IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Ksshitij Singare
17.2.2024

Outline

Executive Summary

- Utilized data sourced from both the public SpaceX API and the SpaceX Wikipedia page to create a labeled dataset, categorizing successful landings. Employed SQL queries, visualizations, folium maps, and dashboards for exploratory data analysis. Extracted pertinent features and encoded categorical variables into binary format. Standardized the data and employed GridSearchCV to optimize parameters for four machine learning models.
- Developed four machine learning models—Logistic Regression, Support Vector Machine, Decision Tree Classifier, and K Nearest Neighbors—using the prepared dataset. Despite achieving a consistent accuracy rate of approximately 83.33%, all models tended to over-predict successful landings. Recognized the need for additional data to refine model performance and enhance accuracy in predicting successful landings.

Introduction

- Background: We're amidst the commercial space age, where SpaceX leads with the best pricing at $62 million compared to $165 million USD. This is largely attributed to their capability to recover a portion of the rocket, specifically Stage 1. Now, Space Y aims to enter the competition against SpaceX.
- Problem: Space Y has assigned us the task of training a machine learning model to forecast the successful recovery of Stage 1 rockets.

Section 1

Methodology

Methodology

Executive Summary

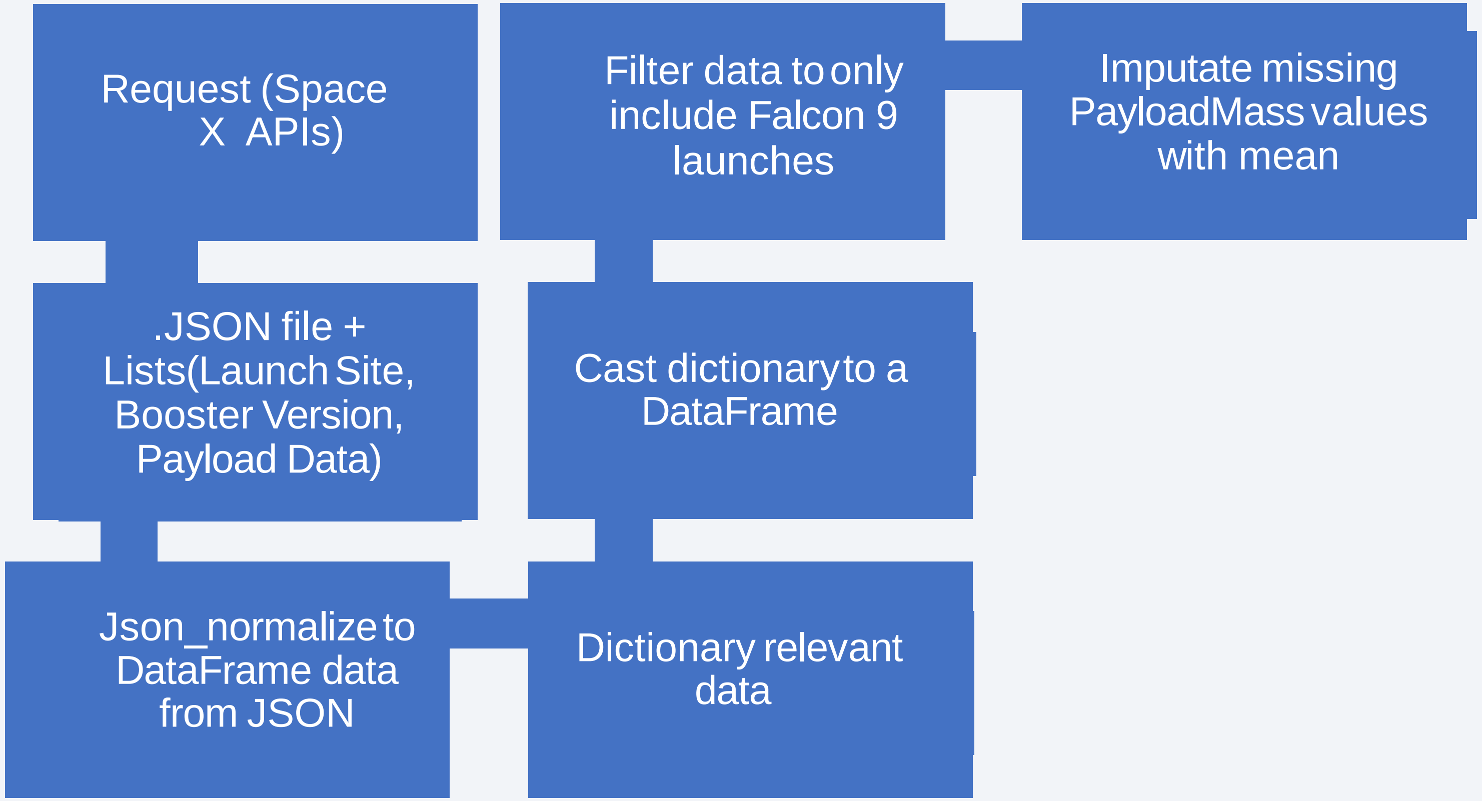
- Data collection methodology:
    - Combined data from SpaceX public API and SpaceX Wikipedia page
- Perform data wrangling
    - Classifying true landings as successful and unsuccessful otherwise
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
    - Trained models using GridSearchCV

Data Collection

- Data was collected through a dual approach: firstly, by making API requests to SpaceX's public API, and secondly, by scraping data from a table within SpaceX's Wikipedia page.
- The SpaceX API provided data with columns such as FlightNumber, Date, BoosterVersion, PayloadMass, Orbit, LaunchSite, Outcome, Flights, GridFins, Reused, Legs, LandingPad, Block, ReusedCount, Serial, Longitude, and Latitude.
- On the other hand, the Wikipedia web scraping process retrieved data with columns including Flight No., Launch site, Payload, PayloadMass, Orbit, Customer, Launch outcome, Version Booster, Booster landing, Date, and Time.
- The next slide will illustrate the flowchart detailing the data collection process from the API, while the subsequent one will depict the flowchart outlining the data collection from web scraping.

