

Nombre del Proyecto

Análisis de Twitter durante la Copa Mundial

Nombre de la materia

Big Data

Nombre de la profesora

Tania Daniela Garcia Munoz

Nombres de los Integrantes

Kevin Rodolfo García Basurto

Jesús Adrián Magaña López

**Objetivo del Proyecto**

El objetivo principal de este proyecto es llevar a cabo un análisis en relación con los datos generados en la red social Twitter durante el transcurso de la Copa Mundial. Buscamos identificar y comparar métricas de comportamiento significativo entre los usuarios durante este periodo como por ejemplo, las palabras más comunes utilizadas, los hashtags más utilizados, el número de tweets publicados por dia, cuál de los sexos hacia más publicaciones, el tipo de plataforma que más se utilizó y palabras promedio por tweet. También utilizaremos métricas como cual sexo más utilizaba ciertas plataformas, los tweets más largos y los más cortos, verificar si algunos usuarios pertenecen a la categoría de spam, y por último una comparación de likes entre sexo, región y plataforma.

A través de la aplicación de técnicas de procesamiento de lenguaje natural se creará análisis, limpieza y casteo de datos para el cálculo de estadísticas descriptivas con el objetivo de crear visualizaciones informativas para revelar tendencias y relaciones durante el transcurso de la Copa Mundial, este proyecto se propone responder a las siguientes preguntas:

1. ¿Cuáles fueron las palabras y hashtags más usados? (Identificando los términos y etiquetas centrales de la conversación).
2. ¿Cuántos tweets se publicaron por día? (Analizando la actividad temporal y los picos de interés).
3. ¿Qué género publicó más? (Explorando la participación demográfica).
4. ¿Se puede detectar contenido potencialmente considerado spam? (Identificando cuentas o mensajes con comportamiento atípico o no deseado).
5. ¿Qué regiones y plataformas se utilizaron más? (Mapeando la distribución geográfica y las preferencias de dispositivos de los usuarios).
6. ¿Cuál es el promedio de palabras por tweet? (Cuantificando la longitud típica de los mensajes).
7. ¿Cuáles fueron los tweets más largos y más cortos? (Identificando los extremos en la longitud de los mensajes y su posible contexto).
8. ¿Cómo varían los likes por género, región y plataforma? (Analizando las diferencias en la recepción de los tweets según estas características).

**Flujo de Trabajo**

Fuente de Datos

La fuente utilizada fue un archivo CSV llamado mundial\_tweets.csv, que contiene datos recolectados de Twitter relacionados con la Copa Mundial. Este dataset incluye tanto contenido textual (tweets) como metadatos (fecha, nombre, hashtags, número de likes, lugar, etc.).

Datos de Entrada

Cada tweet contendrá una variedad de información, incluyendo, pero no limitado a:

|  |  |
| --- | --- |
| **Columna** | **Descripción** |
| Tweet | Texto del tweet publicado |
| Name | Nombre del usuario |
| Date | Fecha y hora del tweet |
| Likes | Número de likes que recibió el tweet |
| Followers | Número de seguidores del usuario |
| Friends | Número de amigos del usuario |
| Place | Región o país desde donde se publicó |
| Hashtags | Hashtags utilizados en el tweet |
| Source | Plataforma o dispositivo desde el cual se publicó el tweet |

Datos de salida

Después del procesamiento y análisis, se generan:

* Visualizaciones gráficas (barras, líneas, pie charts) sobre tendencias.
* Columnas nuevas para análisis: Fecha, Hora, Sexo, Plataforma, Palabras\_por\_tweet, Es\_spam, Longitud, etc.
* Resultados numéricos impresos en consola: promedios, máximos, mínimos, porcentajes y contadores.

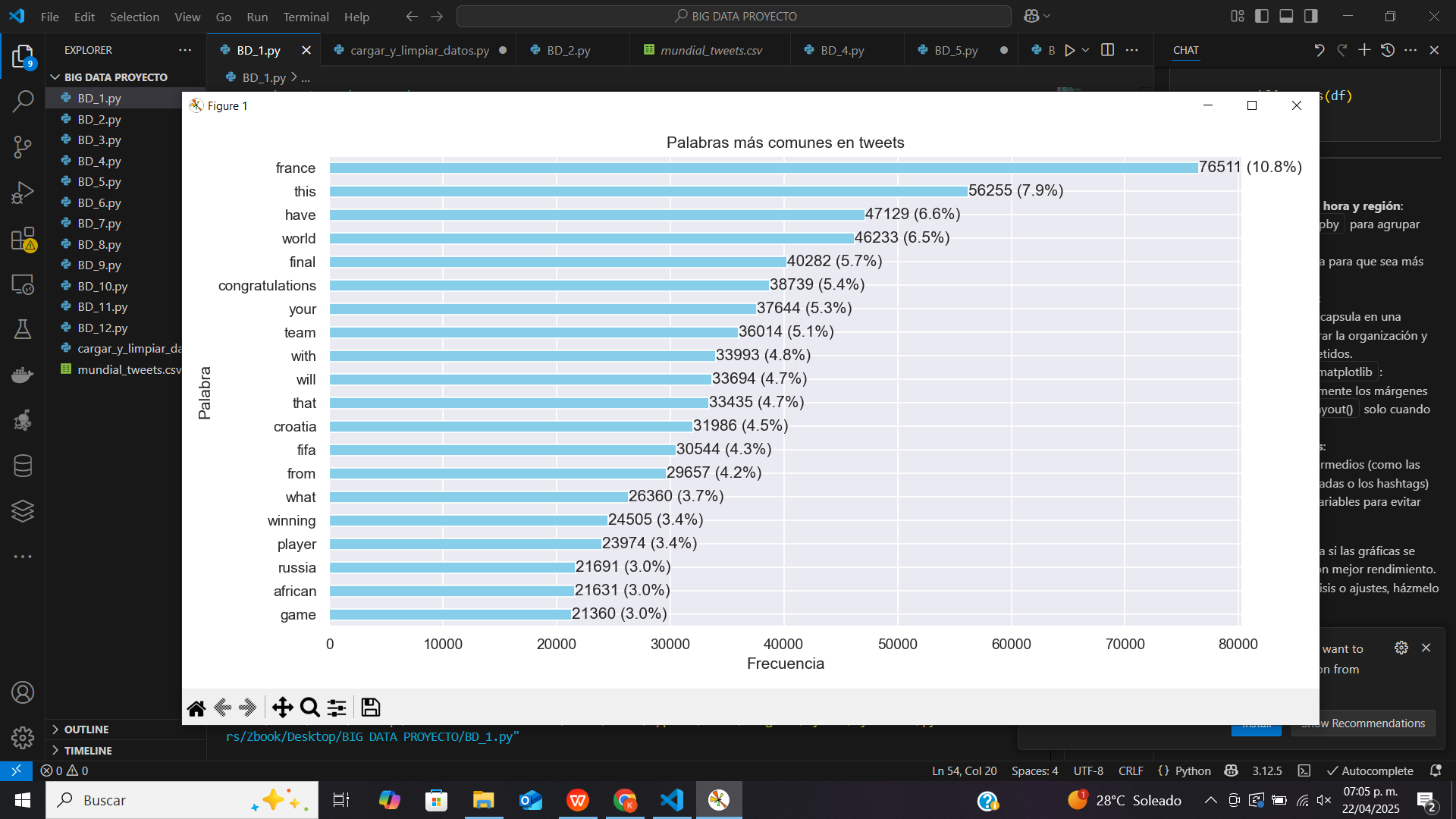
Transformación de Datos

|  |  |
| --- | --- |
| **Proceso** | **Descripción** |
| **Conversión de fechas** | Se separan fecha (Fecha) y hora (Hora) de la columna Date original. |
| **Normalización de texto** | Se convierten los tweets a minúsculas y se eliminan signos de puntuación y palabras vacías (stopwords). |
| **Extracción de hashtags** | Se separan múltiples hashtags en listas y se transforman a minúsculas. |
| **Clasificación de plataformas** | Se agrupan las fuentes en categorías: iPhone, Android, Web, iPad y Otro. |
| **Detección de spam** | Se marca como spam a cuentas con pocos seguidores y amigos. |
| **Predicción de género** | Se infiere el sexo del usuario a partir del primer nombre usando gender-guesser. |
| **Cálculo de métricas** | Se generan columnas como longitud del tweet, número de palabras por tweet, etc. |

