



Sociedad Ecuatoriana de Estadística.

Análisis de Comportamiento de Redes Sociales



PYTHON WEEKEND

Presentación: Angel Catagua Gonzalez



- Senior Business Intelligence Analyst - Vitapro.
- Profesor - Escuela de los Chefs.
- **Director del Núcleo Guayas de la Sociedad Ecuatoriana de Estadística.**



- * Data Science – SEE & Espol
- * Ingeniero en estadística Informática - ESPOL
- * Product Manager - Kurios



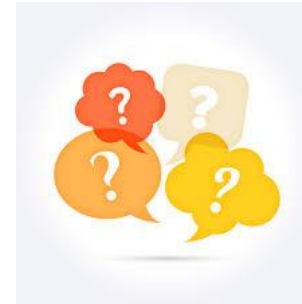
- . Cel: 0985045599
- . LinkedIn: Angel Catagua
- . TW: Anglucat



En este espacio



Tu participación
es importante



Siéntete libre de hacer
preguntas



Estamos para
ayudarte



Procuremos evitar
Distracciones



ANÁLISIS DE COMPORTAMIENTO EN REDES SOCIALES

CONFERENCISTA:

Angel Catagua



PYTHON WEEKEND



Rabo de res encendido.

INGREDIENTES:

- 2 rabos de res, limpios y cortados en pedazos de más o menos 10 centímetros
- ½ taza de aceite de oliva
- ½ kilogramo de cebolla
- ½ cabeza de ajo
- 3 pimientos grandes asados y pelados, o una lata de pimientos morrones
- ½ taza de perejil bien picadito
- 1 taza de salsa de tomate
- 1 ½ cucharada de sal
- ½ cucharadita de comino
- ¼ cucharadita de orégano
- 3 tazas de vino seco o tinto
- Ajíes picantes o salsa picante de ajíes a gusto



Menú

- Entrada: Creación de Token y entorno desarrolladores.
- Entremés: Limpieza y EDA
- Plata Fuerte: Wordcloud y Visualizaciones
- Postre: Análisis de Sentimiento

nuevo!
MENÚ
DEGUSTACIÓN



PYTHON WEEKEND

¿Que red social analizar?

Recorte de prensa:

- Las organizaciones que esperan hacer de la IA un diferenciador deben recurrir a conjuntos de datos alternativos, que pueden tener que crear ellos mismos
- Los datos significativos son mejores que los datos completos.



¿Que red social analizar?

<https://www.menti.com/hkqs3w4oha>



¿Por qué Twitter?

Dinámica propia de la red,

Me mantengo actualizado sobre asuntos que le interesan.

Facilita interacción con las empresas: las personas son más abiertas a aceptar un tweet, que tiene una lectura rápida y está mezclado entre otros tweets



¿Por qué Twitter?

1. Comparte el contenido que produces
2. Fácil adhesión
3. Un nuevo canal de atención
4. Construir relaciones
5. Branding
6. Monitoreo



