

Total de píxeles en la imagen: 2560 x 1440 = 3686400

En cada píxel hay 30 bits, por lo tanto, la cantidad de información total será de:

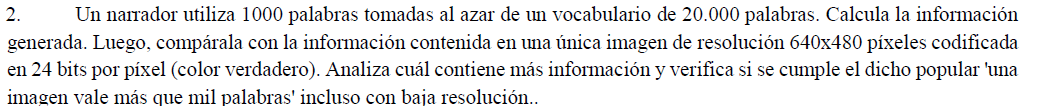
Total de píxeles \* 30 = 110592000 bits

Para el segundo caso vamos a tener

Total de píxeles en la imagen: 7680 x 4320 = 33177600

Se nos dice que hay 12 bits por canal de color, como son 3 los colores (RGB) vamos a tener en total 36 bits por píxel. En otras palabras, podemos decir que hay 36 bits de información por píxel. Por lo tanto, la información total será de:

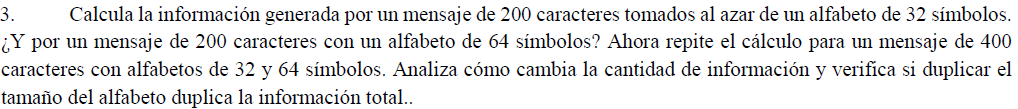
Total de píxeles \* 36 = 1194393600 bits



Calculamos la información generada por el narrador I = 1000 \* -log2(1/20000) = 14287,71 bits

Luego tenemos 640x480 = 307200 píxeles, entonces la información contenida en una imágen de dicha resolución será de 24 \* 307200 = 7372800 bits

Podemos ver que 14287,21 < 7372800. Por lo tanto, se cumple el dicho “una imagen vale más que mil palabras”



Información generada por un mensaje de 200 caracteres pertenecientes a un alfabeto de 32 símbolos

I = 200 \* -log2(1/32) = 1000 bits

Información generada por un mensaje de 200 caracteres pertenecientes a un alfabeto de 64 símbolos

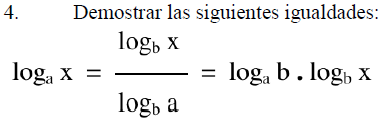
I = 200 \* -log2(1/64) = 1200 bits

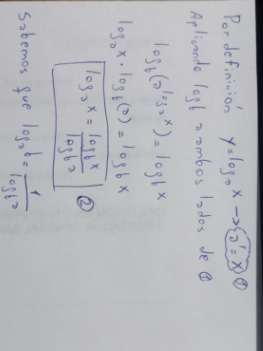
Información generada por un mensaje de 400 caracteres pertenecientes a un alfabeto de 32 símbolos (A) y pertenecientes a un alfabeto de 64 símbolos (B)

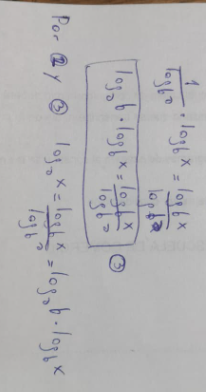
IA = 400 \* -log2(1/32) = 2000 bits

IB = 400 \* -log2(1/64) = 2400 bits

Si analizamos la información generada por carácter para un alfabeto de 32 símbolos y 64 símbolos tendremos que es de 5 bits y 6 bits respectivamente. Por lo tanto, duplicar la cantidad de símbolos del alfabeto no duplica la información total, solo incrementa en 1 bit. Pero duplicar el tamaño del mensaje si lo hace.









Información individual de los símbolos:

I1 = -log2(p1) = -log2(0,4) = 1,32 bits

I2 = -log2(p2) = -log2(0,2) = 2,32 bits

I3 = -log2(p3) = -log2(0,15) = 2,73 bits

I4 = -log2(p4) = -log2(0,1) = 3,32 bits

I5 = -log2(p5) = -log2(0,1) = 3,32 bits

I6 = -log2(p6) = -log2(0,05) = 4,32 bits

Entropía de la fuente:

H(F) = = 0,4 \* 1,32 + 0,2 \* 2,32 + 0,15 \* 2,73 + 0,1 \* 3,32 + 0,1 \* 3,32 + 0,05 \* 4,32 = 2,28 bits



La entropía mide la cantidad de información promedio. La distribución uniforme es una distribución de probabilidad donde los eventos tienen la misma probabilidad de ocurrir. Entonces, si todos los símbolos son igual de probables, la incertidumbre es máxima, porque antes de observar el evento no tenemos idea de cuál ocurrirá.

