IBM Developer
SKILLS NETWORK

# Winning Space Race with Data Science

Vishal Meena
11/1/25

# Outline

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

# Executive Summary

- Summary of methodologies
  - Data Collection through 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 result
  - Interactive analytics in screenshots
  - Predictive Analytics result

# Introduction

- Project background and context
  - Space X 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 Space X can reuse the first stage. Therefore, if we can 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. This goal of the project is to create a machine learning pipeline to predict if the first stage will land successfully.

- Problems you want to find answers
  - What factors determine if the rocket will land successfully?
  - The interaction amongst various features that determine the success rate of a successful landing.
  - What operating conditions needs to be in place to ensure a successful landing program.

Section 1

# Methodology

# Methodology

## Executive Summary

- Data collection methodology:
  - Data was collected using SpaceX API and web scraping from Wikipedia.
- 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
  - How to build, tune, evaluate classification models

# Data Collection

- The data was collected using various methods

  Data collection was done using get request to the SpaceX API.

  - Next, we decoded the response content as a Json using .json() function call and turn it into a pandas dataframe using .json_normalize().
  - We then cleaned the data, checked for missing values and fill in missing values where necessary.
  - In addition, we performed web scraping from Wikipedia for Falcon 9 launch records with BeautifulSoup.
  - The objective was to extract the launch records as HTML table, parse the table and convert it to a pandas dataframe for future analysis.

# Data Collection – SpaceX API

- Requested and parse the SpaceX launch data using the GET request -> Filtered the dataframe to only include Falcon 9 launches

- the GitHub URL of the completed SpaceX API calls notebook: https://github.com/Vishal-jeph/starship/blob/main/jupyter-labs-spacex-data-collection-api.ipynb as an external reference and peer-review purpose

| | FlightNumber | Date | BoosterVersion | PayloadMass | Orbit | LaunchSite | Outcome | Flights | GridFins | Reused | Legs | LandingPad | Block | ReusedCount | Serial | Longitude | Lati |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **4** | 1 | 2010-06-04 | Falcon 9 | NaN | LEO | CCSFS SLC 40 | None None | 1 | False | False | False | None | 1.0 | 0 | B0003 | -80.577366 | 28.56 |
| **5** | 2 | 2012-05-22 | Falcon 9 | 525.0 | LEO | CCSFS SLC 40 | None None | 1 | False | False | False | None | 1.0 | 0 | B0005 | -80.577366 | 28.56 |
| **6** | 3 | 2013-03-01 | Falcon 9 | 677.0 | ISS | CCSFS SLC 40 | None None | 1 | False | False | False | None | 1.0 | 0 | B0007 | -80.577366 | 28.56 |
| **7** | 4 | 2013-09-29 | Falcon 9 | 500.0 | PO | VAFB SLC 4E | False Ocean | 1 | False | False | False | None | 1.0 | 0 | B1003 | -120.610829 | 34.63 |
| **8** | 5 | 2013-12-03 | Falcon 9 | 3170.0 | GTO | CCSFS SLC 40 | None None | 1 | False | False | False | None | 1.0 | 0 | B1004 | -80.577366 | 28.56 |
| **...** | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | |
| **89** | 86 | 2020-09-03 | Falcon 9 | 15600.0 | VLEO | KSC LC 39A | True ASDS | 2 | True | True | True | 5e9e3032383ecb6bb234e7ca | 5.0 | 12 | B1060 | -80.603956 | 28.60 |
| **90** | 87 | 2020-10-06 | Falcon 9 | 15600.0 | VLEO | KSC LC 39A | True ASDS | 3 | True | True | True | 5e9e3032383ecb6bb234e7ca | 5.0 | 13 | B1058 | -80.603956 | 28.60 |
| **91** | 88 | 2020-10-18 | Falcon 9 | 15600.0 | VLEO | KSC LC 39A | True ASDS | 6 | True | True | True | 5e9e3032383ecb6bb234e7ca | 5.0 | 12 | B1051 | -80.603956 | 28.60 |
| **92** | 89 | 2020-10-24 | Falcon 9 | 15600.0 | VLEO | CCSFS SLC 40 | True ASDS | 3 | True | True | True | 5e9e3033383ecbb9e534e7cc | 5.0 | 12 | B1060 | -80.577366 | 28.56 |
| **93** | 90 | 2020-11-05 | Falcon 9 | 3681.0 | MEO | CCSFS SLC 40 | True ASDS | 1 | True | False | True | 5e9e3032383ecb6bb234e7ca | 5.0 | 8 | B1062 | -80.577366 | 28.56 |

# Data Collection - Scraping

| | 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 v1.07B0003.18 | Failure | 24 January 2021 | 18:45 |
| **1** | 1 | CCAFS | Dragon Spacecraft Qualification Unit | 0 | LEO | NASA | Success | F9 v1.07B0003.18 | Failure | 4 June 2010 | 18:45 |
| **2** | 1 | CCAFS | Dragon | 525 kg | LEO | NASA | Success | F9 v1.07B0003.18 | No attempt\n | 4 June 2010 | 18:45 |
| **3** | 1 | CCAFS | Dragon | 4,700 kg | LEO | NASA | Success\n | F9 v1.07B0003.18 | No attempt | 4 June 2010 | 18:45 |
| **4** | 2 | CCAFS | SpaceX CRS-1 | 4,877 kg | LEO | NASA | Success\n | F9 v1.07B0004.18 | No attempt\n | 4 June 2010 | 15:43 |

- -> Requested the Falcon9 launch WIKI page from url -> Extracted all column/variables names ftom the HTML table header ->created a data frame by parsing the launch HTML tables

- the GitHub URL of the completed web scraping notebook : https://github.com/Vishal-jeph/starship/blob/main/jupyter-labs-webscraping%20(1).ipynb

# Data Wrangling

- Deal with missing values by replaces them with mean values for payload mass column
- the GitHub URL of your completed data wrangling : https://github.com/Vishal-jeph/starship/blob/main/jupyter-labs-spacex-data-collection-api.ipynb

```
FlightNumber       0
Date               0
BoosterVersion     0
PayloadMass        0
Orbit              0
LaunchSite         0
Outcome            0
Flights            0
GridFins           0
Reused             0
Legs               0
LandingPad        26
Block              0
ReusedCount        0
Serial             0
Longitude          0
Latitude           0
dtype: int64
```

# EDA with Data Visualization

- We have made charts to visualize the relation between different variables like:
  - Flight Number and Launch Site
  - Payload Mass and Launch Site
  - Success rate of each orbit type
  - FIlght Number and orbit type
  - Payload Mass and orbit type
  - Launch success yearly trend

- The GitHub URL of your completed EDA with data visualization notebook : https://github.com/Vishal-jeph/starship/blob/main/edadataviz.ipynb

# EDA with SQL

- Some of the SQL queries performed are
  - To Display the names of the unique launch sites in the space mission : select distinct(Launch_Site) from SPACEXTBL
  - To Display 5 records where launch sites begin with the string 'CCA' : select * from SPACEXTBL where LAUNCH_SITE like 'CCA%' limit 5
  - To Display the total payload mass carried by boosters launched by NASA (CRS) : select sum(PAYLOAD_MASS__KG_) from SPACEXTBL where CUSTOMER = "NASA (CRS)"
  - To Display average payload mass carried by booster version F9 v1.1 : select avg(PAYLOAD_MASS__KG_) from SPACEXTBL where Booster_Version like "F9 v1.1"

- The GitHub URL of EDA with SQL notebook : https://github.com/Vishal-jeph/starship/blob/main/jupyter-labs-eda-sql-coursera_sqllite.ipynb

# Build an Interactive Map with Folium

- Used Folium to mark all the launch sites on global mape with markers indicating number success or failuers
- Also Marked the distance between launcg sites and near by coast area or railyway line to get insites about the kind of places choosen to build a launch pad
- we added markes describing failuers and success of the flight to determine what result we get if we launch from a perticular launch pad
- The GitHub URL of interactive map with Folium map : https://github.com/Vishal-jeph/starship/blob/main/lab_jupyter_launch_site_location%20(1).ipynb

