ANÁLISIS DE CUPIDO

AUTORES/AS
Santiago Ramos Grateron ✉
Karen Lorena Félix Zarama ✉
Julian Santiago Florez Castañeda ✉
Natalia Andrea Granados Vallejos ✉

AFILIACIÓN
Universidad EAN
Universidad EAN
Universidad EAN
Universidad EAN

1. INTRODUCCIÓN

El análisis de los perfiles de OkCupid permite identificar patrones de comportamiento y características comunes entre las personas que buscan una relación en línea. A partir de esta exploración se busca comprender mejor cómo influyen variables como el género, la edad, los hábitos y la orientación sexual en la forma en que los usuarios se presentan y participan dentro de la plataforma. En este sentido, resulta relevante examinar la proporción entre hombres y mujeres, los hábitos nocivos más frecuentes, los rangos de edad con mayor presencia, la atención prestada al tipo de cuerpo, así como la influencia de la orientación sexual y la edad en la actividad general. Del mismo modo, se indagará en los empleos más representativos, con el fin de construir un panorama integral sobre los perfiles de quienes utilizan este espacio digital para conectar con otros.

OBJETIVO PRINCIPAL

Analizar los perfiles de OkCupid, para encontrar patrones de comportamiento en las personas interesadas en una relación.

PREGUNTAS DE INVESTIGACIÓN:

- ¿Qué proporción de hombres y mujeres se registra en la plataforma?
- ¿Cuáles son los hábitos nocivos (alcohol, drogas, cigarrillos) más comunes entre los usuarios?
- ¿Cuál es el rango de edad con mayor participación en la búsqueda de pareja?
- ¿Existe un género que preste más atención al tipo de cuerpo en los perfiles?
- ¿Cómo influyen la orientación sexual y la edad en la actividad general de los usuarios?
- ¿Qué empleos estan mas presentes?

2. BÚSQUEDA Y Y OBTENCIÓN

La información utilizada en este análisis proviene de una base de datos pública disponible en la plataforma Kaggle, bajo el nombre OkCupid Data for Introductory Statistics and Data Science Courses, elaborada por Albert Kim y Adriana Escobedo-Land en el año 2015. Esta fuente contiene aproximadamente 60,000 perfiles antiguos de usuarios de la aplicación OkCupid.

El conjunto de datos incluye 31 variables, que pueden clasificarse de la siguiente manera:

- Numéricas: edad (age), altura (height), ingresos (income).
- Categóricas: sexo (sex), estado civil (status), orientación sexual (orientation), tipo de cuerpo (body_type), dieta (diet), consumo de alcohol (drinks), consumo de drogas (drugs), nivel educativo (education), etnia (ethnicity), ocupación (job), hijos (offspring), mascotas (pets), religión (religion), signo zodiacal (sign), idiomas hablados (speaks), ubicación (location).
- Temporales: última conexión (last_online).
- Textuales: ensayos o descripciones abiertas (essay0 – essay9).

La elección de esta base de datos se justifica en que ofrece información amplia y diversa sobre los usuarios, lo que permite responder las preguntas de investigación planteadas. En particular, posibilita:

- Determinar proporciones de género y rangos de edad predominantes en la plataforma.
- Analizar hábitos nocivos como el consumo de alcohol, tabaco y drogas.
- Estudiar cómo influyen factores sociodemográficos (educación, empleo, orientación sexual) en la participación de los usuarios.

Dado el tamaño de la muestra y la variedad de las variables, los datos constituyen una base sólida y representativa para la extracción de patrones de comportamiento en personas interesadas en una relación.

```
# Importar librerias necesarias
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
import numpy as np
import plotly.express as px
import plotly.graph_objects as go
from sklearn.preprocessing import StandardScaler
from sklearn.linear_model import LogisticRegression
from sklearn.preprocessing import LabelEncoder
from sklearn.model_selection import train_test_split
from sklearn.metrics import confusion_matrix, classification_report, accuracy
import plotly.io as pio
from imblearn.over_sampling import RandomOverSampler
from sklearn.metrics import confusion_matrix, ConfusionMatrixDisplay
from IPython.display import display, Markdown
pio.renderers.default = "notebook"
pd.set_option('future.no_silent_downcasting', True)
```

3. PRIMER ACERCAMIENTO

RESUMEN DESCRIPTIVO DE LOS DATOS

El dataset corresponde a perfiles de usuarios de la plataforma OkCupid, con información demográfica, hábitos, creencias, ubicación y fragmentos de texto autodescriptivo.

Número de observaciones (filas): cada fila representa un usuario.

Número de variables (columnas): 31 en total, que combinan datos numéricos, categóricos, ordinales y texto libre.

Existen valores faltantes (NaN) en varias variables, como drugs, diet, ethnicity, essays, entre otras.

Algunas variables tienen un orden implícito (drinks, drugs, education, smokes), mientras que otras son puramente categóricas (sex, orientation, body_type).

Los ensayos (essay0 – essay9) son variables de texto que requieren técnicas de procesamiento de lenguaje natural (NLP) para extraer patrones.

VARIABLE	TIPO DE VARIABLE	EJEMPLO DE VALOR
age	Numérica (entera, cuantitativa continua)	22
status	Categórica nominal	single
sex	Categórica nominal	m
orientation	Categórica nominal	straight
body_type	Categórica nominal	a little extra
diet	Categórica nominal	strictly anything
drinks	Categórica ordinal	socially
drugs	Categórica ordinal	never

VARIABLE	TIPO DE VARIABLE	EJEMPLO DE VALOR
education	Categórica ordinal	working on college/university
ethnicity	Categórica nominal	asian, white
height	Numérica (continua, en pulgadas)	75.0
income	Numérica discreta (−1 = no respuesta)	30000
job	Categórica nominal	transportation
last_online	Fecha y hora (variable temporal)	2012-06-28-20-30
location	Texto estructurado (categoría geográfica)	south san francisco, california
offspring	Categórica nominal	doesn’t have kids, but might want them
pets	Categórica nominal	likes dogs and likes cats
religion	Categórica nominal	agnosticism and very serious about it
sign	Categórica nominal	gemini
smokes	Categórica ordinal	sometimes

VARIABLE	TIPO DE VARIABLE	EJEMPLO DE VALOR
speaks	Texto estructurado (lista de idiomas)	english
essay0	Texto libre	about me: i would love to think that i was some kind of intellectual…
essay1	Texto libre	currently working as an international agent for a freight forwarding company…
essay2	Texto libre	making people laugh. ranting about a good salting…
essay3	Texto libre	the way i look. i am a six foot half asian, half caucasian mutt…
essay4	Texto libre	books: absurdistan, the republic, of mice and men…
essay5	Texto libre	food. water. cell phone. shelter.
essay6	Texto libre	duality and humorous things
essay7	Texto libre	trying to find someone to hang out with…
essay8	Texto libre	i am new to california and looking for someone to whisper my secrets to…
essay9	Texto libre	you want to be swept off your feet! you are tired of the norm…

CARGA Y PRIMERAS FILAS DEL DATASET

```
# Carga de datos
df = pd.read_csv('https://media.githubusercontent.com/media/Julian-Florez/Dat
pd.options.display.max_columns = None
pd.options.display.max_rows = None
df.head()
```

	AGE	STATUS	SEX	ORIENTATION	BODY_TYPE	DIET	DRINKS	DRUGS
0	22	single	m	straight	a little extra	strictly anything	socially	never
1	35	single	m	straight	average	mostly other	often	sometime
2	38	available	m	straight	thin	anything	socially	NaN
3	23	single	m	straight	thin	vegetarian	socially	NaN

	AGE	STATUS	SEX	ORIENTATION	BODY_TYPE	DIET	DRINKS	DRUGS
4	29	single	m	straight	athletic	NaN	socially	never

REVISAR ESTRUCTURA Y TIPOS DE DATOS

```
display(Markdown("### Estadísticos descriptivos\n\n" + df.describe().T.rename
```

ESTADÍSTICOS DESCRIPTIVOS

VARIABLE	COUNT	MEAN	STD	MIN	25%	50%	75%	MAX
age	59946	32.3403	9.45278	18	26	30	37	110
height	59943	68.2953	3.9948	1	66	68	71	95
income	59946	20033.2	97346.2	-1	-1	-1	-1	1e+06

```
display(Markdown("### Columnas del DataFrame\n\n" + "\n".join(f"- {c}" for c
```

COLUMNAS DEL DATAFRAME

- age
- status
- sex
- orientation
- body_type
- diet
- drinks
- drugs
- education
- ethnicity
- height
- income
- job
- last_online

- location
- offspring
- pets
- religion
- sign
- smokes
- speaks
- essay0
- essay1
- essay2
- essay3
- essay4
- essay5
- essay6
- essay7
- essay8
- essay9

```
display(Markdown(f"**Filas:** `{df.shape[0]}`  \n**Columnas:** `{df.shape[1]}`
```

Filas: `59946`
Columnas: `31`

Estructura de los datos (info()/str())

- age → int64 (numérica).
- status, sex, orientation, body_type, diet, drinks, drugs, education, ethnicity, job, offspring, pets, religion, sign, smokes, speaks → object (categóricas).
- height, income → float64/int64 (numéricas).
- last_online → datetime (fecha y hora).
- location → object (texto estructurado).
- essay0 – essay9 → object (texto libre).

Variables útiles para responder las preguntas de investigación

1. ¿Qué proporción de hombres y mujeres se registra en la plataforma?

- Variable: sex.

2. ¿Cuáles son los hábitos nocivos (alcohol, drogas, cigarrillos) más comunes?

- Variables: drinks, drugs, smokes.

3. ¿Cuál es el rango de edad con mayor participación?

- Variable: age.

4. ¿Existe un género que preste más atención al tipo de cuerpo en los perfiles?

- Variables: sex, body_type (posible cruce con los textos essays si se quiere profundizar).

5. ¿Cómo influyen la orientación sexual y la edad en la actividad general de los usuarios?

- Variables: orientation, age, last_online.

6. ¿Qué empleos están más presentes?

- Variable: job.

4. LIMPIEZA Y PREPARACIÓN DE LOS DATOS

DETECCIÓN Y MANEJO DE DUPLICADOS

```
print(df.duplicated().sum())
```

```
0
```

No hay duplicados en el dataframe

MANEJO DE DATOS FALTANTES

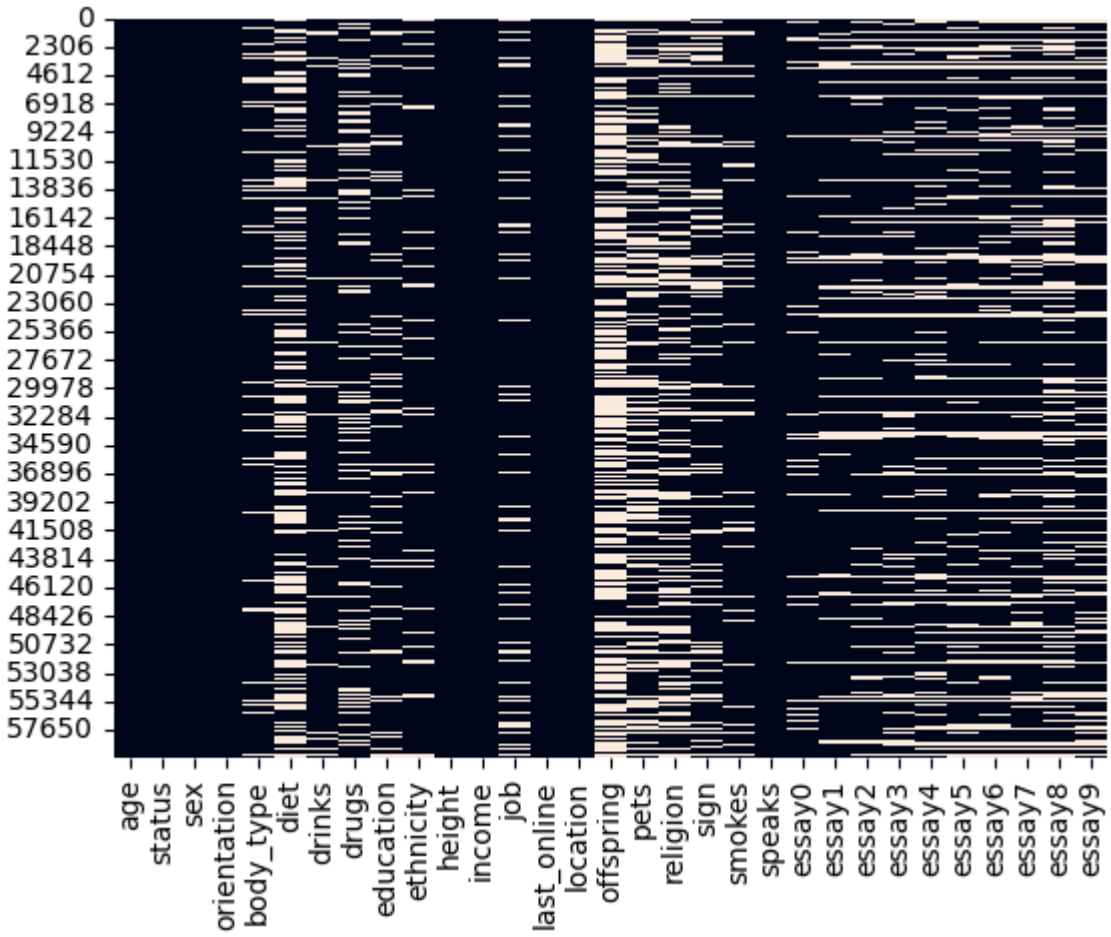
```
display(Markdown("### Valores nulos por columna\n\n" + df.isnull().sum().to_t
sns.heatmap(df.isnull(), cbar=False)
plt.show()
```

VALORES NULOS POR COLUMNA

COLUMNA	NULOS
age	0
status	0
sex	0
orientation	0
body_type	5296
diet	24395

COLUMNA	NULOS
drinks	2985
drugs	14080
education	6628
ethnicity	5680
height	3
income	0
job	8198
last_online	0
location	0
offspring	35561
pets	19921
religion	20226
sign	11056
smokes	5512
speaks	50
essay0	5488
essay1	7572
essay2	9638
essay3	11476
essay4	10537
essay5	10850
essay6	13771
essay7	12451

COLUMNA	NULOS
essay8	19225
essay9	12603



se identificaron las columnas con valores nulos y su distribución.

