Librerias ¶

In [ ]:
```
# Librería para manejar las contracciones que se presentan en el inglés.

!pip install contractions
```

```
Looking in indexes: https://pypi.org/simple, (https://pypi.org/simple,) h
ttps://us-python.pkg.dev/colab-wheels/public/simple/ (https://us-python.p
kg.dev/colab-wheels/public/simple/)
Requirement already satisfied: contractions in /usr/local/lib/python3.7/d
ist-packages (0.1.72)
Requirement already satisfied: textsearch>=0.0.21 in /usr/local/lib/pytho
n3.7/dist-packages (from contractions) (0.0.24)
Requirement already satisfied: pyahocorasick in /usr/local/lib/python3.7/
dist-packages (from textsearch>=0.0.21->contractions) (1.4.4)
Requirement already satisfied: anyascii in /usr/local/lib/python3.7/dist-
packages (from textsearch>=0.0.21->contractions) (0.3.1)
```

In [ ]:
```
# Librería para manejar las flexiones gramaticales en el idioma inglés.

!pip install inflect
!pip install pandas-profiling==2.7.1
```

```
Looking in indexes: https://pypi.org/simple, (https://pypi.org/simple,)
https://us-python.pkg.dev/colab-wheels/public/simple/ (https://us-pytho
n.pkg.dev/colab-wheels/public/simple/)
Requirement already satisfied: inflect in /usr/local/lib/python3.7/dist
-packages (2.1.0)
Looking in indexes: https://pypi.org/simple, (https://pypi.org/simple,)
https://us-python.pkg.dev/colab-wheels/public/simple/ (https://us-pytho
n.pkg.dev/colab-wheels/public/simple/)
Requirement already satisfied: pandas-profiling==2.7.1 in /usr/local/li
b/python3.7/dist-packages (2.7.1)
Requirement already satisfied: jinja2>=2.11.1 in /usr/local/lib/python
3.7/dist-packages (from pandas-profiling==2.7.1) (2.11.3)
Requirement already satisfied: matplotlib>=3.2.0 in /usr/local/lib/pyth
on3.7/dist-packages (from pandas-profiling==2.7.1) (3.2.2)
Requirement already satisfied: visions[type_image_path]==0.4.1 in /usr/
local/lib/python3.7/dist-packages (from pandas-profiling==2.7.1) (0.4.
1)
Requirement already satisfied: phik>=0.9.10 in /usr/local/lib/python3.
7/dist-packages (from pandas-profiling==2.7.1) (0.12.2)
```

In [ ]:
```
# Librería Natural Language Toolkit, usada para trabajar con textos

import nltk
```

In [ ]:
```
# Punkt permite separar un texto en frases.

nltk.download('punkt')
```

```
[nltk_data] Downloading package punkt to /root/nltk_data...
[nltk_data]   Package punkt is already up-to-date!
```

Out[5]: True

In [ ]:
```
# Descarga todas las palabras vacias, es decir, aquellas que no aportan nad

nltk.download('stopwords')
```

```
[nltk_data] Downloading package stopwords to /root/nltk_data...
[nltk_data]   Package stopwords is already up-to-date!
```

Out[6]: True

In [ ]:
```
# Descarga de paquete WordNetLemmatizer, este es usado para encontrar el le

nltk.download('wordnet')
```

```
[nltk_data] Downloading package wordnet to /root/nltk_data...
[nltk_data]   Package wordnet is already up-to-date!
```

Out[7]: True

In [ ]:
```
!pip install escape
```

```
Looking in indexes: https://pypi.org/simple, (https://pypi.org/simple,) h
ttps://us-python.pkg.dev/colab-wheels/public/simple/ (https://us-python.p
kg.dev/colab-wheels/public/simple/)
Requirement already satisfied: escape in /usr/local/lib/python3.7/dist-pa
ckages (1.1)
```