# Build a Dashboard with Plotly Dash

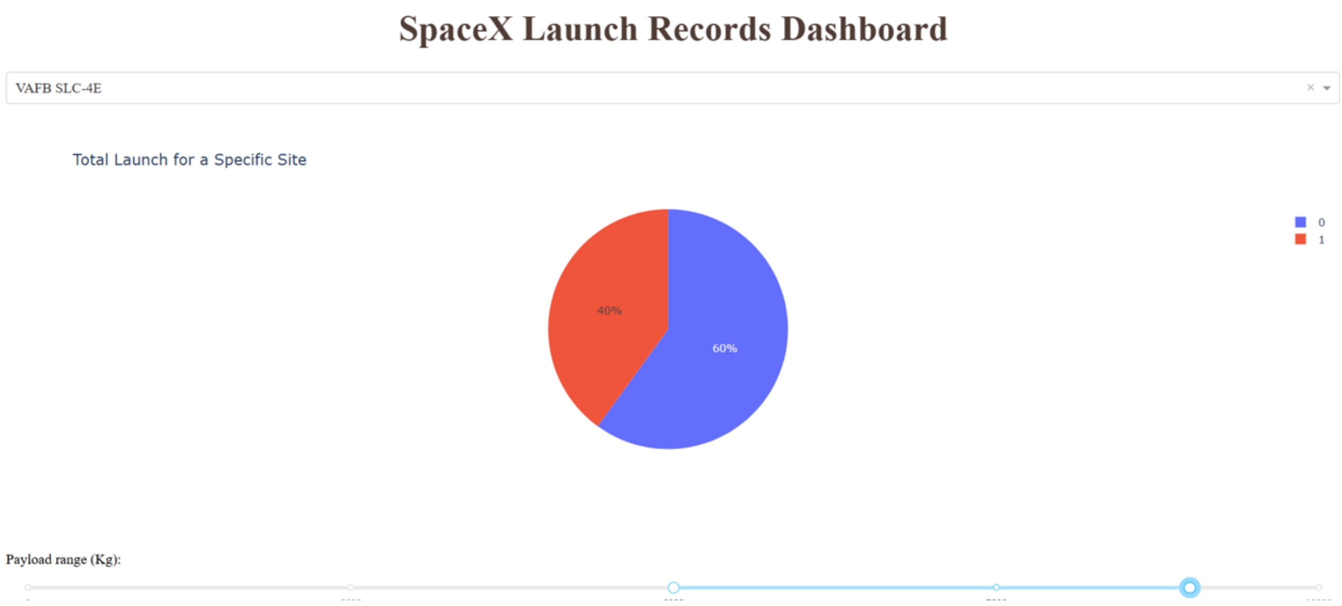
- We used Dash to plot a pie chart describing success and failuer of all the sites individually and as a whole to check the best loaction to launch from
- Also we added an interactive silder to change the value of Paylaod mass and according creating a scatter plot for success or failuer with paylaod mass to check a realationship between them
- the GitHub URL of completed Plotly Dash : https://github.com/Vishal-jeph/starship/blob/main/spacex_dash_app.py

# Predictive Analysis (Classification)

- Started with preprocessing the data by tansforming it using standaredization then split -> the data to training set for traning tht maodel and test set to check accuracy for the created model
- Then created 4 models – logistic regression, SVM, KNN and decision tree to check accuracy and compare them to find best result also for every model which is created we used gridsearchCV for hyperparametring and finding the optimal results
- The GitHub URL of complete predictive analysis : https://github.com/Vishal-jeph/starship/blob/main/SpaceX_Machine%20Learning%20Prediction_Part_5.ipynb

# Results

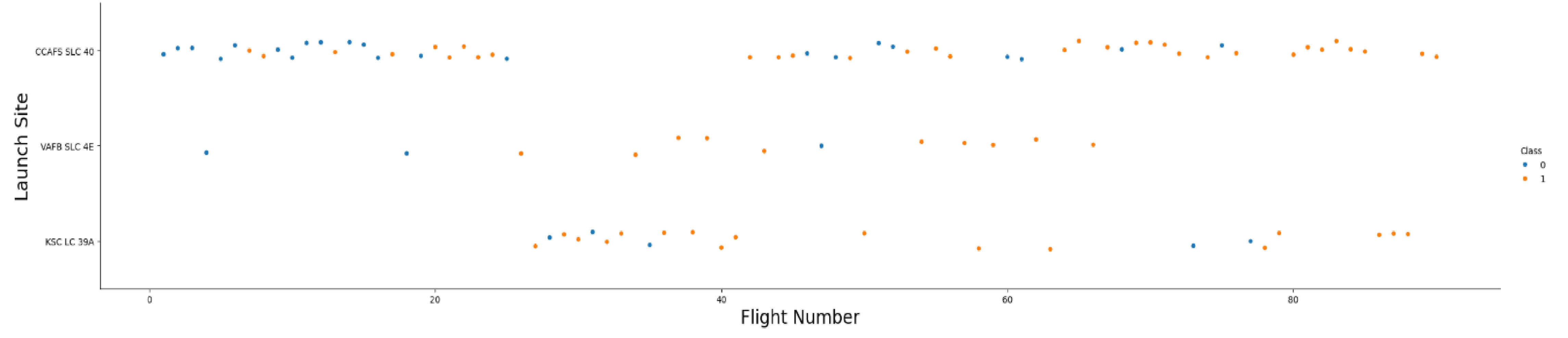
- There were 5 Null values which were replaces by mean value of the column and then created dashboard to showcase the success of launch with payload mass and location
- Interactive analytics demo in screenshots
- After using 4 different models to predict we got the best results from decision tree



Section 2

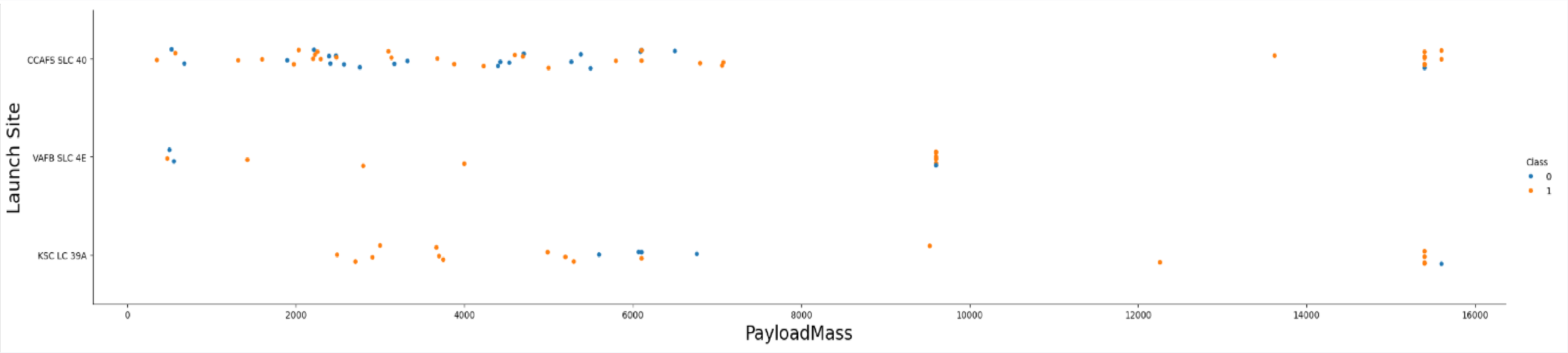
# Insights drawn from EDA

# Flight Number vs. Launch Site




Most of the Flight are launched from CCAFS SLC 40 but more successful results are given by VAFB SLC 4E with lest number of failures