ESTRATEGIAS DE LIMPIEZA APLICABLES:

- Eliminar columnas con demasiados valores nulos y que no sean críticas (ejemplo: algunos essays, income).
- Imputar valores en columnas relevantes:
  - Numéricas → media, mediana o modelos de imputación (KNN, regresión).
  - Categóricas → moda (valor más frecuente).
  - Fechas (last_online) → conversión a datetime, valores nulos pueden imputarse como “desconocido” o eliminarse si son pocos.

```
df['age'] = df['age'].fillna(df['age'].median())       # imputar mediana en e
df['drinks'] = df['drinks'].fillna(df['drinks'].mode()[0])  # imputar moda en
```

```
df = df[df['age'] < 80]
```

```
# Crear una copia del df original para evitar su alteración en el modelado
data = df.copy()
data = data[["drinks", "drugs", "age", "sex", "smokes"]].dropna()
data["drinks"] = data["drinks"].replace({"not at all": 0, "rarely": 1, "socia
data["drugs"] = data["drugs"].replace({"never": 0, "sometimes": 1, "often": 1
data["smokes"] = data["smokes"].replace({"no": 0, "sometimes": 1, "when drink
data["sex"] = data["sex"].replace({"m": 0, "f": 1}).astype(int)

# Reemplazar varios valores
df["sex"] = df["sex"].replace({"m": "Masculino", "f": "Femenino"})
df["orientation"] = df["orientation"].replace({"straight": "Heterosexual", "g
df["job"] = df["job"].replace({"transportation": "Transporte","hospitality /
df["drugs"] = df["drugs"].replace({"never": "Nunca", "sometimes": "A veces",
df["drinks"] = df["drinks"].replace({"not at all": "Nunca", "rarely": "Rarame
df["smokes"] = df["smokes"].replace({"no": "No", "sometimes": "A veces", "whe
df["education"] = df["education"].replace({"working on college/university": '
df["body_type"] = df["body_type"].replace({"a little extra": "Un poco de sobr
df["status"] = df["status"].replace({"single": "Soltero/a","available": "Disp
```

5. NORMALIZACIÓN O ESTANDARIZACIÓN DE DATOS

Para este análisis no fue necesario realizar una normalización, ya que la mayoría de los datos eran categóricos y no requerían ningún tipo de transformación previa. En cuanto a los datos numéricos, el único utilizado fue la variable Edad, la cual tampoco necesitó ser normalizada debido a que su escala y rango eran adecuados para los métodos aplicados. Por lo tanto, los datos se mantuvieron en su formato original, lo que facilitó el procesamiento y evitó posibles distorsiones en los resultados.

6. ANÁLISIS EXPLORATORIO BÁSICO (EDA)

```
# Medidas para 'age'
age_stats = pd.DataFrame({
    "Media": [df['age'].mean()],
    "Mediana": [df['age'].median()],
    "Moda": [df['age'].mode()[0]]
})
display(Markdown("### Medidas para 'age'\n" + age_stats.to_markdown(index=Fal

# Moda para variables categóricas
categorical_cols = ['sex', 'body_type', 'drinks', 'drugs', 'smokes', 'educati
modes = []
for col in categorical_cols:
    if col in df.columns:
        mode_val = df[col].mode()
        if not mode_val.empty:
```

```
            modes.append({"Variable": col, "Moda": mode_val[0]})
        else:
            modes.append({"Variable": col, "Moda": "No disponible"})

modes_df = pd.DataFrame(modes)
display(Markdown("### Moda para variables categóricas\n" + modes_df.to_markdo
```

MEDIDAS PARA ‘AGE’

MEDIA	MEDIANA	MODA
32.3377	30	26

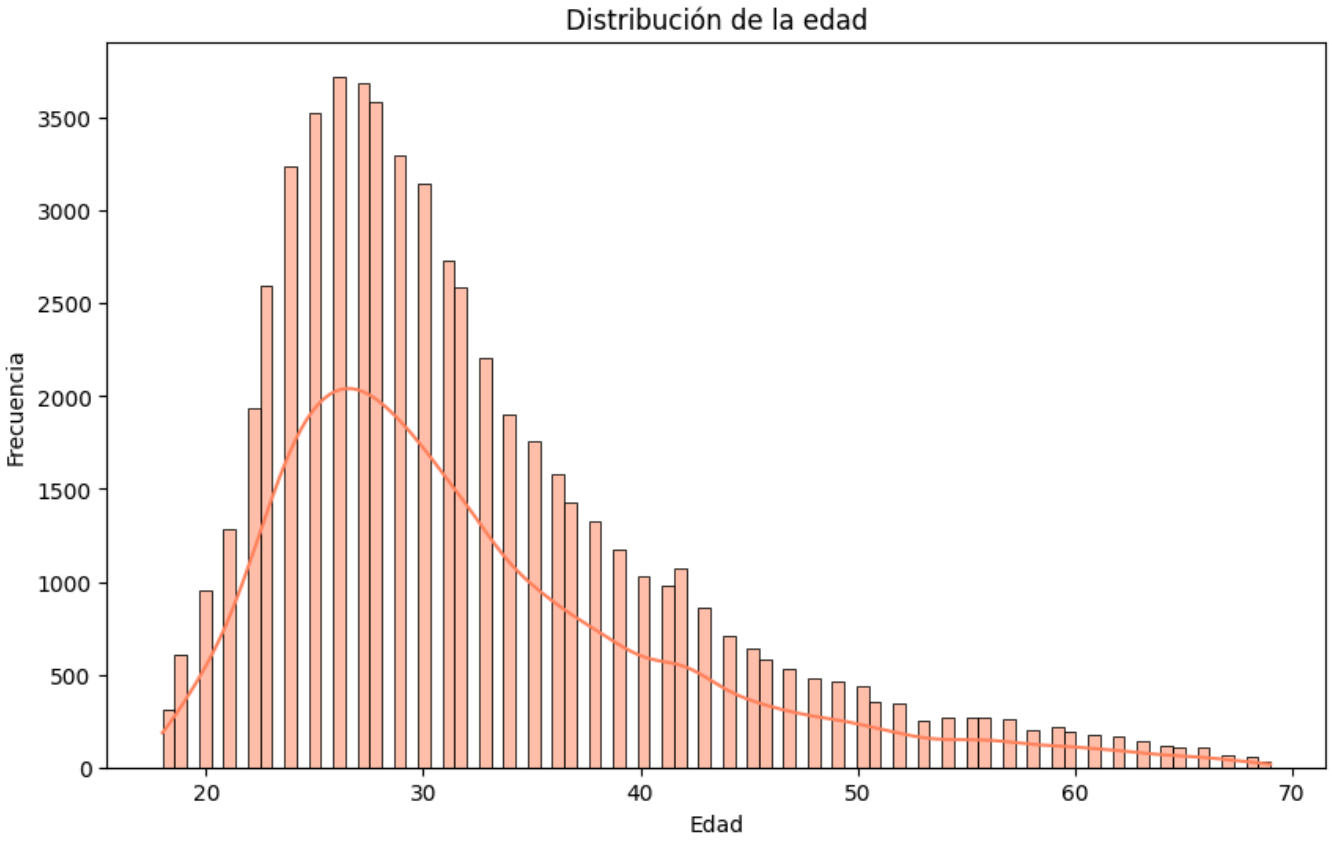
MODA PARA VARIABLES CATEGÓRICAS

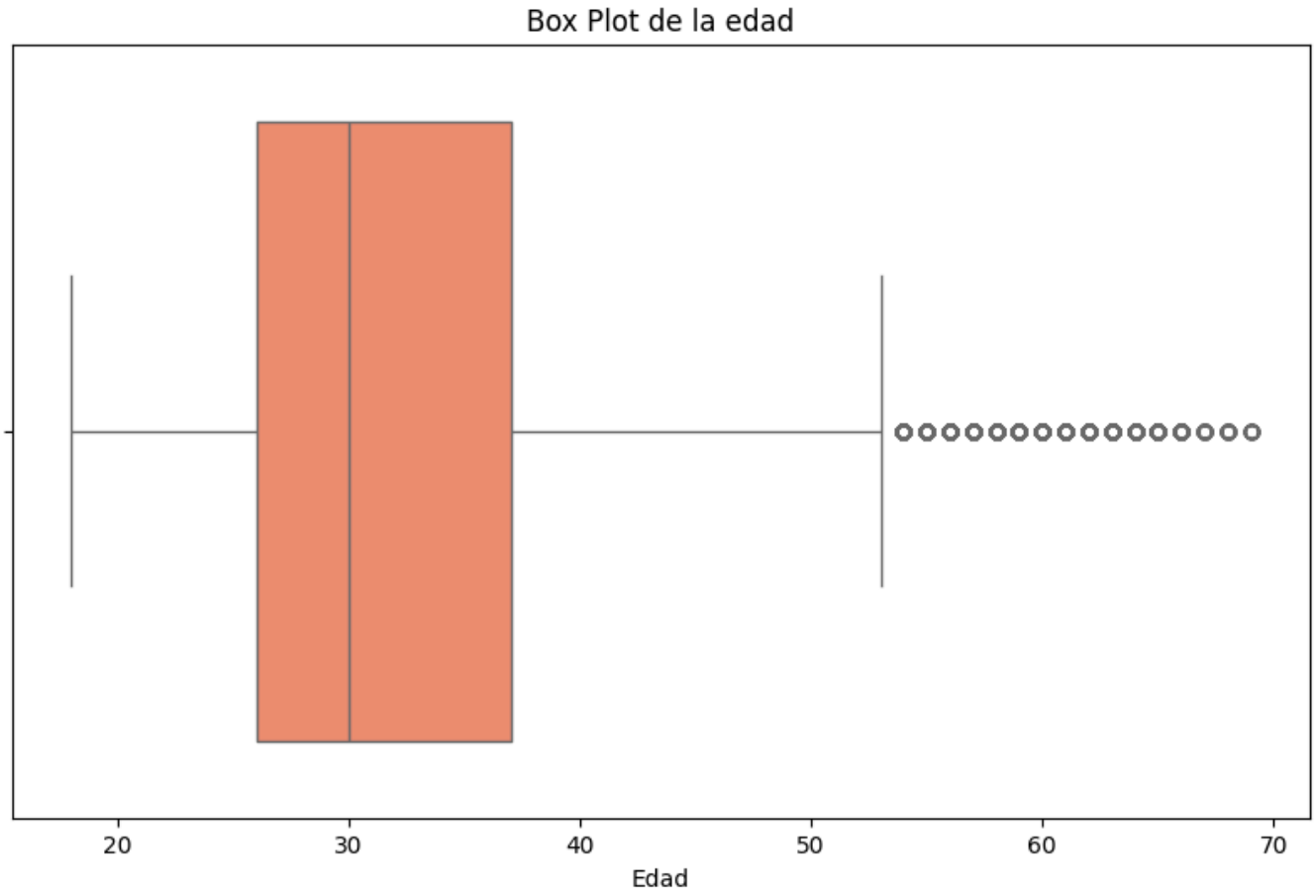
VARIABLE	MODA
sex	Masculino
body_type	Promedio
drinks	Socialmente
drugs	Nunca
smokes	No
education	Graduado de universidad
job	Otro
orientation	Heterosexual
status	Soltero/a

Los estadísticos descriptivos para la variable numérica ‘age’ indican que la media se sitúa alrededor de 32.3 años, con una mediana de 30 y una moda de 26, lo que sugiere una distribución ligeramente sesgada hacia edades jóvenes adultas. En cuanto a las variables categóricas, se observa que la categoría más frecuente para el sexo es masculino, y en cuanto a hábitos, la mayoría se identifica como consumidores sociales de bebidas alcohólicas, no fumadores y que nunca han consumido drogas. En términos educativos, la mayoría de los individuos han alcanzado un nivel de graduación universitaria. Además, la mayoría se identifica con orientaciones heterosexuales y un estado civil soltero, con una distribución diversa en las ocupaciones. Estos resultados proporcionan un perfil general representativo de la muestra, que puede servir de base para análisis posteriores o para la formulación de hipótesis específicas.

