IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Jian Liu

June 19, 2024

Outline

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

Executive Summary

- Data for SpaceX first stage rocket landing were collected from SpaceX API as well as Wikipedia, combined and wrangled. Exploratory data analysis and visualization were performed using SQL and plots. Additional visualization were performed using map views, and interactive Dashboard. Classification models were developed to predict launch outcomes based on selected features
- The steady increase in successful launch rate close to 80% demonstrate that the novel idea of reuse first stage rocket is **feasible**. Key features including booster version and payload mass and others were used to develop a model to **predict** the landing outcome with accuracy as high as 94%

Introduction

- Sending rockets to space is prohibitively expensive
- SpaceX sets an unprecedent example to achieve this goal as the first privately owned company
- One conceptual change for SpaceX is the idea to reuse the first stage rocket to have brought down the cost tremendously
- It is of great interest and importance to analyze data from testing launches to obtain insights into answering two questions
  - Is it feasible to reuse the first-stage rocket?
  - Can models be built with the existing data to predict which factors contribute to the success of reusing first-stage rocket?

Section 1

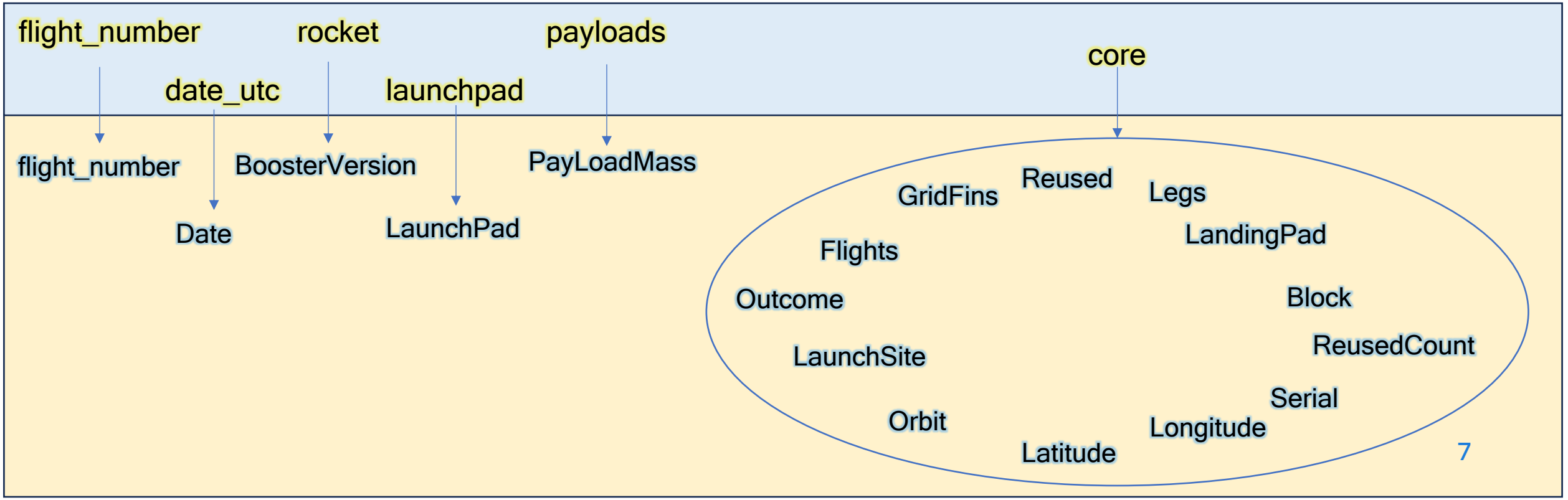
Methodology

Methodology (Executive Summary)

- Data Collection Methodology
  - Launch data were collected from both SpaceX API and Wikipedia
- Perform data wrangling
  - Missing data were replaced when appropriate
  - Rocket landing outcomes were categorized two classes: success & failure
  - Features were selected and values converted for further analysis and modeling
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - Data were separated into training and testing
  - Four classification models were tested for predicting the landing outcome
  - Optimized parameters were used for each model and accuracies for predicting the testing data was calculated and compared

Data Collection from SpaceX API

- Access SpaceX API at: https://github.com/r-spacex/SpaceX-API
- The URL for requesting the data should be something similar to: https://api.spacexdata.com/v4/launches/past (a json file)
- From 6 columns, a total of 17 features were extracted into a Dataframe (data)



Data Collection from SpaceX API

- Data was filtered by *BoosterVersion* to contain Falcon 9 type only (data_falcon9)
- An image of data_falcon9 is shown below:

```
data_falcon9.head()
```

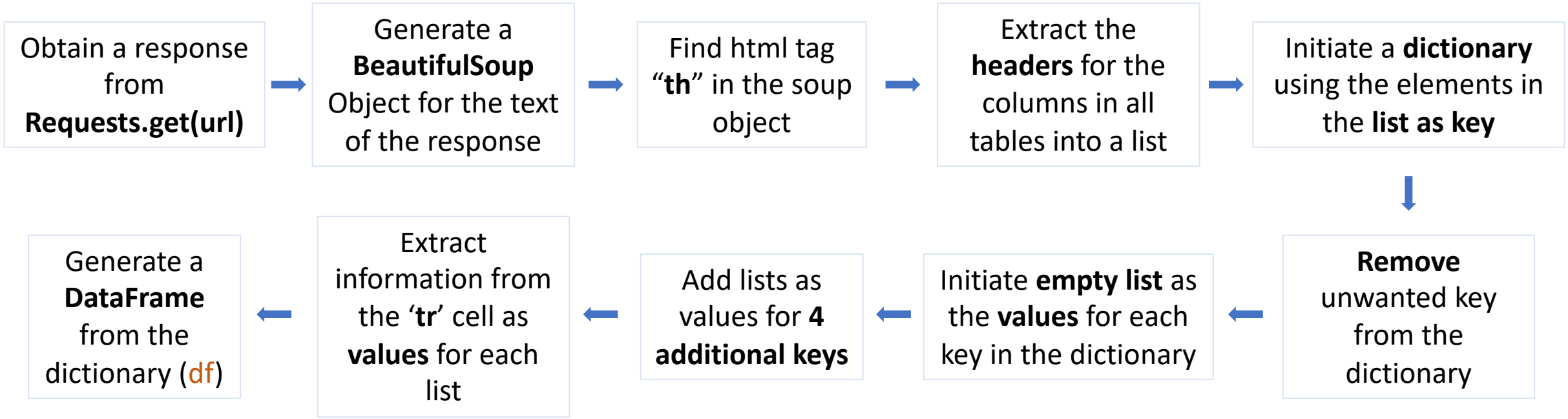
	FlightNumber	Date	BoosterVersion	PayloadMass	Orbit	LaunchSite	Outcome	Flights	GridFins	Reused	Legs	LandingPad	Block	ReusedCount	Serial	Longitude	Latitude
4	1	2010-06-04	Falcon 9	NaN	LEO	CCSFS SLC 40	None None	1	False	False	False	None	1.0	0	B0003	-80.577366	28.561857
5	2	2012-05-22	Falcon 9	525.0	LEO	CCSFS SLC 40	None None	1	False	False	False	None	1.0	0	B0005	-80.577366	28.561857
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	28.561857
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	34.632093
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	28.561857

Git URL_1:

https://github.com/jjliusf/JJ_DataSci_2024/blob/main/Capstone_spacex_data_collection_api.ipynb

Date Collection from Web Scraping

- The URL for web scraping is: https://en.wikipedia.org/w/index.php?title=List_of_Falcon_9_and_Falcon_Heavy_launches&oldid=1027686922
- The step by step flowchart for generating a DataFrame from web scraping is shown below:



Date Collection from Web Scraping

- An image of the Dataframe df is shown below:

```
df.head()
```

	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	B0003.1	Failure	4 June 2010	18:45
1	2	CCAFS	Dragon	0	LEO		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	F9 v1.0B0006.1	No attempt	8 October 2012	00:35
4	5	CCAFS	SpaceX CRS-2	4,877 kg	LEO	NASA	Success	F9 v1.0B0007.1	No attempt\n	1 March 2013	15:10

Git URL_2:

https://github.com/jjliusf/JJ_DataSci_2024/blob/main/Capstone_spacex_webscraping.ipynb

Data Wrangling

- Missing data were replaced when appropriate
  - Five missing data in PayloadMass were replaced by the mean() of PayloadMass
  - Replacement for missing data in LandingPad was not performed
- Rocket landing outcomes were categorized two classes: success & failure
  - There are a total of 8 descriptions for the landing outcome which were further categorized in to success (1) when the description contained “True” and failure for the rest (0)
- Feature engineering and values converted to float type for further analysis and modeling
  - A total of 12 features were selected for model development
  - Values for the 4 categorical features were converted to numerical values using the One Hot Encoding method
  - One_Hot features were combined with the 8 non_categorical features into the DataFrame
  - Finally, all values in the DataFrame were converted to float datatype necessary for modeling

Data Wrangling: Feature Engineering

- The image of the original DataFrame with 12 columns

```
features.head()
```

	FlightNumber	PayloadMass	Orbit	LaunchSite	Flights	GridFins	Reused	Legs	LandingPad	Block	ReusedCount	Serial
0	1	6104.959412	LEO	CCAFS SLC 40	1	False	False	False	NaN	1.0	0	B0003
1	2	525.000000	LEO	CCAFS SLC 40	1	False	False	False	NaN	1.0	0	B0005
2	3	677.000000	ISS	CCAFS SLC 40	1	False	False	False	NaN	1.0	0	B0007
3	4	500.000000	PO	VAFB SLC 4E	1	False	False	False	NaN	1.0	0	B1003
4	5	3170.000000	GTO	CCAFS SLC 40	1	False	False	False	NaN	1.0	0	B1004

- The image of the DataFrame after using One Hot Encoding and converting to float datatype (notice now there are 80 columns instead of 12)

```
features_one_hot.head()
```

	FlightNumber	PayloadMass	Flights	GridFins	Reused	Legs	Block	ReusedCount	Orbit_ES-L1	Orbit_GEO	...	Serial_B1048	Serial_B1049	Serial_B1050	Serial_B1051	Serial_B1054	Serial_B1056	Serial_B1058	Serial_B1059	Serial_B1060	Serial_B1062
0	1	6104.959412	1	0.0	0.0	0.0	1.0	0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
1	2	525.000000	1	0.0	0.0	0.0	1.0	0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
2	3	677.000000	1	0.0	0.0	0.0	1.0	0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
3	4	500.000000	1	0.0	0.0	0.0	1.0	0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
4	5	3170.000000	1	0.0	0.0	0.0	1.0	0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

5 rows × 80 columns

Git URL_3:

https://github.com/jjliusf/JJ_DataSci_2024/blob/main/Capstone_spacex_Data_wrangling.ipynb

EDA with Data Visualization

- An image of the Dataset used for Data Visualization is shown below

```
df.head(5)
```

	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
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	28.561857	0

- Three sets of visualization plots were generated (results and conclusion shown in the following 3 slides)
- These plots shed insights to whether there is a progress in successful launches with more flights and if so, which Payload, Orbit or Launch Site is the most promising for future launches (this answers Why)

Git URL_4:
https://github.com/jjliusf/JJ_DataSci_2024/blob/main/Capstone_spacex_eda_dataviz.ipynb

EDA with SQL: SQL Queries Performed

```
%sql SELECT DISTINCT(Launch_Site) from SPACEXTABLE; (Task1)
%sql SELECT DISTINCT(Landing_Outcome) from SPACEXTABLE; (optional)
%sql SELECT * from SPACEXTABLE WHERE Launch_Site like 'CCA%' LIMIT 5; (Task2)
%sql SELECT SUM(PAYLOAD_MASS__KG_) from SPACEXTABLE WHERE Customer like 'NASA%(CRS)%';(Task3)
%sql SELECT AVG(PAYLOAD_MASS__KG_) from SPACEXTABLE WHERE Booster_Version like 'F9 v1.1%'; (Task4)
%sql SELECT MIN(Date) from SPACEXTABLE WHERE Landing_Outcome = 'Success (ground pad)';(Task5)
%sql SELECT DISTINCT(Booster_Version) from SPACEXTABLE \
    WHERE Landing_Outcome = 'Success (drone ship)' AND PAYLOAD_MASS__KG_ between 4000 and 6000; (Task6)
%sql SELECT COUNT(Landing_Outcome) as Total_Success from SPACEXTABLE WHERE Landing_Outcome like 'Success%';  (Task7)
%sql SELECT COUNT(Landing_Outcome) as Total_Failure from SPACEXTABLE WHERE Landing_Outcome like 'Failure%';  (Task7)
%sql SELECT DISTINCT(Booster_Version) from SPACEXTABLE \
    WHERE PAYLOAD_MASS__KG_ = (select MAX(PAYLOAD_MASS__KG_) from SPACEXTABLE);(Task8)
%sql SELECT substr(Date,6,2) as Month, Booster_Version, Launch_Site, Landing_Outcome from SPACEXTABLE \
    WHERE substr(Date,0,5)='2015' and Landing_Outcome = 'Failure (drone ship)'; (Task9)
%sql SELECT Landing_Outcome, COUNT(Landing_Outcome) from SPACEXTABLE \
    WHERE (Landing_Outcome = 'Failure (drone ship)' OR Landing_Outcome = 'Success (ground pad)') \
    AND (Date between '2010-06-04' and '2017-03-20') \
    GROUP BY Landing_Outcome ORDER BY COUNT(Landing_Outcome) DESC; (Task10)
```

Git URL_5:
https://github.com/jjliusf/JJ_DataSci_2024/blob/main/Capstone_pacex_eda_sqllite.ipynb

Build an Interactive Map with Folium

- To visualize the launch sites on the map
  - Map objects of **circles** and **markers** were created and added to the Folium map for each of the four launch sites
- To visualize the outcome of launches as each launch site
  - A Map object of **MarkerCluster** consisting of Map Object of **markers f**or each launch colored by its outcome for each launch site were added to the Folium map
- To assist visualization of the launch site and its association with the locations of coastlines, railways, highways and cities
  - The distances from the launch site to 4 other types of locations were calculated for each launch site
  - Map objects of **markers** for labeling the distances and **lines** resulted from applying PolyLine function to the distance were added to the Folium map

Git URL_6:
https://github.com/jjliusf/JJ_DataSci_2024/blob/main/Capstone_spacex_LaunchSite_Folium.ipynb