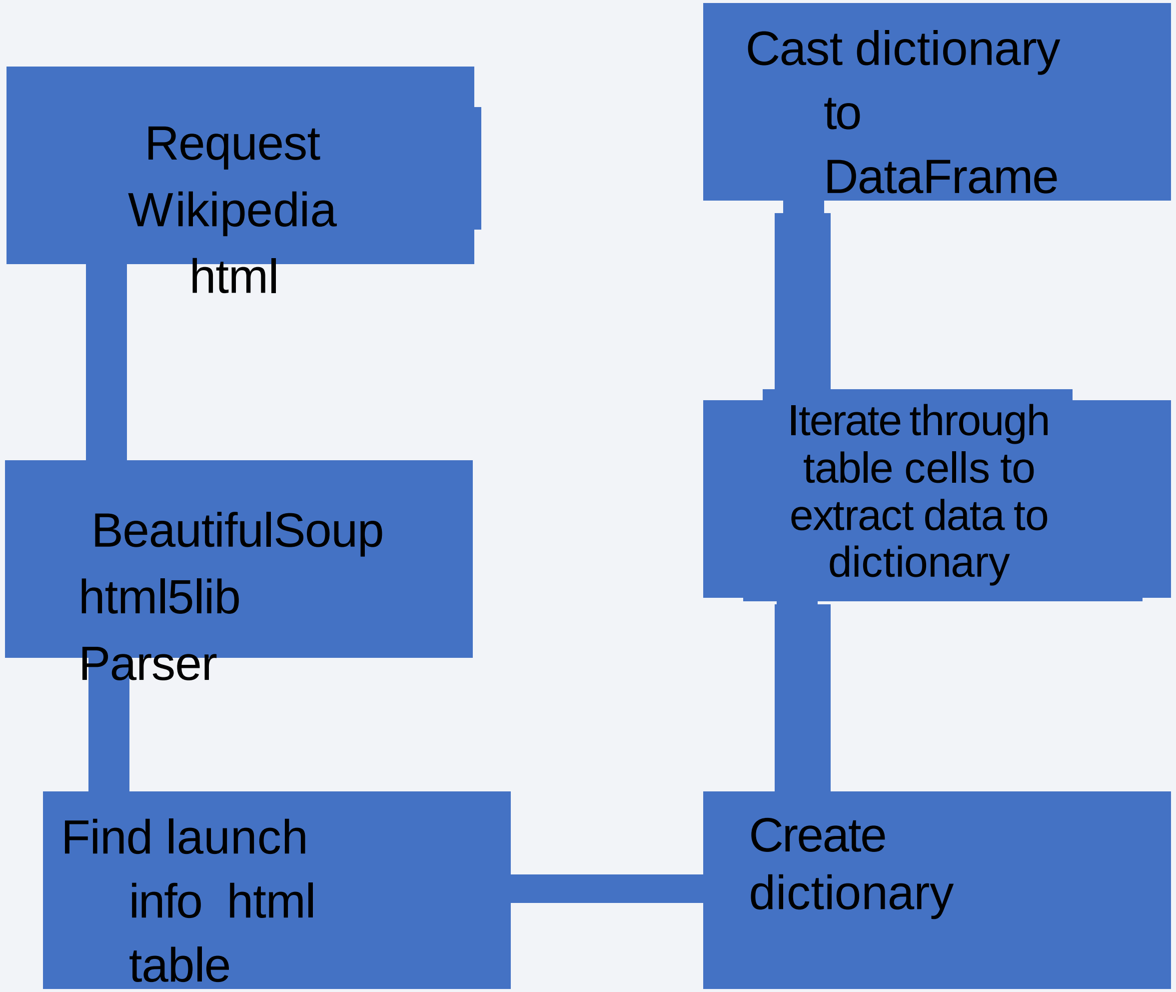
Data Collection – SpaceX API

- https://github.com/Ksshitij-Singare/IBM_DataScience_Professional_Proj./blob/main/Data_Collection_API.ipynb



Data Collection - Scraping

- https://github.com/Ksshitij-Singare/IBM_DataScience_Professional_Proj./blob/main/Data_Collection_WebScraping.ipynb



Data Wrangling

1. We're establishing a training label to distinguish between successful (1) and unsuccessful (0) landings.
2. The 'Outcome' column comprises two components: 'Mission Outcome' and 'Landing Location'.
3. Introducing a new column called 'class' to represent the training label.
4. Assigning a value of 1 to 'class' if the 'Mission Outcome' is True, indicating success.
5. Setting 'class' to 0 otherwise, encompassing scenarios such as None or False outcomes for ASDS, Ocean, and RTLS landings.

GitHub Link:

https://github.com/Ksshitij-Singare/IBM_DataScience_Professional_Proj./blob/main/Data_Wrangling.ipynb

EDA with Data Visualization

- Exploratory Data Analysis performed on variables Flight Number, Payload Mass, Launch Site, Orbit, Class and Year.
- <u>Plots Used:</u>
- Flight Number vs. Payload Mass, Flight Number vs. Launch Site, Payload Mass vs. Launch Site, Orbit vs. Success Rate, Flight Number vs. Orbit, Payload vs Orbit, and Success Yearly Trend
- Scatter plots, line charts, and bar plots were used to compare relationships between variables to
- decide if a relationship exists so that they could be used in training the machine learning model

GitHub Link:

https://github.com/Ksshitij-Singare/IBM_DataScience_Professional_Proj./blob/main/EDA%20Visualization.ipynb

EDA with SQL

- Loaded data set into IBM DB2 Database.
- Queried using SQL Python integration.
- Queries were made to get a better understanding of the dataset.
- Queried information about launch site names, mission outcomes, various pay load sizes of customers and booster versions, and landing outcomes

GitHub Link:

https://github.com/Ksshitij-Singare/IBM_DataScience_Professional_Proj./blob/main/EDA%20SQL.ipynb

Build an Interactive Map with Folium

•Folium maps showcase Launch Sites, successful and unsuccessful landings, and proximity to key locations such as Railway, Highway, Coast, and City.
•This visualization helps to understand the rationale behind the choice of launch site locations.
•Additionally, it provides a visual representation of successful landings in relation to nearby features, aiding in analysis and decision-making

GitHub Link:
https://github.com/Ksshitij-Singare/IBM_DataScience_Professional_Proj./blob/main/Visual%20Analytics%20with%20Folium.ipynb

Build a Dashboard with Plotly Dash

- The dashboard comprises a pie chart and a scatter plot.
- The pie chart allows users to toggle between displaying the distribution of successful landings across all launch sites and viewing the success rates of individual launch sites.
- Users can interact with the scatter plot by selecting either all sites or a specific site and adjusting a slider to explore payload mass ranging from 0 to 10000 kg.
- The pie chart serves to visualize launch site success rates.
- Meanwhile, the scatter plot facilitates analysis of how success rates vary across launch sites, payload masses, and booster version categories.

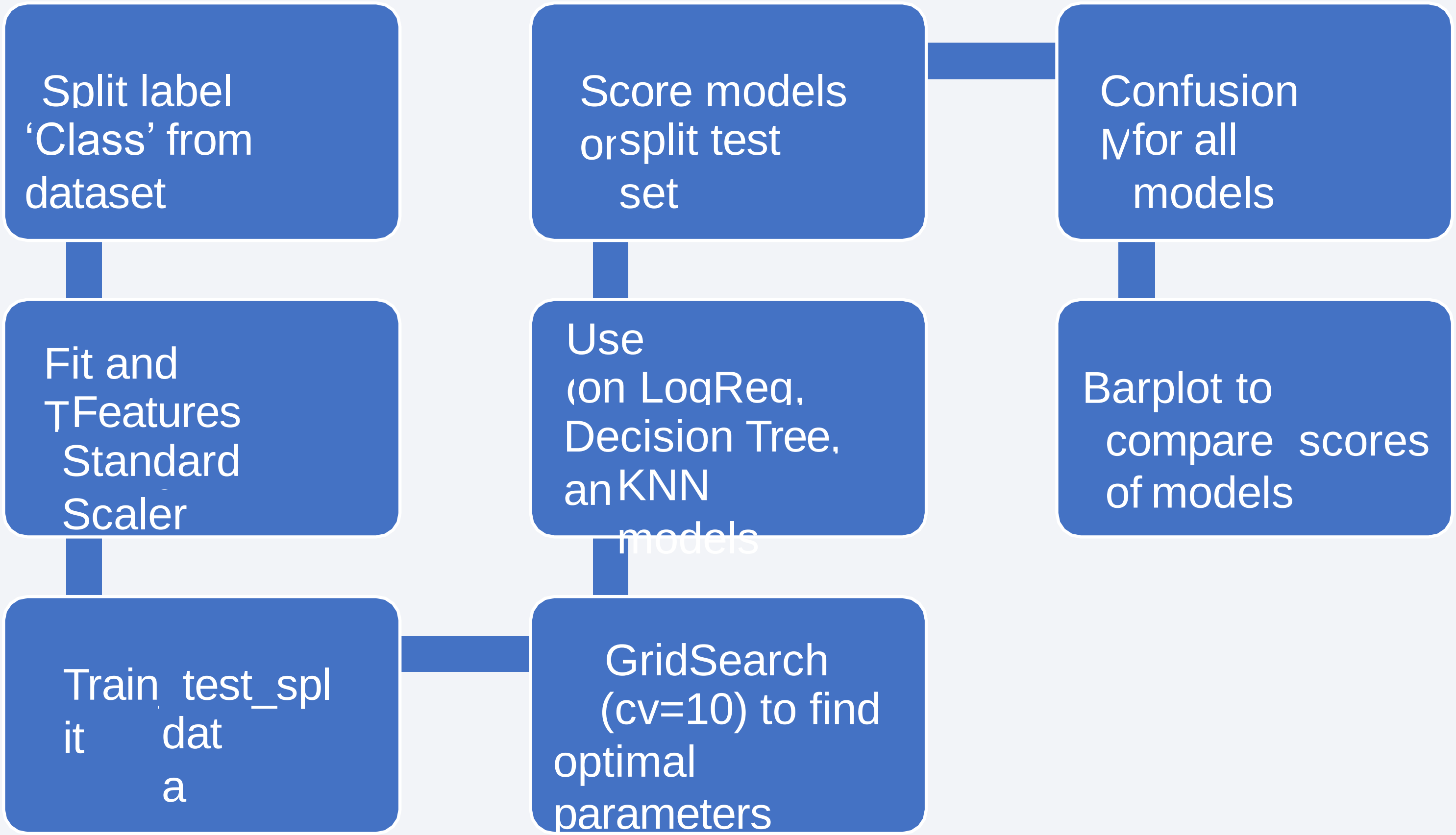
GitHub Link:

https://github.com/Ksshitij-Singare/IBM_DataScience_Professional_Proj./blob/main/spacex_dash_app.py