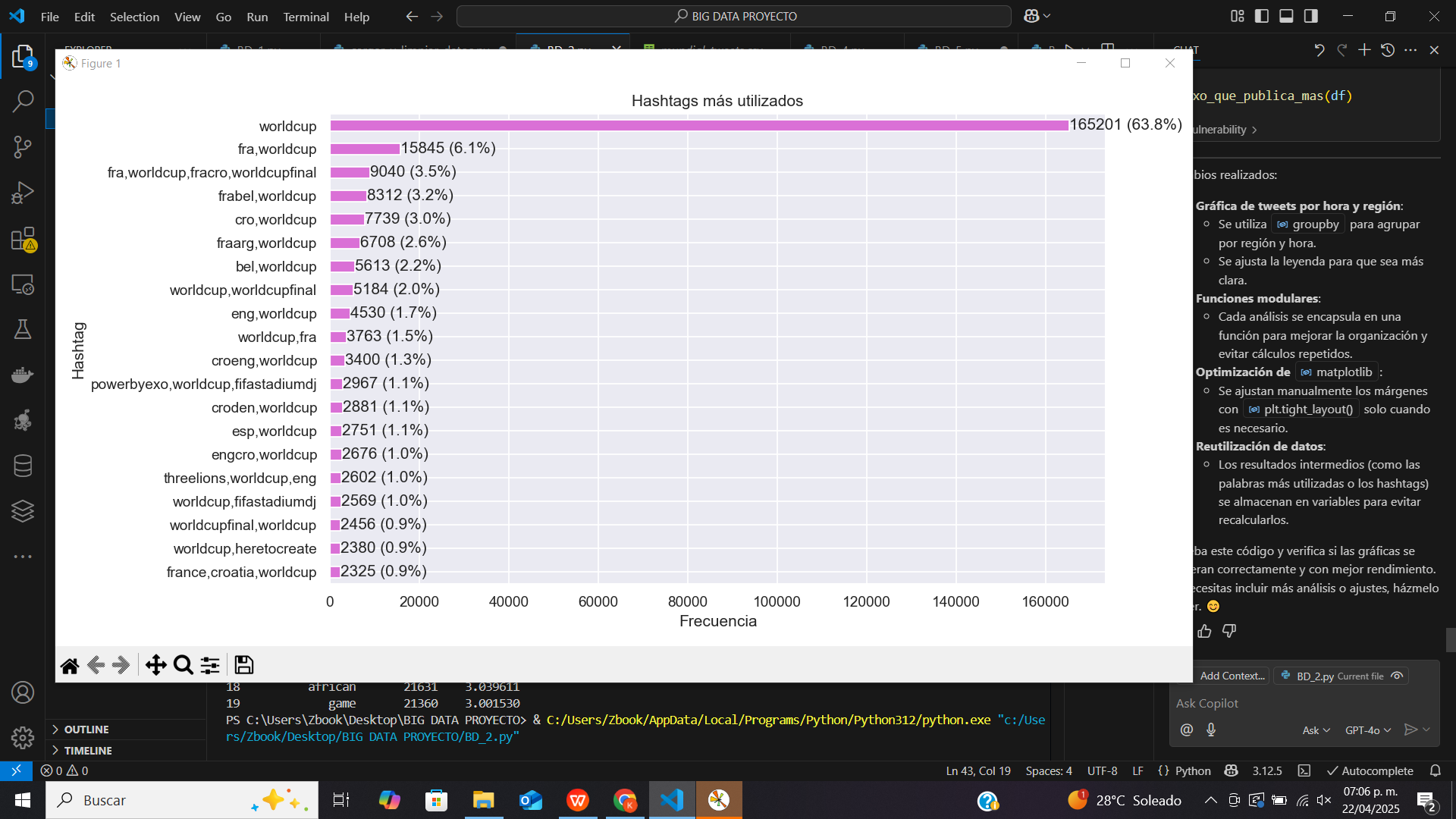
Análisis de Datos

|  |  |
| --- | --- |
| **Análisis** | **Objetivo** |
| **Palabras y hashtags más comunes** | Identificar temas frecuentes en los tweets. |
| **Actividad por día y hora** | Detectar los momentos de mayor interacción en Twitter. |
| **Género de los usuarios** | Conocer la proporción de publicaciones por sexo. |
| **Plataforma más utilizada** | Saber desde qué dispositivos se publica más. |
| **Tweets extremos (largos/cortos)** | Observar la variedad en la longitud de los mensajes. |
| **Likes promedio** | Evaluar el impacto de los tweets según el sexo, región o plataforma. |

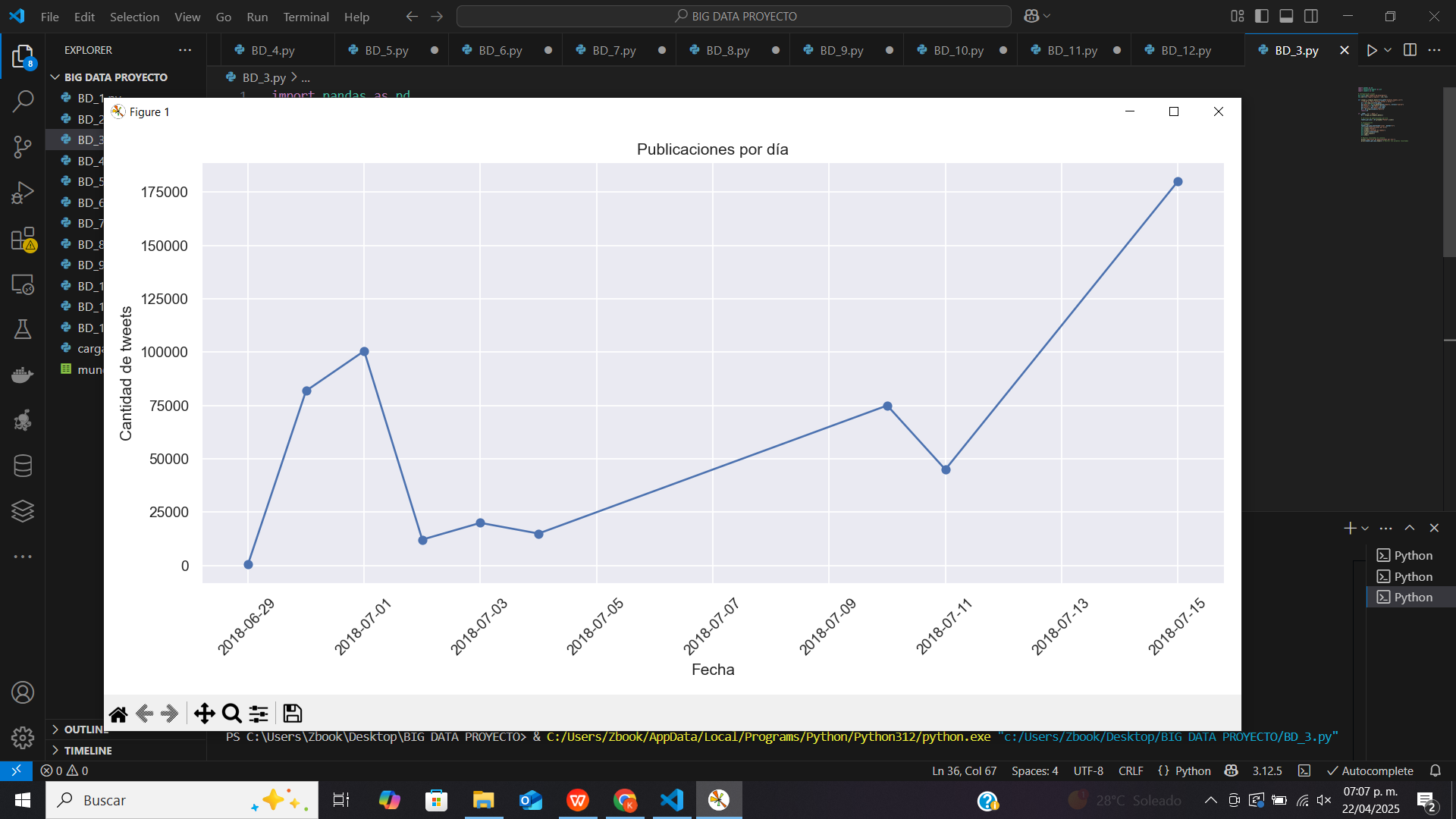
Visualizaciones



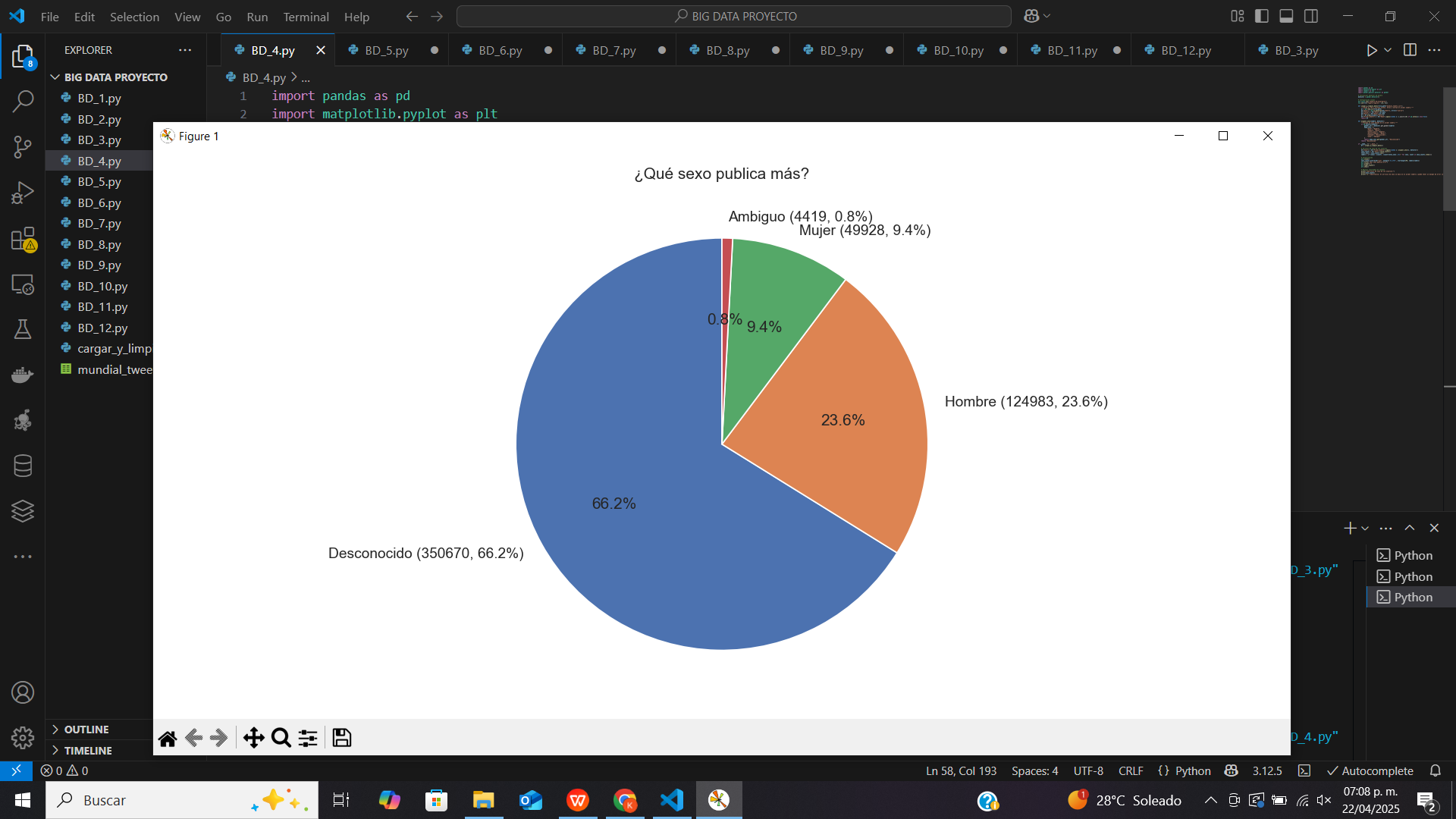
Como se puede apreciar en esta gráfica, la palabra france (El nombre de Francia en ingles) es la común en los tweets, por lo que podemos inferir que Francia es el pais que probablemente gano el mundial y que parece ser un equipo favorito de los espectadores debido a que las siguientes palabras más populares tienen implicaciones positivas.