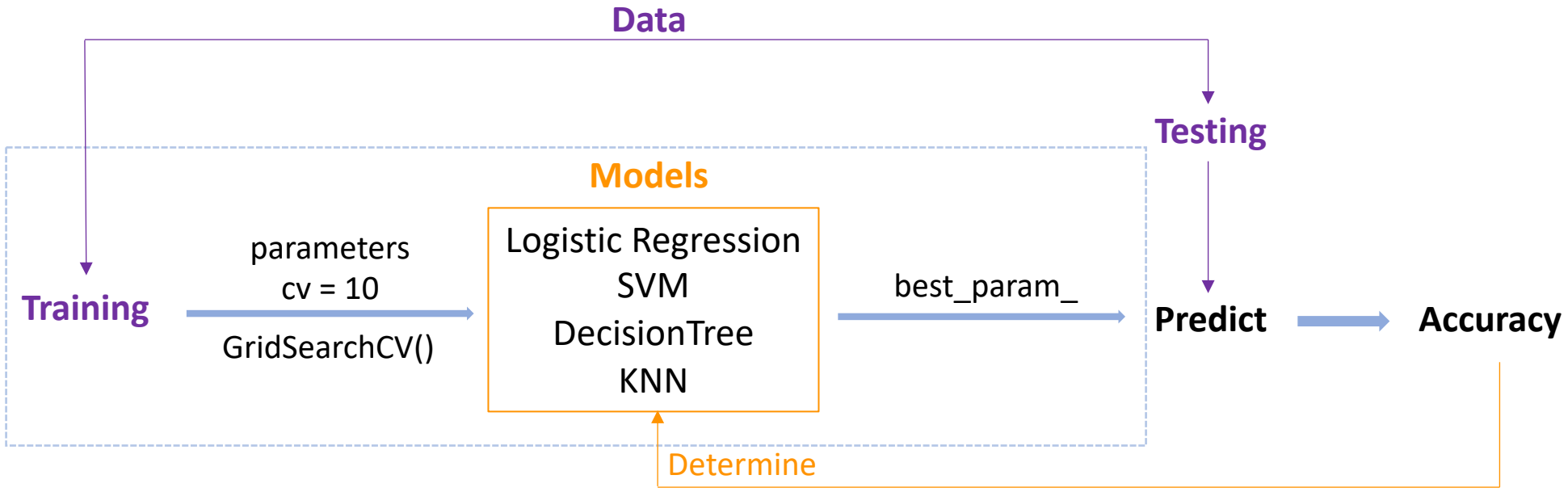
Build a Dashboard with Plotly Dash

- Two types of interactions were added to the interactive Dashboards, each with a type of plot to provide the results
- The first interaction is a dropdown list with the options of selecting each of the 4 launch site or all four sites
  - Pie charts were used for the result of this interaction because the success rate is clearly and immediately revealed by the size the pie
  - All sites option is also included to show the relative success rate for each of the four sites
- The second interaction is a sliding bar in which one can control the low and high end of the bar scale as the range of Payload for each site or all four sites
  - Scatter plots were used for the result of this interaction with each dot colored by the type of Booster Version because one can directly visualize each launch with the Booster version changed with Payload range for each launch site
  - Again all sites option is included to show the information collectively

Git URL_7: https://github.com/jjliusf/JJ_DataSci_2024/blob/main/Csapstone_sapcex_Dashboard.py

Predictive Analysis (Classification)

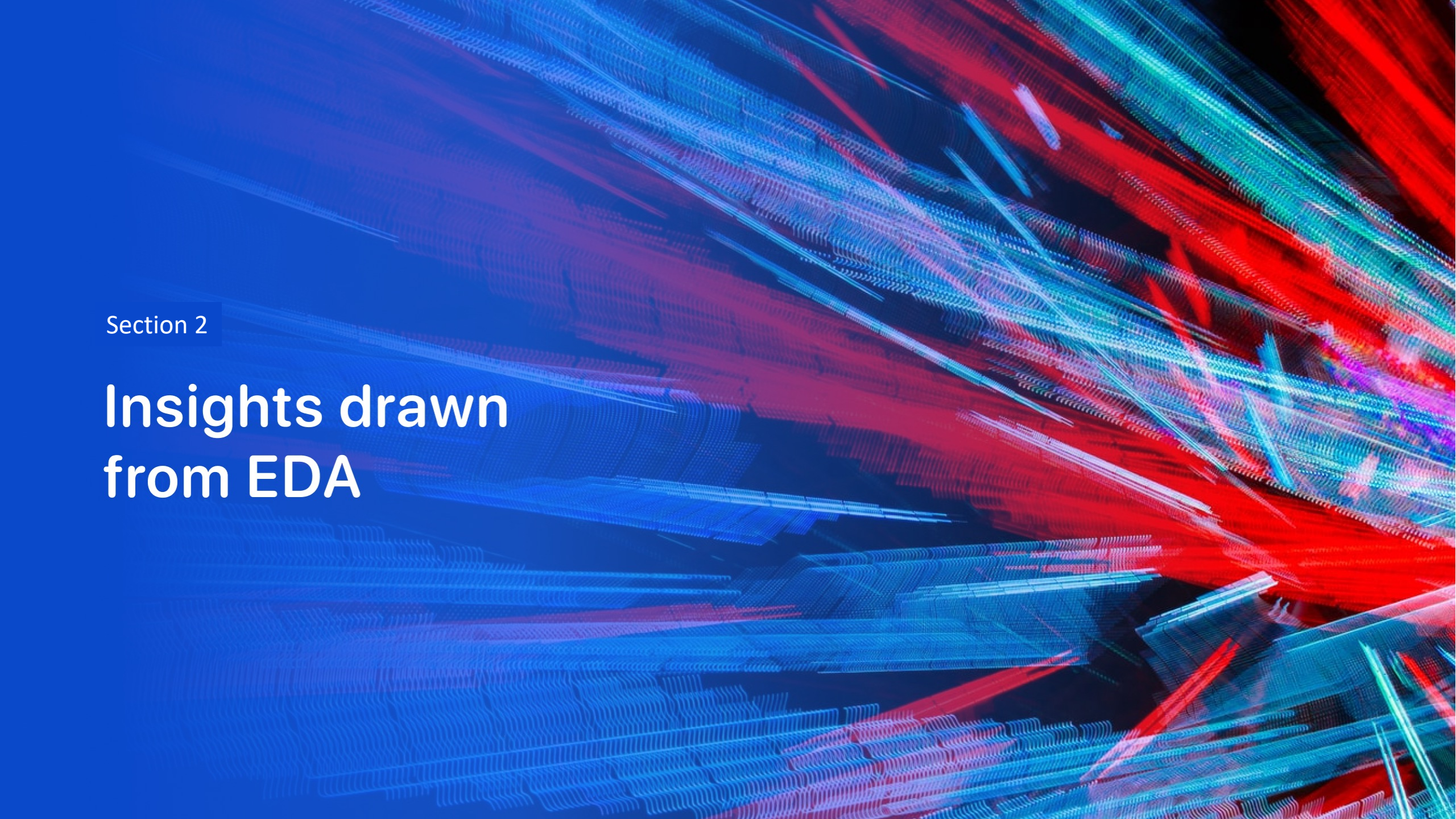
- To predict whether a launch outcome is success or not, classification models were used to develop models
- Four classification models including Logistic Regression, Supporting Vector Machine (SVM), Decision Tree and k nearest neighbor (KNN) were selected to determine which model results in the most accurate prediction of the launch outcome
- Data were separated into training and testing data
- Cross validation value of 10 was used for all models
- Different parameters were tested to determine the optimized parameters for each model
- The accuracies for each model were compared and the best model was determined



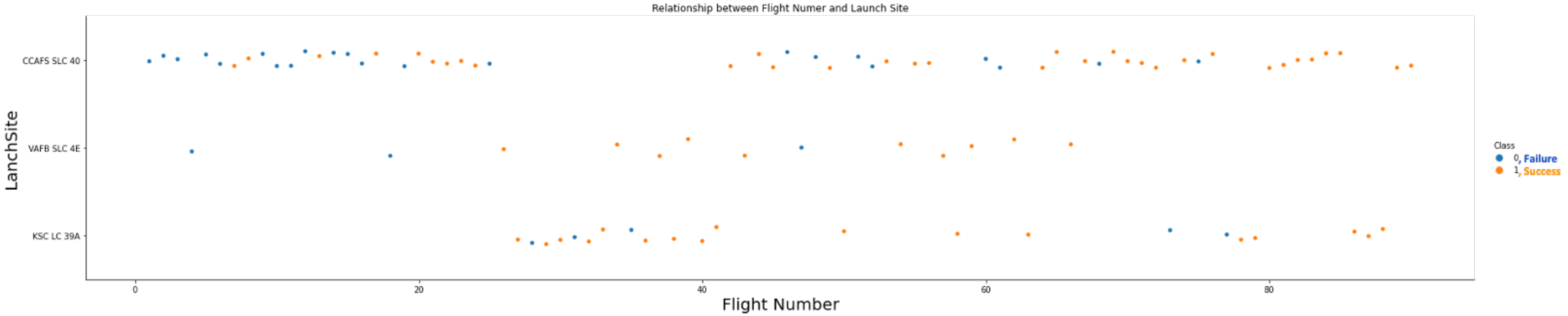
Git URL_8: https://github.com/jjliusf/JJ_DataSci_2024/blob/main/Capstone_spacex_ML_prediction.ipynb

Results

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

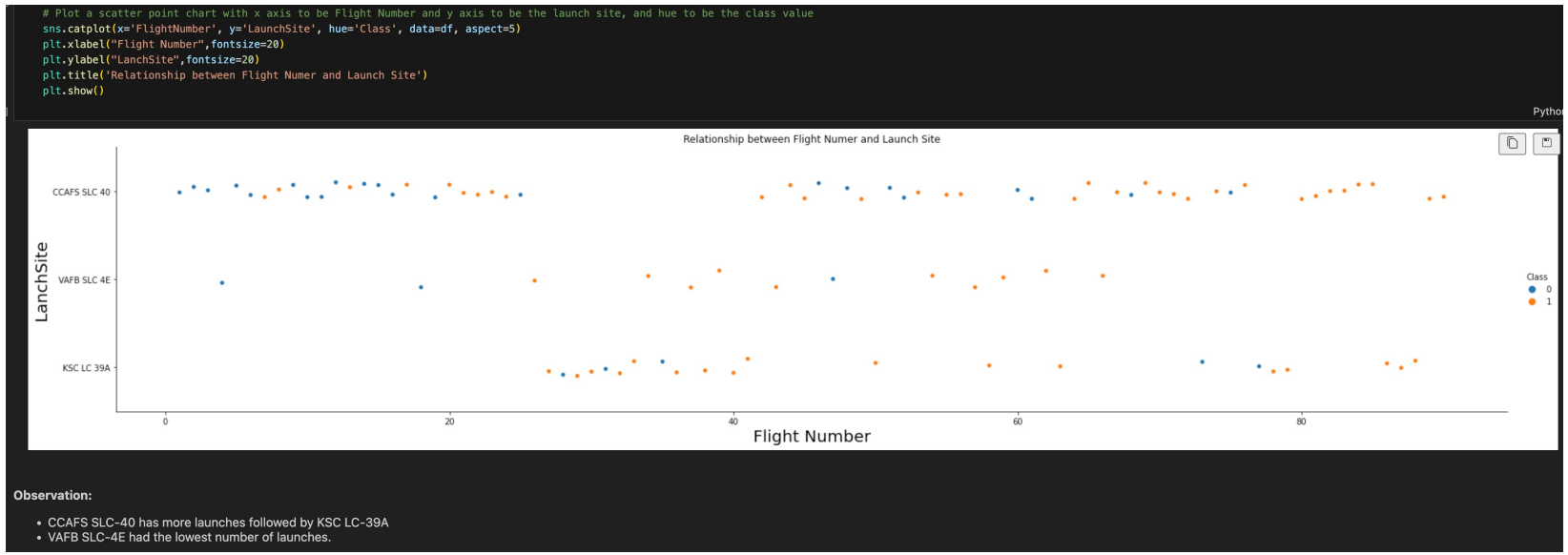

Section 2
Insights drawn from EDA

Flight Number vs Launch Site

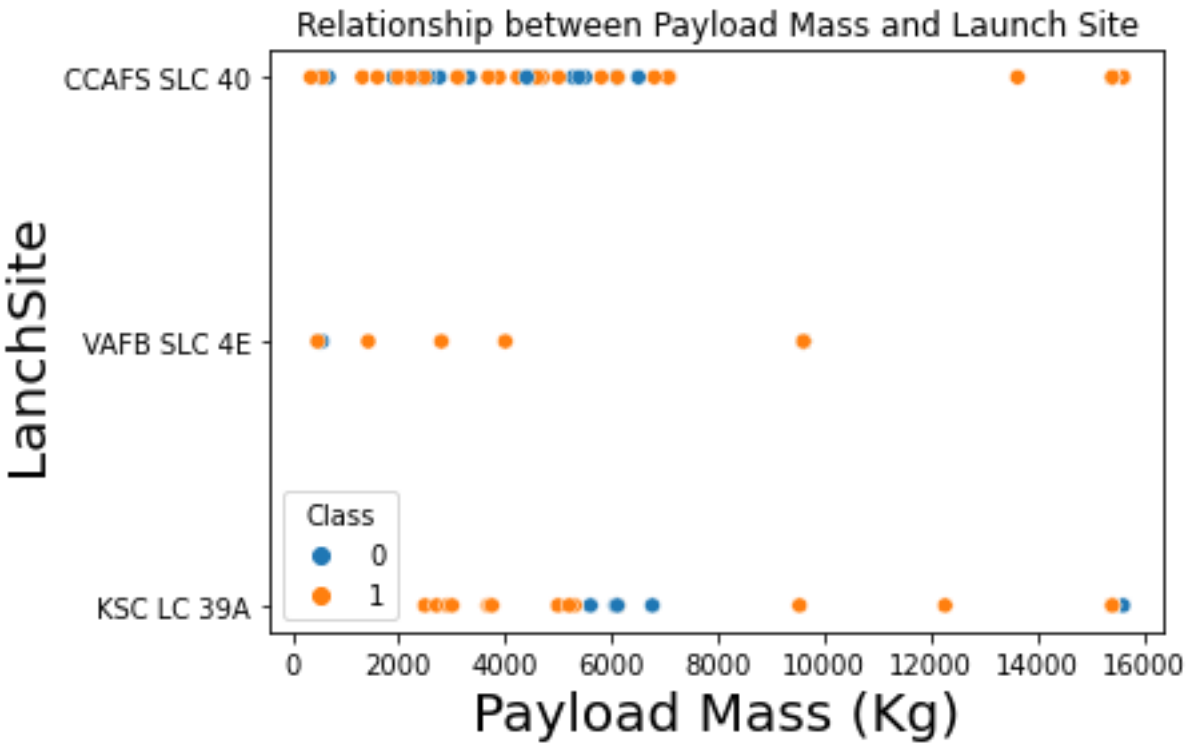


- CCAFS SLC-40 has more launches followed by KSC LC-39A
- VAFB SLC-4E had the lowest number of launches.