HISTOGRAMA Y BOXPLOT DE LA EDAD

```
# Histograma de edad
plt.figure(figsize=(10, 6))
sns.histplot(df['age'].dropna(), kde=True, color='#ff825e')
plt.title('Distribución de la edad')
plt.xlabel('Edad')
plt.ylabel('Frecuencia')
plt.show()

# Box plot de edad
plt.figure(figsize=(10, 6))
sns.boxplot(x=df['age'].dropna(), color='#ff825e')
plt.title('Box Plot de la edad')
plt.xlabel('Edad')
plt.show()
```



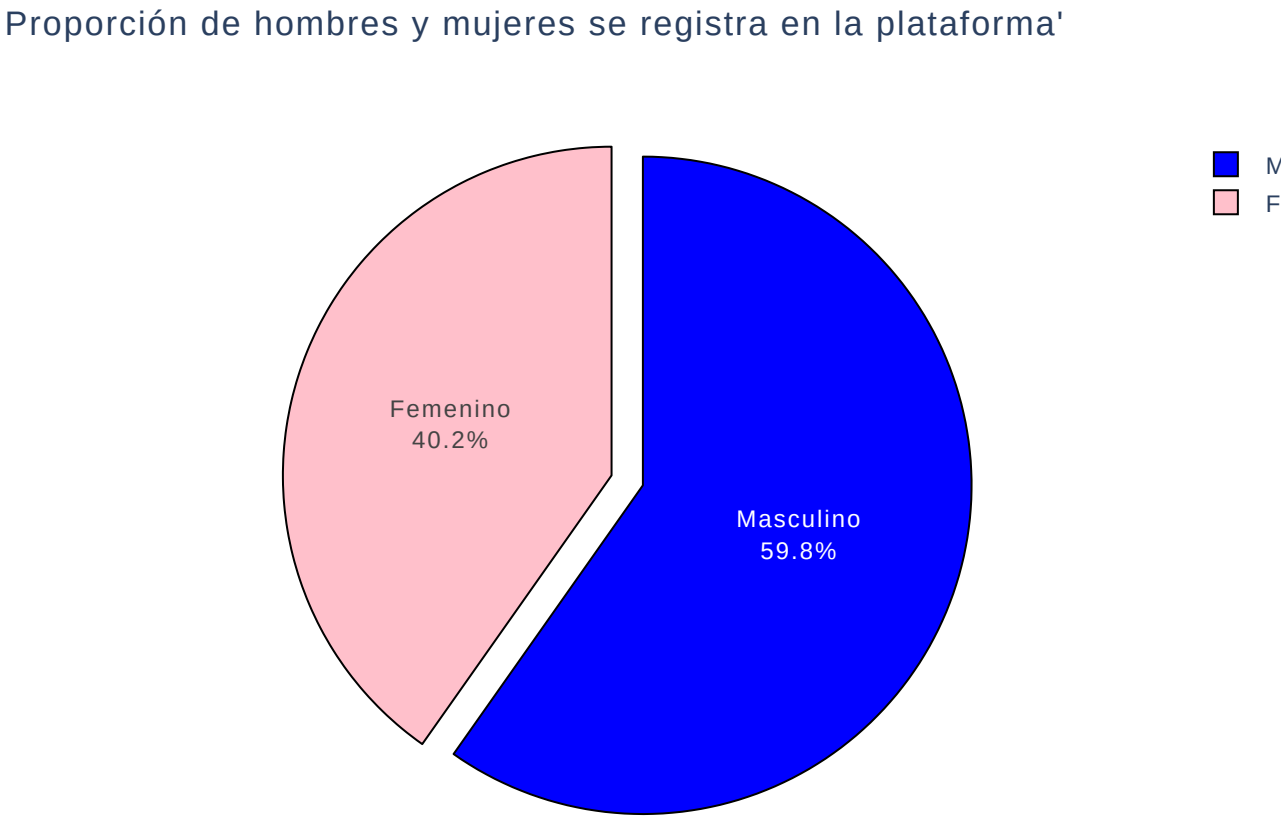


El rango de edad con mayor participación en la búsqueda de pareja en OkCupid se encuentra entre los 20 y 35 años aproximadamente. Esto se evidencia claramente en el histograma de distribución de edad, donde las barras de mayor altura, que representan la mayor frecuencia de usuarios, se concentran en ese intervalo. El diagrama de caja (box plot) complementa esta información, mostrando que el 50% de los datos (la caja) se encuentra en un rango de edad similar, con la mediana ubicada alrededor de los 30 años.

PROPORCIÓN DE HOMBRES Y MUJERES EN LA PLATAFORMA

```
filtro = df[df['sex'].notna()]

conteo = filtro['sex'].value_counts()
fig = go.Figure(
    go.Pie(
        labels=conteo.index,
        values=conteo.values,
        hole=0,
        textinfo='label+percent',
        pull=[0.05]*len(conteo),
        marker=dict(line=dict(color='#000000', width=1)),
    )
)

fig.update_traces(marker=dict(
    colors=['blue','pink','#8da0cb'],
```

```
    line=dict(color='black', width=1)
))
fig.update_layout(
    title="Proporción de hombres y mujeres se registra en la plataforma'",
    showlegend=True
)
fig.show()
```

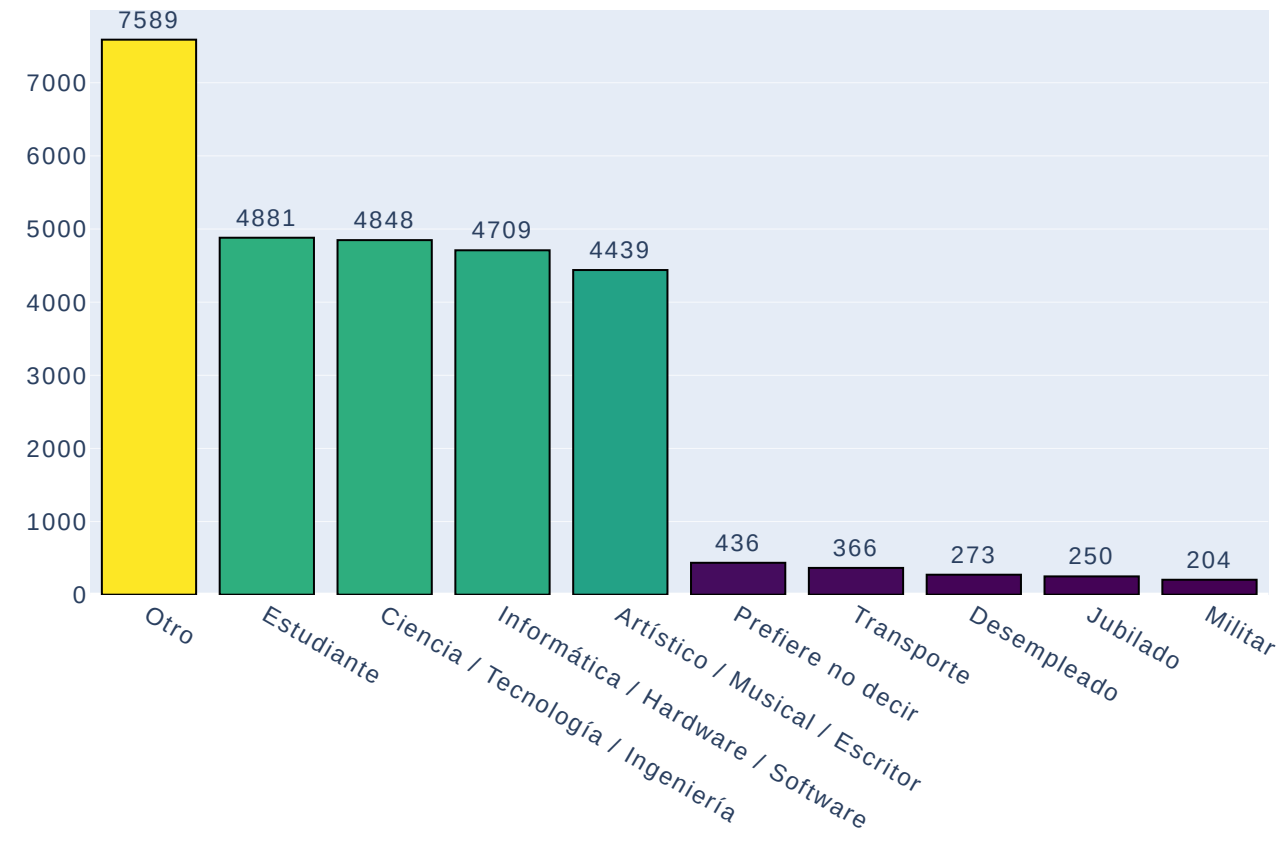


La proporción de usuarios en OkCupid revela una marcada diferencia de género. La plataforma está compuesta en su mayoría por hombres, que representan un 59.8% del total, mientras que las mujeres constituyen un 40.2%. Este desequilibrio sugiere que, en el contexto de la búsqueda de pareja en línea, la cantidad de perfiles masculinos supera significativamente la de los femeninos, lo que podría influir en la dinámica de las interacciones y la competencia dentro de la plataforma.

TRABAJOS MÁS FRECUENTES E INFRECUENTES

```
filtro = df[df['job'].notna()]
conteo = filtro['job'].value_counts()

# Obtener 5 trabajos mas frecuentes y 5 menos frecuentes
top_3 = conteo.head(5)
bottom_3 = conteo.tail(5)
```

```
# Combine top and bottom counts
conteo_filtered = pd.concat([top_3, bottom_3])

fig = go.Figure(
    go.Bar(
        x=conteo_filtered.index,
        y=conteo_filtered.values,
        text=conteo_filtered.values,
        textposition='outside',
        marker=dict(
            color=conteo_filtered.values,
            colorscale='Viridis',
            line=dict(color='#000000', width=1)
        ),
    )
)
fig.update_layout(
    title="Trabajos mas frecuentes e infrecuentes",
    showlegend=False
)

fig.show()
```



EMPLEOS MÁS PRESENTES

Los trabajos más comunes se agrupan en las siguientes categorías, con un número significativo de perfiles:

“Otro”: Esta categoría es, con diferencia, la más frecuente, con 7589 registros. Esto puede incluir una amplia variedad de ocupaciones que no encajan en las categorías predefinidas.

Estudiante: La segunda categoría más numerosa, con 4881 usuarios. Esto coincide con el hecho de que el rango de edad más activo en la plataforma se encuentra entre los 20 y 35 años.

Ciencia / Tecnología / Ingeniería / Informática / Hardware / Software: Estas categorías relacionadas con el sector tecnológico y científico también muestran una alta presencia, con 4848 y 4709 usuarios, respectivamente.

EMPLEOS MENOS PRESENTES

Las ocupaciones con menor representación en la plataforma son las siguientes:

Artístico / Musical / Escritor: Solo 436 usuarios.

“Prefiere no decir”: Un total de 366 usuarios optaron por no especificar su ocupación.

Transporte: Con solo 273 registros.