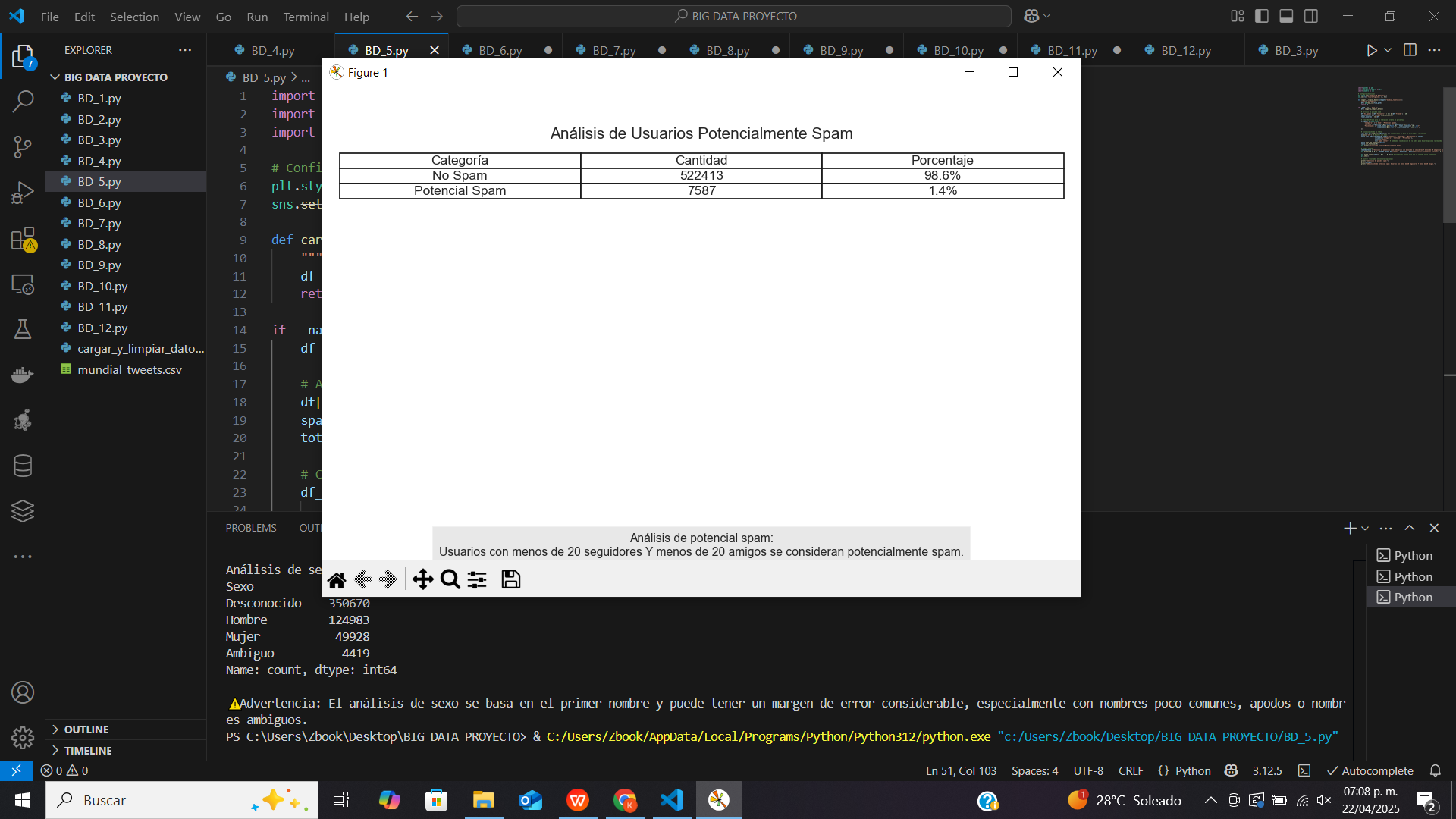
En la siguiente gráfica se visualiza los hashtags más utilizados durante el periodo de la copa mundial, viendo las gráficas podemos teorizar de que la mayoría de los hashtags son simplemente menciones de la copa mundial o menciones de países relevantes de la copa mundial, probablemente los que llegaron a los finales y semifinales.



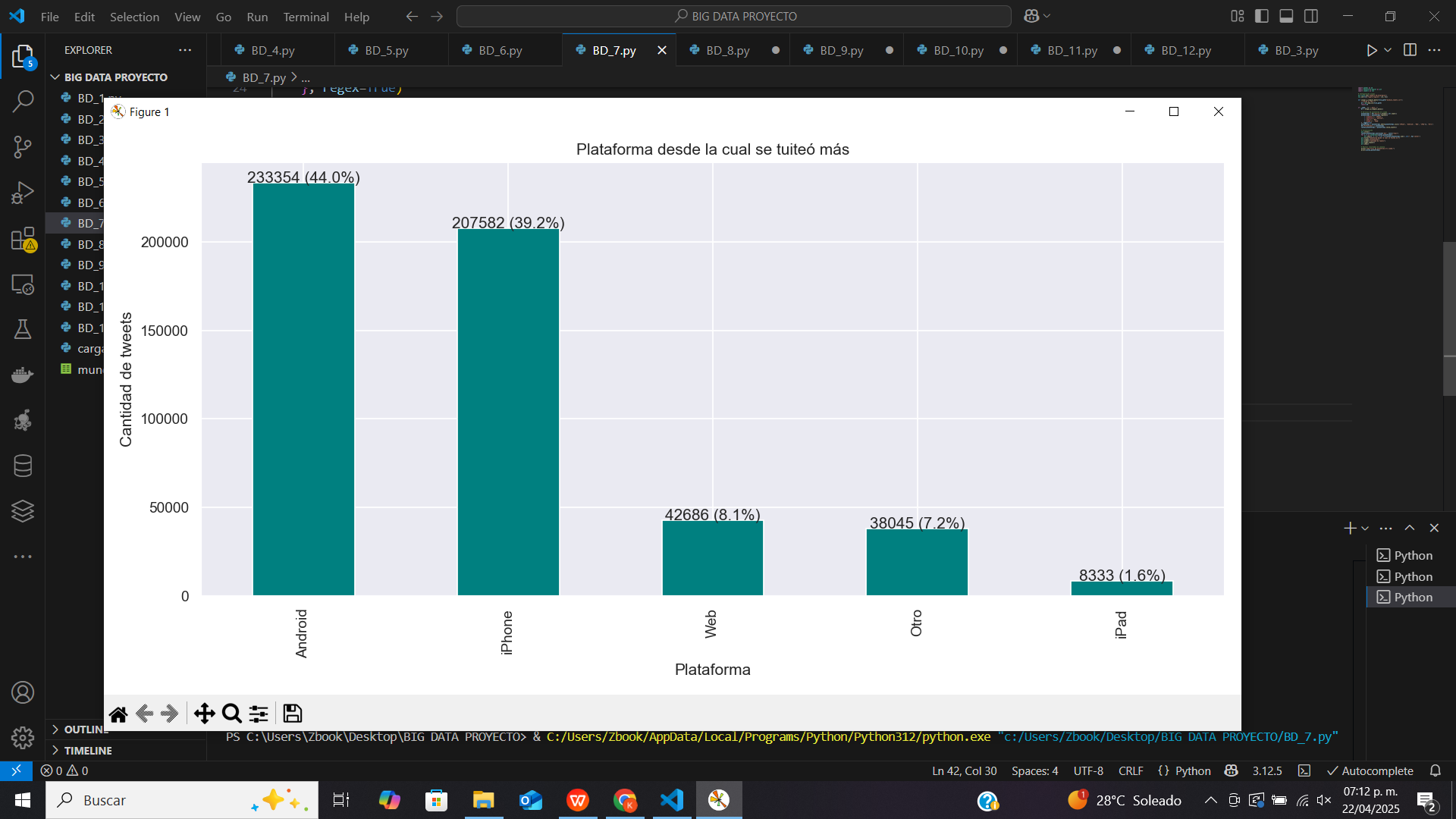
La siguiente gráfica muestra las publicaciones por dia, aquí podemos apreciar que las primeras fechas existe una cantidad decente de tweets relacionados con el mundial, pero al pasar el tiempo las publicaciones toman una bajada considerable al inicio del mes de julio. Este ritmo se empieza a elevar de manera gradual con cada dia que pasa, después entre el dia 9 y 11 de julio hay una bajada momentánea, sin embargo al iniciar el 11 de julio, el número de tweets se dispara hacia arriba de manera explosiva. De esto podemos inferir que un evento importante ocurrió ese dia, por ejemplo uno de los equipos favoritos metió un gol ganador o gano un juego importante.



