SEM309 – 10160 SEMINARIO DE PRÁCTICA EN CIENCIA DE DATOS

TRABAJO PRÁCTICO N° 2

ANÁLISIS EXPLORATORIO DE RESEÑAS DE VINOS

MARIANO NATIELLO (VLDC000632)

OCTUBRE 2024

1. ÍNDICE

2. CONTEXTO *2*

3. OBJETIVOS *2*

4. NOTA ACLARATORIA *2*

5. ACTIVIDADES REALIZADAS *2*

Carga del archivo 2

Análisis exploratorio de datos 3

Tratamiento de datos faltantes 7

Análisis de datos atípicos 9

6. REFERENCIA BIBLIOGRÁFICA *11*

2. CONTEXTO

Continuando con la información presentada en el marco del Trabajo Práctico N° 1, y considerando el dataset que contiene aproximadamente unas 130.000 reseñas de vinos obtenidas del sitio Wine Enthusiast, se realiza un análisis exploratorio de los datos, considerando aquellos datos faltantes y un análisis de los datos atípicos y su impacto en la toma de decisiones, según se detalla en el presente documento.

3. OBJETIVOS

Para este segundo trabajo práctico, se contempla realizar un análisis que permita comprender la estructura de los datos relacionados con la reseña de los vinos, identificar potenciales problemas y preparar los datos para un análisis más avanzado.

4. NOTA ACLARATORIA

Si bien se partió de la base de utilizar el dataset brindado para este trabajo práctico, una opción era utilizar solamente aquellos datos del dataset que se podían leer sin problema pero representaban solo el 5% de los datos. Una segunda opción era buscar programáticamente los diferentes casos, lo cual resultaba inviable debido a que requiere revisar los posibles errores entre el 95% de los datos restantes. Adicionalmente, la visualización de forma programática no es fácil desde ninguno de los puntos de vista.

Considerando lo anterior, se concluyó que la mejor opción era hacer la transformación de los datos utilizando una herramienta como Excel, dando como resultado el archivo “winemag-data-130k-v2.csv” utilizado para la realización de este trabajo práctico, el cual se comparte como referencia.

5. ACTIVIDADES REALIZADAS

A fin de brindar un orden de las actividades realizadas, se consideraron aquellas que fueron establecidas en la consigna del trabajo práctico. A continuación, se amplia con mayor detalle cada uno de los temas tratados:

Carga del archivo

1. Se utilizó Google Colab y Python para la carga del archivo “winemag-data-130k-v2.csv”, ubicado en la carpeta “home”.

```
# Importar librerías necesarias
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns

# Cargar el archivo de datos
datos = pd.read_csv('/home/winemag-data-130k-v2.csv', delimiter='\t')
```

Nota: se renombró la primera columna “ID”

2. El archivo se cargó correctamente, verificándose el número de filas, columnas, nombre de cada una de estas, tipos de datos presentes y por último, se obtuvo el encabezado del dataframe.

```
# Revisar el formato del archivo y verificar información de los datos
print("Cantidad de filas y columnas:", datos.shape)
print("Nombre de columnas:", datos.columns)
print("Tipos de datos:", datos.dtypes)
```

Ejecutando lo anterior, se obtiene la siguiente información:

```
Cantidad de filas y columnas: (129971, 14)
Nombre de columnas: Index(['ID', 'country', 'description', 'designation', 'points', 'price',
       'province', 'region_1', 'region_2', 'taster_name',
       'taster_twitter_handle', 'title', 'variety', 'winery'],
      dtype='object')
Tipos de datos: ID                         int64
country                   object
description               object
designation               object
points                   float64
price                    float64
province                  object
region_1                  object
region_2                  object
taster_name               object
taster_twitter_handle     object
title                     object
variety                   object
winery                    object
dtype: object
```

Análisis exploratorio de datos

De acuerdo a la información obtenida, mediante *print (datos.head),* se obtiene el siguiente resumen de las variables y sus características principales:

Variable	Descripción	Clasificación
ID	Identificador numérico	Numérica (1)
country	País de origen del vino	Categórica (3)
description	Notas de cata detalladas que un sommelier utilizaría para evaluar un vino.	Texto (3)
designation	Viñedo especifico dentro de la bodega	Categórica (3)
points	Puntuación en una escala del 1 a 100.	Numérica (1)
price	Precio de la botella de vino.	Numérica (2)
province	Provincia o estado de origen	Categórica (3)
region_1	Área vinícola especifica	Categórica (3)
region_2		Categórica (3)
taster_name	Información sobre el catador que realizó la reseña.	Texto (3)
taster_twitter_handle		Texto (3)
title	Título de la reseña.	Texto (3)
variety	Variedad de la uva	Categórica (3)
winery	Bodega que produjo el vino	Texto (3)

(1) Variable cuantitativa discreta

(2) Variable cuantitativa continua

(3) Variable cualitativa

De las variables detalladas, se realizó un análisis con las variables numéricas y categóricas dando como resultado el resumen estadístico detallado a continuación:

```
print(datos.describe())  # Variables numéricas
```

```
                ID         points         price
count  129971.000000  127071.000000  118213.00000
mean    64985.000000      88.439778      35.27222
std     37519.540256       3.033253      40.84197
min         0.000000      80.000000       4.00000
25%     32492.500000      86.000000      17.00000
50%     64985.000000      88.000000      25.00000
75%     97477.500000      91.000000      42.00000
max    129970.000000     100.000000    3300.00000
```

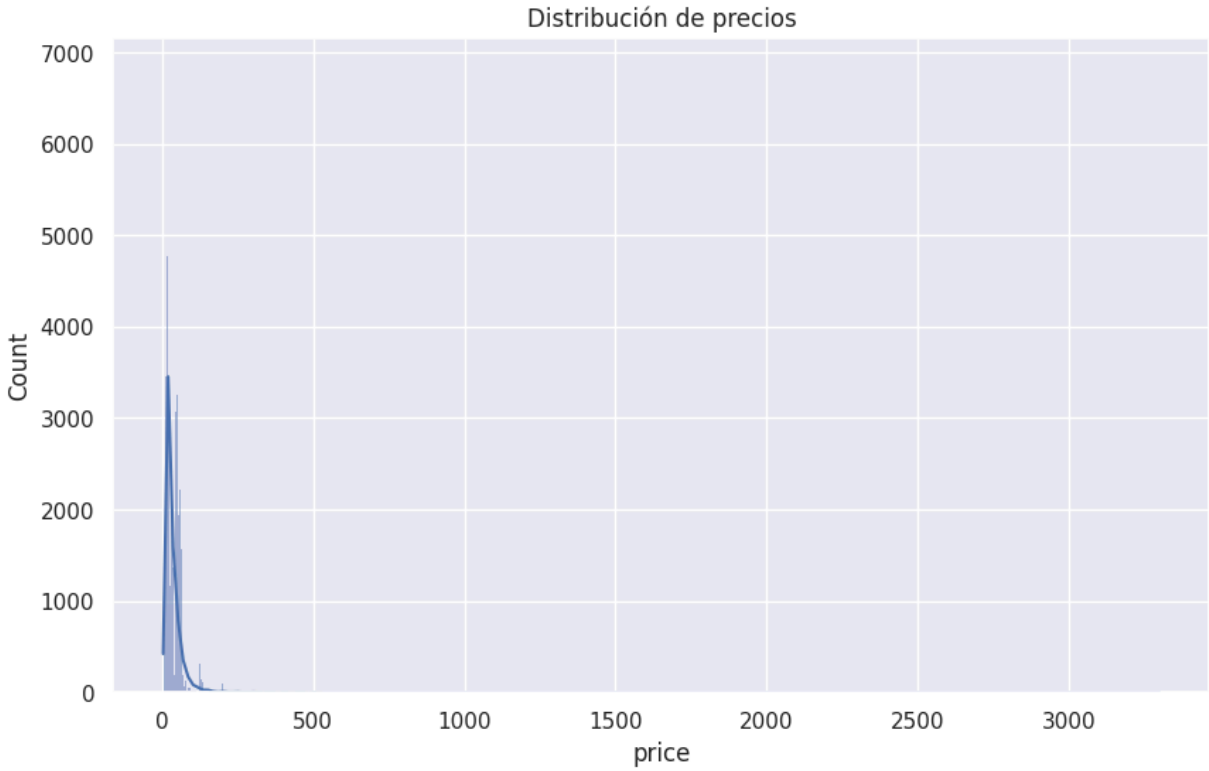
```
print(datos['country'].value_counts())  # Variable categórica
print("Total reg:", sum(datos['country'].value_counts()))
```