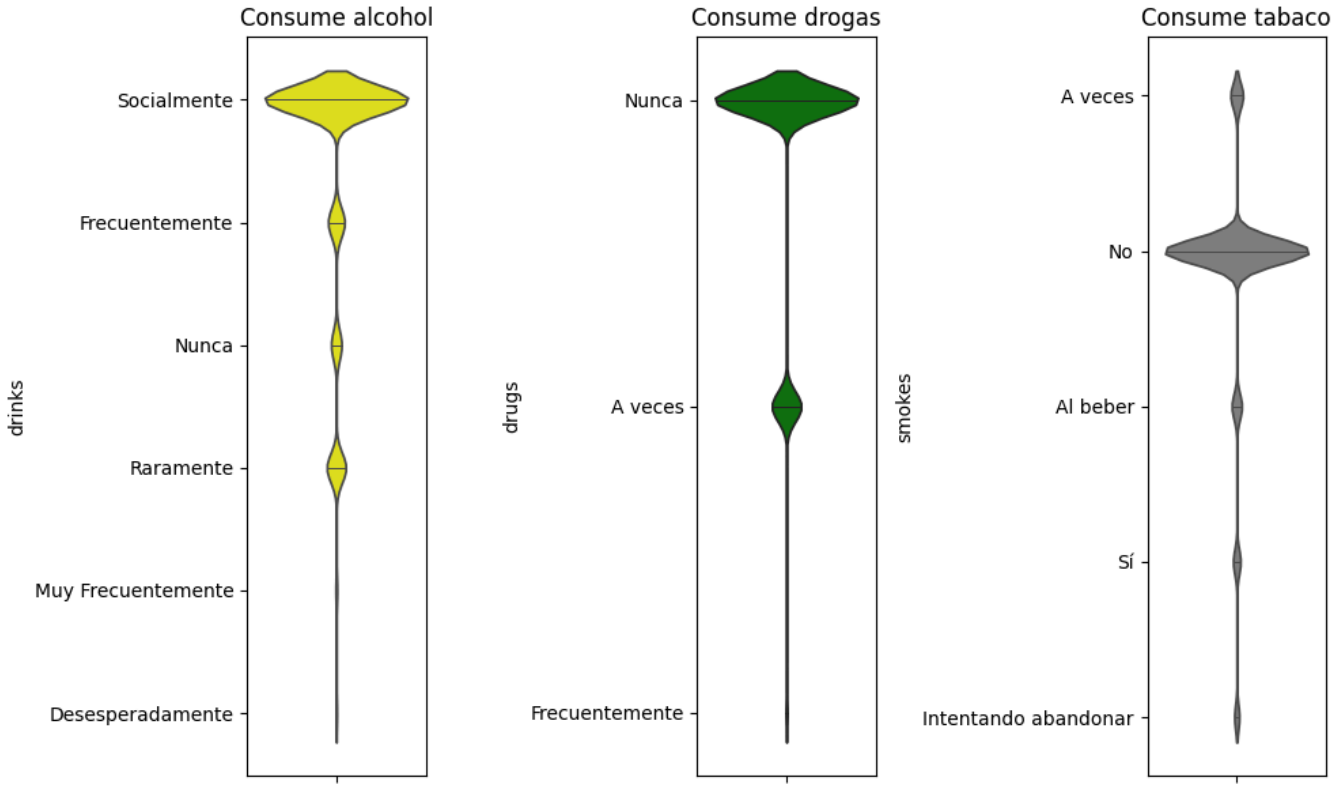
Desempleado: Esta categoría tiene 250 perfiles.

Jubilado: Con 204 usuarios, es la penúltima categoría en cuanto a frecuencia.

Militar: Es el empleo menos común, con tan solo 204 usuarios registrados.

DIAGRAMA DE VIOLIN DE HÁBITOS

```
plt.figure(figsize=(10, 6))
filtro1 = df[df['drinks'].notna()]
filtro2 = df[df['drugs'].notna()]
filtro3 = df[df['smokes'].notna()]
# Crear diagramas de violín para drinks, drugs y smokes
plt.subplot(1, 3, 1)
sns.violinplot(y='drinks', data=filtro1, color='yellow',inner='stick')
plt.title('Consume alcohol')

plt.subplot(1, 3, 2)
sns.violinplot(y='drugs', data=filtro2,color='green', inner='stick')
plt.title('Consume drogas')

plt.subplot(1, 3, 3)
sns.violinplot(y='smokes', data=filtro3, color='gray',inner='stick')
plt.title('Consume tabaco')

plt.tight_layout()
plt.show()
```

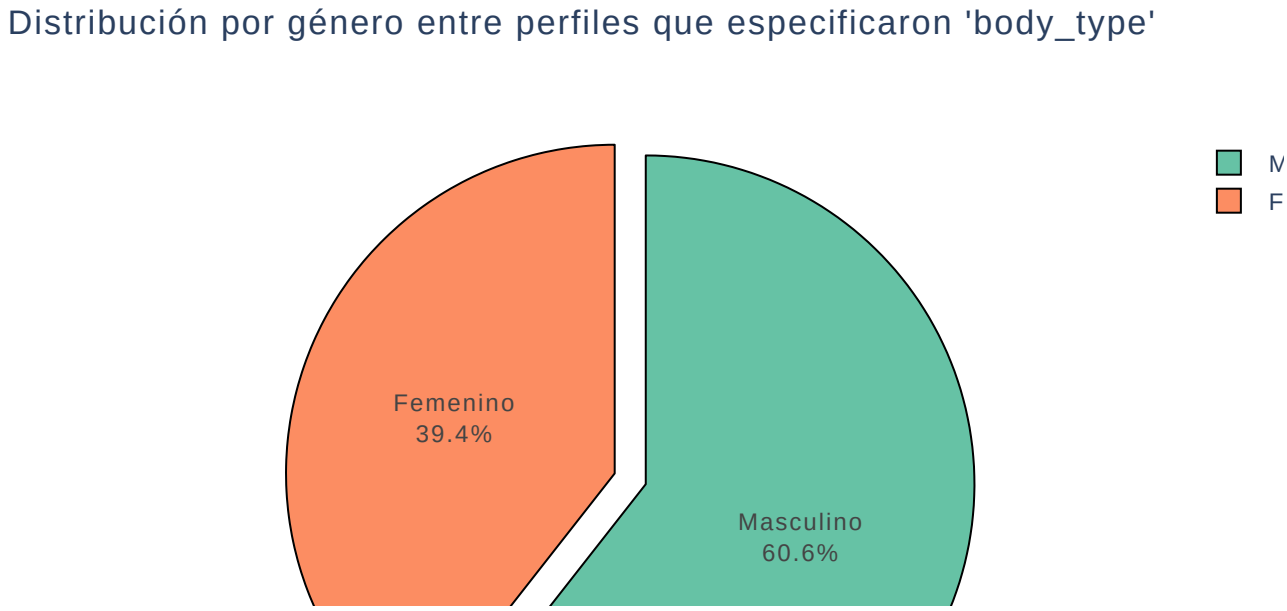


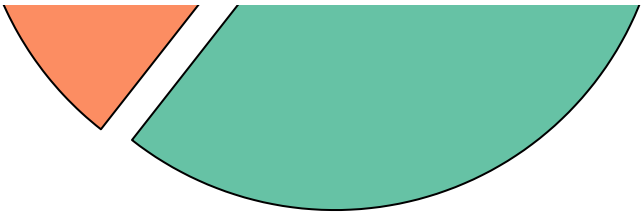
- Alcohol (drinks): La mayoría de los usuarios consumen alcohol de forma “Socialmente”, lo que indica que es una práctica común y aceptada. Las categorías “Frecuentemente” y “Nunca” también muestran una presencia, pero con un número significativamente menor de usuarios en comparación.
- Drogas (drugs): El consumo de drogas es un hábito muy poco frecuente en la plataforma. La inmensa mayoría de los usuarios declara que “Nunca” consume drogas. La categoría “A veces” tiene una presencia mínima, y “Frecuentemente” es prácticamente inexistente.
- Tabaco (smokes): El hábito de fumar tabaco también parece ser poco común. La mayor concentración de usuarios se encuentra en la respuesta “No”, lo que indica que la mayoría de las personas no fuman. Las categorías “A veces” y “Al beber” también tienen una presencia, pero son notablemente menos comunes.

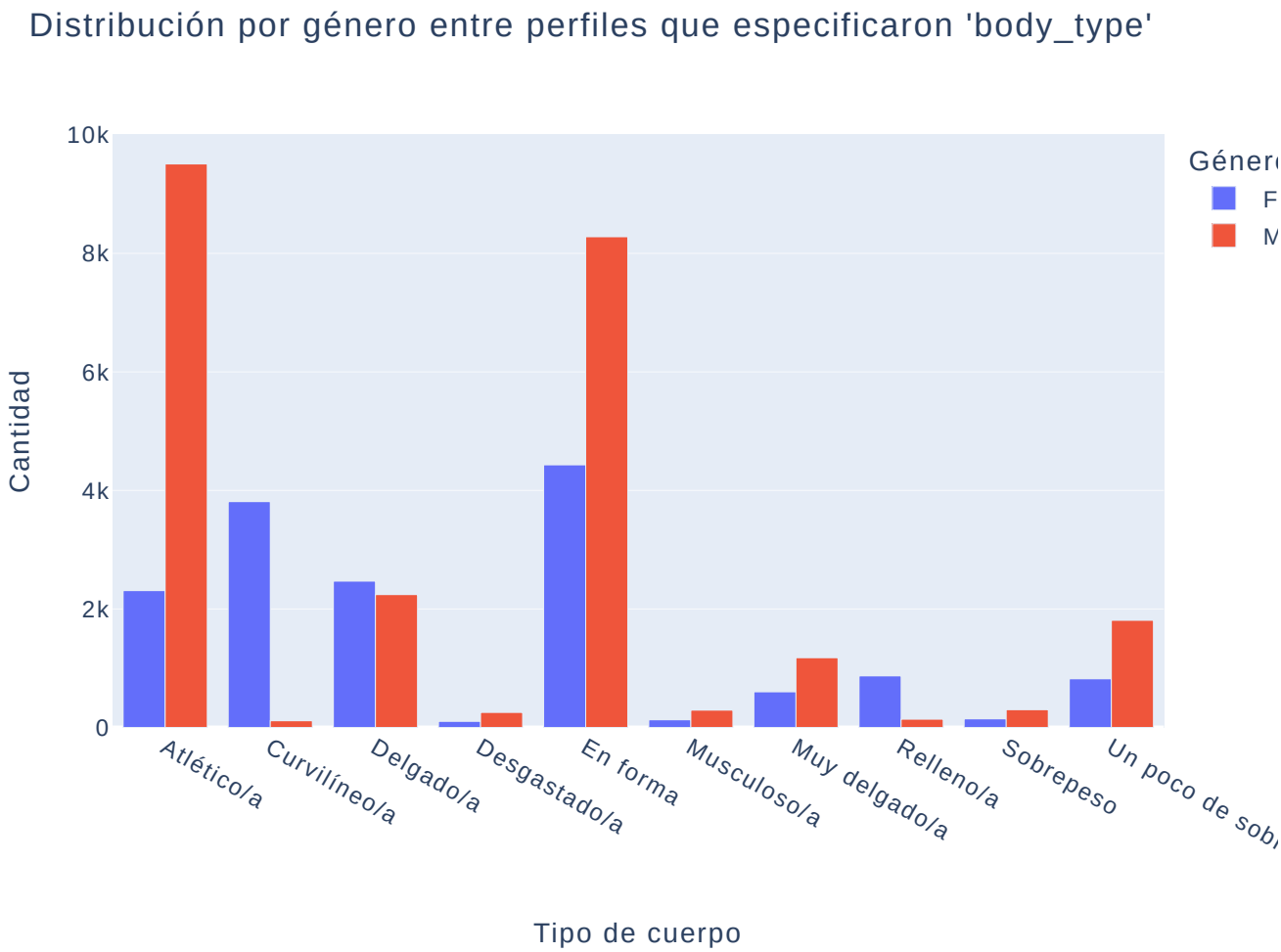
DISTRIBUCIÓN POR GÉNERO ENTRE PERFILES QUE ESPECIFICARON ‘BODY_TYPE’

```
# Filtramos perfiles con género y body_type válido
df_torta = df[df['sex'].notna() & (df['body_type'] != 'Promedio') & (df['body
# Conteo por género
conteoSex = df_torta['sex'].value_counts()
conteoType = df_torta['body_type'].value_counts()

# Torta interactiva
fig = go.Figure(
    go.Pie(
        labels=conteoSex.index,
        values=conteoSex.values,
```

```
        hole=0,  # Si quieres tipo dona pon hole=0.4
        textinfo='label+percent',
        pull=[0.05]*len(conteoSex),  # separa un poco las porciones
        marker=dict(line=dict(color='#000000', width=1)),
    )
)

# Simular efecto 3D (usando layout y sombreado)
fig.update_traces(marker=dict(
    colors=['#66c2a5','#fc8d62','#8da0cb'],
    line=dict(color='black', width=1)
))
fig.update_layout(
    title="Distribución por género entre perfiles que especificaron 'body_typ
    showlegend=True
)
fig.show()

# Filtrar perfiles con género y body_type válido
df_torta = df[df['sex'].notna() & (df['body_type'] != 'Promedio') & (df['body

# Conteo por género y tipo de cuerpo
conteo_group = df_torta.groupby(['body_type', 'sex']).size().reset_index(name

# Gráfico de barras agrupadas
fig = px.bar(
    conteo_group,
    x='body_type',
    y='count',
    color='sex',
    barmode='group',
    title="Distribución por género entre perfiles que especificaron 'body_typ
    labels={'body_type': 'Tipo de cuerpo', 'count': 'Cantidad', 'sex': 'Gener
)
fig.show()
```





