

Winning Space Race with Data Science

Antonio Fernández Salcedo
Novembre 2024





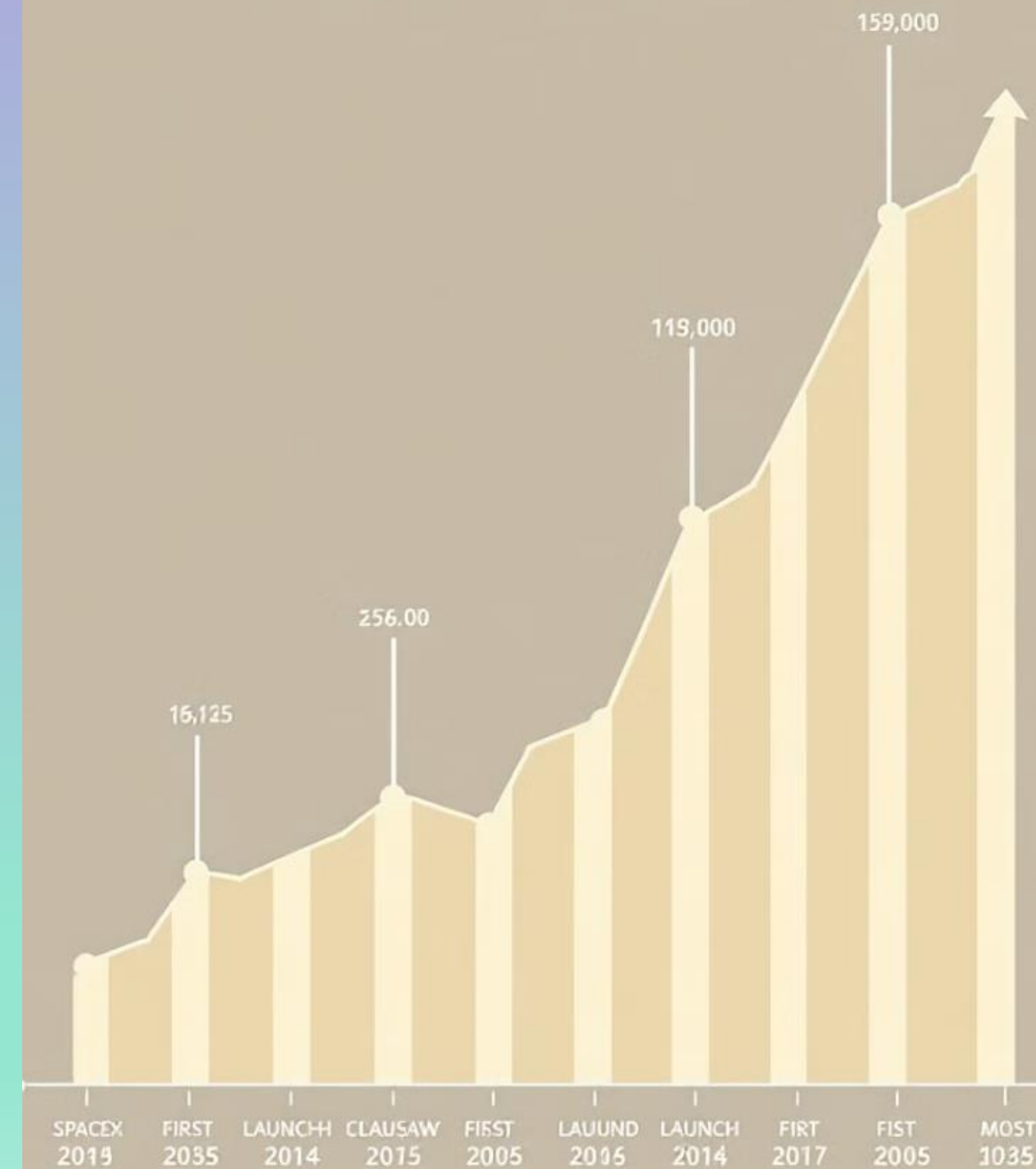
Antonio Fernández Salcedo

Noviembre 2024

Contenido

- Resumen ejecutivo
- Introducción
- Sección 1. Metodología
- Sección 2. Perspectivas extraídas de EDA. (*Insights drawn from EDA*)
- Sección 3. Análisis de proximidad de sitios de lanzamiento
- Sección 4. Construir un tablero con *Plotly-Dash*.
- Sección 5. Análisis Predictivo (Clasificación)
- Conclusiones
- Recomendaciones y consideraciones finales
- Apéndice del proyecto
- Fuentes de datos utilizadas

SPaCEX rocket reuse rate



Resumen ejecutivo

Este proyecto de ciencia de datos tiene como objetivo explorar si SpaceX reutilizará la primera etapa de sus cohetes. Utilizaremos datos de la API pública de SpaceX y la página de Wikipedia para realizar un análisis exhaustivo que nos ayude a entender mejor este proceso.



Resumen ejecutivo

Metodología:

- Se recopilaron datos utilizando la API pública de SpaceX y web scraping de la página de Wikipedia de SpaceX.
- Se realizó la limpieza y transformación de datos, incluyendo la creación de una etiqueta de clase para clasificar los aterrizajes exitosos.
- Se llevó a cabo un análisis exploratorio de datos (EDA) utilizando visualizaciones y SQL.
- Se realizaron análisis visuales interactivos utilizando Folium y Plotly Dash.
- Se realizó un análisis predictivo utilizando cuatro modelos de clasificación de aprendizaje automático: regresión logística, máquinas de vectores de soporte, vecinos de soporte, vecinos más cercanos K y árbol de decisión.

Resultados:

- Los cuatro modelos de aprendizaje automático lograron una precisión de aproximadamente el 83,33%.
- Todos los modelos sobrestimaron los aterrizajes exitosos, lo que sugiere la necesidad de más datos para mejorar la precisión.
- El análisis exploratorio de datos reveló una correlación positiva entre el número de vuelos y la tasa de éxito.
- Se observaron tasas de éxito más altas en órbitas geosíncronas como SSO, HEO, GEO y ES-L1.
- Los sitios de lanzamiento están estratégicamente ubicados cerca de la costa, lejos de las ciudades y cerca de carreteras y ferrocarriles.

Resumen ejecutivo

Conclusiones:

- SpaceX puede utilizar este modelo para predecir con una precisión relativamente alta si un lanzamiento tendrá un aterrizaje exitoso de la Etapa 1 antes del lanzamiento.
- Se recomienda recopilar más datos para mejorar la precisión del modelo.
- El análisis sugiere que la reutilización de la primera etapa de los cohetes SpaceX es factible y podría reducir significativamente los costos de lanzamiento.

Consideraciones Adicionales:

- Se deben considerar factores como la masa de la carga útil y la selección del sitio de lanzamiento para optimizar las posibilidades de éxito.
- Se necesitan investigaciones adicionales para explorar tecnologías innovadoras como cohetes que no utilizan combustible tradicional.

Recomendaciones:

- Continuar recopilando datos para mejorar la precisión del modelo predictivo.
- Explorar el desarrollo de tecnologías de cohetes más sostenibles.
- Enfocarse en lanzamientos a órbitas cercanas a la Tierra con cargas útiles ligeras para ganar experiencia.



Introducción a SpaceX y sus cohetes reutilizables

1

Misión de SpaceX

Reducir los costos de acceso al espacio mediante el desarrollo de cohetes reutilizables.

2

Tecnología de Aterrizaje Vertical

SpaceX utiliza sistemas avanzados que permiten a los primeros estadios aterrizar verticalmente para su reutilización.

3

Ahorro Significativo

La reutilización de cohetes reduce drásticamente los costos por lanzamiento en comparación con los modelos desechables.

Introducción

El proyecto se centra en la industria espacial comercial, que está experimentando un crecimiento acelerado con empresas como SpaceX que como SpaceX que hacen que los viajes espaciales sean más accesibles. SpaceX ha logrado una ventaja competitiva al fabricar cohetes cohetes reutilizables, específicamente la primera etapa del cohete Falcon 9, lo que les permite ofrecer lanzamientos a un precio precio significativamente menor en comparación con sus competidores.

El objetivo principal del proyecto es desarrollar un modelo de aprendizaje automático capaz de predecir la recuperación exitosa de la primera exitosa de la primera etapa del cohete Falcon 9. Este modelo sería una herramienta valiosa para SpaceX, ya que les permitiría tomar decisiones tomar decisiones informadas sobre la viabilidad de un lanzamiento y la reutilización de la primera etapa.

El proyecto se basa en la hipótesis de que ciertos factores, como el sitio de lanzamiento, la masa de la carga útil y el tipo de órbita, influyen de órbita, influyen en la probabilidad de un aterrizaje exitoso. Para probar esta hipótesis, se recopilaron datos de la API pública de SpaceX y la pública de SpaceX y la página de Wikipedia de SpaceX.

Introducción

Los datos se procesaron y analizaron utilizando diversas técnicas, incluyendo:

- **Limpieza y transformación de datos:** Creación de una etiqueta de clase para clasificar los aterrizajes exitosos y conversión de variables de variables categóricas a binarias.
- **Análisis exploratorio de datos (EDA):** Visualización de datos para identificar patrones y tendencias utilizando bibliotecas como Matplotlib y como Matplotlib y Seaborn, y análisis de datos con SQL.
- **Análisis visual interactivo:** Creación de mapas interactivos con Folium para visualizar la ubicación de los sitios de lanzamiento y los lanzamiento y los resultados de los aterrizajes, y desarrollo de paneles de control con Plotly Dash para explorar la relación entre diferentes entre diferentes variables.
- **Análisis predictivo:** Entrenamiento y evaluación de cuatro modelos de clasificación de aprendizaje automático: regresión logística, máquinas de vectores de soporte, vecinos más cercanos K y árbol de decisión.

Introducción

El proyecto busca responder a las siguientes preguntas:

- ¿Qué factores influyen en la probabilidad de una recuperación exitosa de la primera etapa del cohete Falcon 9?
- ¿Se puede desarrollar un modelo de aprendizaje automático para predecir con precisión el éxito de un aterrizaje?
- ¿Cómo se pueden optimizar las operaciones de lanzamiento de SpaceX para maximizar la tasa de éxito de los aterrizajes y la reutilización de cohetes?

El proyecto también busca abordar los siguientes desafíos:

- La necesidad de datos adicionales para mejorar la precisión del modelo predictivo.
- La identificación de las mejores prácticas para la selección del sitio de lanzamiento, la masa de la carga útil y el tipo de órbita para aumentar la probabilidad de un aterrizaje exitoso.
- La exploración de tecnologías innovadoras, como cohetes que no utilizan combustible tradicional, para hacer que los viajes espaciales sean viajes espaciales sean más sostenibles.

Seccion 1. Metodología

Section 1

Methodology



Análisis exploratorio de los datos

Recopilación

Importar y limpiar los datos de lanzamientos de cohetes de SpaceX.

1

2

Análisis Inicial

Explorar tendencias y patrones en los datos para entender mejor el proceso de reutilización.

Visualización

Crear gráficos y representaciones visuales que ayuden a comunicar los hallazgos clave.