country	
US	54504
France	22093
Italy	19540
Spain	6645
Portugal	5691
Chile	4472
Argentina	3800
Austria	3345
Australia	2329
Germany	2165
New Zealand	1419
South Africa	1401
Israel	505
Greece	466
Canada	257
Hungary	146
Bulgaria	141
Romania	120
Uruguay	109
Turkey	90
Slovenia	87
Georgia	86
England	74
Croatia	73
Mexico	70
Moldova	59
Brazil	52
Lebanon	35
Morocco	28
Peru	16
Ukraine	14
Serbia	12
Czech Republic	12
Macedonia	12
Cyprus	11
India	9
Switzerland	7
Luxembourg	6
Bosnia and Herzegovina	2
Armenia	2
Slovakia	1
China	1
Egypt	1

Name: count, dtype: int64
Total: 129908

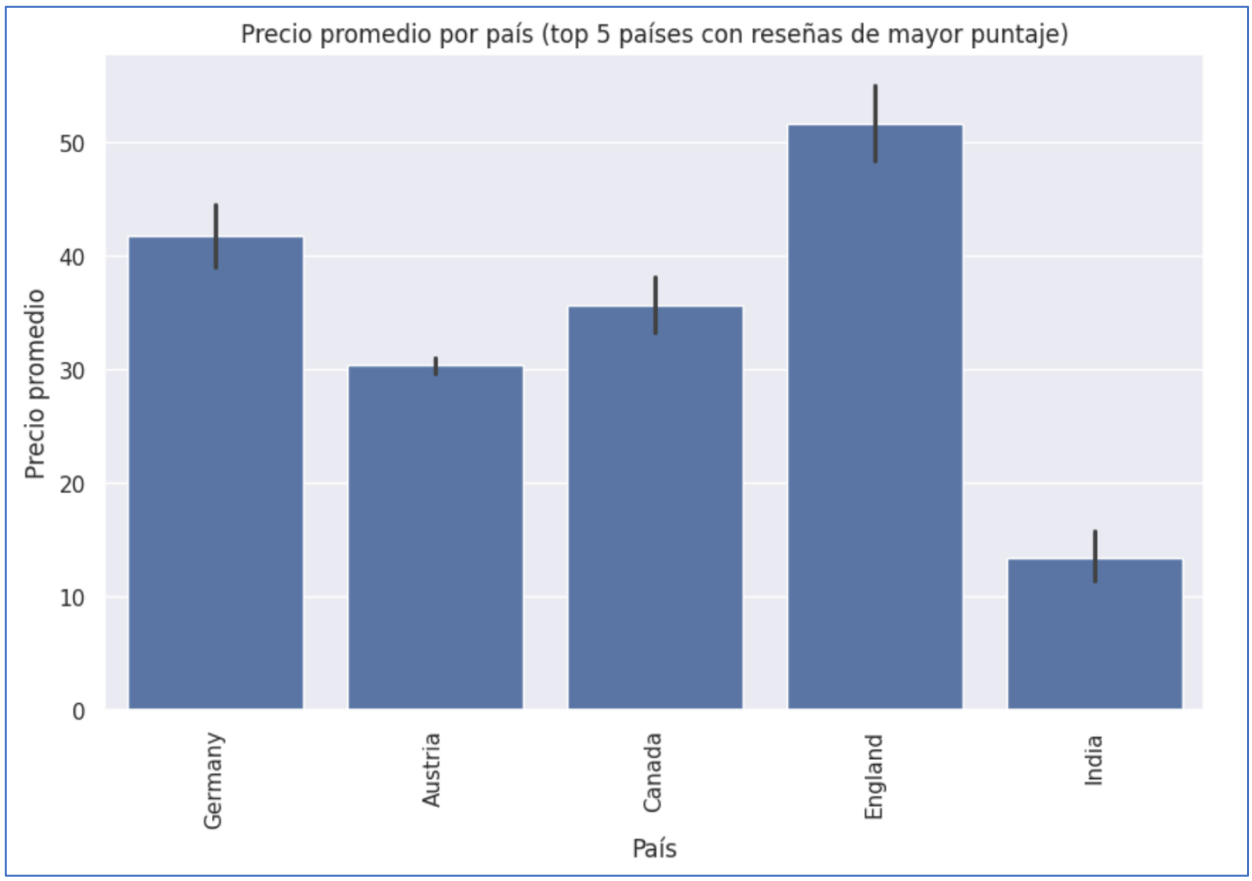
Para explorar la distribución de las variables numéricas y categorías, se utilizó el siguiente gráfico:

```
1 # Gráficos para distribución de variables
sns.set()
plt.figure(figsize=(10, 6))
sns.histplot(datos['price'], kde=True)
plt.title('Distribución de precios')
plt.show()
```




```
2 # Relación entre países y precio considerando los 5 países con reseñas de mayor puntaje.
top_5_paises = datos.groupby('country')['points'].mean().sort_values(ascending=False).head(5).index

10# Filtra los datos para solo incluir los 5 países seleccionados
df_filtrado = datos[datos['country'].isin(top_5_paises)]
# Crea un gráfico de barras que muestra el precio promedio por país
plt.figure(figsize=(10,6))
sns.barplot(x='country', y='price', data=df_filtrado)
plt.xlabel('País')
plt.ylabel('Precio promedio')
plt.title('Precio promedio por país (top 5 países con reseñas de mayor puntaje)')
plt.xticks(rotation=90)
plt.show()
```



