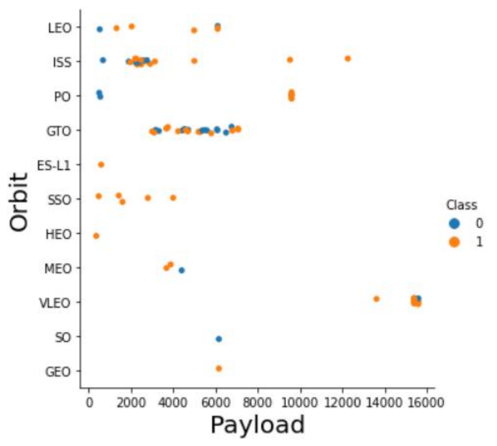
You should see that in the LEO orbit the Success appears related to the number of flights; on the other hand, there seems to be no relationship between flight number when in GTO orbit.

Payload vs. Orbit Type

- Show a scatter point of payload vs. orbit type



- Show the screenshot of the scatter plot with explanations

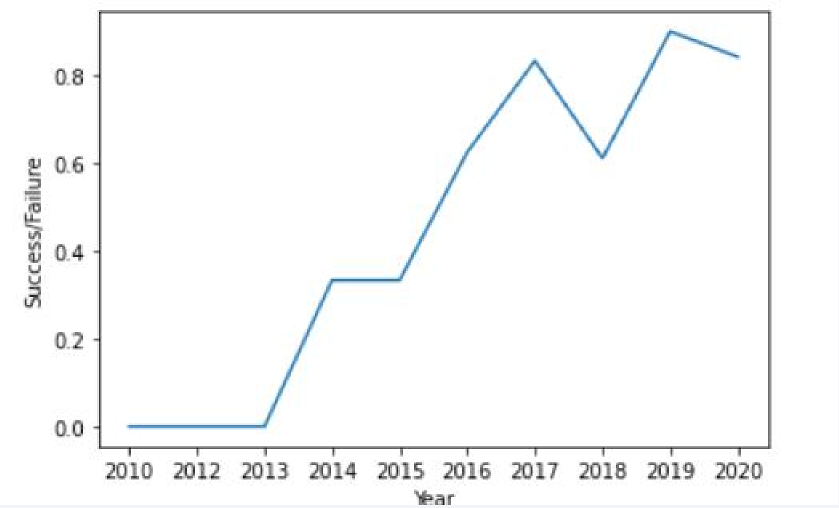


With heavy payloads the successful landing or positive landing rate are more for Polar,LEO and ISS.

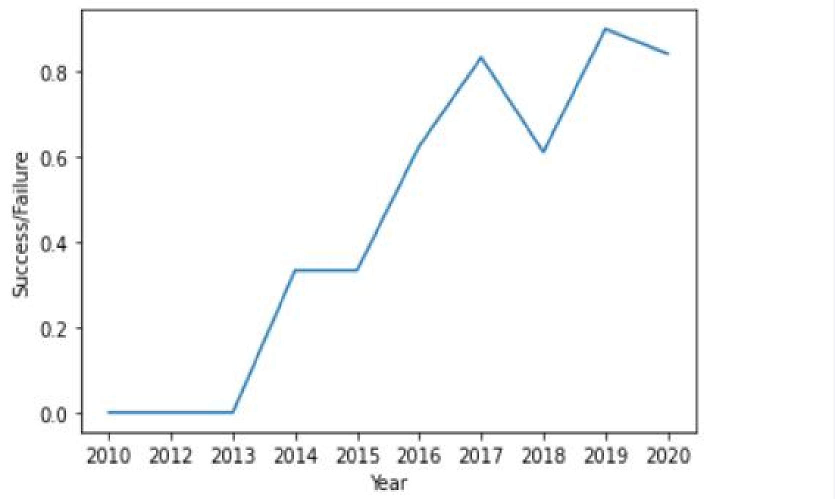
However for GTO we cannot distinguish this well as both positive landing rate and negative landing(unsuccessful mission) are both there here.