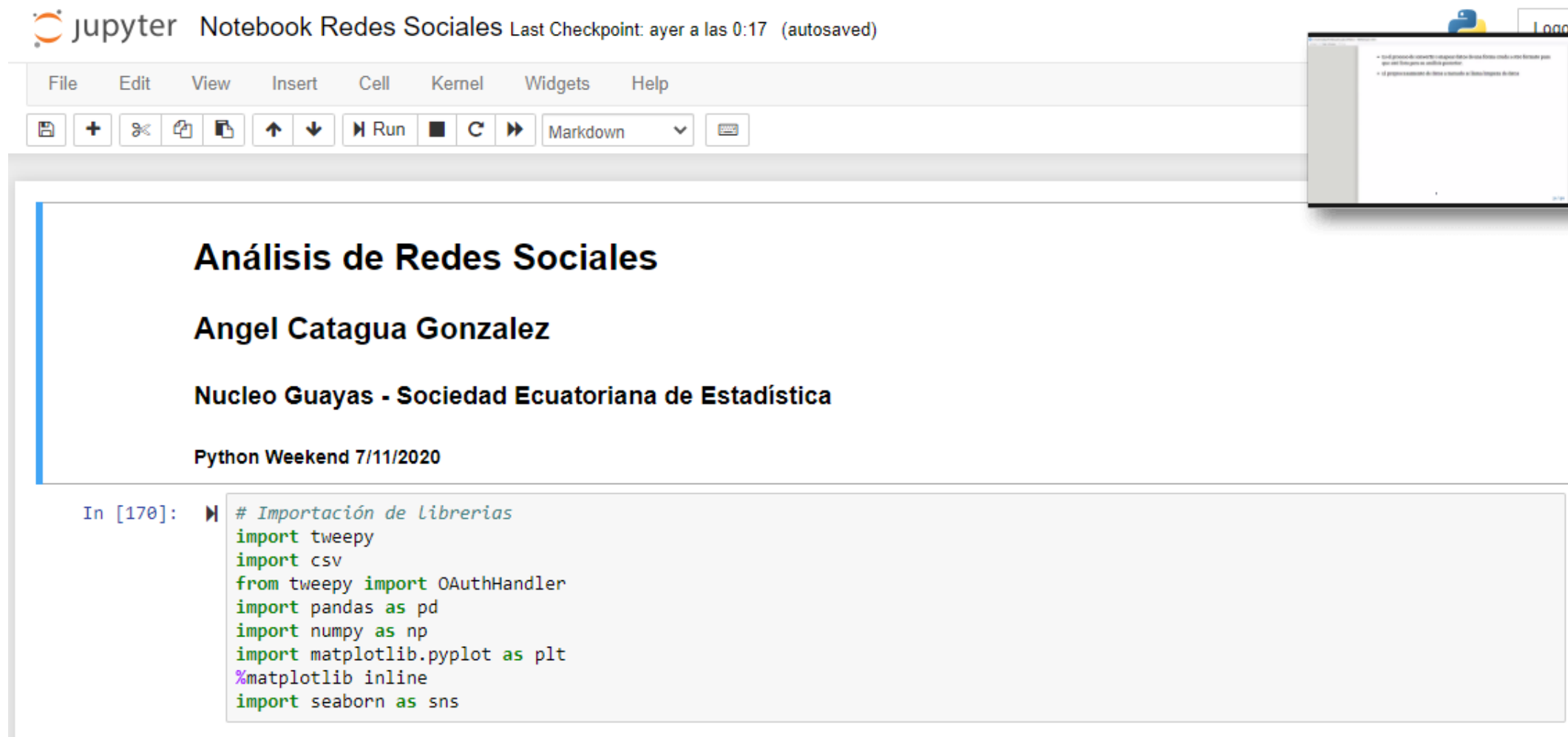
Preguntas



Jupyter Notebook



Jupyter Notebook



The screenshot shows a Jupyter Notebook interface. At the top, the title bar reads "jupyter Notebook Redes Sociales" followed by "Last Checkpoint: ayer a las 0:17 (autosaved)". Below this is a menu bar with "File", "Edit", "View", "Insert", "Cell", "Kernel", "Widgets", and "Help". A toolbar contains icons for saving, adding cells, undo, redo, and running code, along with a "Markdown" dropdown menu. The main content area has a blue vertical bar on the left and contains the following text:

Análisis de Redes Sociales

Angel Catagua Gonzalez

Nucleo Guayas - Sociedad Ecuatoriana de Estadística

Python Weekend 7/11/2020

Below the text is a code cell with the prompt "In [170]:". The code inside the cell is:

```
# Importación de librerías
import tweepy
import csv
from tweepy import OAuthHandler
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
%matplotlib inline
import seaborn as sns
```

In the top right corner, there is a small floating window with a Python logo and a "Logout" button. It contains two lines of text: "• El proceso de conversión de datos de redes sociales a un formato para su análisis estadístico se realiza mediante:" and "• El primer paso es la obtención de los datos de las redes sociales".

