Actividad - Estadística básica

```
* **Nombre:** Sebastian Flores Lemus
* **Matrícula:** A01709229
```

- **Nombre:** Sebastian Flores Lemus
- **Matrícula:** A01709229

Entregar: Archivo PDF de la actividad, así como el archivo .ipynb en tu repositorio. **Nota:** Recuerda habrá una penalización de **50** puntos si la actividad fue entregada fuera de la fecha límite.

Carga el conjunto de datos `bestsellers with categories.csv` (se encuentra en el repositorio de la clase) y realiza un análisis estadístico de las variables.

```
# Carga las librerías necesarias.
import numpy as np
import pandas as pd

import matplotlib.pyplot as plt
import seaborn as sns


# Carga el conjunto de datos al ambiente de Google Colab y muestra los primeros
# 6 renglones.

from google.colab import files

uploaded = files.upload()

for fn in uploaded.keys():
  print('User uploaded file "{name}" with length {length} bytes'.format(
      name=fn, length=len(uploaded[fn])))

# 6 renglones.
df = pd.read_csv('bestsellers with categories.csv')
df.head(6)
```

El conjunto de datos es una tabla que contiene el top 50 de los libros más vendidos por Amazon por año desde 2009 hasta 2019. Cada libro está clasificado como Ficción o No ficción.

Las variables que contiene son:

- **Name**: Nombre del libro.
- **Author**: Autor.
- **User Rating**: Calificación promedio que los usuarios asignaron al libro (1-5).
- **Reviews**: Número de reseñas.
- **Price**: Precio del libro.
- **Year**: Año de publicación.
- **Genre**: Género literario (ficción/no ficción).

```
# Crea una tabla resumen con los estadísticas generales de las variables
# numéricas.

df.describe()
```

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 550 entries, 0 to 549
Data columns (total 7 columns):
```

```
 #   Column       Non-Null Count  Dtype  
---  ------       --------------  -----  
 0   Name         550 non-null    object 
 1   Author       550 non-null    object 
 2   User Rating  550 non-null    float64
 3   Reviews      550 non-null    int64  
 4   Price        550 non-null    int64  
 5   Year         550 non-null    int64  
 6   Genre        550 non-null    object 
dtypes: float64(1), int64(3), object(3)
memory usage: 30.2+ KB
```

```
# ¿Cuál es el género con más publicaciones? Muéstralo en un gráfico.

# Tamaño de la imagen
fig = plt.figure(figsize=(6,4))

# Gráfico countplot para hacer barras con la frecuencia de cada genero
sns.countplot(data=df, x = 'Genre')

# Ejes y título. Colocamos la etiqueta correcta de acuerdo a la orientación.
plt.title('¿Cuál es el género con más publicaciones?')
plt.xlabel('Genre')
plt.ylabel('Frecuencia')
```

```
# ¿Cuántos libros del top 50 se publicaron por género en cada año? ¿Hay algún
# año donde hubo más libros de ficción en el top 50?. Muéstralo en un gráfico.

sns.histplot (data=df, x='Genre', hue='Year', bins=10, kde= True, color='B')

# si cambiamos 'kde' a 'False' nos queda igual un gráfico pero construido de otra manera dife
```

```
# ¿Cómo se distribuye la variable Review? Muéstra el histografa.

# Grafico con barras (y)

fig = plt.figure(figsize=(16, 14))

# Gráfico countplot para hacer barras con las reviews
sns.countplot(data=df, y = 'Reviews')

# Ejes y título. Colocamos la etiqueta correcta de acuerdo a la orientación.
plt.title('¿Cómo se distribuye la variable Review?')
plt.xlabel('Frecuencia')
plt.ylabel('Genre')
```

```
# ¿Cómo se distribuye la variable Review? Muéstra el histografa.

# histograma

sns.histplot(data=df, x='Reviews')
```

```
# Ahora muéstralo en un gráfico de caja y bigote.

# Tamaño de la imagen
fig = plt.figure(figsize=(7,5))
```

```
# Gráfico boxplot
sns.boxplot(data=df, x='Reviews')

# Ejes y título
plt.title('Distribucion de "reviews" (caja y bigote)')
```

```
# ¿Cómo se compara la evaluación del libro por género? ¿Qué genero es mejor
# evaluado por los lectores? Muéstralo en un solo gráfico de caja y bigote.

