

## ACTIVIDAD 1 - ANÁLISIS DE DATOS

```
# Instalar seaborn y openpyxl si no están instalados
import subprocess
import sys

# Verificar si las bibliotecas están instaladas, si no instalarlas
def install(package):
    subprocess.check_call([sys.executable, "-m", "pip", "install", package])

try:
    import seaborn as sns
except ImportError:
    install('seaborn')

try:
    import openpyxl
except ImportError:
    install('openpyxl')
```

```
# Importar las bibliotecas necesarias
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import os

# Verificar la ruta actual y el contenido del directorio para asegurarnos de
que el archivo existe
print("Directorio actual:", os.getcwd())
print("Contenido del directorio actual:", os.listdir())

# Ruta relativa al archivo Excel
relative_path = 'data s sosten/analisis de datos/Datos Actividad.xlsx'

# Convertir la ruta relativa en absoluta
absolute_path = os.path.abspath(relative_path)
print("Ruta absoluta:", absolute_path)

# Verificar si una ruta es absoluta
is_absolute = os.path.isabs(relative_path)
print("¿Es una ruta absoluta?", is_absolute)

# Si no es absoluta, usar la ruta convertida a absoluta
file_path = absolute_path if is_absolute else os.path.join(os.getcwd(),
relative_path)
```

