

Visualización del conjunto de datos Pokemon

Utilice Seaborn para la visualización de datos

Cómo utilizar Seaborn para la visualización de datos?

- La visualización de datos es el arte de representar datos en forma de gráficos. Es una herramienta útil para los profesionales que trabajan con datos, es decir, analistas financieros, analistas de negocio, analistas de datos, científicos de datos, por nombrar algunos ejemplos. En este tutorial trabajaremos con Seaborn, una librería de Python.

Seaborn

- Seaborn es una biblioteca Python de código abierto construida sobre matplotlib. Se utiliza para la visualización de datos y el análisis exploratorio de datos. Seaborn funciona fácilmente con dataframes y la biblioteca Pandas. Los gráficos creados también se pueden personalizar fácilmente. A continuación se presentan algunos beneficios de la visualización de datos.

Instalación de Seaborn

- Si está trabajando en Google Colab, puede omitir el paso de instalación. Sin embargo, si está trabajando en su máquina local, tendrá que instalar Seaborn. Recomendando encarecidamente crear un entorno virtual para gestionar mejor los paquetes.

```
python -m venv venv  
venv/Scripts/activate  
pip install pandas, matplotlib, seaborn
```

Importar Seaborn y cargar el conjunto de datos

```
import seaborn as sns
import pandas
import matplotlib.pyplot as plt
```

- Seaborn tiene 18 conjuntos de datos incorporados, que se pueden encontrar utilizando el siguiente comando.

```
sns.get_dataset_names()
```

- Ejemplo dataset del Titanic

```
df = sns.load_dataset('titanic')
df.head()
```

Importar Seaborn y cargar el conjunto de datos

```
import seaborn as sns
import pandas
import matplotlib.pyplot as plt
```

- Seaborn tiene 18 conjuntos de datos incorporados, que se pueden encontrar utilizando el siguiente comando.

```
sns.get_dataset_names()
```

- Ejemplo dataset del Titanic

```
df = sns.load_dataset('titanic')
df.head()
```

Ejercicio dataset Pokemon

- En este ejercicio ocuparemos la biblioteca seaborn par visualizar un dataset de pokemon.
- Puedes descargar el conjunto de datos Pokemon desde [aquí](#).

Para subir un archivo a google drive.

```
from google.colab import files  
files.upload()
```

Ejercicio dataset Pokemon

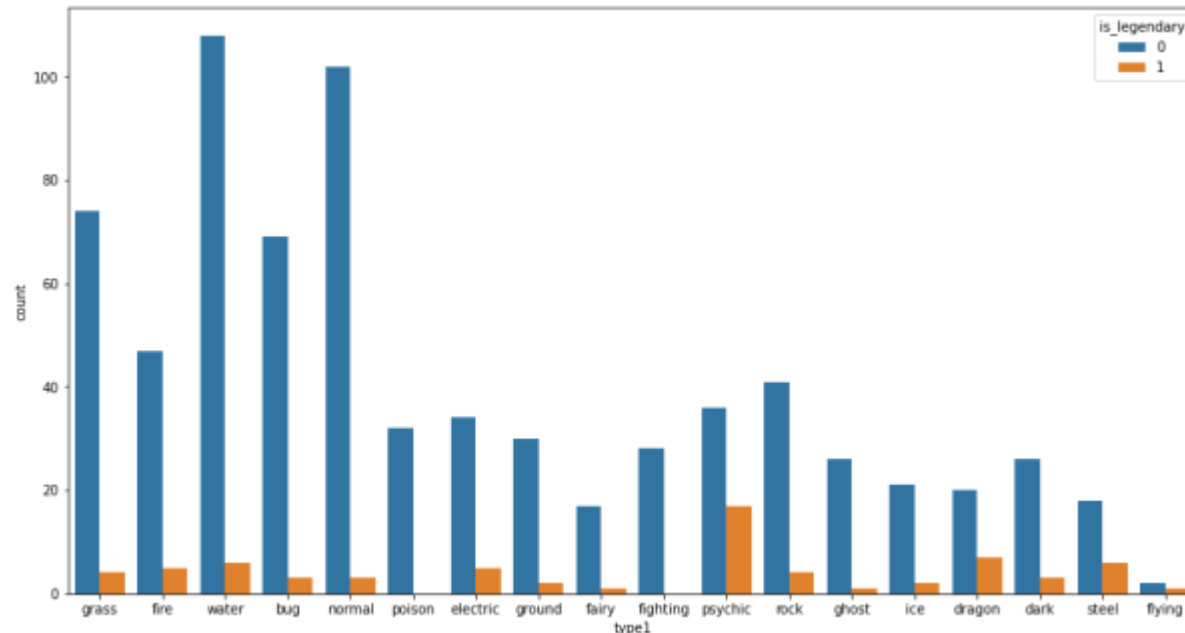
- Puedes saltarte el paso anterior si estás trabajando en tu máquina local. Asegúrate de que el archivo descargado está en la misma carpeta que tu archivo Python.

```
pokemon_df = pd.read_csv('pokemon.csv')  
pokemon_df.head()
```