Tratamiento de datos faltantes

```
# Identificar y manejar los datos faltantes en el conjunto de datos
Print("Cantidad de datos faltantes por columna:")
print(datos.isnull().sum())
```

```
Cantidad de datos faltantes por columna:
ID                          0
country                    63
description                 0
designation             39568
points                   2900
price                   11758
province                 2962
region_1                23581
region_2                80475
taster_name             28744
taster_twitter_handle   33705
title                    2900
variety                  2957
winery                   2957
```

Como puede apreciarse, existen varias columnas sin un valor asignado.

En cuanto al tratamiento, a nivel de las variables cualitativas, resulta ser un desafío ya que la naturaleza de los datos requiere de un enfoque diferente si comparamos con las variables cuantitativas. De igual manera, podríamos abordar el problema mediante las siguientes estrategias:

1. Eliminar filas con datos faltantes: siempre que el porcentaje de datos faltantes sea relativamente bajo.
2. Eliminar columnas con datos faltantes: si una columna contiene gran cantidad de datos faltantes, como sucede en el atributo region_2 cuyos valores nulos representan el 62% del set de datos, podríamos considerar eliminar dicha columna siempre que no resulte crítica para nuestro análisis.
3. Crear una categoría “No especificado” y asignarlo en aquellos casos que no exista un valor.
4. Imputar valores faltantes utilizando la moda de la variable cualitativa. Desde ya que aplicar esta estrategia dependerá de la variable de nuestro conjunto de datos, ya que como fue comentado anteriormente, asignar un valor calculado para la moda podría no ser correcto (por ejemplo a nivel de “province”, “region_1”, “region_2”) el valor podría no reflejar la realidad.
5. Entrenar un modelo de clasificación utilizando los datos completos para predecir aquellos valores faltantes.

Por otra parte y a nivel de las variables cuantitativas, si bien aplican las primeras dos estrategias definidas anteriormente, se podría considerar adicionalmente:

a. Imputar datos faltantes con la media de las variables, teniendo en cuenta que puede ser sensible a valores atípicos.
b. Imputar datos faltantes con la mediana de las variables. En este caso, por ejemplo con el precio (price), tal como puede apreciarse en el código que se comparte adjunto, creamos un dataframe, recorremos el mismo validando si no existe un valor asignado a la columna “price” y en tal caso, asignamos el valor de la mediana.

c. Imputar basado en otros valores utilizando regresión para predecir valores faltantes, teniendo en cuanta otras variables.

d. Imputar basado en otros valores utilizando la técnica de KNN (K-Nearest Neighbors)