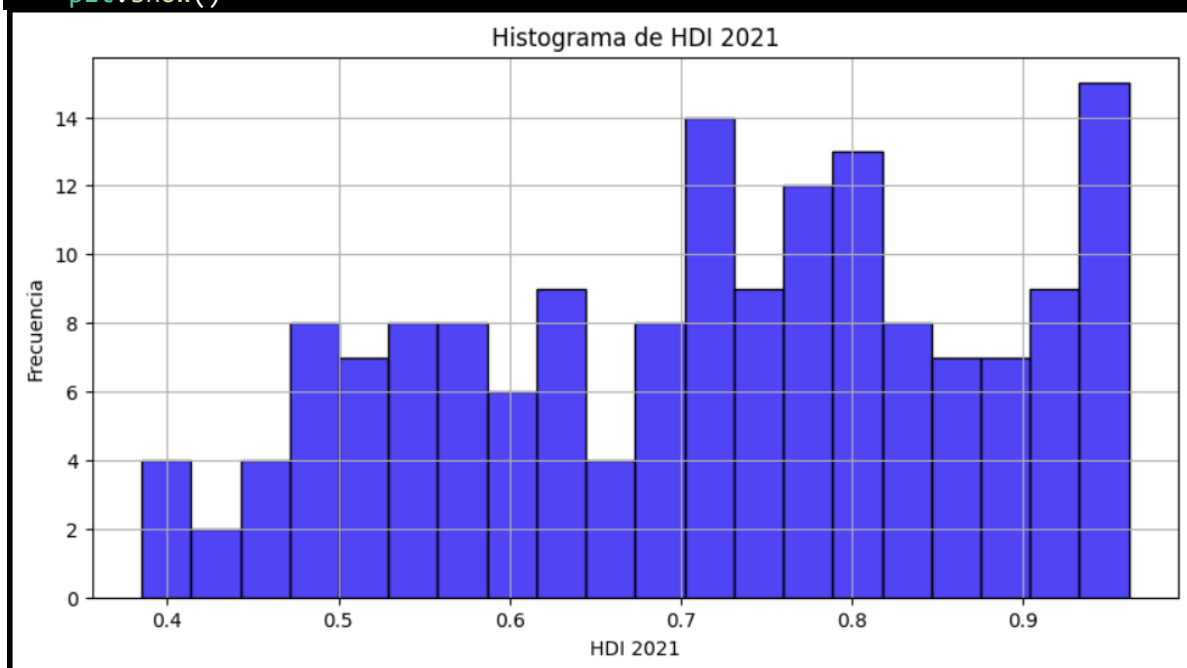
```
print("Ruta a usar:", file_path)

# Verificar si el archivo existe en la ruta especificada antes de intentar
cargarlo
if not os.path.exists(file_path):
    print(f"Error: No se encontró el archivo en la ruta especificada:
{file_path}")
else:
    # Cargar los datos desde el archivo Excel
    df = pd.read_excel(file_path)
    print("Archivo cargado exitosamente.")

# Mostrar las primeras filas del DataFrame para entender la estructura de los
datos
print("Primeras filas del DataFrame:")
print(df.head())

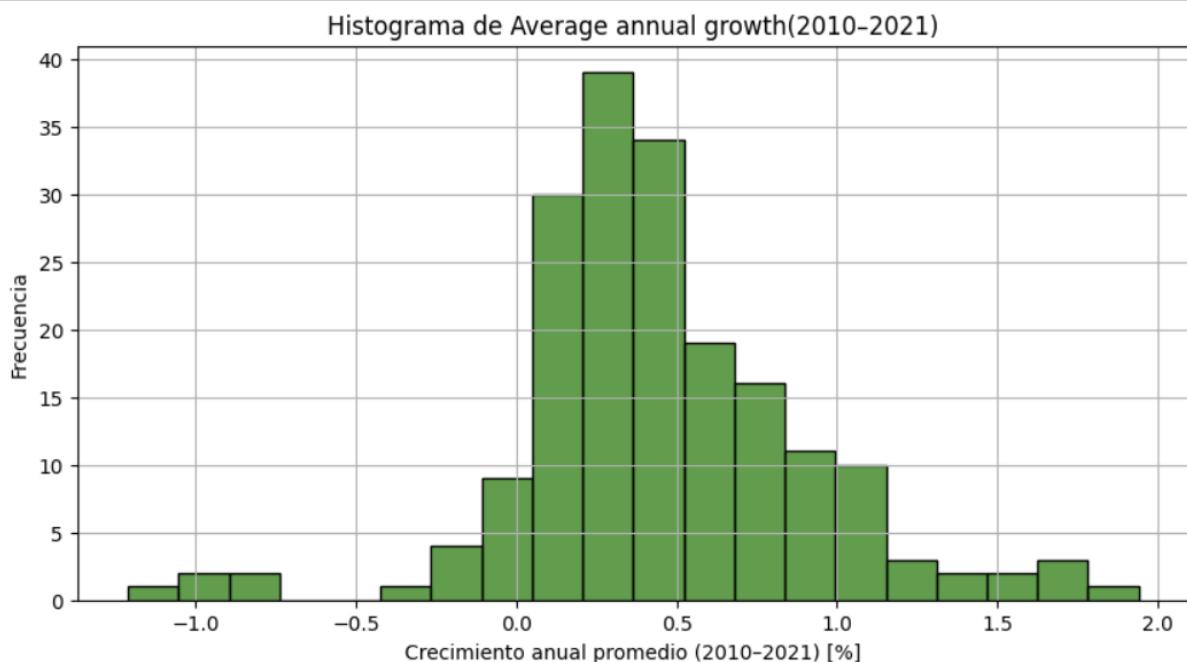
# Ejercicio 2: Análisis de las variables
# 2.1 Genera un histograma para la variable HDI 2021 y Average annual
growth(2010-2021)

# Histograma de HDI 2021
plt.figure(figsize=(10, 5))
sns.histplot(df['HDI 2021'].dropna(), kde=False, bins=20, color='blue')
plt.title('Histograma de HDI 2021')
plt.xlabel('HDI 2021')
plt.ylabel('Frecuencia')
plt.grid(True)
plt.show()
```



