

Practica 1 Seguridad Informática

March 11, 2021

1 Fuente de información asociada a un texto

```
[26]: import math
import operator
```

```
[27]: # Se obtienen los datos del fichero y son formateados
file = open('data.txt', 'r', encoding='utf8')
text = file.read()
file.close()

print("El texto introducido es:")
print(text)

# Se cambia cada salto de linea por dos espacios como manda el gui3n
text = text.replace("\n", " ")
```

El texto introducido es:

La noche cae, brumosa ya y morada.

Vagas claridades malvas y verdes perduran tras la torre de la iglesia.

El camino sube, lleno de sombras, de campanillas, de fragancia de hierba,
de canciones, de cansancio y de anhelo.

```
[28]: # Se obtiene la frecuencia y probabilidad de cada caracter
frequencies = dict()
for symbol in text:
    if symbol not in frequencies:
        frequencies[symbol] = text.count(symbol)

absFreq = sum(frequencies.values())

odds = dict()
for key in frequencies:
    odds[key] = frequencies[key]/absFreq
```

1.1 Frecuencia absoluta y la probabilidad para el símbolo “d”.

```
[29]: print("Frecuencia para el simbolo d:", frequencies["d"])
      print("Probabilidad para el simbolo d:" , odds["d"])
      print("Probabilidad para el simbolo d redondeada al tercer decimal:" ,
            ↪round(odds["d"], 3))
```

Frecuencia para el simbolo d: 13

Probabilidad para el simbolo d: 0.05855855855855856

Probabilidad para el simbolo d redondeada al tercer decimal: 0.059

1.2 Entropia en bits de la fuente de información.

```
[30]: entropy:float = 0
      for key in odds:
          entropy += odds[key] * math.log2(1/odds[key])

      print("Entropia de la fuente de informacion:", entropy)
      print("Entropia de la fuente de informacion redondeada al tercer decimal:",
            ↪round(entropy,3))
```

Entropia de la fuente de informacion: 4.034650223284869

Entropia de la fuente de informacion redondeada al tercer decimal: 4.035

1.3 Ordena la fuente de información de mayor a menos probabilidad (tanto el símbolo, como la frecuencia absoluta, como la probabilidad).

```
[31]: # Se ordena el diccionario por probabilidad para imprimirlo
      oddsSorted = sorted(odds.items(), key=operator.itemgetter(1), reverse=True)

      print("Simbolo\t\tFrecuencia\tProbabilidad\tProbabilidad sin redondeo")
      for element in oddsSorted:
          key = element[0]
          print("'" +str(key)+ "'\t\t'" +str(frequencies[key])+ "'\t\t'"
                ↪+str(round(odds[key], 3))+ "'\t\t'" +str(odds[key])+ "'")
```

Simbolo	Frecuencia	Probabilidad	Probabilidad sin redondeo
' '	'40'	'0.18'	'0.18018018018018017'
'a'	'30'	'0.135'	'0.13513513513513514'
'e'	'21'	'0.095'	'0.0945945945945946'
's'	'13'	'0.059'	'0.05855855855855856'
'd'	'13'	'0.059'	'0.05855855855855856'
'r'	'12'	'0.054'	'0.05405405405405406'
'n'	'11'	'0.05'	'0.04954954954954955'
'l'	'11'	'0.05'	'0.04954954954954955'
'o'	'10'	'0.045'	'0.04504504504504504'
'c'	'10'	'0.045'	'0.04504504504504504'
'i'	'9'	'0.041'	'0.04054054054054054'
','	'6'	'0.027'	'0.02702702702702703'

'm'	'6'	'0.027'	'0.02702702702702703'
'b'	'4'	'0.018'	'0.018018018018018018'
'y'	'4'	'0.018'	'0.018018018018018018'
'h'	'3'	'0.014'	'0.013513513513513514'
'u'	'3'	'0.014'	'0.013513513513513514'
'.'	'3'	'0.014'	'0.013513513513513514'
'g'	'3'	'0.014'	'0.013513513513513514'
'v'	'2'	'0.009'	'0.009009009009009009'
'p'	'2'	'0.009'	'0.009009009009009009'
't'	'2'	'0.009'	'0.009009009009009009'
'L'	'1'	'0.005'	'0.0045045045045045045'
'V'	'1'	'0.005'	'0.0045045045045045045'
'E'	'1'	'0.005'	'0.0045045045045045045'
'f'	'1'	'0.005'	'0.0045045045045045045'

1.4 Extra: Obtener la fuente asociada al texto anterior, considerando como símbolos de la fuente las secuencias de longitud dos que van apareciendo en dicho texto. Ordena esta fuente de información de mayor a menor probabilidad.

```
[32]: # Se usa un bucle while para recorrer el texto de 2 en 2
frequenciesExtra = dict()
i = 0
while i < len(text)-1:
    symbol = text[i] + text[i+1]
    if symbol in frequenciesExtra:
        frequenciesExtra[symbol] += 1
    else:
        frequenciesExtra[symbol] = 1
    i += 2

# Se obtiene la frecuencia absoluta y probabilidades igual que en el ejercicio
↪ anterior
absFreqExtra = sum(frequenciesExtra.values())
oddsExtra = dict()

for key in frequenciesExtra:
    oddsExtra[key] = frequenciesExtra[key]/absFreqExtra

# Se ordena el diccionario por probabilidad para imprimirlo
oddsSortedExtra = sorted(oddsExtra.items(), key=operator.itemgetter(1),
↪ reverse=True)
print("Simbolo\t\tFrecuencia\tProbabilidad\tProbabilidad sin redondeo")
for element in oddsSortedExtra:
    key = element[0]
    print("'" + str(key) + "'\t\t'" + str(frequenciesExtra[key]) + "'\t\t'"
↪ + str(round(oddsExtra[key], 3)) + "'\t\t'" + str(oddsExtra[key]) + "'")
```