Screenshot →



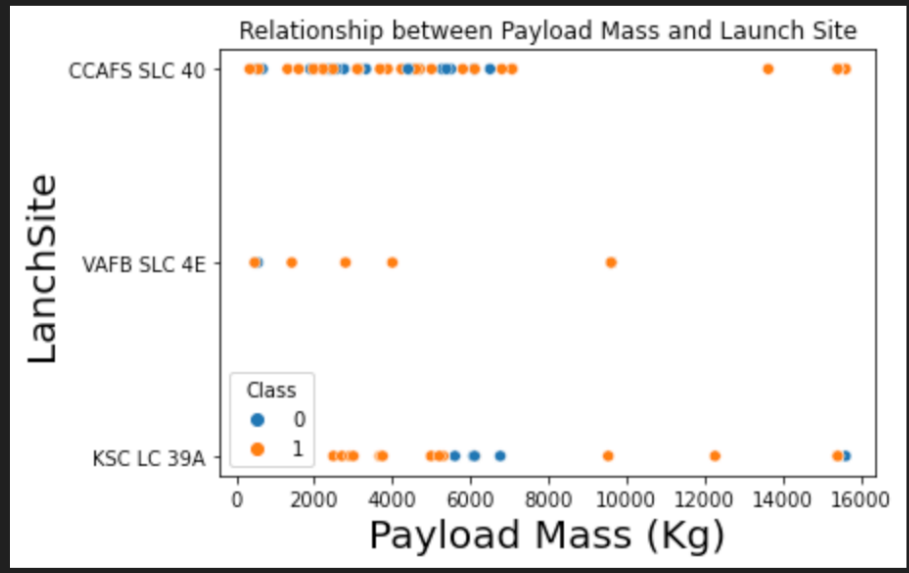
Payload vs Launch Site



- The Payload at VAFB SLC-4E never exceeded 10K Kg
- A few higher Payloads were launched at both CCAFS SLC-40 and KSC LC-39A sites

Screenshot

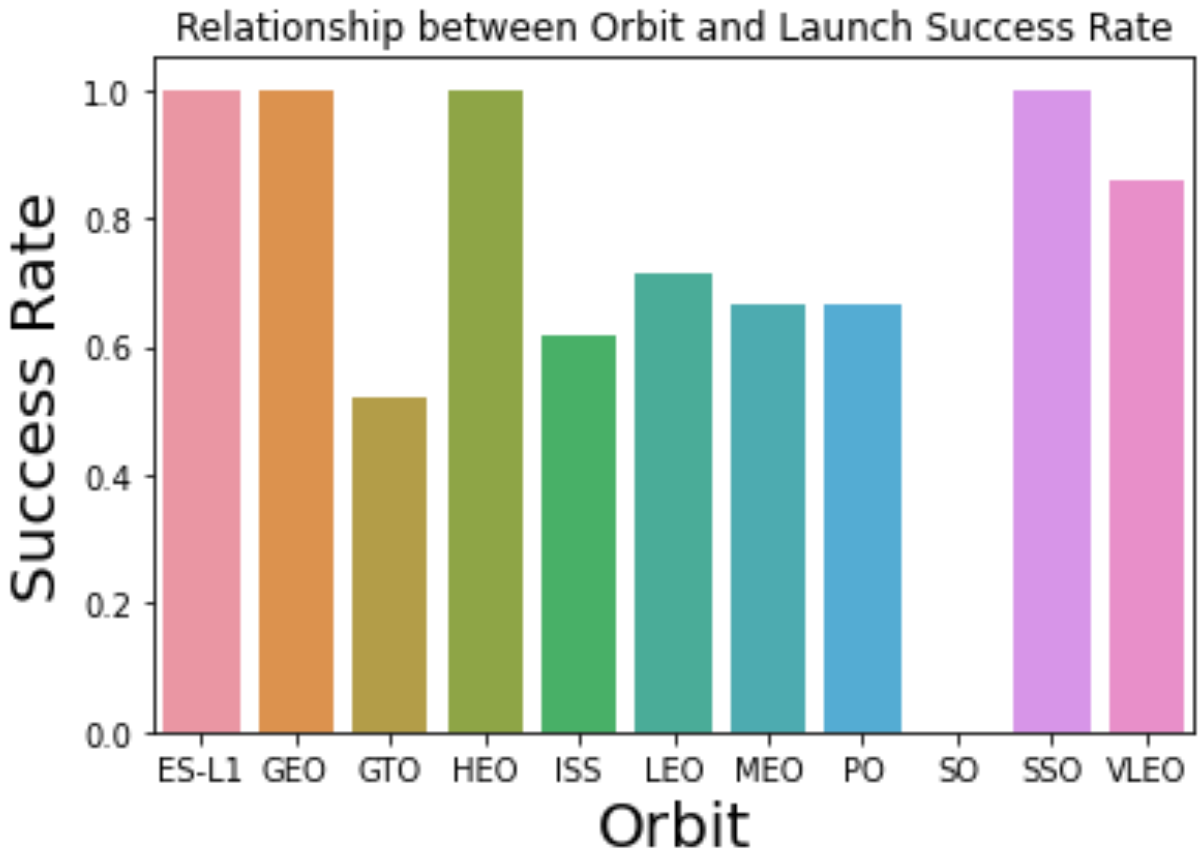
```
sns.scatterplot(x='PayloadMass', y='LaunchSite', hue='Class', data=df)
plt.xlabel("Payload Mass (Kg)",fontsize=20)
plt.ylabel("LanchSite",fontsize=20)
plt.title('Relationship between Payload Mass and Launch Site')
plt.show()
```



Observations

- The Payload at VAFB SLC-4E never exceeded 10K Kg
- A few higher Payloads were launched at both CCAFS SLC-40 and KSC LC-39A sites

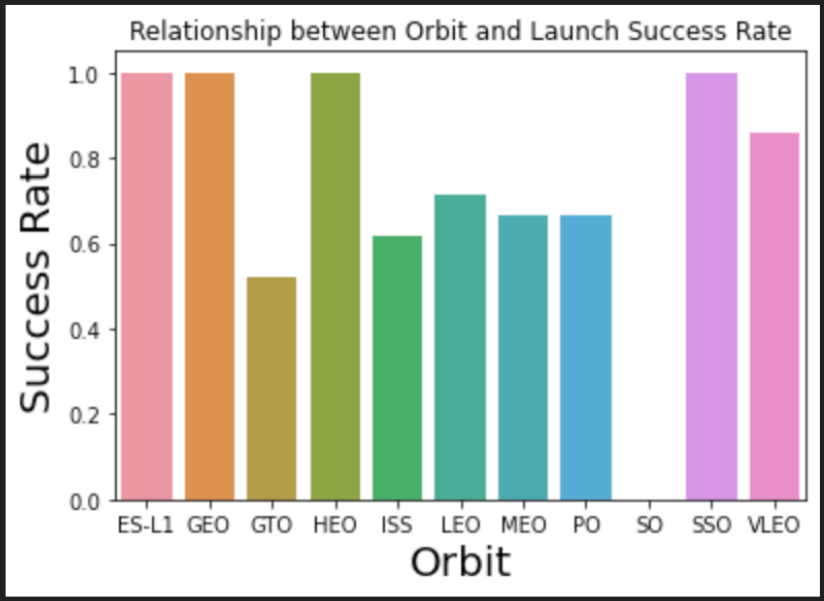
Success Rate vs Orbit Type



- ES-L1, GEO , HEO and SSO orbits have 100% successful launch rate

Screenshot

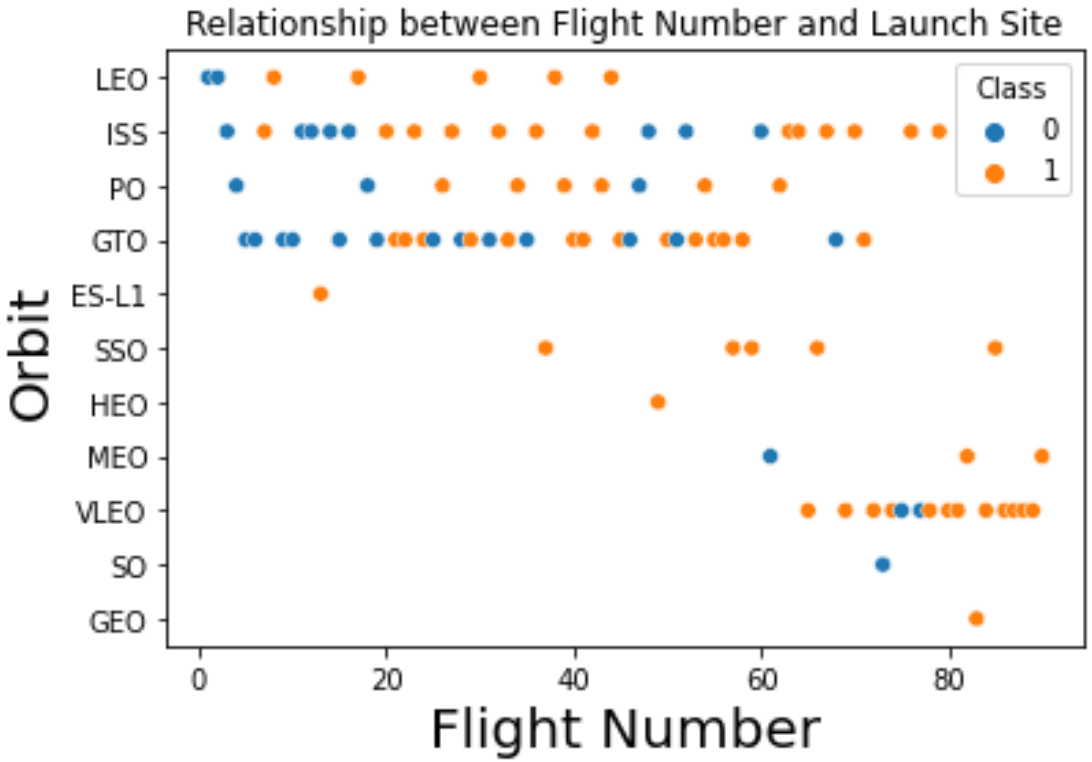
```
sns.barplot(x=df_orbit.index, y=df_orbit.values)
plt.xlabel("Orbit",fontsize=20)
plt.ylabel("Success Rate",fontsize=20)
plt.title('Relationship between Orbit and Launch Success Rate')
plt.show()
```



Observations

- ES-L1, GEO , HEO and SSO orbits have 100% successful launch rate

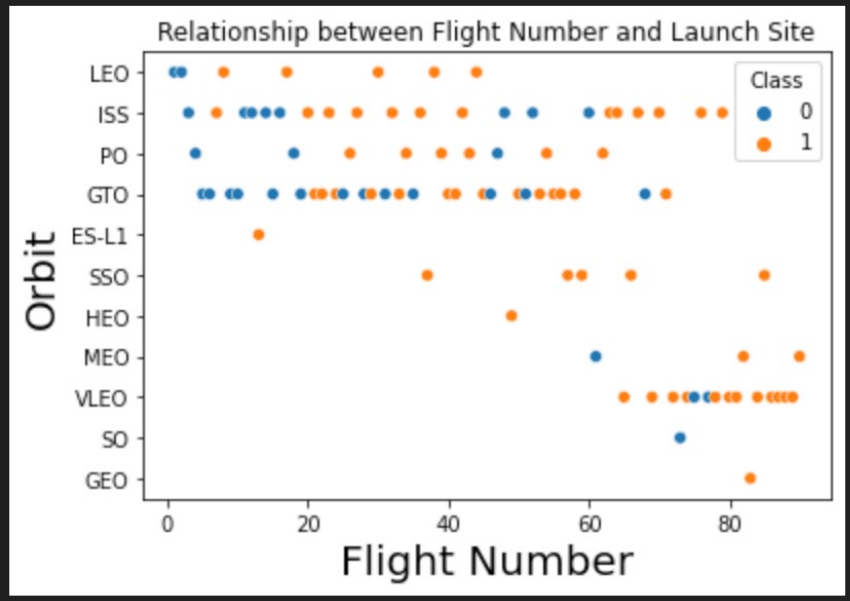
Flight Number vs Orbit Type



- Earlier Fights were to a limited number of orbit types (i.e. LEO, ISS, PO & GTO)
- Later flights went to more different types of orbits

Screenshot

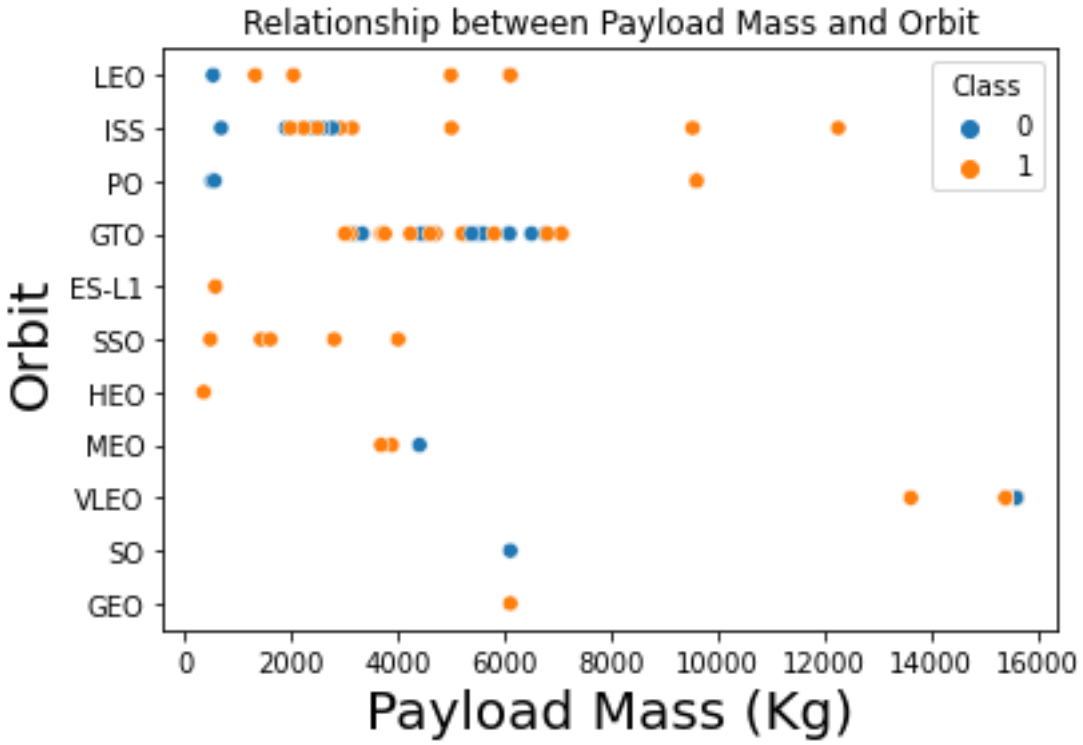
```
sns.scatterplot(x='FlightNumber', y='Orbit', hue='Class', data=df)
plt.xlabel("Flight Number",fontsize=20)
plt.ylabel("Orbit",fontsize=20)
plt.title('Relationship between Flight Number and Orbit')
plt.show()
```



Observations

- Earlier Fights were to a limited number of orbit types (i.e. LEO, ISS, PO & GTO)
- Later flights went to more different types of orbits

