Marín Parra José Guadalupe de Jesús
Rangel Limón Erik

1 Alineamientos con secuencias

1. Con el siguiente esquema de puntaje $\sigma = 2$, $\delta(a, b) = -1$ y $\delta(a, a) = 1$ haz los siguientes alineamientos.

   a) $g1 = CTTAGA$ y $g2 = GTAA$

   Solución. Escribamos las secuencias en una tabla para encontrar los alineamientos. La tabla queda de la siguiente manera.

S	-	C	T	T	A	G	A
-	0	0	0	0	0	0	0
G	0	0	0	0	0	1	0
T	0	0	1	1	0	0	0
A	0	0	0	0	2	0	1
A	0	0	0	0	1	0	1

   Por lo que el alineamiento el TA.

```
CTTAGA
 ||
G-TAA-
```

   b) $g1 = ACCCTACCT$ y $g2 = AGCCTCT$

   Solución. Escribamos las secuencias en una tabla para encontrar los alineamientos. La tabla queda de la siguiente manera.

S	-	A	C	C	C	T	A	C	C	T
-	0	0	0	0	0	0	0	0	0	0
A	0	1	0	0	0	0	1	0	0	0
G	0	0	0	0	0	0	0	0	0	0
C	0	0	1	1	1	0	0	1	1	0
C	0	0	1	2	2	0	0	1	2	0
T	0	0	0	0	1	3	1	0	0	3
C	0	0	1	1	1	1	2	2	1	1
T	0	0	0	0	0	2	0	1	1	2

   Por lo que el alineamiento el TA.

```
A-CCCTACCT
|   | | |
AG-CCTCT--
```

2. Implementa el algoritmo Needleman-Wunsch como lo vimos en clase para computar el alineamiento global entre dos secuencias. Tu implementación deberá reportar el *score* máximo en la entrada $S_{n,m}$ así como las secuencias apareadas. Una consideración que deberás tomar en cuenta para la reconstrucción es que en caso de que en una entrada de la matriz tengas más de una opción para alinear (i.e. arriba y en diagonal o izquierda y en diagonal), solo considerarás la opción diagonal.

   Prueba tu implementación con las secuencias del primer ejercicio.

   Solución.

   a) $g1 = CTTAGA$ y $g2 = GTAA$

   Solución. Escribamos las secuencias en una tabla para encontrar los alineamientos. La tabla queda de la siguiente manera.

S	-	C	T	T	A	G	A
-	0	-2	-4	-6	-8	-10	-12
G	-2	-1	-3	-5	-7	-7	-9
T	-4	-3	0	-2	-4	-6	-8
A	-6	-5	-2	-1	-1	-3	-5
A	-8	-7	-4	-3	0	-2	-2

   Por lo que el alineamiento queda de la siguiente manera con el *score* máximo = -2.

```
CTTAGA
 | | |
GT-A-A
```

   b) $g1 = ACCCTACCT$ y $g2 = AGCCTCT$

   Solución. Escribamos las secuencias en una tabla para encontrar los alineamientos. La tabla queda de la siguiente manera.

S	-	A	C	C	C	T	A	C	C	T
-	0	-2	-4	-6	-8	-10	-12	-14	-16	-18
A	-2	1	-1	-3	-5	-7	-9	-11	-13	-15
G	-4	-1	0	-2	-4	-6	-8	-10	-12	-14
C	-6	-3	0	1	-1	-3	-5	-7	-9	-11
C	-8	-5	-2	1	2	0	-2	-4	-6	-8
T	-10	-7	-4	-1	0	3	1	-1	-3	-5
C	-12	-9	-6	-3	0	1	2	2	0	-2
T	-14	-11	-8	-5	-2	1	0	1	1	1

   Por lo que el alineamiento queda de la siguiente manera con el *score* máximo = 1.

```
AC CCTACCT
|  | || | |
AGCCT-C-T
```

3. Realiza un alineamiento usando Smith-Waterman de las secuencias $g1 = AGCGTAG$ y $g2 = CTCGTG$.

   Solución. Escribamos las secuencias en una tabla para encontrar los alineamientos. La tabla queda de la siguiente manera.