Launch Success Yearly Trend

- Show a line chart of yearly average success rate



- Show the screenshot of the scatter plot with explanations



you can observe that the sucess rate since 2013 kept increasing till 2020

All Launch Site Names

- Find the names of the unique launch sites
- SELECT DISTINCT LAUNCH_SITEFROM SPACEXTBL;
- Present your query result with a short explanation here
- Total 4 launch site

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

Launch Site Names Begin with 'CCA'

- Find 5 records where launch sites begin with `CCA`
- SELECT LAUNCH_SITE FROM SPACEXTBLWHERE LAUNCH_SITE LIKE 'CCA%'LIMIT 5;
- Present your query result with a short explanation here
- 5 CCA launch sites

1. CCAFS LC-40
2. CCAFS LC-40
3. CCAFS LC-40
4. CCAFS LC-40
5. CCAFS LC-40

Total Payload Mass

- Calculate the total payload carried by boosters from NASA
  - SELECT SUM(PAYLOAD_MASS__KG_) FROM SPACEXTBL WHERE Customer = 'NASA (CRS)';
- Present your query result with a short explanation here
  - The total payload by bosster from NASA is 45596

Average Payload Mass by F9 v1.1

- Calculate the average payload mass carried by booster version F9 v1.1
- SELECT AVG(PAYLOAD_MASS__KG_) FROM SPACEXTBL WHERE Booster_Version LIKE 'F9 v1.0%';
- Present your query result with a short explanation here
- Booster F9 V1.1 carried 340.

First Successful Ground Landing Date

- Find the dates of the first successful landing outcome on ground pad
- SELECT MIN(Date) FROM SPACEXTBL WHERE Landing__Outcome = 'Success (ground pad)';
- Present your query result with a short explanation here
- The first successful landing on ground pas was at 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
- SELECT BOOSTER_VERSION FROM SPACEXTBL WHERE LANDING__OUTCOME = 'Success (drone ship)' AND 4000 < PAYLOAD_MASS__KG_ < 6000;
- Present your query result with a short explanation here

The successful drone ship landing booster_version between 4000 and 6000

- F9 FT B1021.1
- F9 FT B1023.1
- F9 FT B1029.2
- F9 FT B1038.1
- F9 B4 B1042.1
- F9 B4 B1045.1
- F9 B5 B1046.1

Total Number of Successful and Failure Mission Outcomes

- Calculate the total number of successful and failure mission outcomes
- ELECT MISSION_OUTCOME, COUNT(MISSION_OUTCOME) AS TOTAL_NUMBER FROM SPACEXTBL GROUP BY MISSION_OUTCOME;
- Present your query result with a short explanation here
- 99% successful rate

mission_outcome	total_number
Failure (in flight)	1
Success	99
Success (payload status unclear)	1

Boosters Carried Maximum Payload

- List the names of the booster which have carried the maximum payload mass
- SELECT DISTINCT BOOSTER_VERSION FROM SPACEXTBL WHERE PAYLOAD_MASS__KG_ = (SELECT MAX(PAYLOAD_MASS__KG_) FROM SPACEXTBL);
- Present your query result with a short explanation here
- The maximum payload booster version

F9 B5 B1048.4 F9 B5 B1048.5 F9 B5 B1049.4 F9 B5 B1049.5 F9 B5 B1049.7

F9 B5 B1051.3 F9 B5 B1051.4 F9 B5 B1051.6 F9 B5 B1056.4

F9 B5 B1058.3 F9 B5 B1060.2 F9 B5 B1060.3

2015 Launch Records

- List the failed landing_outcomes in drone ship, their booster versions, and launch site names for in year 2015
- SELECT LANDING__OUTCOME, BOOSTER_VERSION, LAUNCH_SITE
- FROM SPACEXTBL
- WHERE Landing__Outcome = 'Failure (drone ship)'
- AND YEAR(DATE) = 2015
- Present your query result with a short explanation here
- The 2015 launch records

landing__outcome	booster_version	launch_site
Failure (drone ship)	F9 v1.1 B1012	CCAFS LC-40
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
  - SELECT LANDING__OUTCOME, COUNT(LANDING__OUTCOME) AS TOTAL_NUMBER
  - FROM SPACEXTBL
  - WHERE DATE BETWEEN '2010-06-04' AND '2017-03-20'
  - GROUP BY LANDING__OUTCOME
  - ORDER BY TOTAL_NUMBER DESC
- Present your query result with a short explanation here
- Rank result summary in table