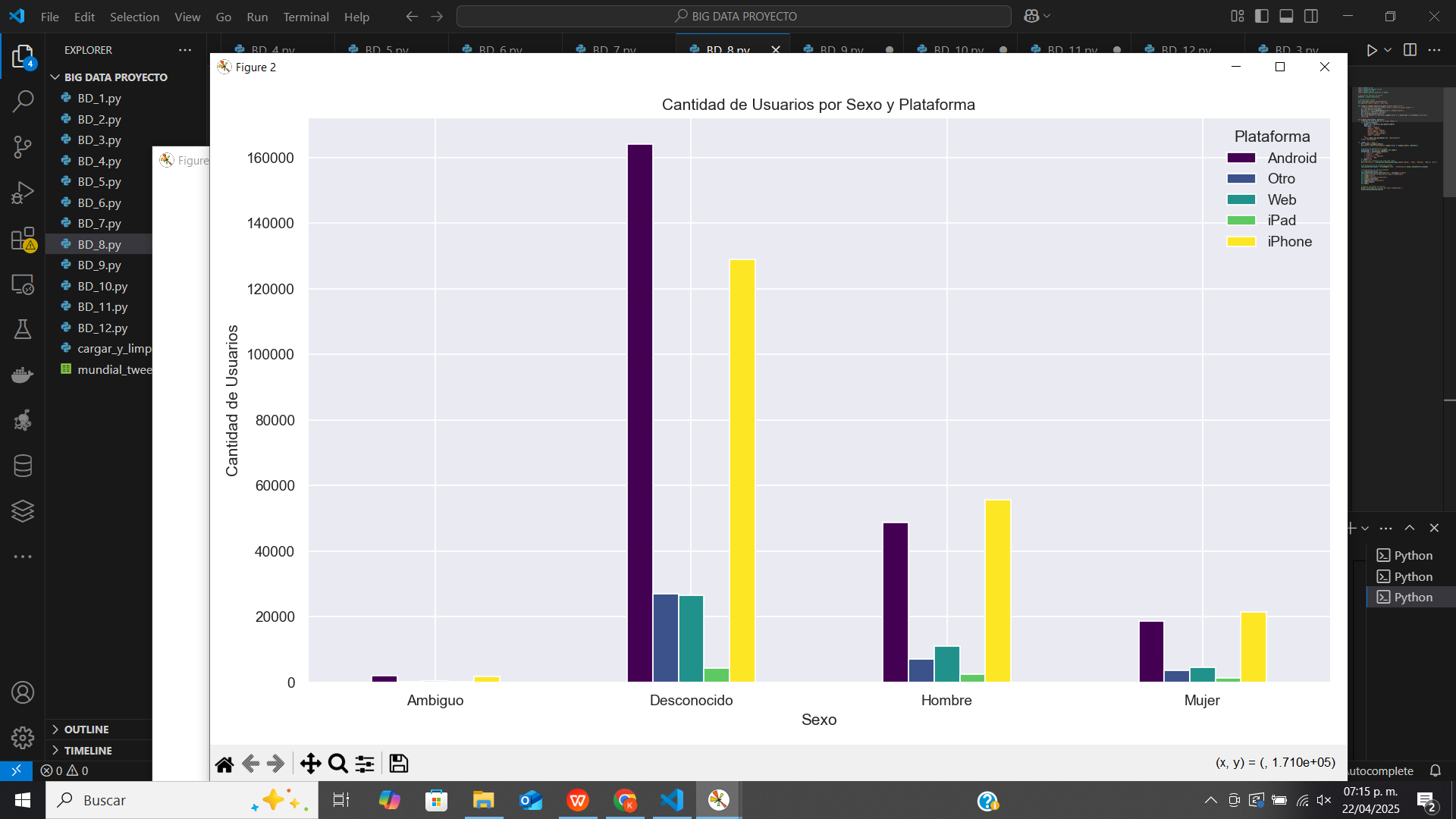
La siguiente gráfica muestra los sexos que más hicieron publicaciones, esta gráfica muestra que la mayoría de las personas que publican en Twitter no especifican su sexo, la segunda categoría más grande es la de hombres, la tercera es de mujeres y la última y más pequeña son de sexo ambiguo. Con esto podríamos teorizar que la mayoría de la gente que hace publicaciones prefiere o no le interesa que otras personas sepan que sexo son, esto podria deberse a miedo o indiferencia.

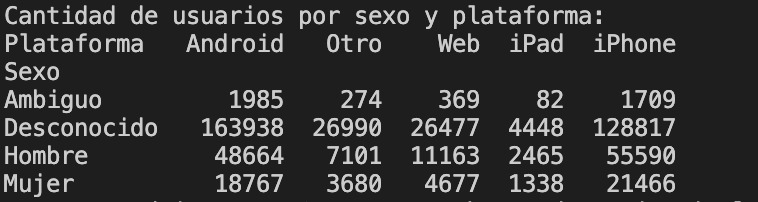


Este gráfico muestra el análisis de usuarios que podrias ser spam potencialmente, esto se debe a que este tipo de páginas suelen atraer robots debido al gran número de usuarios que tiene. Sin embargo podemos ver gracias al análisis que la mayoría de los tweets no parecen ser de cuentas spam, el número ni siquiera llega a ser el 2%.

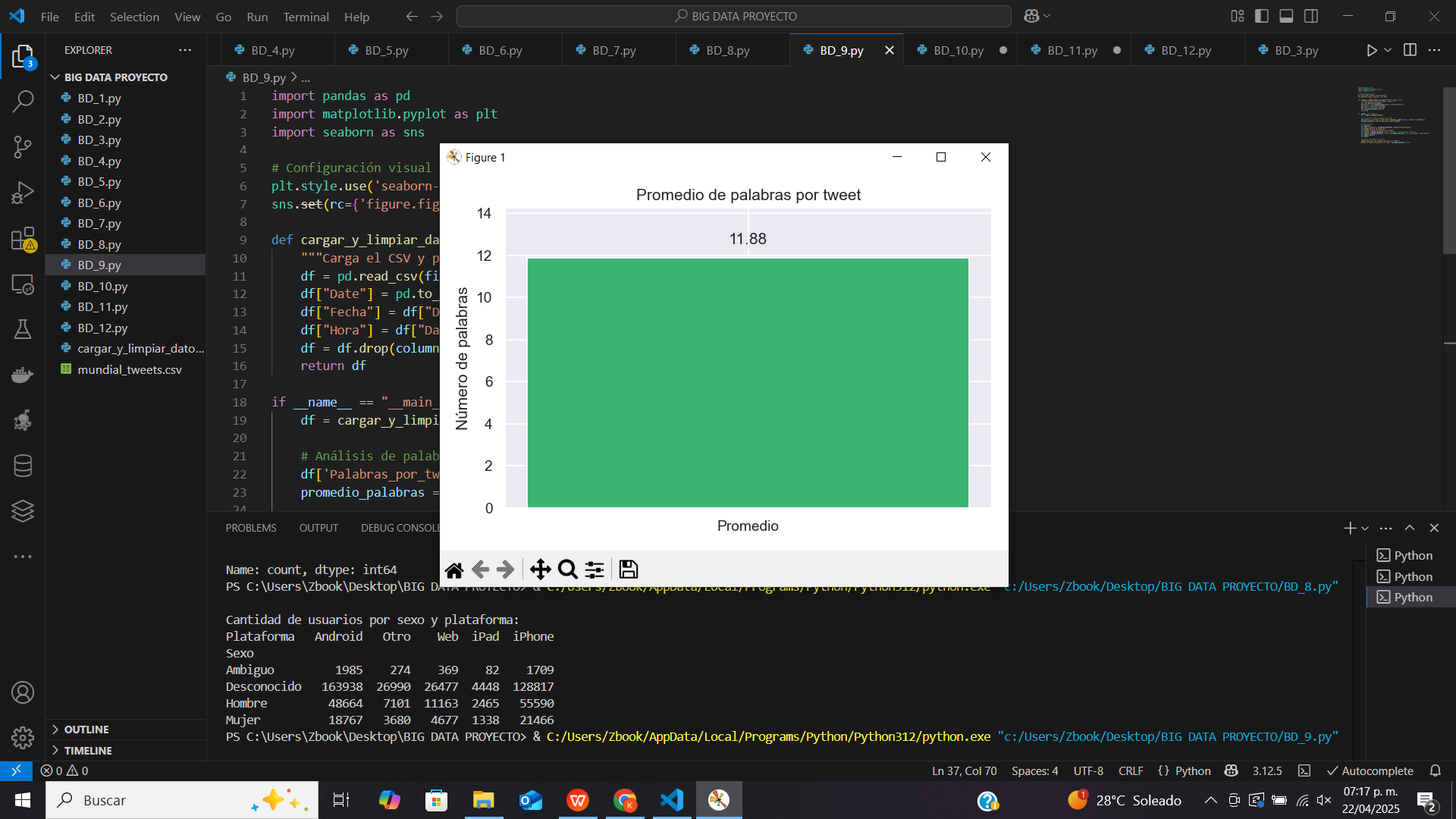


La siguiente gráfica muestra las plataformas más populares que se utilizaron para hacer publicaciones en Twitter durante el periodo de la copa mundial. Como podemos ver tanto Android e iPhone son las plataformas más populares para el uso de Twitter, esto implica que la mayoría de las personas estan realizando sus publicaciones a traves de sus teléfonos. Esto podria deberse a que los teléfonos son portátiles y cómodos de usar a comparación con otras plataformas.