Ejercicio dataset Pokemon

Distribución de tipos Pokemon

```
plt.figure(figsize=(15,8))
sns.countplot(x = 'type1' , data = pokemon_df,
hue = 'is_legendary')
```

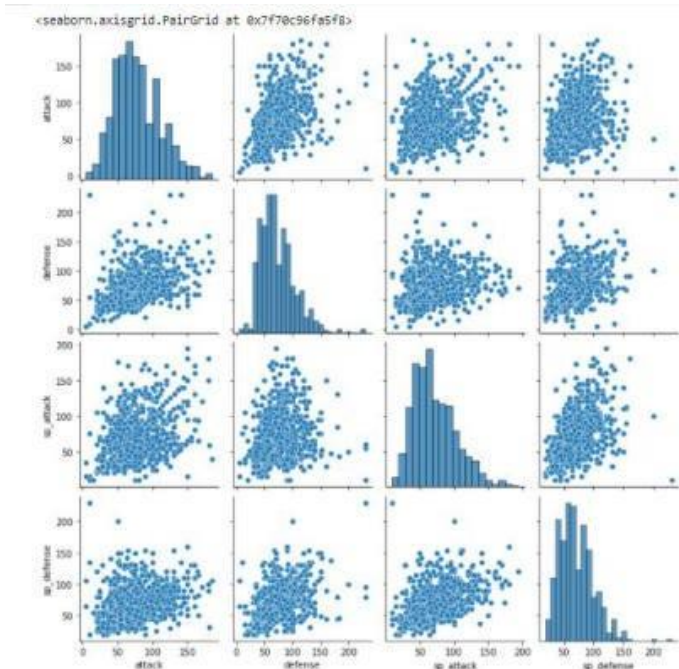


Como Seaborn está construido sobre matplotlib, podemos usar la función de matplotlib para aumentar el tamaño de nuestro gráfico. Mirando el gráfico, podemos concluir que el Agua y Normal son los tipos de Pokemon dominantes. También podemos concluir que el tipo dominante para Pokemons Legendarios es Psíquico.

Ejercicio dataset Pokemon

Relación entre ataque y defensa

```
sns.pairplot(x_vars=['attack', 'defense', 'sp_attack', 'sp_defense'],  
             y_vars=['attack', 'defense', 'sp_attack', 'sp_defense'],  
             data = pokemon_df)
```



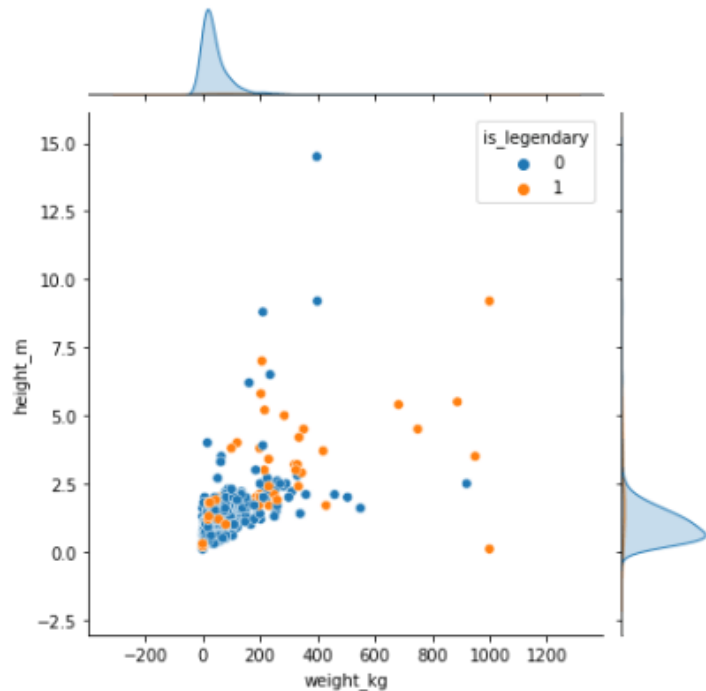
Opcionalmente, también puede pasar el parámetro **hue** para obtener más información.

En seaborn, el parámetro **hue** determina qué columna del data frame debe utilizarse para la codificación del color

Ejercicio dataset Pokemon

Relación entre altura y peso

```
sns.jointplot(x = 'weight_kg', y = 'height_m',  
data = pokemon_df,hue = 'is_legendary')
```



Basándonos en el gráfico, podemos concluir que la altura y el peso de un Pokemon no tienen ninguna correlación.

Probar hacer estos gráficos

- Distribución de Altura/Peso.
- Número de Pokémon legendarios frente a Pokémon no legendarios.
- Gráfico de regresión de ataque especial y defensa especial.
- Heatmap para mostrar la correlación entre las características..