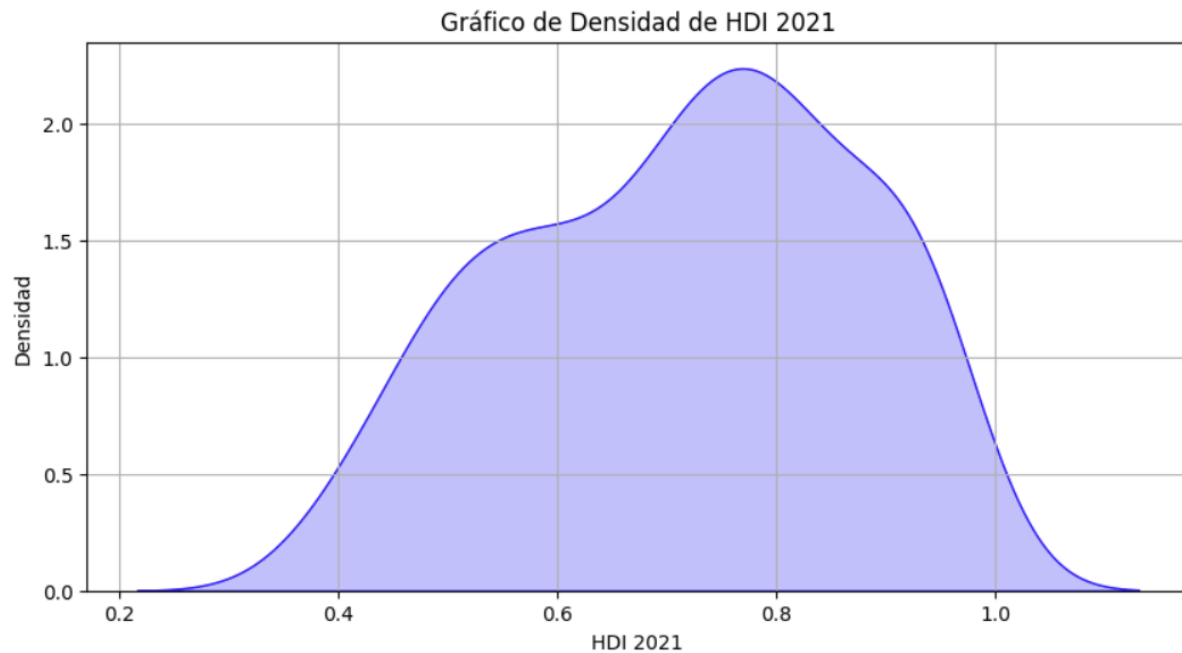
```
# Verificar los nombres de las columnas
print("Columnas en el DataFrame:", df.columns)

# Histograma de Average annual growth(2010-2021)
plt.figure(figsize=(10, 5))
sns.histplot(df['Average annual growth(2010-2021)'].dropna() * 100,
kde=False, bins=20, color='green')
plt.title('Histograma de Average annual growth(2010-2021)')
plt.xlabel('Crecimiento anual promedio (2010-2021) [%]')
plt.ylabel('Frecuencia')
plt.grid(True)
plt.show()
```