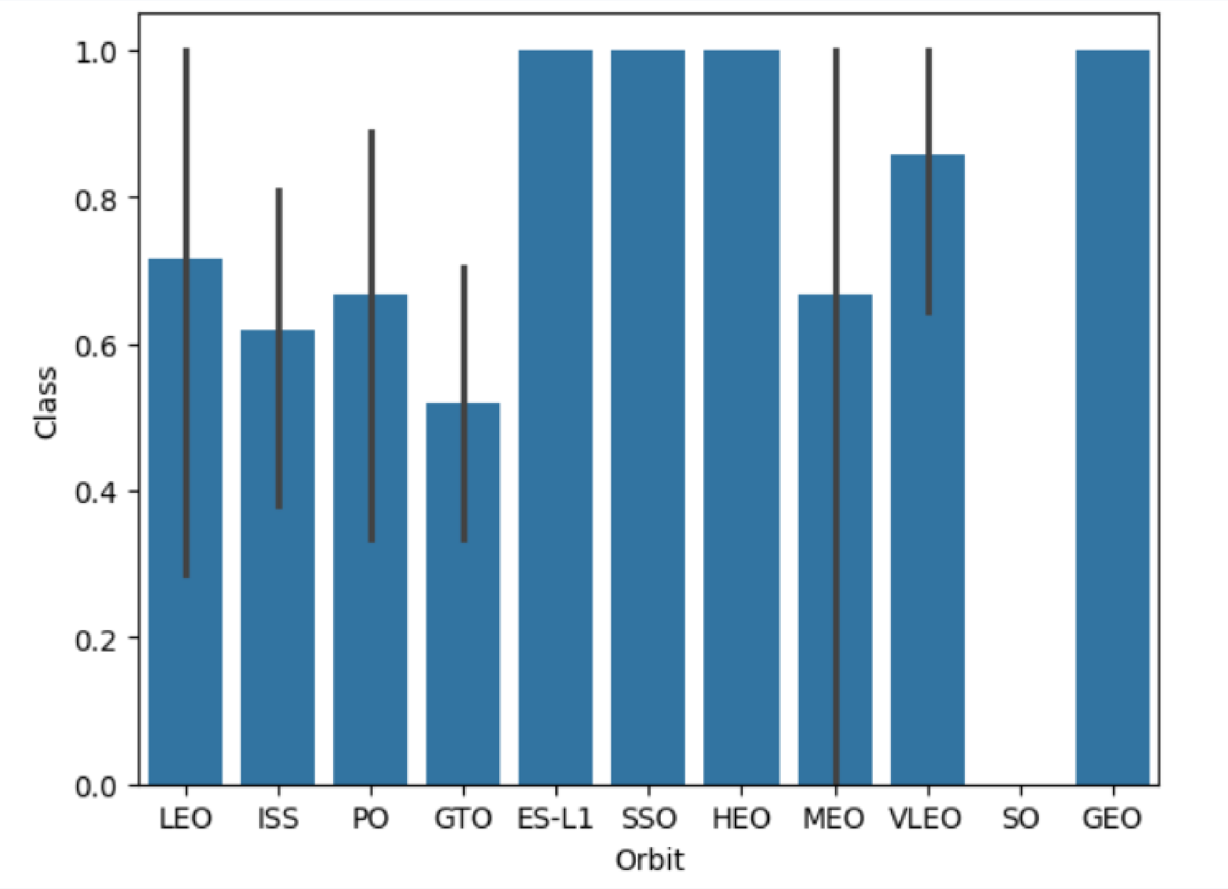
# Payload vs. Launch Site




- With heavy Payload mass there are lot of successful flights so meaning success of flight does not depends on payload
- There are no heavy launches form VAFB SLC 4E launch pad

# Success Rate vs. Orbit Type

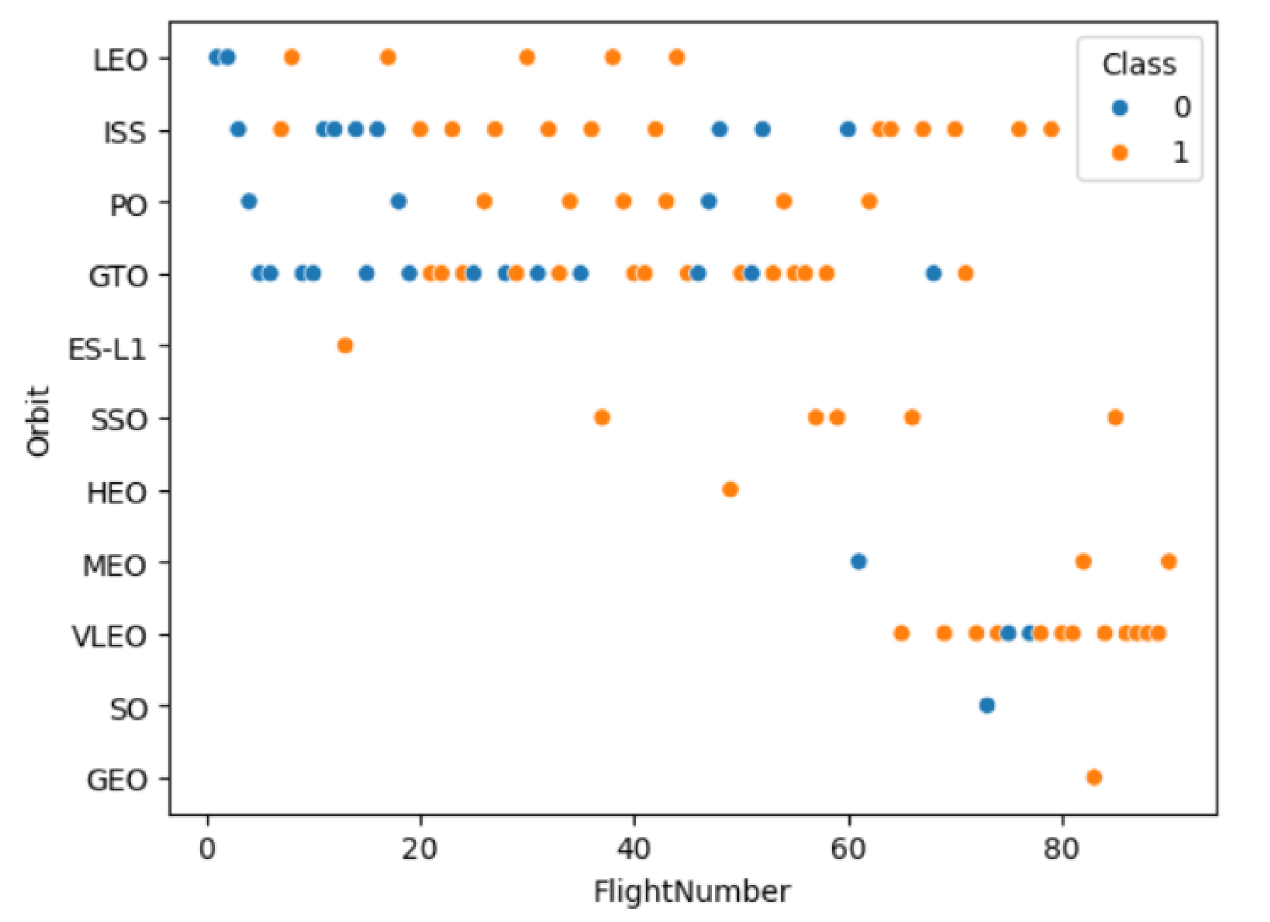
- Full Success rate for orbits ES-l1, SSO,HEO,GEO
- Least for SO



