Tecnología Digital IV: Redes de Computadoras

Clase 26: Introducción a la Teoría de la Información - Parte 2

Lucio Santi & Emmanuel Iarussi

Licenciatura en Tecnología Digital
Universidad Torcuato Di Tella

24 de junio de 2025

Información y entropía

Experimento aleatorio: tirar una moneda

Supongamos que modelamos el experimento de tirar una moneda cargada con una variable aleatoria

En 100 tiradas, la moneda cae 90 veces con la *cruz* hacia arriba y solo 10 con la *cara* hacia arriba

Experimento aleatorio: tirar una moneda

Supongamos que modelamos el experimento de tirar una moneda cargada con una variable aleatoria

En 100 tiradas, la moneda cae 90 veces con la *cruz* hacia arriba y solo 10 con la *cara* hacia arriba

$$x_z = sale\ cruz$$
$$x_c = sale\ cara$$

$$p(X) = \{p(x_z), p(x_c)\}$$
$$= \{0.9, 0.1\}$$ *moneda "cargada"*

Cuantificando la “sorpresa”

Cuando tiramos la moneda, esperamos que salga *cruz*, es decir que **“nos sorprende menos”** que si sale cara

Cuantificando la “sorpresa”

Cuando tiramos la moneda, esperamos que salga *cruz*, es decir que **“nos sorprende menos”** que si sale cara

Cuanto más improbable la salida de la variable aleatoria, más sorprendidos estamos al observar el evento

Cuantificando la “sorpresa”

Cuando tiramos la moneda, esperamos que salga *cruz*, es decir que **“nos sorprende menos”** que si sale cara

Cuanto más improbable la salida de la variable aleatoria, más sorprendidos estamos al observar el evento

¿Cómo podríamos cuantificar la cantidad de sorpresa?

Cuantificando la “sorpresa”

Una forma de expresar la *sorpresa* es cuantificándola en base a la probabilidad de la salida de la variable aleatoria:

$$sorpresa = \frac{1}{p(x)}$$

Cuantificando la “sorpresa”

Una forma de expresar la *sorpresa* es cuantificándola en base a la probabilidad de la salida de la variable aleatoria:

$$sorpresa = \frac{1}{p(x)}$$

$p(x) \approx 1$ *sorpresa baja*

$p(x) \approx 0$ *sorpresa alta*

Información de Shannon

Una forma de expresar la *sorpresa* es cuantificándola en base a la probabilidad de la salida de la variable aleatoria:

$$h(x) = \log_2 \frac{1}{p(x)}$$

Información de Shannon

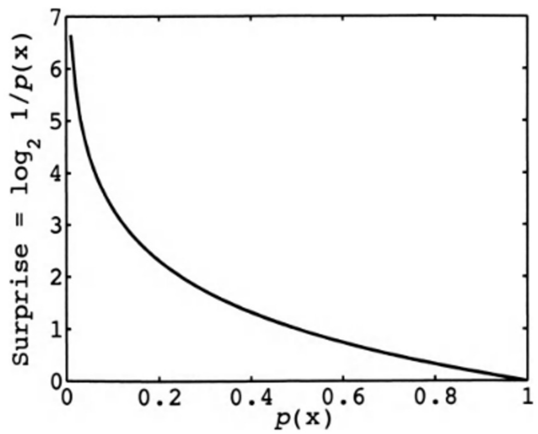
Una forma de expresar la *sorpresa* es cuantificándola en base a la probabilidad de la salida de la variable aleatoria:

$$
\begin{aligned}
h(x) &= \log_2 \frac{1}{p(x)} \\
&= -\log_2 p(x)
\end{aligned}
$$

Información de Shannon

Una forma de expresar la *sorpresa* es cuantificándola en base a la probabilidad de la salida de la variable aleatoria:

$$h(x) = \log_2 \frac{1}{p(x)}$$
$$= -\log_2 p(x)$$



Intuición: Es más informativo saber que ocurrió un evento improbable que conocer la ocurrencia de uno probable.