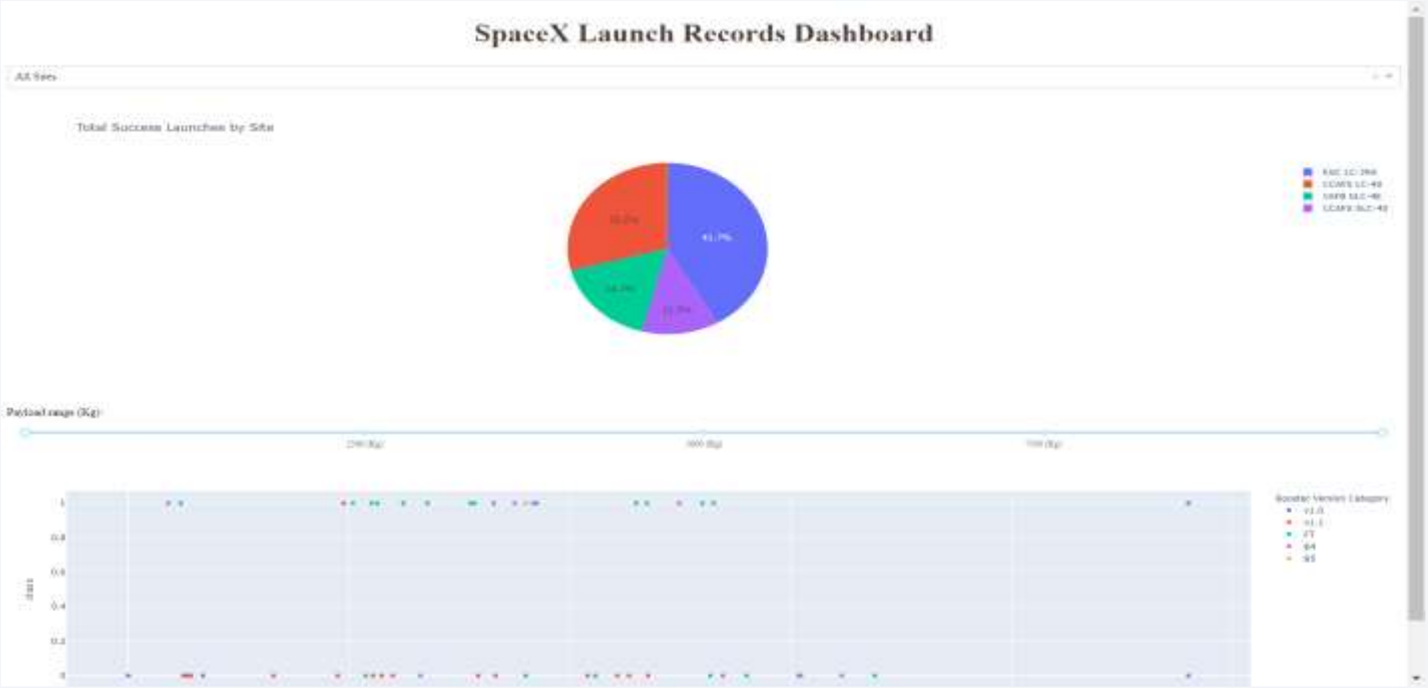
Predictive Analysis (Classification)

- GitHub Link:

https://github.com/Ksshitij-Singare/IBM_DataScience_Profesional_Proj./blob/main/Predictive%20analysis.ipynb



Results

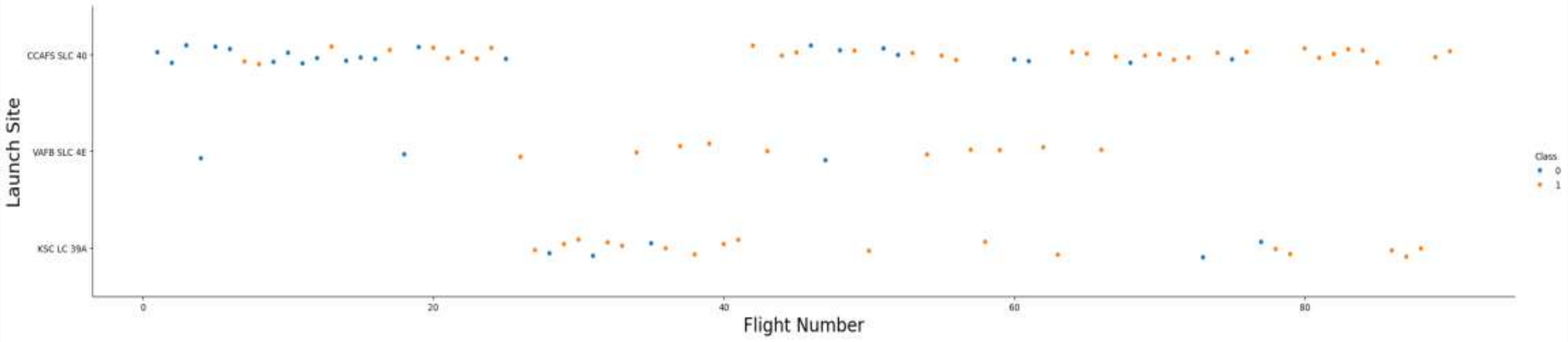


- Exploratory data analysis results
  Dashboard _

Section 2

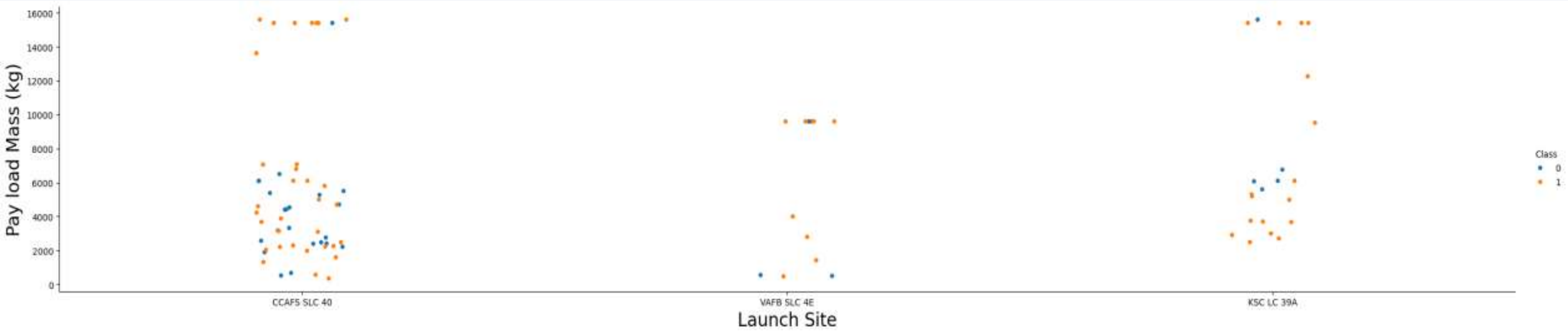
Insights drawn from EDA

Flight Number vs. Launch Site



- Graphic suggests an increase in success rate over time (indicated in Flight Number). Likely a big breakthrough around flight 20 which significantly increased success rate. CCAFS appears to be the main launch site as it has the most volume.