Ejemplo:

```
# Imputar los valores faltantes para el precio usando la mediana
# Crear un nuevo dataframe
df = pd.DataFrame(datos)
# Calcular la mediana de la columna 'price'
mediana_price = df['price'].median()

# Recorremos el nuevo dataframe consultando si el valor es nulo
for row_index, row in df.iterrows():
    if pd.isnull(row['price']):
        # esto es a modo de ejemplo para capturar que efectivamente tenía valor null
        print(f'antes de asignar: {df.at[row_index, "price"]}')
        # Asignamos la mediana al valor null
        df.at[row_index, 'price'] = mediana_price
         # esto es a modo de ejemplo para capturar que luego ya tiene el valor (mediana)
        print(f'despues de asignar: {df.at[row_index, "price"]}')
    else:
        print(f'El valor en la fila {row_index} es: {row["price"]}')

# Mostrar campo ID y price (solo las primeras 1000 líneas para verificar)
print(datos[['ID', 'price']].head(1000))
```

Análisis de datos atípicos

En una primera instancia, se detectaron datos atípicos en el precio (price), cuya información se detalla a continuación:

```
# Detectar posibles valores atípicos en las variables numéricas relevantes
print("Valores atípicos en la variable 'price':")
print(datos['price'].describe())
```

```
Valores atípicos en la variable 'price':
count    4842.000000
mean       35.003717
std        51.617361
min         4.000000
25%        17.000000
50%        25.000000
75%        40.000000
max      1900.000000
Name: price, dtype: float64
```

La media obtenida significa que en promedio, los precios son de 35 pero si consideramos la desviación estándar alta, la media podría estar sesgada por valores extremos (valor máximo).

El precio mínimo podría ser un precio tratado por ejemplo como parte de una oferta, pero si consideramos el valor máximo, es evidente la diferencia que existe entre ambos precios. Sin embargo, considerando el producto podría darse el caso que el valor sea correcto.

El caso mas evidente se presenta en el valor máximo (1900), siendo un valor atípico a simple vista ya que se encuentra por encima del tercer cuartil.

Al momento de considerar el tratamiento de estos datos atípicos, podríamos plantear las siguientes acciones:

1. Analizar por ejemplo si el valor máximo (1900) se trata de un error de entrada de los datos o bien, podría tratarse de un vino muy caro. En ambos opciones, es necesario considerar el contexto.
2. Analizar que un 25% de los precios son menores o iguales a 17%, lo cual nos da a entender que una gran cantidad de vinos tiene precios bajos.
3. Analizar la distribución, considerando la mediana en 25 (la mitad de los precios son iguales o están por debajo de este valor).
4. Analizar que un 75% de los precios es menor o igual a 40, lo que podríamos interpretar que la gran mayoría de los precios están en el rango bajo-medio.

Ahora bien, estos valores atípicos como es el caso del precio en 1900 cuando la mayoría de los precios están muy por debajo de ese valor podría tener un impacto más que significativo en nuestro análisis de datos, considerando el efecto por ejemplo en la media al ser sensible a valores extremos.

En otras palabras, si basamos nuestro análisis en la media para tomar decisiones o entender el comportamiento atípico de precios, podríamos tener una “imagen” incorrecta ya que no estaríamos considerando el valor medio de los productos.

Similar escenario se daría en el impacto de la desviación estándar en donde al momento de medir la dispersión de los precios alrededor de la media, es muy sensible a valores atípicos.

En conclusión, los valores atípicos mencionados pueden distorsionar gráficos que utilicemos para mostrar la información. Imaginemos un histograma de precios, considerando el valor atípico de 1900, puede ocasionar que el resto de los datos (entre 4 y 75) se agrupen en un rango muy estrecho, dificultando enormemente su interpretación real de los datos.

6. REFERENCIA BIBLIOGRÁFICA

Enunciado del Trabajo Práctico N° 2 de la materia Seminario de práctica en Ciencia de Datos de la carrera Licenciatura en Ciencias de Datos.

Material de la materia Introducción en Ciencia de Datos de la carrera Licenciatura en Ciencias de Datos.

Utilización de Google Colab (https://colab.research.google.com)

Utilización de Microsoft Excel.