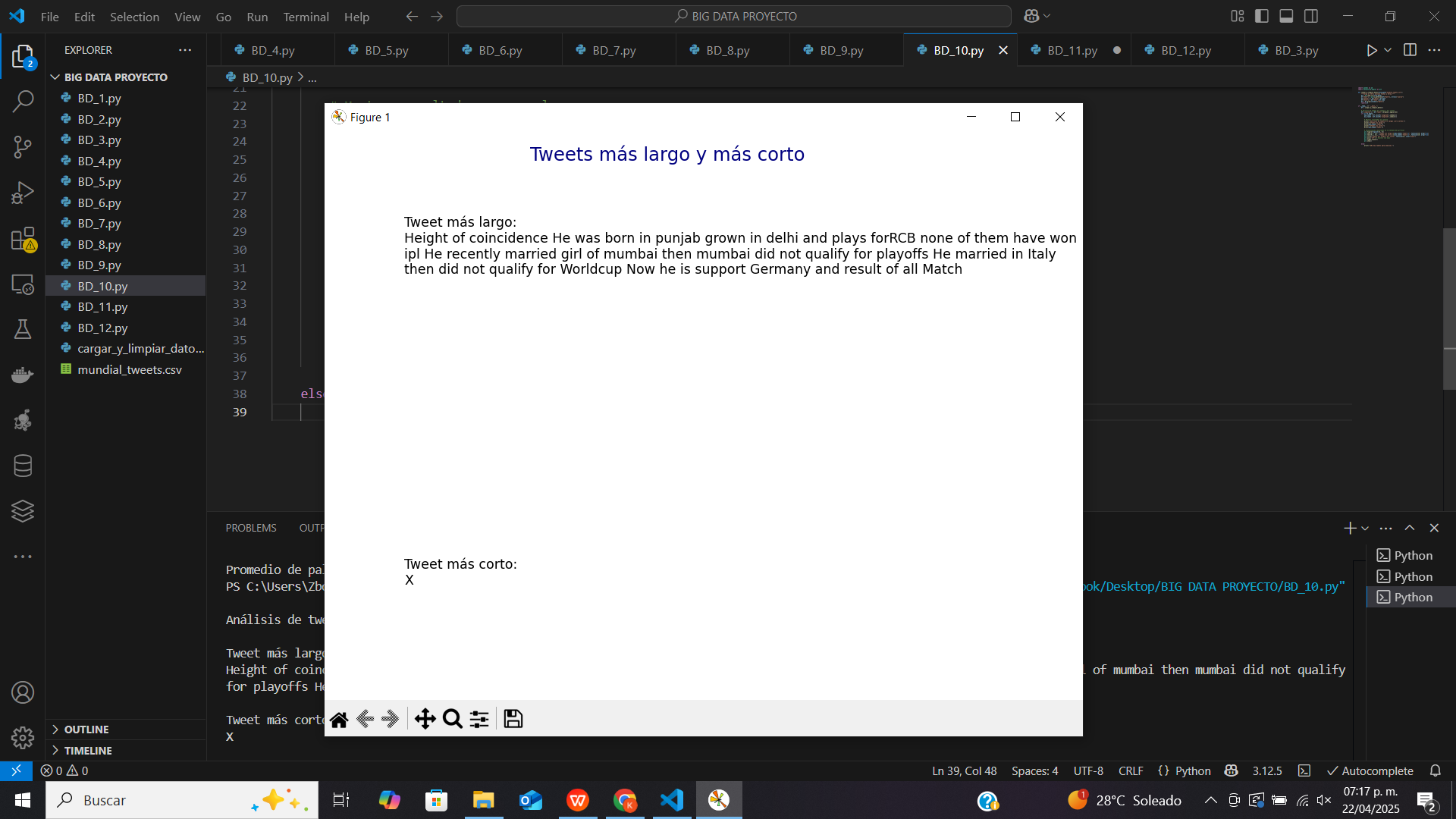




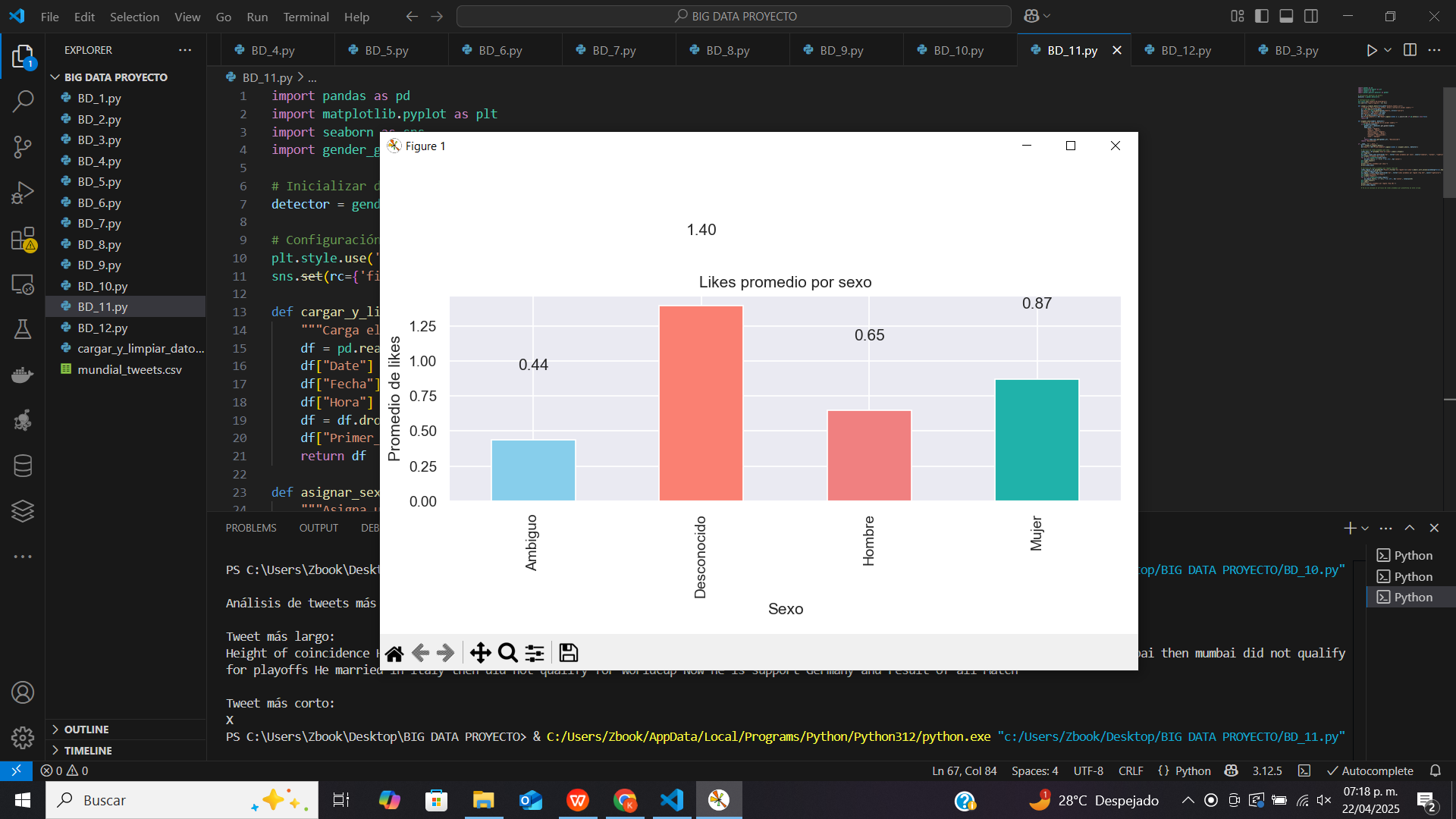
Esta siguiente gráfica muestra una comparación entre que sexos utilizan más ciertas plataformas, esta información muestra que a pesar de que los hombres y de sexo conocido utilizan otras plataformas de manera considerable, el teléfono sigue siendo la plataforma preferida para el uso de Twitter a traves de todos los sexos.