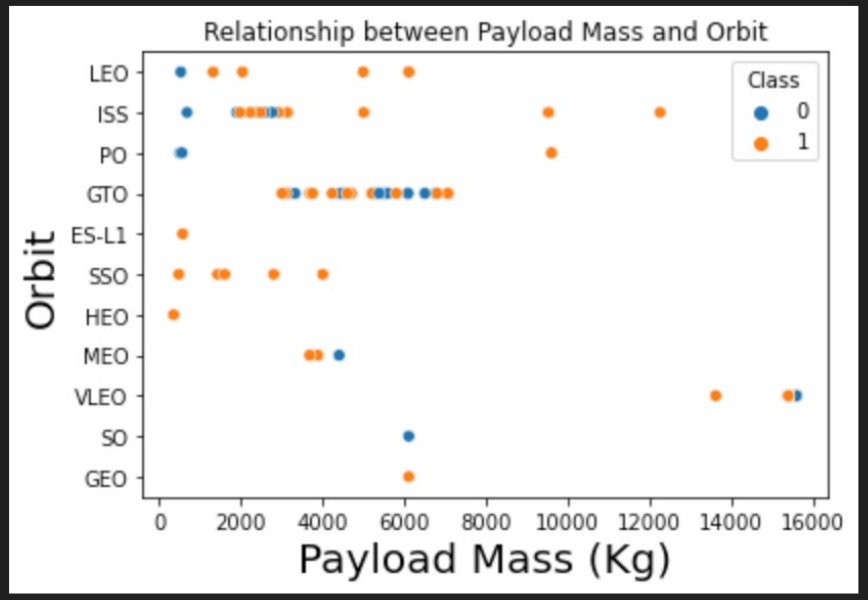
Payload vs Orbit Type



- Limited numbers of Payloads higher than 10K were launched to only ISS, PO and VLEO
- Payloads not higher than 8K went to more orbit types
- Success rates are not consistent with all orbit types

Screenshot

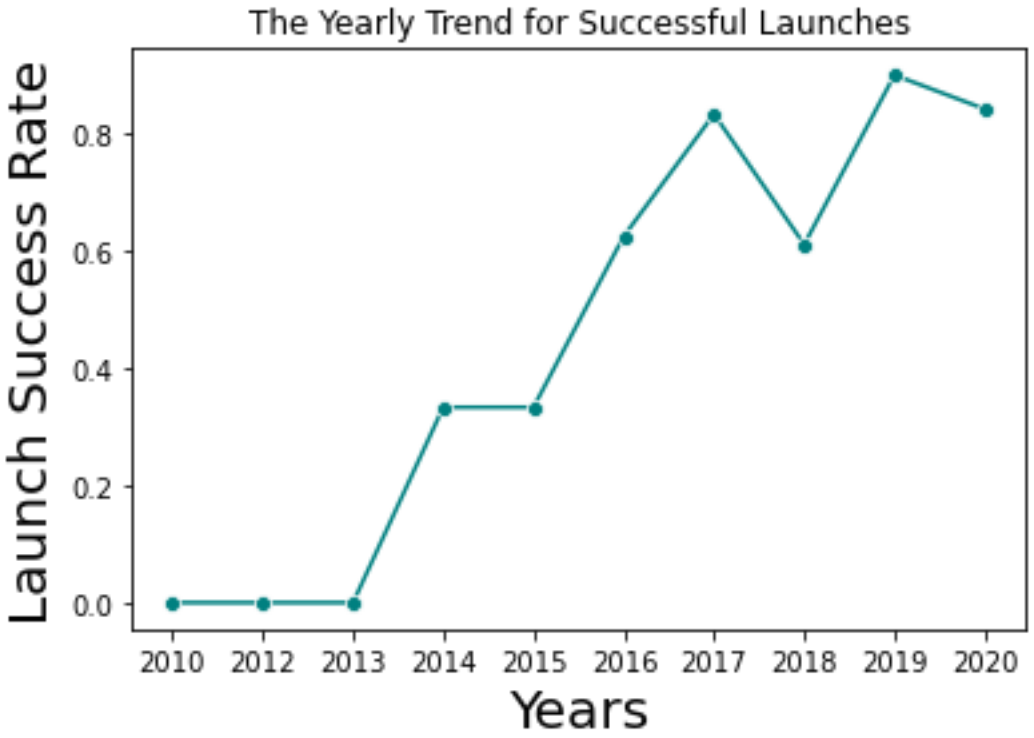
```
sns.scatterplot(x='PayloadMass', y='Orbit', hue='Class', data=df)
plt.xlabel("Payload Mass (Kg)",fontsize=20)
plt.ylabel("Orbit",fontsize=20)
plt.title('Relationship between Payload Mass and Orbit')
plt.show()
```



Observations

- Limited numbers of Payloads higher than 10K were launched to only ISS, PO and VLEO
- Payloads not higher than 8K went to more orbit types
- Success rates are not consistent with all orbit types

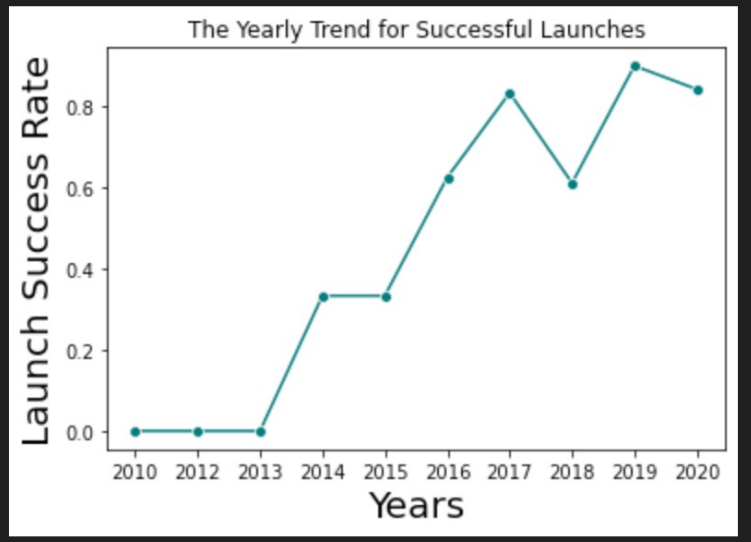
Launch Success Yearly Trend



- Launch success rate remain low in early years (2010-2013) but started to and continuously increase later on
- The trend fluctuated between 2018 and 2020 but is definitely positive

Screenshot

```
sns.lineplot(x=df_year.index, y=df_year.values, dashes=True, marker='o', color='teal')
plt.xlabel("Years",fontsize=20)
plt.ylabel("Launch Success Rate",fontsize=20)
plt.title('The Yearly Trend for Successful Launches')
plt.show()
```



Observations

- Launch success rate remain low in early years (2010-2013) but started to and continuously increase later
- The trend fluctuated between 2018 and 2020 but is definitely positive

All Launch Site Names (Task 1)

```
%sql SELECT DISTINCT(Launch_Site) from SPACEXTABLE;
```

```
 * sqlite:///my_data1.db
Done.
```

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

Launch Site Names

DISTINCT is used here for the same launch site was used for different Flights.

Launch Site Names Begin with ‘CCA’ (Task 2)

```
%sql SELECT * from SPACEXTABLE WHERE Launch_Site like 'CCA%' LIMIT 5;
```

```
 * sqlite:///my_data1.db
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	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

To select names begin with ‘CCA’, the **WHERE** clause with **like** begins with **CAA** and followed by wild card symbol **%** will do the trick. Limit display to 5 records.

Total Payload Mass by NASA (Task 3)

```
%sql SELECT SUM(PAYLOAD_MASS__KG_) from SPACEXTABLE WHERE Customer like 'NASA%(CRS)%';
```

```
 * sqlite:///my_data1.db
Done.
```

SUM(PAYLOAD_MASS__KG_)
48213

SUM() function was used to calculate the total Payload and the **WHERE** clause selected ‘NASA’ or ”NASA(CRS)’ using ‘**like**’.

Average Payload Mass by F9 v1.1 (Task 4)

```
%sql SELECT AVG(PAYLOAD_MASS__KG_) from SPACEXTABLE WHERE Booster_Version like 'F9 v1.1%';
```

```
 * sqlite:///my_data1.db
Done.
```

AVG(PAYLOAD_MASS__KG_)
2534.6666666666665

AVG() function was used to calculate the average Payload and the **WHERE** clause selected booster that has ‘F9 v1.1’ in the name using ‘**like**’.

First Successful Ground Landing Date (Task 5)

```
%sql SELECT MIN(Date) from SPACEXTABLE WHERE Landing_Outcome = 'Success (ground pad)';
```

```
 * sqlite:///my_data1.db
Done.
```

MIN(Date)
2015-12-22

The first or the earliest was achieved by using **MIN()** function with the **WHERE** clause selecting ‘Success (ground pad)’.

Successful Drone Ship Landing with Payload between 4000 and 6000 (Task 6)

```
%sql SELECT DISTINCT(Booster_Version) from SPACEXTABLE WHERE Landing_Outcome = 'Success (drone ship)' AND PAYLOAD_MASS__KG_ between 4000 and 6000;
```

```
 * sqlite:///my_data1.db
Done.
```

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

DISTINCT was used to retrieve the unique name of Booster Versions and the Payload range was specified in the **WHERE** clause using "**between** … **and** …'.

Total Number of Successful and Failure Mission Outcomes (Task 7)

```
%sql SELECT COUNT(Landing_Outcome) as Total_Success from SPACEXTABLE where Landing_Outcome like 'Success%';
```

```
 * sqlite:///my_data1.db
Done.
```

Total_Success
61

```
%sql SELECT COUNT(Landing_Outcome) as Total_Failure from SPACEXTABLE where Landing_Outcome like 'Failure%';
```

```
 * sqlite:///my_data1.db
Done.
```

Total_Failure
10

- The results were obtained by 2 SQL queries
- **COUNT()** was used to obtain the number of outcomes and the **WHERE** clause selected outcome descriptions containing either ‘Success’ or ‘Failure’ with **like**

Boosters Carried Maximum Payload (Task 8)

```
%sql SELECT DISTINCT(Booster_Version) from SPACEXTABLE where PAYLOAD_MASS__KG_ = (select MAX(PAYLOAD_MASS__KG_) from SPACEXTABLE);
```

```
 * sqlite:///my_data1.db
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

MAX() cannot be used directly in the **WHERE** clause so the **subquery** was used.

2015 Launch Records for ‘Failure (drone ship)’ (Task 9)

```
%sql SELECT substr(Date,6,2) as Month, Booster_Version, Launch_Site, Landing_Outcome \
    from SPACEXTABLE WHERE substr(Date,0,5)='2015' and Landing_Outcome = 'Failure (drone ship)';
```

```
 * sqlite:///my_data1.db
Done.
```

Month	Booster_Version	Launch_Site	Landing_Outcome
01	F9 v1.1 B1012	CCAFS LC-40	Failure (drone ship)
04	F9 v1.1 B1015	CCAFS LC-40	Failure (drone ship)

- **substr(Date, 6,2)** and **substr(Date,0,5)** were used to slice out the strings in the Date that contained **Month** and **Year** information, respectively
- The year and landing outcome were specified in the **WHERE** clause

Rank Landing Outcomes for ‘Failure (drone ship) or ‘Success (ground pad)’ Between 2010-06-04 and 2017-03-20 (Task 10)

```
%sql SELECT Landing_Outcome, COUNT(Landing_Outcome) from SPACEXTABLE \
    WHERE (Landing_Outcome = 'Failure (drone ship)' OR Landing_Outcome = 'Success (ground pad)') \
    AND (Date between '2010-06-04' and '2017-03-20') \
    GROUP BY Landing_Outcome ORDER BY COUNT(Landing_Outcome) DESC;
```

```
 * sqlite:///my_data1.db
Done.
```

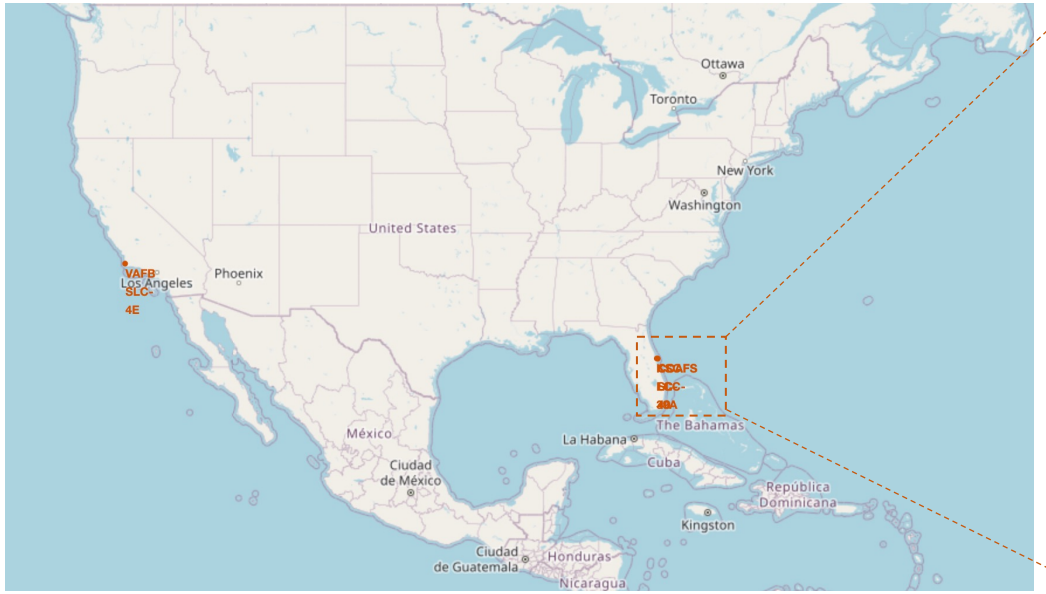
Landing_Outcome	COUNT(Landing_Outcome)
Failure (drone ship)	5
Success (ground pad)	3

- **COUNT()** was used to obtain the **number of outcome** for the selected type in the **WHERE** clause
- The **time constrai**n was also included in the **WHERE** clause
- Rank is for the grouped selected outcome type (**GROUP BY**)
- Finally ranking by **ORDER BY** in descending order (**DESC**)