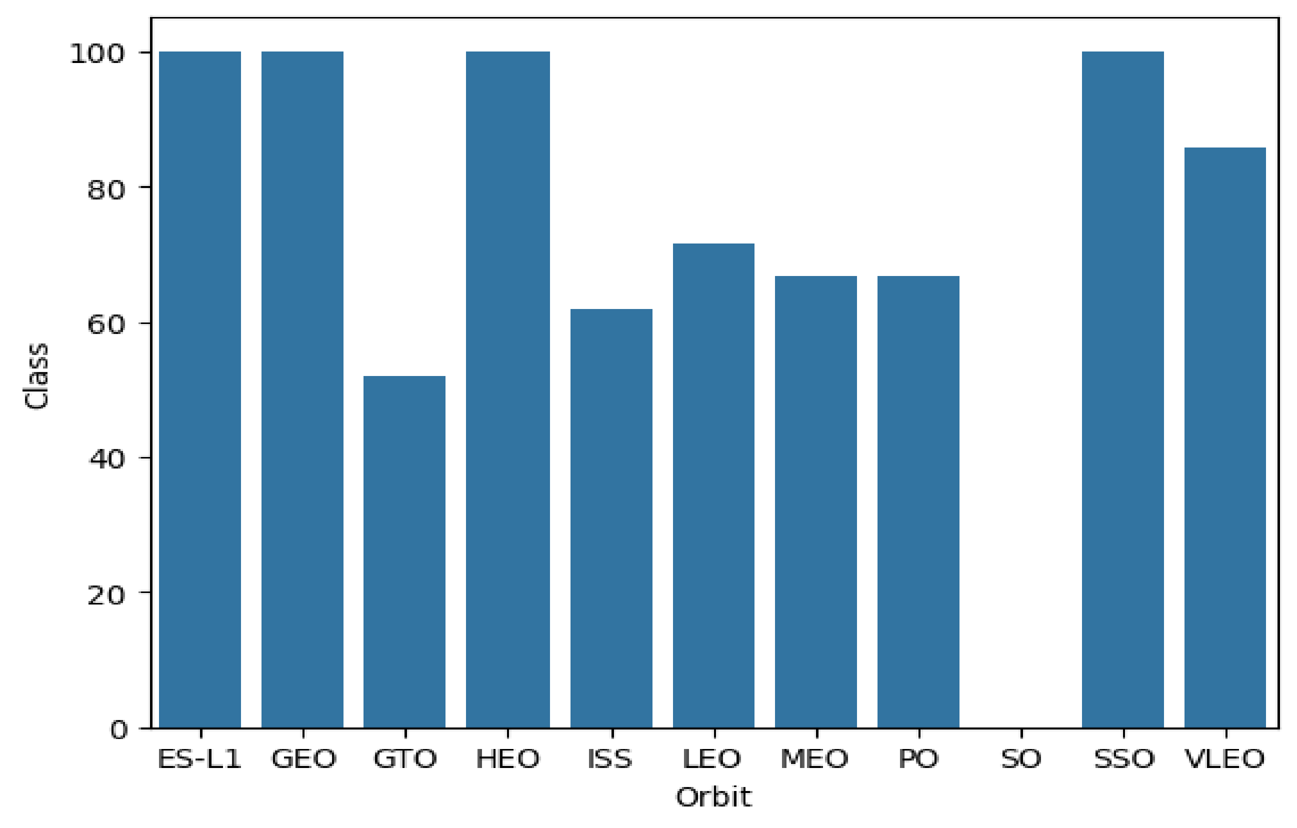
Payload vs. Launch Site



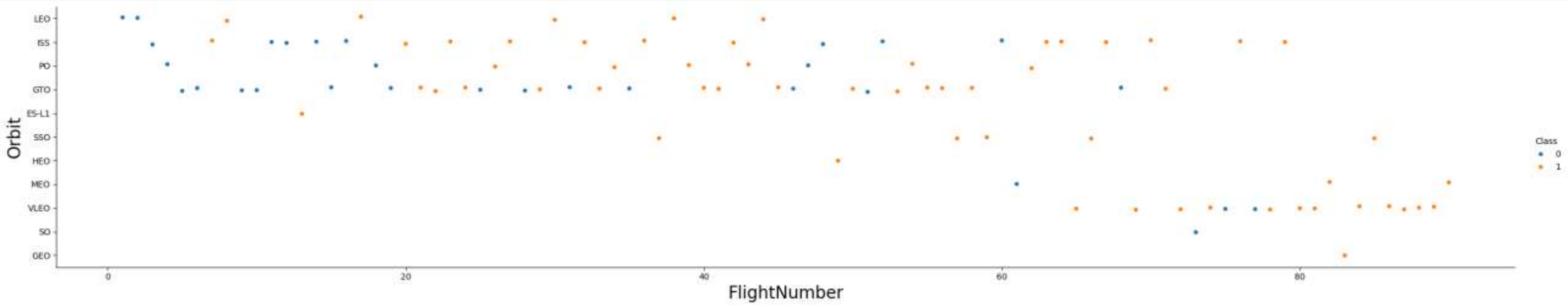
- Payload mass appears to fall mostly between 0-6000 kg. Different launch sites also seem to use different payload mass.

Success Rate vs. Orbit Type

- ES-L1 (1), GEO (1), HEO (1) have 100% success rate (sample sizes in parenthesis) SSO (5) has 100% success rate
- VLEO (14) has decent success rate and attempts
- SO (1) has 0% success rate
- GTO (27) has the around 50% success rate but largest sample

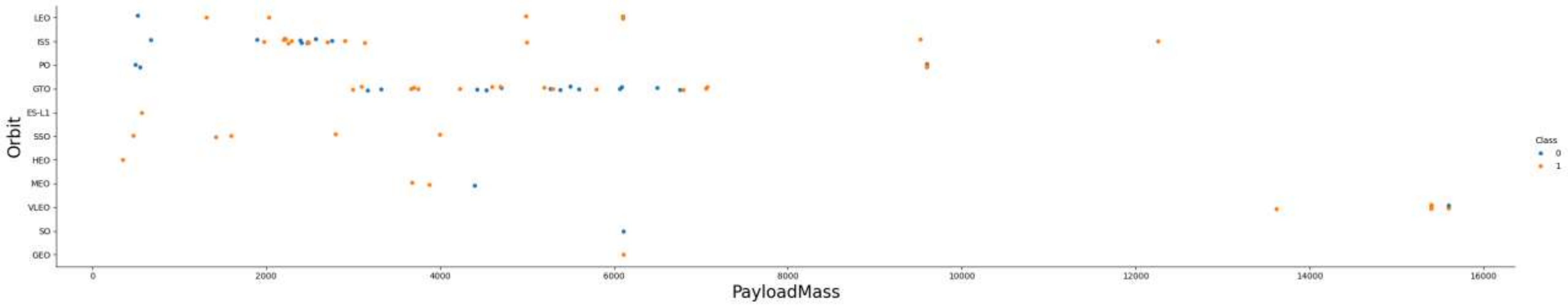


Flight Number vs. Orbit Type



- Launch Orbit preferences changed over Flight Number. Launch Outcome seems to correlate with this preference.
- SpaceX started with LEO orbits which saw moderate success LEO and returned to VLEO in recent launches SpaceX appears to perform better in lower orbits or Sun-synchronous orbits