Calculamos la entropía para el dado justo

H(A)= = - (1/6 \* log2(1/6) + 1/6 \* log2(1/6) + 1/6 \* log2(1/6) + 1/6 \* log2(1/6) + 1/6 \* log2(1/6) + 1/6 \* log2(1/6)) = 2,58 bits

Calculamos la entropía para el dado sesgado

H(B)= = -(0,3 \* log2(0,3) + 0,25 \* log2(0,25) + 0,15 \* log2(0,15) + 0,15 \* log2(0,15) + 0,1 \* log2(0,1) + 0,05 \* log2(0,05)) = 2,39 bits

Como se puede ver, tenemos que la entropía para el dado justo es mayor que la del dado sesgado y esto ocurre debido a que en el primero no es posible saber con anticipación cuál de las 6 caras saldrá, justamente debido a que las 6 caras tienen la misma probabilidad. Mientras que, para el segundo, tenemos algo de idea de que cara saldrá primero, ya que las probabilidades de las caras difieren, en este caso es mucho más probable que salga primero la primera cara con respecto al resto de las caras.



**Cálculo de información obtenida en cada pesada**

li=log r (1/Pi)

En esta fórmula tenemos que:

* logr es log2.
* Pi es ⅓ debido a que en cada pesada podemos obtener uno de 3 eventos distintos los cuales son: platos iguales, plato izquierdo más arriba que el derecho o al revés.

Con estos datos vamos a obtener que:

li=log2 (1/(1/3))=log2 3=1.58 bits de información

**Cálculo de cantidad de pesadas a realizar para encontrar la moneda distinta y su peso**

l=log2 (2x cantidad de monedas)=log 2 24=4.58 bits de información

Dado que ahora sabemos que para determinar cuál es la moneda distinta y su peso se requieren 4.58 bits de información, podemos concluir que será necesario realizar 3 pesadas, ya que 3 × 1.58 = 4.74.



Para una letra (alfabeto de M=28):

* 𝐼1 = log2(27) = 4,754 bits

Para un par de letras (suponiendo independencia y equiprobabilidad):

* 𝐼2 = log⁡2(272) = 9,509 bits

Para una terna de letras (mismas suposiciones):

* 𝐼3 = log2(273) = 14,264 bits



En la escritura real en castellano, la información efectiva va a ser menor porque:

* Las letras no tienen la misma frecuencia. Por ejemplo, las letras a, e, s, o, n aparecen muchísimo más que las letras w y k.
* Existen fuertes dependencias entre ellas. Por ejemplo, si viene una q casi seguro que viene una u.
* Hay redundancia lingüística.

# 

1. **No negatividad**: I(A)≥0I(A).  
    *Ejemplo*: si P(A)=1/2,I(A)=−log⁡2(1/2)=1 bit ≥0.
2. **Monotonía (más raro ⇒ más información)**: si P(A)<P(B), entonces I(A)>I(B).  
    *Ejemplo*: sacar un “6” en un dado justo: I=log2​6≈2.585 bits es mayor que lanzar “cara” en una moneda: 1 bit.
3. **Certeza da 0 información**: si P(A)=1, I(A)=0.  
    *Ejemplo*: “El sol sale por el este” (evento seguro en el modelo) → 0 bits.
4. **Aditividad para eventos independientes**: si AAA y BBB son independientes,  
    I(A∩B)=I(A)+I(B)  
    *Ejemplo*: moneda justa y dado justo: ver “cara” **y** “6” ⇒ 1+log⁡261+\log2 61+log2​6 bits.
5. **Regla de cadena (dependencias)**: I(A,B)=I(A)+I(B∣A) Ejemplo: en español, después de “q” casi siempre viene “u”, por lo que 𝐼(u’∣‘q’)I(‘u’∣‘q’) es pequeño; la información conjunta de “qu” es menor que si fueran independientes.
6. **Cambio de base**: cambiar la base solo reescala:  
   Ibase b(A)=log2 b / Ibase 2(A)  
    *Ejemplo*: 1 bit equivale a log10​ 2≈ 0.3010 dígitos decimales.