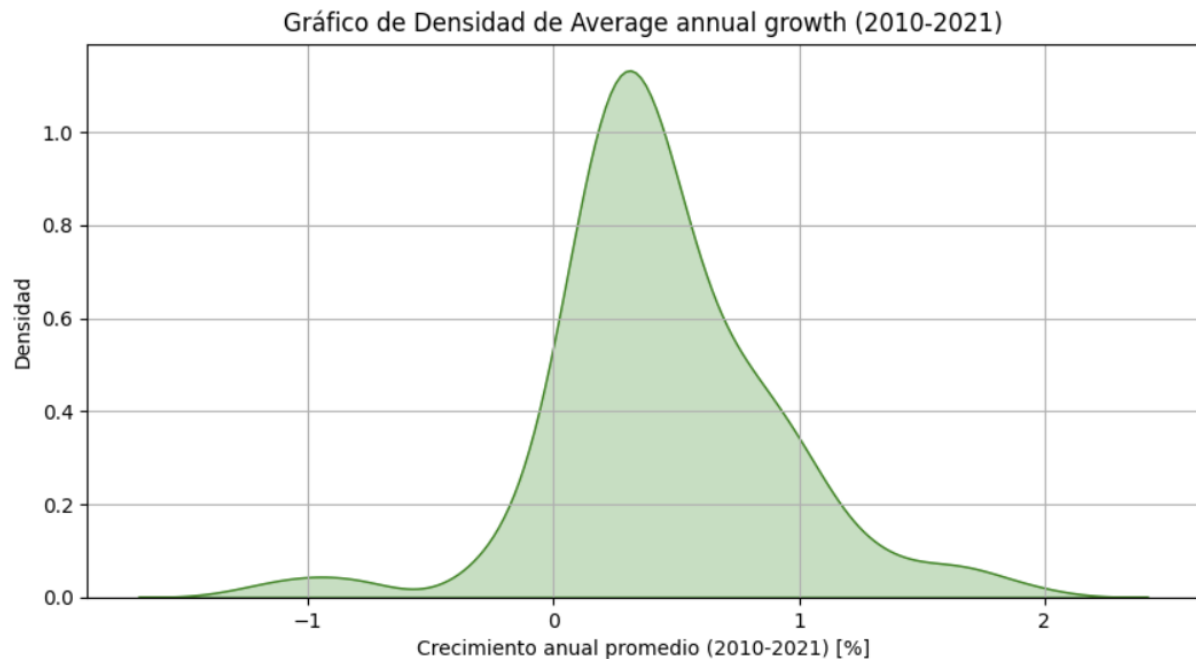
```
# Ejercicio 3: Genera un gráfico de densidad para las variables anteriores
```

```
# Gráfico de densidad de HDI 2021
plt.figure(figsize=(10, 5))
sns.kdeplot(df['HDI 2021'].dropna(), shade=True, color='blue')
plt.title('Gráfico de Densidad de HDI 2021')
plt.xlabel('HDI 2021')
plt.ylabel('Densidad')
plt.grid(True)
plt.show()
```



```
# Verificar los nombres de las columnas
print("Columnas en el DataFrame:", df.columns)

# Gráfico de densidad de Average annual growth(2010-2021)
plt.figure(figsize=(10, 5))
sns.kdeplot(df['Average annual growth(2010-2021)'].dropna() * 100,
shade=True, color='green')
plt.title('Gráfico de Densidad de Average annual growth (2010-2021)')
plt.xlabel('Crecimiento anual promedio (2010-2021) [%]')
plt.ylabel('Densidad')
plt.grid(True)
plt.show()
```



```
# Ejercicio 4: Cálculo de estadísticas descriptivas
def calcular_estadisticas(data):
    estadisticas = {
        'media': data.mean(),
        'mediana': data.median(),
        'desviación estándar': data.std(),
        'coeficiente de variación': data.std() / data.mean(),
        'rango intercuartil': data.quantile(0.75) - data.quantile(0.25),
        'primer cuartil': data.quantile(0.25),
        'segundo cuartil (mediana)': data.quantile(0.50),
        'tercer cuartil': data.quantile(0.75),
        'mínimo': data.min(),
        'máximo': data.max()
    }
    return pd.Series(estadisticas)

# Estadísticas para HDI 2021
print("\nEstadísticas descriptivas para HDI 2021:")
hdi_2021_estadisticas = calcular_estadisticas(df['HDI 2021'].dropna())
print(hdi_2021_estadisticas)

# Estadísticas para Average annual growth(2010-2021)
print("\nEstadísticas descriptivas para Average annual growth(2010-2021):")
```

```
average_growth_estadisticas = calcular_estadisticas(df['Average annual  
growth(2010-2021)'].dropna() * 100)  
print(average_growth_estadisticas)
```

#Estadísticas descriptivas para HDI 2021:

media	0.716099
mediana	0.730500
desviación estándar	0.156051
coeficiente de variación	0.217918
rango intercuartil	0.248750
primer cuartil	0.587750
segundo cuartil (mediana)	0.730500
tercer cuartil	0.836500
mínimo	0.385000
máximo	0.962000
dtype:	float64

#Estadísticas descriptivas para Average annual growth(2010-2021):

media	0.446349
mediana	0.390000
desviación estándar	0.458577
coeficiente de variación	1.027394
rango intercuartil	0.500000
primer cuartil	0.200000
segundo cuartil (mediana)	0.390000
tercer cuartil	0.700000
mínimo	-1.210000
máximo	1.940000
dtype:	float64

## Conclusiones:

### -HDI 2021:

- No hay valores atípicos (la media es muy similar a la mediana).
- Distribución simétrica con una ligera concentración alrededor de la media ( $M = 0.7161$ ,  $Mediana = 0.7305$ ,  $IQR = 0.2488$ ).
- Menor variabilidad en sus medidas ( $SD = 0.1561$ ,  $CV = 0.2179$ ).

### -Average Annual Growth (2010-2021):

- Presenta valores atípicos significativos (mínimo = -1.2100, máximo = 1.9400).
- Mayor variabilidad en sus medidas ( $SD = 0.4586$ ,  $CV = 1.0274$ ).
- Mayor dispersión y concentración de valores en los extremos ( $M = 0.4463$ ,  $Mediana = 0.3900$ ,  $IQR = 0.5000$ ).