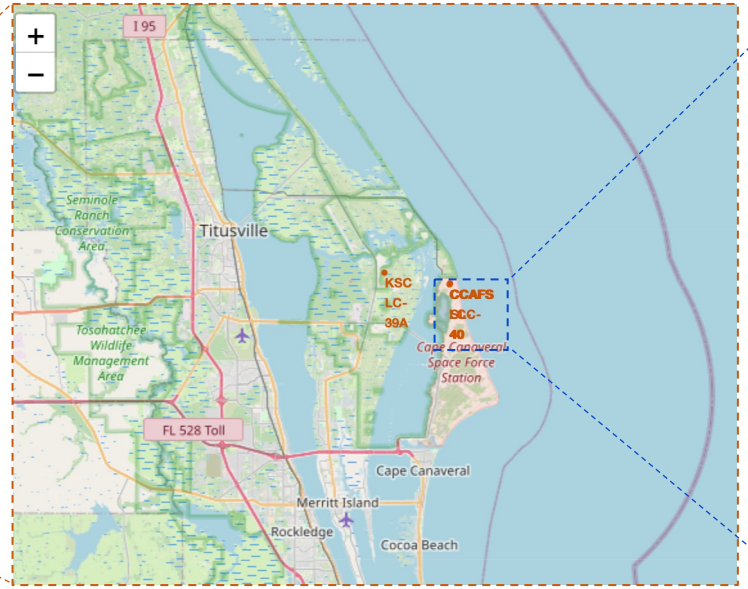

Section 3
Launch Sites
Proximities Analysis

Visualization of Launch Site on the Map using Folium

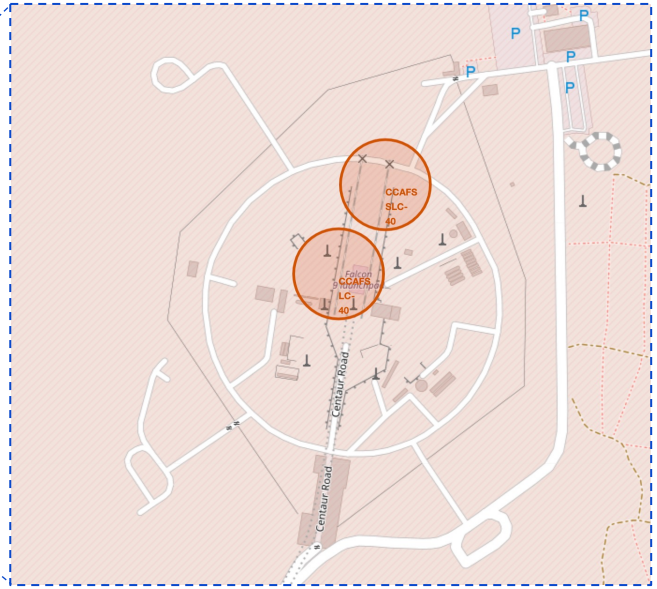
Screenshot 1



Screenshot 2



Screenshot 3



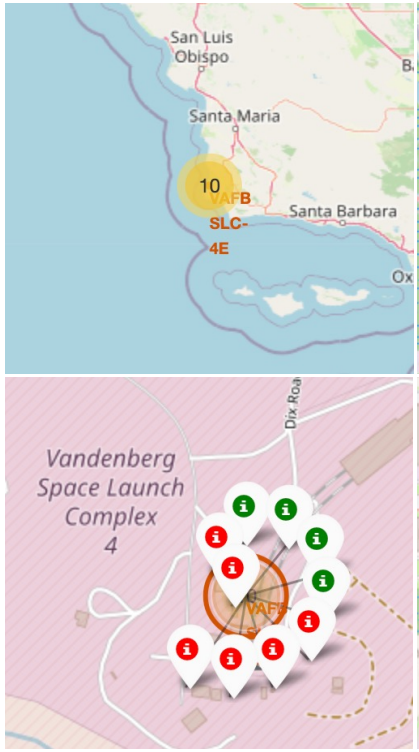
Name of Launch Site

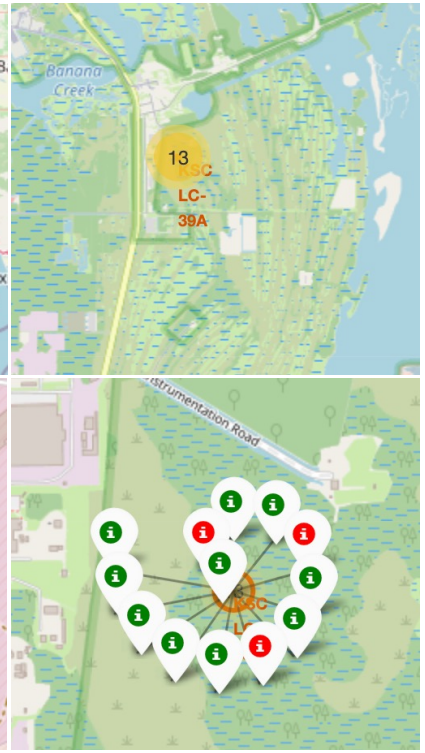
- CCAFS LC-40
- CCAFS SLC-40
- KSC LC-39A
- VAFB SLC-4E

Screenshot Caption Each launch site is labeled with the name (in red) and a filled red circle on the map shown in there different resolutions (*Screenshot 1-3*). Three sites on the East Coast (CCAFS LC-40, CCAFS SLC-40, KSC LC-39A) are overlapping and not resolved in *Screenshot 1*. One site (KSC LC-39A) is further resolved from the other two sites (still overlapping on top of each other) on the East Coast in *Screenshot 2*. The two sites (CCAFS LC-40, CCAFS SLC-40) on the East Coast were seen separated in *Screenshot 3*.

- Launch sites are located on the Coast NOT inland
- One is on the West Coast while three are on the East Coast
- Two launch sites are extremely close to each other on the East Coast
- The above observations suggest **locations for launch sites are limited to the Coast, more East than West**

Visualization of the number of Launches colored by Success and Failure for each Launch Site on the Map using Folium

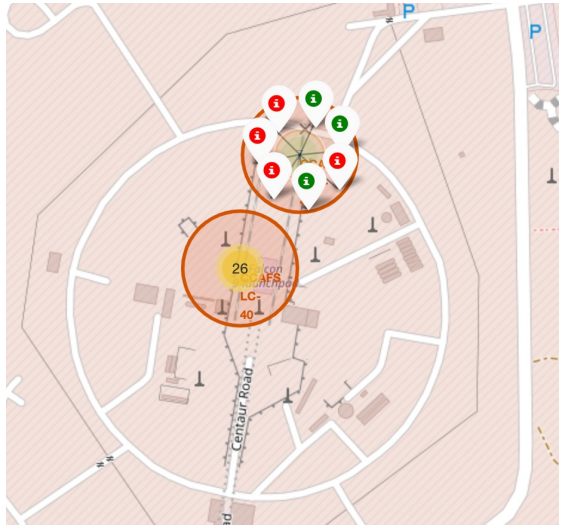




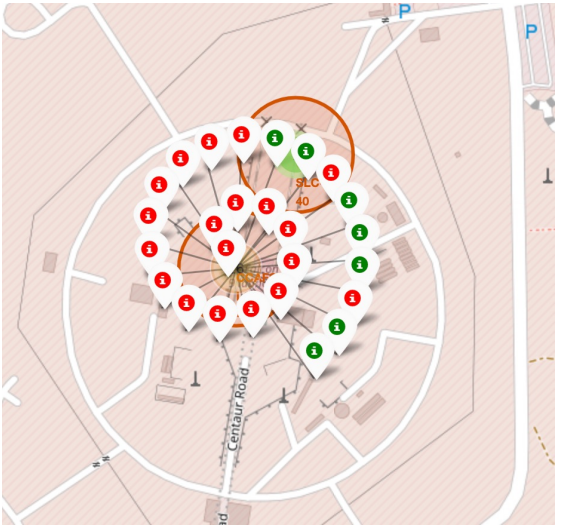
VAFB SLC-4E (10)

KSC LC-39A (13)

CCAFS SLC-40 (7)
CCAFS LC-40 (26)



CCAFS SLC-40 (7)

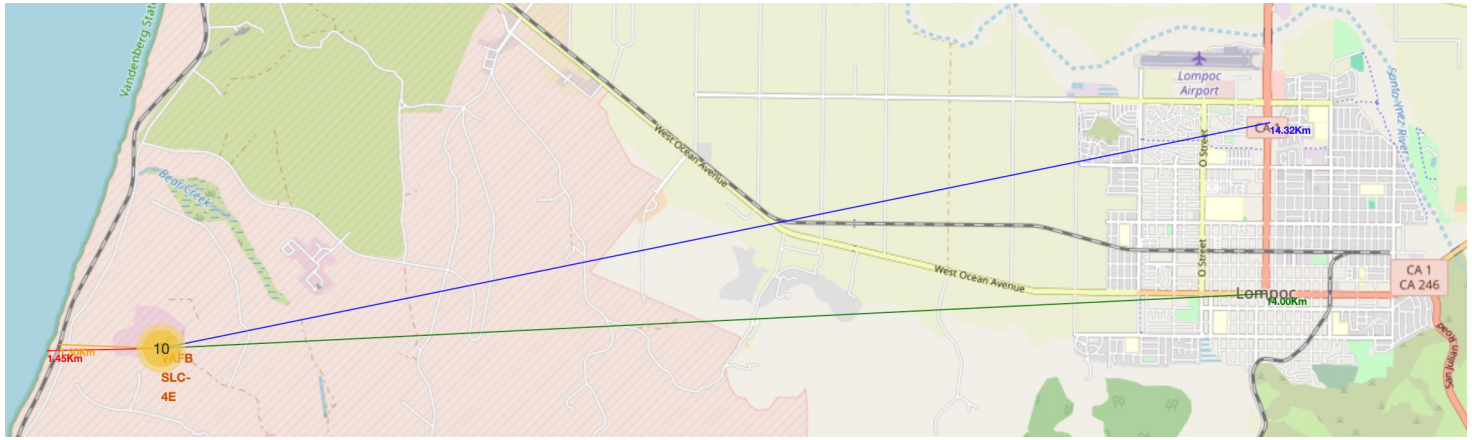


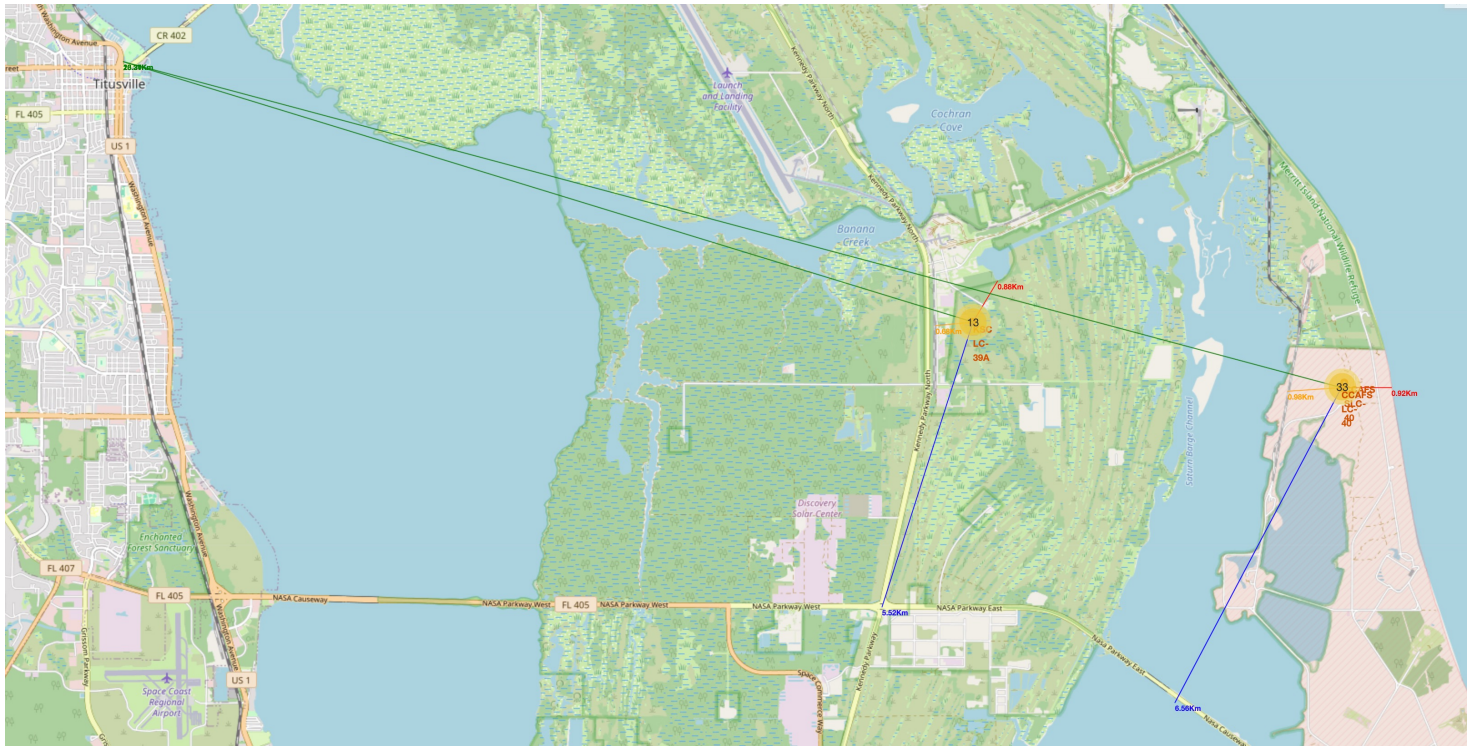
CCAFS LC-40(26)

Screenshot Captions Two types of map images are screen captured here. One with a number in a circle indicating the total number of launches at the site. The other has the same number of water drop markers with either filled green or red dots in the middle indicating successful and failed launches, respectively.