Jupyter Notebook

Librerías.

```
In [170]: ▶ # Importación de Librerías
import tweepy
import csv
from tweepy import OAuthHandler
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
%matplotlib inline
import seaborn as sns
```

Jupyter Notebook

Credenciales

```
In [ ]: ▶ #Credenciales y autenticaciones
consumer_key = 'XXXXXXXXXXXXXXXXXXXXXXXXXXXX'
consumer_secret = 'XXXXXXXXXXXXXXXXXXXXXXXXXXXX'
access_token = 'XXXXXXXXXXXXXXXXXXXXXXXXXXXX'
access_secret = 'XXXXXXXXXXXXXXXXXXXXXXXXXXXX|'
auth = OAuthHandler(consumer_key, consumer_secret)
auth.set_access_token(access_token, access_secret)
api = tweepy.API(auth, wait_on_rate_limit=True)
```


Jupyter Notebook

My Home timeline

```
In [3]: ▶ #Revisar los tuits que se escribieron en los todos los contactos
for status in tweepy.Cursor(api.home_timeline).items(10):
    print(status.text)
```

```
RT @elcomerciocom: ¡GRACIAS POR TANTO RICHIE! 🙌😭❤️ Richard Carapaz te
España 2020. La 'Loco...
Al CM de Liga le debe estar picando los deditos con lo de Richard Carapa:
RT @RichardPAConway: Dr Yokose presents data from the CARDIA study. Fast
erum urate. Extra...
```

Jupyter Notebook

My user timeline

In [5]: ▶

```
#Revisar los tuits que realizo el user  
for status in tweepy.Cursor(api.user_timeline).items(10):  
    print(status.text)
```

RT @Masappdata: ¿Ya estás inscrito? ¡MAÑANA! ➡ Taller gratuito Intr
lizar los datos...

RT @gbaqueri: Comunicado de la ESPOL en conjunto con la EPN.

RT @MashiRafael: Según Cedatos. 😊 <https://t.co/uEsM4jRnkN>

RT @NestorMontano: Desde @Masappdata hemos querido analizar lo que s
mada desde 22-j...

RT @Masappdata: #Guayaquil celebró su #Bicentenario y nos preguntamo

Jupyter Notebook

My home timeline

```
In [3]: ▶ #Revisar los tuits que se escribieron en los todos los contactos
for status in tweepy.Cursor(api.home_timeline).items(10):
    print(status.text)
```

```
RT @elcomerciocom: ¡GRACIAS POR TANTO RICHIE! 🙌😭❤️ Richard Carapaz te
España 2020. La 'Loco...
Al CM de Liga le debe estar picando los deditos con lo de Richard Carapa:
RT @RichardPAConway: Dr Yokose presents data from the CARDIA study. Fast
erum urate. Extra...
```

Jupyter Notebook







Guardar Información

```
In [172]: ► # http://docs.tweepy.org/en/v3.5.0/api.html
# https://towardsdatascience.com/how-to-scrape-more-information-from-tweets-on-twitter
# Que elementos se pueden capturar
with open ("Guayaquil.csv","w", newline = '\n', encoding = "utf-8") as csvwriter:
    writer = csv.writer(csvwriter,delimiter = "|")
    for status in tweepy.Cursor(api.search, q = "#Guayaquil").items(100):
        writer.writerow([status.created_at, status.user.screen_name,status.user.name,
```

Jupyter Notebook

Guardar Información

» » Escritorio » RES-AC » MCD » Charla Redes Sociales »

| <input type="checkbox"/> Nombre | Fecha de modificación | Tipo | |
|--|-----------------------|-----------------------|------|
|  .ipynb_checkpoints | 04/11/2020 22:35 | Carpeta de archivos | |
|  Analisis de Redes Sociales | 06/11/2020 0:17 | Presentación de ... | 1.05 |
|  cleaning_2.ipynb | 24/10/2020 15:04 | Archivo IPYNB | 209 |
|  concat_merge_2_visualizaciones.ipynb | 30/10/2020 21:55 | Archivo IPYNB | 177 |
|  Guayaquil | 07/11/2020 11:37 | Archivo de valores... | 21 |
|  Notebook Redes Sociales.ipynb | 07/11/2020 11:37 | Archivo IPYNB | 380 |

Jupyter Notebook

Guardar Información

```
In [172]: ▶ # http://docs.tweepy.org/en/v3.5.0/api.html
# https://towardsdatascience.com/how-to-scrape-more-information-from-tweets-on-twitter
# Que elementos se pueden capturar
with open ("Guayaquil.csv","w", newline = '\n', encoding = "utf-8") as csvwriter:
    writer = csv.writer(csvwriter,delimiter = "|")
    for status in tweepy.Cursor(api.search, q = "#Guayaquil").items(100):
        writer.writerow([status.created_at, status.user.screen_name,status.user.name,
```