### Paso 1: Establecer la fórmula a minimizar

La cantidad de información I necesaria para representar V sucesos en una base b es igual al número de dígitos L requeridos. La relación es V=bL. Tomando el logaritmo de ambos lados, se obtiene: L = I = logb​(V) =

La relación a minimizar es I⋅b. Sustituyendo la expresión para I, la función a minimizar es:

f(b) =

### Paso 2: Derivar la función e igualar a cero

Para encontrar el mínimo de la función f(b), se deriva con respecto a b y se iguala a cero. Usando la regla del cociente para la derivación:

f′(b) = ln(V)⋅(ln(b)⋅1−b⋅1​/b)/(ln(b))2

f′(b) = ln(V)⋅(ln(b)−1​/(ln(b))2)

Para que la derivada sea cero, el numerador debe ser cero, ya que ln(V) es una constante positiva y (ln(b))2 no puede ser cero:

ln(b)−1 = 0

### Paso 3: Resolver para b

Resolviendo la ecuación del paso anterior, se obtiene el valor de la base óptima b:

ln(b) = 1

b = e

El objetivo es minimizar la relación I⋅b, que representa la cantidad de información por base. La variedad de sucesos es V = 2000.

### Conclusión

La base óptima de representación que minimiza la cantidad de información total es e≈2.71828. Aunque el valor de V=2000 es relevante para el cálculo de la cantidad de información total, no afecta el valor de la base óptima, ya que es una constante universal para este tipo de problemas de optimización.



**Paso 1: Frecuencia de caracteres**

Para calcular la entropía, primero debemos contar la frecuencia de cada carácter en ambos textos. Se consideran mayúsculas y minúsculas como símbolos distintos. El carácter "espacio" también se considera un símbolo.

1. Texto QR

| Carácter | Frecuencia | Probabilidad (pi​) |
| --- | --- | --- |
| e' | 31 | 0.1286 |
| o' | 22 | 0.0913 |
| a' | 21 | 0.0871 |
| s' | 20 | 0.0830 |
| t' | 17 | 0.0705 |
| r' | 16 | 0.0664 |
| l' | 16 | 0.0664 |
| n' | 15 | 0.0622 |
| i' | 13 | 0.0539 |
| p' | 10 | 0.0415 |
| u' | 10 | 0.0415 |
| c' | 9 | 0.0373 |
| m' | 9 | 0.0373 |
| d' | 8 | 0.0332 |
| g' | 7 | 0.0290 |
| q' | 1 | 0.0041 |
| h' | 1 | 0.0041 |
| k' | 1 | 0.0041 |
| y' | 1 | 0.0041 |
| w' | 1 | 0.0041 |
| S' | 1 | 0.0041 |
| b' | 1 | 0.0041 |
| .' | 2 | 0.0083 |
| ' | 19 | 0.0788 |

2. Texto Corregido

| Carácter | Frecuencia | Probabilidad (pi​) |
| --- | --- | --- |
| a' | 31 | 0.1152 |
| e' | 28 | 0.1041 |
| o' | 22 | 0.0818 |
| s' | 22 | 0.0818 |
| t' | 18 | 0.0669 |
| r' | 16 | 0.0595 |
| l' | 16 | 0.0595 |
| n' | 15 | 0.0558 |
| i' | 13 | 0.0483 |
| p' | 10 | 0.0372 |
| u' | 10 | 0.0372 |
| c' | 9 | 0.0335 |
| m' | 9 | 0.0335 |
| d' | 8 | 0.0297 |
| g' | 7 | 0.0260 |
| y' | 1 | 0.0037 |
| j' | 1 | 0.0037 |
| b' | 1 | 0.0037 |
| h' | 1 | 0.0037 |
| S' | 1 | 0.0037 |
| .' | 2 | 0.0074 |
| ' | 25 | 0.0929 |

### Paso 2: Cálculo de la entropía

Usando la fórmula de entropía de orden cero, se calcula la entropía para cada texto.