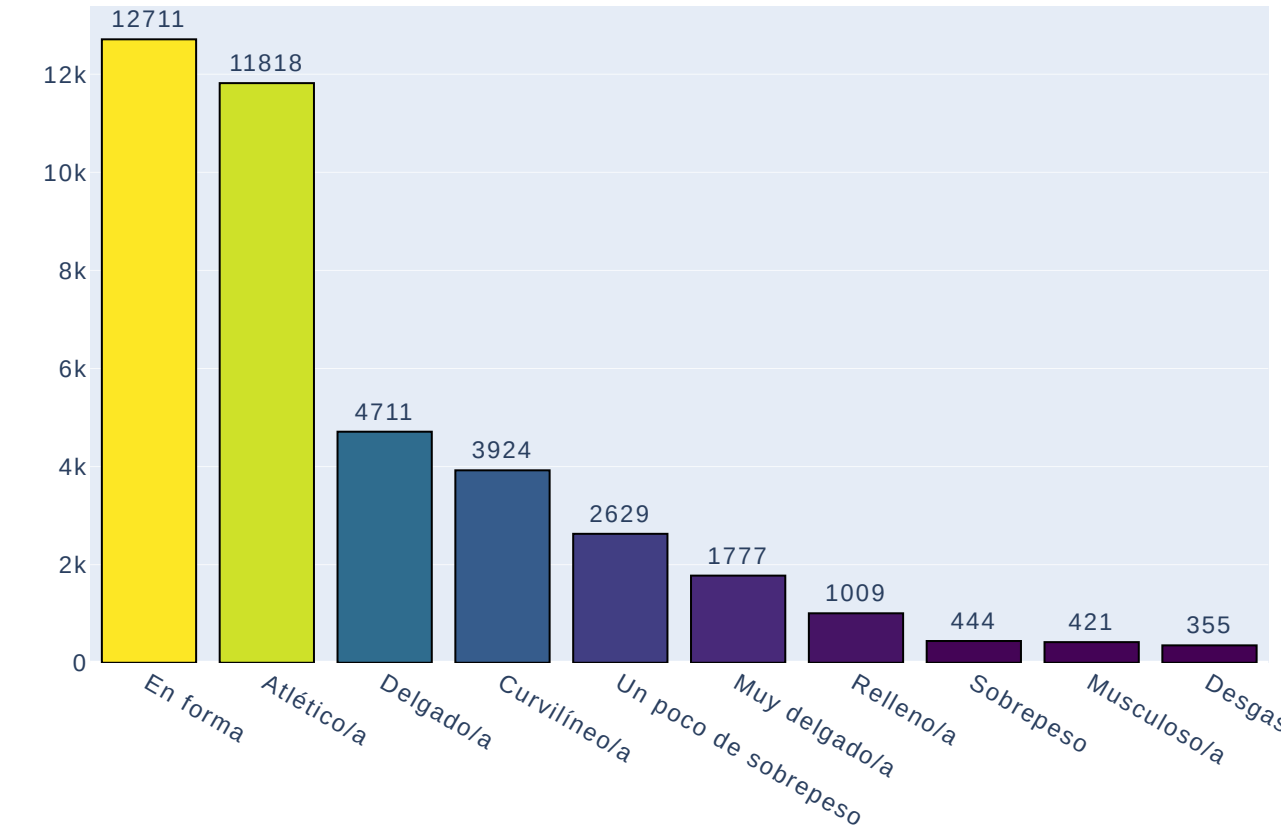


Se observa que los perfiles masculinos tienden a especificar su tipo de cuerpo con mayor frecuencia que los femeninos. El gráfico circular muestra que, del total de perfiles que proporcionaron esta información, el 60.6% son de hombres, mientras que el 39.4% son de mujeres.

El gráfico de barras desglosa esta información por tipo de cuerpo. Si bien ambos géneros se identifican con diversas categorías, la cantidad absoluta de hombres que reportan un tipo de cuerpo es consistentemente mayor que la de mujeres en la mayoría de las categorías, especialmente en “Atlético/a”, “En forma”, y “Musculoso/a”. Esto sugiere que, al menos en esta muestra, los hombres están más inclinados a incluir esta información en sus perfiles.

DISTRIBUCIÓN DE TIPOS DE CUERPO

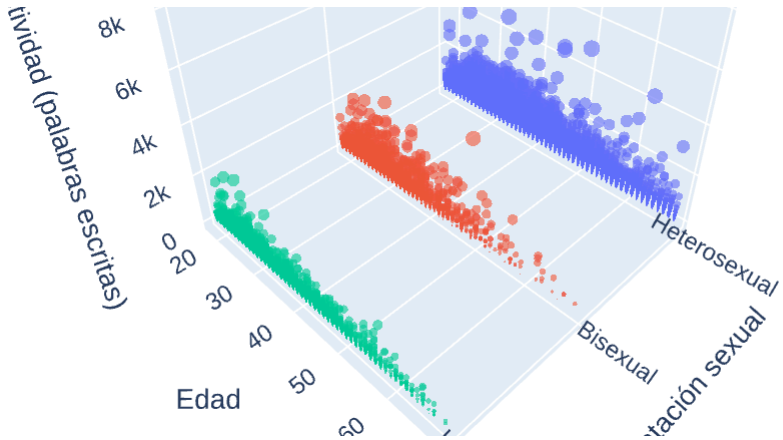
```python
# Filtramos perfiles con género y body_type válido
df_torta = df[df['sex'].notna() & (df['body_type'] != 'Promedio') & (df['body
# Conteo por body_type
conteoType = df_torta['body_type'].value_counts()

# Barra interactiva
fig = go.Figure(
    go.Bar(
        x=conteoType.index,
        y=conteoType.values,
        text=conteoType.values,
        textposition='outside',
        marker=dict(
            color=conteoType.values,
            colorscale='Viridis',
            line=dict(color='#000000', width=1)
        ),
    )
)
fig.update_layout(
    title="Distribución de tipos de cuerpo (Excluyendo 'Promedio' and 'Prefie
    showlegend=False
)

fig.show()
```



Los tipos de cuerpo más comunes entre los perfiles de OkCupid son “En forma” con 12,711 registros y “Atlético/a” con 11,818. Los menos comunes son “Sobrepeso” y “Musculoso/a”, con 444 y 421 registros, respectivamente, y finalmente “Desgastado/a” con solo 355.

ORIENTACIÓN SEXUAL, EDAD Y ACTIVIDAD GENERAL

```
# --- Medir "actividad general" ---
essay_cols = [f"essay{i}" for i in range(10)]
df['actividad'] = df[essay_cols].fillna('').apply(lambda row: ' '.join(row),

# Datos a analizar
df_comp = df[df['age'].notna() & df['orientation'].notna()].copy()

# Limit age between 18 to 80
df_comp = df_comp[(df_comp['age'] >= 18) & (df_comp['age'] <= 90)]

# Gráfico interactivo 3D
fig = px.scatter_3d(
    df_comp,
    x="orientation",    # Eje X
    y="age",            # Eje Y
    z="actividad",      # Eje Z
    color="orientation",
    size="actividad",
    opacity=0.6,
    title="Orientación sexual, edad y actividad general",
    labels={
        "orientation": "Orientación sexual",
        "age": "Edad",
        "actividad": "Actividad (palabras escritas)"
    }
)

fig.update_traces(marker=dict(line=dict(width=0)))  # Sin bordes
fig.show()
```

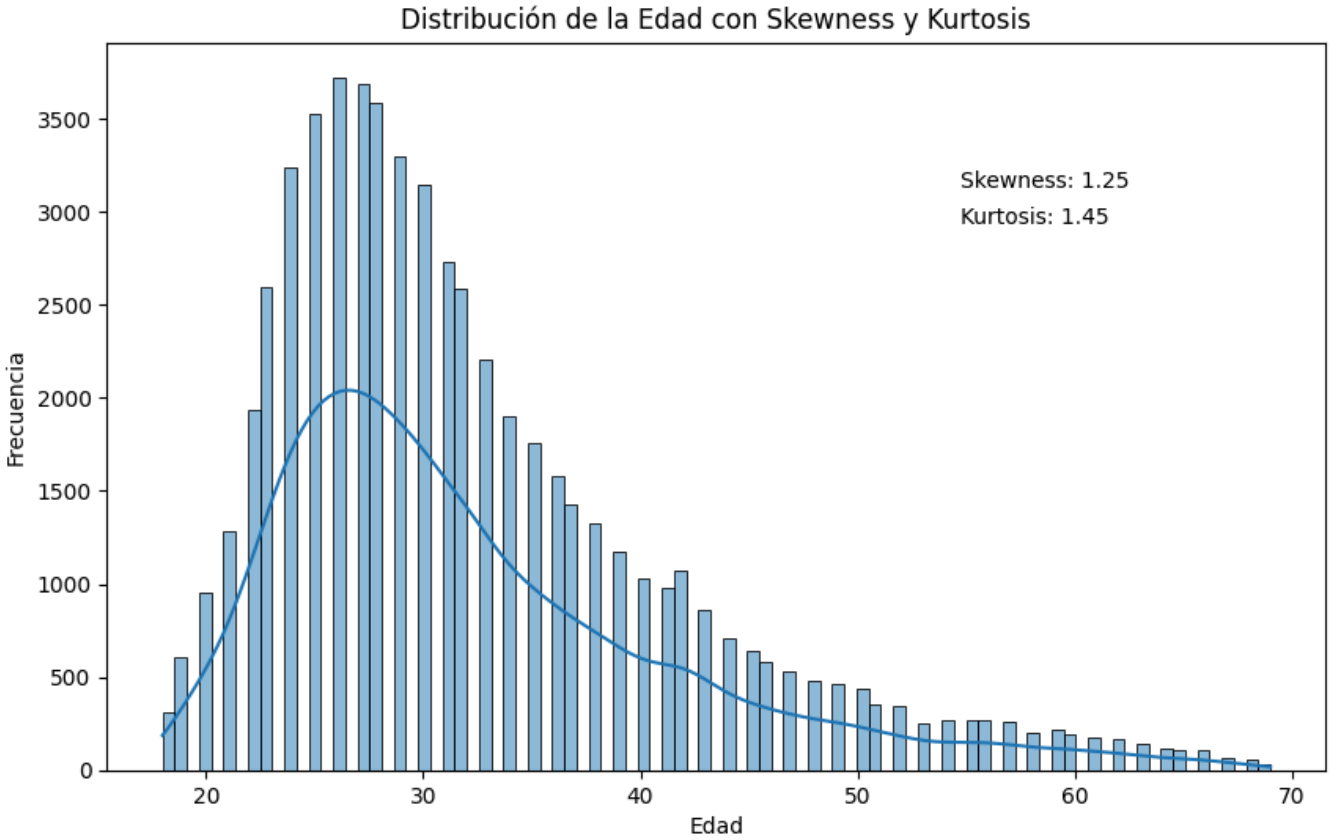


- Influencia de la Orientación Sexual: Los usuarios heterosexuales son, con diferencia, el grupo más numeroso y muestran un rango más amplio de actividad. Los perfiles bisexuales y homosexuales son menos numerosos, pero también exhiben una actividad considerable. Sin embargo, en términos de la distribución total, los heterosexuales dominan el panorama.
- Influencia de la Edad: A medida que la edad avanza, se observa una disminución general en la cantidad de palabras escritas en los perfiles. Esto sugiere que los usuarios más jóvenes (entre 20 y 40 años, que es el rango de mayor participación) tienden a ser más descriptivos y activos en sus perfiles, mientras que los usuarios de mayor edad (50 años en adelante) suelen escribir menos. Esta tendencia se mantiene constante en las tres orientaciones sexuales.