Jupyter Notebook

Lecturas y Pandas

```
In [5]: ► tweets_df = pd.read_csv('Guayaquil.csv', names=['Fecha', 'Screen_name', 'user_name', 'user_location', 'coordinates', 'place', 't'
```


Jupyter Notebook

Encabezados

```
tweets_df.head()
```

4]:

| | Fecha | Screen_name | user_name | user_location | coordinates | place | retweet_count | favorite_count | text |
|---|------------------------|-----------------|---------------------------------|---------------------|-------------|-------|---------------|----------------|---|
| 0 | 2020-11-06 05:01:50 | Vidasanaecuador | ♥GYM EN LINEA Vida Sana Ecuador | Guayaquil - Ecuador | NaN | NaN | 0 | 0 | #GUAYAQUIL Mostly Sunny tomorrow! With a high ... |
| 1 | 2020-11-06 05:00:49 | spncns | Sandry Nogales | Machala, Ecuador | NaN | NaN | 80 | 0 | RT @lahistoriaec: Incidentes en el hospital de... |
| 2 | 2020-11-06 04:59:05 | patrici03034357 | patricia barrera | NaN | NaN | NaN | 33 | 0 | RT @AlertaMePerdi: Perrito #ENCONTRADO en el c... |
| 3 | 2020-11-06 04:58:26 | AlbertoDentoFa | Alberto | NaN | NaN | NaN | 80 | 0 | RT @lahistoriaec: Incidentes en el hospital de... |
| 4 | 2020-11-06 04:56:57 | Alex_F_Dom | Alex_Dominguez | Gye-Ecuador | NaN | NaN | 80 | 0 | RT @lahistoriaec: Incidentes en el hospital de... |

Jupyter Notebook

dtypes

```
tweets_df.head()
```

4]:

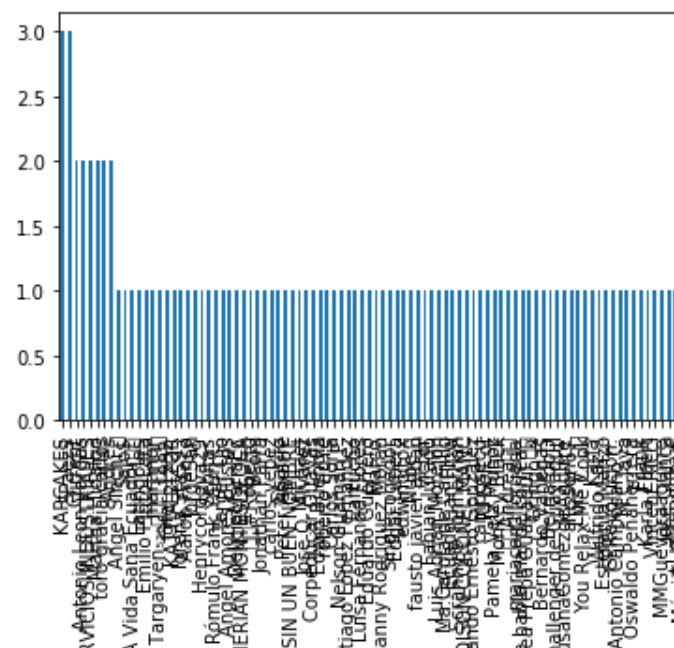
| | Fecha | Screen_name | user_name | user_location | coordinates | place | retweet_count | favorite_count | text |
|---|---------------------|-----------------|---------------------------------|---------------------|-------------|-------|---------------|----------------|---|
| 0 | 2020-11-06 05:01:50 | Vidasanaecuador | ♥GYM EN LINEA Vida Sana Ecuador | Guayaquil - Ecuador | NaN | NaN | 0 | 0 | #GUAYAQUIL Mostly Sunny tomorrow! With a high ... |
| 1 | 2020-11-06 05:00:49 | spncns | Sandry Nogales | Machala, Ecuador | NaN | NaN | 80 | 0 | RT @lahistoriaec: Incidentes en el hospital de... |
| 2 | 2020-11-06 04:59:05 | patrici03034357 | patricia barrera | NaN | NaN | NaN | 33 | 0 | RT @AlertaMePerdi: Perrito #ENCONTRADO en el c... |
| 3 | 2020-11-06 04:58:26 | AlbertoDentoFa | Alberto | NaN | NaN | NaN | 80 | 0 | RT @lahistoriaec: Incidentes en el hospital de... |
| 4 | 2020-11-06 04:56:57 | Alex_F_Dom | Alex_Dominguez | Gye-Ecuador | NaN | NaN | 80 | 0 | RT @lahistoriaec: Incidentes en el hospital de... |