**1. Entropía del Texto QR**

* H0​=−∑pi​⋅log2​(pi​)
* H0​=−(0.1286⋅log2​(0.1286)+0.0913⋅log2​(0.0913)+...+0.0041⋅log2​(0.0041))
* H0​≈4.15 **bits/símbolo**

**2. Entropía del Texto Corregido**

* H0​=−∑pi​⋅log2​(pi​)
* H0​=−(0.1152⋅log2​(0.1152)+0.1041⋅log2​(0.1041)+...+0.0037⋅log2​(0.0037))
* H0​≈4.21 **bits/símbolo**

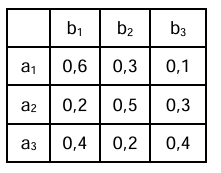
### Comparación

* **Entropía del texto QR:** 4.15 bits/símbolo
* **Entropía del texto Corregido:** 4.21 bits/símbolo

La entropía del texto corregido es ligeramente mayor. Esto sugiere que su distribución de caracteres es más uniforme (es decir, las probabilidades de cada carácter son más parecidas), lo que lo hace un poco más impredecible y, por lo tanto, más "rico" en información desde el punto de vista de la entropía de orden cero.

**PRÁCTICO 2 CANAL DE INFORMACIÓN**

1) Sea el siguiente canal:



Calcular los valores de p(ai/bj) y las probabilidades de salida para el caso particular de p(a1)=0.5, p(a2)=0.25, p(a3)=0.25

**Cálculo de Probabilidades a Posteriori**

****

Dado los valores de entrada p(a1​)=0.5, p(a2​)=0.25 y p(a3​)=0.25:

* **Para p(b1​)**: p(b1​)=p(b1​∣a1​)\*p(a1​)+p(b1​∣a2​)\*p(a2​)+p(b1​∣a3​)*p(a3​) p(b1​)=(0.6)\**(0.5)+(0.2)\*(0.25)+(0.4)\*(0.25)=0.3+0.05+0.1=0.45
* **Para p(b2​)**: p(b2​)=p(b2​∣a1​)\*p(a1​)+p(b2​∣a2​)\*p(a2​)+p(b2​∣a3​)\*p(a3​) p(b2​)=(0.3)\*(0.5)+(0.5)(0.25)+(0.2)(0.25)=0.15+0.125+0.05=0.325
* **Para p(b3​)**: p(b3​)=p(b3​∣a1​)p(a1​)+p(b3​∣a2​)p(a2​)+p(b3​∣a3​)p(a3​) p(b3​)=(0.1)(0.5)+(0.3)(0.25)+(0.4)(0.25)=0.05+0.075+0.1=0.225

Verificamos que la suma de las probabilidades de salida sea 1: 0.45+0.325+0.225=1.

**Cálculo de Probabilidades Condicionales Inversas p(ai|bj)**

****

* Para p(a1​∣b1​): p(a1​∣b1​)=p(b1​)p(b1​∣a1​)p(a1​)​=0.45(0.6)(0.5)​=0.450.3​≈0.667
* Para p(a2​∣b1​): p(a2​∣b1​)=p(b1​)p(b1​∣a2​)p(a2​)​=0.45(0.2)(0.25)​=0.450.05​≈0.111
* Para p(a3​∣b1​): p(a3​∣b1​)=p(b1​)p(b1​∣a3​)p(a3​)​=0.45(0.4)(0.25)​=0.450.1​≈0.222

Verificación: 0.667+0.111+0.222≈1.

* Para p(a1​∣b2​): p(a1​∣b2​)=p(b2​)p(b2​∣a1​)p(a1​)​=0.325(0.3)(0.5)​=0.3250.15​≈0.462
* Para p(a2​∣b2​): p(a2​∣b2​)=p(b2​)p(b2​∣a2​)p(a2​)​=0.325(0.5)(0.25)​=0.3250.125​≈0.385
* Para p(a3​∣b2​): p(a3​∣b2​)=p(b2​)p(b2​∣a3​)p(a3​)​=0.325(0.2)(0.25)​=0.3250.05​≈0.154

Verificación: 0.462+0.385+0.154≈1.

