IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Raphael Mazini Aguiar
Data Scientist

April 11, 2023

Outline

- Sumário executivo
- Introdução
- Metodologia
- Resultados
- Conclusão
- Apêndice

Executive Summary

- ## Summary of methodologies

Data collection – coleta de dados

Data wrangling – Disputa de dados

EDA with data visualization

EDA with SQL

Building an interactive map with Folium

Building a Dashboard with Plotly Dash

Predictive analysis (Classification)

- ## Summary of all results

Exploratory data analysis results

Interactive analytics demo in screenshots

Predictive analysis results

Introduction

- ## Contexto e plano de fundo do projeto

Devido ao desenvolvimento tecnológico que estamos experimentando atualmente, empresas privadas como a SpaceX agora estão autorizadas a desenvolver e comercializar viagens espaciais. A empresa SpaceX anuncia a distribuição de viagens ao espaço e anuncia ao custo de cada lançamento a um valor de US $ 62 milhões de dólares, enquanto outras empresas anunciam ao custo de US $ 165 milhões de dólares cada lançamento.

Mas para a SpaceX ser capaz de obter o custo de US $ 62 milhões para cada lançamento, será necessário reutilizar o primeiro estágio, e para isso prevemos, através dos dados que temos no momento, se o primeiro estágio pousará com sucesso ou não, se o pouso de retorno for bem-sucedido, o custo estará dentro de US $ 62 milhões. E para isso, o Machine Learning foi usado para um modelo de classificação.

- ## Problemas que você deseja encontrar respostas

O que influencia o Falcon 9 a ter sucesso em seu pouso de retorno?

Que efeitos tem cada relação entre as variáveis na determinação da taxa de sucesso do pouso de retorno?

Que condições a SpaceX precisa alcançar, para obter os melhores resultados e garantir a melhor taxa de sucesso em seus pousos de retorno do Falcon 9?

Section 1

Methodology

Methodology

- ## Metodologia de coleta de dados:

Trabalhamos com dados de lançamento da SpaceX coletados de uma API, especificamente a API REST da SpaceX (**https://api.spacexdata.com/v4/launches/past**). Esta API nos dará dados sobre lançamentos, incluindo informações sobre o foguete usado, carga útil entregue, especificações de lançamento, especificações de pouso e resultado do pouso.

O método Scraping web também foi usado para coletar dados de registros históricos de lançamento do Falcon 9, de uma página da Wikipédia intitulada "Lista de lançamentos do Falcon 9 e Falcon Heavy", que foram raspados. (https://en.wikipedia.org/wiki/List_of_Falcon/_9/_and_Falcon_Heavy_launches). Todos os dados foram tabulados e as respectivas tabelas transformadas em DataFrame, para que tenhamos maior flexibilidade no tratamento e análise dos dados.

- ## Executar disputa de dados

Em meio ao processo de coleta de dados, os dados faltantes, nulos, foram encontrados na coluna "PayloadMass", e foram devidamente tratados, sendo substituídos pela média dos mesmos recursos dos quais faziam parte. Há um total de 90 lançamentos, 55 para o local "CCAFS SLC 40", 22 para o local "KSC LC 39A" e 13 para o local "VAFB SLC 4E". O site KSC LC-39A localizado no Condado de Brevard, Flórida, EUA. Site do CCAFS LC-40 localizado em Cabo Canaveral, FL, EUA. O site VAFB SLC-4E está localizado no Condado de Santa Barbara, CA, EUA. de cada órbita, saber qual a órbita que tem o maior número de lançamentos e a menor entre as órbitas: 'LEO', 'ISS', 'PO', GTO', 'ES-L1', 'SSO', 'HEO' , 'MEO', 'VLEO'. A órbita com o maior número foi a órbita "GTO" com 27 lançamentos, e as órbitas com o menor número de lançamentos foram: "GEO", "ES-L1", "HEO", "SO" com apenas 1 lançamento. Que transformaríamos dados brutos em um conjunto de dados limpos que forneceriam informações sobre a situação que estamos tentando resolver. Os dados coletados foram enriquecidos com a criação de um rótulo de resultado de aterrissagem, com base nos dados do resultado após uma síntese e análise dos recursos, e obtivemos a informação de que "Classe = 1(Amarelo)" significa sucesso e "Classe = 0(Azul)" significa fracasso

Methodology

- **Executar análise exploratória de dados (EDA) usando visualização em SQL**
- Plotagem: Gráficos de dispersão, gráficos de barras para mostrar relações entre variáveis para mostrar padrões de dados.
- **Execute análises visuais interativas usando o Folium e o Plotly Dash**
- **Realizar análise preditiva usando modelos de classificação**
- Como construir, ajustar, avaliar modelos de classificação

Data Collection

- ## Descrição do conjunto de dados..

Trabalhamos com dados de lançamento da SpaceX coletados de uma API, especificamente a API REST da SpaceX

O método web Scraping também foi usado para coletar dados de registros históricos de lançamento do Falcon 9, de uma página da Wikipédia intitulada "Lista de lançamentos do Falcon 9 e Falcon Heavy"

Os seguintes conjuntos de dados foram coletados por meio de uma API REST da SpaceX: Trabalhamos com dados de lançamento da SpaceX., a API nos dará dados sobre lançamentos, incluindo informações sobre o foguete usado, carga útil entregue, especificações de lançamento, resultado do pouso de especificações. Nosso objetivo é usar esses dados para prever se a SpaceX tentará lançar um foguete ou não. Os pontos de extremidade da API REST da SpaceX, ou URLs, começam com api.spacexdata.com/v4/.Another fonte de dados popular para obter dados de lançamento do Falcon 9 é a coleta de dados da Wikipédia usando o Beautiful Soup.

Data Collection
SpaceX API

SpaceX API

Use SpaceX REST API → API returns SpaceX data in .JSON → Normalize data into flat data file such as .csv

Esta API nos dará dados sobre lançamentos, incluindo informações sobre o foguete usado, carga útil entregue, especificações de lançamento, especificações de pouso e resultado do pouso

Github [Data_Colletion_API]

simplified flow chart

1 .Getting Response from API

```
spacex_url="https://api.spacexdata.com/v4/launches/past"
response = requests.get(spacex_url).json()
```

2. Converting Response to a .json file

```
response = requests.get(static_json_url).json()
data = pd.json_normalize(response)
```

3. Apply custom functions to clean data

```
getLaunchSite(data)
getPayloadData(data)
getCoreData(data)
```

```
getBoosterVersion(data)
```

4. Assign list to dictionary then dataframe

```
launch_dict = {'FlightNumber': list(data['flight_number']),
'Date': list(data['date']),
'BoosterVersion':BoosterVersion,
'PayloadMass':PayloadMass,
'Orbit':Orbit,
'LaunchSite':LaunchSite,
'Outcome':Outcome,
'Flights':Flights,
'GridFins':GridFins,
'Reused':Reused,
'Legs':Legs,
'LandingPad':LandingPad,
'Block':Block,
'ReusedCount':ReusedCount,
'Serial':Serial,
'Longitude': Longitude,
'Latitude': Latitude}
```

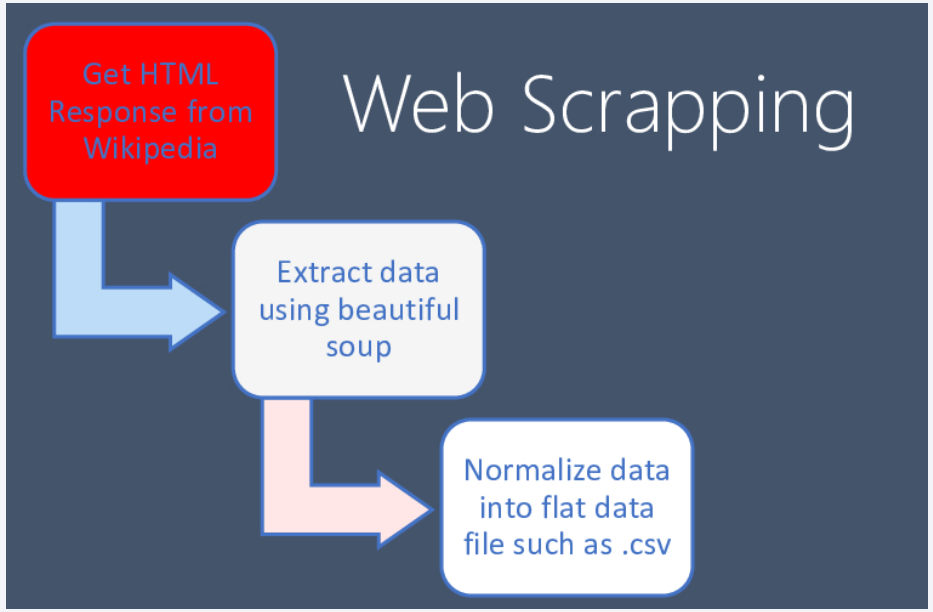
```
df = pd.DataFrame.from_dict(launch_dict)
```

5. Filter dataframe and export to flat file (.csv)

```
data_falcon9 = df.loc[df['BoosterVersion']!="Falcon 1"]
```

```
data_falcon9.to_csv('dataset_part_1.csv', index=False)
```

Data Collection Scraping



O método web Scraping também foi usado para coletar dados de registros históricos de lançamento do Falcon 9, de uma página da Wikipédia intitulada "Lista de lançamentos do Falcon 9 e Falcon Heavy", que foram raspados.