In [ ]:
```
# Instalación de librerias

import pandas as pd
import numpy as np
import sys
from pandas_profiling import ProfileReport

import re, string, unicodedata
import contractions
import inflect
from nltk import word_tokenize, sent_tokenize
from nltk.corpus import stopwords
from nltk.stem import LancasterStemmer, WordNetLemmatizer
from nltk.stem import PorterStemmer

from sklearn.model_selection import train_test_split,GridSearchCV
from sklearn.feature_extraction.text import TfidfVectorizer, CountVectorize
from sklearn.pipeline import Pipeline, FeatureUnion
from sklearn.svm import SVC
from sklearn.ensemble import BaggingClassifier, RandomForestClassifier, Ada
from sklearn.naive_bayes import BernoulliNB
from sklearn.metrics import classification_report, confusion_matrix, plot_p
from sklearn.base import BaseEstimator, ClassifierMixin

import matplotlib.pyplot as plt

from tqdm import tqdm

import nltk

from ast import literal_eval
```

Cargar Datos

In [ ]:
```
# Uso de la libreria pandas para la lectura de archivos

data = pd.read_csv('DatosSuicidio/SuicidiosProyecto.csv', sep=',', encoding
data = data.rename(columns={'Unnamed: 0': 'id'})

data.head()
```

Out[11]:

	id	text	class
0	173271	i want to destroy myselffor once everything wa...	suicide
1	336321	I kinda got behind schedule with learning for ...	non-suicide
2	256637	I'm just not sure anymoreFirst and foremost: I...	suicide
3	303772	please give me a reason to liveThats too much ...	suicide
4	293747	27f struggling to find meaning moving forwardI...	suicide

In [ ]:
```
# Asignación a una nueva variable de los datos leidos

data_read = data
```

Entendimiento y perfilamiento de los datos

Entendimiento de datos

In [ ]:
```
from statistics import mode

textos = data_read.copy()
textos['Conteo'] = [len(x) for x in textos['text']]
textos['Max'] = [[max([len(x) for x in i.split(' ')])][0] for i in textos['
textos['Min'] = [[min([len(x) for x in i.split(' ')])][0] for i in textos['

# Se realiza un perfilamiento de los datos con la librería pandas profiling
ProfileReport(textos)
```

```
Summarize dataset:   0%|          | 0/18 [00:00<?, ?it/s]
```

Preparación de los datos

Limpieza de datos

In [ ]:
```
def remove_non_ascii(words):
    """Remove non-ASCII characters from list of tokenized words"""
    new_words = []
    for word in words:
        new_word = unicodedata.normalize('NFKD', word).encode('ascii', 'ign
        new_words.append(new_word)
    return new_words

def to_lowercase(words):
    """Convert all characters to lowercase from list of tokenized words"""
    new_words = words.lower()
    return new_words

def remove_punctuation(words):
    """Remove punctuation from list of tokenized words"""
    new_words = []
    for word in words:
        new_word = re.sub(r'[^\w\s]', '', word)
        if new_word != '':
            new_words.append(new_word)
    return new_words

def replace_numbers(words):
    """Replace all interger occurrences in list of tokenized words with tex
    p = inflect.engine()
    new_words = []
    for word in words:
        if word.isdigit():
            new_word = p.number_to_words(word)
            new_words.append(new_word)
        else:
            new_words.append(word)
    return new_words

def remove_stopwords(words):
    """Remove stop words from list of tokenized words"""
    stop_words = set(stopwords.words('english'))
    filtered_sentence = [w for w in words if not w.lower() in stop_words]
    filtered_sentence = []
    for w in words:
        if w not in stop_words:
            filtered_sentence.append(w)
    return filtered_sentence

def preprocessing(words):
    words = to_lowercase(words)
    words = replace_numbers(words)
    words = remove_punctuation(words)
    words = remove_non_ascii(words)
    words = remove_stopwords(words)
    return words
```