- These map images are visually powerful in conveying information the number and success rate of launches at each launch site directly without having to perform any calculations
- CCAFS LC-40 has the highest (26) number of launches and its neighbor CCAFS SLC-40 has the lowest (7)
- The success rate is the highest at KSC LC-39A site, lowest at CCAFS LC-40 while more or less comparably for the other two sites

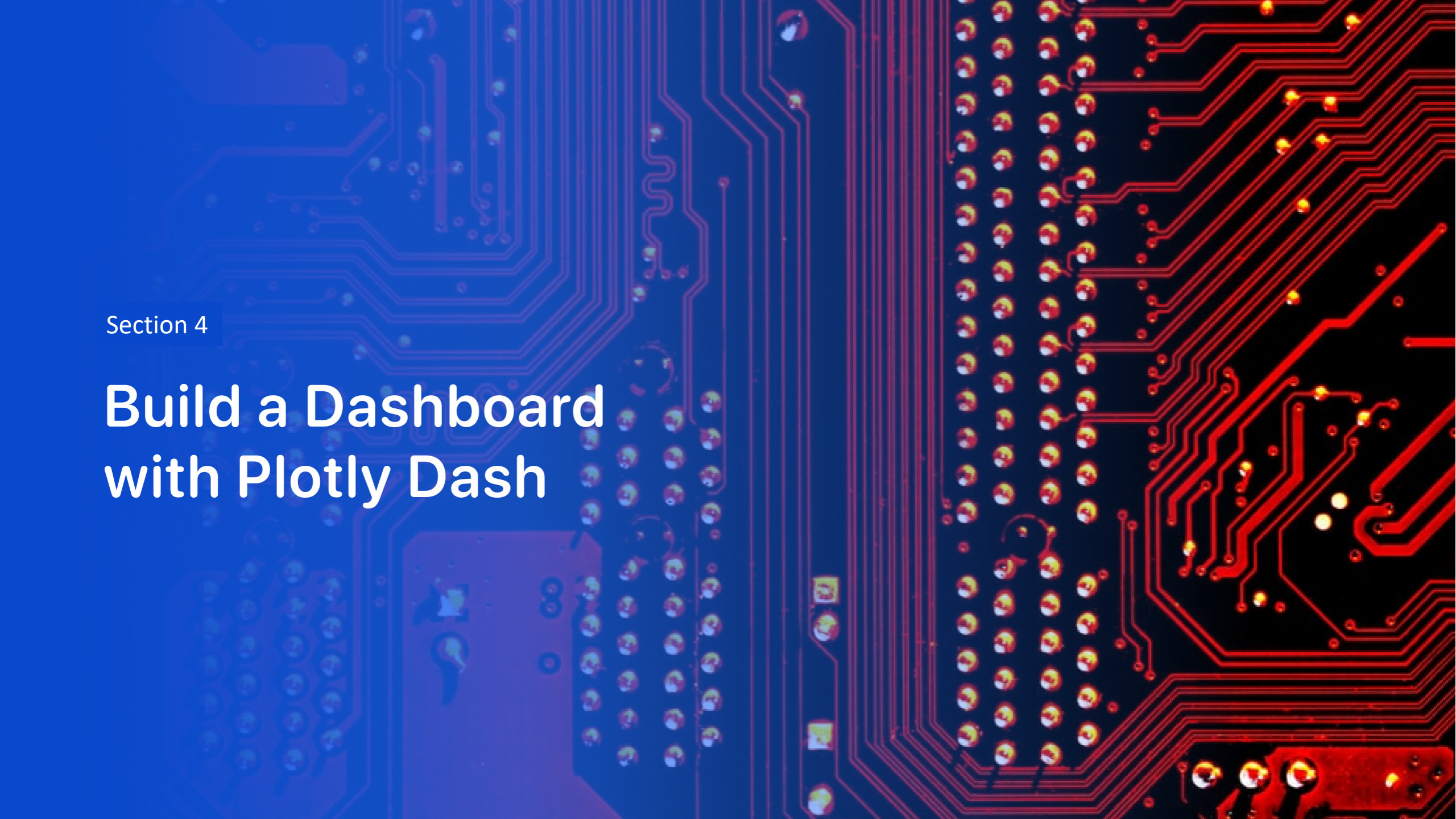
Visualization of labeled distances to coastlines, railways, highways and cities from each Launch Site on the Map using Folium





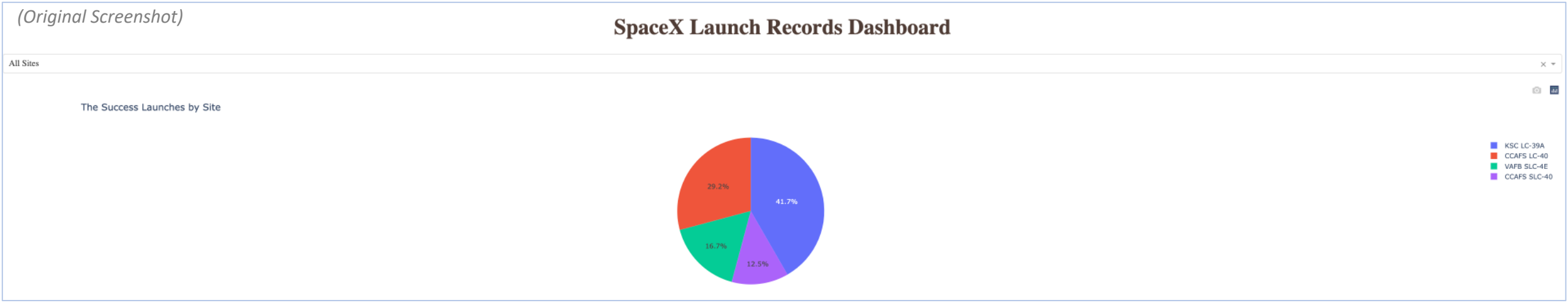
Screenshot Caption Distances to coastlines (red), railways (orange) , highways (blue) and cities (green) from three launch sites (VAFB SLC-4E, KSC LC-39A, and CCAFS LC-40) are shown in by a line and a marker at the end of the line in the same indicated colors.

- A common pattern for the launch sites seen from the labeled distances is that they are close to coastlines (by the nature of locating in the coasts) and railways but much farther from the highways and cities
- This feature is extremely important due to the safety concerns for the launch as well as the infrastructures and people in the surroundings

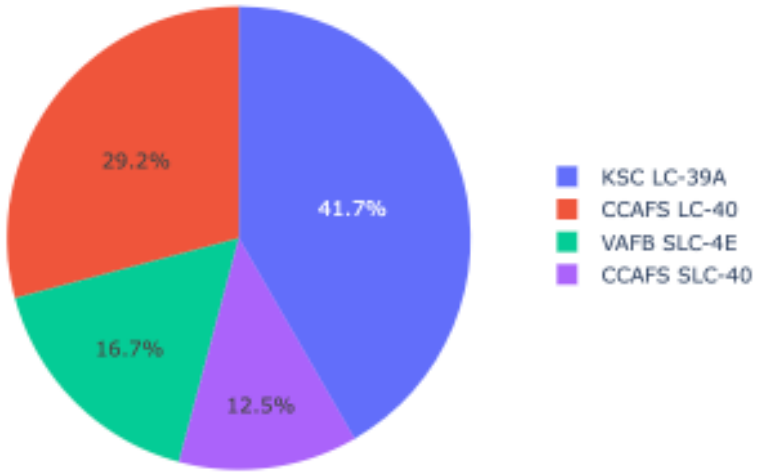

Section 4
Build a Dashboard with Plotly Dash

Interactive Visualization of Successful Launches at All Sites

(Original Screenshot)



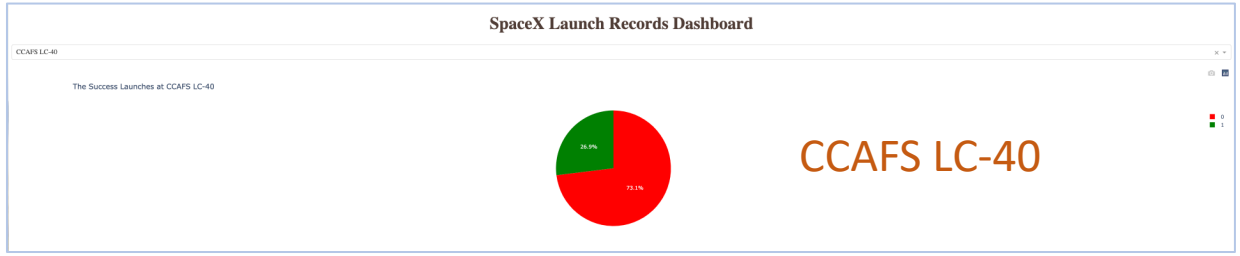
Successful Launches at All Sites



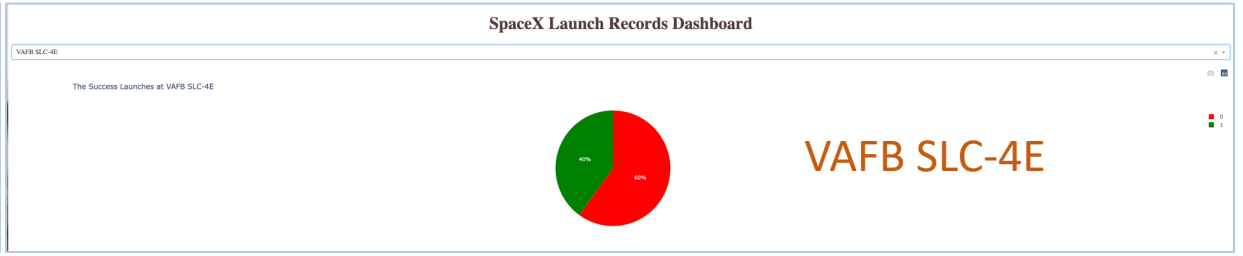
(Reconstructed Figure from the Screenshot)

When considering the total number of successful launches from all sites together, KSC LC-39A has the highest number which represents almost 42% of among all successful launches, while CCAFS SLC-40 has the lowest number

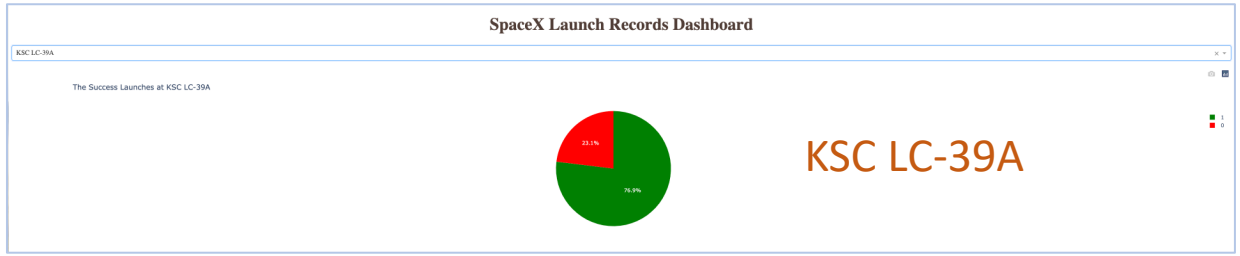
Interactive Visualization of the Success Rate at Each Launch Site



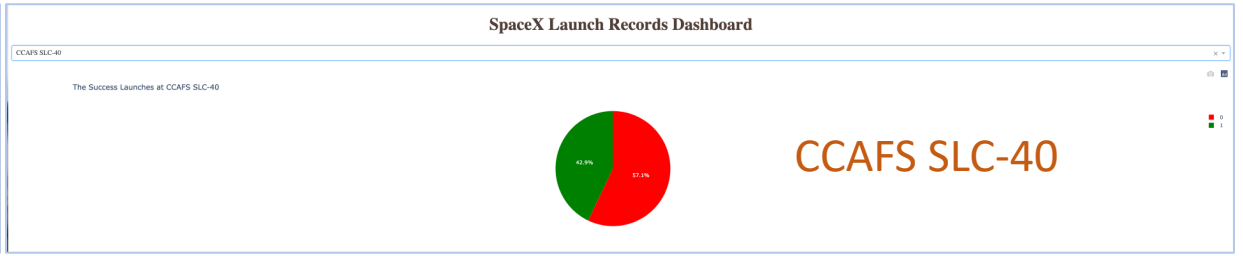
CCAFS LC-40



VAFB SLC-4E



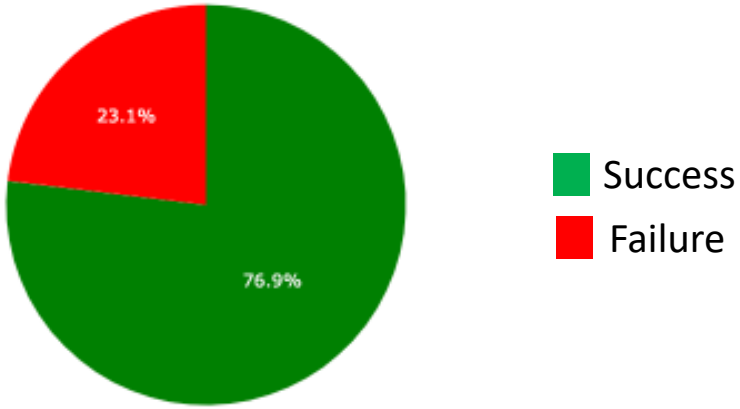
KSC LC-39A



CCAFS SLC-40

(Original Screenshots)

Highest Success Rate at KSC LC-39A



(Reconstructed from the Screenshot)

- KSC LC-39A site has the highest rate for launch success while CCAFS LC-40 has the lowest rate
- This result is consistent with the visualization from the Folium map (Page 42)

Interactive Visualization of Payload Range and Booster Type Associated with Launch Success at All Sites

Payload Range (Kg)

0 – 10K

Booster Version Category
- v1.0
- v1.1
- FT
- B4
- B5

2K – 6K

Booster Version Category
- v1.1
- FT
- B4
- B5

6K – 10K

Booster Version Category
- FT
- B4

- There are more launches with lower Payload and very few with Payload exceeding 9K
- FT Booster has the highest success rate for Payload up to 6K, while B4 might be good for higher Payload but more data is needed

Section 5

Predictive Analysis (Classification)

Classification Accuracy

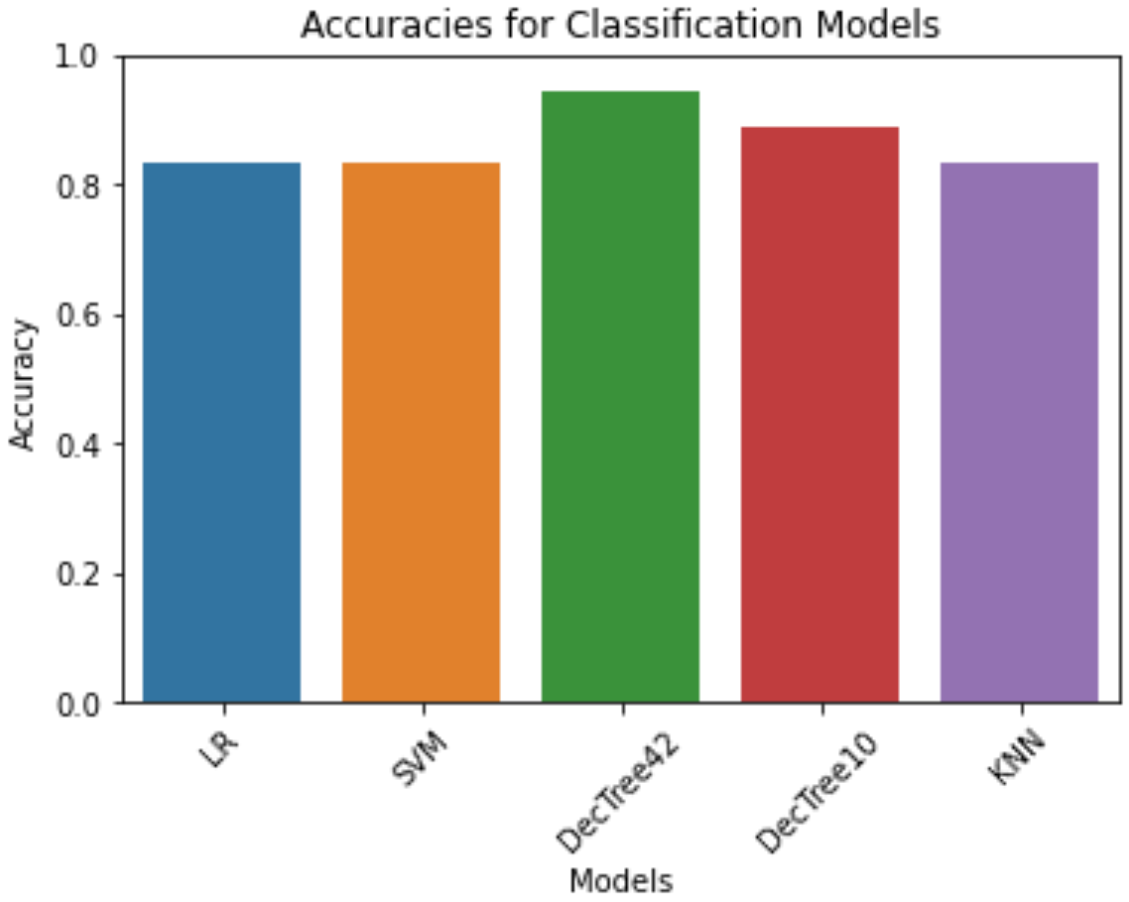
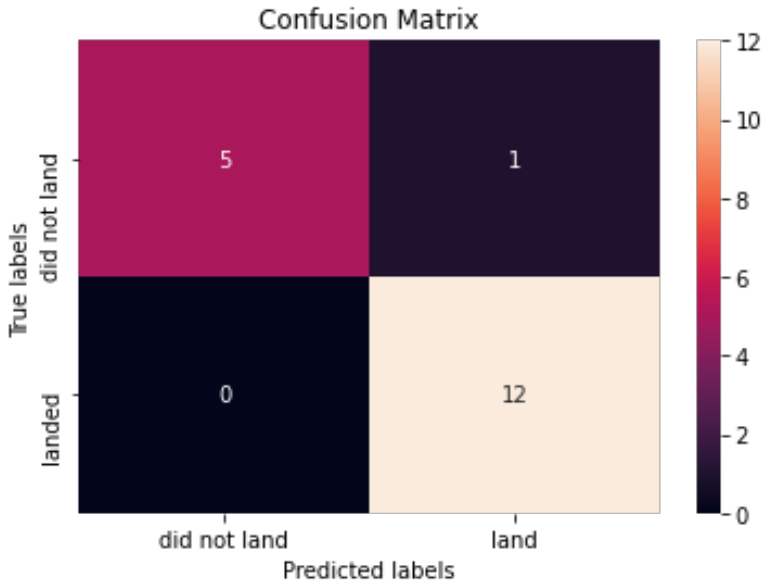


Figure Caption Accuracies using test data for classification models: Logistic Regression (LR), Supporting Vector Machine (SVM), DecisionTree (DecTree42 and DecTree10, for setting values for parameter random_state at 42 and 10, respectively), and K-nearest neighbor (KNN).