* Para p(a1​∣b3​): p(a1​∣b3​)=p(b3​)p(b3​∣a1​)p(a1​)​=0.225(0.1)(0.5)​=0.2250.05​≈0.222
* Para p(a2​∣b3​): p(a2​∣b3​)=p(b3​)p(b3​∣a2​)p(a2​)​=0.225(0.3)(0.25)​=0.2250.075​≈0.333
* Para p(a3​∣b3​): p(a3​∣b3​)=p(b3​)p(b3​∣a3​)p(a3​)​=0.225(0.4)(0.25)​=0.2250.1​≈0.444

Verificación: 0.222+0.333+0.444≈1.

2) Considera un Canal Binario Asimétrico con las siguientes probabilidades:

* p(a1)=0,4
* p(b1/a1)=4/5
* p(b1/a2)=1/4

**obtención del resto de probabilidades**

Probabilidad de la segunda entrada: p(a2​)=1−p(a1​)=1−0.4=0.6.

Probabilidades de canal restantes:

* p(b2​∣a1​)=1−p(b1​∣a1​)=1−0.8=0.2.
* p(b2​∣a2​)=1−p(b1​∣a2​)=1−0.25=0.75.

a) Calcula las probabilidades condicionales hacia atrás y las probabilidades conjuntas.

#### 1. Probabilidades Conjuntas p(ai​,bj​)

Se calculan usando la fórmula: p(ai​,bj​)=p(bj​∣ai​)⋅p(ai​).

* p(a1​,b1​)=p(b1​∣a1​)⋅p(a1​)=0.8⋅0.4=0.32
* p(a1​,b2​)=p(b2​∣a1​)⋅p(a1​)=0.2⋅0.4=0.08
* p(a2​,b1​)=p(b1​∣a2​)⋅p(a2​)=0.25⋅0.6=0.15
* p(a2​,b2​)=p(b2​∣a2​)⋅p(a2​)=0.75⋅0.6=0.45

Verificación: 0.32+0.08+0.15+0.45=1.

#### 2. Probabilidades de Salida p(bj​)

Se calculan sumando las probabilidades conjuntas por columna:

* p(b1​)=p(a1​,b1​)+p(a2​,b1​)=0.32+0.15=0.47
* p(b2​)=p(a1​,b2​)+p(a2​,b2​)=0.08+0.45=0.53

Verificación: 0.47+0.53=1.

#### 3. Probabilidades Condicionales Hacia Atrás p(ai​∣bj​)

* p(a1​∣b1​)=p(b1​)p(a1​,b1​)​=0.470.32​≈0.6809
* p(a2​∣b1​)=p(b1​)p(a2​,b1​)​=0.470.15​≈0.3191
* p(a1​∣b2​)=p(b2​)p(a1​,b2​)​=0.530.08​≈0.1509
* p(a2​∣b2​)=p(b2​)p(a2​,b2​)​=0.530.45​≈0.8491

b) Obtén la entropía del emisor, y las entropías condicionales de la fuente para cualquier símbolo de salida.



#### 1. Entropía del Emisor (Fuente) H(A)

* H(A)=−[p(a1​)log2​p(a1​)+p(a2​)log2​p(a2​)]
* H(A)=−[0.4log2​0.4+0.6log2​0.6]
* H(A)=−[0.4(−1.3219)+0.6(−0.7370)]=−[−0.5288−0.4422]=0.971 bits.

### Datos del canal

Se nos da un Canal Binario Asimétrico con las siguientes probabilidades:

* p(a1​)=0.4
* p(b1​∣a1​)=4/5=0.8
* p(b1​∣a2​)=1/4=0.25

A partir de estas probabilidades, podemos deducir el resto de las probabilidades de entrada y de canal. Dado que es un canal binario, solo hay dos entradas (a1​,a2​) y dos salidas (b1​,b2​).