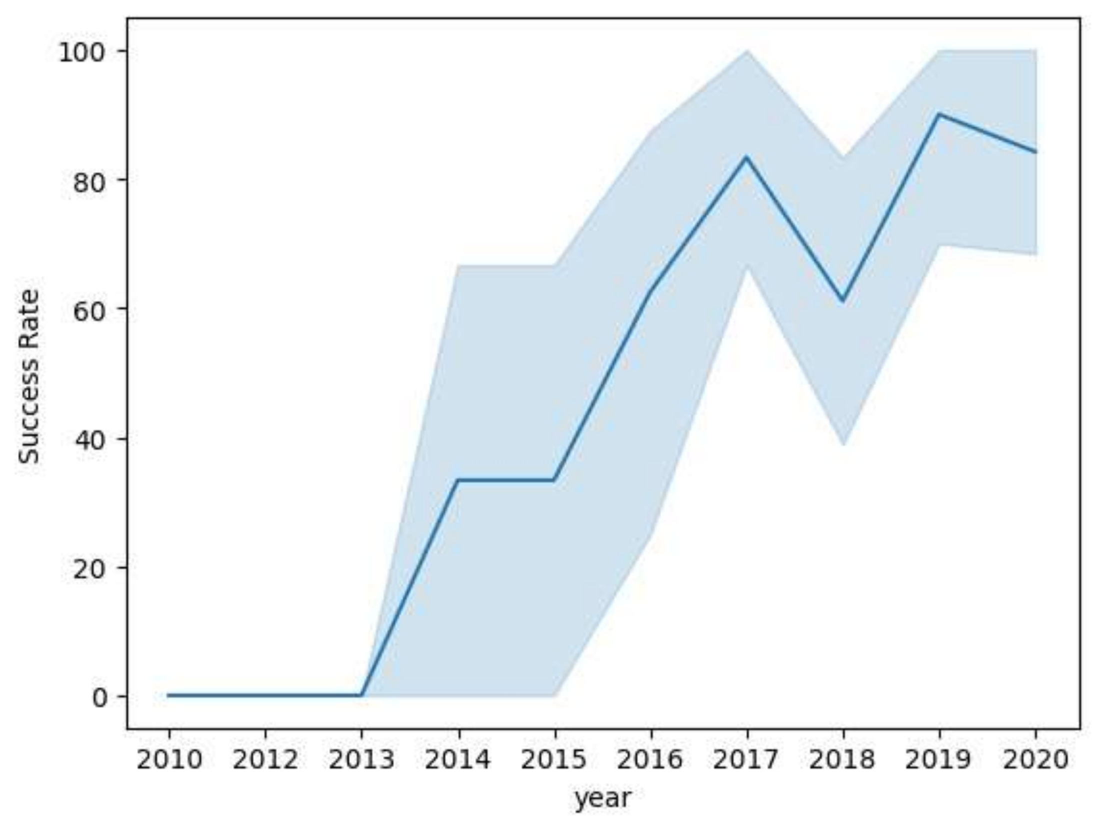
Payload vs. Orbit Type



- Payload mass seems to correlate with orbit
- LEO and SSO seem to have relatively low payload mass
- The other most successful orbit VLEO only has payload mass values in the higher end of the range

Launch Success Yearly Trend

- Success generally increases over time since 2013 with a slight dip in 2018
- Success in recent years at around 80%



All Launch Site Names

- Query unique launch site names from database.
- CCAFS SLC-40 and CCAFSSLC-40 likely all represent the same
- launch site with data entry errors.
- CCAFS LC-40 was the previous name. Likely only 3 unique launch_site

values: CCAFS SLC-40, KSC LC-39A, VAFB SLC-4E

```
[ ]  %sql select DISTINCT LAUNCH_SITE from SPACEXDATASET

 * ibm_db_sa://nxs27972:***@54a2f15b-5c0f-46df-8954-7e38e612c2bd.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud:32733/BLUDB
Done.
  launch_site
 CCAFS LC-40
 CCAFS SLC-40
 KSC LC-39A
 VAFB SLC-4E
```

Launch Site Names Begin with 'CCA'

- First five entries in database with Launch Site name beginning with CCA.

```
[ ] %sql select * from SPACEXDATASET where launch_site like 'CCA%' limit 5
```

```
 * ibm_db_sa://nxs27972:***@54a2f15b-5c0f-46df-8954-7e38e612c2bd.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud:32733/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

- This query sums the total payload mass in kg where NASA was the customer.
- CRS stands for Commercial Resupply Services which indicates that these payloads were sent to the International Space Station (ISS).

```
%sql select sum(payload_mass__kg_) as sum from SPACEXDATASET where customer like 'NASA (CRS)'
```

```
 * ibm_db_sa://nxs27972:***@54a2f15b-5c0f-46df-8954-7e38e612c2bd.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud:32733/BLUDB
Done.
```

SUM
45596

Average Payload Mass by F9 v1.1

- This query calculates the average payload mass or launches which used booster version F9 v1.1
- Average payload mass of F9 1.1 is on the low end of our payload mass range

```
[ ] %sql select avg(payload_mass__kg_) as Average from SPACEXDATASET where booster_version like 'F9 v1.1%'

 * ibm_db_sa://nxs27972:***@54a2f15b-5c0f-46df-8954-7e38e612c2bd.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud:32733/BLUDB
Done.
average
2534
```

First Successful Ground Landing Date

- This query returns the first successful ground pad landing date.
- First ground pad landing wasn’t
- until the end of 2015.
- Successful landings in general
- appear starting 2010.

```
[ ]  %sql select min(date) as Date from SPACEXDATASET where mission_outcome like 'Success'

 * ibm_db_sa://nxs27972:***@54a2f15b-5c0f-46df-8954-7e38e612c2bd.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud:32733/BLUDB
Done.
```

DATE
2010-06-04

Successful Drone Ship Landing with Payload between 4000 and 6000

- This query returns the four booster versions that had successful drone ship landings and a payload mass between 4000 and 6000 noninclusively

```
[ ] %sql select booster_version from SPACEXDATASET where (mission_outcome like 'Success')
    AND (payload_mass__kg_ BETWEEN 4000 AND 6000) AND (landing__outcome like 'Success (drone ship)')

 * ibm_db_sa://nxs27972:***@54a2f15b-5c0f-46df-8954-7e38e612c2bd.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud:32733/BLUDB
Done.
```

booster_version
F9 FT B1022
F9 FT B1026
F9 FT B1021.2
F9 FT B1031.2

Total Number of Successful and Failure Mission Outcomes

- This query returns a count of each
- mission outcome.
- SpaceX appears to achieve its mission outcome nearly 99% of the time.
- This means that most of the landing
- failures are intended.
- Interestingly, one launch has an unclear payload status and unfortunately one failed in flight.

```
[ ] %sql SELECT mission_outcome, count(*) as Count FROM SPACEXDATASET GROUP by mission_outcome ORDER BY mission_outcome

 * ibm_db_sa://nxs27972:***@54a2f15b-5c0f-46df-8954-7e38e612c2bd.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud:32733/BLUDB
Done.
```

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

Boosters Carried Maximum Payload

- This query returns the booster versions that carried the highest payload mass of 15600 kg.
- These booster versions are very similar and all are of the F9 B5 B10xx.x variety.
- This likely indicates payload mass correlates with the booster version that is used.

```
maxm = %sql select max(payload_mass__kg_) from SPACEXDATASET
maxv = maxm[0][0]
%sql select booster_version from SPACEXDATASET where
payload_mass__kg_=(select max(payload_mass__kg_) from SPACEXDATASET)
```

```
 * ibm_db_sa://nxs27972:***@54a2f15b-5c0f-46df-8954-7e38e612c2bd.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud:32733/BLUDB
Done.
 * ibm_db_sa://nxs27972:***@54a2f15b-5c0f-46df-8954-7e38e612c2bd.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud:32733/BLUDB
Done.
```

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

- This query returns the Month, Landing Outcome, Booster Version, Payload Mass (kg), and Launch site of 2015 launches where stage 1 failed to land on a drone ship.
- There were two such occurrences.