Github: [Data_Collation_Scraping]

simplified flow chart

1 .Getting Response from HTML

```
page = requests.get(static_url)
```

2. Creating BeautifulSoup Object

```
soup = BeautifulSoup(page.text, 'html.parser')
```

3. Finding tables

```
html_tables = soup.find_all('table')
```

4. Getting column names

```
column_names = []
temp = soup.find_all('th')
for x in range(len(temp)):
    try:
     name = extract_column_from_header(temp[x])
     if (name is not None and len(name) > 0):
        column_names.append(name)
    except:
     pass
```

5. Creation of dictionary

```
launch_dict= dict.fromkeys(column_names)

# Remove an irrelvant column
del launch_dict['Date and time ( )']


launch_dict['Flight No.'] = []
launch_dict['Launch site'] = []
launch_dict['Payload'] = []
launch_dict['Payload mass'] = []
launch_dict['Orbit'] = []
launch_dict['Customer'] = []
launch_dict['Launch outcome'] = []
launch_dict['Version Booster']=[]
launch_dict['Booster landing']=[]
launch_dict['Date']=[]
launch_dict['Time']=[]
```

6. Appending data to keys (refer) to notebook block 12

```
In [12]: extracted_row = 0
         #Extract each table
         for table_number,table in enumerate(
            # get table row
             for rows in table.find_all("tr")
                 #check to see if first table
```

7. Converting dictionary to dataframe

```
df = pd.DataFrame.from_dict(launch_dict)
```

8. Dataframe to .CSV

```
df.to_csv('spacex_web_scraped.csv', index=False)
```

9

Data Wrangling

No conjunto de dados, há vários casos diferentes em que o booster não pousou com sucesso. Às vezes, um pouso foi tentado, mas falhou devido a um acidente; por exemplo, True Ocean significa que o resultado da missão foi desembarcado com sucesso em uma região específica do oceano, enquanto False Ocean significa que o resultado da missão foi desembarcado sem sucesso em uma região específica do oceano. True RTLS significa que o resultado da missão foi pousado com sucesso em uma plataforma terrestre False RTLS significa que o resultado da missão foi pousado sem sucesso em uma plataforma terrestre. O verdadeiro ASDS significa que o resultado da missão foi pousado com sucesso em um navio drone O falso ASDS significa que o resultado da missão foi pousado sem sucesso em um navio drone. Convertemos principalmente esses resultados em Rótulos de Treinamento com 1 significa que o impulsionador aterrissou com sucesso 0 significa que não foi bem-sucedido.

Process

Perform Exploratory Data Analysis EDA on dataset

Calculate the number of launches at each site

Calculate the number and occurrence of each orbit

Calculate the number and occurrence of mission outcome per orbit type

Export dataset as .CSV

Create a landing outcome label from Outcome column

Work out success rate for every landing in dataset

Each launch aims to an dedicated orbit, and here are some common orbit types:

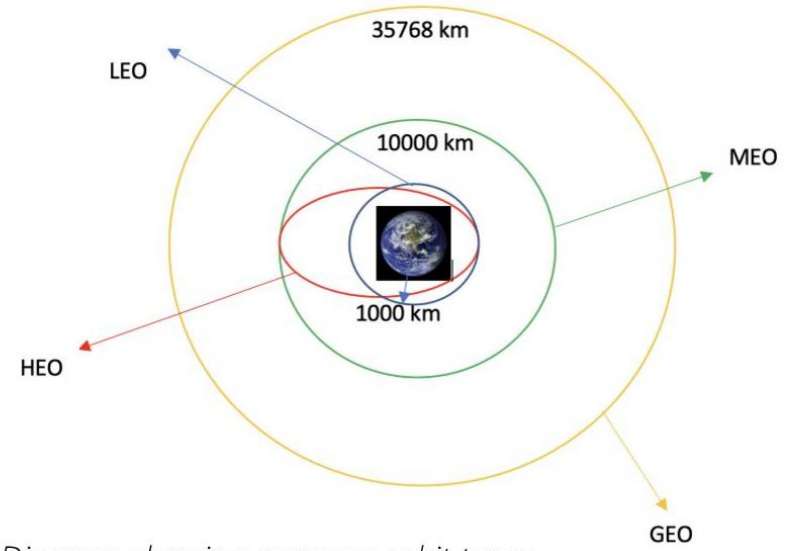


Diagram showing common orbit types SpaceX uses

Github: Data_Wrangling

EDA with Data Visualization

Gráficos de dispersão

Flight Number **VS.** Payload Mass

- Flight Number **VS**. Launch Site
- PayloadMass **VS.** Launch Site
- Orbit **VS**. Flight Number
- PayloadMass **VS.** Orbit

Os gráficos de dispersão mostram o quanto uma variável é afetada por outra. A relação entre duas variáveis é chamada de correlação. Os gráficos de dispersão geralmente consistem em um grande conjunto de dados.

Github: EDA_Visualization

Gráfico de barras

- Success Rate VS. Orbit

Um diagrama de barras facilita a comparação de conjuntos de dados entre diferentes grupos em um piscar de olhos. O gráfico representa categorias em um eixo e um valor discreto no outro. O objetivo é mostrar a relação entre os dois eixos. Os gráficos de barras também podem mostrar grandes mudanças no dat ao longo do tempo.

Gráfico de linhas

- Success Rate **VS.** Year

Os gráficos de linhas são úteis na medida em que mostram variáveis de dados e tendências com muita clareza e podem ajudar a fazer previsões sobre os resultados de dados ainda não registrados.

EDA WITH SQL

Executou consultas SQL para coletar informações sobre o conjunto de dados. Por exemplo, de algumas perguntas que nos foram feitas sobre os dados sobre os quais precisávamos de informações. Quais estamos usando consultas SQL para obter as respostas no conjunto de dados :

- Exibindo os nomes dos locais de lançamento exclusivos na missão espacial
- Exibindo 5 registros onde os locais de lançamento começam com a string 'KSC'
- Exibindo a massa total da carga útil transportada pelos propulsores lançados pela NASA (CRS)
- Exibindo a massa média de carga útil transportada pelo booster versão F9 v1.1
- Listar a data em que o resultado bem-sucedido do pouso em um navio drone foi alcançado.
- Listando os nomes dos propulsores que têm sucesso na almofada de terra e têm massa de carga útil superior a 4000, mas inferior a 6000
- Listar o número total de resultados bem-sucedidos e fracassados da missão
- Listar os nomes dos booster_versions que carregaram a massa máxima de carga útil.
- Listando os registros que exibirão os nomes dos meses, ilanding_outcomes bem-sucedidos no ground pad, versões de booster, launch_site para os meses do ano de 2017
- Classificação da contagem de landing_outcomes bem-sucedidas entre a data de 04/06/2010 e 2017/03/20 em ordem decrescente.

Github: EDA_SQL

Build an Interactive Map with Folium

A taxa de sucesso do lançamento pode depender de muitos fatores, como massa de carga útil, tipo de órbita e assim por diante. Também pode depender da localização e proximidade de um local de lançamento, ou seja, a posição inicial das trajetórias dos foguetes. Como os locais de lançamento exercem uma forte influência no sucesso da missão, analisamos a oposição geográfica desses pontos e entendemos sua localização. O mapa ilustra o posicionamento dos locais de lançamento CCAFS LC-40, CCAFS SLC-40, KSC LC-39A, VAFB SLC-4E. Todos os locais de lançamento não estão perto do equador, eles estão a aproximadamente 30 ° N do equador, e todos os locais de lançamento estão muito perto da costa. Observe que as postagens ocorrem apenas em um dos quatro locais de postagem, o que significa que muitos registros de postagem terão exatamente a mesma coordenação.

Github: [Foium_maps]

- Os locais de lançamento estão próximos dos caminhos-de-ferro? Não
- Os locais de lançamento estão próximos das rodovias? Não
- Os locais de lançamento estão próximos da costa? Sim
- Os locais de lançamento mantêm uma certa distância das cidades? Sim
- Folium.Marker() foi usado para criar marcas nos mapas
- Folium. Circle() foi usado para criar um círculo acima dos marcadores nos mapas.
- Folium. Icon() foi usado para criar um ícone nos mapas.
- Folium. PolyLine() foi usado para criar uma linha polinomial entre os pontos.
- markerCluster() foi usado para simplificar os mapas que contêm vários marcadores com coordenação idêntica

Build a Dashboard with Plotly Dash

- O painel é construído com o framework web Flask e Dash.
- O gráfico de dispersão mostra a relação com a massa útil (Kg) para as diferentes versões de Booster
- A base do painel interativo é mostrar a relação entre duas variáveis.
- Melhor método para mostrar um padrão não linear. O intervalo de fluxo de dados, ou seja, o valor máximo e mínimo, pode ser determinado.
- Gráfico de pizza mostrando o total de lançamentos para um determinado site/todos os sites
- Exibir proporções relativas a várias classes de dados.
- O tamanho do círculo pode ser proporcional à quantidade total que ele representa.
- Os componentes Dash e html foram usados, pois são a coisa mais importante e quase tudo depende deles, como gráficos, tabelas, dropdowns, etc.
- O Pandas foi usado para simplificar o trabalho criando um dataframe.
- Plotly foi usado para plotar gráficos.
- Gráfico de pizza e gráfico de dispersão usados para fins de plotagem.
- O controle deslizante de alcance foi usado para a seleção da faixa de massa da carga útil.
- A lista suspensa foi usada para sites de lançamento. **Github: Dash_plotly**

Predictive Analysis (Classification)

- No processo, continuamos a iniciar uma análise preditiva, a fim de prever se o primeiro estágio do Falcon 9 será bem-sucedido em seu pouso de retorno. O pré-processamento foi realizado nos dados, e dividimos o conjunto de dados em um conjunto de dados para treinamento, para o modelo, e um conjunto de dados para teste, para validação do mesmo modelo em questão, e lembrando que esses dados do mesmo conjunto de dados. O modelo será treinado, e a Pesquisa em Grade será aplicada, permitindo encontrar os hiperparâmetros que determinaram que um determinado algoritmo tem o melhor desempenho. Usando os melhores valores de hiperparâmetros, o modelo com a melhor precisão será determinado. Os algoritmos testados, para a confecção do modelo, foram aqueles relacionados à classificação de quais se encaixavam no problema a ser resolvido, e foram os seguintes: **Classifier Tree of Decision** , **Logistic Regression**, **Support Vector Machines (SVM)** and **K Nearest Neighbors (KNN).**
- Para que possamos visualizar o desempenho dos modelos, foi extraída uma Matriz de Confusão, para determinar o real valor dos erros e sucessos que os modelos testados obtiveram, as matrizes de confusão atuam através da acurácia calculada, onde atingindo os coordenadores que geraram na matriz de confusão os verdadeiros positivos (TP), os verdadeiros negativos (TN), os falsos positivos (PF) e os falsos negativos (FN). A imagem ilustra o que é uma Matriz de Confusão.
- A precisão da classificação é devida ao número de previsões corretas dividido pelo número total de previsões definidas como**: Accuracy = {TP+TN} / {TP+FP+TN+FN}**

