Universidad
Carlos III de Madrid

Universidad Carlos III de Madrid

Métodos Bayesianos, Grado en Estadística y Empresa

Análisis y clasificación de textos

Fabio Scielzo Ortiz

Índice

1 Introducción

En este trabajo se va a realizar un análisis y clasificación de textos. Para ellos se utilizaran dos lenguajes de programación, `Python` y `R`. El trabajo puede dividirse en dos partes bien diferenciadas, una primera parte en la que se trabaja con `Python` y una segunda en la que se usa `R`.

En la primera parte, en la que trabajamos con `Python`, se llevará acabo una descripción y preprocesado del data-set con el que trabajaremos, posteriormente se llevara acabo un análisis de texto, y para finalizar se realizaran tareas de clasificación aplicando algoritmos de clasificación supervisada, especialmente el algoritmo de clasificación ingenua bayesiana.

En la parte en la que trabajamos con `R` se seguirán los pasos del ejemplo ilustrado en clase.

2 Descripción de los datos (`Python`)

El data-set con el que vamos a trabajar contiene como observaciones noticias, y como variables la fecha, el título y el texto de la noticia, y si es una noticia falsa (fake new) o es verdadera (no fake new).

Importamos la libreria `pandas`, que es la liberia de `Python` mas usada para la manipulación y manejo de datos en formato de tabla, es decir, data-frames.

```
import pandas as pd
```

Ahora importamos los datos, que originalmente estan distribuidos en dos data-sets, uno que contiene las fake news (`df_Fake`) y otro que contiene las no fake news (`df_True`):

```
df_Fake = pd.read_csv('Fake.csv')
df_True = pd.read_csv('True.csv')
```

Creamos una variable que indicará en nuestro data-set final si la noticia es fake o no fake:

```
df_Fake['Fake'] = 1
df_True['Fake'] = 0
```

Si para una noticia la nueva variable creada `Fake` toma el valor 1 , indica que es fake new, y si toma el 0 indica que no es fake new.

Ahora concatenamos (por filas) los dos data-sets anteriores, para generar el data-set con el que trabajaremos:

```
Fake_News_Data = pd.concat([df_Fake, df_True])
```

Seleccionamos las columnas (variables) de nuestro interés:

```
Fake_News_Data = Fake_News_Data.loc[: , ['Fake', 'title', 'text', 'date']
↪  ]
```

Añadimos un índice al data-set:

```
Fake_News_Data.index = range(0 , len(Fake_News_Data))
```

Ahora vamos a ver de qué tipo son nuestras variables en `Python` :

```
Fake_News_Data.dtypes
```

```
Fake      int64
title    object
text     object
date     object
dtype: object
```

El tipo `object` es propio de variables no cuantitativos, como categoricas o texto, y el tipo `int64` es propio de variables enteras.

En este caso dejaremos los types como están, salvo el de la variable `Fake` que es categorica y por tanto es más adecuado que su type sea `object`

```
Fake_News_Data['Fake'] = Fake_News_Data['Fake'].astype('object')
```

Hacemos una breve descripción estadistica de las variables del data-set:

```
Fake_News_Data.describe(include='all')
```

Fake

title

text

date

count

44898

44898

44898

44898

unique

2

38729

38646

2397

top

1

Factbox: Trump fills top jobs for his administ...

December 20, 2017

freq

23481

14

627

182

Calculamos el numero de valores faltantes (NA) en cada una de las variables:

```
Fake_News_Data.isnull().sum()
```

```
Fake     0
title    0
text     0
date     0
dtype: int64
```

```
import numpy as np

import seaborn as sns
import matplotlib as mpl
import matplotlib.pyplot as plt

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

```
prop_Fake_yes = len( Fake_News_Data.loc[ Fake_News_Data['Fake']== 1 , :] )
↪  / len(Fake_News_Data)

prop_Fake_no = len( Fake_News_Data.loc[ Fake_News_Data['Fake']== 0 , :] )
↪  / len(Fake_News_Data)
```