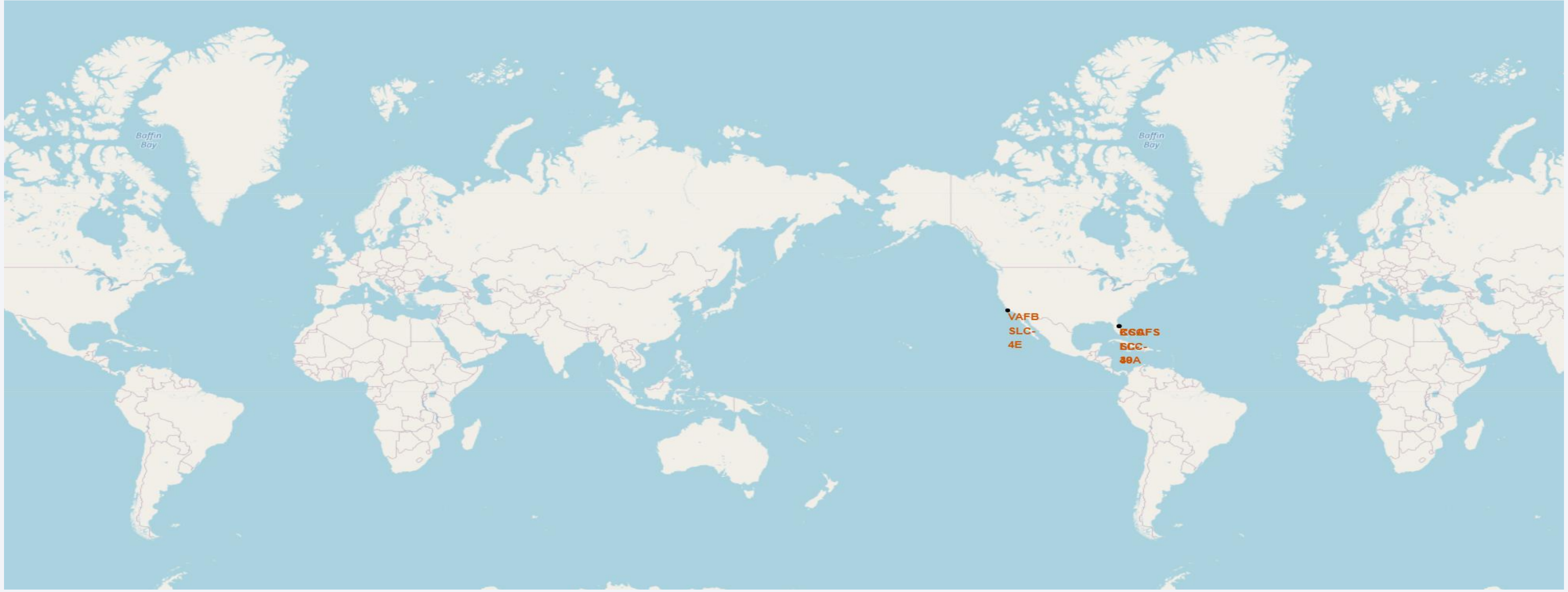
landing__outcome	total_number
No attempt	10
Failure (drone ship)	5
Success (drone ship)	5
Controlled (ocean)	3
Success (ground pad)	3
Failure (parachute)	2
Uncontrolled (ocean)	2
Precluded (drone ship)	1