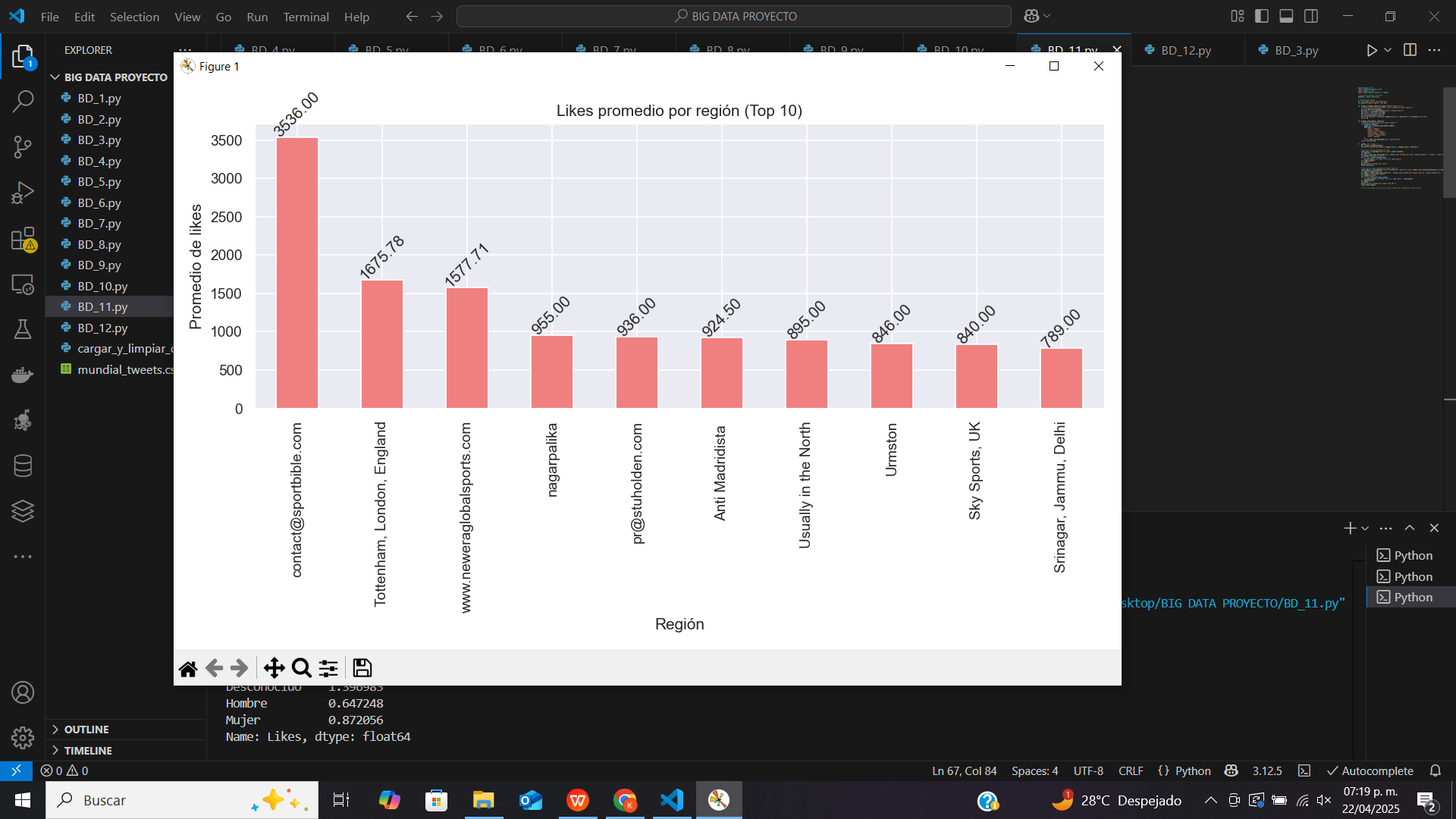


Esta grafica solo muestra las palabras promedio por tweet, esta informacion implica que la mayoria de publicaciones son cortas, lo que probablemente indica que la mayoria podrian ser celebraciones para los equipos debido a que las menciones tambien son incluidas en las palabras.



Esta siguiente parte solo muestra el tweet más largo y el tweet más corto, el más largo probablemente se refiere a la historia interesante de un jugador o fan del deporte, mientras que el más corto solo muestra la palabra X lo que realmente no ayuda mucho al momento de analizar el propósito de la publicación a comparación del tweet largo.





Por último tenemos unas gráficas que hacen comparaciones de los likes entre sexos y regiones. Por lo que se puede observar la mayoría de los likes por publicaciones pertenecen al grupo de desconocido y al de mujeres. Mientras que el grupo de desconocido podria deberse a que ese grupo pertenecé a la mayoría de los usuarios, el hecho de que el grupo de mujeres tenga más likes que el de hombres es interesante debido a que este grupo es menor que el de hombres. Podríamos teorizar que el grupo de mujeres tiene más likes porque talvez sus mensajes podrían ser más positivos que el de los hombres. Después la siguiente y última gráfica muestra que regiones tuvieron más likes por promedio, sin embargo como también hay sitios web mezclados con las regiones, es un poco difícil hacer un análisis conclusivo. Pero si ignoramos las páginas web o términos, la mayoría de las regiones parecen ser del Reino Unido y la segunda región más popular parece ser China.

Conclusiones

Tras realizar un análisis profundo de más de 500,000 tweets relacionados con la Copa Mundial, se pueden extraer las siguientes conclusiones significativas:

1. Contenido y tendencias en el discurso

Las palabras más utilizadas reflejan una fuerte presencia de términos relacionados con equipos, jugadores y emociones comunes durante partidos intensos.

Los hashtags más utilizados permitieron identificar los temas más virales, campañas promocionales y etiquetas oficiales del evento, confirmando su valor como indicadores de conversación en tiempo real.

2. Comportamiento temporal de los usuarios

El análisis por hora reveló picos de publicación durante el inicio del mes de julio, lo que sugiere que los usuarios tienden a comentar en tiempo real durante los partidos o inmediatamente después.

3. Demografía de los usuarios

A través del análisis de nombres, se detectó que una mayoría de los tweets provienen de usuarios clasificados como desconocidos, aunque también existe una porción significativa de Hombres, Mujeres.

Este enfoque basado en genero tiene limitaciones, por lo que debe considerarse como una aproximación general.

4. Posible actividad de cuentas spam

Se identificó un número negligente de cuentas con baja interacción (menos de 20 seguidores y amigos), lo que podría indicar poca actividad automatizada o cuentas falsas.

5. Uso de plataformas

Se confirmó que los usuarios acceden principalmente desde dispositivos móviles, siendo los sistemas operativos iOS y Android los más predominantes.

La plataforma Web se usa en menor medida, pero aún relevante, y se detectaron otras plataformas como iPad o no identificadas agrupadas como “Otro”.

6. Sexo vs Plataforma

Se hallaron patrones interesantes como la mayoría de los generos utilizando las plataformas de iPhone Android a comparación con las otras plataformas.

7. Estructura y estilo de los tweets

El promedio de palabras por tweet fue de aproximadamente 11.88 palabras.

8. Se identificaron tweets extremadamente largos y cortos, lo cual sugiere distintos estilos de comunicación: algunos más informativos y otros más expresivos o simbólicos.

9. Interacción con el contenido

El promedio de likes por tweet varía considerablemente según:

Sexo del usuario

Región geográfica

Plataforma utilizada

Esto muestra que hay factores sociales y tecnológicos que influyen en el alcance o recepción de los tweets.

Codigo de python

**Cargar y limpiar datos**

import pandas as pd

import nltk

import re

from nltk.corpus import stopwords

import gender\_guesser.detector as gender

# Descargar stopwords si no están descargados

try:

stopwords.words('spanish')

except LookupError:

nltk.download('stopwords')

stopwords\_es = set(stopwords.words('spanish'))

# Inicializar detector de género

detector = gender.Detector()

def cargar\_y\_limpiar\_datos(file\_path="mundial\_tweets.csv"):

"""Carga el CSV, procesa fechas y horas."""

df = pd.read\_csv(file\_path)

df["Date"] = pd.to\_datetime(df["Date"], errors="coerce")

df["Fecha"] = df["Date"].dt.date

df["Hora"] = df["Date"].dt.time

df = df.drop(columns=["Date"])

return df

def extraer\_primer\_nombre(nombre):

"""Extrae el primer nombre de una cadena."""

if pd.isna(nombre):

return None

return nombre.split()[0]

def asignar\_sexo(nombre, detector):

"""Asigna un sexo basado en el primer nombre."""

if pd.notnull(nombre):

gender\_str = detector.get\_gender(nombre)

mapa\_sexo = {

"male": "Hombre",

"female": "Mujer",

"mostly\_male": "Hombre",

"mostly\_female": "Mujer",

"unknown": "Desconocido",

"andy": "Ambiguo"

}

return mapa\_sexo.get(gender\_str, "Desconocido")

return "Desconocido"

if \_\_name\_\_ == "\_\_main\_\_":

df\_base = cargar\_y\_limpiar\_datos()

print("Datos cargados y limpieza básica realizada.")

**Análisis de palabras más utilizadas**

import pandas as pd

import matplotlib.pyplot as plt

import seaborn as sns

from collections import Counter

import re

from nltk.corpus import stopwords

# Descargar stopwords

try:

stopwords.words('spanish')

except LookupError:

import nltk

nltk.download('stopwords')

stopwords\_es = set(stopwords.words('spanish'))

# Configuración visual

plt.style.use('seaborn-v0\_8-darkgrid')

sns.set(rc={'figure.figsize': (12, 6)})

def cargar\_y\_limpiar\_datos(file\_path="mundial\_tweets.csv"):

"""Carga el CSV y procesa fechas y horas."""

df = pd.read\_csv(file\_path)

df["Date"] = pd.to\_datetime(df["Date"], errors="coerce")

df["Fecha"] = df["Date"].dt.date

df["Hora"] = df["Date"].dt.time

df = df.drop(columns=["Date"])

return df

if \_\_name\_\_ == "\_\_main\_\_":

df = cargar\_y\_limpiar\_datos()

# Análisis de palabras más utilizadas

all\_words = ' '.join(df['Tweet'].dropna()).lower()

all\_words = re.findall(r'\b\w+\b', all\_words)

filtered\_words = [w for w in all\_words if w not in stopwords\_es and len(w) > 3]

word\_freq = Counter(filtered\_words).most\_common(20)

df\_words = pd.DataFrame(word\_freq, columns=["Palabra", "Frecuencia"])

df\_words['Porcentaje'] = df\_words['Frecuencia'] / df\_words['Frecuencia'].sum() \* 100