# Flight Number vs. Orbit Type

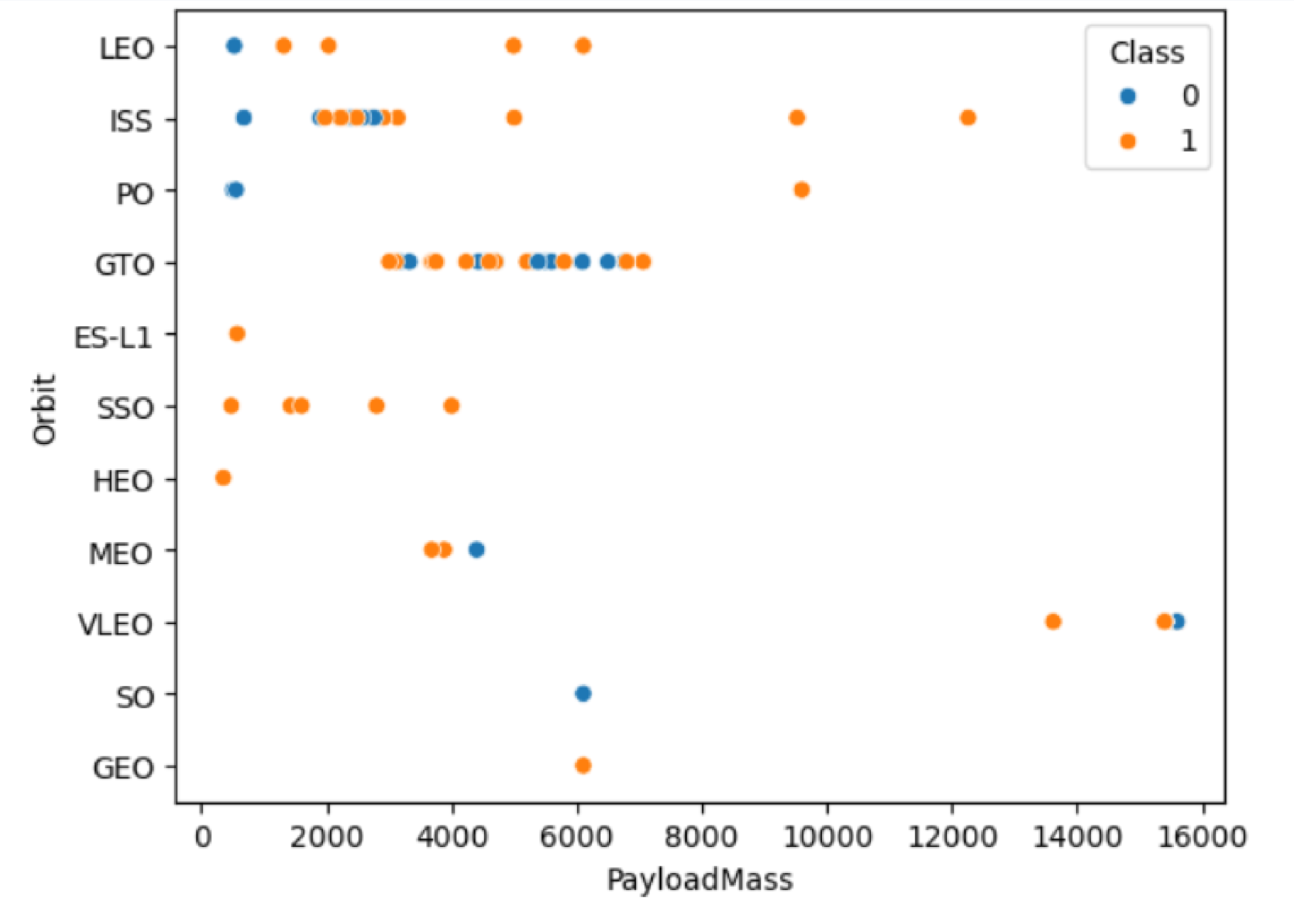
- Max number of filghts are launched for GTO in which the ratio of success and failuer is similer

- Only one flight is launched for SO orbit which also failed



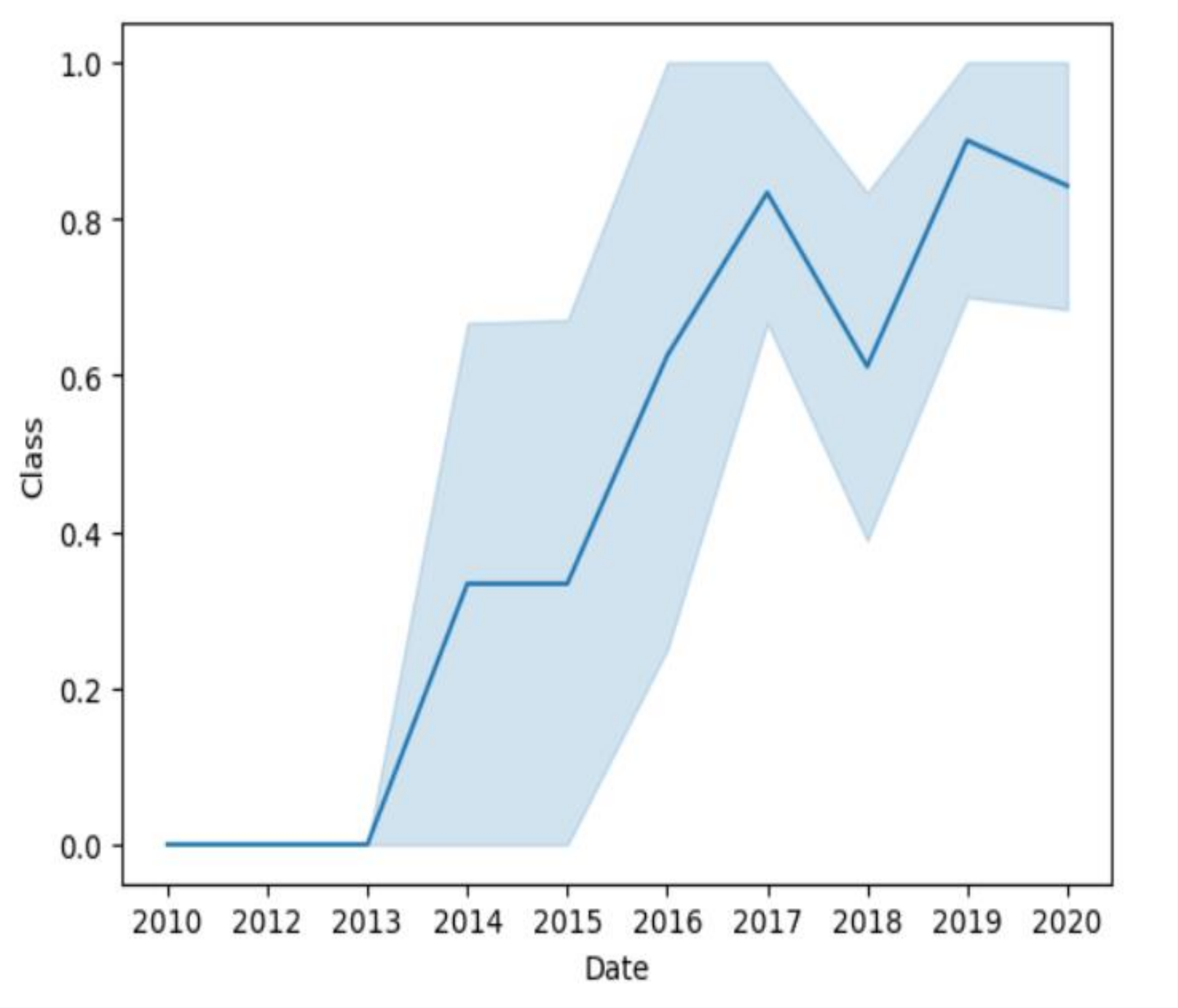
# Payload vs. Orbit Type

- The maximum payload mass was sent to VLEO orbit which was a failuer
- High payload mass does not impact success to launch to ISS orbit



# Launch Success Yearly Trend

- Year on year there is a growth in success of launches
- There was a dip in year 2018-2019 but it was recovered



# All Launch Site Names

- These are the launching sites

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

# Launch Site Names Begin with 'CCA'

Here are 5 records where launch sites begin with `CCA`

| 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 | 7:44:00 | F9 v1.0 B0005 | CCAFS LC-40 | Dragon demo flight C2 | 525 | LEO (ISS) | NASA (COTS) | Success | No attempt |
| 2012-10-08 | 0:35:00 | 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