3

Recopilación de datos – API de SpaceX

- Dirección URL de GitHub del cuaderno de llamadas a la API de SpaceX completado: [jupyter-labs-spacex-data-collection-api.ipynb](#)

Diagrama de flujo que describe el proceso de recopilación de datos, con solicitudes API:

1. Obtener y analizar los datos de lanzamiento de SpaceX.
2. Normalizar la respuesta JSON en un marco de datos.
3. Extraer solo columnas útiles usando funciones auxiliares.
4. Crear un nuevo marco de datos de Pandas a partir del diccionario.
5. Filtrar el marco de datos para incluir solo lanzamientos de Falcon 9.
6. Gestionar los valores faltantes.
7. Exportar a un archivo CSV.

	FlightNumber	Date	BoosterVersion	PayloadMass	Orbit
0	1	2010-06-04	Falcon 9	6123.547647	LEO
1	2	2012-05-22	Falcon 9	525.000000	LEO
2	3	2013-03-01	Falcon 9	677.000000	ISS
3	4	2013-09-29	Falcon 9	500.000000	PO
4	5	2013-12-03	Falcon 9	3170.000000	GTO

Recopilación de datos – Web Scraping

- Dirección URL de GitHub del cuaderno de Web Scraping completado: [jupyter-labs-webscraping.ipynb](https://github.com/jupyter-labs/jupyter-labs-webscraping/blob/master/01%20Web%20Scraping.ipynb)

Diagrama de flujo que describe el proceso de recopilación de datos Web Scraping: Scraping:

1. Obtener datos de lanzamiento de cohetes de la página de Wikipedia.
2. Extraer todos los nombres de columna/variable de la tabla HTML, encabezados.
3. Crear un marco de datos analizando las tablas HTML de inicio.
4. Exportar a un archivo CSV.

	Flight No.	Launch site	Payload	Payload mass	Orbit
0	1	CCAFS	Dragon Spacecraft Qualification Unit	0	LEO
1	2	CCAFS	Dragon	0	LEO
2	3	CCAFS	Dragon	525 kg	LEO
3	4	CCAFS	SpaceX CRS-1	4,700 kg	LEO
4	5	CCAFS	SpaceX CRS-2	4,877 kg	LEO

Data Wrangling

En esta etapa, se analizaron cada uno de los atributos y algunos de ellos se transformaron. El objetivo de esta etapa era realizar un análisis realizar un análisis exploratorio de datos (EDA) para encontrar algunos patrones en los datos y principalmente convertir esos resultados en resultados en etiquetas de entrenamiento, donde 1 significa que el refuerzo aterrizó con éxito y 0 significa que no tuvo éxito.

Estos son los pasos que se tomaron para manipular los datos:

1. Identificar y calcular el porcentaje de valores faltantes en cada atributo.
 2. Calcular el número de lanzamientos en cada sitio.
 3. Calcular el número y la ocurrencia de cada órbita.
 4. Calcular el número y la ocurrencia del resultado de la misión por tipo de órbita.
 5. Crear una etiqueta de resultado de aterrizaje a partir de la columna Resultado y evaluar la tasa de éxito de los aterrizajes.
 6. Exportar a un archivo CSV.
- Dirección URL de GitHub del cuaderno de Wrangling de datos completado: [labs-jupyter-spacex-Data%20wrangling.ipynb](https://github.com/sergioferraz/labs-jupyter-spacex-Data%20wrangling.ipynb)

EDA con visualización de datos

Análisis Exploratorio de Datos (EDA)

El EDA se realizó utilizando herramientas de visualización como las bibliotecas Matplotlib y Seaborn de Python, así como SQL.

Se utilizaron los siguientes tipos de visualizaciones para el EDA:

- **Diagramas de dispersión:** para representar la relación entre dos variables diferentes y comparar conjuntos de características, como el número de vuelo frente al sitio de lanzamiento, la carga útil frente al sitio de lanzamiento, el número de vuelo frente al tipo de órbita y la carga útil frente al tipo de órbita.
- **Gráfico de barras:** facilitó la comparación de valores entre varios grupos a la vez, donde el eje x representaba una categoría y el eje y representaba un valor discreto. Los gráficos de barras se utilizaron para comparar la tasa de éxito para diferentes tipos de órbitas.
- **Gráfico de líneas:** fueron útiles para mostrar las tendencias de los datos a lo largo del tiempo. Se utilizó un gráfico de líneas para mostrar las tasas de éxito durante varios años.
- Dirección URL de GitHub del cuaderno de EDA completado con visualización: : [jupyter-labs-eda-dataviz-v2.ipynb](https://github.com/jupyter-labs/eda-dataviz-v2/blob/main/eda-dataviz-v2.ipynb)

Construya un mapa interactivo con Folium

Análisis Visual Interactivo con Folium

Objetos del mapa:

- Marcadores amarillos: Representan la ubicación de todos los sitios de lanzamiento de SpaceX en los Estados Unidos. Estos marcadores se agregaron para mostrar la distribución geográfica de los sitios de lanzamiento.
- Marcadores verdes y rojos: Indican aterrizajes exitosos (verde) y fallidos (rojo) en cada sitio de lanzamiento. Al hacer clic en los grupos de marcadores, se muestra información sobre el número de aterrizajes exitosos y fallidos en cada sitio.

Justificación:

- Los objetos del mapa en Folium se utilizaron para visualizar la ubicación de los sitios de lanzamiento de SpaceX y la tasa de éxito de los aterrizajes en cada sitio. La proximidad a la costa, las carreteras, las vías férreas y las ciudades se tuvieron en cuenta para comprender la ubicación estratégica de los sitios de lanzamiento.

URL de GitHub de su mapa interactivo completado con el mapa de Folium: [lab-jupyter-launch-site-location-v2.ipynb](https://github.com/yourusername/lab-jupyter-launch-site-location-v2.ipynb)

Construya un tablero con Plotly Dash

Análisis Visual Interactivo con Plotly Dash

- Gráfico circular: Muestra la tasa de éxito para cada sitio de lanzamiento. Este gráfico permite una comparación rápida del rendimiento de los diferentes sitios.
- Gráfico de dispersión: Muestra cómo el éxito varía en función de los sitios de lanzamiento, la masa de la carga útil y la categoría de la versión del propulsor.
 1. Selector de rango de carga útil: Permite al usuario filtrar los datos y centrarse en rangos específicos de masa de carga útil.
 2. Codificación de colores: La categoría de la versión del propulsor se representa con diferentes colores, lo que permite al usuario identificar patrones relacionados con versiones específicas.

Los gráficos e interacciones en Plotly Dash proporcionan una visión más detallada de los factores que influyen en el éxito del aterrizaje. El gráfico circular resume el rendimiento general del sitio, mientras que el gráfico de dispersión permite un análisis más profundo de la relación entre el éxito, la masa de la carga útil, la versión del propulsor y el sitio de lanzamiento. URL de GitHub del cuaderno de Plotly Dash completado:

[spacex_dash_app.ipynb](#)

Modelo predictivo para la reutilización de cohetes

1

Datos Históricos

Entrenar un modelo de machine learning con los registros de lanzamientos y reutilización de cohetes.

2

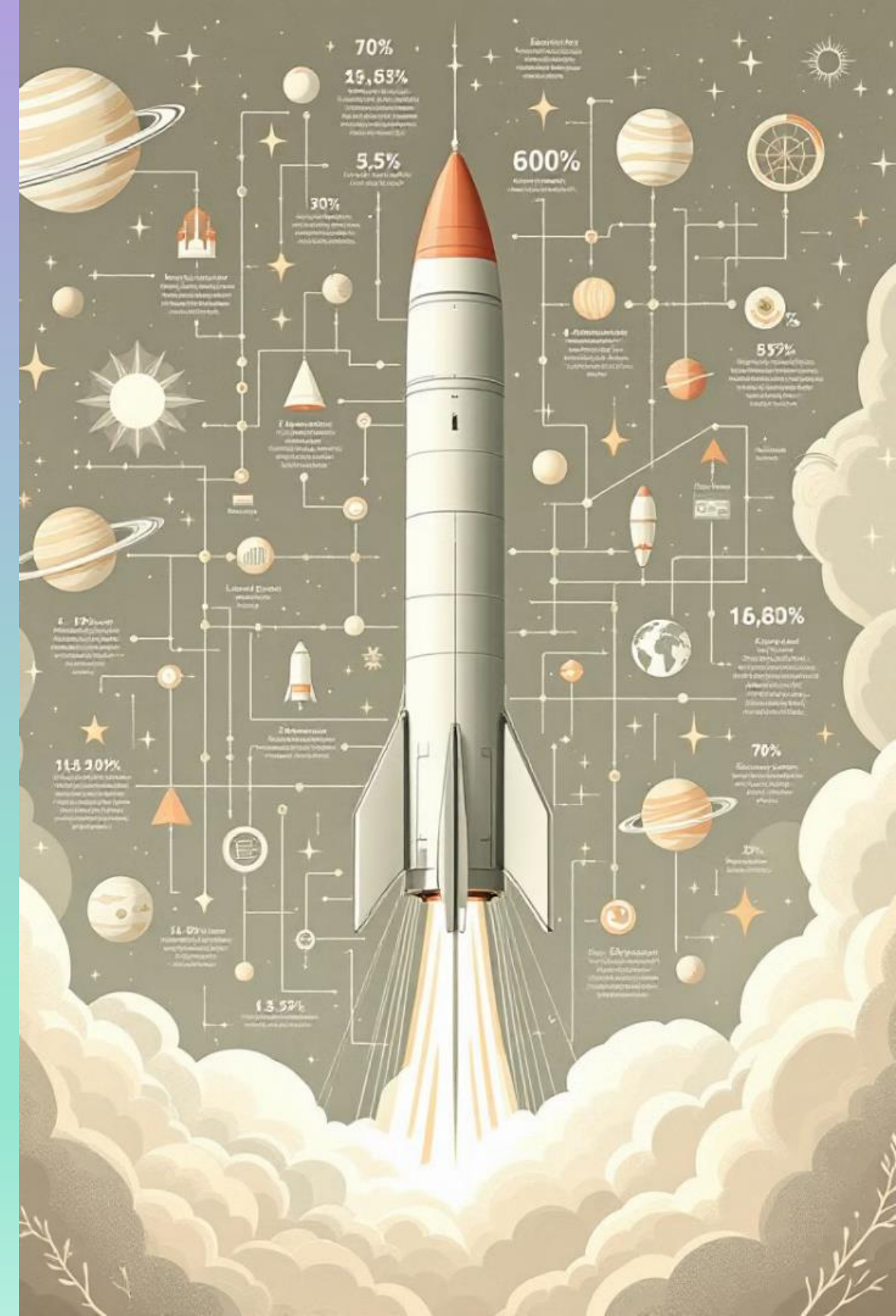
Factores Clave

Identificar las variables más influyentes en la tasa de reutilización, como estado del combustible, condiciones meteorológicas, etc.

3

Predicciones

Utilizar el modelo entrenado para pronosticar la probabilidad de reutilización de un cohete en futuras misiones.