Results

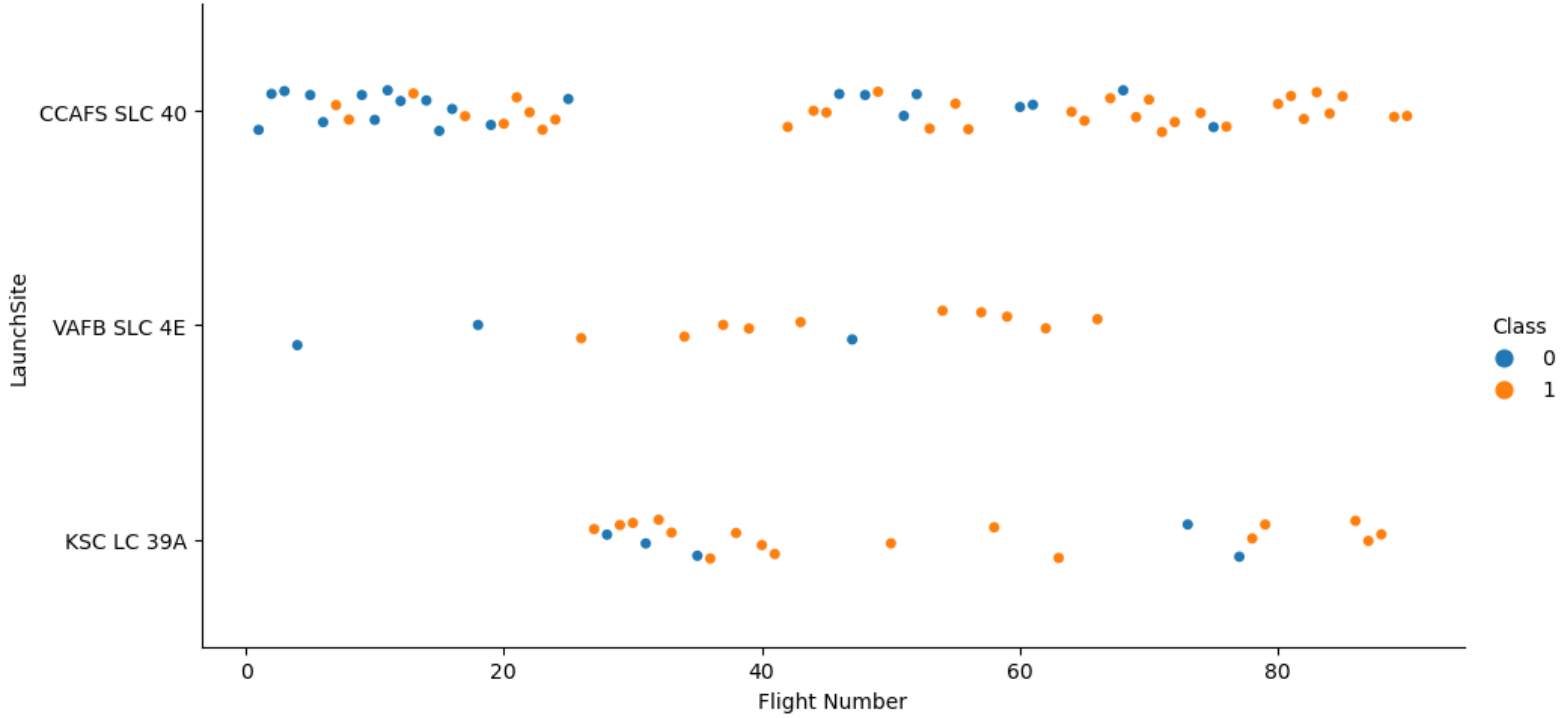
- Resultados da análise exploratória dos dados
- Demonstração de análise interativa em capturas de tela
- Resultados da análise preditiva

Section 2

Insights drawn from EDA

Flight Number vs. Launch Site

Cada local foi analisado em detalhes, para que os registros de pouso pudessem ser visualizados mais detalhadamente através de um gráfico.

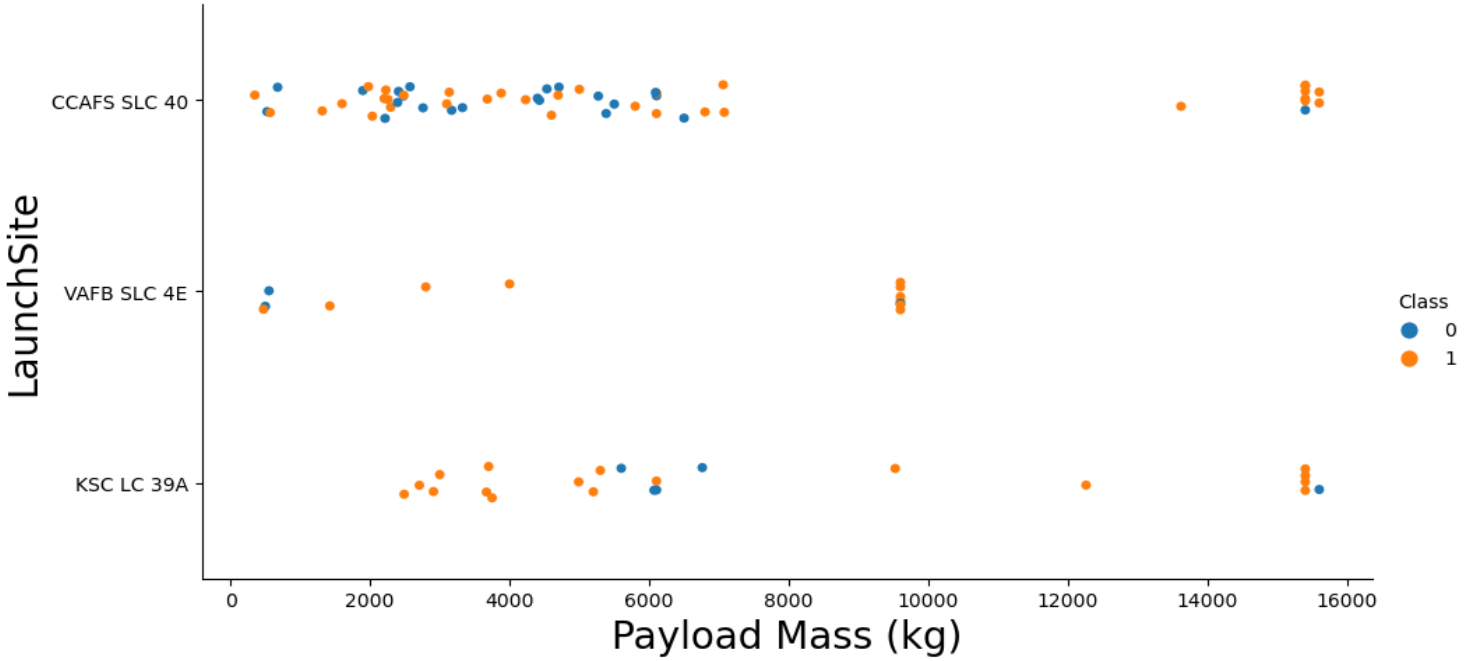


O gráfico acima demonstra a relação entre o número de lançamentos, variável "FlightNumber" e o local de lançamento, variável "Launch Site", em relação ao desempenho de pouso de retorno, variável "Class"

Podemos observar que, a partir da faixa de 80 em relação ao número de lançamentos, os pousos sempre começam a ser bem-sucedidos, no entanto, antes da faixa de 80, apenas alguns lugares podemos dizer que há mais casos de sucesso do que casos malsucedidos, podemos destacar a localização VAFB SLC 4E que a partir de 20, há desembarques mais bem-sucedidos, no entanto, a localização do CCAFS SLC 40 só começa a obter pousos bem-sucedidos a partir da pista 80 em diante.

Payload vs. Launch Site

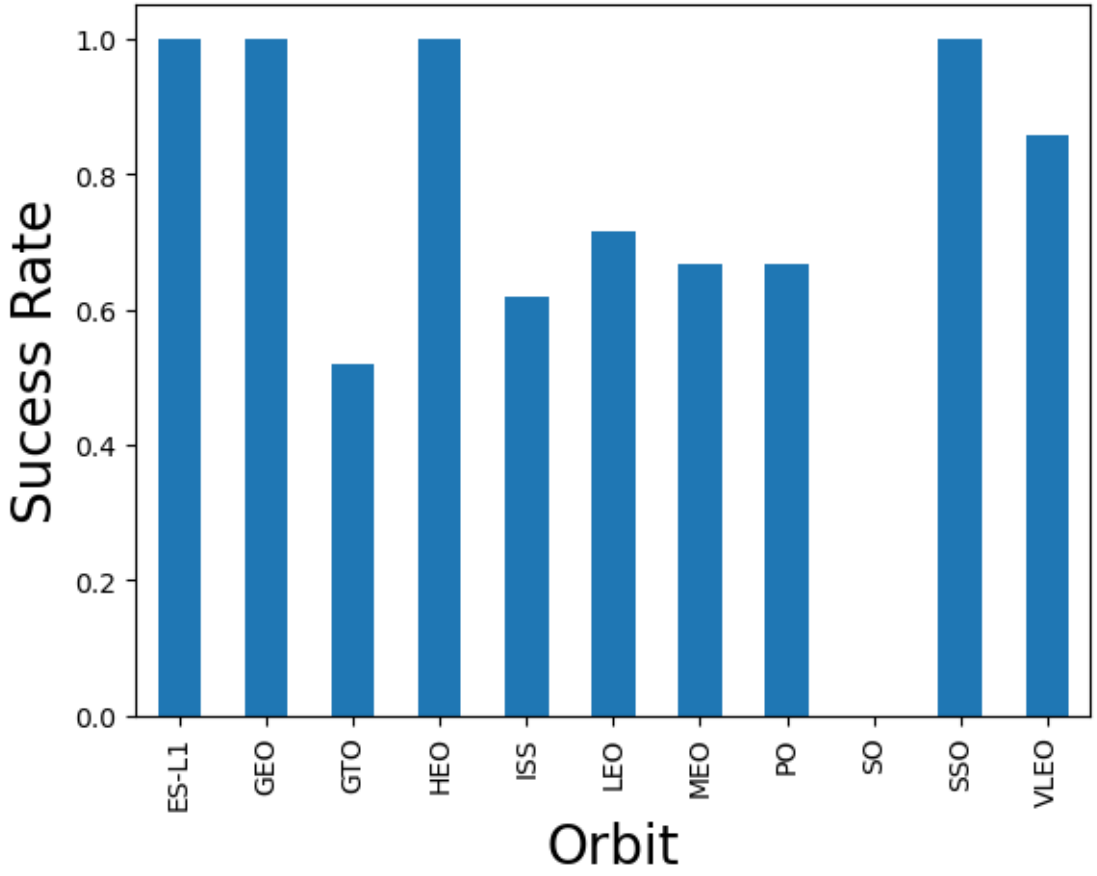
Também foi observado através do gráfico abaixo, se há alguma relação entre os locais de lançamento, "LaunchSite" e a massa da carga útil "PayloadMass"