7. ANÁLISIS ESTADÍSTICO MÁS PROFUNDO

ASIMETRIA Y CURTOSIS

```
# Histograma de edad con skewness y kurtosis
plt.figure(figsize=(10, 6))
sns.histplot(df['age'].dropna(), kde=True)
plt.title('Distribución de la Edad con Skewness y Kurtosis')
plt.xlabel('Edad')
plt.ylabel('Frecuencia')

# Añadir texto para valores de skewness y kurtosis en el gráfico
skewness = df['age'].skew()
kurtosis = df['age'].kurtosis()
plt.text(0.7, 0.8, f'Skewness: {skewness:.2f}', transform=plt.gca().transAxes
plt.text(0.7, 0.75, f'Kurtosis: {kurtosis:.2f}', transform=plt.gca().transAxe

plt.show()
```



ASIMETRÍA

La distribución de la edad presenta una asimetría positiva de 1.25, lo que significa que la mayor concentración de personas se encuentra en edades más bajas, especialmente entre los 20 y 30 años. Sin embargo, la cola de la distribución se extiende hacia la derecha, lo que refleja la presencia de personas de mayor edad, aunque en menor proporción.

CURTOSIS

La curtosis de 1.45 indica que la distribución es más plana que una distribución normal. Esto implica que el pico central no es tan pronunciado y que las colas son más ligeras, por lo que existen menos valores extremos de lo que se esperaría en una distribución normal.

8. MODELADO DE LOS DATOS

REGRESIÓN LOGISTICA

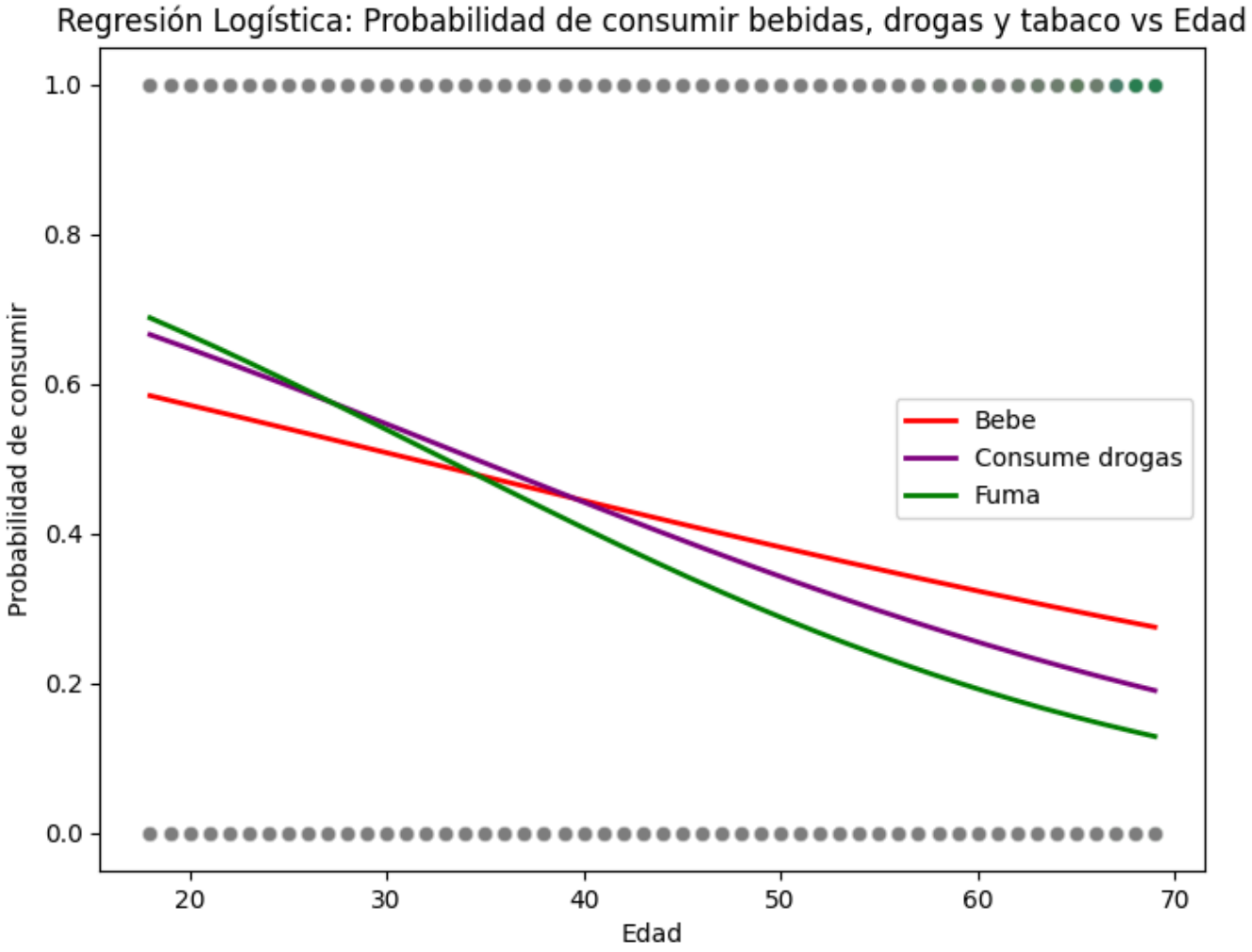
Se eligió este modelo porque permite realizar clasificaciones basadas en el comportamiento y los patrones observados en las personas. En este caso, el análisis se centra en los hábitos que presentan según su edad y su sexo.

ENTRENAMIENTO Y VALIDACIÓN

Se entrenaron 3 modelos, uno para cada habito (fumar, tomar, consumo de drogas).

```
# Seleccionar columnas y eliminar valores nulos
data = data[["drinks", "drugs", "age", "sex", "smokes"]].dropna()

# Para sobre-muestrear los datos
ros = RandomOverSampler(random_state=42)

# Usa 'age' para graficar la curva logística
X = data[["age","sex"]]
y = data["drinks"]
y2 = data["drugs"]
y3 = data["smokes"]

# Ajustar modelo para 'drinks'
Xr, yr = ros.fit_resample(X, y)

model_drinks = LogisticRegression()
model_drinks.fit(Xr, yr)

# Ajustar modelo para 'drugs'
Xr, yr = ros.fit_resample(X, y2)


model_drugs = LogisticRegression()
model_drugs.fit(Xr, yr)

# Ajustar modelo para 'smokes'
Xr, yr = ros.fit_resample(X, y3)


model_smokes = LogisticRegression()
model_smokes.fit(Xr, yr)

# Crear rango de edades para predicción de la curva
X_range = pd.DataFrame({'age': np.linspace(X['age'].min(), X['age'].max(), 30
y_prob_drinks = model_drinks.predict_proba(X_range)[:, 1]  # probabilidad par
y_prob_drugs = model_drugs.predict_proba(X_range)[:, 1]  # probabilidad para
y_prob_smokes = model_smokes.predict_proba(X_range)[:, 1] # probabilidad para

# Graficar
plt.figure(figsize=(8,6))
sns.scatterplot(x="age", y="drinks", data=data, alpha=0.3)
sns.scatterplot(x="age", y="drugs", data=data, alpha=0.3, color='green')
sns.scatterplot(x="age", y="smokes", data=data, alpha=0.3, color='gray')

plt.plot(X_range['age'], y_prob_drinks, color="red", linewidth=2, label="Bebe
plt.plot(X_range['age'], y_prob_drugs, color="purple", linewidth=2, label="Co
plt.plot(X_range['age'], y_prob_smokes, color="green", linewidth=2, label="Fu


plt.title("Regresión Logística: Probabilidad de consumir bebidas, drogas y ta
plt.xlabel("Edad")
plt.ylabel("Probabilidad de consumir")
```

```
plt.legend()
plt.show()
```

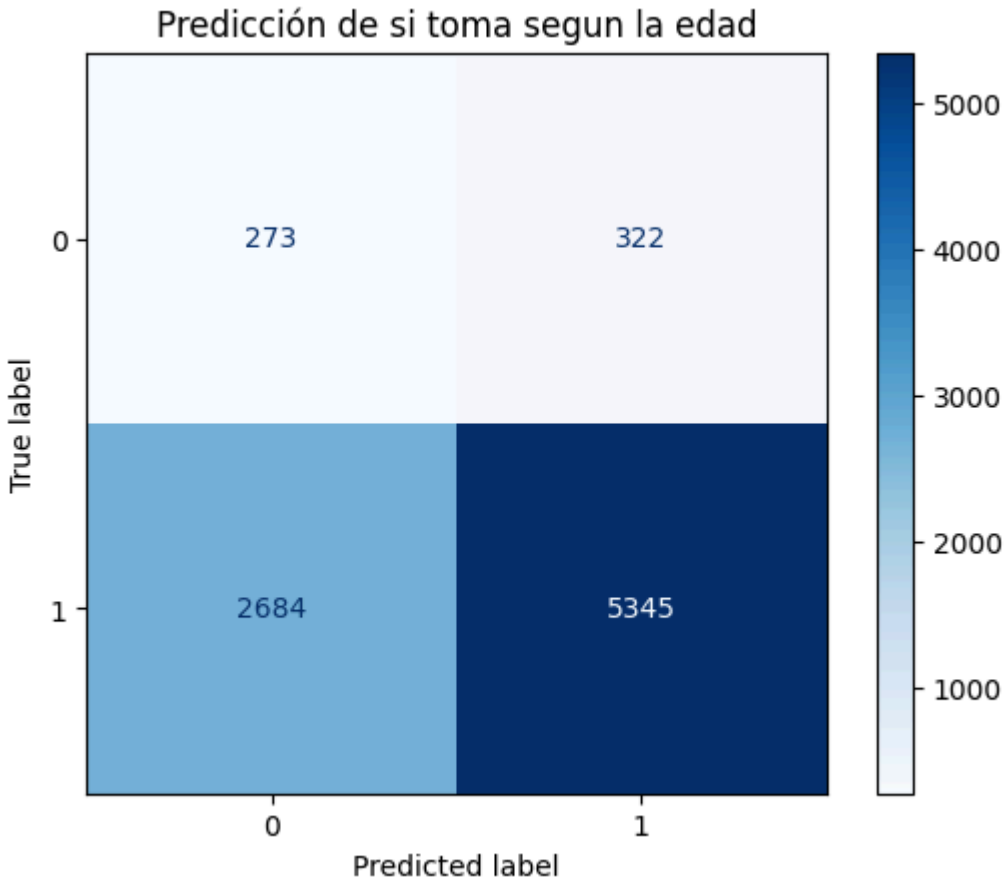


EVALUACIÓN DE RESULTADOS

CONSUME ALCOHOL

```
# Evaluar modelo para 'drinks'
X_train, X_test, y_train, y_test = train_test_split(
    X, y, test_size=0.2, random_state=42
)

X_train_resampled, y_train_resampled = ros.fit_resample(X_train, y_train)

model_drinks.fit(X_train_resampled, y_train_resampled)
y_pred = model_drinks.predict(X_test)

# Gráfica: matriz de confusión
disp = ConfusionMatrixDisplay(confusion_matrix= confusion_matrix(y_test, y_pr
disp.plot(cmap="Blues")
plt.title("Predicción de si toma segun la edad")
plt.show()
```

```python
# Reporte de clasificación drinks
report = classification_report(y_test, y_pred, zero_division=0, output_dict=T
report_df = pd.DataFrame(report).transpose()
display(Markdown("### Reporte de clasificación drinks"))
display(report_df.style.format({"precision": "{:.2f}", "recall": "{:.2f}", "f

# Exactitud del modelo drinks
accuracy = accuracy_score(y_test, y_pred)
display(Markdown(f"**Exactitud del modelo drinks:** `{accuracy:.2%}`"))

# Coeficientes del modelo
coefficients = pd.DataFrame({
    "Variable": X.columns,
    "Coeficiente": model_drinks.coef_[0]
})
display(Markdown("### Coeficientes de la regresión logística drinks"))
display(coefficients)

# Probabilidades de tomar según edad y sexo
prob_rows = []
for i in range(21, 70, 10):
    prob_hombre = model_drinks.predict_proba(pd.DataFrame({"age": [i], "sex":
    prob_mujer = model_drinks.predict_proba(pd.DataFrame({"age": [i], "sex":
    prob_rows.append({"Edad": i, "Sexo": "Hombre", "Probabilidad de tomar": p
    prob_rows.append({"Edad": i, "Sexo": "Mujer", "Probabilidad de tomar": pr

prob_df = pd.DataFrame(prob_rows)
display(Markdown("### Probabilidad de que tome según edad y sexo"))
display(prob_df.style.format({"Probabilidad de tomar": "{:.2%}"}))
```



REPORTE DE CLASIFICACIÓN DRINKS

	PRECISION	RECALL	F1-SCORE	SUPPORT
0	0.09	0.46	0.15	595
1	0.94	0.67	0.78	8029
accuracy	0.65	0.65	0.65	1
macro avg	0.52	0.56	0.47	8624
weighted avg	0.88	0.65	0.74	8624

Exactitud del modelo drinks: `65.14%`

COEFICIENTES DE LA REGRESIÓN LOGÍSTICA DRINKS

	VARIABLE	COEFICIENTE
0	age	-0.025098

	VARIABLE	COEFICIENTE
1	sex	0.177119

PROBABILIDAD DE QUE TOME SEGÚN EDAD Y SEXO

	EDAD	SEXO	PROBABILIDAD DE TOMAR
0	21	Hombre	56.34%
1	21	Mujer	60.63%
2	31	Hombre	50.10%
3	31	Mujer	54.51%
4	41	Hombre	43.85%
5	41	Mujer	48.25%
6	51	Hombre	37.80%
7	51	Mujer	42.04%
8	61	Hombre	32.10%
9	61	Mujer	36.08%

MATRIZ DE CONFUSIÓN TOMA

273 verdaderos negativos (0 bien clasificado, no toma).