* **Probabilidad de la segunda entrada**: p(a2​)=1−p(a1​)=1−0.4=0.6.
* **Probabilidades de canal restantes**:
  + p(b2​∣a1​)=1−p(b1​∣a1​)=1−0.8=0.2.
  + p(b2​∣a2​)=1−p(b1​∣a2​)=1−0.25=0.75.

### a) Probabilidades Condicionales y Conjuntas

#### 1. Probabilidades Conjuntas p(ai​,bj​)

Se calculan usando la fórmula: p(ai​,bj​)=p(bj​∣ai​)⋅p(ai​).

* p(a1​,b1​)=p(b1​∣a1​)⋅p(a1​)=0.8⋅0.4=0.32
* p(a1​,b2​)=p(b2​∣a1​)⋅p(a1​)=0.2⋅0.4=0.08
* p(a2​,b1​)=p(b1​∣a2​)⋅p(a2​)=0.25⋅0.6=0.15
* p(a2​,b2​)=p(b2​∣a2​)⋅p(a2​)=0.75⋅0.6=0.45

Verificación: 0.32+0.08+0.15+0.45=1.

#### 2. Probabilidades de Salida p(bj​)

Se calculan sumando las probabilidades conjuntas por columna:

* p(b1​)=p(a1​,b1​)+p(a2​,b1​)=0.32+0.15=0.47
* p(b2​)=p(a1​,b2​)+p(a2​,b2​)=0.08+0.45=0.53

Verificación: 0.47+0.53=1.

#### 3. Probabilidades Condicionales Hacia Atrás p(ai​∣bj​)

Se calculan usando el Teorema de Bayes: p(ai​∣bj​)=p(bj​)p(ai​,bj​)​.

* p(a1​∣b1​)=p(b1​)p(a1​,b1​)​=0.470.32​≈0.6809
* p(a2​∣b1​)=p(b1​)p(a2​,b1​)​=0.470.15​≈0.3191
* p(a1​∣b2​)=p(b2​)p(a1​,b2​)​=0.530.08​≈0.1509
* p(a2​∣b2​)=p(b2​)p(a2​,b2​)​=0.530.45​≈0.8491

### b) Entropías

La fórmula para la entropía de una variable discreta X es: H(X)=−∑i​p(xi​)log2​p(xi​)

#### 1. Entropía del Emisor (Fuente) H(A)

* H(A)=−[p(a1​)log2​p(a1​)+p(a2​)log2​p(a2​)]
* H(A)=−[0.4log2​0.4+0.6log2​0.6]
* H(A)=−[0.4(−1.3219)+0.6(−0.7370)]=−[−0.5288−0.4422]=0.971 bits.

#### 2. Entropías Condicionales de la Fuente H(A∣bj​)

* **Para la salida b1​**:
  + H(A∣b1​)=−[p(a1​∣b1​)log2​p(a1​∣b1​)+p(a2​∣b1​)log2​p(a2​∣b1​)]
  + H(A∣b1​)=−[0.6809log2​0.6809+0.3191log2​0.3191]
  + H(A∣b1​)=−[0.6809(−0.5562)+0.3191(−1.6468)]=−[−0.3787−0.5252]=0.9039 bits.
* **Para la salida b2​**:
  + H(A∣b2​)=−[p(a1​∣b2​)log2​p(a1​∣b2​)+p(a2​∣b2​)log2​p(a2​∣b2​)]
  + H(A∣b2​)=−[0.1509log2​0.1509+0.8491log2​0.8491]
  + H(A∣b2​)=−[0.1509(−2.7291)+0.8491(−0.2372)]=−[−0.4116−0.2014]=0.613 bits.

**3.Entropía condicional total H(A∣B)**

H(A∣B)=p(b1​)H(A∣b1​)+p(b2​)H(A∣b2​)

H(A∣B)=0.47⋅0.903453555207+0.53⋅0.612196127491=0.749087118517 bits.

c) Calcula la información mutua y la capacidad del canal.