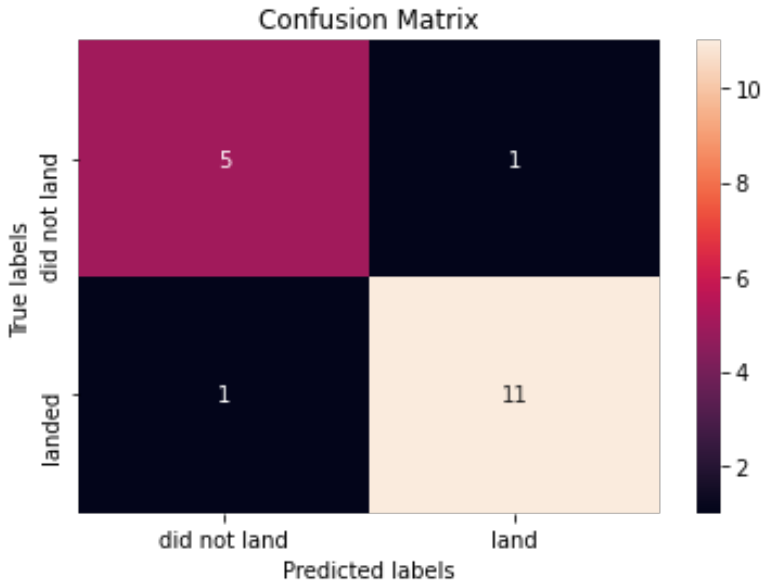
- The outcome from the DecisionTree Model is dependent on the initial setting of the random state
- Results from the common default value 42 and random value 10 are included for the bar chart
- The model that ***can*** result in the highest rate of predicting the launch outcome is the DecisionTree Model, 94%
- However random state set at other values vary and can result in lower accuracies (data not shown)
- The accuracies for the predicting launch success from the other three classification model are exactly the same: 83%

Confusion Matrix

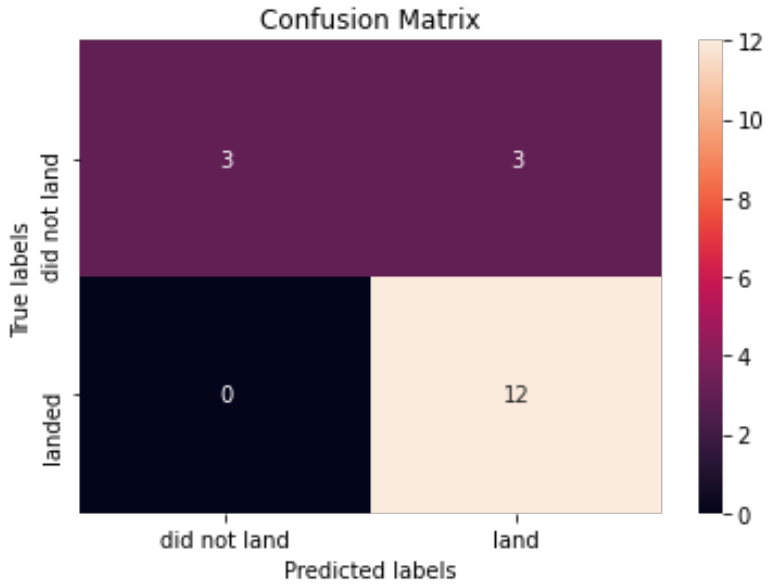
DecisionTree42



DecisionTree10



Logistic Regression (LR)



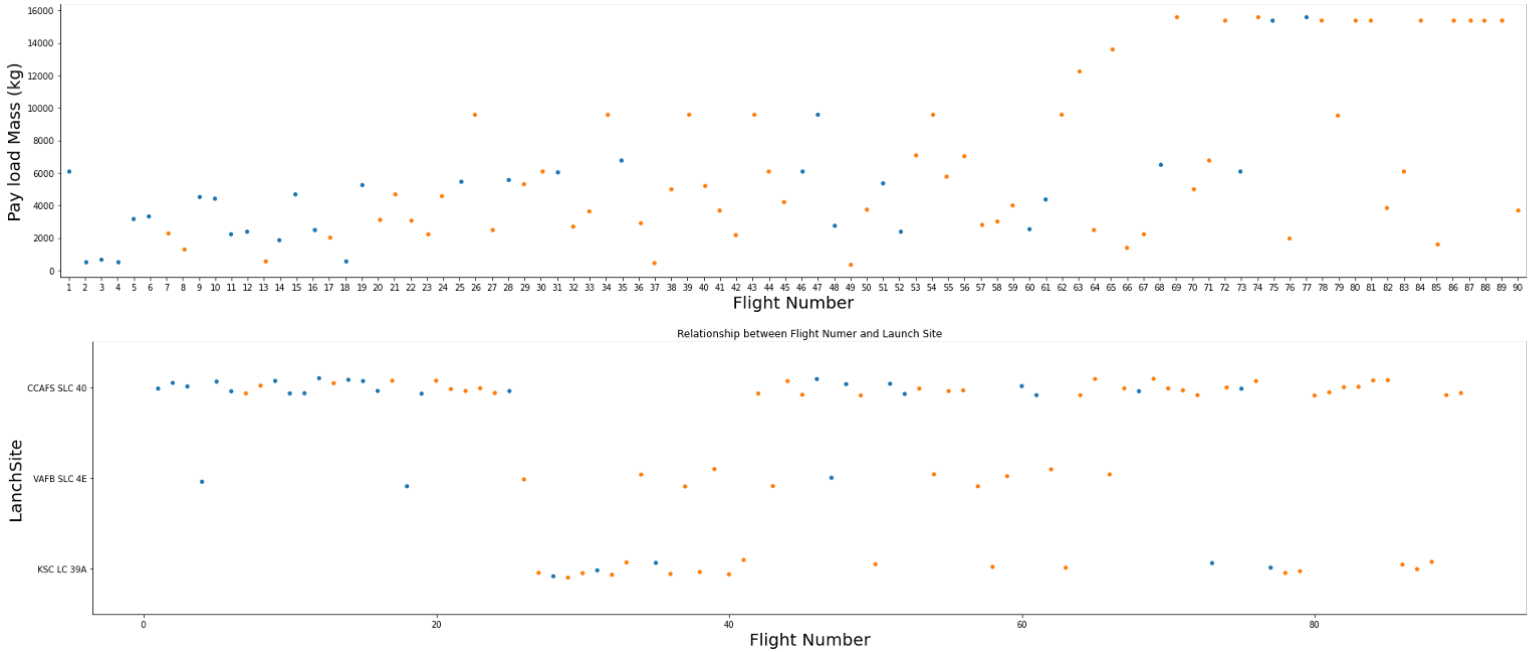
Accuracy	**94.4%**	88.9%	83.3%

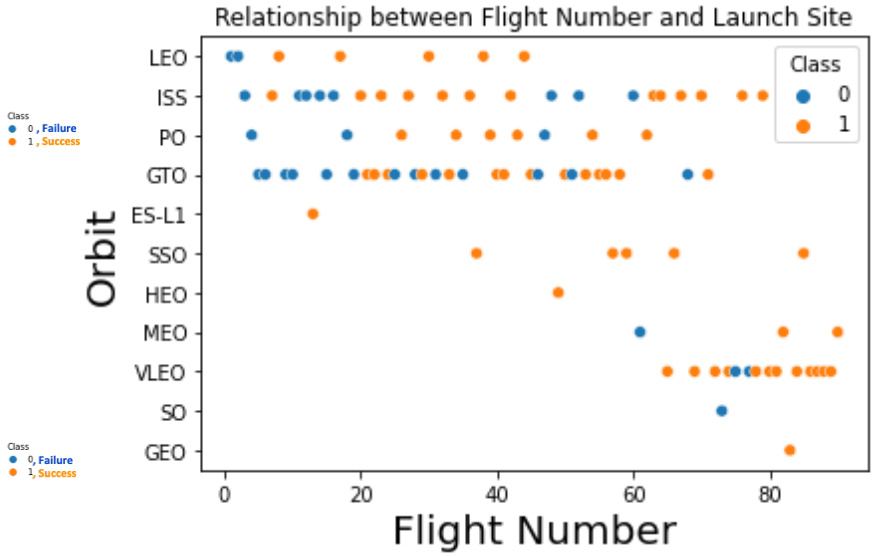
- The name (top) and accuracy (bottom) of the model are also displayed together with the confusion matrix
- The model with the highest accuracy is DecisionTree Model with random_state value set at 42 (DecisionTree42)
- This model correctly predicted all successful launches (meaning no false negative) while predicted **1 false positive** out of 6 failed launches
- DecisionTree10 did have 1 false negative prediction out of 12 successful launches with a different value (10) for the random_state parameter
- The outcomes from three other models are the same as shown in LR. The less accuracy came from the **increased false positive** prediction without any false negative prediction

Conclusion

- The success rate for the first stage rocket landing at SpaceX has continuously increased since 2013 and is relatively stable approaching 80% in recent years (up to 2020) suggesting its reuse is a **feasible and viable** approach
- The success rate is the highest at **KSC LC-39A** launch site reaching almost 78% and 100% to ES-L1, GEO , HEO and SSO orbits
- Success rate is more reliable with payload mass in the range of 2K to 6K (Kg) using **Falcon 9 FT booster**
- Success rate is poor with higher payload and **more data is needed** to determine which booster is best suited for heavy payload launch
- The Decision Tree classification Model can be used to predict the landing success at accuracy as high as **94%** based on the launch features including launch site, payload mass, orbit type, booster version, and others

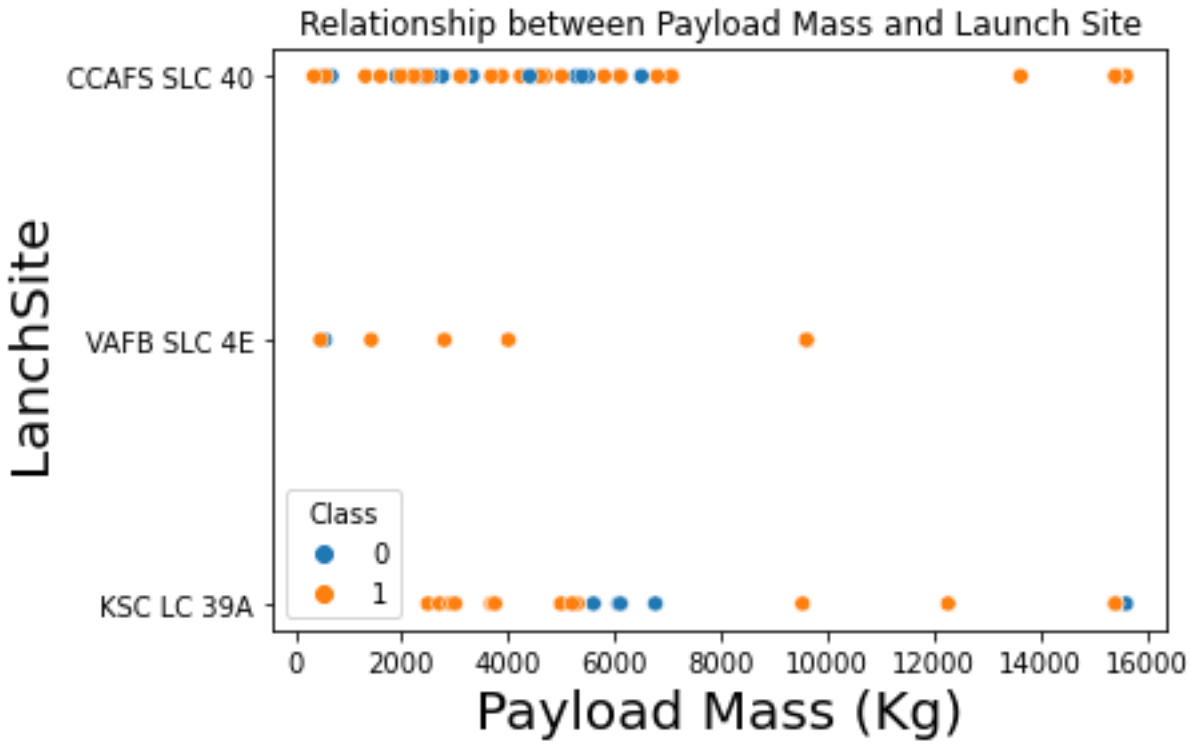
Appendix 1: EDA with Data Visualization (Alternative):
Relationships between Landing Outcomes and Payloads, Launch Sites & Orbits