- the total payload carried by boosters from NASA

| sum(PAYLOAD_MASS__KG_) |
| --- |
| 45596 |

# Average Payload Mass by F9 v1.1

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

| avg(PAYLOAD_MASS__KG_) |
|---|
| 2928.4 |

# First Successful Ground Landing Date

The dates of the first successful landing outcome on ground pad

| min(Date) |
|---|
| 2015-12-22 |

## Successful Drone Ship Landing with Payload between 4000 and 6000

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

| Booster_Version |
|---|
| F9 FT B1022 |
| F9 FT B1026 |
| F9 FT B1021.2 |
| F9 FT B1031.2 |

# Total Number of Successful and Failure Mission Outcomes

- The total number of successful and failure mission outcomes

| count(MISSION_OUTCOME) |
| ---: |
| 99 |

# Boosters Carried Maximum Payload

- Here are the names of the booster which have carried the maximum payload mass

| Booster_Version |
| --- |
| F9 B5 B1048.4 |
| F9 B5 B1049.4 |
| F9 B5 B1051.3 |
| F9 B5 B1056.4 |
| F9 B5 B1048.5 |
| F9 B5 B1051.4 |
| F9 B5 B1049.5 |
| F9 B5 B1060.2 |
| F9 B5 B1058.3 |
| F9 B5 B1051.6 |
| F9 B5 B1060.3 |
| F9 B5 B1049.7 |

# 2015 Launch Records

- Here are list of the failed landing_outcomes in drone ship, their booster versions, and launch site names for in year 2015

| substr(Date, 6,2) | Mission_Outcome | Landing_Outcome | Booster_Version | Launch_Site |
|---|---|---|---|---|
| 01 | Success | Failure (drone ship) | F9 v1.1 B1012 | CCAFS LC-40 |
| 04 | Success | Failure (drone ship) | F9 v1.1 B1015 | CCAFS LC-40 |

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

Here are the rank of 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

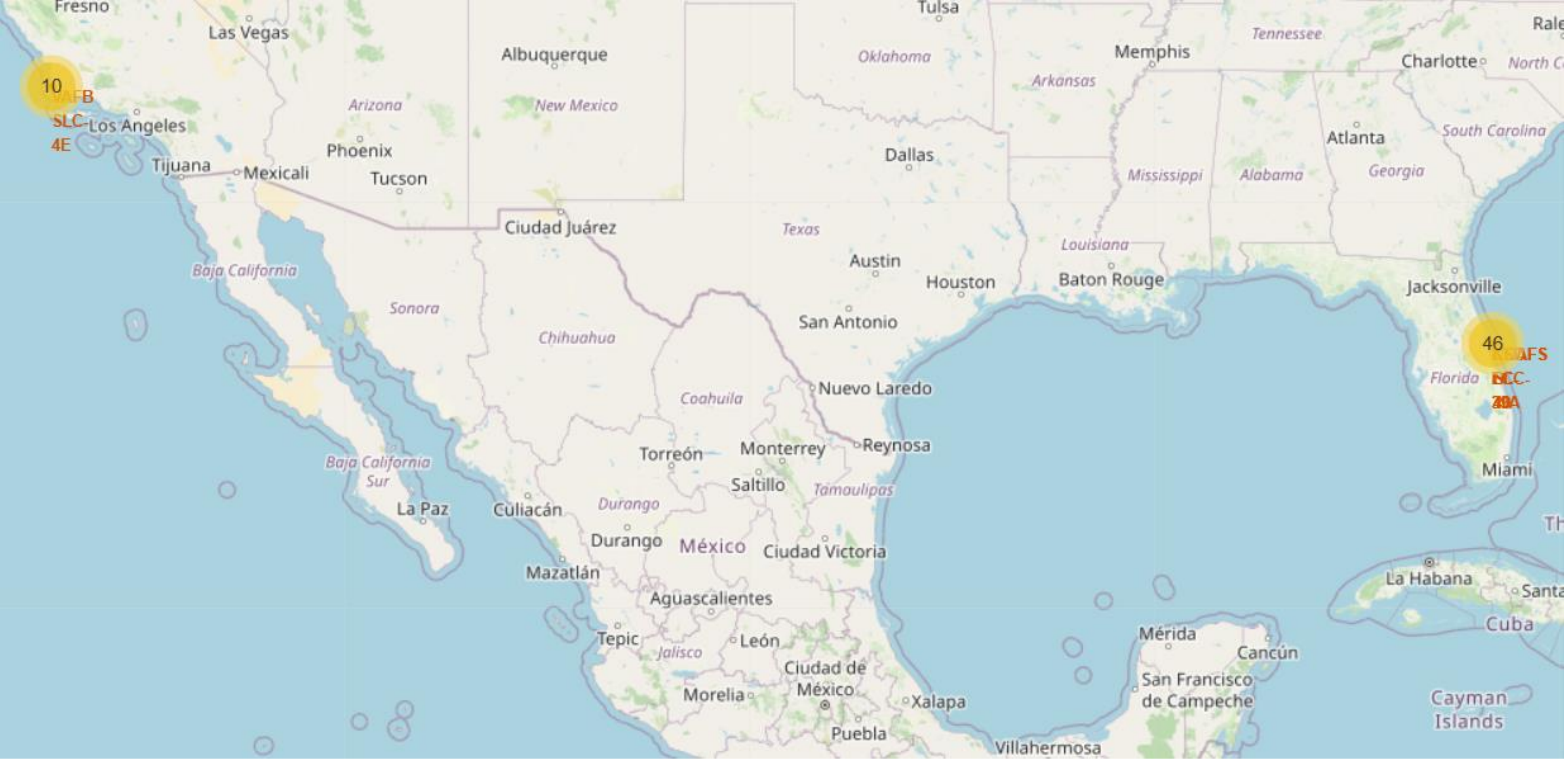
| Landing_Outcome | numbers |
|---:|---:|
| No attempt | 10 |
| Success (drone ship) | 5 |
| Failure (drone ship) | 5 |
| Success (ground pad) | 3 |
| Controlled (ocean) | 3 |
| Uncontrolled (ocean) | 2 |
| Failure (parachute) | 2 |
| Precluded (drone ship) | 1 |