# Visualización

plt.close()

df\_words.plot(kind="barh", x="Palabra", y="Frecuencia", legend=False, color="skyblue")

for i, (freq, pct) in enumerate(zip(df\_words['Frecuencia'], df\_words['Porcentaje'])):

plt.text(freq + 1, i, f'{freq} ({pct:.1f}%)', va='center')

plt.title("Palabras más comunes en tweets")

plt.xlabel("Frecuencia")

plt.gca().invert\_yaxis()

plt.tight\_layout()

plt.show()

# Mostrar resultados en consola

print("\nAnálisis de palabras más utilizadas:")

print(df\_words)

**Análisis de hashtags más utilizados**

import pandas as pd

import matplotlib.pyplot as plt

import seaborn as sns

from collections import Counter

# Configuración visual

plt.style.use('seaborn-v0\_8-darkgrid')

sns.set(rc={'figure.figsize': (12, 6)})

def cargar\_y\_limpiar\_datos(file\_path="mundial\_tweets.csv"):

"""Carga el CSV y procesa fechas y horas."""

df = pd.read\_csv(file\_path)

df["Date"] = pd.to\_datetime(df["Date"], errors="coerce")

df["Fecha"] = df["Date"].dt.date

df["Hora"] = df["Date"].dt.time

df = df.drop(columns=["Date"])

return df

if \_\_name\_\_ == "\_\_main\_\_":

df = cargar\_y\_limpiar\_datos()

# Análisis de hashtags más utilizados

hashtags = df['Hashtags'].dropna().astype(str).str.lower().str.split(', ')

hashtags\_flat = [ht for sublist in hashtags for ht in sublist if ht]

hashtag\_freq = Counter(hashtags\_flat).most\_common(20)

df\_hash = pd.DataFrame(hashtag\_freq, columns=["Hashtag", "Frecuencia"])

df\_hash['Porcentaje'] = df\_hash['Frecuencia'] / df\_hash['Frecuencia'].sum() \* 100

# Visualización

plt.close()

df\_hash.plot(kind="barh", x="Hashtag", y="Frecuencia", legend=False, color="orchid")

for i, (freq, pct) in enumerate(zip(df\_hash['Frecuencia'], df\_hash['Porcentaje'])):

plt.text(freq + 1, i, f'{freq} ({pct:.1f}%)', va='center')

plt.title("Hashtags más utilizados")

plt.xlabel("Frecuencia")

plt.gca().invert\_yaxis()

plt.tight\_layout()

plt.show()

# Mostrar resultados en consola

print("\nAnálisis de hashtags más utilizados:")

print(df\_hash)

**Análisis de publicaciones por día**

import pandas as pd

import matplotlib.pyplot as plt

import seaborn as sns

# Configuración visual

plt.style.use('seaborn-v0\_8-darkgrid')

sns.set(rc={'figure.figsize': (12, 6)})

def cargar\_y\_limpiar\_datos(file\_path="mundial\_tweets.csv"):

"""Carga el CSV y procesa fechas y horas."""

df = pd.read\_csv(file\_path)

df["Date"] = pd.to\_datetime(df["Date"], errors="coerce")

df["Fecha"] = df["Date"].dt.date

df["Hora"] = df["Date"].dt.time

df = df.drop(columns=["Date"])

return df

if \_\_name\_\_ == "\_\_main\_\_":

df = cargar\_y\_limpiar\_datos()

# Análisis de publicaciones por día

tweets\_por\_dia = df.groupby("Fecha").size()

# Visualización

plt.close()

tweets\_por\_dia.plot(kind="line", marker="o")

plt.title("Publicaciones por día")

plt.xlabel("Fecha")

plt.ylabel("Cantidad de tweets")

plt.xticks(rotation=45)

plt.tight\_layout()

plt.show()

# Mostrar resultados en consola

print("\nAnálisis de publicaciones por día:")

print(tweets\_por\_dia.head()) # Mostrar los primeros resultados

**Análisis de sexo de los usuarios**

import pandas as pd

import matplotlib.pyplot as plt

import seaborn as sns

import gender\_guesser.detector as gender

# Inicializar detector de género

detector = gender.Detector()

# Configuración visual

plt.style.use('seaborn-v0\_8-darkgrid')

sns.set(rc={'figure.figsize': (12, 6)})

def cargar\_y\_limpiar\_datos(file\_path="mundial\_tweets.csv"):

"""Carga el CSV y procesa fechas, horas y extrae el primer nombre."""

df = pd.read\_csv(file\_path)

df["Date"] = pd.to\_datetime(df["Date"], errors="coerce")

df["Fecha"] = df["Date"].dt.date

df["Hora"] = df["Date"].dt.time

df = df.drop(columns=["Date"])

df["Primer\_Nombre"] = df["Name"].apply(lambda x: x.split()[0] if pd.notna(x) else None)

return df

def asignar\_sexo(nombre, detector):

"""Asigna un sexo basado en el primer nombre."""

if pd.notnull(nombre):

gender\_str = detector.get\_gender(nombre)

mapa\_sexo = {

"male": "Hombre",

"female": "Mujer",

"mostly\_male": "Hombre",

"mostly\_female": "Mujer",

"unknown": "Desconocido",

"andy": "Ambiguo"

}

return mapa\_sexo.get(gender\_str, "Desconocido")

return "Desconocido"

if \_\_name\_\_ == "\_\_main\_\_":

df = cargar\_y\_limpiar\_datos()

# Análisis de sexo de los usuarios

df["Sexo"] = df["Primer\_Nombre"].apply(lambda x: asignar\_sexo(x, detector))

sexo\_counts = df['Sexo'].value\_counts()

total\_sexo = sexo\_counts.sum()

labels = [f"{sexo} ({count}, {count/total\_sexo:.1%})" for sexo, count in sexo\_counts.items()]

# Visualización

plt.close()

sexo\_counts.plot(kind="pie", autopct='%1.1f%%', startangle=90, labels=labels)

plt.title("¿Qué sexo publica más?")

plt.ylabel("")

plt.tight\_layout()

plt.show()

# Mostrar resultados en consola

print("\nAnálisis de sexo de los usuarios:")

print(sexo\_counts)

print("\n⚠️ Advertencia: El análisis de sexo se basa en el primer nombre y puede tener un margen de error considerable, especialmente con nombres poco comunes, apodos o nombres ambiguos.")

**Análisis de posible spam**

import pandas as pd

import matplotlib.pyplot as plt

import seaborn as sns

# Configuración visual

plt.style.use('seaborn-v0\_8-darkgrid')

sns.set(rc={'figure.figsize': (6, 4)})

def cargar\_y\_limpiar\_datos(file\_path="mundial\_tweets.csv"):

"""Carga el CSV."""

df = pd.read\_csv(file\_path)

return df

if \_\_name\_\_ == "\_\_main\_\_":

df = cargar\_y\_limpiar\_datos()

# Análisis de posible spam

df['Es\_spam'] = (df['Followers'] < 20) & (df['Friends'] < 20)

spam\_counts = df['Es\_spam'].value\_counts()

total\_usuarios = len(df)

# Crear DataFrame para la tabla con formato de porcentaje

df\_spam = pd.DataFrame({

'Categoría': ['No Spam', 'Potencial Spam'],

'Cantidad': [spam\_counts.get(False, 0), spam\_counts.get(True, 0)],

'Porcentaje': [f"{spam\_counts.get(False, 0) / total\_usuarios \* 100:.1f}%",

f"{spam\_counts.get(True, 0) / total\_usuarios \* 100:.1f}%"]

})

# Visualización de la tabla

fig, ax = plt.subplots(figsize=(8, 5)) # Aumentamos un poco la altura para la leyenda

ax.axis('off') # Ocultar los ejes

tabla = ax.table(cellText=df\_spam[['Categoría', 'Cantidad', 'Porcentaje']].values,

colLabels=['Categoría', 'Cantidad', 'Porcentaje'],

cellLoc='center',

loc='upper center') # Cambiamos la ubicación de la tabla para dejar espacio a la leyenda

tabla.auto\_set\_font\_size(False)

tabla.set\_fontsize(10)

plt.title("Análisis de Usuarios Potencialmente Spam")

# Añadir leyenda

leyenda\_texto = "Análisis de potencial spam:\nUsuarios con menos de 20 seguidores Y menos de 20 amigos se consideran potencialmente spam."

plt.figtext(0.5, 0.01, leyenda\_texto, ha="center", fontsize=9, bbox={"facecolor":"lightgrey", "alpha":0.5, "pad":5})

plt.tight\_layout(rect=[0, 0.1, 1, 0.95]) # Ajustamos el layout para que la leyenda no se superponga

plt.show()

# Mostrar resultados en consola (opcional)

print("\nAnálisis de posible spam:")

print(df\_spam)

print("\nDefinición de potencial spam: Usuarios con menos de 20 seguidores Y menos de 20 amigos.")