Pode ser visto no gráfico LaunchSite vs PayloadMass que, para lançamentos localizados no local VAFB-SLC 4E, não há foguetes lançados para uma massa de carga útil superior a 10.000. Conclui-se também que este mesmo local tem, em quase todos os seus registros, os desembarques mais bem sucedidos entre cargas de 0 a perto de 10.000. O site KSC LC 39A não possui lançamentos com cargas úteis de 0 a 2.000, apenas acima, mas observe que, como todos os lançamentos realizados com cargas úteis acima de 2.000 e abaixo de 6.000, praticamente todos os pousos de retorno foram bem-sucedidos, e isso se repete entre cargas acima de 9.000 e abaixo de 16.000. O site CCAFS SLC 40 não possui lançamentos entre cargas acima de 8.000 e abaixo de 12.000, e entre as faixas de carga abaixo de 8.000 carga de massa, não é possível obter uma definição exata sobre a influência do peso com o local de lançamento, no entanto, acima de 12.000 massa de carga, este local obteve praticamente todos os lançamentos, o retorno foi bem sucedido

Success Rate vs. Orbit Type

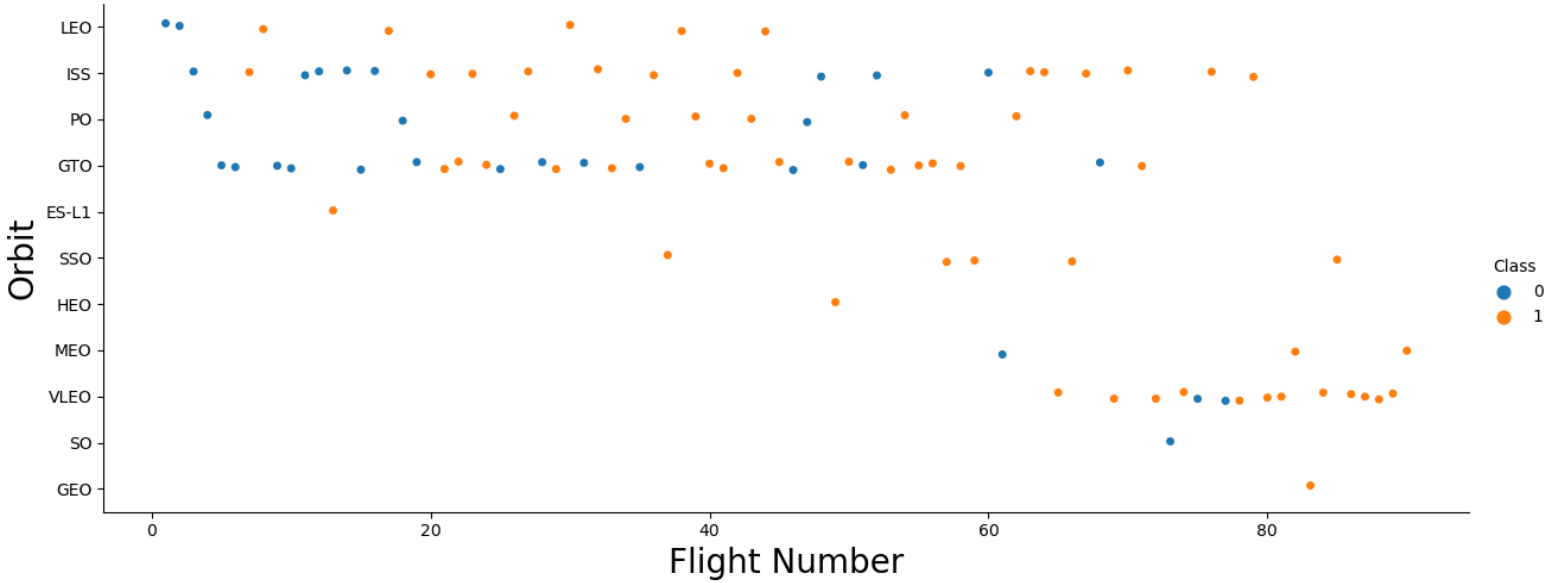
Em seguida, analisou-se visualmente se há alguma relação entre a taxa de sucesso e o tipo de órbita, como mostra o gráfico. “Success Rate” vs “Orbit”



Na análise gráfica, nota-se que as órbitas com maior taxa de sucesso estão entre as órbitas. ``ES-L1, GEO, HEO, SSO``

Orbit Type vs. Flight Number

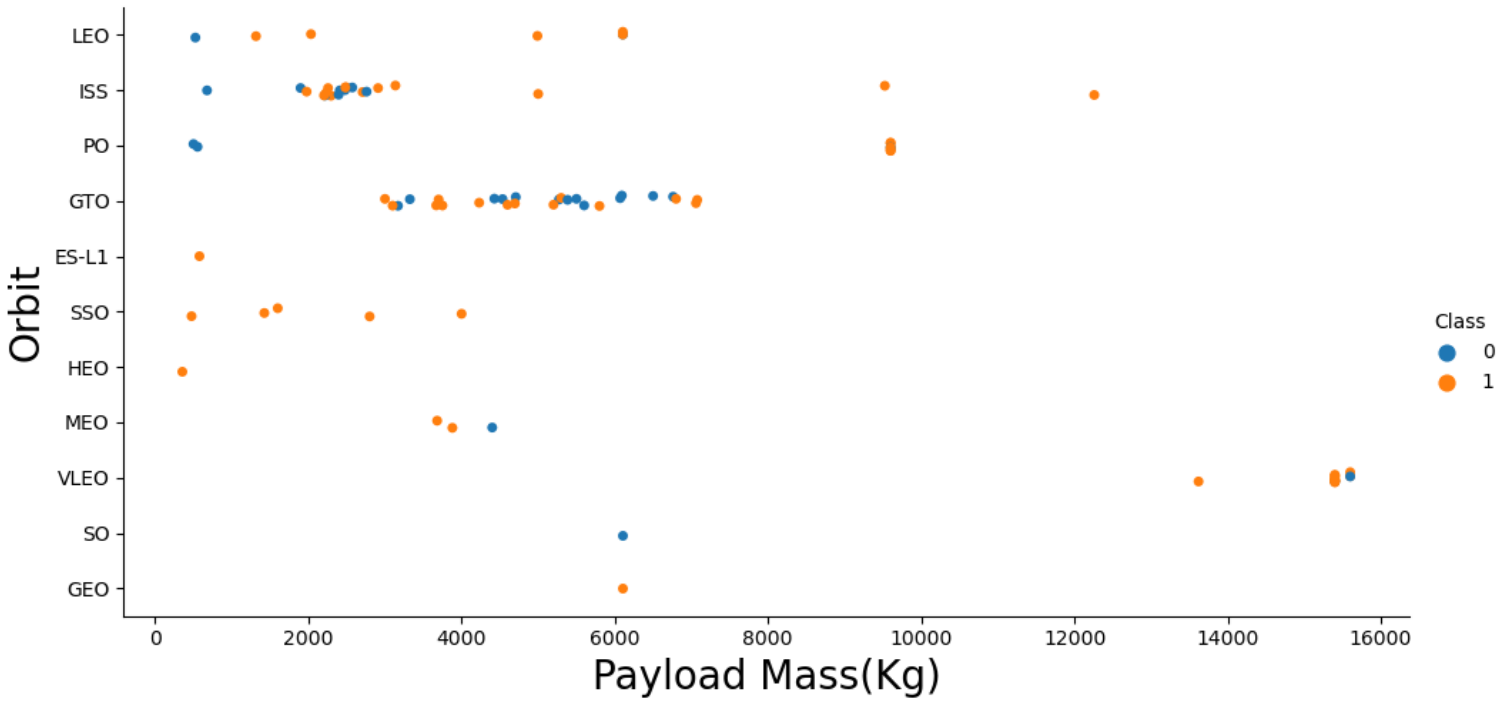
Para continuar analisando a variável "Órbita", a relação entre as órbitas e o número de lançamentos foi observada, o “FlightNumber” variável.



A partir da Órbita LEO pode-se observar que, acima do número de voos 10 e abaixo de 40, todos os pousos de retorno foram bem-sucedidos, mas abaixo dele foi o oposto. A Orbit ISS só conseguiu retornar pousos acima de 20 e abaixo de 45 números de voo, e também acima de 60. O Orbit PO, acima de 20 números de voo, tem uma taxa de sucesso maior em relação a pousos de retorno malsucedidos. Na órbita GTO não é possível obter uma leitura precisa, pois os dados não mostram homogeneidade na classe resultante. ES Orbit – L1 Há um único lançamento abaixo do voo número 20, e este com uma taxa de pouso de retorno bem-sucedida. Todos os lançamentos da Orbit SSO foram bem sucedidos no pouso de retorno, isso sugere uma forte relação entre o número de voos e a Orbit em questão. As órbitas MEO, VLEO, SO e GEO não têm um registro de pouso de retorno abaixo do número de voos 60, apenas acima, e pode-se ver que há uma relação entre as variáveis “Flight Number” vs “Orbit”.