Tokenización

In [ ]:
```
for t in tqdm(range(len(data_read))):
    try:
        contractions.fix(data_read['text'][t])
    except:
        print(data_read['text'][t])
```

```
 75%|███████▌   | 146270/195700 [00:12<00:03, 13063.23it/s]

İ am losing my mind...İ dont know how i can endure this bullshit ...
İam 21 and suffered almost every stage of my life , things are not going
on my way , worst thing is everyone hates me even my family too . They th
ink iam a failure.
İam an university student but my grades like an  rotten apple on the tre
e... i have no motivation or energy. And dont have a girlfriend still vir
gin . Why i should keep up for nothing ?,  for more suffer ? or more fail
ure ?
İ just want peace , love and some money...
İ know there is still some hope but i tired keep fighting it is pointless
, i hate it i just want some victory . İ am looking for a gun but it is h
ard to access on my country . İ just dont want hurt anymore... it is enou
gh.  İf people interested in motivational videos please watch
(Why we choose suicide Mark Henic) it relaxed me one bit . İ need your pr
ays too

100%|██████████| 195700/195700 [00:16<00:00, 11760.61it/s]
```

In [ ]:
```
# Aplicar la eliminación del ruido

data_read['text'] = data_read['text'].apply(word_tokenize)
data_read.head()
```

Out[17]:

	id	text	class
0	173271	[i, want, to, destroy, myselffor, once, everyt...	suicide
1	336321	[I, kinda, got, behind, schedule, with, learni...	non-suicide
2	256637	[I, 'm, just, not, sure, anymoreFirst, and, fo...	suicide
3	303772	[please, give, me, a, reason, to, liveThats, t...	suicide
4	293747	[27f, struggling, to, find, meaning, moving, f...	suicide

In [ ]:
```
# Guardar datos tokenizados en un archivo

data_read.to_csv('DatosSuicidio/DatosTokenizados.csv')
```

In [ ]:
```
# Leer archivo de datos tokenizados

data_tokenized = pd.read_csv('DatosSuicidio/DatosTokenizados.csv', sep=',',

data_tokenized.head()
```

Out[19]:

	id	text	class
0	173271	['i', 'want', 'to', 'destroy', 'myselffor', 'o...	suicide
1	336321	['I', 'kinda', 'got', 'behind', 'schedule', 'w...	non-suicide
2	256637	['I', "'m", 'just', 'not', 'sure', 'anymoreFir...	suicide
3	303772	['please', 'give', 'me', 'a', 'reason', 'to', ...	suicide
4	293747	['27f', 'struggling', 'to', 'find', 'meaning',...	suicide

In [ ]:
```
for t in tqdm(range(len(data_tokenized))):
    try:
        data_tokenized['text'][t]=remove_stopwords(remove_non_ascii(remove_
    except:
        print(t)
```

```
  0%|          | 0/195700 [00:00<?, ?it/s]/usr/local/lib/python3.7/dist-p
ackages/ipykernel_launcher.py:3: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: https://pandas.pydata.org/pandas-do
cs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy (http
s://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returni
ng-a-view-versus-a-copy)
  This is separate from the ipykernel package so we can avoid doing impor
ts until
100%|██████████| 195700/195700 [22:57<00:00, 142.06it/s]
```

In [ ]:
```
# Guardar datos tokenizados sin stop words en un archivo

data_tokenized.to_csv('DatosSuicidio/DatosTokenizadosSinStopWords.csv')
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call las
t)
<ipython-input-10-4cd0317c05fe> in <module>
      1 # Guardar datos tokenizados sin stop words en un archivo
      2
----> 3 data_tokenized.to_csv('DatosSuicidio/DatosTokenizadosSinStopWord
s.csv')

NameError: name 'data_tokenized' is not defined
```

In [ ]:
```
# Leer archivo de datos tokenizados sin stop words

data_no_stop_words = pd.read_csv('DatosSuicidio/DatosTokenizadosSinStopWord

data_no_stop_words['text'] = data_no_stop_words['text'].apply(literal_eval)
data_no_stop_words.head()
```

Out[9]:

	id	text	class
0	173271	[want, destroy, myselffor, everything, startin...	suicide
1	336321	[kind, got, behind, schedule, learning, next, ...	non-suicide
2	256637	[sure, anymorefirst, foremost, brazil, judge, ...	suicide
3	303772	[please, give, reason, livethats, much, reason...	suicide
4	293747	[27f, struggling, find, meaning, moving, forwa...	suicide