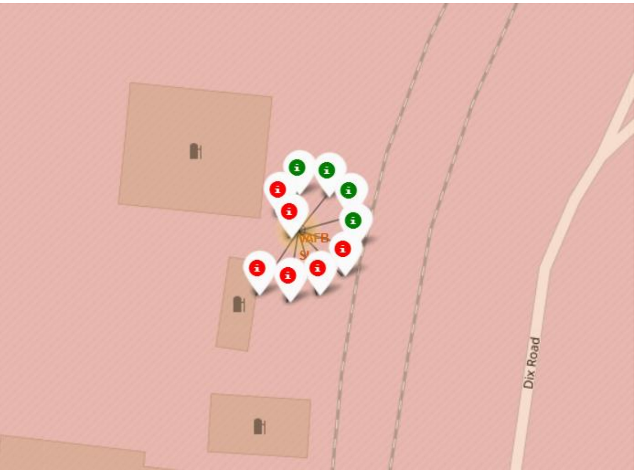
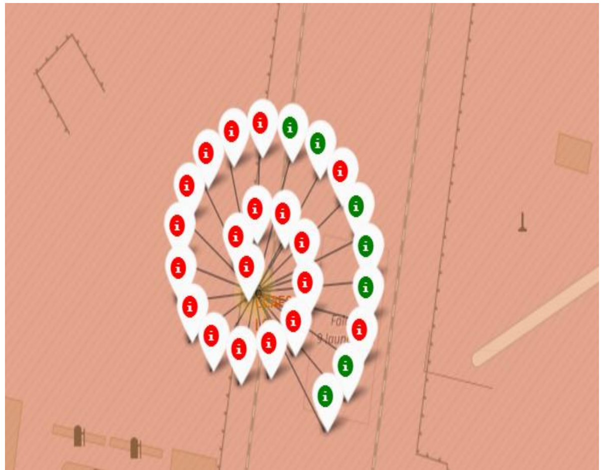
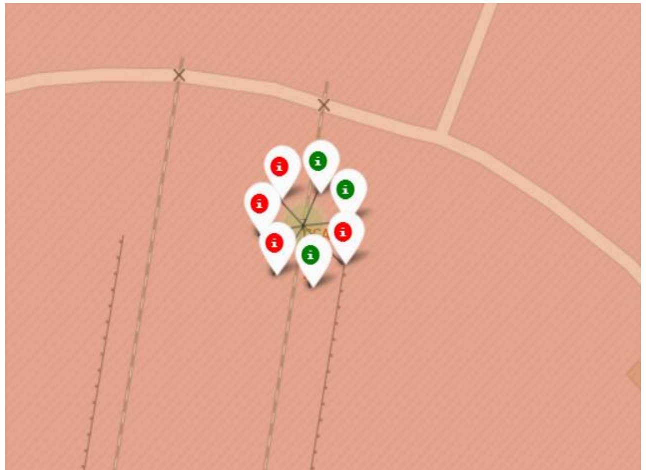
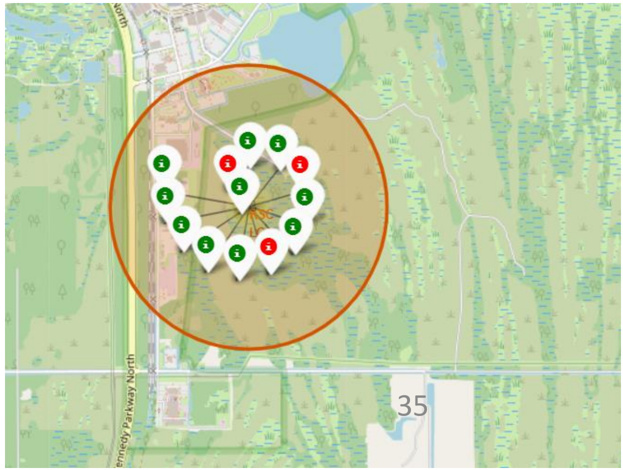
Section 3

# Launch Sites Proximities Analysis

# All launching Sites and the color-labeled launch outcomes

- There are only two major places for all launch pads
- All the launching pads are near ocean
- the division of launch site are 10 and 46.