Jupyter Notebook

Usuarios

```
In [7]: tweets_df["user_name"].value_counts().plot(kind='bar')
```

```
Out[7]: <matplotlib.axes._subplots.AxesSubplot at 0x1f61af43710>
```

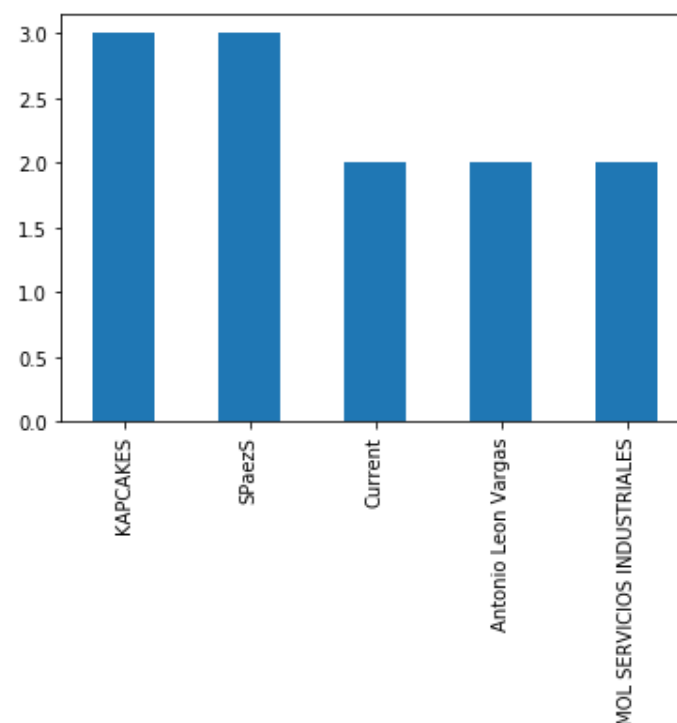


Jupyter Notebook

Usuarios

```
In [8]: #Seleccionar top de usuarios  
tweets_df["user_name"].value_counts().nlargest(5).plot(kind='bar')
```

```
Out[8]: <matplotlib.axes._subplots.AxesSubplot at 0x1f61b43f7f0>
```



Jupyter Notebook

Suma de RT

```
In [9]: #Data frame con mas RT de tuits  
tweets_df.groupby(["user_name"])["retweet_count"].sum().nlarge
```

Out[9]:

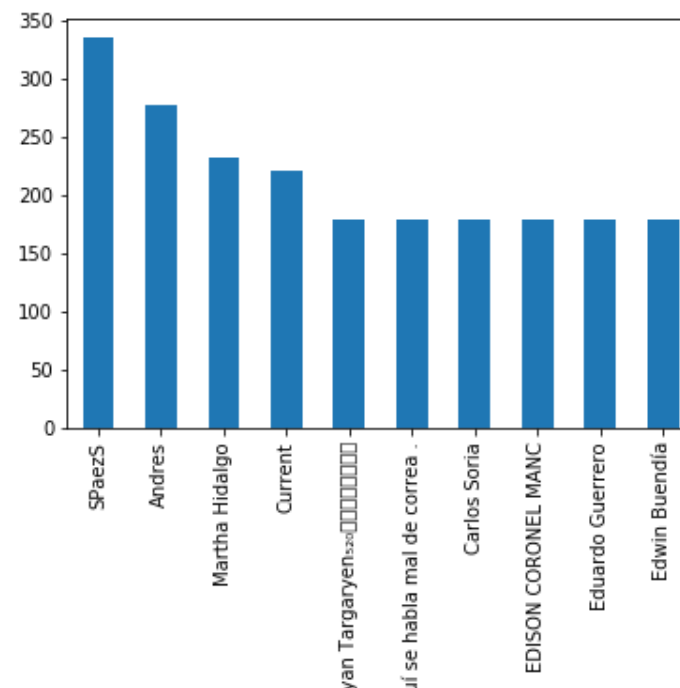
| | user_name | retweet_count |
|---|-------------------------------|---------------|
| 0 | SPaezS | 334 |
| 1 | Andres | 277 |
| 2 | Martha Hidalgo | 232 |
| 3 | Current | 221 |
| 4 | Aedryan Targaryen:z0💜💚💜💚 | 179 |
| 5 | Aquí se habla mal de correa . | 179 |
| 6 | Carlos Soria | 179 |
| 7 | EDISON CORONEL MANC | 179 |
| 8 | Eduardo Guerrero | 179 |
| 9 | Edwin Buendía | 179 |