Section 3

Launch Sites Proximities Analysis

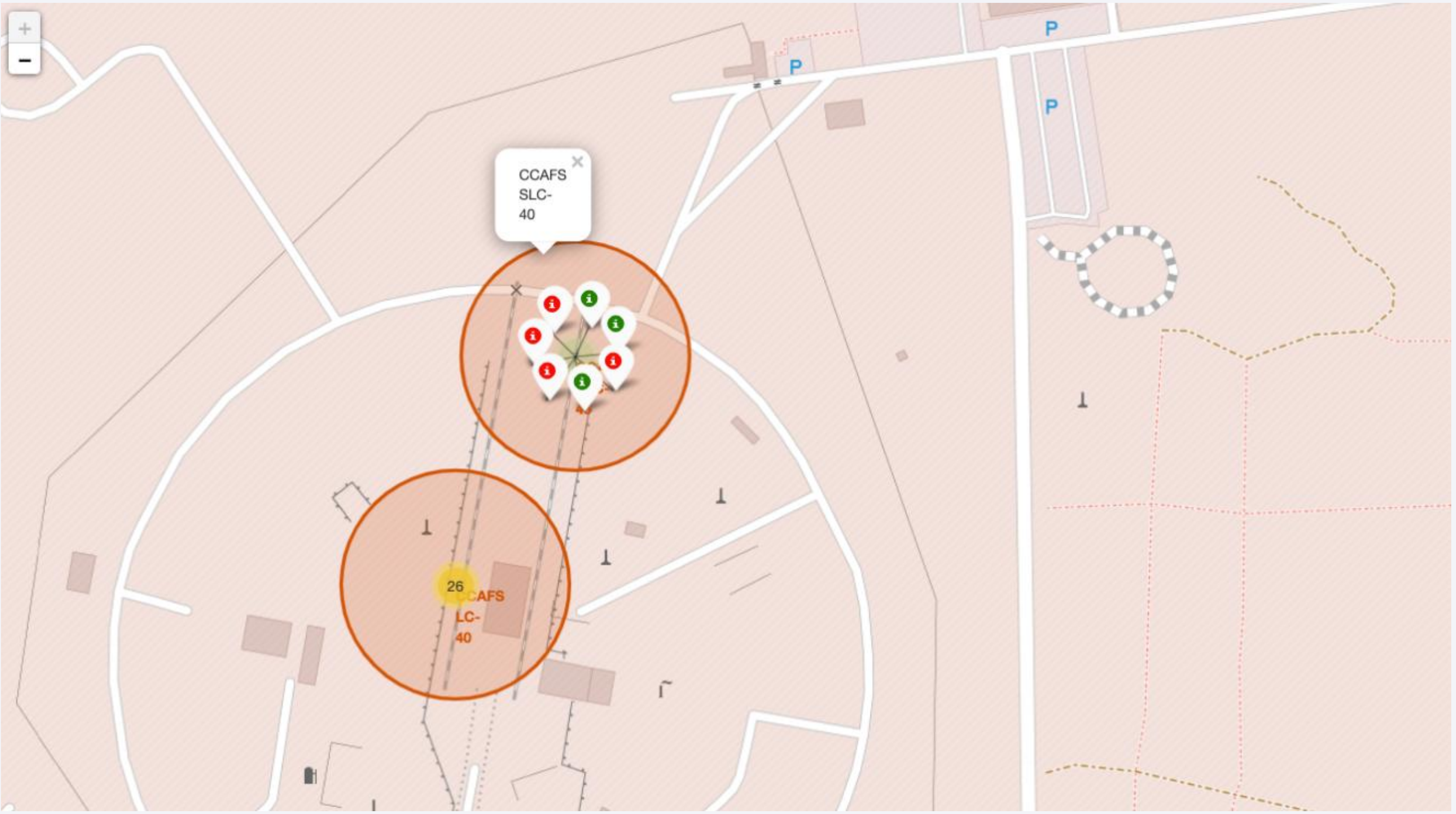
Launch sites loaction map

All launch sites are in proximity to the Equator line and very close proximity to the coast



Success/failed launches for each site on the map

- Success site loaction is green
- Failed site location is red



The distances between a launch site to its proximities

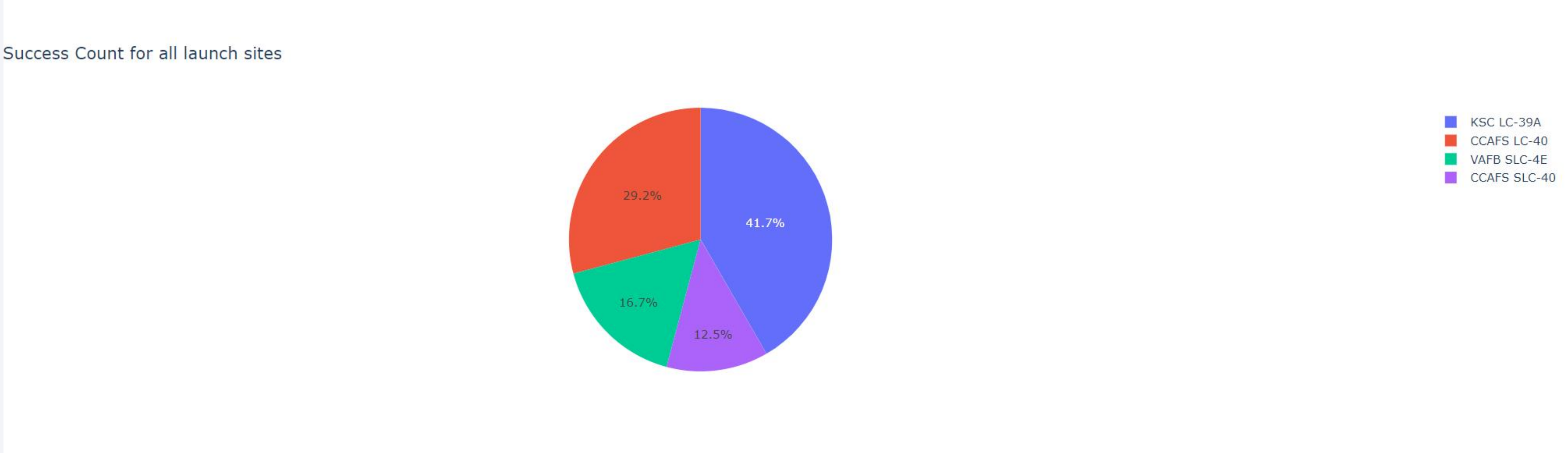
- The launch site is very close to coast