Análisis Predictivo (Clasificación)

Análisis Predictivo Mediante Modelos de Clasificación

Proceso de Desarrollo del Modelo:

- Preparación de datos: Conversión de variables categóricas a numéricas: Se utilizaron variables ficticias para representar las columnas categóricas, como la órbita, el sitio de lanzamiento, la plataforma de aterrizaje y el número de serie.
- Conversión de tipos de datos: Todo el marco de datos se convirtió al tipo de variable float64 para garantizar la compatibilidad con los modelos de aprendizaje automático.
- Entrenamiento y ajuste del modelo: Modelos de clasificación: Se utilizaron cuatro modelos de clasificación diferentes: regresión logística, máquinas de vectores de soporte, k-vecinos más cercanos y árbol de decisión.
- GridSearchCV: Se utilizó GridSearchCV para encontrar los mejores parámetros para cada modelo.
- Evaluación del modelo: Puntuación de precisión: Se utilizó la puntuación de precisión para evaluar el rendimiento de cada modelo en los datos de prueba.
- Matriz de confusión: Se generó una matriz de confusión para visualizar el rendimiento del modelo en términos de verdaderos positivos, verdaderos negativos, falsos positivos y falsos negativos.

Análisis Predictivo (Clasificación)

Diagrama de flujo del proceso de desarrollo del modelo:

- Inicio
- Preparación de datos
- Entrenamiento del modelo
- Ajuste del modelo
- Evaluación del modelo
- Fin

URL de GitHub del cuaderno del análisis predictivo completado: [jupyter-labs-spacex-Machine_Learning_Prediction.ipynb](https://github.com/jupyter-labs-spacex-Machine_Learning_Prediction.ipynb)

Resultados

Resultados del Análisis Predictivo

- Precisión del modelo: Todos los modelos (regresión logística, máquinas de vectores de soporte, k-vecinos más cercanos y árbol de decisión) obtuvieron la misma puntuación de precisión del 83,3%.
- Matriz de confusión: Los modelos predijeron correctamente 12 aterrizajes exitosos (verdaderos positivos) y 3 aterrizajes fallidos (verdaderos negativos). También predijeron 3 aterrizajes exitosos cuando la etiqueta verdadera era un aterrizaje fallido (falsos positivos).

Observaciones

- Es importante tener en cuenta que el tamaño de la muestra utilizado para el análisis predictivo fue pequeño (solo 18). Se necesitan más datos para determinar el mejor modelo y mejorar la precisión.

Resultados: necesidad de más datos

El proyecto sugiere que se necesita más información para mejorar la precisión del modelo debido a que los cuatro modelos de aprendizaje automático utilizados (regresión logística, máquinas de vectores de soporte, vecinos más cercanos K y árbol de decisión) sobrestimaron los aterrizajes exitosos, a pesar de lograr una precisión de aproximadamente el 83,33%. Esto indica que el modelo no aprendió de manera efectiva a partir de los datos de entrenamiento y que los datos eran homogéneos.

Un tamaño de muestra pequeño puede contribuir a este problema. Con un conjunto de datos tan limitado, el modelo puede tener dificultades para generalizar a nuevas generalizar a nuevas instancias y, por lo tanto, sobrestimar las predicciones positivas.

Más datos proporcionarían al modelo más ejemplos de aterrizajes exitosos y fallidos, lo que le permitiría aprender patrones más complejos y mejorar su capacidad para hacer predicciones precisas. Un conjunto de datos más amplio y diverso también ayudaría a evitar el sobreajuste, donde el modelo se ajusta demasiado a los datos de entrenamiento y tiene un rendimiento deficiente en datos nuevos.

En resumen, la necesidad de más información se debe a:

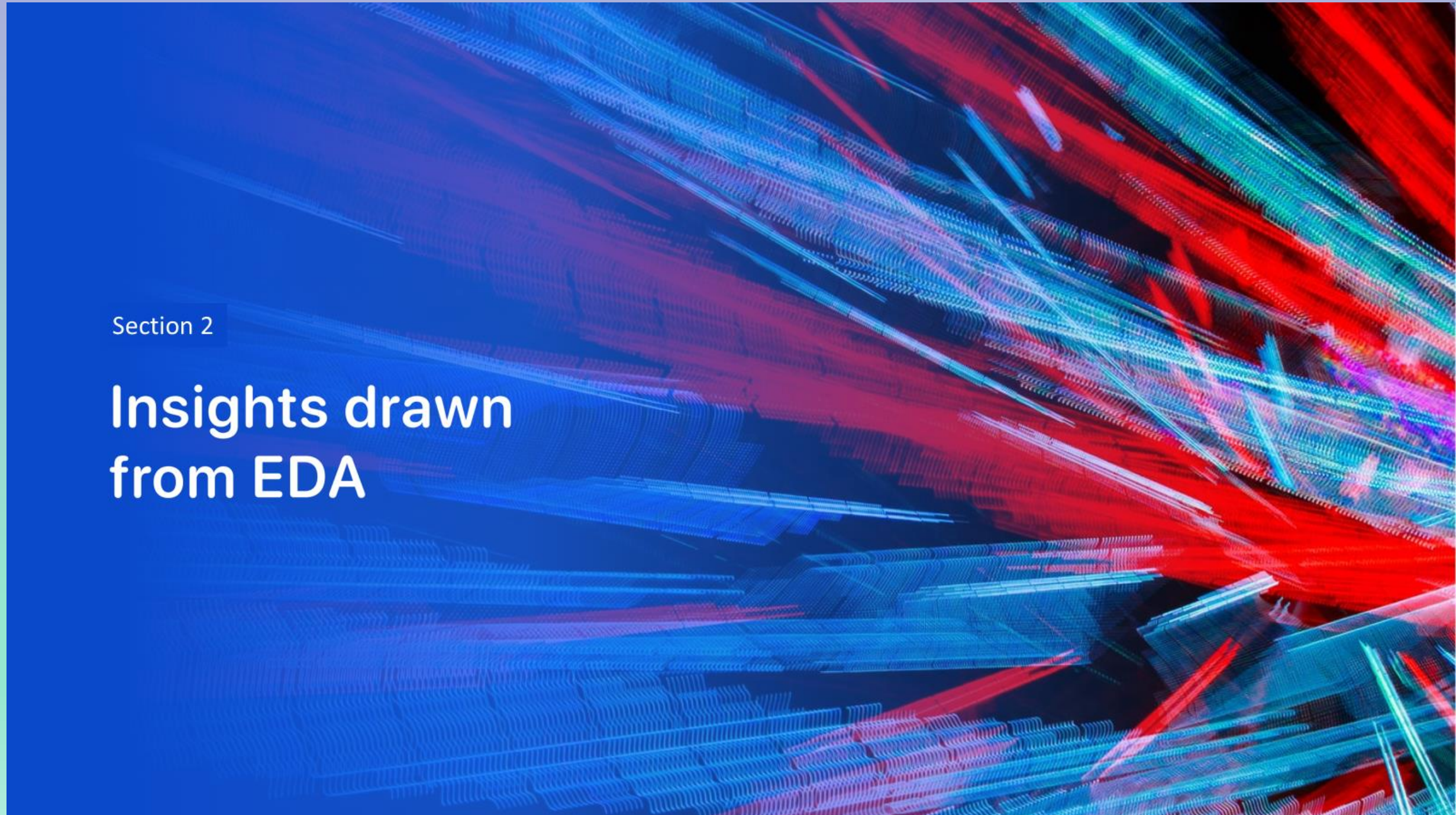
- La sobrestimación de los aterrizajes exitosos por parte de los modelos.
- La falta de aprendizaje efectivo a partir de los datos de entrenamiento.
- La homogeneidad de los datos.
- El tamaño de muestra pequeño.

Al aumentar la cantidad y la diversidad de los datos, el modelo podría aprender patrones más complejos y mejorar su capacidad para generalizar, lo que resultaría en una mayor precisión en las predicciones.

Seccion 2. Perspectivas extraídas de EDA

Section 2

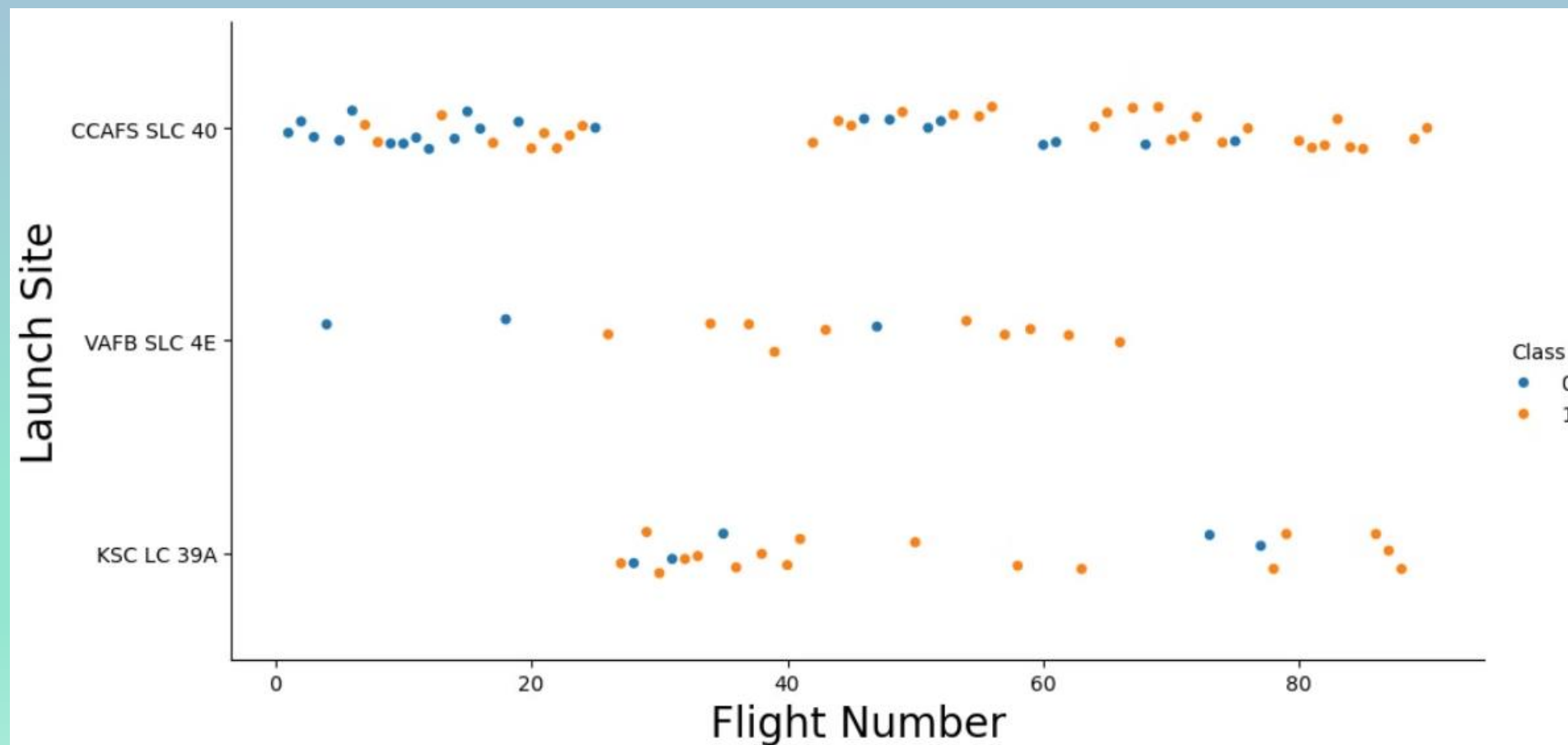
Insights drawn
from EDA



EDA con visualización de datos

Diagrama de dispersión: el número de vuelo frente al sitio de lanzamiento.