```
[ ] %sql select MONTHNAME(DATE) as Month, landing__outcome, booster_version, launch_site
    from SPACEXDATASET where DATE like '2015%' AND landing__outcome like 'Failure (drone ship)'

 * ibm_db_sa://nxs27972:***@54a2f15b-5c0f-46df-8954-7e38e612c2bd.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud:32733/BLUDB
Done.
```

MONTH	landing__outcome	booster_version	launch_site
January	Failure (drone ship)	F9 v1.1 B1012	CCAFS LC-40
April	Failure (drone ship)	F9 v1.1 B1015	CCAFS LC-40

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

- This query returns a list of successful landings and between 2010-06-04 and 2017-03-20 inclusively.
- There are two types of successful landing outcomes: drone ship and ground pad landings.
- There were 8 successful landings in total during this time period

```
[ ] %sql select landing__outcome, count(*) as count from SPACEXDATASET
    where Date >= '2010-06-04' AND Date <= '2017-03-20'
    GROUP by landing__outcome ORDER BY count Desc
```

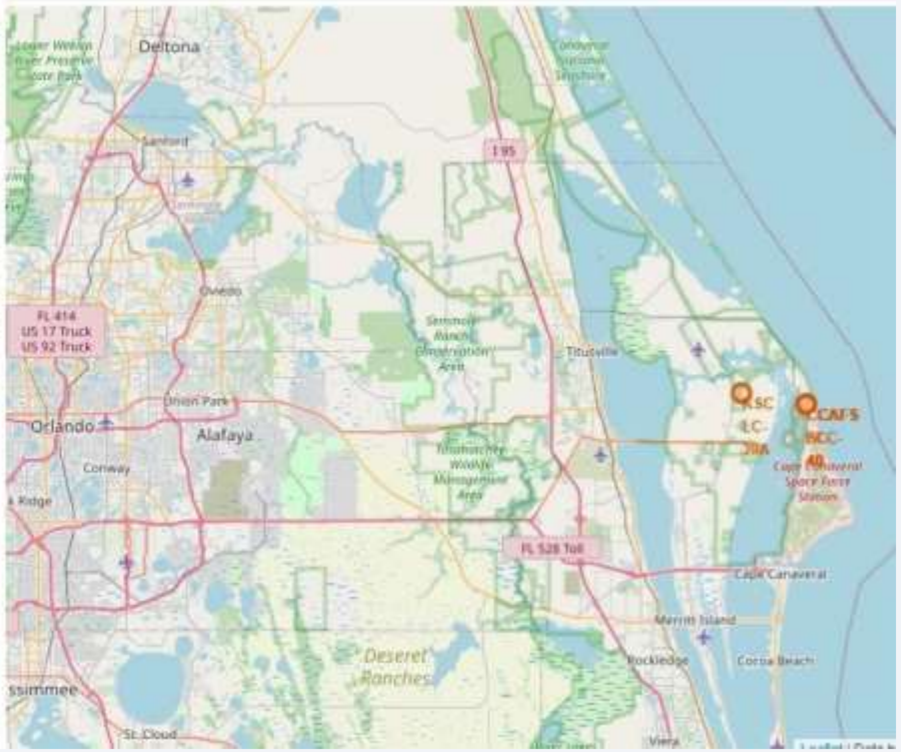
```
 * ibm_db_sa://nxs27972:***@54a2f15b-5c0f-46df-8954-7e38e612c2bd.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud:32733/BLUDB
Done.
```

landing__outcome	COUNT
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

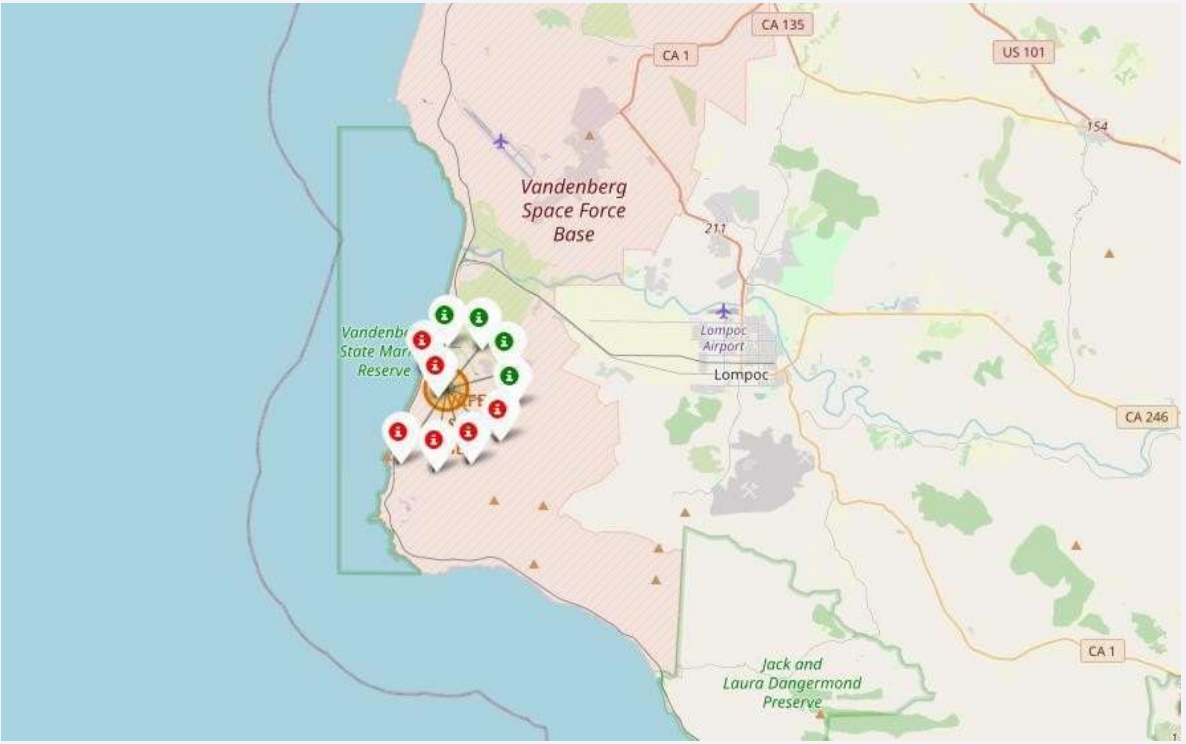
Launch Site Locations



- The left map shows all launch sites relative US map. The right map shows the two Florida launch sites since they are very close to each other. All launch sites are near the ocean.

Color-Coded Launch Markers

- Clusters on Folium map can be clicked on to display each successful landing (green icon) and failed
- landing (red icon). In this example VAFB SLC-4E shows 4 successful landings and 6 failed landings.

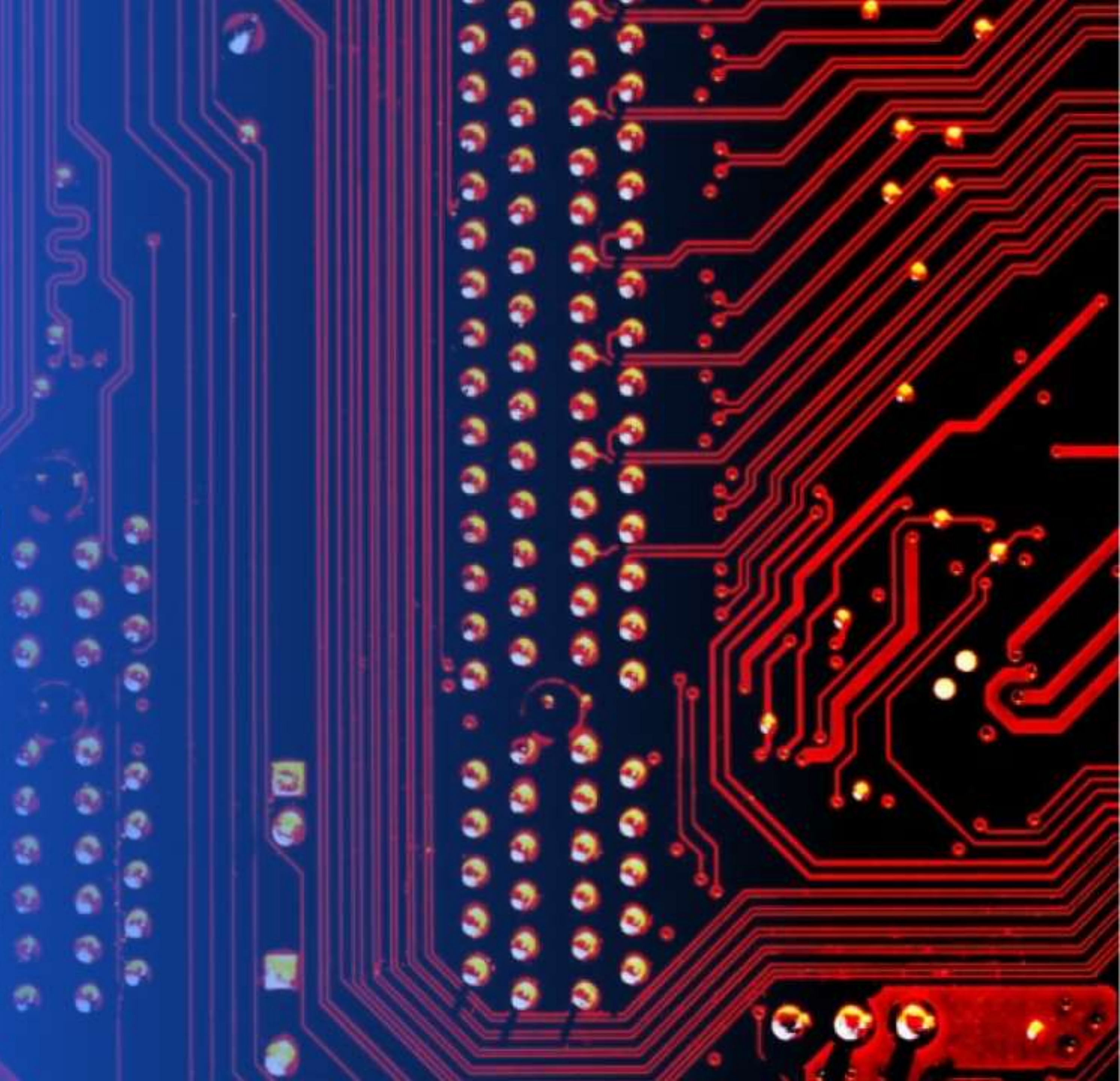


Key Location Proximities