S	-	A	G	C	G	T	A	G
-	0	0	0	0	0	0	0	0
C	0	0	0	1	0	0	0	0
T	0	0	0	0	0	1	0	0
C	0	0	0	1	0	0	0	0
G	0	0	1	0	2	0	0	1
T	0	0	0	0	0	3	1	0
G	0	0	1	0	1	1	2	2

   Por lo que el alineamiento el CGT.

```
AGCGTAG
| | |
CTCGTG-
```

2 Información mútua

1. Recordando que la definición de **entropía de Shannon** de la variable aleatoria $X = x_1, x_2, x_3, ..., x_n$ es como sigue

$$H(X) = -\Sigma p(x_i) \cdot log_2(p(x_i))$$

   Responde las siguientes preguntas.

   a) ¿Cuál es la entropía de Shannon en bits de las siguientes cadenas? En cada caso reporta la distribución de los símbolos que las conforman.

      1) 00101111010101001000000010101 1

      Solución. Tenemos que el alfabeto de la cadena es $0, 1$.
      Y las frecuencias de los símbolos es la siguiente.
      $0 \rightarrow 0.567$
      $1 \rightarrow 0.433$
      Por lo que calculamos la entropía quedando de la siguiente manera.
      $H(X) = -[(0.567log_2(0.567)) + (0.433log_2(0.433))]$
      $H(X) = -[(-0.464) + (-0.523)]$
      $H(X) = -[-0.987]$
      $H(X) = 0.987$

      Por lo tanto, la Entropía en bits es 0.987

2) 10011110011110111011011100111
Solución. Tenemos que el alfabeto de la cadena es $0, 1$.
Y las frecuencias de los símbolos es la siguiente.
$0 \to 0.3$
$1 \to 0.7$
Por lo que calculamos la entropía quedando de la siguiente manera.
$H(X) = -[(0.3log_2(0.3)) + (0.7log_2(0.7))]$
$H(X) = -[(-0.521) + (-0.36)]$
$H(X) = -[-0.881]$
$H(X) = 0.881$

Por lo tanto, la Entropía en bits es 0.881

3) 00000000000000011111111111111111
Solución. Tenemos que el alfabeto de la cadena es $0, 1$.
Y las frecuencias de los símbolos es la siguiente.
$0 \to 0.5$
$1 \to 0.5$
Por lo que calculamos la entropía quedando de la siguiente manera.
$H(X) = -[(0.5log_2(0.5)) + (0.5log_2(0.5))]$
$H(X) = -[(-0.5) + (-0.5)]$
$H(X) = -[-1]$
$H(X) = 1$

Por lo tanto, la Entropía en bits es 1

b) Considerando que la función de Información Mútua se puede calcular como

$$I(X : Y) = H(X) + H(Y) - H(X, Y)$$

donde $H(X, Y)$ es la entropía conjunta, calcula la información mútua para las siguientes secuencias e indica en una tabla de contingencias entre las variables la correlación entre cada una de ellas.

1) 00101111010101111000000010101101010111111010000101
La cadena la denotaremos como X_1 y el alfabeto de la cadena es 0, 1
La frecuencia de los símbolos es la siguiente

$$0 \to \frac{12}{25}$$

$$1 \to \frac{13}{25}$$

De manera que

$$H(X_1) = -\left(\frac{12}{25}\log_2\left(\frac{12}{25}\right) + \frac{13}{25}\log_2\left(\frac{13}{25}\right)\right)$$

$$H(X_1) = -\left(\frac{12}{25}(\log_2(12) - \log_2(25)) + \frac{13}{25}(\log_2(13) - \log_2(25)\right)$$

$$H(X_1) = -(-0.998846)$$

Marín Parra José Guadalupe de Jesús
Rangel Limón Erik

$$H(X_1) = 0.998846$$

2) 00110111011101111000011110101001010111111010000101
La cadena la denotaremos como X_2 y el alfabeto de la cadena es 0, 1
La frecuencia de los símbolos es la siguiente

$$0 \to \frac{21}{50}$$

$$1 \to \frac{29}{50}$$

De manera que

$$H(X_2) = -\left(\frac{21}{50}\log_2\left(\frac{21}{50}\right) + \frac{29}{50}\log_2\left(\frac{29}{50}\right)\right)$$

$$H(X_2) = 0.981454$$

3) 10010110011110111011101110011010111010100010111010
La cadena la denotaremos como X_3 y el alfabeto de la cadena es 0, 1
La frecuencia de los símbolos es la siguiente

$$0 \to \frac{2}{5}$$

$$1 \to \frac{3}{5}$$

De manera que

$$H(X_3) = -\left(\frac{2}{5}\log_2\left(\frac{2}{5}\right) + \frac{3}{5}\log_2\left(\frac{3}{5}\right)\right)$$

$$H(X_3) = 0.970951$$

4) 100100111011101110111011100101001010101000010111010
La cadena la denotaremos como X_4 y el alfabeto de la cadena es 0, 1
La frecuencia de los símbolos es la siguiente