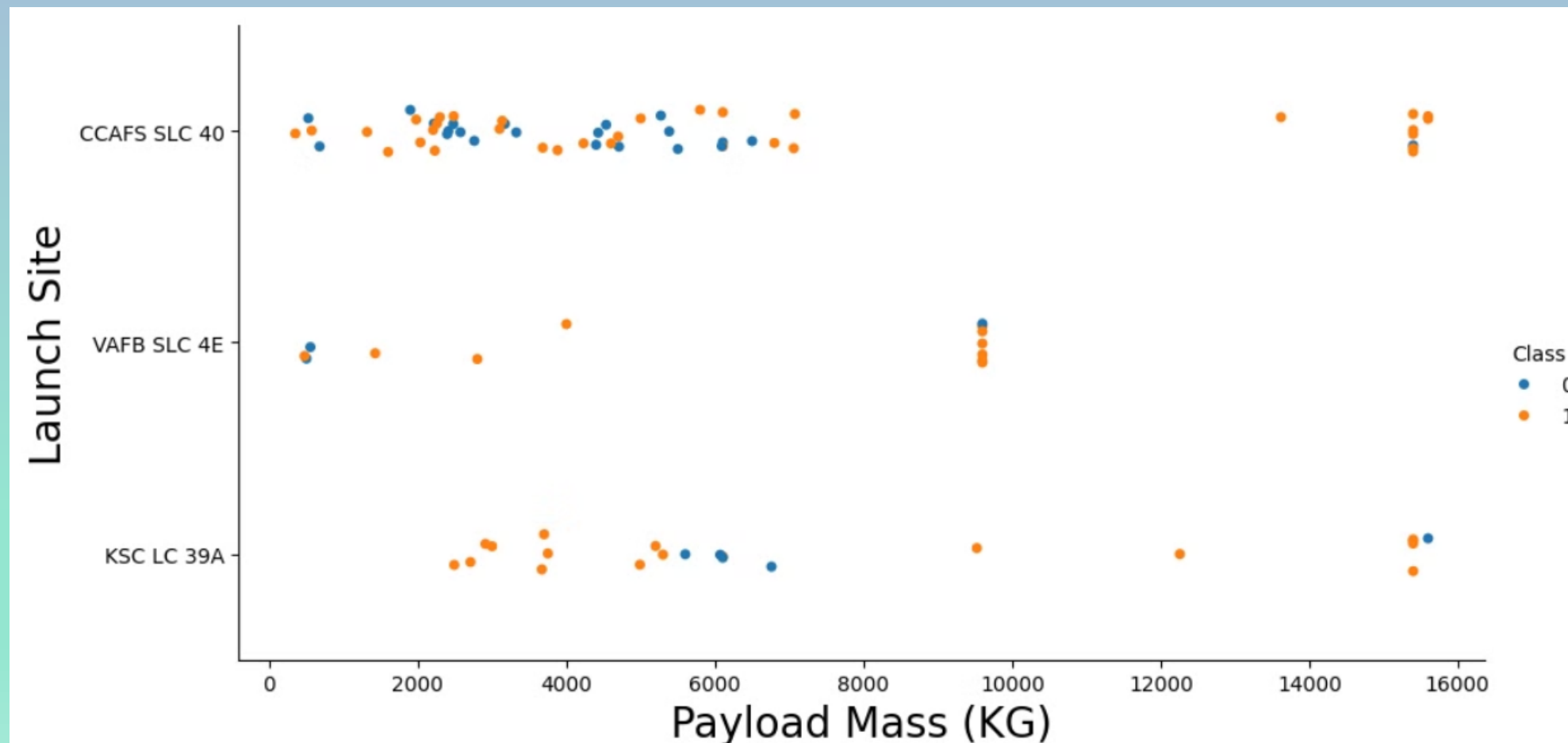
- Dirección URL de GitHub del cuaderno de EDA completado con visualización: [jupyter-labs-eda-dataviz-v2.ipynb](https://github.com/jupyter-labs/eda-dataviz-v2.ipynb)



EDA con visualización de datos

Diagrama de dispersión: la carga útil frente al sitio de lanzamiento.

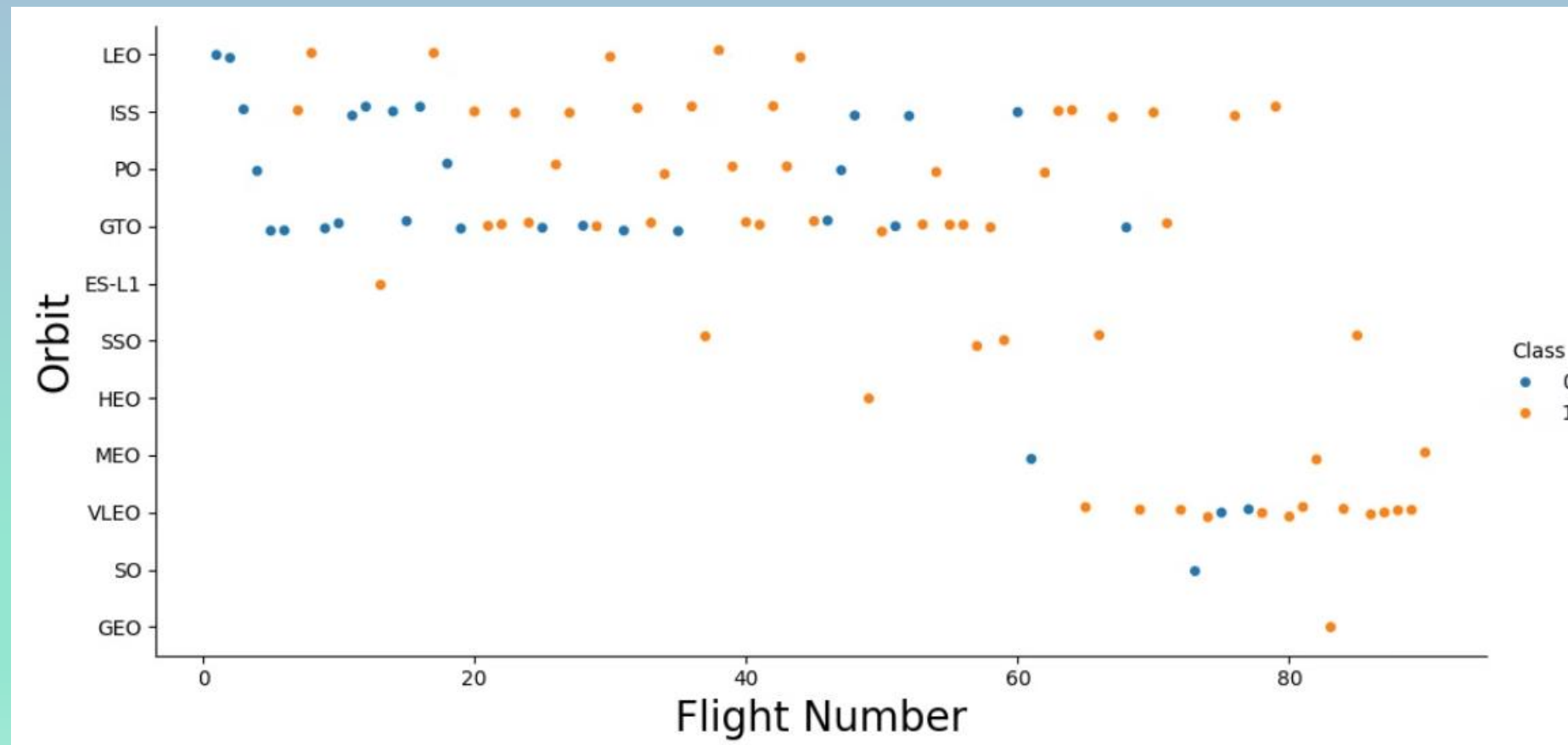
- Dirección URL de GitHub del cuaderno de EDA completado con visualización: : [jupyter-labs-eda-dataviz-v2.ipynb](#)



EDA con visualización de datos

Diagrama de dispersión: el número de vuelo frente al tipo de órbita.

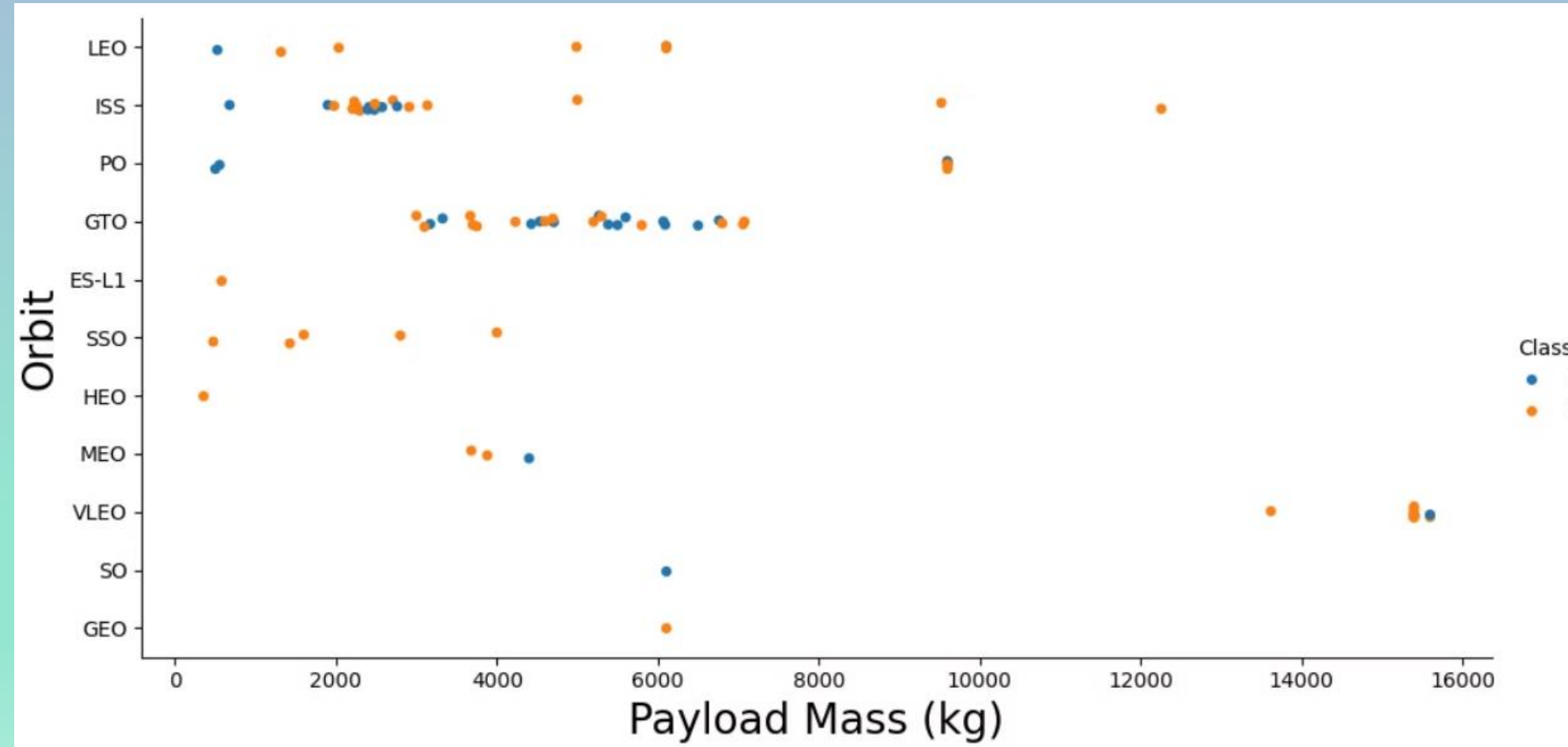
- Dirección URL de GitHub del cuaderno de EDA completado con visualización: [jupyter-labs-eda-dataviz-v2.ipynb](https://github.com/jupyter-labs/eda-dataviz-v2.ipynb)



EDA con visualización de datos

Diagrama de dispersión: la carga útil frente al tipo de órbita.