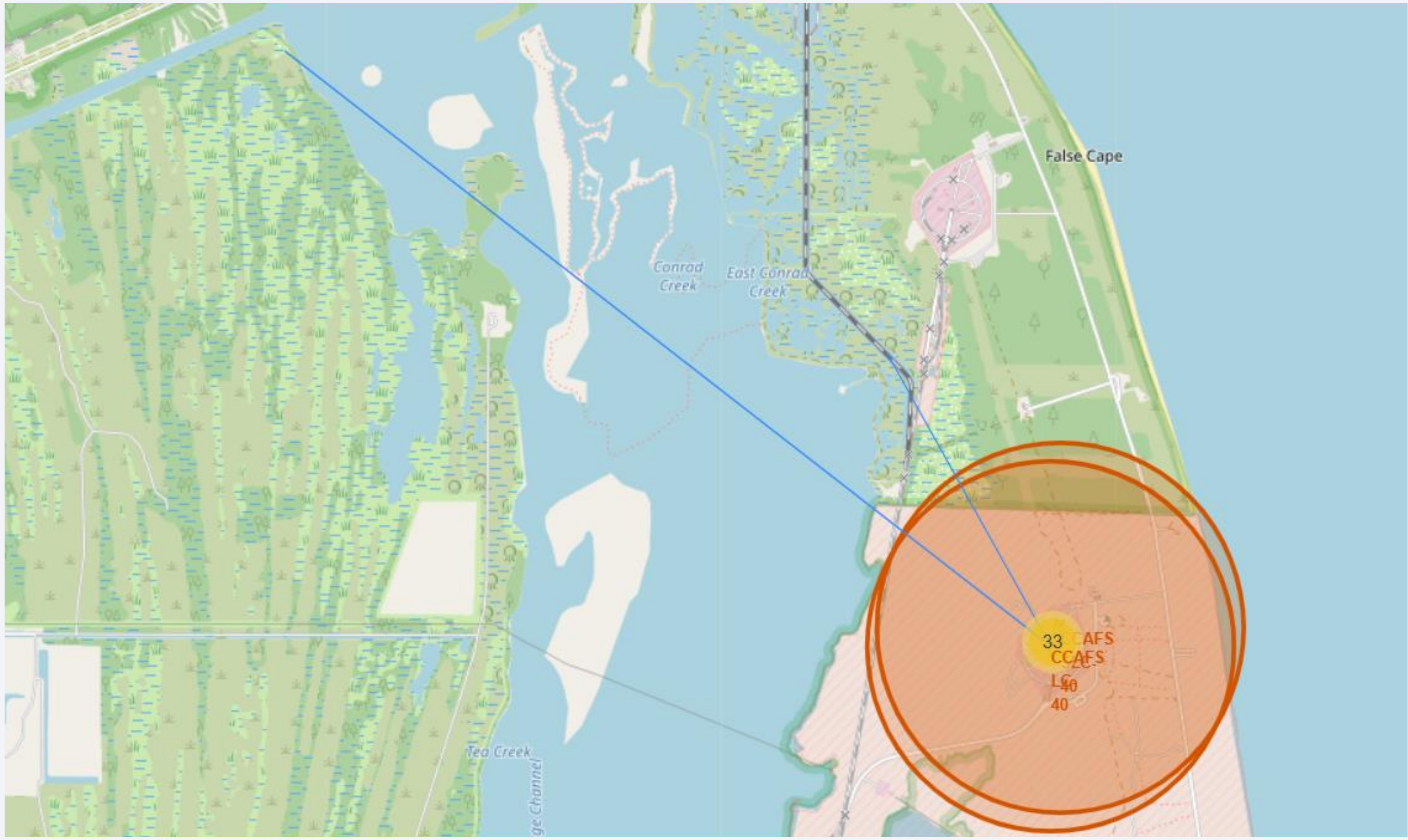


# launch site to its proximities railway, and coastline

- launch site to its proximities railway, and coastline

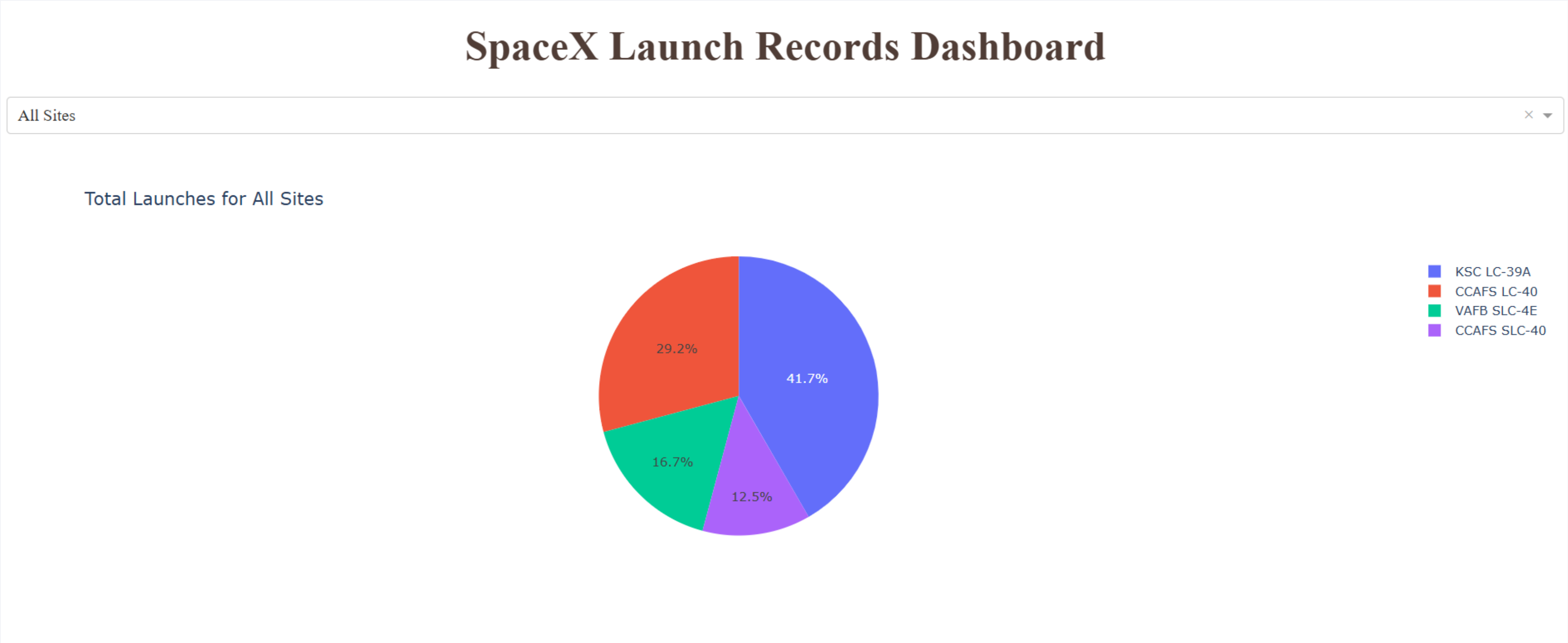
- The launch pads have both railway and coastline close to it as there is a high requirement to bring mattrial for which railway can be used



Section 4

# Build a Dashboard with Plotly Dash

# launch success count for all sites

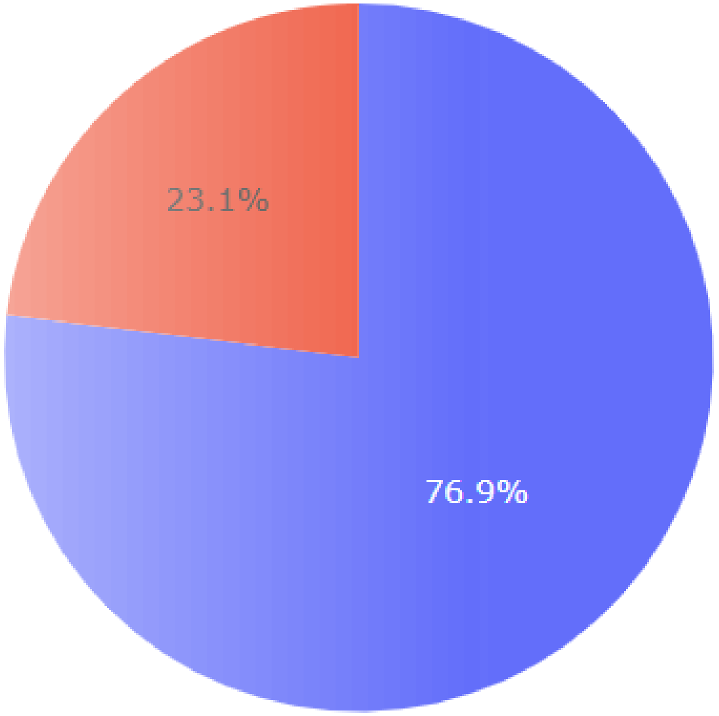



- The screenshot of launch success count for all sites, in a piechart
- Maximum launches are from KSC LC-39A which is 41.7%

# Launch site with highest launch success ration

- Here is the  piechart for the launch site with highest launch success ratio

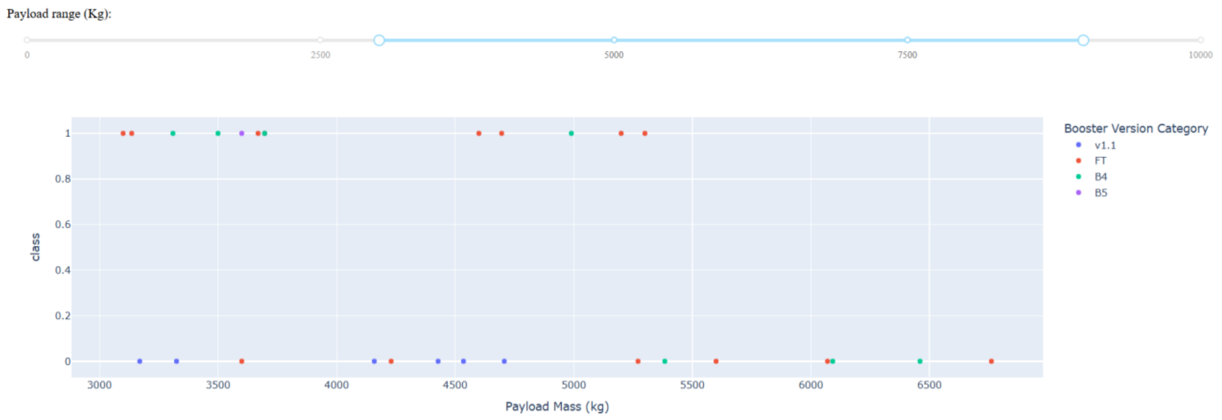
- This pie chart is of KSC LC-39A which has 76.9% success rate

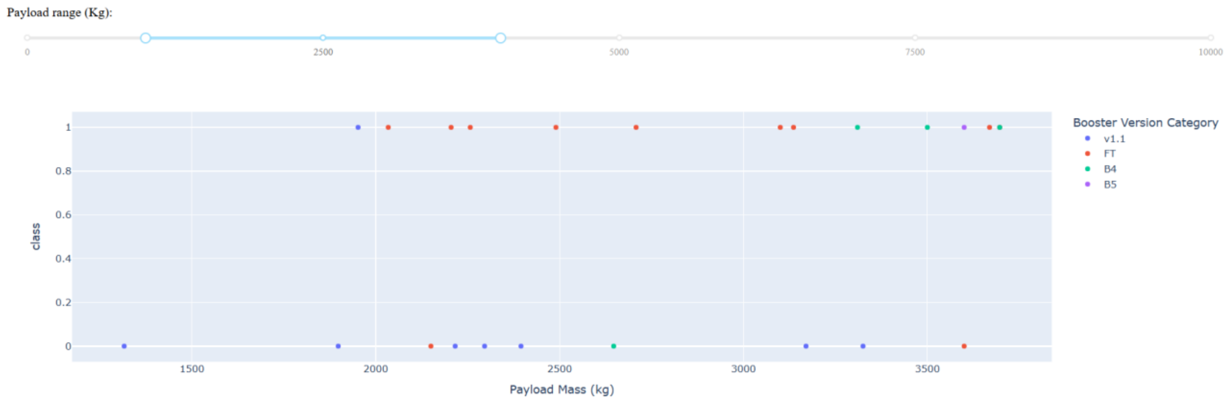


# Payload vs. Launch Outcome scatter plot for all sites, with different payload selected in the range slider

- Shown plot of Payload vs. Launch  scatter plot for all sites, with different payload selected in the range slider

- Showcasing the working of silder to identifly the impact of playload range on scatter plot of payload vs. Launch outcome