**Información Mutua**

I(A;B)=H(A)−H(A∣B)=0.970950594455−0.749087118517=0.221863475938 bits

**Capacidad del canal C=max⁡pI(A;B)**

Procedimiento (resumen):

* Tomamos p en [0,1].
* Para cada p calculamos p(b1)=p⋅0.8+(1−p)⋅0.25, las condicionales hacia atrás, H(A), H(A∣B) y I(p)=H(A)−H(A∣B)
* Buscamos el p que maximiza I(p).

Resultado numérico (búsqueda con grilla fina y comprobación):

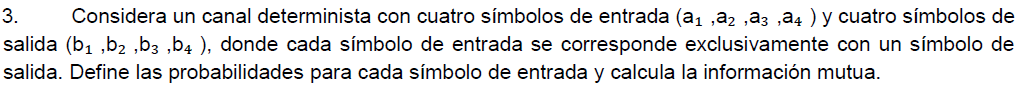
* p⋆≈0.5056739217392174,
* C=I(p⋆)≈0.23162096721381986 bits.

Para p⋆ muestro los valores intermedios (redondeados):

* p(a1)=0.5056739,
* p(a2)=0.4943261,
* p(b1)≈0.5281350,
* p(b2)≈0.4718650,
* H(A)≈0.999906251646 bits,
* H(A∣B)≈0.768285284432 bits,I
* (p⋆)=H(A)−H(A∣B)≈0.231620967214 bits.

**Resultado (capacidad del canal)**:

C≈0.2316209672 bits, logrado en p(a1)≈0.5056739.



Vamos a suponer el caso en el que los cuatro símbolos de entrada tienen la misma probabilidad, es decir, p1= ¼, p2=¼, p3=¼ y p4=¼ . En este caso, como cada símbolo de entrada se corresponde exclusivamente con un símbolo de salida, la probabilidad de los símbolos de salida es igual a la de los símbolos de entrada.

La fórmula de información mútua es I(A, B) = H(B) - H(B/A)

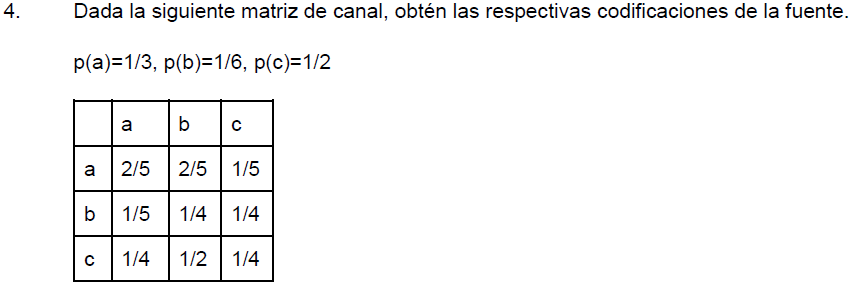
H(B) = = -(¼ \* log2(¼) + ¼ \* log2(¼) + ¼ \* log2(¼) + ¼ \* log2(¼)) = 2

H(B/A) =

Debido a que estamos considerando un canal determinista entonces p(bj/ai) = 1 con i=j, caso contrario será 0.

H(B/A) = (¼ \* 1 \* log2(1/1)) +(¼ \* 1 \* log2(1/1)) + (¼ \* 1 \* log2(1/1)) + (¼ \* 1 \* log2(1/1)) = 0

I(A, B) = H(B) - H(B/A) = 2 - 0 = 2 bits



Trabajaremos con la codificación de Huffman

1. Calculamos las probabilidades de salida

p(a) = p(a) \* p(a/a) + p(b) \* p(a/b) + p(c) \* p(a/c)

p(a) = ⅓ \* ⅖ + ⅙ \* ⅕ + ⅖ \* ¼ = 13/40

p(b) = p(a) \* p(b/a) + p(b) \* p(b/b) + p(c) \* p(b/c)

p(b) = ⅓ \* ⅖ + ⅙ \* ⅕ + ½ \* ½ = 5/12

p(c) = p(a) \* p(c/a) + p(b) \* p(c/b) + p(c) \* p(c/c)

p(c) = ⅓ \* ⅕ + ⅙ \*⅖ + ½ \* ¼ = 31/120

1. Procedemos con la codificación de Huffman