- Using KSC LC-39A as an example, launch sites are very close to railways for large part and supply transportation. Launch sites are close to highways for human and supply transport. Launch sites are also close to coasts and relatively far from cities so that launch failures can land in the sea to avoid rockets falling on densely populated areas.

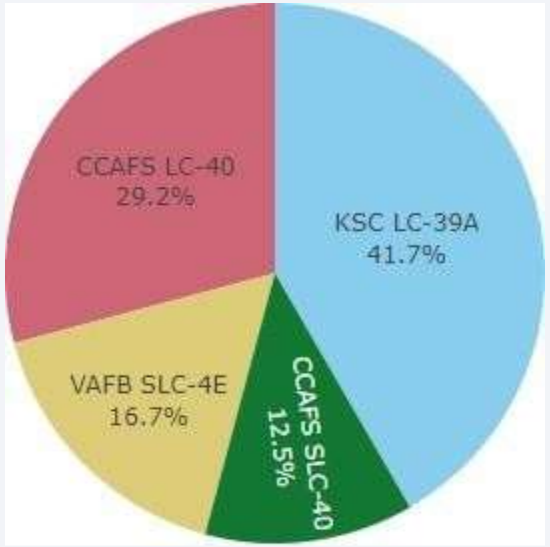
Section 4

Build a Dashboard with Plotly Dash



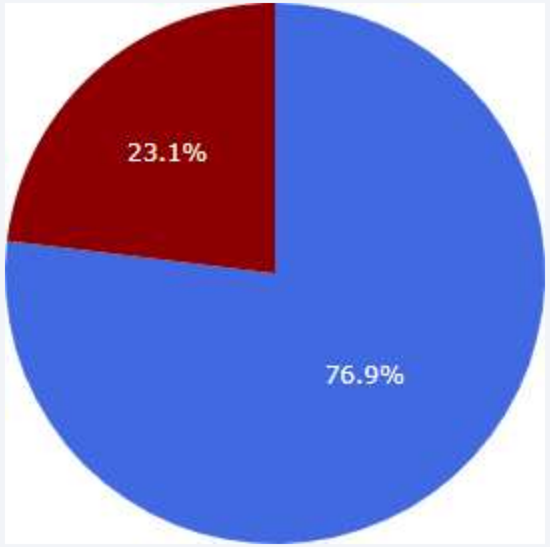
Successful Launches Across Launch Sites

- This is the distribution of successful landings across all launch sites. CCAFS LC-40 is the old name of CCAFS SLC-40 so CCAFS and KSC have the same amount of successful landings, but a majority of the successful landings where performed before the name change. VAFB has the smallest share of successful landings. This may be due to smaller sample and increase in difficulty of launching in the west coast.



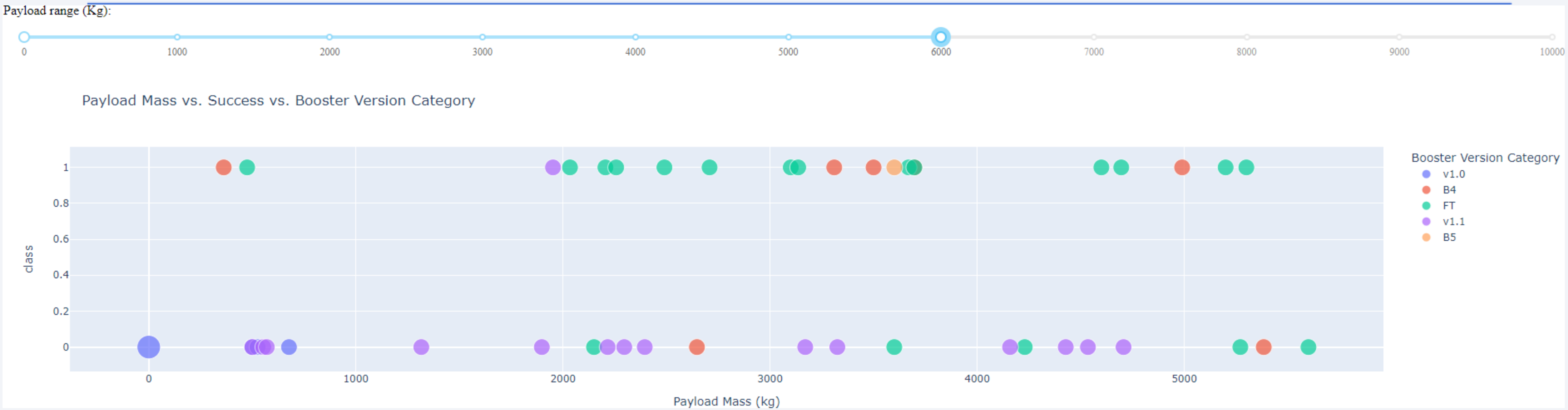
Highest Success Rate Launch Site

- KSC LC-39A has the highest success rate with 10 successful landings and 3 failed landings.



KSC LC-39A Success Rate (blue=success)

Payload Mass vs. Success vs. Booster Version Category

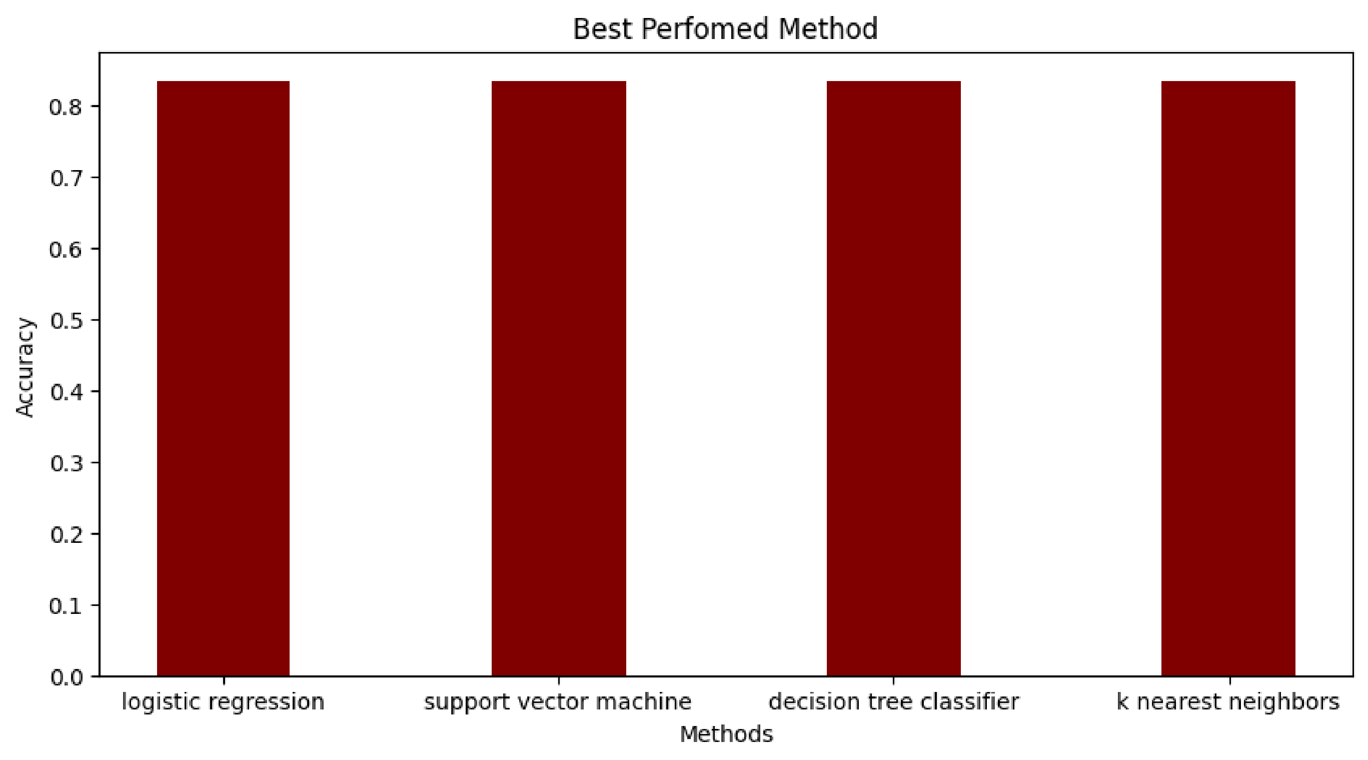


- Plotly dashboard has a Payload range selector. However, this is set from 0-10000 instead of the max Payload of 15600. Class indicates 1 for successful landing and 0 for failure. Scatter plot also accounts for booster version category in color and number of launches in point size. In this particular range of 0-6000, interestingly there are two failed landings with payloads of zero kg.

Section 5

Predictive Analysis (Classification)

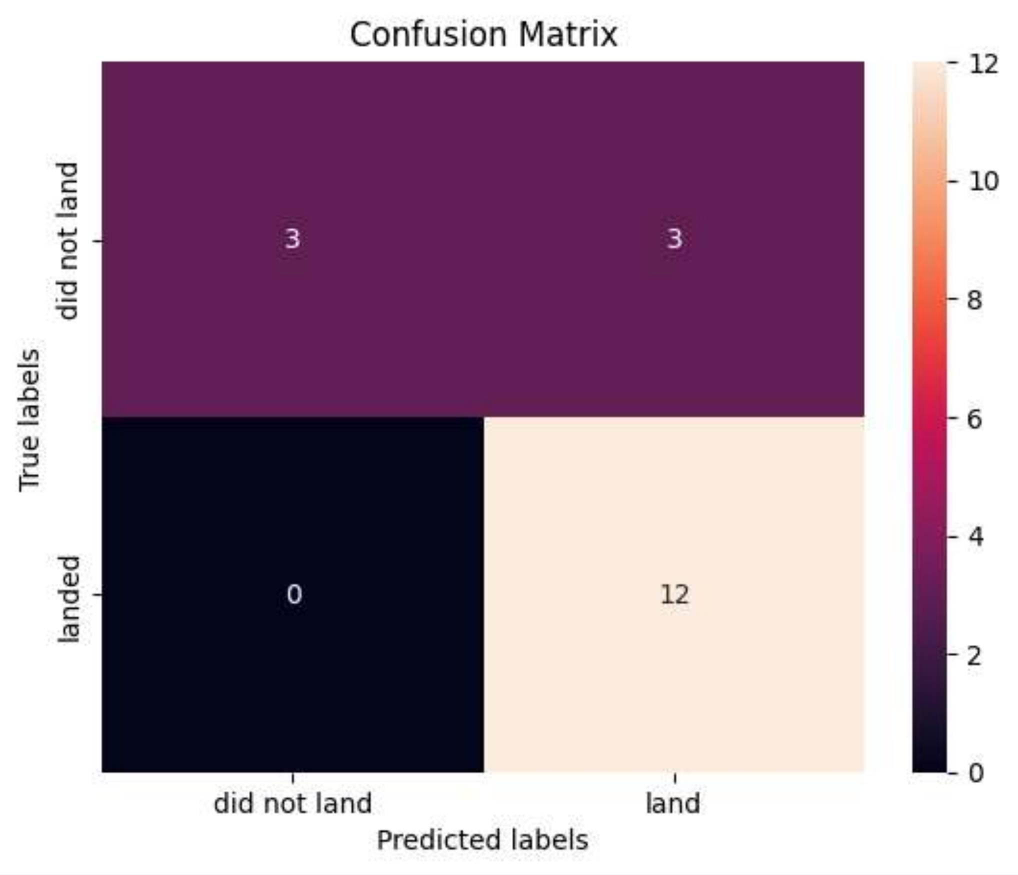
Classification Accuracy



- All models had virtually the same accuracy on the test set at 83.33% accuracy. It should be noted that test size is small at only sample size of 18.