$$0 \to \frac{11}{25}$$

$$1 \to \frac{14}{25}$$

De manera que

$$H(X_4) = -\left(\frac{11}{25}\log_2\left(\frac{11}{25}\right) + \frac{14}{25}\log_2\left(\frac{14}{25}\right)\right)$$

$$H(X_4) = 0.989588$$

5) 10010111101010111011000001010100010101000010111010
La cadena la denotaremos como X_5 y el alfabeto de la cadena es 0, 1
La frecuencia de los símbolos es la siguiente

$$0 \to \frac{13}{25}$$

$$1 \to \frac{12}{25}$$

De manera que

$$H(X_5) = -\left(\frac{13}{25}\log_2\left(\frac{13}{25}\right) + \frac{12}{25}\log_2\left(\frac{12}{25}\right)\right)$$

$$H(X_5) = 0.998846$$

Solución.
Haremos una tabla de 5×5, que la llenaremos con la siguiente fórmula para cada elemento fila i y en la columna j

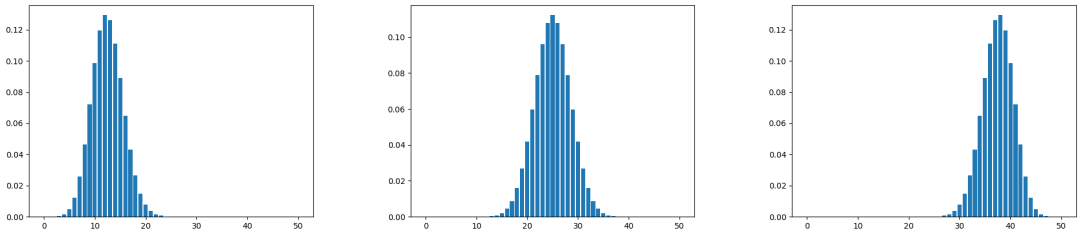
$$\begin{aligned} I(X_i : X_j) &= H(X_i) + H(X_j) - H(X,Y) \\ &= H(X_i) + H(X_j) + \sum p(x_i, x_j)\log_2(p(x_i, x_j)) \\ &= H(X_i) + H(X_j) + \sum p(x_i)p(x_j)\log_2(p(x_i)p(x_j)) \end{aligned}$$

Lo último dado que X_i y X_j son independientes (siempre que sean distintas).
Por lo tanto

	X_1	X_2	X_3	X_4	X_5
X_1	0.998846	5.68971×10^{-7}	8.6955×10^{-7}	9.42783×10^{-7}	-9.07199×10^{-6}
X_2	5.68971×10^{-7}	0.981454	5.10512×10^{-7}	5.83744×10^{-7}	5.68971×10^{-7}
X_3	8.6955×10^{-7}	5.10512×10^{-7}	0.970951	8.84323×10^{-7}	8.6955×10^{-7}
X_4	9.42783×10^{-7}	5.83744×10^{-7}	8.84323×10^{-7}	0.989588	9.42783×10^{-7}
X_5	-9.07199×10^{-6}	5.68971×10^{-7}	5.68971×10^{-7}	9.42783×10^{-7}	0.998846

Table 1: Tabla de contingencias

c) Reproduce la gráfica de un proceso Bernoulli. Para esta pregunta describe la construcción metemática o el código usado para tal reproducción. ¿Cuál es su interpretación?
Solución.
Supongamos que tenemos variables aleatoria $X_i \sim Bernuolli(p)$, que puede representar el tiro de una moneda en un momento discreto i, recordemos que $X_i \in \{0, 1\}$. Denotaremos a la variable aleatoria $B = \sum_{i=1}^{n} X_i$ al experimento de tirar n veces la misma moneda. Sabemos que $B \sim Binomial(n, p)$, por lo que la gráfica del proceso Bernuolli debe verse de manera similar a la gráfica de la densidad de B.
Recordemos que $\mathbb{P}(B = k) = \binom{n}{k}p^k(1-p)^{n-k}$
A manera de ejemplo podremos algunas gráficas de la densidad de B, con distintos valores n y p.
Graficar éstas funciones se hicieron mediante las siguientes funciones en python.