Jupyter Notebook

Lo mismo pero mas bonito

```
In [10]: #Data frame con mas RT de tuits
tweets_df.groupby(["user_name"])["retweet_count"].sum().nlarg
```

```
Out[10]: <matplotlib.axes._subplots.AxesSubplot at 0x1f61b5534e0>
```



Jupyter Notebook

Un tuit

```
In [11]:  from wordcloud import WordCloud, STOPWORDS, ImageColorGenerator
```

```
In [12]:  #Extraigo un tuit  
text = tweets_df.text[0]
```

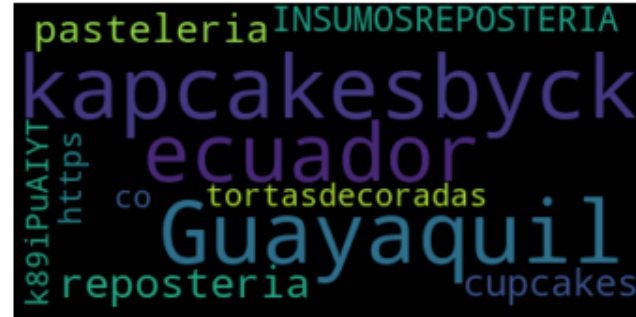
```
In [13]:  text
```

```
Out[13]:  '#kaptakesbyck #Guayaquil #ecuador #reposteria #pasteleria #cupcakes #INSUMOSREPOSTERIA #tor  
uAIYT'
```


Jupyter Notebook

Word Cloud

```
In [14]: #genero un word cloud de ese tuit
wordcloud = WordCloud().generate(text)
plt.imshow(wordcloud, interpolation='bilinear')
plt.axis("off")
plt.show()
```



```
In [184]: #Otro modelo de word cloud  
wordcloud = WordCloud(max_font_size=50, max_words=100, background_color="white").generate(text)  
plt.figure()  
plt.imshow(wordcloud, interpolation="bilinear")  
plt.axis("off")  
plt.show()
```

Jupyter Notebook

Guardar Word Cloud

```
In [185]: ► wordcloud.to_file("Word.png")
```

```
Out[185]: <wordcloud.wordcloud.WordCloud at 0x169c952ebe0>
```

Jupyter Notebook

Cuántas palabras hay en total

```
In [15]: ▶ #acumular el texto
text = " ".join(review for review in tweets_df.text)
print ("Hay {} palabras en la combinación de todos los tuits.".format(len(text)))
```

Hay 13397 palabras en la combinación de todos los tuits.