- This can cause large variance in accuracy results, such as those in Decision Tree Classifier model in repeated runs.
- We likely need more data to determine the best model.

Confusion Matrix

- Since all models performed the same for the test set, the confusion matrix is the same across all models. The models predicted 12 successful landings when the true label was successful landing.
- The models predicted 3 unsuccessful landings when the true label was unsuccessful landing.
- The models predicted 3 successful landings when the true label was unsuccessful landings (false positives). Our models over predict successful landings.



Conclusions

- Our objective was to build a machine learning tool for SpaceY to compete with SpaceX. The aim was to forecast when the first stage of a rocket will land successfully, potentially saving around $100 million USD per launch.
- To achieve this, we gathered data from a SpaceX API and scraped information from the SpaceX Wikipedia page. This data was carefully labeled and stored in a DB2 SQL database. We also developed a user-friendly dashboard for visualization purposes.
- After rigorous development, our machine learning model reached an impressive accuracy of 83%. This means that Allon Mask of SpaceY can rely on this model to predict, with considerable confidence, whether a launch will witness a successful first stage landing before proceeding. Such insights are invaluable for making informed decisions about whether to proceed with a launch or not.
- Moving forward, it's beneficial to continue gathering more data to refine our model further and enhance its accuracy. This ongoing data collection will enable us to fine-tune our machine learning algorithms and ensure they deliver the best possible predictions for SpaceY.

Appendix

- Main Repository Link:

  https://github.com/Ksshitij-Singare/IBM_DataScience_Professional_Proj.

Thank you!