# Tamaño de la imagen
fig = plt.figure(figsize=(7,5))

# Gráfico boxplot
sns.boxplot(data=df, y='Genre', x='Reviews')

# Ejes y título
plt.title('¿Cómo se compara la evaluación del libro por género?')
```

```
# ¿Cuál es la relación entre el número de reseñas y precios? Muéstralo en un
# gráfico de dispersión.

# Tamaño de la imagen.
fig = plt.figure(figsize=(6, 4))

# Gráfico scatterplot.
sns.scatterplot(data=df, x ='Reviews', y='Price') #, hue='Price')

# Ejes y título. Colocamos la etiqueta correcta de acuerdo a la orientación.
plt.title('Relación entre el número de reseñas y precios')
plt.xlabel('Reviews')
plt.ylabel('Price')

# EL grid es opcional pero ayuda hacer mejores aproximaciones
plt.grid(True)

# pairplot es una versión más compleja y grande del gráfico de dispersión.
sns.pairplot(data=df)
```

```
# De la pregunta anterior, ¿influye algo el año de publicación? ¿Cuál es la
# relación entre el número de reseñar, el precio y el año de publicación?
# IMPORTANTE: Selecciona una paleta de colores adeacuada.
fig = plt.figure(figsize=(6, 4))

# Gráfico scatterplot.

sns.scatterplot(data=df, x ='Reviews', y='Price', hue='Year')
sns.color_palette(['red', '#33AAFF','#FFDD33', 'blue','black'])


# Ejes y título. Colocamos la etiqueta correcta de acuerdo a la orientación.
plt.title('Relación entre el número de reseñas, precio y año de publicación')
plt.xlabel('Reviews')
plt.ylabel('Price')
```

```
# Aquí hice otro dataframe ('d2') sin los valores de 'Year' para el siguiente ejercicio

df2=df.drop(columns=['Year'], axis=1)
df2.head(6)
```

```
# ¿Cuál es la correlación entre las variables numéricas? Muéstralo en un
# gráfico. La variable año, a pesar de ser numérica, la vamos a considerar como
# cualitativa, así que la eliminaremos del análisis.

# Utilice el segundo dataframe que no tiene 'Year' porque así se nos pide
iris_corr = df2.corr()

# Para graficar el mapa de calor usamos heatmap. No necesitamos especificar x ni y
sns.heatmap(data=iris_corr, annot=True)
```

```
# ¿Cuál es la correlación entre las variables numéricas? Muéstralo en un


# esta incluye 'year'
iris_corr = df.corr()
sns.heatmap(data=iris_corr,annot=True)
```

```
# otra forma de plasmarlo

sns.heatmap(data=iris_corr, vmin=-1, vmax=1, cmap = 'RdBu', annot=True, square = True)
```

¿Cuáles variables tiene una fuerte relación positiva entre sí y cuáles tienen una fuerte relación

** Escribe tu respuesta **: En la correlación nos encontramos que los valores estarán en un rango de -1 a 1. Mientras mas cercano sea el valor al numero 1 significa que hay una mayor correlación, mientras mas cercanos sean a -1 significa que la correlación es negativa o no hay fuerte correlación. Con la ayuda de: "annot=True" podemos poner el valor exacto y ver la correlación de las variables. Si eliminamos la variable 'year' del análisis como se nos pidió, nos damos cuenta que nos encontramos con valores negativos. Lo cual indica que la correlación no es del todo fuerte. 'User Rating' y 'Reviews' tiene una correlación de -0.0017 la cual no es para nada alta pero si es mas fuerte que la correlación entre 'Price' y 'User-rating'y que la de 'Reviews' y 'Price'. 'User rating' y 'price' tienen una correlación de -0.13 siendo esta la mas alejada del valor 1 y mas cercana a el valor -1, lo cual significa que tiene una fuerte relación negativa. 'Reviews' y 'Price' tienen una correlación de -0.11 la cual también se considera una correlación negativa.

```
# Haz una gráfica donde podemos comparar la relación entre las tres variables
# numéricas (User Rating, Reviews y Price) y que, además, podamos ver el efecto
# del libro. La variable año, a pesar de ser numérica, la vamos a considerar como
# cualitativa, así que la eliminaremos del análisis.

# Con pairplot podemos crear un gráfico mas grande y completo y podemos observar la dispersió
sns.pairplot(data=df2)
```

```
iris_corr = df2.corr()

# Para graficar el mapa de calor usamos heatmap. No necesitamos especificar x ni y
sns.heatmap(data=iris_corr, annot=True)
```