(a) Densidad con $n = 50$ y $p = \frac{1}{4}$ (b) Densidad con $n = 50$ y $p = \frac{1}{2}$ (c) Densidad con $n = 50$ y $p = \frac{3}{4}$

Figure 1: Algunas densidades de B

```python
import matplotlib.pyplot as plt
import numpy as np

def factorial(n):
    fact = 1
    for i in range (1,n+1):
        fact = fact * i
    return fact

def tail(n,t):
    fact = 1
    for i in range(n-t+1,n+1):
        fact = fact * i
    return fact

def binom(n,m):
    return tail(n,m)/factorial(m)

def binomial(n,p,x):
    return binom(n,x) * p**(x) * (1-p)**(n-x)

def plotbinomial(n,p):
    x = np.array(range(n+1))
    y = np.array([ binomial(n,p,i) for i in x ])
    plt.bar(x,y)
    plt.show()
```

Y luego, para recrear el proceso Bernuolli, crearemos la variable aleatoria X, el cual llamaremos `Moneda`, ésta va a guardar la probabilidad que tiene que el resultado sea águila, resultado el cual denotaremos como 1. Ésta moneda podrá ser lanzada y nos regresará 0 o 1 con la probabilidad que guarda la moneda.

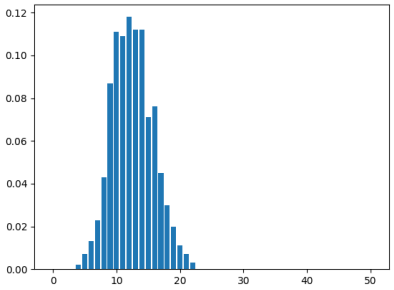
Para regresar 0 o 1 con la probabilidad que nos dice la moneda, si debe tener una probabilidad p de que salga 1, redondearemos los dos primeros dígitos después del punto decimal de p y al resultado lo multiplicamos por 100, llamaremos a éste resultado a. Creamos un arreglo B de 100 elementos donde a elementos del arreglo serán 1 y $100 - a$ elementos serán 0. Finalmente escogemos un elemento de B de manera uniforme para representar el tiro de la moneda.

Si queremos ver el resultado del proceso Bernuolli debemos repetir los n tiros de la moneda varias veces, cada vez que repitamos el experimento al terminar los n tiros, si apareció m veces águila durante los n tiros, en un arreglo R de tamaño $n + 1$ hacemos $R[m] \leftarrow R[m] + 1$, y esto lo repetimos varias veces hasta terminar el proceso.

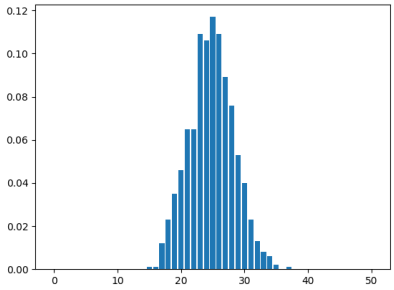
Al terminar éste último proceso, a cada elemento del arreglo R lo dividimos entre las veces

que repetimos el experimento, para aproximar la probabilidad con la que el experimento resultó $R[i]$.

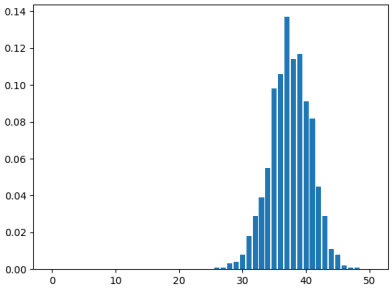
Al terminar sólo graficamos el histograma del arreglo R, y tendríamos algo como sigue:



(a) $1,000$ intentos, $n = 50$, $p = \frac{1}{4}$



(b) $1,000$ intentos, $n = 50$, $p = \frac{1}{2}$



(c) $1,000$ intentos, $n = 50$, $p = \frac{3}{4}$

Figure 2: Caption

Utilizamos el siguiente código en python para graficarlas

```
import matplotlib.pyplot as plt
import numpy as np
import random

class Moneda:

    def __init__(self,probabilidad):
        self.probabilidad = probabilidad

    def lanzar(self):
        a = round(self.probabilidad, 2)
        a = int(a * 100)
        x = [ 1 for _ in range(a) ]
        x.extend([ 0 for _ in range(100-a)])
        return x[random.randint(0,99)]

def procesobernuolli(moneda, tiros, intentos):
    y = [ 0 for _ in range(tiros+1) ]
    for _ in range(intentos):
        i = np.sum([ moneda.lanzar() for _ in range(tiros) ])
        y[i] = y[i] + 1
    for j in range(tiros+1):
```

```
23          y[j] = y[j]/intentos
24      x = np.array(range(tiros+1))
25      plt.bar(x,y)
26      plt.show()
```

Y notemos que las últimas gráficas son muy similares a las que calculamos anteriormente. Podemos interpretar el resultado anterior a que siempre que repitamos el proceso Bernuolli obtendremos por lo general un resultado entre cierto rango la gran mayoría de las veces.