Información y sorpresa

Es más informativo saber que ocurrió un evento improbable que conocer la ocurrencia de uno probable:

Información y sorpresa

Es más informativo saber que ocurrió un evento improbable que conocer la ocurrencia de uno probable:

Información y sorpresa

Es más informativo saber que ocurrió un evento improbable que conocer la ocurrencia de uno probable:

Información y sorpresa

Podemos cuantificar la sorpresa utilizando la noción de información de Shannon

Necesitamos conocer las probabilidades de los eventos:

$$x_b = bolita \quad p(x_b) = 0,33$$

$$x_e = vacio \quad p(x_e) = 0,67$$

Información y sorpresa

Podemos cuantificar la sorpresa utilizando la noción de información de Shannon

Necesitamos conocer las probabilidades de los eventos:

$$x_b = bolita \quad p(x_b) = 0,33$$

$$x_e = vacio \quad p(x_e) = 0,67$$

Información de Shannon:

$$h(x_b) = \log_2 \frac{1}{p(x_b)} = \log_2 \frac{1}{0.33} = 1.6$$

$$h(x_e) = \log_2 \frac{1}{p(x_e)} = \log_2 \frac{1}{0.67} = 0.58$$

Información y sorpresa

Dado que estamos cuantificando la información, podemos medirla en *bits* (unidad de información)

También se conoce a esta unidad como *shannon (Sh)*

$$x_b = bolita \quad p(x_b) = 0,33$$

$$x_e = vacio \quad p(x_e) = 0,67$$

Información de Shannon:

$$h(x_b) = \log_2 \frac{1}{p(x_b)} = \log_2 \frac{1}{0.33} = 1.6$$

$$h(x_e) = \log_2 \frac{1}{p(x_e)} = \log_2 \frac{1}{0.67} = 0.58$$

Sorpresa promedio de una variable

No estamos interesados en un solo evento o valor de una variable aleatoria

En general, nos va a interesar saber cuánta *sorpresa* hay en promedio para todo el conjunto de valores de una variable aleatoria:

$$H(X) \sim \frac{1}{n}\sum_{i=1}^{n} \log_2 \frac{1}{p(x_i)}$$

Sorpresa promedio de una variable

Moneda justa:

$$H(X) \sim \frac{\sum_{j=1}^{48} \log_2 \left(1/p\left(x_z\right)\right) + \sum_{j=1}^{52} \log_2 \left(1/p\left(x_c\right)\right)}{100}$$

Sorpresa promedio de una variable

Moneda justa:

$$H(X) \sim \frac{\sum_{j=1}^{48} \log_2 \left(1/p\left(x_z\right)\right) + \sum_{j=1}^{52} \log_2 \left(1/p\left(x_c\right)\right)}{100}$$

$$= \frac{48 \times \log_2 (1/0.5) + 52 \times \log_2 (1/0.5)}{100} = 1$$ *bit por tirada*

Sorpresa promedio de una variable

Moneda justa:

$$H(X) \sim \frac{\sum_{j=1}^{48} \log_2 \left(1/p\left(x_z\right)\right) + \sum_{j=1}^{52} \log_2 \left(1/p\left(x_c\right)\right)}{100}$$

$$= \frac{48 \times \log_2(1/0.5) + 52 \times \log_2(1/0.5)}{100} = 1$$ *bit por tirada*

Moneda cargada:

$$H(X) \sim \frac{\sum_{j=1}^{87} \log_2 \left(1/p\left(x_z\right)\right) + \sum_{j=1}^{13} \log_2 \left(1/p\left(x_c\right)\right)}{100}$$

$$= \frac{87 \times \log_2(1/0.9) + 13 \times \log_2(1/0.1)}{100} = 0.564$$ *bits por tirada*

Entropía de una variable aleatoria

La entropía es una medida de la incerteza: sabemos menos de la moneda justa que de la sesgada.

Entropía de una variable aleatoria

La entropía es una medida de la incerteza: sabemos menos de la moneda justa que de la sesgada

$$H(X) \sim \frac{87 \times \log_2(1/0.9) + 13 \times \log_2(1/0.1)}{100} = 0.564$$

Una entropía de 0.564 bits quiere decir que podríamos representar la información de, por ejemplo, 1000 tiradas usando tan solo 564 dígitos binarios (en vez de 1000!)