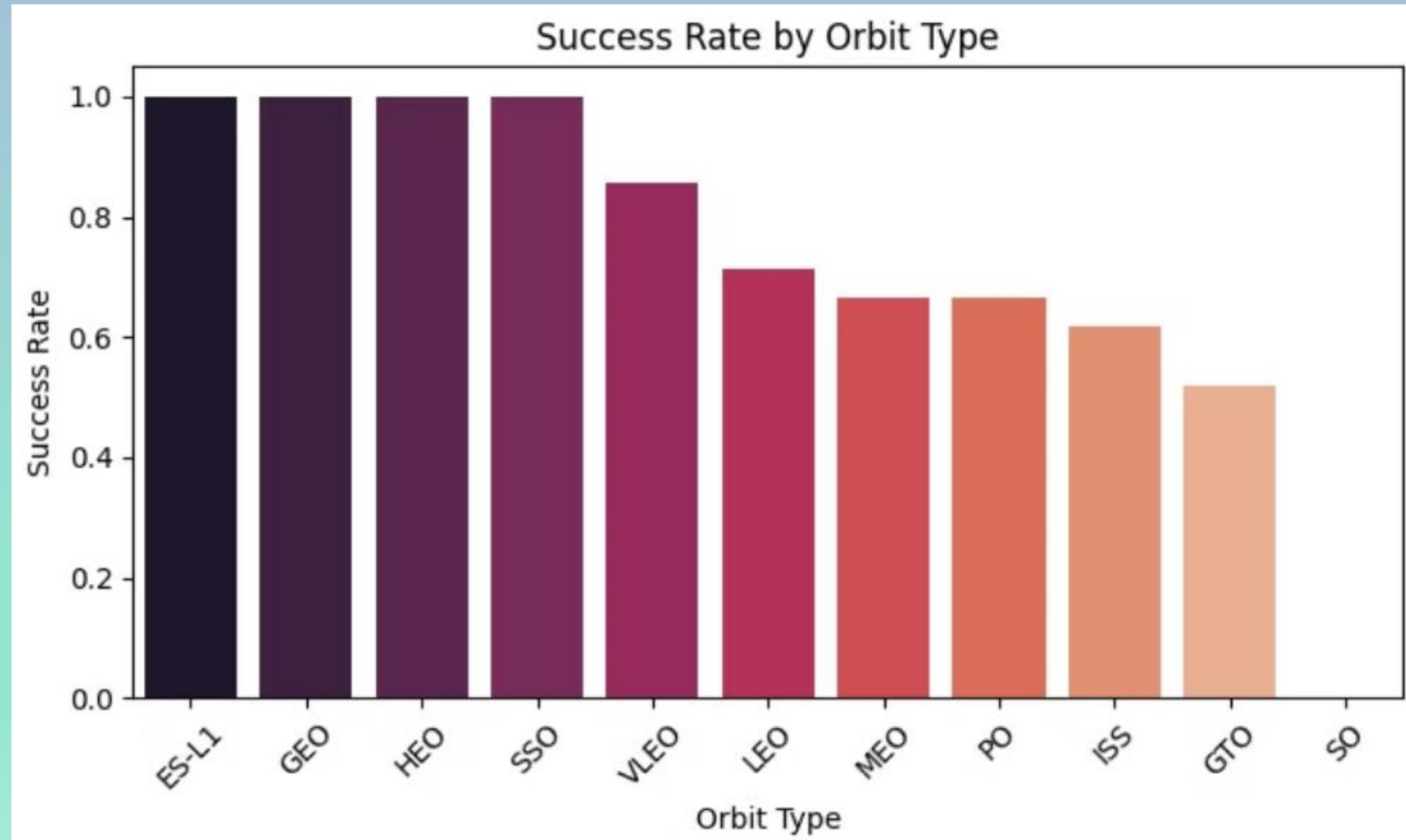
- Dirección URL de GitHub del cuaderno de EDA completado con visualización: [jupyter-labs-eda-dataviz-v2.ipynb](https://github.com/jupyterlabs-eda-dataviz-v2.ipynb)



EDA con visualización de datos

Gráfico de barras: facilitó la comparación de valores entre varios grupos a la vez, donde el eje x representaba una categoría y el eje y representaba un valor discreto. Los gráficos de barras se utilizaron para comparar la tasa de éxito para diferentes tipos de órbitas.

- Dirección URL de GitHub del cuaderno de EDA completado con visualización: [jupyter-labs-eda-dataviz-v2.ipynb](https://github.com/jupyterlabs-eda-dataviz-v2.ipynb)



EDA con visualización de datos

Gráfico de líneas: fueron útiles para mostrar las tendencias de los datos a lo largo del tiempo. Se utilizó un gráfico de líneas para mostrar las tasas de éxito durante varios años.

- Dirección URL de GitHub del cuaderno de EDA completado con visualización: [jupyter-labs-eda-dataviz-v2.ipynb](https://github.com/jupyter-labs/eda-dataviz-v2.ipynb)



EDA con SQL

Se utilizaron las siguientes consultas SQL para el EDA:

- **Consulta 1:** Muestra los nombres de los sitios de lanzamiento únicos en la misión espacial.
- Dirección URL de GitHub del EDA completado con SQL Notebook : [jupyter-labs-eda-sql-coursera sqlite.ipynb](https://github.com/jupyter-labs-eda-sql-coursera/sqlite.ipynb)

```
%sql SELECT DISTINCT LAUNCH_SITE FROM SPACEXTBL;
```

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

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

EDA con SQL

- **Consulta 2:** Muestra 5 registros donde los sitios de lanzamiento comienzan con la cadena 'CCA'.
- Dirección URL de GitHub del EDA completado con SQL Notebook : [jupyter-labs-eda-sql-coursera sqlite.ipynb](https://github.com/jupyter-labs/eda-sql-coursera/blob/master/sqlite.ipynb)

```
%sql SELECT * FROM SPACEXTBL 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
2010-06-04	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX
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
2012-05-22	7:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)
2012-10-08	0:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500	LEO (ISS)	NASA (CRS)
2013-03-01	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677	LEO (ISS)	NASA (CRS)

EDA con SQL

- **Consulta 3:** Masa total de la carga útil de la NASA.
- **Consulta 4:** Masa promedio de la carga útil por F9v1.1.
- Dirección URL de GitHub del EDA completado con SQL Notebook : [jupyter-labs-eda-sql-coursera_sqlite.ipynb](https://github.com/jupyter-labs/eda-sql-coursera/blob/master/sqlite.ipynb)

```
%sql SELECCIONAR SUMA ( PAYLOAD_MASS__KG_ ) DE SPACEXTBL DONDE CLIENTE = 'NASA (CRS)'
```

```
* sqlite:///my_data1.db  
Done.  
SUM(PAYLOAD_MASS__KG_)  
45596
```

```
%sql SELECT AVG(PAYLOAD_MASS__KG_) FROM SPACEXTBL WHERE BOOSTER_VERSION = 'F9 v1.1'
```

```
* sqlite:///my_data1.db  
Done.  
AVG(PAYLOAD_MASS__KG_)  
2928.4
```

EDA con SQL

- **Consulta 5:** Se logró el primer resultado de aterrizaje exitoso en la plataforma terrestre.
- **Consulta 6:** Aterrizaje exitoso de Drone Ship con carga útil entre 4000 y 6000.
- Dirección URL de GitHub del EDA completado con SQL Notebook : [jupyter-labs-eda-sql-coursera_sqlite.ipynb](https://github.com/jupyter-labs/eda-sql-coursera/blob/master/sqlite.ipynb)

```
%sql select min(Date) from SPACEXTBL where Landing_Outcome LIKE 'Success (ground pad)';
```

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

```
Done.
```

```
min(Date)
```

```
2015-12-22
```

```
%sql select Booster_Version from SPACEXTBL \
      where PAYLOAD_MASS__KG_ >4000 and PAYLOAD_MASS__KG_ < 6000 \
      and Landing_Outcome LIKE 'Success (drone ship)';
```

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

```
Done.
```

```
Booster_Version
```

```
F9 FT B1022
```

```
F9 FT B1026
```

```
F9 FT B1021.2
```

```
F9 FT B1031.2
```


EDA con SQL

- **Consulta 7:** Número total de cada resultado de la misión.
- Dirección URL de GitHub del EDA completado con SQL Notebook : [jupyter-labs-eda-sql-coursera_sqlite.ipynb](https://github.com/jupyter-labs/eda-sql-coursera/blob/master/sqlite.ipynb)

```
%sql select MISSION_OUTCOME, COUNT(MISSION_OUTCOME) from SPACEXTBL Group by MISSION_OUTCOME;
```

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

```
Done.
```

Mission_Outcome	COUNT(MISSION_OUTCOME)
Failure (in flight)	1
Success	98
Success	1
Success (payload status unclear)	1

EDA con SQL

- **Consulta 8:** Impulsores que llevaron la carga útil máxima.
- Dirección URL de GitHub del EDA completado con SQL Notebook : [jupyter-labs-eda-sql-coursera_sqlite.ipynb](https://github.com/jupyter-labs-eda-sql-coursera/sqlite.ipynb)

```
%%sql SELECT DISTINCT Booster_Version, PAYLOAD_MASS__KG_  
from SPACEXTBL  
WHERE PAYLOAD_MASS__KG_ = (SELECT max(PAYLOAD_MASS__KG_) FROM SPACEXTBL);
```

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

```
Done.
```

Booster_Version	PAYLOAD_MASS__KG_
F9 B5 B1048.4	15600
F9 B5 B1049.4	15600
F9 B5 B1051.3	15600
F9 B5 B1056.4	15600
F9 B5 B1048.5	15600
F9 B5 B1051.4	15600
F9 B5 B1049.5	15600
F9 B5 B1060.2	15600
F9 B5 B1058.3	15600
F9 B5 B1051.6	15600
F9 B5 B1060.3	15600
F9 B5 B1049.7	15600

EDA con SQL

- **Consulta 9:** Registros de aterrizaje fallidos de Drone Ship en 2015.
- **Consulta 10:** Clasificación de los recuentos de aterrizajes exitosos entre 2010-06-04 y 2017-03-20 (como Fallo (barco no tripulado) o Éxito (plataforma terrestre)). (plataforma terrestre)).
- Dirección URL de GitHub del EDA completado con SQL Notebook : [jupyter-labs-eda-sql-coursera_sqlite.ipynb](https://github.com/jupyter-labs/eda-sql-coursera/blob/master/sqlite.ipynb)