Section 4

Build a Dashboard with Plotly Dash

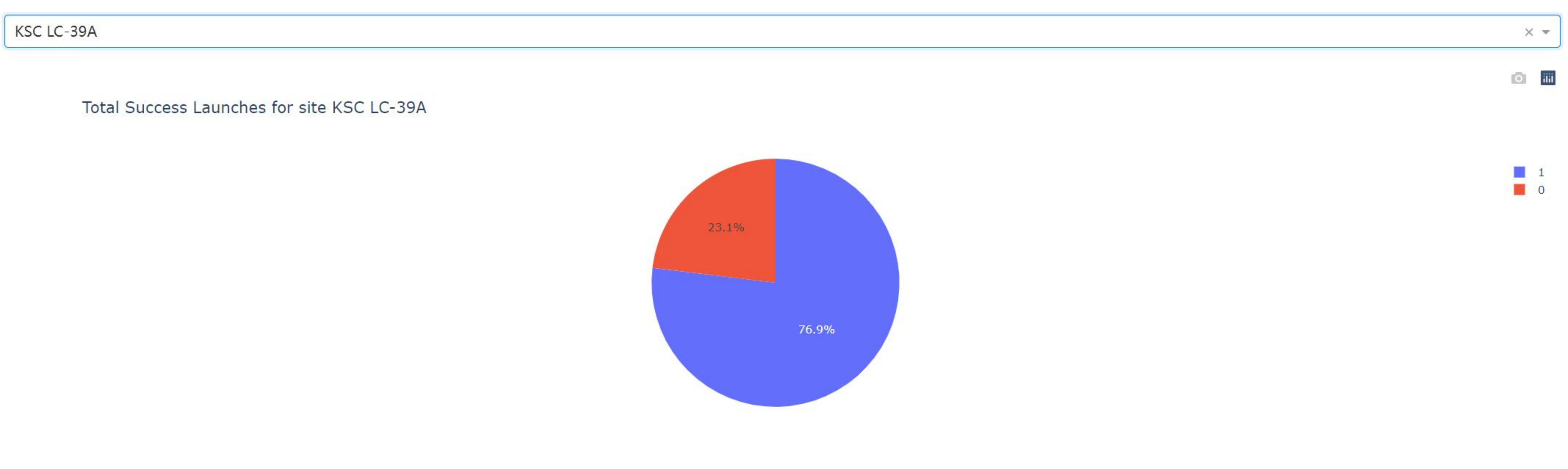
Launch success count for all sites

- KSC LC-39A has the most success count of all launch sites



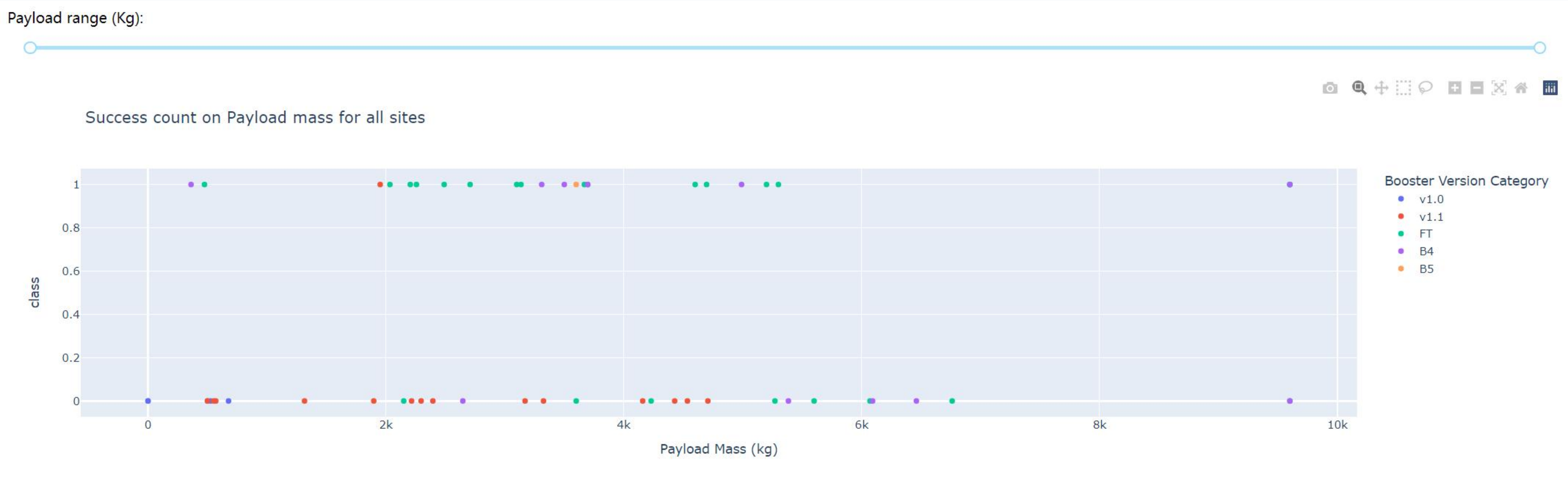
highest launch success ratio

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



Payload vs. Launch Outcome

- FT has the most success rate in payload mass range 2K to 4K
- V1.1 has the most success rate in payload mass under 6K