```
Fake_News_Data['proportion_Fakes'] = 0


for i in range(0, len(Fake_News_Data)):

    if Fake_News_Data['Fake'][i] == 1 :

        Fake_News_Data['proportion_Fakes'][i] = prop_Fake_yes

    else :

        Fake_News_Data['proportion_Fakes'][i] = prop_Fake_no
```

```
C:\Users\Usuario\AppData\Local\Temp\ipykernel_19644\2699169446.py:8: SettingWithCopyWarni
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-docs/stable/user_g
  Fake_News_Data['proportion_Fakes'][i] = prop_Fake_yes
```

```
p1 = sns.barplot(x='Fake', y='proportion_Fakes', data=Fake_News_Data,
↪  palette="Spectral")
p1.set_yticks( np.arange(0, 0.85, 0.1)  )
p1.set_xticklabels(['No', 'Yes'])
p1.axes.set(xlabel='Fakes', ylabel='proportion')
```

```
[Text(0.5, 0, 'Fakes'), Text(0, 0.5, 'proportion')]
```

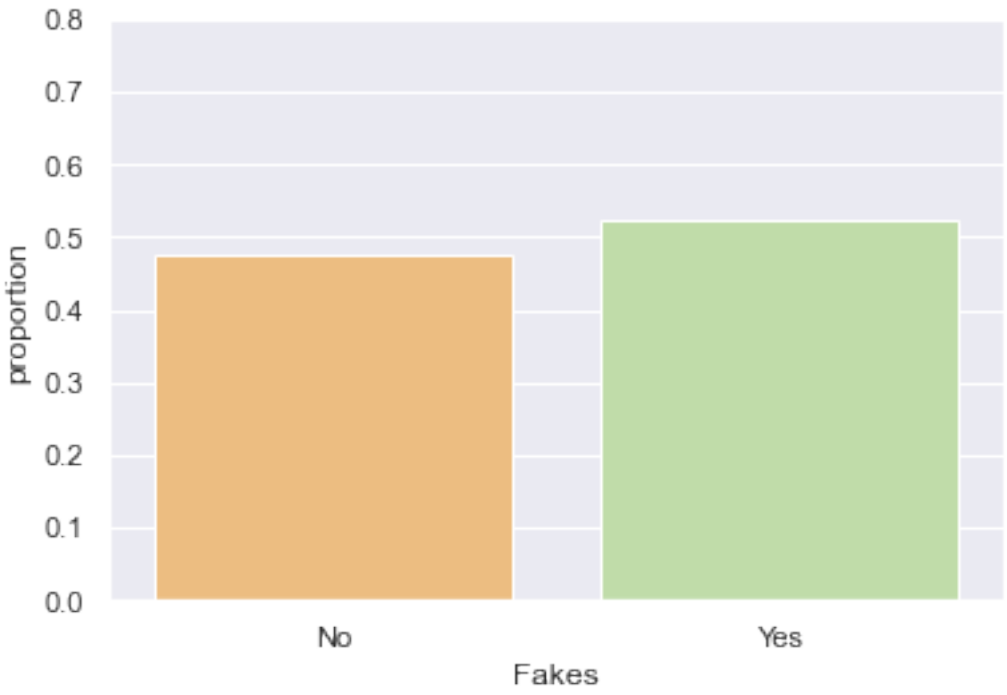

Figure 1: png

```
[prop_Fake_no , prop_Fake_yes]
```

```
[0.47701456635039424, 0.5229854336496058]
```

```
[prop_Fake_no*len(Fake_News_Data) , prop_Fake_yes*len(Fake_News_Data)]
```

```
[21417.0, 23481.0]
```

```
Fake_News_Data = Fake_News_Data.loc[ : , Fake_News_Data.columns !=
↪  'proportion_Fakes']
```

```
Fake_News_Data['word_count'] =
↪  Fake_News_Data['text'].str.split().str.len()
```