Orbit Type vs. Payload

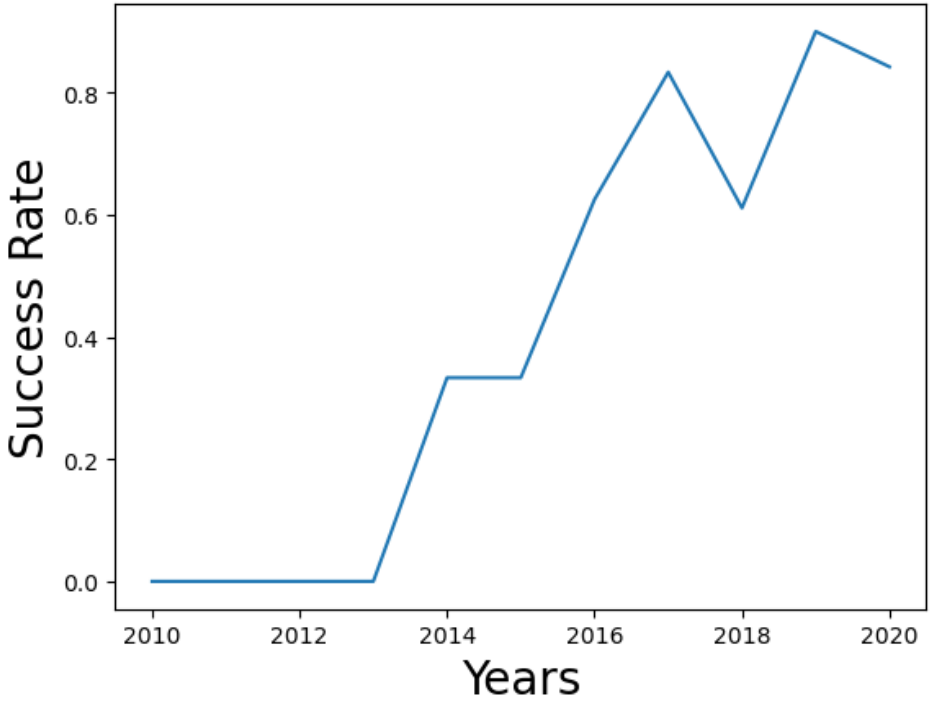
E, da mesma forma, analisou se havia relação entre as variáveis. "Orbit" vs "PayloadMass":



Com cargas pesadas acima de 4000, a taxa de pouso bem-sucedida ou positiva é maior para PO, LEO e ISS. No entanto, para o GTO, não podemos distinguir isso bem, pois tanto a taxa de pouso positiva quanto o pouso negativo, missões malsucedidas, estão em equivalência em seus resultados de pouso de retorno positivo e negativo, o que torna qualquer declaração sobre isso inexistente. Mas podemos concluir depois de ver o gráfico "Orbit" vs "PayloadMass" que existe uma forte relação entre essas variáveis em relação ao sucesso ou fracasso do pouso de retorno, o que torna essas variáveis também relevantes para o estudo que lidera.

Launch Success Yearly Trend

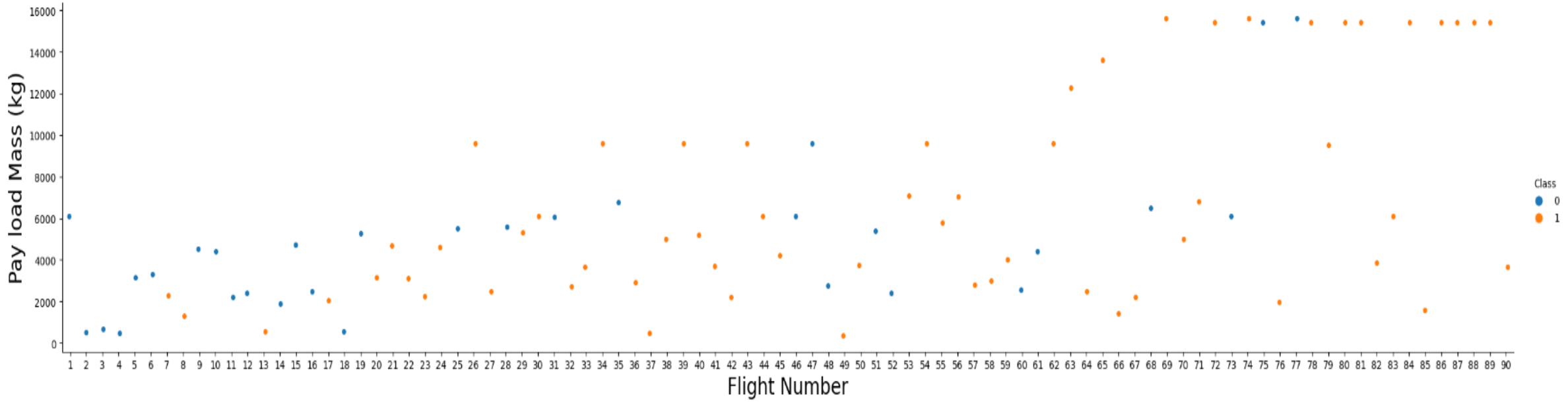
Os anos também foram analisados em relação ao sucesso médio de pouso de retorno, em comparação com os lançamentos, a fim de visualizar a tendência de sucesso da missão ao longo dos anos.



Olhando para a linha no gráfico, a taxa de sucesso desde cerca de 2013 continuou a aumentar até 2020, mudando completamente o cenário anteriormente estagnado.

PayloadMass vs. Flight Number

Primeiro analisamos a relação entre as variáveis “FlightNumber” variável que indica o número de tentativas, umd “PayloadMass” variáveis que estão relacionadas com a massa da carga útil. Observe como essas variáveis afetam o resultado do lançamento. A relação entre as variáveis “FlightNumber” vs. “PayloadMass”, e essas variáveis sobrepostas, a taxa de sucesso ou fracasso referida pela variável l "Classe", onde 0 zero está ligado a um pouso malsucedido e 1 um a um pouso bem-sucedido.



Vemos que, à medida que o número de voos aumenta, é mais provável que o primeiro estágio pouse com sucesso, mas a massa da carga útil também é importante, pois parece que quanto mais maciça a carga útil, menor a probabilidade de que o primeiro estágio retorne. Conclui-se também que diferentes locais de lançamento têm diferentes taxas de sucesso. CCAFS LC-40, tem uma taxa de sucesso de 60%, enquanto KSC LC-39A e VAFB SLC 4E têm uma taxa de sucesso de 77%.

SQL QUERIES

All Launch Site Names

```
%sql select DISTINCT(LAUNCH_SITE) from spacex;
```

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

- Podemos obter os valores exclusivos usando " DISTINCT "

Launch Site Names Begin with 'CCA'

```
%sql SELECT * from SPACEXTBL where LAUNCH_SITE LIKE 'CCA%' LIMIT 5;
```

0	2010-06-04	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success	Failure (parachute)
1	2010-12-08	15:43:00	F9 v1.0 B0004	CCAFS LC-40	Dragon demo flight C1, two CubeSats, barrel of...	0	LEO (ISS)	NASA (COTS) NRO	Success	Failure (parachute)
2	2012-05-22	07:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success	No attempt
3	2012-10-08	00:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500	LEO (ISS)	NASA (CRS)	Success	No attempt
4	2013-03-01	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677	LEO (ISS)	NASA (CRS)	Success	No attempt

- Podemos obter apenas 5 linhas usando " LIMIT "
- Podemos filtrar por caractere inserido na tabela " LIKE 'characters of interest%' "

Total Payload Mass

```
%sql select sum(PAYLOAD_MASS__KG_) from SPACEXTBL WHERE Customer="NASA (CRS)";
```

0	45596

- **Usando a função " SUM " soma o total na coluna** PAYLOAD_MASS_KG_
- **O " WHERE " A cláusula filtra o conjunto de dados para executar cálculos somente em Customer** NASA (CRS)

Average Payload Mass by F9 v1.1

```
%sql select avg(PAYLOAD_MASS__KG_) from SPACEXTBL WHERE booster_version LIKE "F9 v1.1%";
```

0	2928

- **Usando a função** “ AVG “calcula a média na coluna PAYLOAD_MASS_KG_
- **O** “ WHERE “A cláusula filtra o conjunto de dados para executar apenas cálculos em Booster_version F9 v1.1

First Successful Ground Landing Date

```
%sql select min(DATE) from SPACEXTBL where LANDING _OUTCOME = Success (ground pad);
```

0	2015-12-22

- We can get the first successful data using " MIN " because the first date is equal to the minimum date

Successful Drone Ship Landing with Payload between 4000 and 6000

```
%sql select BOOSTER_VERSION from SPACEXTBL where \
LANDING__OUTCOME='Success (drone ship)' and PAYLOAD_MASS__KG_ BETWEEN 4000 and 6000;
```

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

- The payload mass data was taken " between 4000 and 6000 " only, and the landing outcome was determined to be success drone ship

Total Number of Successful and Failure Mission Outcomes

```
%sql select count(Mission_Outcome) from SPACEXTBL where Mission_Outcome like "Success%";
```

0	100

- We can get number of all the successful mission by using " COUNT " and " LIKE Successful% "

```
%sql select count(Mission_Outcome) from SPACEXTBL where Mission_Outcome like "Failure%";
```

0	1

- We can get number of all the successful mission by using " COUNT " and " LIKE Failure% "

Boosters Carried Maximum Payload

```
%sql select distinct(Booster_Version) from spacex \
where PAYLOAD_MASS__KG_ = (select max(PAYLOAD_MASS__KG_) from spacex);
```

0	F9 B5 B1048.4
1	F9 B5 B1049.4
2	F9 B5 B1051.3
3	F9 B5 B1056.4
4	F9 B5 B1048.5
5	F9 B5 B1051.4
6	F9 B5 B1049.5
7	F9 B5 B1060.2
8	F9 B5 B1058.3
9	F9 B5 B1051.6
10	F9 B5 B1060.3
11	F9 B5 B1049.7