Jupyter Notebook

Stopword y Wordcloud

```
In [16]: stopwords = set(STOPWORDS)
stopwords.update(["https", "mis", "por", "el", "de", "mcdc_mcdc", "RT"])
```

```
In [17]: #https://www.datacamp.com/community/tutorials/wordcloud-python
wordcloud = WordCloud(stopwords=stopwords, background_color="white").generate(text)
plt.imshow(wordcloud, interpolation='bilinear')
plt.axis("off")
plt.show()
```



Jupyter Notebook

Mi nuevo Word cloud

```
#https://www.datacamp.com/community/tutorials/wordcloud-python
wordcloud = WordCloud(stopwords=stopwords, background_color="white").generate(text)
plt.imshow(wordcloud, interpolation='bilinear')
plt.axis("off")
plt.show()
```



Jupyter Notebook

Otras funciones de limpieza

```
In [22]: ▶ #defino funcion de limpieza
def clean_tweet(tweet):
    '''
    Utility function to clean tweet text by removing links, special characters
    using simple regex statements.
    '''
    return ' '.join(re.sub("(@[A-Za-z0-9]+)|([^0-9A-Za-z \t])|(\w+:\/\/\S+)", " ", tweet).split())
```

```
In [23]: ▶ def get_tweet_sentiment(tweet):
    analysis = TextBlob(tweet)
    language = analysis.detect_language()
    if language == 'en':
        analysis_ready = analysis
    else:
        analysis_ready = analysis.translate(to='en')

    if analysis_ready.sentiment.polarity > 0:
        return 'positive'
    elif analysis_ready.sentiment.polarity == 0:
        return 'neutral'
    else:
        return 'negative'
```

Jupyter Notebook

Sentimientos.

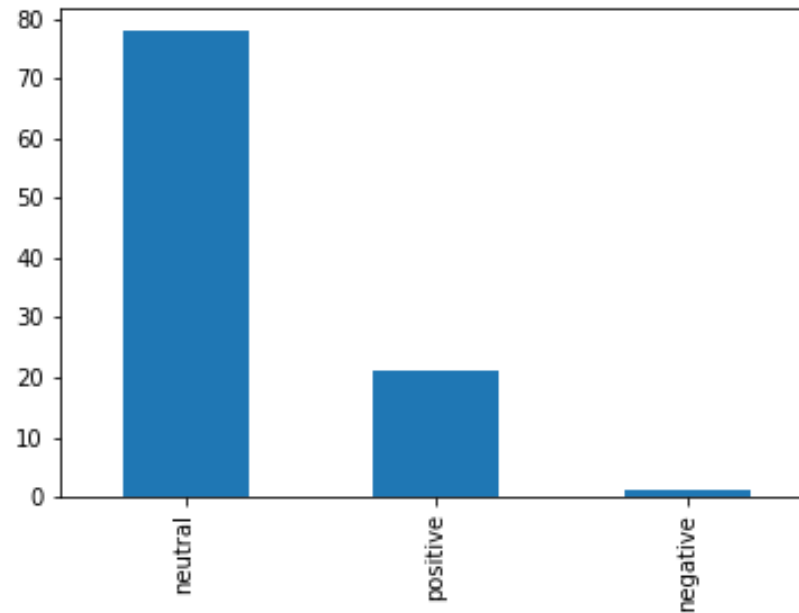
```
In [23]: ▶ def get_tweet_sentiment(tweet):  
            analysis = TextBlob(tweet)  
            language = analysis.detect_language()  
            if language == 'en':  
                analysis_ready = analysis  
            else:  
                analysis_ready = analysis.translate(to='en')  
  
            if analysis_ready.sentiment.polarity > 0:  
                return 'positive'  
            elif analysis_ready.sentiment.polarity == 0:  
                return 'neutral'  
            else:  
                return 'negative'
```


Jupyter Notebook

Sentimientos.

```
In [38]: ▶ #Data frame con mas RT de tuits  
tweets_df["Sentimientos"].value_counts().plot(kind='bar')
```

```
Out[38]: <matplotlib.axes._subplots.AxesSubplot at 0x1f61afc3e80>
```



Jupyter Notebook

Cambiar de lenguaje.

```
In [24]: ► text = tweets_df.text[0]
blob_obj = TextBlob(text)
blob_obj.detect_language()
blob_obj.translate(to="bn")
```

```
Out[24]: TextBlob("# ক্যাপকেকসবাইক # গুয়ায়াকিল # সেকোডর # রিপোসটারিয়া # প্যাসিটোরিয়া # কুপকেকস #INSUMOSREPOSTERIA #
ps://t.co/k89iPuaiYT")
```

```
In [29]: ► text = tweets_df.text[0]
blob_obj = TextBlob(text)
blob_obj.detect_language()
blob_obj.translate(to="en")
```

```
Out[29]: TextBlob("#kapcakesbyck #Guayaquil #ecuador #reposteria #pasteleria #cupcakes #INSUMOSREPOSTERIA # tortasd
t.co/k89iPuAIYT")
```