```
Fake_News_Data
```

Fake

title

text

date

word_count

0

1

Donald Trump Sends Out Embarrassing New Year'...

Donald Trump just couldn t wish all Americans ...

December 31, 2017

495

1

1

Drunk Bragging Trump Staffer Started Russian ...

House Intelligence Committee Chairman Devin Nu...

December 31, 2017

305

2

1

Sheriff David Clarke Becomes An Internet Joke...

On Friday, it was revealed that former Milwauk...

December 30, 2017

580

3

1

Trump Is So Obsessed He Even Has Obama's Name...

On Christmas day, Donald Trump announced that ...

December 29, 2017

444

4

1

Pope Francis Just Called Out Donald Trump Dur...

Pope Francis used his annual Christmas Day mes...

December 25, 2017

420

...

...

...

...

...

...

44893

0

'Fully committed' NATO backs new U.S. approach...

BRUSSELS (Reuters) - NATO allies on Tuesday we...

August 22, 2017

466

44894

0

LexisNexis withdrew two products from Chinese ...

LONDON (Reuters) - LexisNexis, a provider of l...

August 22, 2017

125

44895

0

Minsk cultural hub becomes haven from authorities

MINSK (Reuters) - In the shadow of disused Sov...

August 22, 2017

320

44896

0

Vatican upbeat on possibility of Pope Francis ...

MOSCOW (Reuters) - Vatican Secretary of State ...

August 22, 2017

205

44897

0

Indonesia to buy $1.14 billion worth of Russia...

JAKARTA (Reuters) - Indonesia will buy 11 Sukh...

August 22, 2017

210

44898 rows × 5 columns

```
Fake_News_Data.groupby('Fake')['word_count'].mean()
```

```
Fake
0    385.640099
1    423.197905
Name: word_count, dtype: float64
```

3 Preprocesado de los datos

3.1 Tokenizacion

```
def limpiar_tokenizar(texto):

    import re

    '''
    Esta función limpia y tokeniza el texto en palabras individuales.
    El orden en el que se va limpiando el texto no es arbitrario.
    El listado de signos de puntuación se ha obtenido de:
↪   print(string.punctuation)
    y re.escape(string.punctuation)
    '''

    # Se convierte todo el texto a minúsculas

    nuevo_texto = texto.lower()

    # Eliminacion de paginas web (palabras que empiezan por "http")

    nuevo_texto = re.sub('http\S+', ' ', nuevo_texto)

    # Eliminacion de signos de puntuación

    regex =
↪   '[\\!\\"\\#\\$\\%\\&\\'\\(\\)\\*\\+\\,\\-\\.\\/\\:\\;\\<\\=\\>\\?\\@\\[\\\\\\]\\^_\\

    nuevo_texto = re.sub(regex , ' ', nuevo_texto)

    # Eliminacion de numeros

    nuevo_texto = re.sub("\d+", ' ', nuevo_texto)

    # Eliminacion de espacios en blanco multiples

    nuevo_texto = re.sub("\\s+", ' ', nuevo_texto)

    # Tokenizacion por palabras individuales

    nuevo_texto = nuevo_texto.split(sep = ' ')

    # Eliminacion de tokens con una longitud <= 1

    nuevo_texto = [token for token in nuevo_texto if len(token) >=  2]

    return(nuevo_texto)
```

```
test = "Esto es 1 ejemplo de l'limpieza de6 TEXTO  https://t.co/rnHPgyhx4Z
↪   @cienciadedatos #textmining"
```

```
print(limpiar_tokenizar(texto=test))
```

```
['esto', 'es', 'ejemplo', 'de', 'limpieza', 'de', 'texto', 'cienciadedatos', 'textmining'
```

```
Fake_News_Data['text'][0]
```

```
'Donald Trump just couldn t wish all Americans a Happy New Year and leave it at that. Ins
```

```
print(limpiar_tokenizar(texto=Fake_News_Data['text'][0]))
```

```
['donald', 'trump', 'just', 'couldn', 'wish', 'all', 'americans', 'happy', 'new', 'year',
```