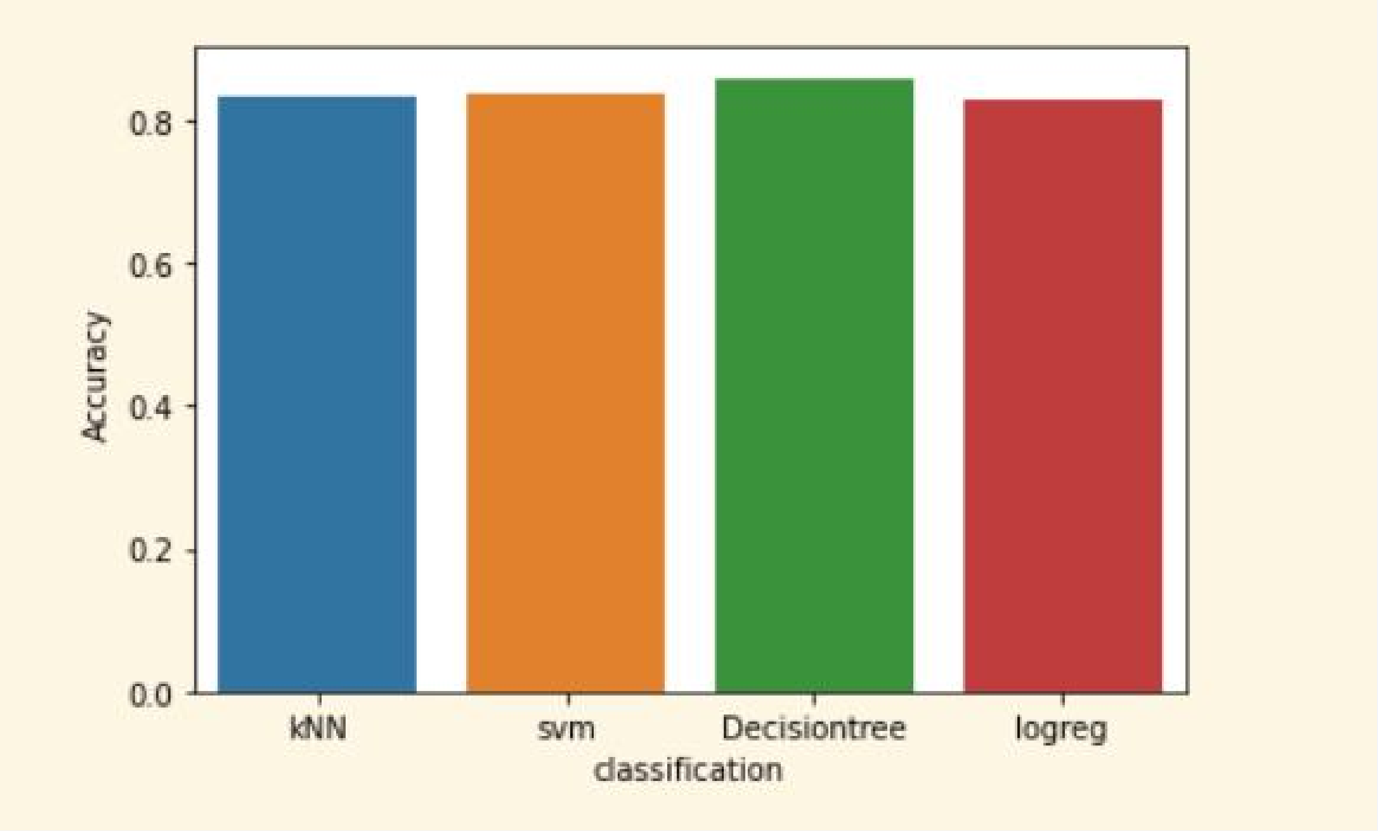


Section 5

Predictive Analysis (Classification)

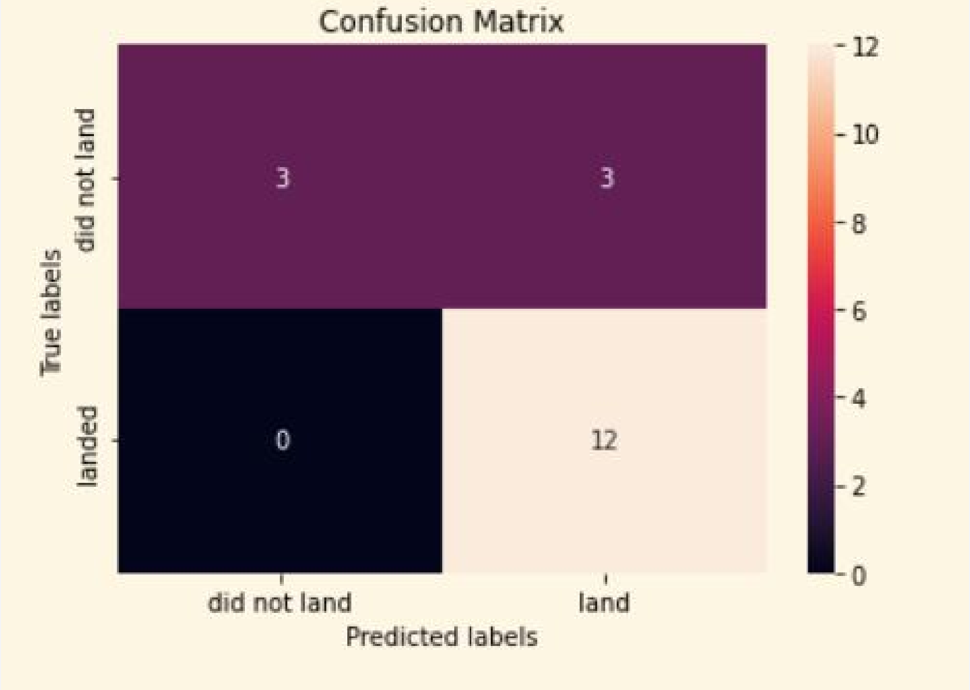
Classification Accuracy

- Decision tree has the hightest accuracy of 0.846.



Confusion Matrix

- Decision tree has the best accuracy.
- The confusion matrix shows in land it has 12 successes in total 15 times, in did not land 100% right.



Conclusions

1. launch sites have very different success rate
2. Booster selection for orbit and payload is very important
3. After years the success rate has increased

Appendix

Bar chart code:

- a=pd.DataFrame({"classification":['kNN','svm','Decisiontree','logreg']})
- b=pd.DataFrame({"Accuracy":[0.833,0.84,0.86,0.83]})
- df=pd.concat([a,b],axis=1)
- sns.barplot(x="classification",y="Accuracy",data=df)

Thank you!