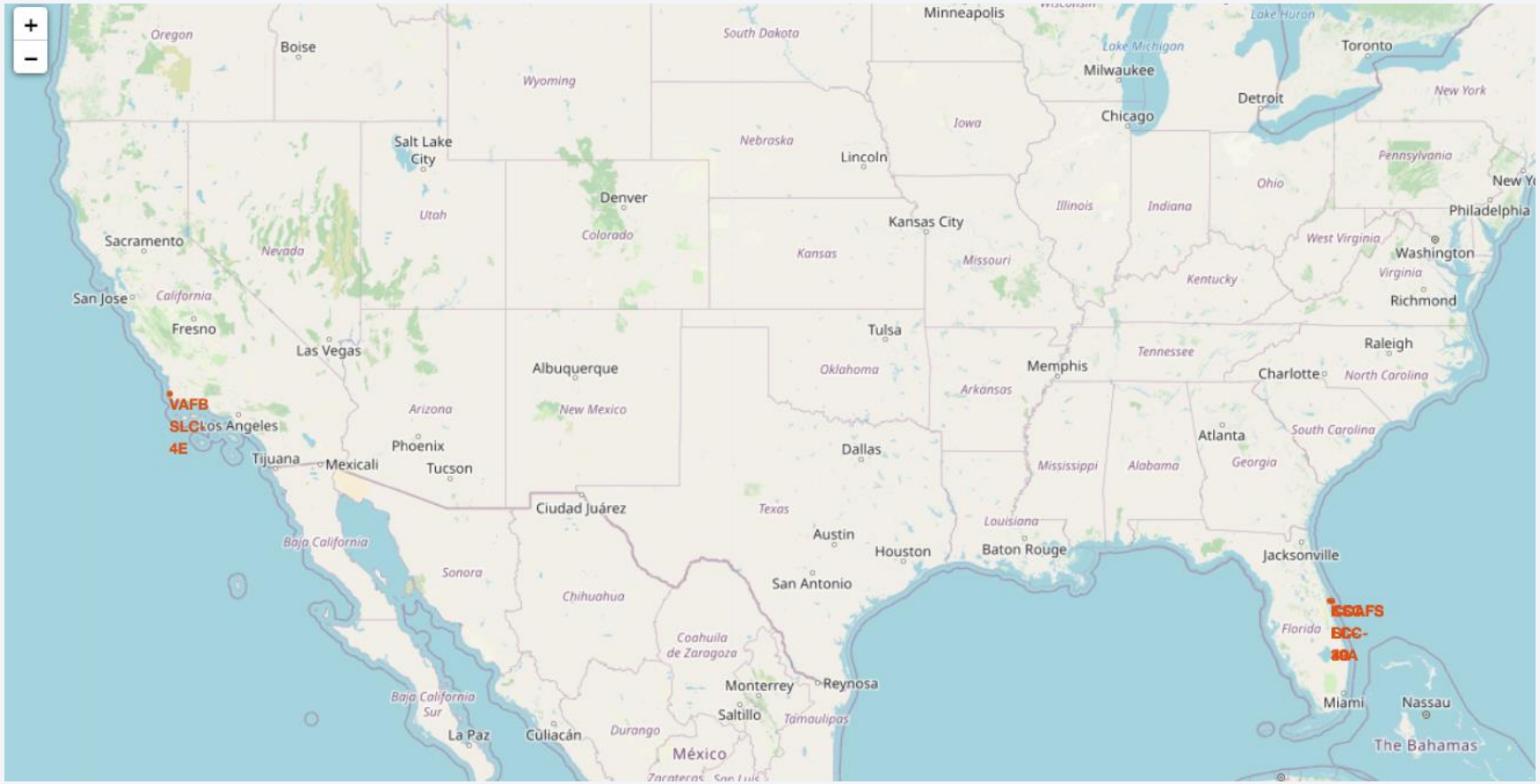
- USING A WORD " DISTINCT " IN THE QUERY MEANS THAT ONLY SHOW UNIQUE VALUES IN Booster_Version COLUMN FROM SPACEX TABLE
- USING A subquery TO OBTAIN THE maximum VALUE OF THE COLUMN PAYLOAD_MASS__KG_

Section 3

Launch Sites Proximities Analysis

Geographic Location

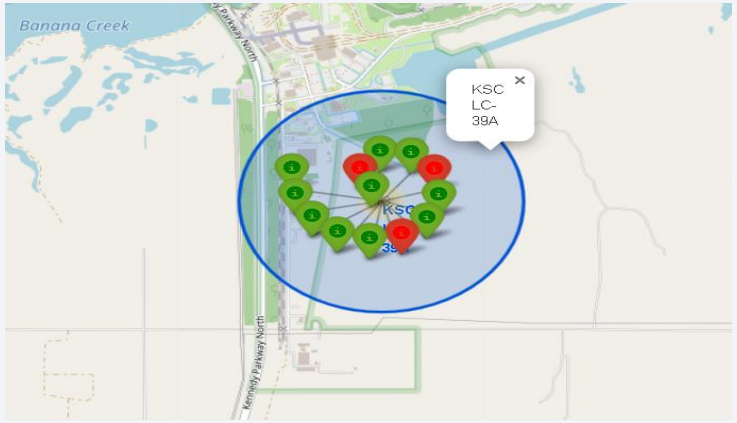
Geographic Location Launch Location SpaceX

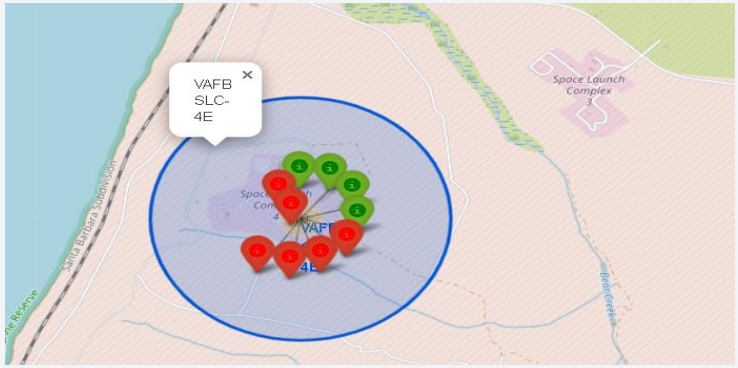


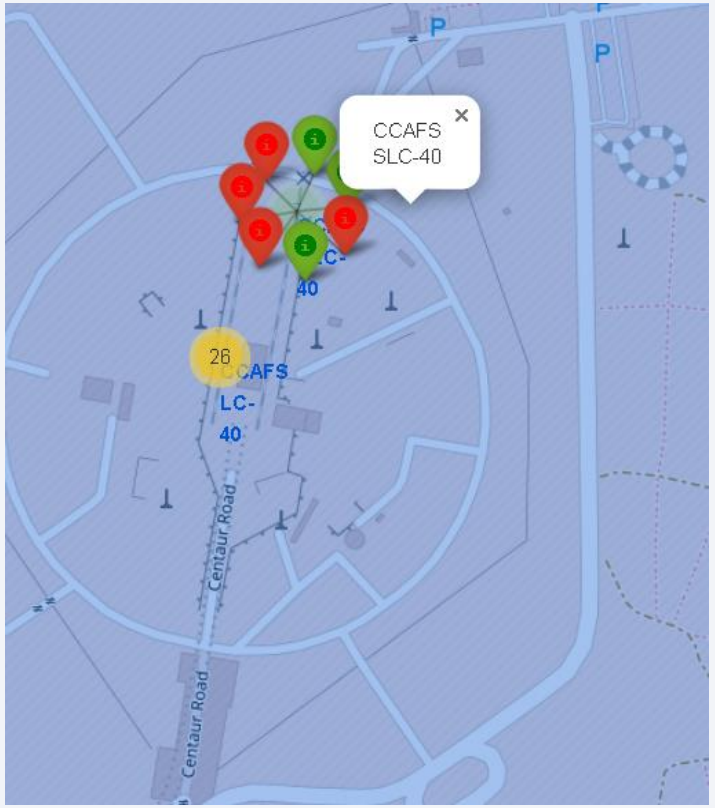
As launch sites have a strong influence on mission success, we analyze the geographic opposition of these points and understand their location. The map illustrates the positioning of the launch sites CCAFS LC-40, CCAFS SLC-40, KSC LC-39A, VAFB SLC-4E. All launch sites in USA, Florida and California

Success Rate by Launch Location

Success Rate Visualization





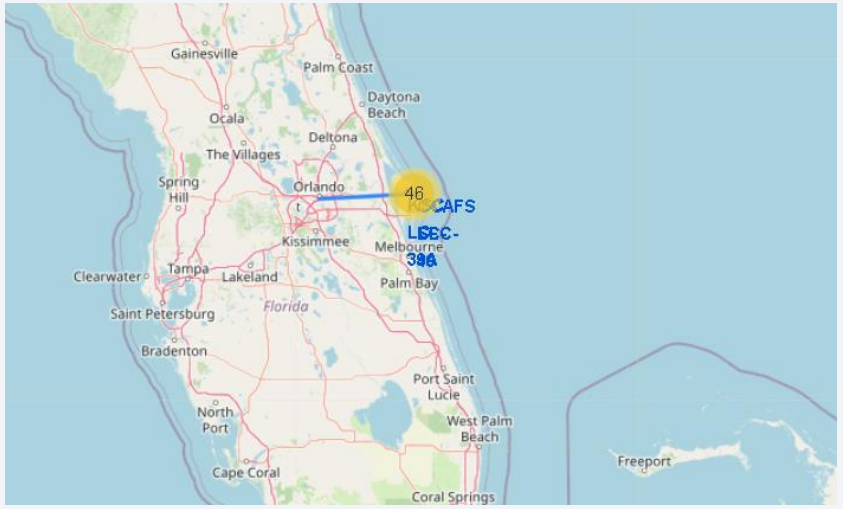


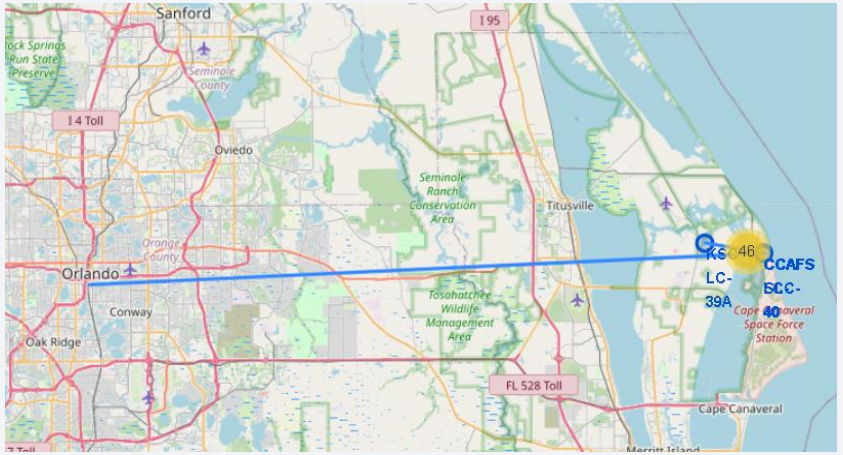
- Map the success rate results for each launch site to see which sites illustrate the highest success rates
- Marks inserted from the map are divided into two classes, Green for Success and Red for Failure.

Launch Site Details

Relationship between the distances of launch sites

- All launch sites are not close to the equator, they are approximately 30°N from the equator, and all launch sites are very close to shore. Note that postings only occur at one of the four posting locations, which means that many posting records will have exactly the same coordination.