Simbolo	Frecuencia	Probabilidad	Probabilidad sin redondeo
' , '	'5'	'0.045'	'0.04504504504504504'
'de'	'5'	'0.045'	'0.04504504504504504'
'e '	'4'	'0.036'	'0.036036036036036036'
' c'	'3'	'0.027'	'0.02702702702702703'
'a '	'3'	'0.027'	'0.02702702702702703'
'as'	'3'	'0.027'	'0.02702702702702703'
'o '	'3'	'0.027'	'0.02702702702702703'
' d'	'3'	'0.027'	'0.02702702702702703'
'br'	'2'	'0.018'	'0.018018018018018018'
' m'	'2'	'0.018'	'0.018018018018018018'
'ad'	'2'	'0.018'	'0.018018018018018018'
'a.'	'2'	'0.018'	'0.018018018018018018'
' '	'2'	'0.018'	'0.018018018018018018'
'ga'	'2'	'0.018'	'0.018018018018018018'
's '	'2'	'0.018'	'0.018018018018018018'
'es'	'2'	'0.018'	'0.018018018018018018'
'y '	'2'	'0.018'	'0.018018018018018018'
'er'	'2'	'0.018'	'0.018018018018018018'
'ra'	'2'	'0.018'	'0.018018018018018018'
' l'	'2'	'0.018'	'0.018018018018018018'
'am'	'2'	'0.018'	'0.018018018018018018'
'll'	'2'	'0.018'	'0.018018018018018018'
'nc'	'2'	'0.018'	'0.018018018018018018'
'ca'	'2'	'0.018'	'0.018018018018018018'
'La'	'1'	'0.009'	'0.009009009009009009'
' n'	'1'	'0.009'	'0.009009009009009009'
'oc'	'1'	'0.009'	'0.009009009009009009'
'he'	'1'	'0.009'	'0.009009009009009009'
'ae'	'1'	'0.009'	'0.009009009009009009'
'um'	'1'	'0.009'	'0.009009009009009009'
'os'	'1'	'0.009'	'0.009009009009009009'
'ya'	'1'	'0.009'	'0.009009009009009009'
' y'	'1'	'0.009'	'0.009009009009009009'
'or'	'1'	'0.009'	'0.009009009009009009'
'Va'	'1'	'0.009'	'0.009009009009009009'
'cl'	'1'	'0.009'	'0.009009009009009009'
'ar'	'1'	'0.009'	'0.009009009009009009'
'id'	'1'	'0.009'	'0.009009009009009009'
'al'	'1'	'0.009'	'0.009009009009009009'
'va'	'1'	'0.009'	'0.009009009009009009'
've'	'1'	'0.009'	'0.009009009009009009'
'rd'	'1'	'0.009'	'0.009009009009009009'
' p'	'1'	'0.009'	'0.009009009009009009'
'du'	'1'	'0.009'	'0.009009009009009009'
'n '	'1'	'0.009'	'0.009009009009009009'
'tr'	'1'	'0.009'	'0.009009009009009009'
'to'	'1'	'0.009'	'0.009009009009009009'

'rr'	'1'	'0.009'	'0.009009009009009009'
'ig'	'1'	'0.009'	'0.009009009009009009'
'le'	'1'	'0.009'	'0.009009009009009009'
'si'	'1'	'0.009'	'0.009009009009009009'
'El'	'1'	'0.009'	'0.009009009009009009'
'in'	'1'	'0.009'	'0.009009009009009009'
'su'	'1'	'0.009'	'0.009009009009009009'
'be'	'1'	'0.009'	'0.009009009009009009'
'en'	'1'	'0.009'	'0.009009009009009009'
' s'	'1'	'0.009'	'0.009009009009009009'
'om'	'1'	'0.009'	'0.009009009009009009'
'pa'	'1'	'0.009'	'0.009009009009009009'
'ni'	'1'	'0.009'	'0.009009009009009009'
' f'	'1'	'0.009'	'0.009009009009009009'
'ia'	'1'	'0.009'	'0.009009009009009009'
'hi'	'1'	'0.009'	'0.009009009009009009'
'ba'	'1'	'0.009'	'0.009009009009009009'
'io'	'1'	'0.009'	'0.009009009009009009'
'ne'	'1'	'0.009'	'0.009009009009009009'
's, '	'1'	'0.009'	'0.009009009009009009'
'ns'	'1'	'0.009'	'0.009009009009009009'
'an'	'1'	'0.009'	'0.009009009009009009'
'ci'	'1'	'0.009'	'0.009009009009009009'
' a'	'1'	'0.009'	'0.009009009009009009'
'nh'	'1'	'0.009'	'0.009009009009009009'
'el'	'1'	'0.009'	'0.009009009009009009'
'o.'	'1'	'0.009'	'0.009009009009009009'

por Adrián Pérez García, 12/03/2021