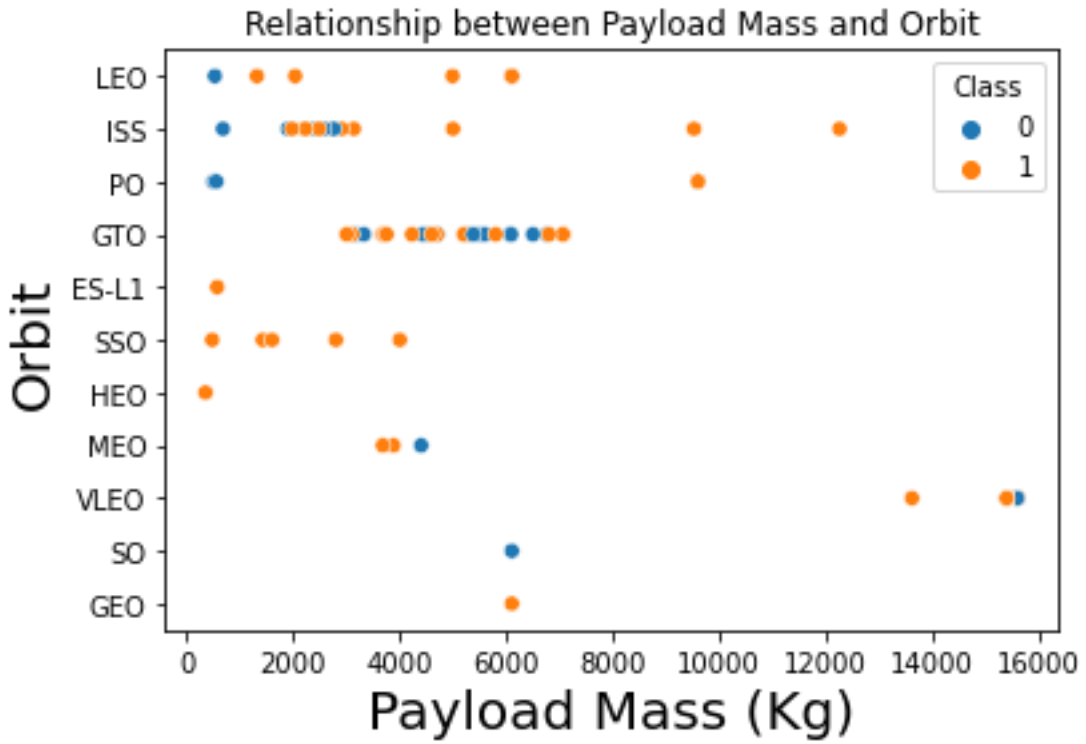




- Higher Payloads were associated with later Flights
- CCAFS SLC-40 has more launches followed by KSC LC39A and VAFB SLC 4E had the lowest number of launches
- Earlier Fights were to a limited number of orbits while later flights went to more different type of orbits
- Though the outcomes of launches with Payloads, launch sites and orbits are mixed, there is a clear trend that **later launches** have **higher success rates** indicating progress

Appendix 2: EDA with Data Visualization (Alternative)
The Relationships between Payloads and Launch Sites, Orbits

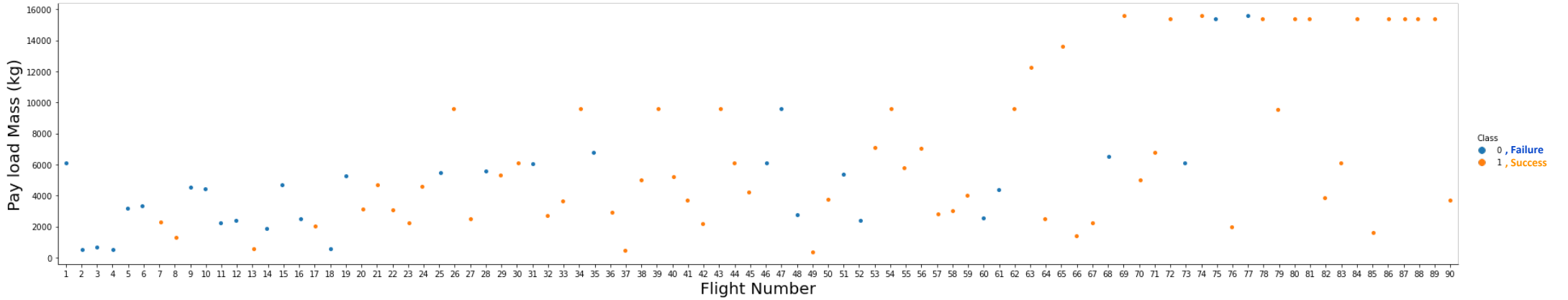


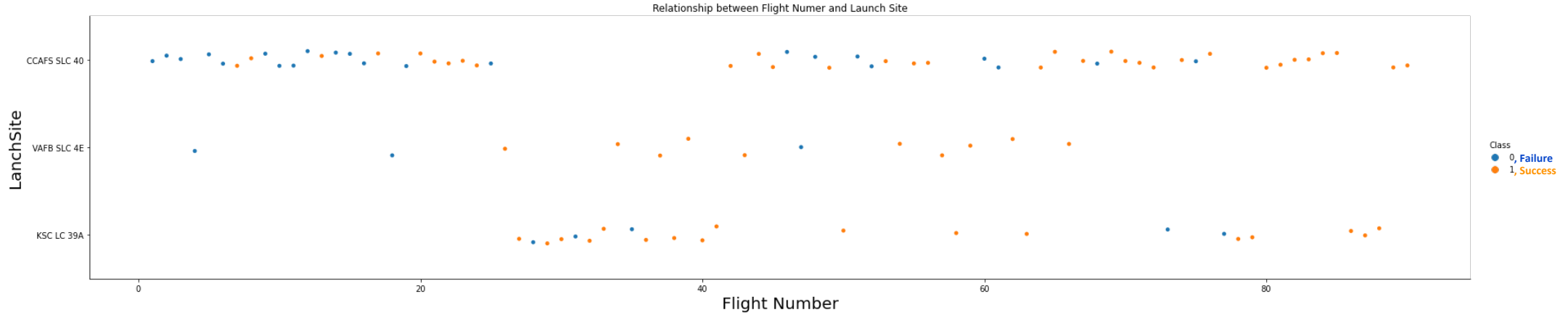


- The Payload at VAFB SLC-4E never exceeded 10K Kg, while a few higher Payloads were launched at both CCAFS SLC-40 and KSC LC-39A sites
- Limited numbers of Payloads higher than 10K were launched to only ISS, PO and VLEO, while most launches are with Payload less than 8K to other orbits

Appendix 3: EDA with Data Visualization (Alternative)

Relationships between Landing Outcomes and Payloads, Launch Sites & Orbit





- Higher Payloads were associated with Later Flights
- CCAFS SLC-40 has the highest number of launches followed by KSC LC39A and VAFB SLC 4E had the lowest number of launches
- Though the outcomes of launches with different Payloads and at different launch sites are mixed, there is a clear trend that **later launches** have **higher success rates** indicating progress

Appendix 4: Additional Useful SQL Queries

Different descriptions for Landing Outcome

```
%sql SELECT DISTINCT(Landing_Outcome) from SPACEXTABLE;
```

✓ 0.0s

```
* sqlite:///my_data1.db
Done.
```

Landing_Outcome
Failure (parachute)
No attempt
Uncontrolled (ocean)
Controlled (ocean)
Failure (drone ship)
Precluded (drone ship)
Success (ground pad)
Success (drone ship)
Success
Failure
No attempt

Default grouping categories in SQL for Landing Outcome

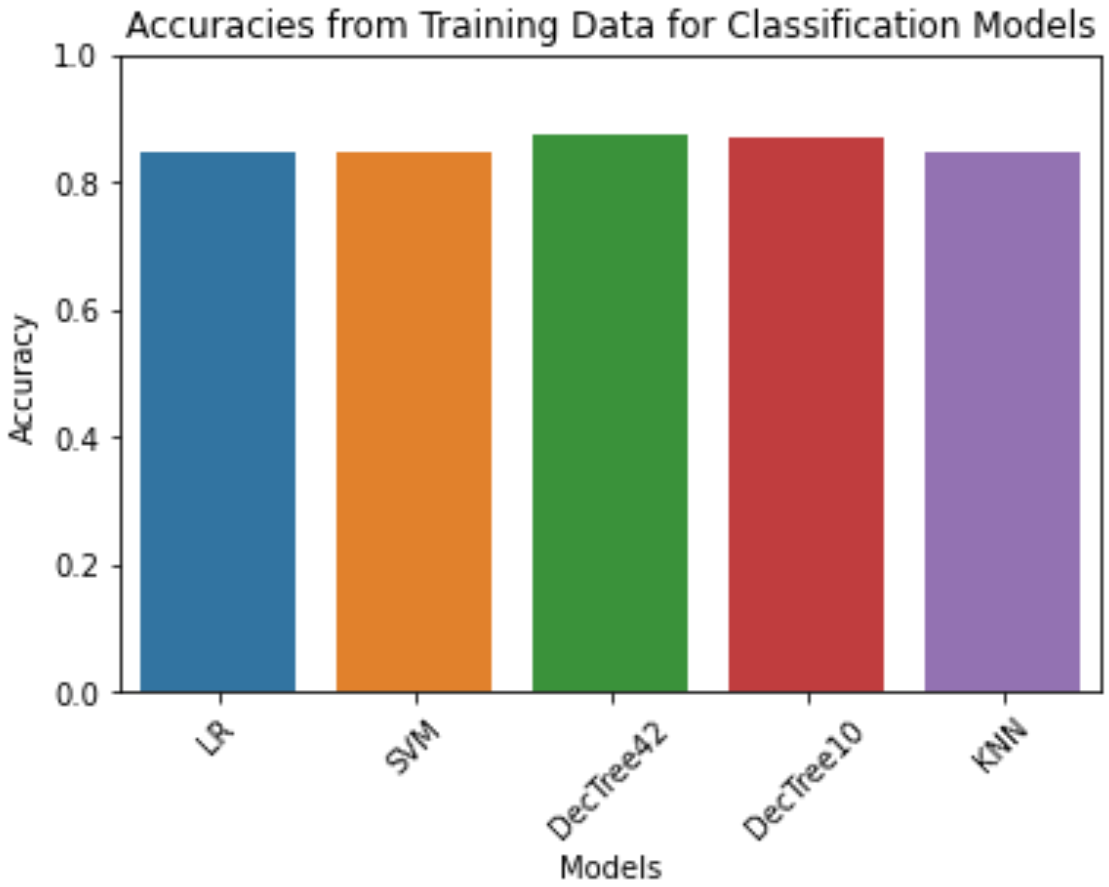
```
%sql SELECT Landing_outcome, COUNT(Landing_Outcome) from SPACEXTABLE \
    GROUP BY Landing_Outcome having Date between '2010-06-04' and '2017-03-20' \
        ORDER BY COUNT(Landing_Outcome) DESC;
```

✓ 0.0s

```
* sqlite:///my_data1.db
Done.
```

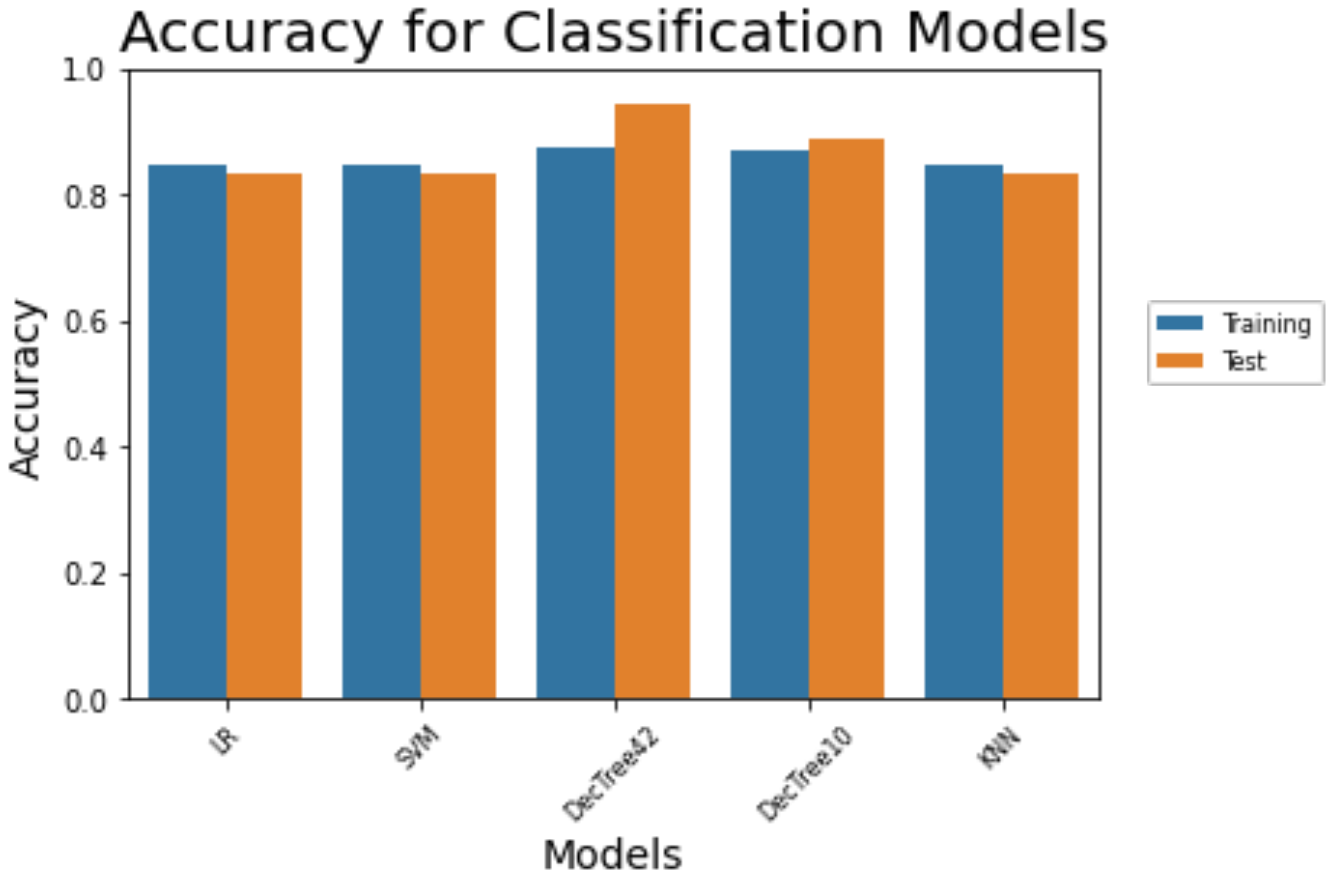
Landing_Outcome	COUNT(Landing_Outcome)
No attempt	21
Success (drone ship)	14
Success (ground pad)	9
Failure (drone ship)	5
Controlled (ocean)	5
Uncontrolled (ocean)	2
Failure (parachute)	2
Precluded (drone ship)	1

Appendix 5:
Accuracy from training data for classification models



- Accuracies from training data varied less (between 85% to 87%) for all tested model predictions suggesting the models did **comparable jobs** in predicting the outcomes on the training data

Appendix 6: Comparison in Model Predicting Accuracies between the Training and Test Data



- LR, SVM and KNN models behaved similarly while the Decision Tree model behaved somewhat differently
- Considering also that the Decision Tree prediction varied on the setting of random_state value

Thank you!