5345 verdaderos positivos (1 bien clasificado, toma).

322 falsos positivos (predijo que toma, pero no toma).

2684 falsos negativos (predijo que no toma, pero sí toma).

COEFICIENTES DE LA REGRESIÓN LOGÍSTICA

age = -0.025098 → A mayor edad, la probabilidad de que se tome disminuye ligeramente.

sex= 0.177119 -Z Los hombres tienden a tomar mas.

MÉTRICAS DEL MODELO

Exactitud (accuracy): ~65%

Precisión 0 (no toma): 9%

Precisión 1 (sí toma): 93%

Recall 0 (no toma): 46%

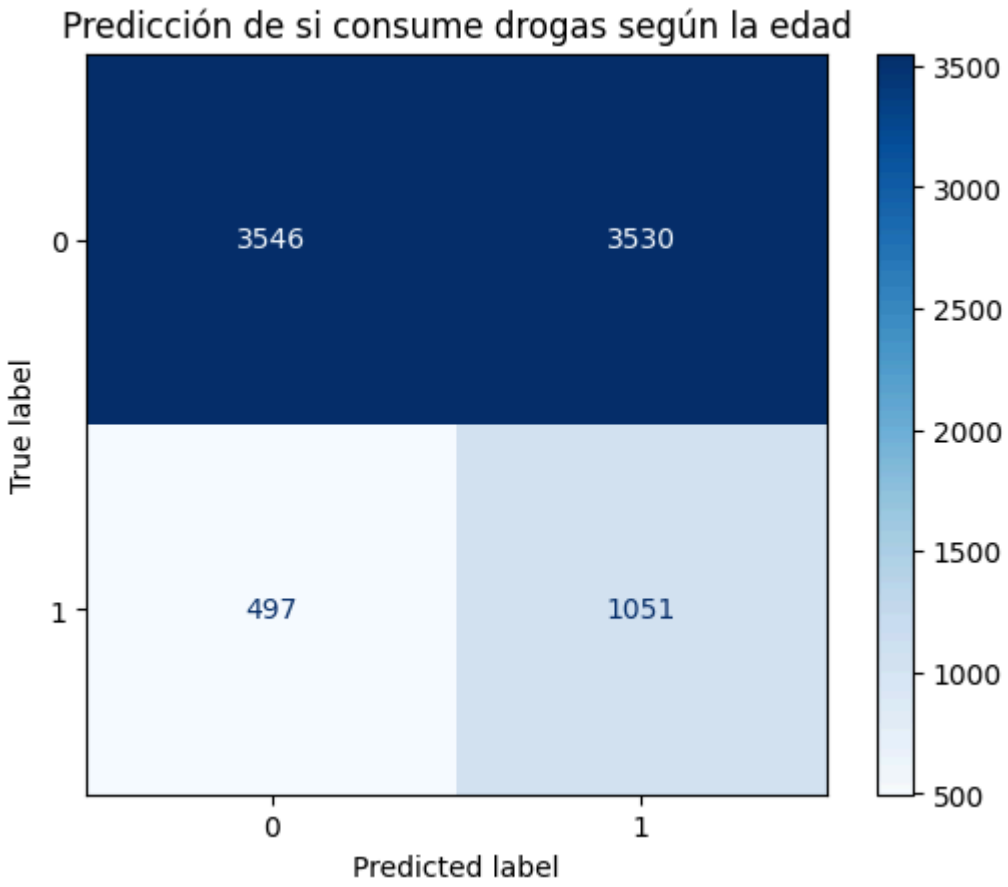
Recall 1 (sí toma): 67%

Esto significa que el modelo es más fuerte prediciendo quién si toma que quién no toma.

CONSUME DROGAS

```
X_train, X_test, y_train, y_test = train_test_split(
    X, y2, test_size=0.2, random_state=42
)

ros = RandomOverSampler(random_state=42)
X_train_resampled, y_train_resampled = ros.fit_resample(X_train, y_train)

model_drugs.fit(X_train_resampled, y_train_resampled)
y_pred = model_drugs.predict(X_test)

# Gráfica: matriz de confusión
disp = ConfusionMatrixDisplay(confusion_matrix=confusion_matrix(y_test, y_pre
disp.plot(cmap="Blues")
plt.title("Predicción de si consume drogas según la edad")
plt.show()

# Reporte de clasificación drugs
report = classification_report(y_test, y_pred, zero_division=0, output_dict=T
report_df = pd.DataFrame(report).transpose()
display(Markdown("### Reporte de clasificación drugs"))
display(report_df.style.format({"precision": "{:.2f}", "recall": "{:.2f}", "f

# Exactitud del modelo drugs
accuracy = accuracy_score(y_test, y_pred)
display(Markdown(f"**Exactitud del modelo drugs:** `{accuracy:.2%}`"))

# Coeficientes del modelo
coefficients = pd.DataFrame({
    "Variable": X.columns,
    "Coeficiente": model_drugs.coef_[0]
})
display(Markdown("### Coeficientes de la regresión logística drugs"))
display(coefficients)

# Probabilidades de consumir drogas según edad y sexo
prob_rows = []
for i in range(21, 70, 10):
    prob_hombre = model_drugs.predict_proba(pd.DataFrame({"age": [i], "sex":
    prob_mujer = model_drugs.predict_proba(pd.DataFrame({"age": [i], "sex": 1
    prob_rows.append({"Edad": i, "Sexo": "Hombre", "Probabilidad de consumir
    prob_rows.append({"Edad": i, "Sexo": "Mujer", "Probabilidad de consumir d
```

```
prob_df = pd.DataFrame(prob_rows)
display(Markdown("### Probabilidad de consumir drogas según edad y sexo"))
display(prob_df.style.format({"Probabilidad de consumir drogas": "{:.2%}"}))
```



REPORTE DE CLASIFICACIÓN DRUGS

	PRECISION	RECALL	F1-SCORE	SUPPORT
0	0.88	0.50	0.64	7076
1	0.23	0.68	0.34	1548
accuracy	0.53	0.53	0.53	1
macro avg	0.55	0.59	0.49	8624
weighted avg	0.76	0.53	0.58	8624

Exactitud del modelo drugs: 53.30%

COEFICIENTES DE LA REGRESIÓN LOGÍSTICA DRUGS

	VARIABLE	COEFICIENTE
0	age	-0.042979
1	sex	-0.316408

PROBABILIDAD DE CONSUMIR DROGAS SEGÚN EDAD Y SEXO

	EDAD	SEXO	PROBABILIDAD DE CONSUMIR DROGAS
0	21	Hombre	63.86%
1	21	Mujer	56.29%
2	31	Hombre	53.48%
3	31	Mujer	45.59%
4	41	Hombre	42.79%
5	41	Mujer	35.28%
6	51	Hombre	32.74%
7	51	Mujer	26.18%
8	61	Hombre	24.05%
9	61	Mujer	18.75%

MATRIZ DE CONFUSIÓN DROGAS

3546 verdaderos negativos (0 bien clasificado, no consume drogas).

1051 verdaderos positivos (1 bien clasificado, consume drogas).

3530 falsos positivos (predijo que consume drogas, pero no).

497 falsos negativos (predijo que no consume drogas, pero sí consume drogas).

####Coeficientes de la regresión logística

age = -0.042979 → A mayor edad, la probabilidad de que se drogue disminuye ligeramente.

sex = -0.316508 → Las mujeres tienden a consumir menos drogas

MÉTRICAS DEL MODELO

Exactitud (accuracy): ~53%

Precisión 0 (no se droga): 88%

Precisión 1 (sí se droga): 23%

Recall 0 (no se droga): 50%

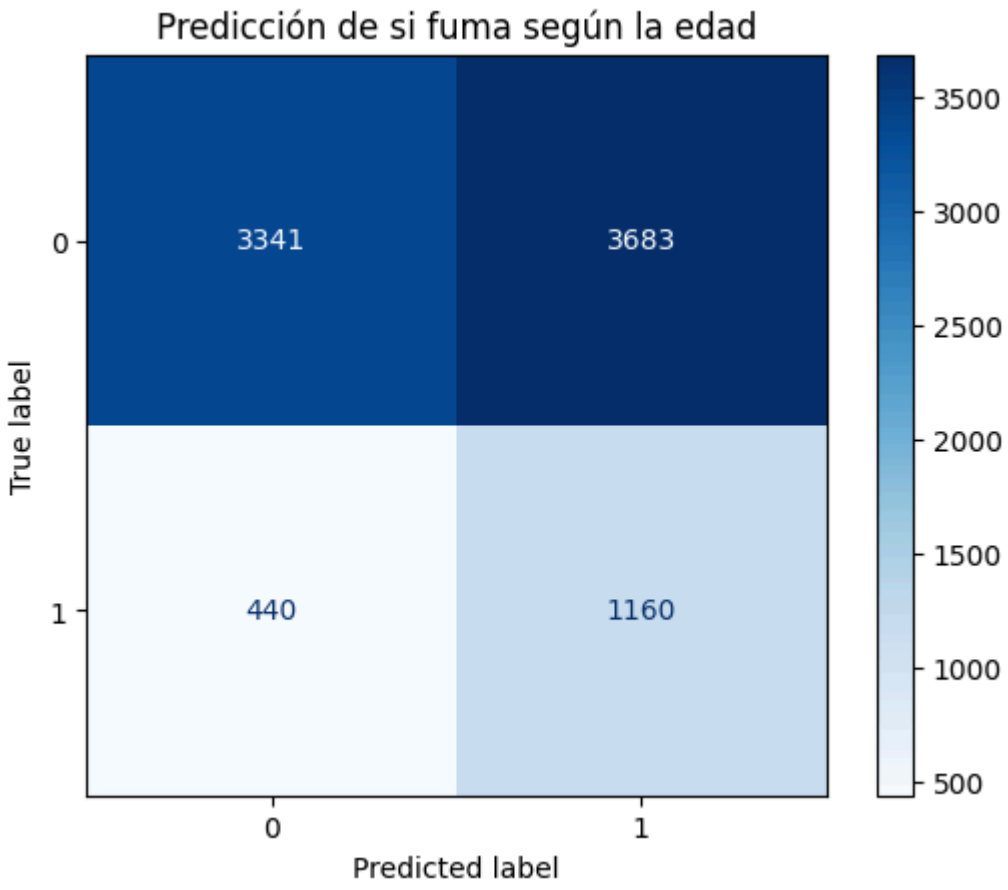
Recall 1 (sí se droga): 68%

Esto significa que el modelo es más fuerte prediciendo quién no se droga que quién si se droga

CONSUME TABACO

```
X_train, X_test, y_train, y_test = train_test_split(
    X, y3, test_size=0.2, random_state=42
)

ros = RandomOverSampler(random_state=42)
X_train_resampled, y_train_resampled = ros.fit_resample(X_train, y_train)

model_smokes.fit(X_train_resampled, y_train_resampled)
y_pred = model_smokes.predict(X_test)

# Gráfica: matriz de confusión
disp = ConfusionMatrixDisplay(confusion_matrix=confusion_matrix(y_test, y_pre
disp.plot(cmap="Blues")
plt.title("Predicción de si fuma según la edad")
plt.show()

# Reporte de clasificación smokes
report = classification_report(y_test, y_pred, zero_division=0, output_dict=T
report_df = pd.DataFrame(report).transpose()
display(Markdown("### Reporte de clasificación smokes"))
display(report_df.style.format({"precision": "{:.2f}", "recall": "{:.2f}", "f

# Exactitud del modelo smokes
accuracy = accuracy_score(y_test, y_pred)
display(Markdown(f"**Exactitud del modelo smokes:** `{accuracy:.2%}`"))

# Coeficientes del modelo smokes
coefficients = pd.DataFrame({
    "Variable": X.columns,
    "Coeficiente": model_smokes.coef_[0]
})
display(Markdown("### Coeficientes de la regresión logística smokes"))
display(coefficients)

# Probabilidades de fumar según edad y sexo
prob_rows = []
for i in range(21, 70, 10):
```

```
    prob_hombre = model_smokes.predict_proba(pd.DataFrame({"age": [i], "sex"
    prob_mujer = model_smokes.predict_proba(pd.DataFrame({"age": [i], "sex":
    prob_rows.append({"Edad": i, "Sexo": "Hombre", "Probabilidad de fumar": p
    prob_rows.append({"Edad": i, "Sexo": "Mujer", "Probabilidad de fumar": pr

prob_df = pd.DataFrame(prob_rows)
display(Markdown("### Probabilidad de que fume según edad y sexo"))
display(prob_df.style.format({"Probabilidad de fumar": "{:.2%}"}))
```