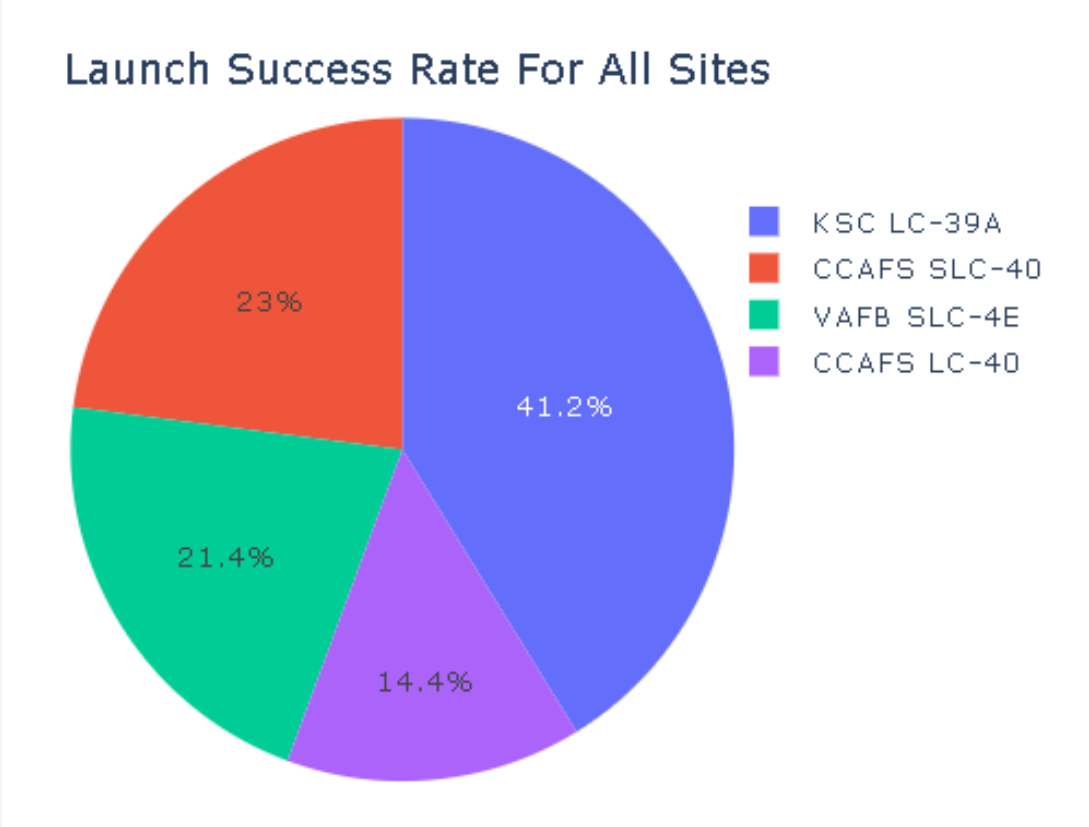
Section 4

Build a Dashboard with Plotly Dash

Launch Records Dashboard

Success Rate Among All

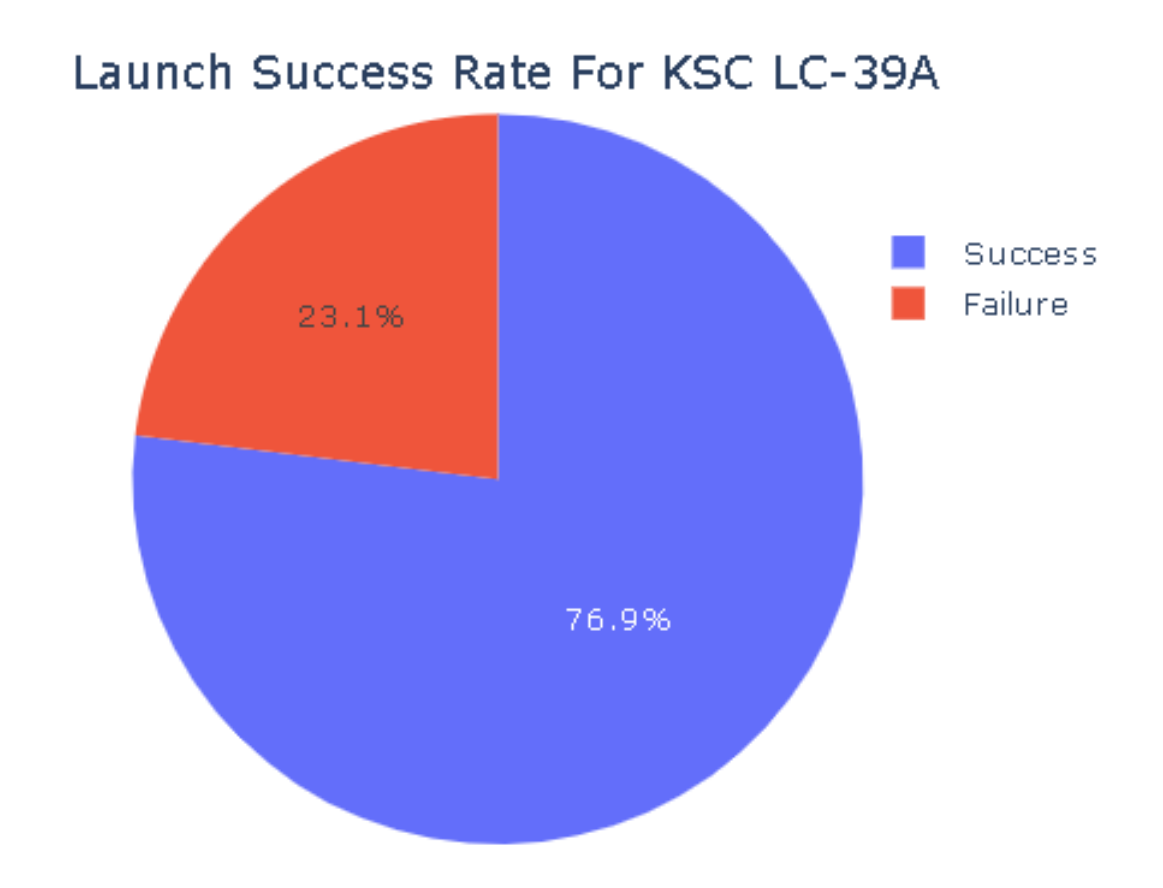
- We can clearly and objectively observe that, among all launch sites, KSC LC - 39A is the one with the highest success rate.



Launch Records Dashboard

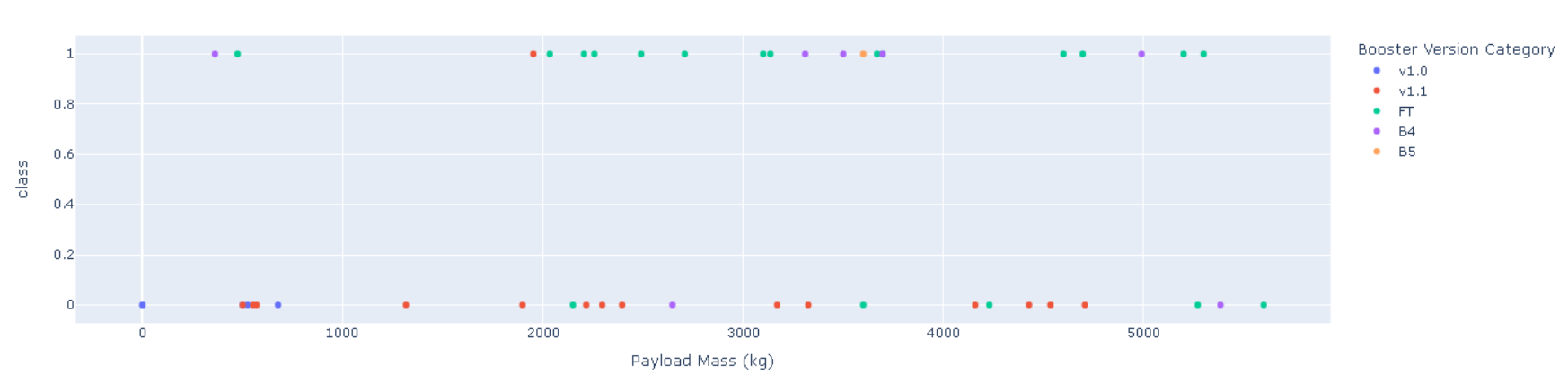
Success and Failure Rate

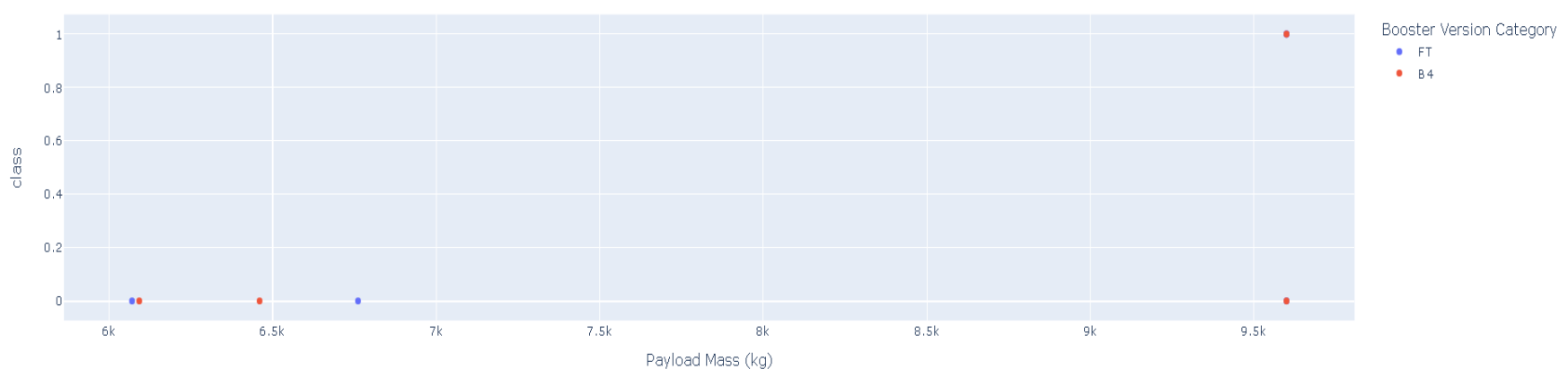
- If we take the site with the highest success rate "KSC LC - 39A" and look at it more closely, we can conclude that 23% failed and 77% succeeded.