```
%%sql select BOOSTER_VERSION,Date, substr(Date, 6, 2) AS Mes, substr(Date, 0, 5)AS Año,Landing_outcome, LAUNCH_SITE
from SPACEXTBL
where Landing_outcome = 'Failure (drone ship)'and DATE like '2015%';
```

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

Booster_Version	Date	Mes	Año	Landing_Outcome	Launch_Site
F9 v1.1 B1012	2015-01-10	01	2015	Failure (drone ship)	CCAFS LC-40
F9 v1.1 B1015	2015-04-14	04	2015	Failure (drone ship)	CCAFS LC-40

```
%%sql select Landing_Outcome, COUNT(Landing_outcome)
from SPACEXTBL
where Landing_Outcome in ('Failure (drone ship)','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

Seccion 3. Análisis de proximidad de sitios de lanzamiento

Section 3

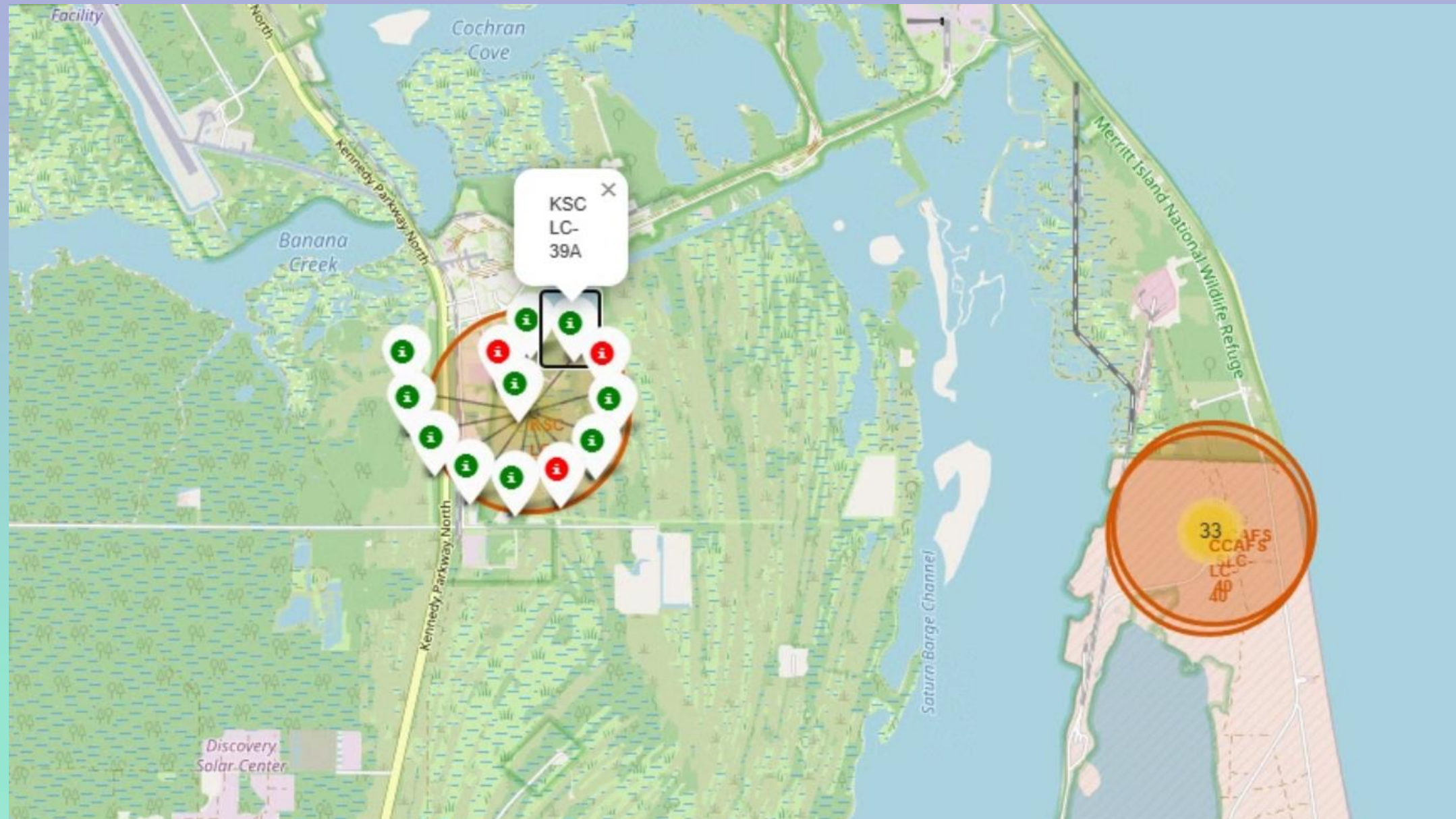
Launch Sites Proximities Analysis



Sitios de lanzamiento en un mapa global

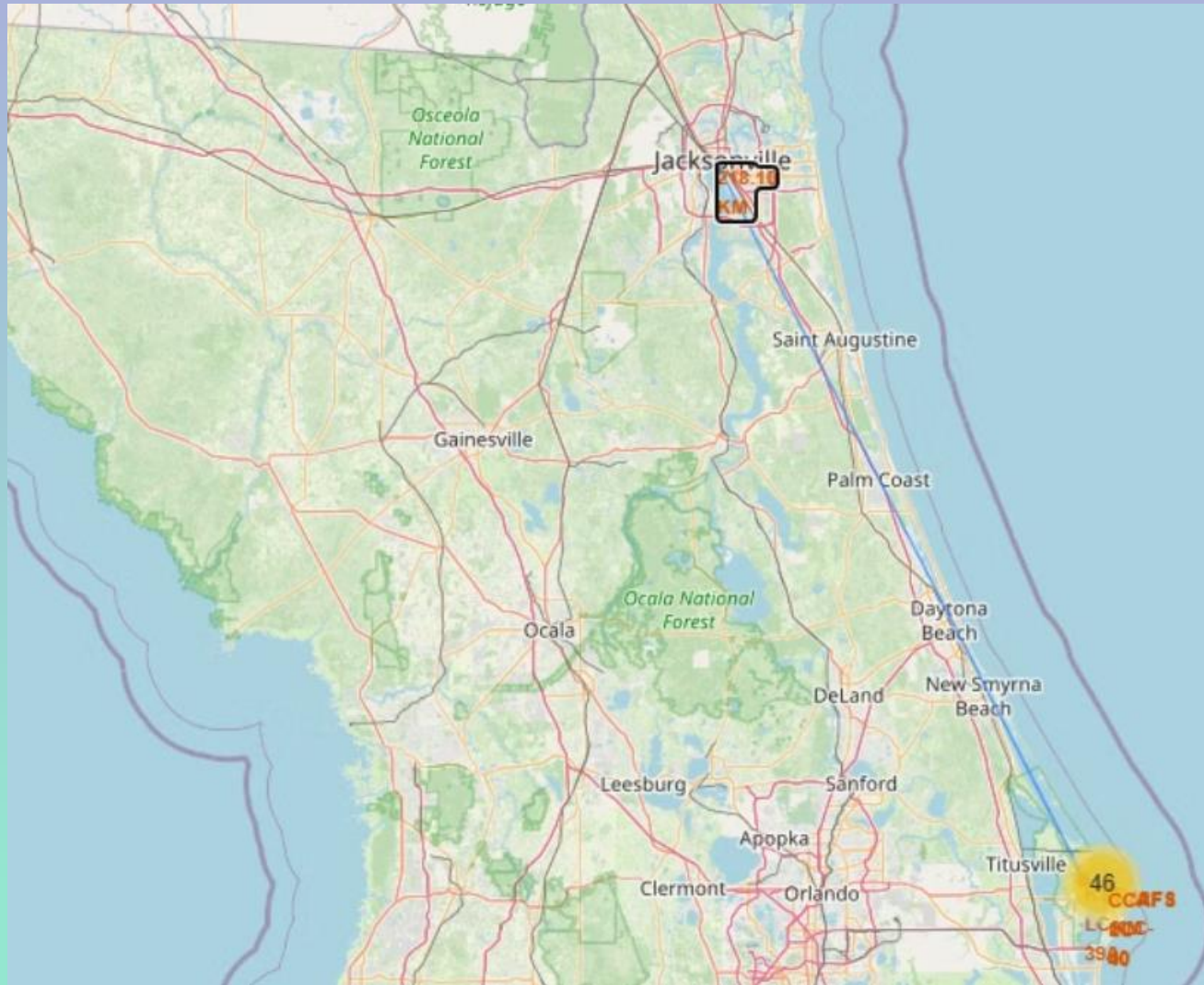


Lanzamientos etiquetados en color en el mapa



Marcados los lanzamientos exitosos/fallidos de cada sitio en el mapa.