**Análisis de plataforma más usada**

import pandas as pd

import matplotlib.pyplot as plt

import seaborn as sns

# Configuración visual

plt.style.use('seaborn-v0\_8-darkgrid')

sns.set(rc={'figure.figsize': (12, 6)})

def cargar\_y\_limpiar\_datos(file\_path="mundial\_tweets.csv"):

"""Carga el CSV."""

df = pd.read\_csv(file\_path)

return df

if \_\_name\_\_ == "\_\_main\_\_":

df = cargar\_y\_limpiar\_datos()

# Análisis de plataforma más usada

plataformas = df['Source'].dropna().str.lower()

plataformas = plataformas.replace({

r'.\*iphone.\*': 'iPhone',

r'.\*android.\*': 'Android',

r'.\*web.\*': 'Web',

r'.\*ipad.\*': 'iPad'

}, regex=True)

plataformas = plataformas.where(plataformas.isin(['iPhone', 'Android', 'Web', 'iPad']), 'Otro')

df['Plataforma'] = plataformas

conteo\_plataformas = plataformas.value\_counts()

# Visualización

plt.close()

conteo\_plataformas.plot(kind='bar', color="teal")

for i, v in enumerate(conteo\_plataformas):

plt.text(i, v + 5, f"{v} ({v/conteo\_plataformas.sum():.1%})", ha='center')

plt.title("Plataforma desde la cual se tuiteó más")

plt.xlabel("Plataforma")

plt.ylabel("Cantidad de tweets")

plt.tight\_layout()

plt.show()

# Mostrar resultados en consola

print("\nAnálisis de plataforma más usada:")

print(conteo\_plataformas)

**Análisis de plataforma más usada**

import pandas as pd

import matplotlib.pyplot as plt

import seaborn as sns

import gender\_guesser.detector as gender

# Inicializar detector de género

detector = gender.Detector()

# Configuración visual

plt.style.use('seaborn-v0\_8-darkgrid')

sns.set(rc={'figure.figsize': (12, 7)})

def cargar\_y\_limpiar\_datos(file\_path="mundial\_tweets.csv"):

"""Carga el CSV y procesa fechas, horas y extrae el primer nombre."""

df = pd.read\_csv(file\_path)

df["Date"] = pd.to\_datetime(df["Date"], errors="coerce")

df["Fecha"] = df["Date"].dt.date

df = df.drop(columns=["Date"])

df["Primer\_Nombre"] = df["Name"].apply(lambda x: x.split()[0] if pd.notna(x) else None)

return df

def asignar\_sexo(nombre, detector):

"""Asigna un sexo basado en el primer nombre."""

if pd.notnull(nombre):

gender\_str = detector.get\_gender(nombre)

mapa\_sexo = {

"male": "Hombre",

"female": "Mujer",

"mostly\_male": "Hombre",

"mostly\_female": "Mujer",

"unknown": "Desconocido",

"andy": "Ambiguo"

}

return mapa\_sexo.get(gender\_str, "Desconocido")

return "Desconocido"

if \_\_name\_\_ == "\_\_main\_\_":

df = cargar\_y\_limpiar\_datos()

df["Sexo"] = df["Primer\_Nombre"].apply(lambda x: asignar\_sexo(x, detector))

# Análisis de plataforma más usada

plataformas = df['Source'].dropna().str.lower()

plataformas = plataformas.replace({

r'.\*iphone.\*': 'iPhone',

r'.\*ipad.\*': 'iPad',

r'.\*android.\*': 'Android',

r'.\*web.\*': 'Web'

}, regex=True)

# Categorizar cualquier otra cosa como 'Otro'

df['Plataforma'] = plataformas.where(plataformas.isin(['iPhone', 'iPad', 'Android', 'Web']), 'Otro')

# Análisis de sexo vs plataforma (conteo)

sexo\_plataforma\_counts = df.groupby(['Sexo', 'Plataforma']).size().unstack(fill\_value=0)

# Visualización con barras agrupadas

plt.figure(figsize=(12, 7))

sexo\_plataforma\_counts.plot(kind='bar', colormap="viridis")

plt.title("Cantidad de Usuarios por Sexo y Plataforma")

plt.xlabel("Sexo")

plt.ylabel("Cantidad de Usuarios")

plt.xticks(rotation=0)

plt.legend(title='Plataforma')

plt.tight\_layout()

plt.show()

# Mostrar resultados en consola

print("\nCantidad de usuarios por sexo y plataforma:")

print(sexo\_plataforma\_counts)

**Análisis de palabras promedio por tweet**

import pandas as pd

import matplotlib.pyplot as plt

import seaborn as sns

# Configuración visual

plt.style.use('seaborn-v0\_8-darkgrid')

sns.set(rc={'figure.figsize': (6, 4)})

def cargar\_y\_limpiar\_datos(file\_path="mundial\_tweets.csv"):

"""Carga el CSV y procesa fechas y horas."""

df = pd.read\_csv(file\_path)

df["Date"] = pd.to\_datetime(df["Date"], errors="coerce")

df["Fecha"] = df["Date"].dt.date

df["Hora"] = df["Date"].dt.time

df = df.drop(columns=["Date"])

return df

if \_\_name\_\_ == "\_\_main\_\_":

df = cargar\_y\_limpiar\_datos()

# Análisis de palabras promedio por tweet

df['Palabras\_por\_tweet'] = df['Tweet'].dropna().apply(lambda x: len(str(x).split()))

promedio\_palabras = df['Palabras\_por\_tweet'].mean()

# Visualización

plt.figure()

plt.bar(["Promedio"], [promedio\_palabras], color="mediumseagreen")

plt.ylabel("Número de palabras")

plt.title("Promedio de palabras por tweet")

plt.ylim(0, promedio\_palabras \* 1.2) # Ajustar límite superior del eje y

plt.text(0, promedio\_palabras + 0.5, f"{promedio\_palabras:.2f}", ha='center', va='bottom')

plt.tight\_layout()

plt.show()

# Mostrar resultados en consola

print("\nAnálisis de palabras promedio por tweet:")

print(f"Promedio de palabras por tweet: {promedio\_palabras:.2f}")

**Análisis de tweets más largos y más cortos**

import pandas as pd

import matplotlib.pyplot as plt

def cargar\_y\_limpiar\_datos(file\_path="mundial\_tweets.csv"):

"""Carga el CSV y procesa fechas y horas."""

df = pd.read\_csv(file\_path)

df["Date"] = pd.to\_datetime(df["Date"], errors="coerce")

df["Fecha"] = df["Date"].dt.date

df["Hora"] = df["Date"].dt.time

df = df.drop(columns=["Date"])

return df

if \_\_name\_\_ == "\_\_main\_\_":

df = cargar\_y\_limpiar\_datos()

# Análisis de tweets más largos y más cortos

df['Longitud'] = df['Tweet'].dropna().apply(len)

if not df.empty:

max\_tweet = df.loc[df['Longitud'].idxmax()]

min\_tweet = df.loc[df['Longitud'].idxmin()]

# Mostrar resultados en consola

print("\nAnálisis de tweets más largos y más cortos:")

print("\nTweet más largo:")

print(max\_tweet['Tweet'])

print("\nTweet más corto:")

print(min\_tweet['Tweet'])

# Visualización (mostrando en la ventana del gráfico)

plt.figure(figsize=(8, 6))

plt.text(0.1, 0.8, f"Tweet más largo:\n{max\_tweet['Tweet']}", fontsize=10, wrap=True)

plt.text(0.1, 0.2, f"Tweet más corto:\n{min\_tweet['Tweet']}", fontsize=10, wrap=True)

plt.title("Tweets más largo y más corto", fontsize=14, color="navy")

plt.axis('off') # Ocultar los ejes

plt.tight\_layout()

plt.show()

else:

print("\nNo hay tweets para analizar.")

**Análisis de likes promedio por sexo**

import pandas as pd

import matplotlib.pyplot as plt

import seaborn as sns

import gender\_guesser.detector as gender

# Inicializar detector de género

detector = gender.Detector()

# Configuración visual

plt.style.use('seaborn-v0\_8-darkgrid')

sns.set(rc={'figure.figsize': (8, 5)})

def cargar\_y\_limpiar\_datos(file\_path="mundial\_tweets.csv"):

"""Carga el CSV y procesa fechas, horas y extrae el primer nombre."""

df = pd.read\_csv(file\_path)

df["Date"] = pd.to\_datetime(df["Date"], errors="coerce")

df["Fecha"] = df["Date"].dt.date

df["Hora"] = df["Date"].dt.time

df = df.drop(columns=["Date"])

df["Primer\_Nombre"] = df["Name"].apply(lambda x: x.split()[0] if pd.notna(x) else None)

return df

def asignar\_sexo(nombre, detector):

"""Asigna un sexo basado en el primer nombre."""

if pd.notnull(nombre):

gender\_str = detector.get\_gender(nombre)

mapa\_sexo = {

"male": "Hombre",

"female": "Mujer",

"mostly\_male": "Hombre",

"mostly\_female": "Mujer",

"unknown": "Desconocido",

"andy": "Ambiguo"

}

return mapa\_sexo.get(gender\_str, "Desconocido")

return "Desconocido"

if \_\_name\_\_ == "\_\_main\_\_":

df = cargar\_y\_limpiar\_datos()

df["Sexo"] = df["Primer\_Nombre"].apply(lambda x: asignar\_sexo(x, detector))

# Análisis de likes promedio por sexo

likes\_sexo = df.groupby('Sexo')['Likes'].mean().dropna()

plt.figure()

ax\_sexo = likes\_sexo.plot(kind='bar', title="Likes promedio por sexo", color=["skyblue", "salmon", "lightcoral", "lightseagreen"])

plt.ylabel("Promedio de likes")

for i, v in enumerate(likes\_sexo):

ax\_sexo.text(i, v + 0.5, f'{v:.2f}', ha='center')

plt.tight\_layout()

plt.show()

print("\nLikes promedio por sexo:")

print(likes\_sexo)

# Análisis de likes promedio por región (Top 10)

likes\_region = df.groupby(df['Place'].fillna('Sin región'))['Likes'].mean().sort\_values(ascending=False).head(10).dropna()

plt.figure(figsize=(10, 6))

ax\_region = likes\_region.plot(kind='bar', title="Likes promedio por región (Top 10)", color="lightcoral")

plt.ylabel("Promedio de likes")

plt.xlabel("Región")

for i, v in enumerate(likes\_region):

ax\_region.text(i, v + 0.5, f'{v:.2f}', ha='center', rotation=45)

plt.tight\_layout()

plt.show()

print("\nLikes promedio por región (Top 10):")

print(likes\_region)