Jupyter Notebook

Subconjunto 2

```
In [41]: ▶ tweets_df_2=tweets_df[1:10]
```

```
In [40]: ▶ def get_tweet_sentiment(tweet):
    analysis = TextBlob(tweet)
    language = analysis.detect_language()
    if language == 'en':
        analysis_ready = analysis
    else:
        analysis_ready = analysis.translate(to='en')

    if analysis_ready.sentiment.polarity > 0:
        return 'positive'
    elif analysis_ready.sentiment.polarity == 0:
        return 'neutral'
    else:
        return 'negative'
```

```
In [43]: ▶ tweets_df_2["Sentimientos2"] = tweets_df_2["text"].apply(get_tweet_sentiment)
```

Jupyter Notebook

Subconjunto 2

```
In [41]: ▶ tweets_df_2=tweets_df[1:10]
```

```
In [40]: ▶ def get_tweet_sentiment(tweet):
    analysis = TextBlob(tweet)
    language = analysis.detect_language()
    if language == 'en':
        analysis_ready = analysis
    else:
        analysis_ready = analysis.translate(to='en')

    if analysis_ready.sentiment.polarity > 0:
        return 'positive'
    elif analysis_ready.sentiment.polarity == 0:
        return 'neutral'
    else:
        return 'negative'
```

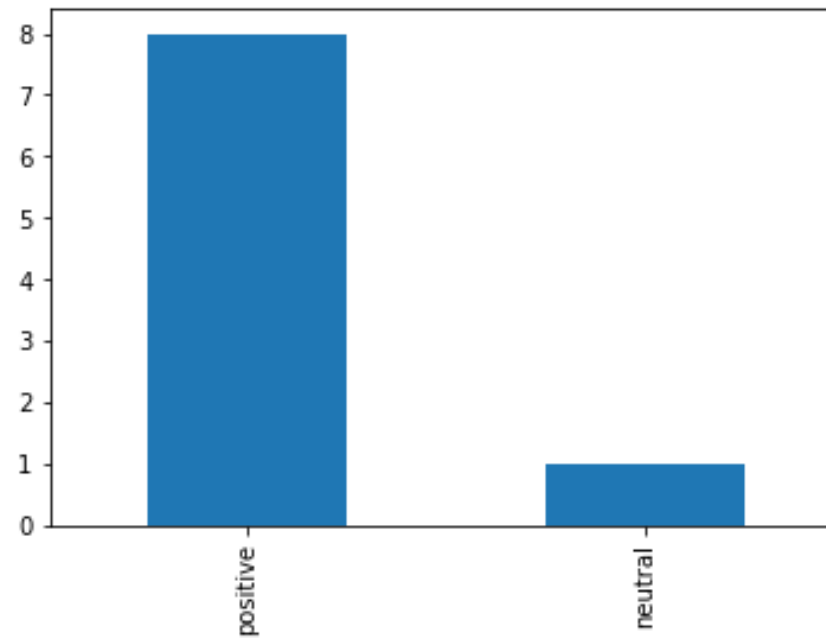
```
In [43]: ▶ tweets_df_2["Sentimientos2"] = tweets_df_2["text"].apply(get_tweet_sentiment)
```

Jupyter Notebook

Subconjunto 2

```
In [45]: ▶ #Data frame con mas RT de tuits  
tweets_df_2["Sentimientos2"].value_counts().plot(kind='bar')
```

```
Out[45]: <matplotlib.axes._subplots.AxesSubplot at 0x1f61df670b8>
```



Jupyter Notebook

Taller: 5 Minutos

Análisis de sentimientos de los 100 tuit sobre las elecciones de
EEUU

Jupyter Notebook

Taller: 5 Minutos

Análisis de sentimientos de los 100 tuit sobre las elecciones de
EEUU

Preguntas Finales



Muchas gracias



Análisis de Redes Sociales



- . Cel: 0985045599
- . LinkedIn: Angel Catagua
- . TW: Anglucatt