REPORTE DE CLASIFICACIÓN SMOKES

	PRECISION	RECALL	F1-SCORE	SUPPORT
0	0.88	0.48	0.62	7024
1	0.24	0.72	0.36	1600
accuracy	0.52	0.52	0.52	1
macro avg	0.56	0.60	0.49	8624
weighted avg	0.76	0.52	0.57	8624

Exactitud del modelo smokes: `52.19%`

COEFICIENTES DE LA REGRESIÓN LOGÍSTICA SMOKES

	VARIABLE	COEFICIENTE
0	age	-0.050437
1	sex	-0.233407

PROBABILIDAD DE QUE FUME SEGÚN EDAD Y SEXO

	EDAD	SEXO	PROBABILIDAD DE FUMAR
0	21	Hombre	64.73%
1	21	Mujer	59.23%
2	31	Hombre	52.56%
3	31	Mujer	46.74%
4	41	Hombre	40.09%
5	41	Mujer	34.63%
6	51	Hombre	28.78%
7	51	Mujer	24.24%
8	61	Hombre	19.62%
9	61	Mujer	16.19%

MATRIZ DE CONFUSIÓN FUMA

3341 verdaderos negativos (0 bien clasificado, no fuma).

1160 verdaderos positivos (1 bien clasificado, fuma).

3693 falsos positivos (predijo que fuma, pero no).

440 falsos negativos (predijo que no fuma, pero sí fuma).

COEFICIENTES DE LA REGRESIÓN LOGÍSTICA

age = -0.050437 → La edad no es tan caracteristica para determinar si se fuma o no.

sex = -0.233407 -> Los hombres tienden a fumar ligeramente mas que las mujeres

MÉTRICAS DEL MODELO

Exactitud (accuracy): ~88%

Precisión 0 (no fuma): 24%

Precisión 1 (sí fuma): 72%

Recall 0 (sí fuma): 48%

Esto significa que el modelo es más fuerte prediciendo quién no fuma que quién si fuma

9. RESULTADOS Y DISCUSIÓN

- En los 3 modelos que se entrenaron, los tres tuvieron un recall superior solo para predecir los casos en los que las personas tienen alguno de los 3 habitos (1). Aparte, en cada categoría se observa una tendencia al sesgo: hacia la clase 1 (consumir) en el caso del alcohol, y hacia la clase 0 (no consumir) en el caso de fumar o consumir drogas. Esto se debe a que en la base de datos no todos los participantes respondieron las tres preguntas. Por esta razón, para el entrenamiento de los modelos fue necesario aplicar un proceso de oversampling con el fin de obtener un conjunto de datos más equilibrado y representativo de la realidad.

- ¿Qué proporción de hombres y mujeres se registra en la plataforma?

  La proporción de usuarios en la plataforma es desigual, con una mayoría de hombres. Específicamente, el 59.8% de los perfiles pertenecen a hombres, mientras que el 40.2% corresponde a mujeres.

- ¿Cuáles son los hábitos nocivos (alcohol, drogas, cigarrillos) más comunes entre los usuarios?

  El habito socialemente mas aceptado fue el alcohol. Sumado a esto, con respecto a los habitos nocivos en general, se encontro que las personas registradas, mientas mas edad tenian, menos probable era que tuvieran alguno de los 3 habitos nocivos.

- ¿Cuál es el rango de edad con mayor participación en la búsqueda de pareja?

  El rango de edad con mayor participación en la búsqueda de pareja en OkCupid se concentra entre los 20 y 35 años. Esto se evidencia en el histograma de distribución, que muestra la mayor frecuencia de usuarios en ese intervalo, y se refuerza con el diagrama de caja, donde el 50% de los datos se ubica en un rango de edad similar.

- ¿Existe un género que preste más atención al tipo de cuerpo en los perfiles?

  Sí, los hombres parecen prestar más atención al tipo de cuerpo en sus perfiles. De los usuarios que especificaron esta información, el 60.6% eran hombres, lo que sugiere que están más inclinados a incluir este dato en sus perfiles que las mujeres, que representan el 39.4%.

- ¿Cómo influyen la orientación sexual y la edad en la actividad general de los usuarios?

La actividad de los usuarios, medida por la cantidad de palabras en el perfil, se ve influenciada por la orientación sexual y la edad. Los perfiles heterosexuales son los más numerosos y activos. Además, para todas las orientaciones sexuales, la actividad tiende a disminuir con la edad, lo que sugiere que los usuarios más jóvenes son más propensos a escribir perfiles más detallados.

- ¿Qué empleos estan mas presentes?

  Las categorías de empleo más comunes en la plataforma son “Otro”, que incluye una amplia variedad de ocupaciones no especificadas, y “Estudiante”. Otras categorías con alta presencia son las relacionadas con el sector tecnológico y de ciencias, como “Ciencia / Tecnología / Ingeniería” e “Informática / Hardware / Software”.

10. CONCLUSIONES Y RECOMENDACIONES

PREGUNTA PRINCIPAL

En el análisis de los perfiles de OkCupid se identifican patrones de comportamiento y características comunes que permiten comprender mejor la dinámica de quienes utilizan la plataforma en búsqueda de una relación. Se observa, en primer lugar, un claro predominio de hombres, quienes constituyen el 59,8% de la muestra frente al 40,2% de mujeres, lo cual sugiere una asimetría de género en la oferta de perfiles. En términos de edad, la participación se concentra entre los 20 y 35 años, con una mediana cercana a los 30, evidenciando que la plataforma es especialmente popular entre jóvenes adultos. En cuanto a los hábitos, el consumo de alcohol es el más extendido, generalmente en contextos sociales, mientras que el uso de drogas y el tabaquismo presentan frecuencias significativamente menores, con una tendencia decreciente conforme aumenta la edad. Respecto al tipo de cuerpo, las categorías “en forma” y “atlético/a” son las más reportadas, siendo los hombres quienes más atención dedican a incluir esta información (60,6% frente a 39,4% en mujeres). La orientación sexual muestra un predominio de usuarios heterosexuales, quienes además son los más activos en la redacción de sus perfiles. No obstante, la actividad general —medida a partir de la extensión de los textos — disminuye progresivamente con la edad, de modo que los usuarios más jóvenes tienden a elaborar descripciones más extensas. Finalmente, en el ámbito ocupacional, se destacan los estudiantes y los profesionales vinculados a las ciencias, la ingeniería y la tecnología, lo que configura un panorama diverso pero con presencia significativa de perfiles académicos y técnicos.

HALLAZGOS DE APOYO

- La proporción de hombres que especifican su tipo de cuerpo es considerablemente mayor que la de las mujeres, lo que indica una diferencia de género en la auto-representación física.

- Aunque el alcohol es el hábito más común, la mayoría de los usuarios se identifican como consumidores ocasionales, lo que refleja un patrón socialmente aceptado más que un comportamiento problemático.

- La categoría de ocupación “otro” concentra una alta proporción de usuarios, lo que muestra la dificultad de encasillar la diversidad laboral en categorías predefinidas.

- Los usuarios jóvenes no solo son más numerosos, sino también más detallados en la construcción de su perfil, lo que sugiere un mayor interés en mostrarse de manera completa.
- La baja frecuencia de usuarios que reportan consumo de drogas refuerza la idea de que este comportamiento es poco aceptado socialmente en contextos de búsqueda de pareja.

LIMITACIONES DE ANÁLISIS

No se pudo encontrar una base de datos más reciente sobre el tema, por lo que fue necesario trabajar con información relativamente antigua. Esta limitación puede afectar la representatividad de los resultados, ya que los patrones y comportamientos actuales podrían diferir de los observados en los datos utilizados. Además, el reducido acceso a fuentes actualizadas restringió la posibilidad de contrastar los hallazgos con otras investigaciones o bases de datos complementarias, lo que limita la validez externa y la generalización de las conclusiones.

RECOMENDACIONES FUTURAS

Para futuros trabajos se recomienda la utilización de bases de datos más recientes y completas, de manera que los resultados reflejen con mayor precisión la realidad actual. Asimismo, sería conveniente incorporar variables adicionales que permitan un análisis más profundo y que ayuden a identificar patrones con mayor nivel de detalle.

11. ANEXOS

Base de datos por = Kim, Albert and Escobedo-Land, Adriana.

Año: 2015

Mes: 07

Titulo: OkCupid Data for Introductory Statistics and Data Science Courses,

Volumen: 23

Periodico: Journal of Statistics Education

Link: OkCupid Profiles

Base de datos: okcupid_profiles.csv

DOCUMENTACIÓN

- pandas: Es una biblioteca para manipulación y análisis de datos. Proporciona estructuras de datos como *DataFrames* que facilitan el trabajo con datos tabulares.

  Documentación: https://pandas.pydata.org/docs/

- seaborn: Es una biblioteca para la visualización de datos estadísticos basada en *matplotlib*. Permite crear gráficos atractivos e informativos de manera sencilla.

  Documentación: https://seaborn.pydata.org/

- matplotlib.pyplot: Es una colección de funciones que hacen que *matplotlib* funcione como MATLAB. Proporciona una interfaz sencilla para crear gráficos.

  Documentación: https://matplotlib.org/stable/api/pyplot_api.html

- numpy: Es una biblioteca fundamental para la computación numérica en Python. Proporciona soporte para *arrays* y *matrices*, junto con una gran colección de funciones matemáticas de alto nivel.

  Documentación: https://numpy.org/doc/stable/

- plotly.express: Es un módulo de Plotly que proporciona una API de alto nivel para crear figuras de forma rápida.

  Documentación: https://plotly.com/python/plotly-express/

- plotly.graph_objects: Es el módulo de bajo nivel de Plotly que permite construir figuras creando instancias de clases como `go.Figure`, `go.Scatter`, etc.

  Documentación: https://plotly.com/python/graph-objects/

- sklearn.preprocessing.StandardScaler: Se utiliza para estandarizar características eliminando la media y escalando a la varianza unitaria.

  Documentación: https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.StandardScaler.html

- sklearn.linear_model.LogisticRegression: Implementa la regresión logística, un modelo lineal para problemas de clasificación binaria y multiclase.

  Documentación: https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LogisticRegression.html

- sklearn.preprocessing.LabelEncoder: Se utiliza para codificar etiquetas objetivo con valores entre 0 y `n_classes-1`.

  Documentación: https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.LabelEncoder.html

- sklearn.model_selection.train_test_split: Divide arrays o matrices en subconjuntos aleatorios de entrenamiento y prueba.

  Documentación: https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.train_test_split.html

- sklearn.metrics.confusion_matrix: Calcula una matriz de confusión para evaluar la precisión de una clasificación.

Documentación: https://scikit-learn.org/stable/modules/generated/sklearn.metrics.confusion_matrix.html

- sklearn.metrics.classification_report: Construye un informe de texto que muestra las principales métricas de clasificación.

  Documentación: https://scikit-learn.org/stable/modules/generated/sklearn.metrics.classification_report.html

- sklearn.metrics.accuracy_score: Calcula la precisión de la clasificación.

  Documentación: https://scikit-learn.org/stable/modules/generated/sklearn.metrics.accuracy_score.html

- plotly.io: Módulo para leer y escribir figuras en diferentes formatos.

  Documentación: https://plotly.com/python-api-reference/generated/plotly.io.html

- imblearn.over_sampling.RandomOverSampler: Una técnica para hacer *oversampling* de la clase minoritaria seleccionando muestras al azar con reemplazo.

  Documentación: https://imbalanced-learn.org/stable/references/generated/imblearn.over_sampling.RandomOverSampler.html

- sklearn.metrics.ConfusionMatrixDisplay: Visualización de la matriz de confusión.

  Documentación: https://scikit-learn.org/stable/modules/generated/sklearn.metrics.ConfusionMatrixDisplay.html

---