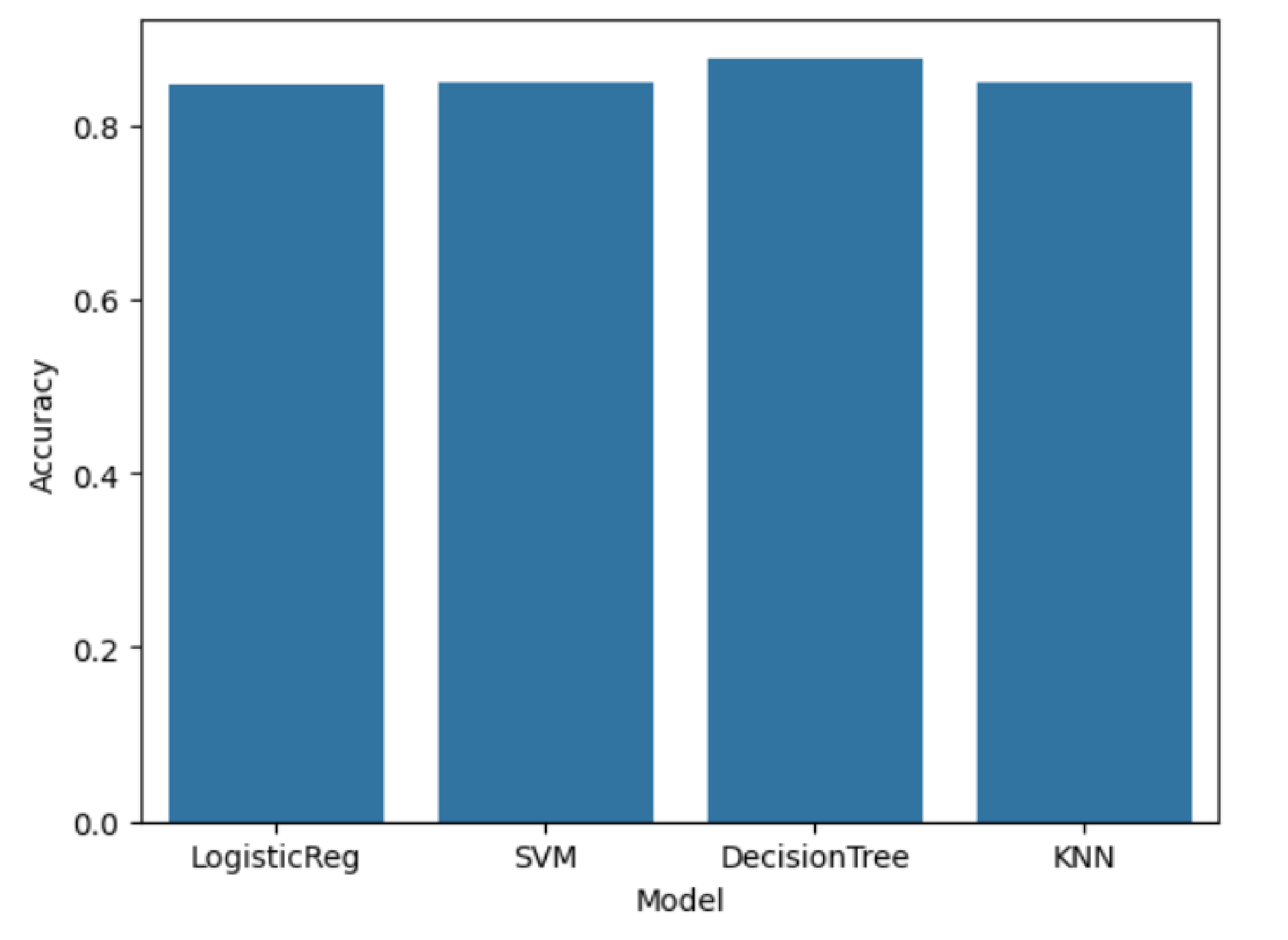



Section 5

# Predictive Analysis (Classification)

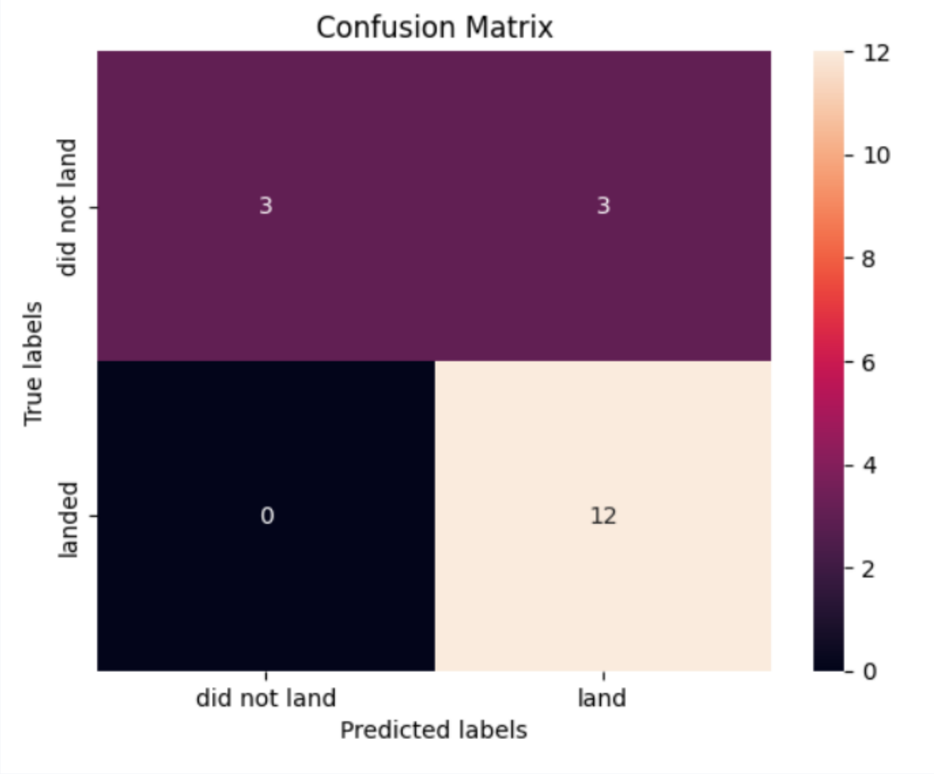
# Classification Accuracy

- Decision Tree has the higest accuracy



# Confusion Matrix

- The confusion matrix of the best performing model which is Decision Tree as we only predicted 3 wrong results



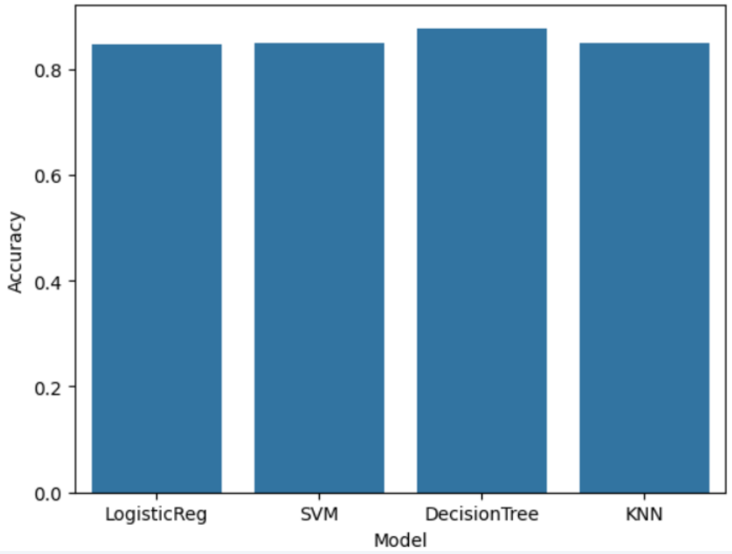
# Conclusions

- Payload mass doesnot impact the success of launch
- Best model to predict on our data set is decision tree
- KSC LC-39A launch pad gives best result on launching a rocket
- Avg. payload which was given to any ship was 2928.4
- All launch pads are constructed near coast areas

# Appendix

- The final conculsion on prediction model

```python
data1 = {'Name':['LogisticReg', 'SVM', 'DecisionTree','KNN'], 'Acc':[logreg_cv.best_score_,svm_cv.best_score_,tree_cv.best_
x = pd.DataFrame(data1)
sns.barplot(data = data1, x = 'Name', y='Acc')
plt.xlabel('Model')
plt.ylabel('Accuracy')
plt.show()
```



Thank you!