Normalización

In [ ]:
```
nltk.download('omw-1.4')

porter = PorterStemmer()
wordnet_lemmatizer = WordNetLemmatizer()

def stem_words(words):
    """Stem words in list of tokenized words"""
    stem_sentence=[]
    for word in words:
        stem_sentence.append(porter.stem(word))
    return stem_sentence

def lemmatize_verbs(words):
    """Lemmatize verbs in list of tokenized words"""
    stem_sentence=[]
    for word in words:
        stem_sentence.append(wordnet_lemmatizer.lemmatize(word))
    return stem_sentence

def stem_and_lemmatize(words):
    stems = stem_words(words)
    lemmas = lemmatize_verbs(words)
    return stems + lemmas

data_no_stop_words['text'] = data_no_stop_words['text'].apply(stem_and_lemm
data_no_stop_words.head()
```

```
[nltk_data] Downloading package omw-1.4 to /root/nltk_data...
```

Out[10]:

	id	text	class
0	173271	[want, destroy, myselffor, everyth, start, fee...	suicide
1	336321	[kind, got, behind, schedul, learn, next, week...	non-suicide
2	256637	[sure, anymorefirst, foremost, brazil, judg, s...	suicide
3	303772	[pleas, give, reason, livethat, much, reason, ...	suicide
4	293747	[27f, struggl, find, mean, move, forwardi, adm...	suicide

In [ ]:
```
# Guardar datos normalizados en un archivo

data_no_stop_words.to_csv('DatosSuicidio/DatosNormalizados.csv')
```

In [ ]:
```
# Leer archivo de datos normalizados

data_normalized = pd.read_csv('DatosSuicidio/DatosNormalizados.csv', sep=',

data_normalized['text'] = data_normalized['text'].apply(literal_eval)
data_normalized.head()
```

Out[9]:

	id	text	class
0	173271	[want, destroy, myselffor, everyth, start, fee...	suicide
1	336321	[kind, got, behind, schedul, learn, next, week...	non-suicide
2	256637	[sure, anymorefirst, foremost, brazil, judg, s...	suicide
3	303772	[pleas, give, reason, livethat, much, reason, ...	suicide
4	293747	[27f, struggl, find, mean, move, forwardi, adm...	suicide

Selección de campos

In [ ]:
```
data_normalized['words'] = data_normalized['text']
data_normalized['text'] = data_normalized['text'].apply(lambda x: ' '.join(
data_normalized
```

Out[10]:

	id	text	class	words
0	173271	want destroy myselffor everyth start feel okay...	suicide	[want, destroy, myselffor, everyth, start, fee...
1	336321	kind got behind schedul learn next week testwe...	non-suicide	[kind, got, behind, schedul, learn, next, week...
2	256637	sure anymorefirst foremost brazil judg second ...	suicide	[sure, anymorefirst, foremost, brazil, judg, s...
3	303772	pleas give reason livethat much reason live li...	suicide	[pleas, give, reason, livethat, much, reason, ...
4	293747	27f struggl find mean move forwardi admit bit ...	suicide	[27f, struggl, find, mean, move, forwardi, adm...
...	...	...	...	...
195695	248038	drop cool new cereal idea like would ideal cer...	non-suicide	[drop, cool, new, cereal, idea, like, would, i...
195696	216516	unpopular opinion cat deserv love respect much...	non-suicide	[unpopular, opinion, cat, deserv, love, respec...
195697	199341	hey guy doin hey guy doin	non-suicide	[hey, guy, doin, hey, guy, doin]
195698	145373	uhm cover dog blanket light wake woke ran wall...	non-suicide	[uhm, cover, dog, blanket, light, wake, woke, ...
195699	305170	____god end life tire could want anyth need so...	suicide	[____god, end, life, tire, could, want, anyth,...