Proximidad de Jacksonville con la distancia calculada

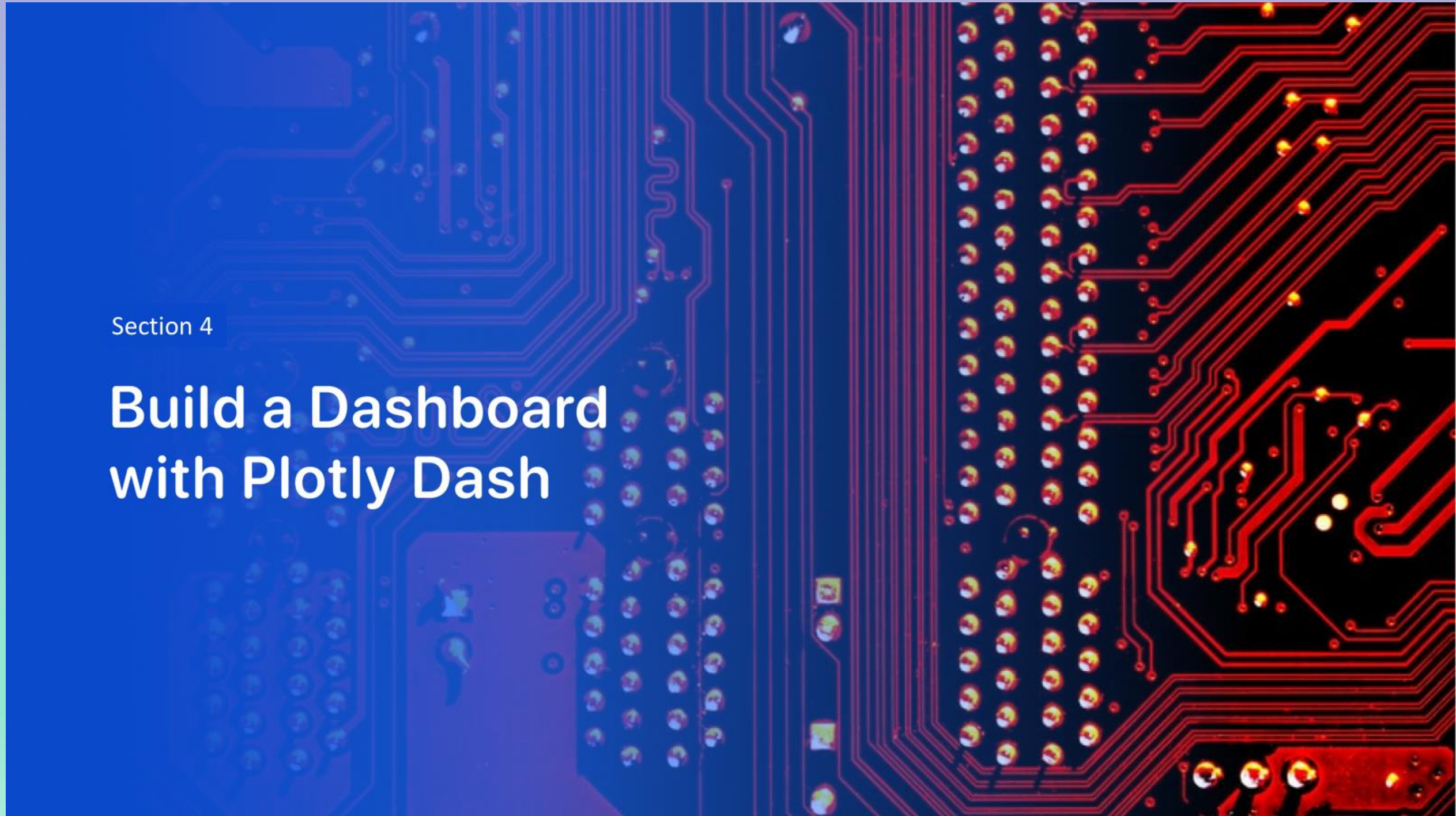


La proximidad a la costa, las carreteras, las vías férreas y las ciudades se tuvieron en cuenta para comprender la ubicación estratégica de los sitios de lanzamiento. En este caso, se analizó la proximidad con Jacksonville.

Sección 4. Construir un tablero con *Plotly-Dash*.

Section 4

Build a Dashboard with Plotly Dash



Recuento de éxito de lanzamiento para todos los sitios los sitios

Total Success Launches by Site



Sitio de lanzamiento con la tasa de éxito de lanzamiento más alta

Total Success Launches by Site: KSC LC-39A

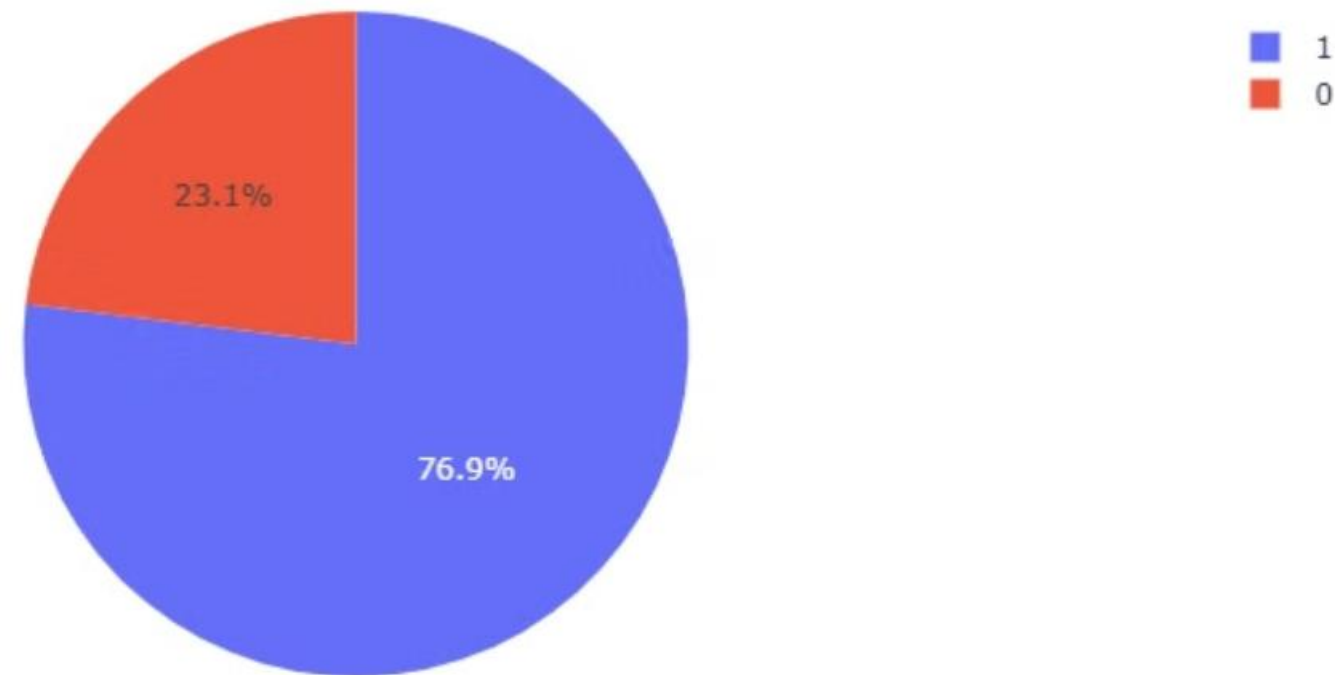


Diagrama de dispersión de la carga útil

Diagrama de dispersión de la carga útil frente al resultado del lanzamiento para todos los sitios, con diferentes cargas útiles seleccionadas en el control seleccionadas en el control deslizante de rango.



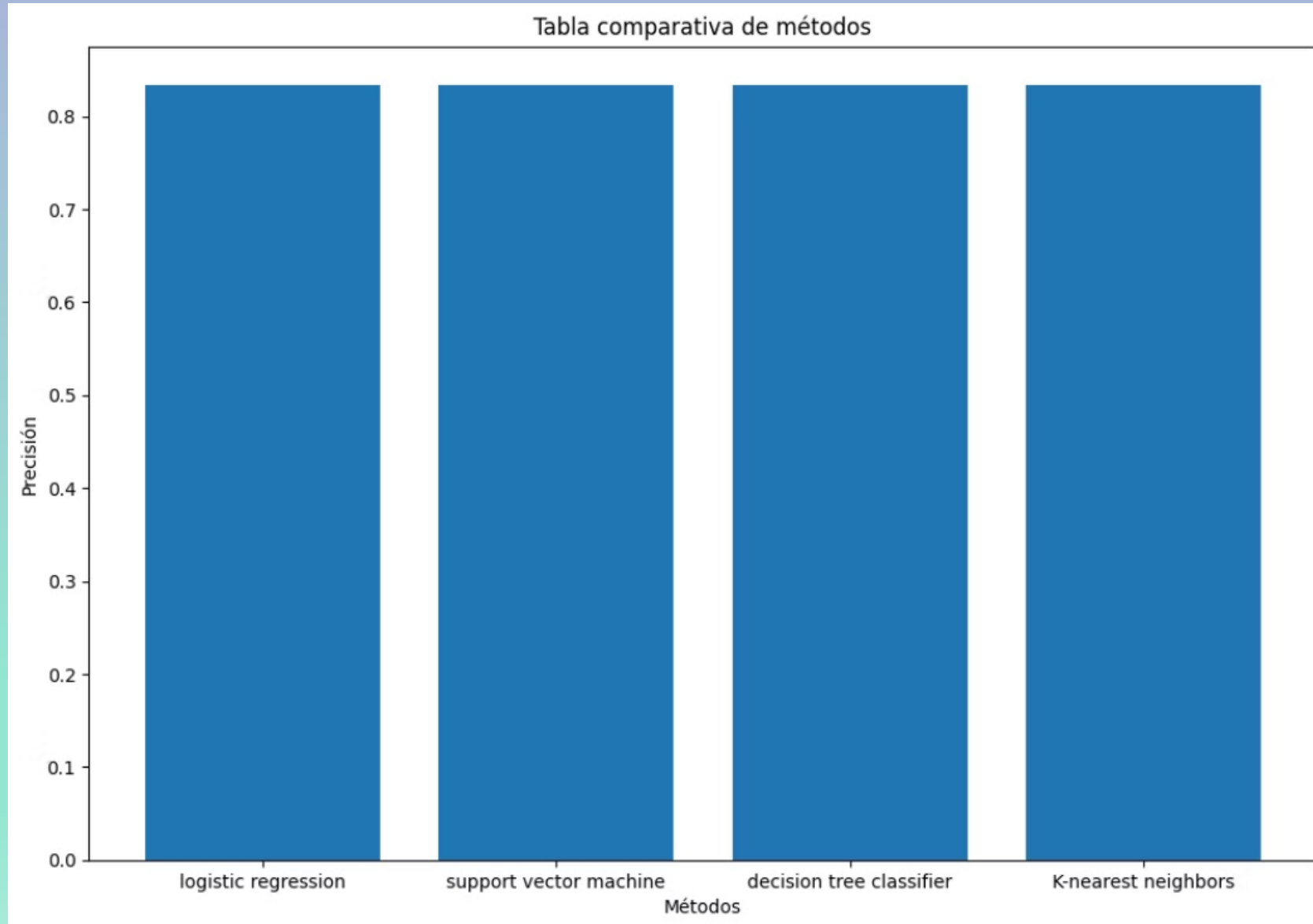
Sección 5. Análisis Predictivo (Clasificación)

Section 5

Predictive Analysis (Classification)