Payload Concentration

Class vs. PayloadMass(kg)





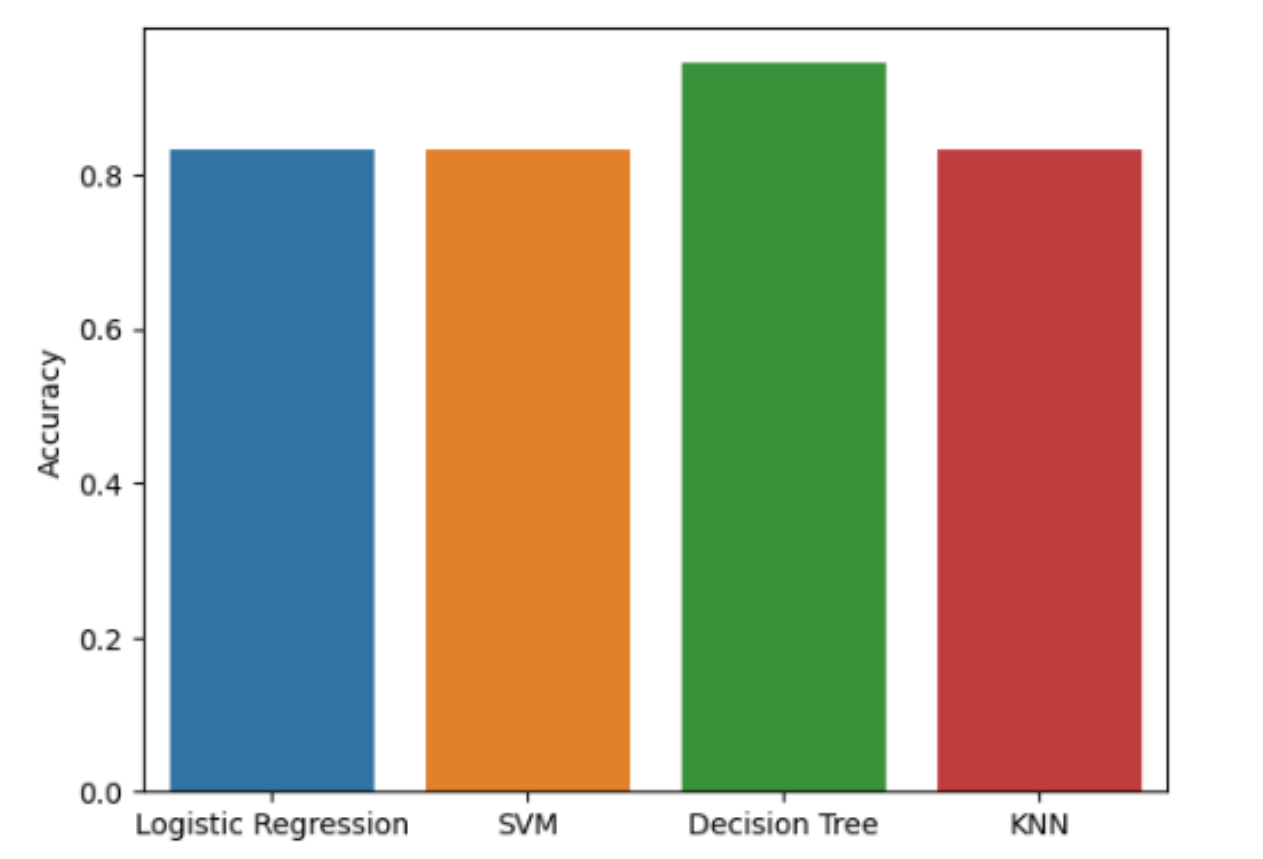
- The highest payloads are in the range between 0 and 6000
- The second largest payload range is between 6,000 and 10,000
- We can see the success rates for low weighted payloads is higher than the heavy weighted payloads

Section 5

Predictive Analysis (Classification)

Classification Accuracy

Classification Accuracy is due to the number of correct predictions divided by the total number of predictions defined as: Accuracy = {TP+TN} / {TP+FP+TN+FN}



After dividing the data into training and testing, apply the GridSearch function, test all models, extracting the best hyperparameters, the model which performed best among all was the Decision Tree Classification model, which reached a coefficient of Accuracy of 0.944444

Confusion Matrix

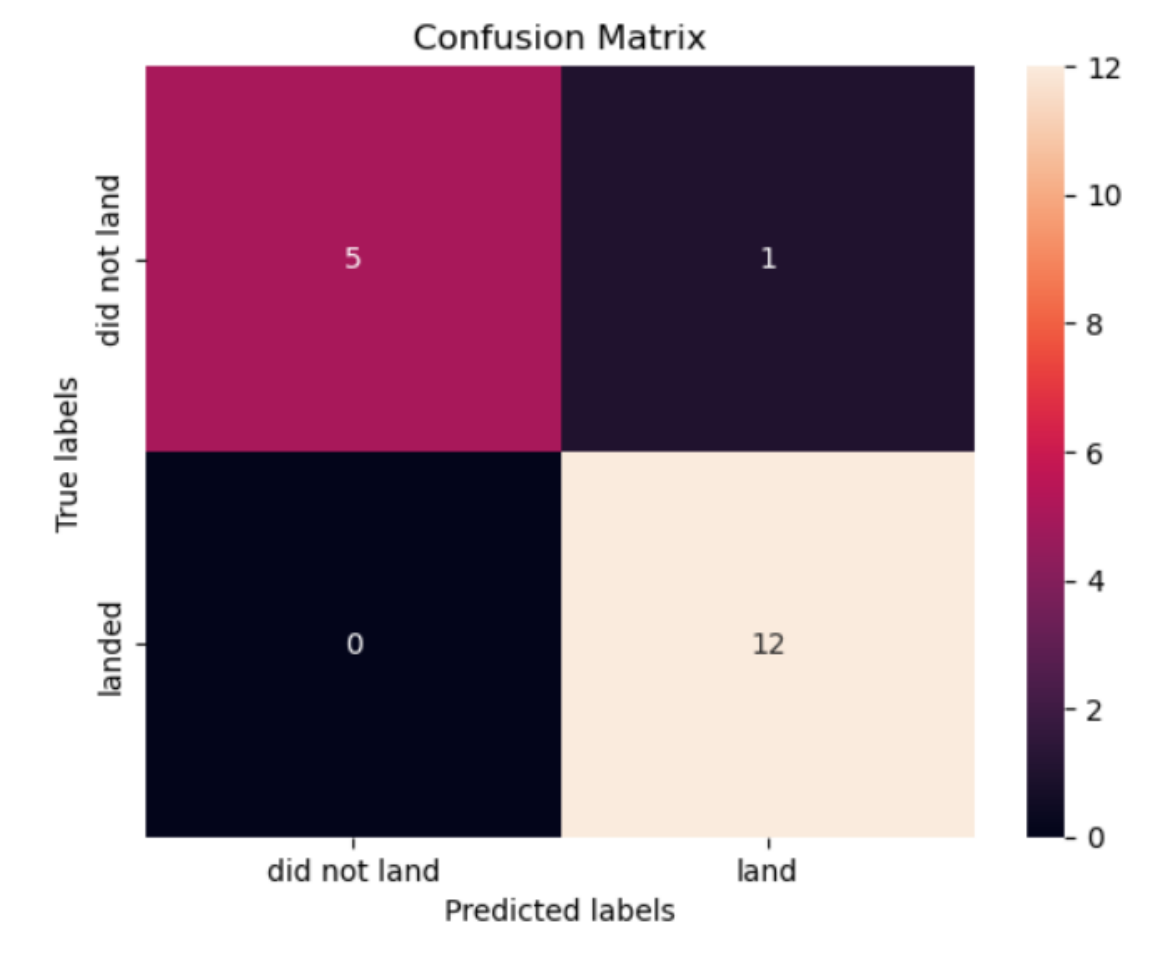
Accuracy = {TP+TN} / {TP+FP+TN+FN}

In order for us to visualize how well the models are performing, a confusion matrix was extracted, to determine the real value of errors and successes that the tested models obtained, the confusion matrices act by calculating the accuracy, reaching a coefficient , which will be generated over the confusion matrix

		Predicted Class: Positive	Predicted Class: Negative	
Actual Class	Positive	True Positive (TP)	False Negative (FN) Type II Error	**Sensitivity** $\frac{TP}{(TP+FN)}$
	Negative	False Positive (FP) Type I Error	True Negative (TN)	**Specificity** $\frac{TN}{(TN+FP)}$
		Precision $\frac{TP}{(TP+FP)}$	**Negative Predictive Value** $\frac{TN}{(TN+FN)}$	**Accuracy** $\frac{TP+TN}{(TP+TN+FP+FN)}$

Confusion Matrix

Confirming through an analysis made on a confusion matrix, generated for each model, confirms the decision tree when the best result is obtained by classifying only 1 as false negative (FN), 5 as true positives (TP) and 12 as true negatives (TN).



Conclusions

Arriving at a final result after all the analyzes and tests carried out, we can state:

- The **Decision Tree Classifier** Algorithm performed best among all
- Low weight **payloads** work better than heavier payloads
- **Success rates** of SpaceX launches, directly **to time in years**, show that eventually there may be improvements in launches over the years if they continue with the process of continuous evolution
- It can also be said that orbit **GEO, HEO, SSO, ES L1** has the best success
- The Site with the best success rate among all was **KSC LC 39A** had the highest success rate of all launch sites

Appendix

Here I leave my thanks to Coursera and IBM, for providing this course which taught me a lot, and is helping me a lot in my journey of professional transformation, thank you all

Below is the access link to the entire data source used in the realization of this project

We work with SpaceX launch data collected from an API, specifically the SpaceX REST API (https://api.spacexdata.com/v4/launches/past). This API will give us data about launches, including information about the rocket used, payload delivered, launch specifications, landing specifications and landing result..

The Scraping web method was also used to collect data from historical Falcon 9 launch records, from a Wikipedia page entitled "List of Falcon 9 and Falcon Heavy launches" which were scraped. (https://en.wikipedia.org/wiki/List_of_Falcon/_9/_and_Falcon_Heavy_launches). All data were tabulated and the respective tables transformed into a DataFrame, so that we have better flexibility in handling and analyzing data

Github:All Notebooks SpaceX Project

Thank you!