195700 rows × 4 columns

In [ ]:
```
# Guardar datos con texto normalizado en un archivo

data_normalized.to_csv('DatosSuicidio/DatosTextoNormalizado.csv')
```

```
---------------------------------------------------------------------------
--
NameError                                 Traceback (most recent call las
t)
<ipython-input-1-477805e8413b> in <module>
      1 # Guardar datos con texto normalizado en un archivo
      2
----> 3 data_normalized.to_csv('DatosSuicidio/DatosTextoNormalizado.csv')

NameError: name 'data_normalized' is not defined
```

In [ ]:
```
# Leer archivo de datos con texto normalizado

data_normalized_text = pd.read_csv('DatosSuicidio/DatosTextoNormalizado.csv

data_normalized_text['words'] = data_normalized_text['words'].apply(literal
data_normalized_text.head()
```

Out[9]:

	id	text	class	words
0	173271	want destroy myselffor everyth start feel okay...	suicide	[want, destroy, myselffor, everyth, start, fee...
1	336321	kind got behind schedul learn next week testwe...	non-suicide	[kind, got, behind, schedul, learn, next, week...
2	256637	sure anymorefirst foremost brazil judg second ...	suicide	[sure, anymorefirst, foremost, brazil, judg, s...
3	303772	pleas give reason livethat much reason live li...	suicide	[pleas, give, reason, livethat, much, reason, ...
4	293747	27f struggl find mean move forwardi admit bit ...	suicide	[27f, struggl, find, mean, move, forwardi, adm...

In [ ]:
```
data_normalized_text = data_normalized_text.dropna()
```

In [ ]:
```
X_data, y_data = data_normalized_text['text'],data_normalized_text['class']
y_data = (y_data == 'suicide').astype(int)
y_data
```

Out[11]:
```
0         1
1         0
2         1
3         1
4         1
         ..
195695    0
195696    0
195697    0
195698    0
195699    1
Name: class, Length: 195668, dtype: int64
```

In [ ]:
```
from sklearn.feature_extraction.text import TfidfVectorizer
tfidf = TfidfVectorizer(min_df = 200, max_df =0.5)
X_tf_idf = tfidf.fit_transform(X_data)
print(X_tf_idf.shape)
```

```
(195668, 4987)
```

In [ ]:
```
print(X_tf_idf.shape)
```

```
(195668, 4987)
```

In [ ]:
```
np.amax(X_tf_idf)
```

Out[14]:
```
1.0
```

In [ ]:
```
print(len(tfidf.vocabulary_))
```

```
4987
```

In [ ]:
```
procesed_data = pd.DataFrame(
    X_tf_idf.todense(),
)
```

In [ ]:
```
procesed_data.head()
```

Out[17]:

	0	1	2	3	4	5	6	7	8	9	...	4977	4978	4979	4980	4981	4982	498
0	0.000000	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0
1	0.000000	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0
2	0.122788	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0
3	0.000000	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0
4	0.000000	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0

5 rows × 4987 columns

In [ ]:
```
procesed_data['suicide'] = y_data
procesed_data
```

Out[21]:

	0	1	2	3	4	5	6	7	8	9	...	4978	4979	4980	4981	4982	498
0	0.000000	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.(
1	0.000000	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.(
2	0.122788	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.(
3	0.000000	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.(
4	0.000000	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.(
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	..
195663	0.000000	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.(
195664	0.000000	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.(
195665	0.000000	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.(
195666	0.000000	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.(
195667	0.000000	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.(

195668 rows × 4988 columns

In [ ]:
```
tfidf.vocabulary_
```

Out[19]:
```
{'want': 4814,
 'destroy': 1257,
 'everyth': 1590,
 'start': 4198,
 'feel': 1736,
 'okay': 3069,
 'came': 701,
 'know': 2515,
 'use': 4723,
 'cope': 1009,
 'reason': 3583,
 'tear': 4406,
 'skin': 4029,
 'shred': 3981,
 'swallow': 4365,
 'everi': 1583,
 'pill': 3262,
 'find': 1768,
 'right': 3742,
```

In [ ]:
```
# Guardar datos procesados en un archivo

procesed_data.to_csv('DatosSuicidio/DatosProcesados.csv')
```