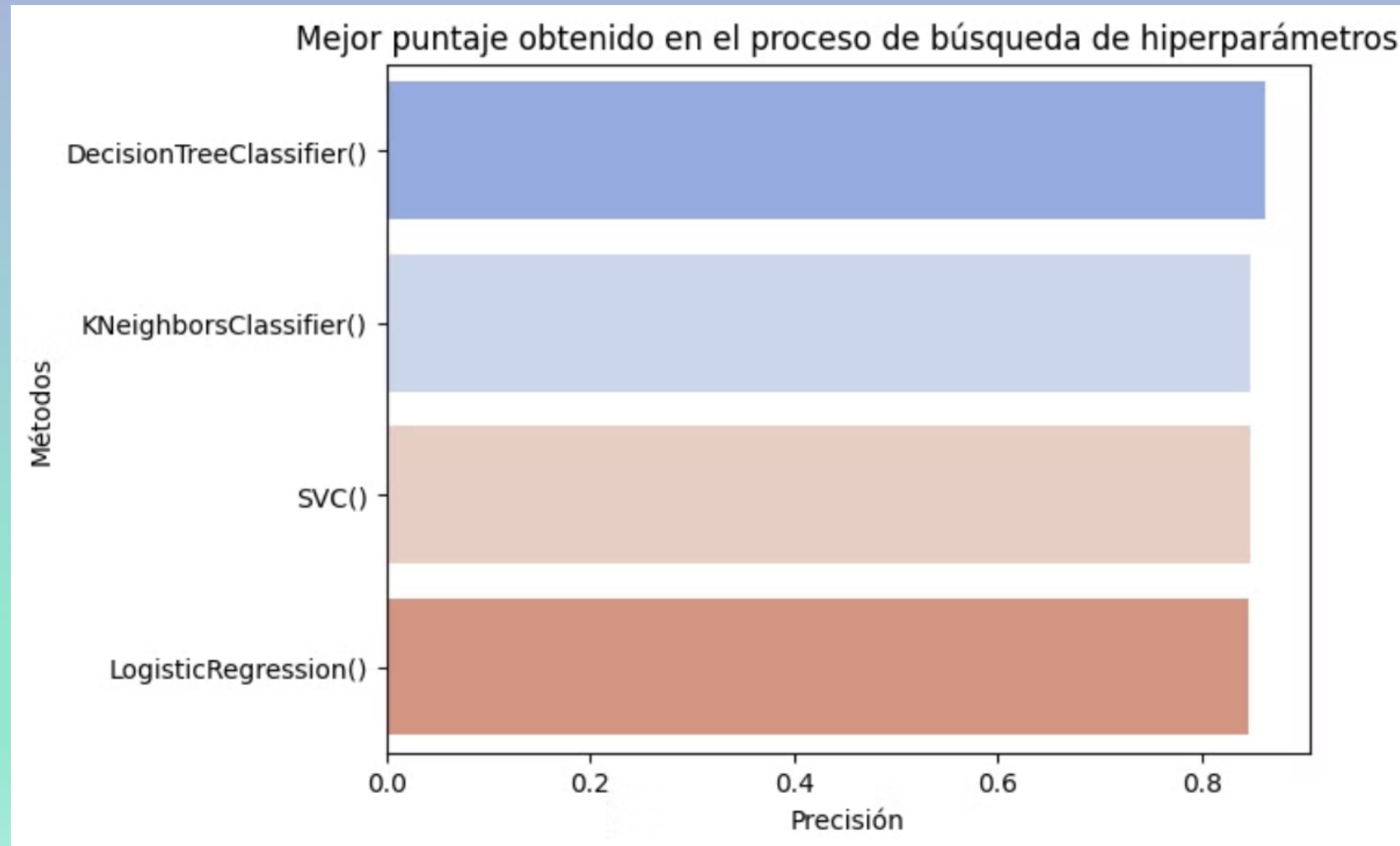
Precisión de la clasificación

- Precisión del modelo creado para los modelos de clasificación construidos.



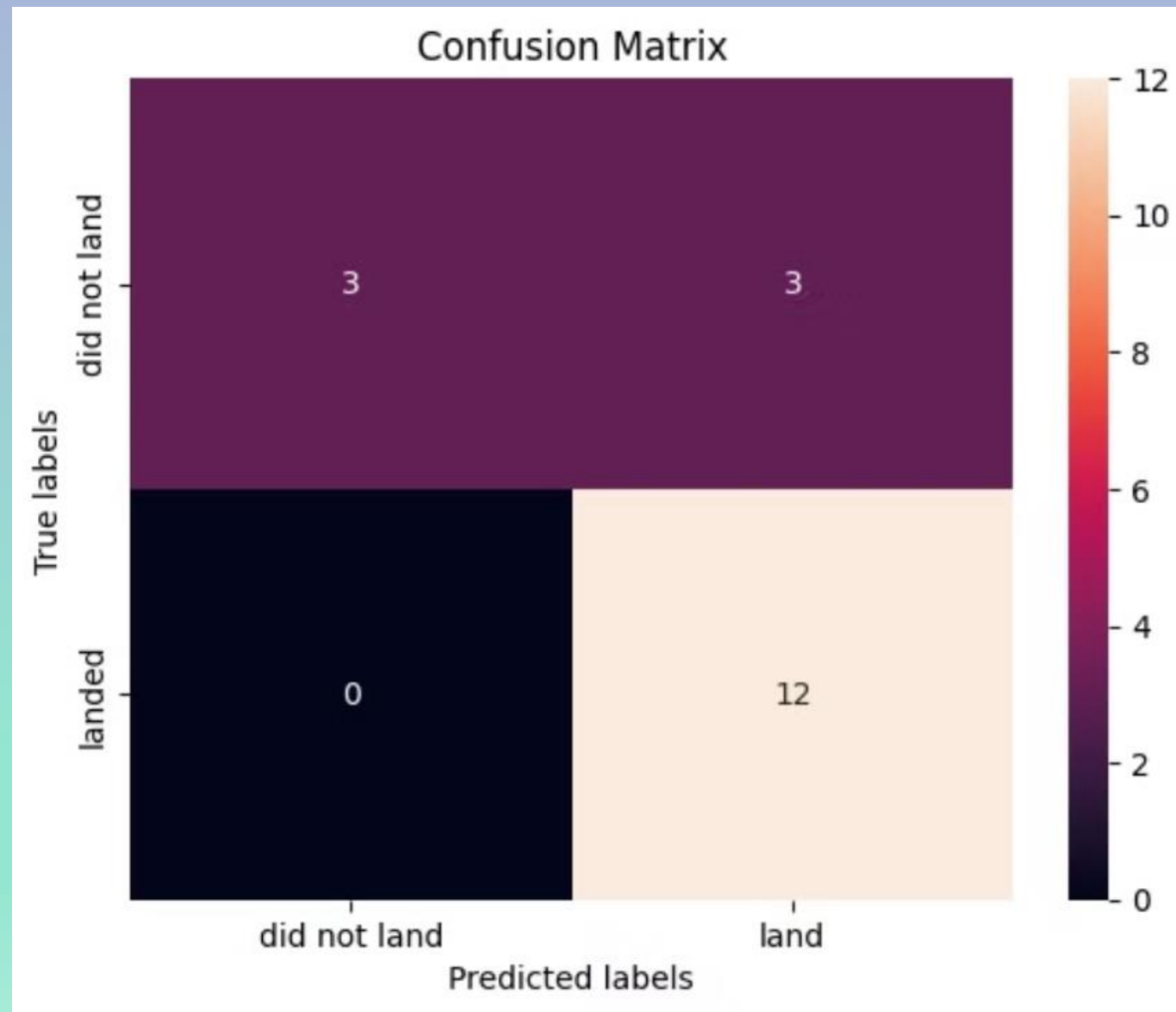
Precisión de la clasificación

- Búsqueda de hiperparámetros



Matriz de confusión

- Matriz de confusión de los 4 modelos.



Conclusiones

El análisis realizado en este proyecto de ciencia de datos, cuyo objetivo era predecir el éxito del aterrizaje de la primera etapa de los cohetes de los cohetes SpaceX, arrojó las siguientes **conclusiones**:

- Existe una correlación positiva entre el número de vuelos y la tasa de éxito de aterrizaje. Cuanto mayor es el número de vuelos, mayor es la probabilidad de éxito. Se ha observado que la tasa de éxito promedio es del 66.6% y ha mejorado con el paso de los años.
- Las órbitas SSO, HEO, GEO y ES-L1 presentan las tasas de éxito más altas (100%). Esto se debe a que son órbitas geoestacionarias, lo que les permite permanecer en una posición fija, característica ideal para operaciones de satélites de comunicaciones. Estas órbitas se encuentran más cerca del ecuador.
- La tasa de éxito también está relacionada con la masa de la carga útil. Las cargas útiles más ligeras generalmente tuvieron más éxito que las más pesadas. El rango de carga útil con mayor tasa de éxito se encuentra entre 2000 y 6000 kg.
- La ubicación de los sitios de lanzamiento es estratégica:
 - Están situados cerca de carreteras y vías férreas para facilitar el transporte de personal y carga.
 - Se encuentran alejados de las ciudades por razones de seguridad.
 - Están próximos a la costa en caso de lanzamientos fallidos.

Recomendaciones y consideraciones finales

Recomendaciones para SpaceX:

- Se puede utilizar este modelo para predecir con una precisión relativamente alta si un lanzamiento tendrá un aterrizaje exitoso de la Etapa 1 antes del lanzamiento. Esta información puede ayudar a determinar si el lanzamiento debe seguir adelante o no.
- Se recomienda recopilar más datos para determinar el mejor modelo de aprendizaje automático y mejorar la precisión. Los datos utilizados en este estudio fueron insuficientes y podrían haber presentado poca variación en las clases, sobreajuste o datos homogéneos.

Consideraciones finales:

- Los modelos de clasificación predicen una probabilidad del 83% de que SpaceX pueda reutilizar la primera etapa de sus cohetes. Si los lanzamientos logran aterrizar con éxito, se puede predecir que cada lanzamiento costaría 62 millones de dólares.
- Se recomienda que la carga útil en cada lanzamiento se encuentre entre 2000 y 4000 kg para garantizar el éxito.
- Se sugiere utilizar los sitios de lanzamiento KSC LC 39A o CCAAF SLC 40, que, aunque han tenido algunos lanzamientos fallidos, su tasa de éxito promedio es del 66% y se espera que mejore a medida que aumenta el número de vuelos.

En general, es factible considerar el lanzamiento de cohetes en los próximos años, siguiendo estas pautas para ofrecerlos a un precio competitivo.

Apéndice del proyecto

Este apéndice incluye información complementaria al análisis principal realizado en el proyecto, proporcionando detalles sobre las fuentes de datos, los procesos de análisis, visualización y modelos predictivos utilizados. A continuación, se enlistan los enlaces a cada uno de los Jupyter Notebook, donde se encuentra el código fuente para cada fase del proyecto:

1. Recopilación de datos

- API de SpaceX :[jupyter-labs-spacex-data-collection-api.ipynb](#)
- Web Scraping: [jupyter-labs-webscraping.ipynb](#)

2. Preparación de datos

- Data Wrangling: [labs-jupyter-spacex-Data%20wrangling.ipynb](#)

3. Análisis exploratorio de datos (EDA)

- EDA con visualizaciones: [jupyter-labs-eda-dataviz-v2.ipynb](#)
- EDA con SQL: : [jupyter-labs-eda-sql-coursera_sqlite.ipynb](#)

4. Análisis visual interactivo

- Mapa de Sitios de Lanzamiento con Folium: [lab-jupyter-launch-site-location-v2.ipynb](#)
- Dashboard interactivo con Plotly Dash: [spacex_dash_app.ipynb](#)

5. Modelos predictivos

- Análisis predictivo con Maquine Learning: [jupyter-labs-spacex-Machine_Learning_Prediction.ipynb](#)

Fuentes de datos utilizadas

Los datos utilizados en este proyecto provienen de las siguientes fuentes:

- [IBM Data Science Professional Certificate](#) : Conjunto de datos y herramientas proporcionados durante el curso.
- **Repositorio del Proyecto en GitHub:** <https://github.com/antoniodeepblue/Proyecto de Ciencia de Datos- de SpaceX Falcon 9>
- **API Pública de SpaceX:** Proporciona información detallada sobre cohetes, plataformas de lanzamiento, cargas útiles, núcleos y lanzamientos núcleos y lanzamientos de SpaceX.
- **Wikipedia:** Información complementaria extraída de la página de Wikipedia de SpaceX, incluyendo detalles como fecha y ubicación de los ubicación de los lanzamientos, cargas útiles, resultados de lanzamientos y versiones de propulsores.
- **Archivos CSV:** Diversos archivos CSV contienen datos adicionales, como la ubicación geográfica de los sitios de lanzamiento y los resultados de los intentos de aterrizaje.

Thank you!