Entropía de una variable aleatoria

Calculamos la entropía como:

$$p(X) = \{p(x_1), \ldots, p(x_m)\}$$ *Distribución de probabilidades de la variable aleatoria X*

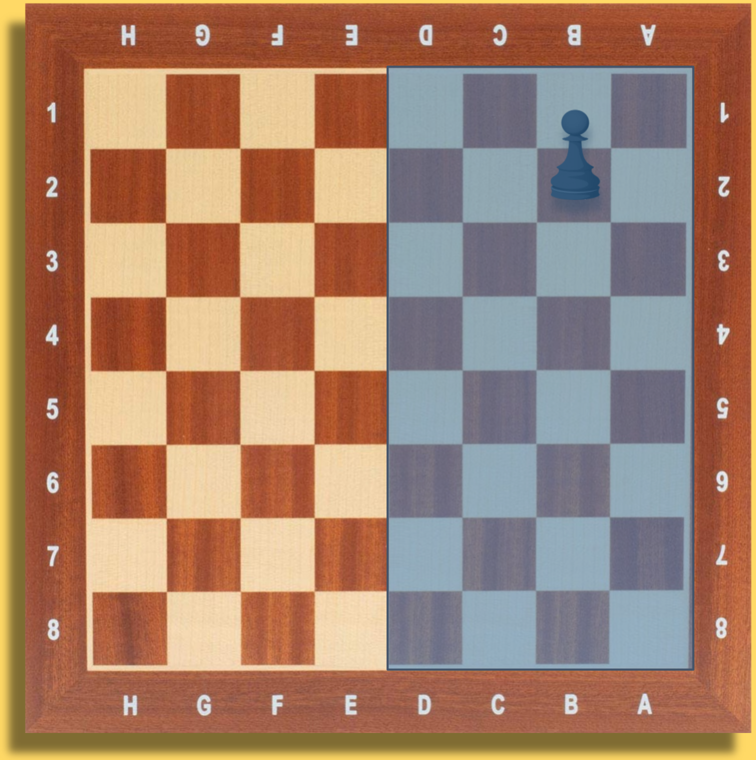
$$H(X) = \sum_{i=1}^{m} p(x_i) \log_2 \frac{1}{p(x_i)} = -\sum_{i=1}^{m} p(x_i) \log_2 p(x_i)$$ *bits*

Ejercicio!

¿Cuántas preguntas con respuesta **sí/no** tengo que hacer para adivinar la posición del peón en un tablero de ajedrez?

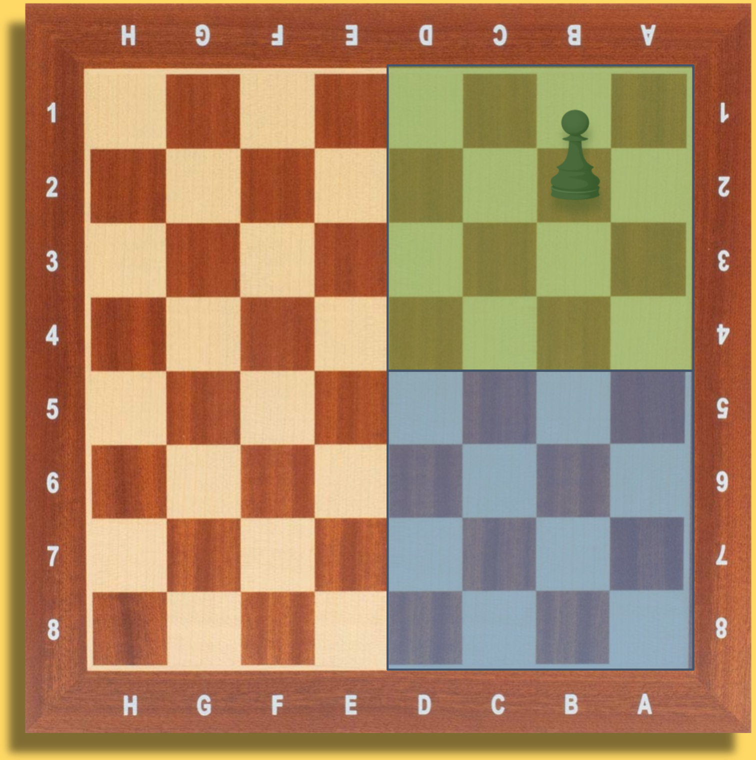
Pensar una **estrategia que minimice** la cantidad de preguntas.

Ejercicio!



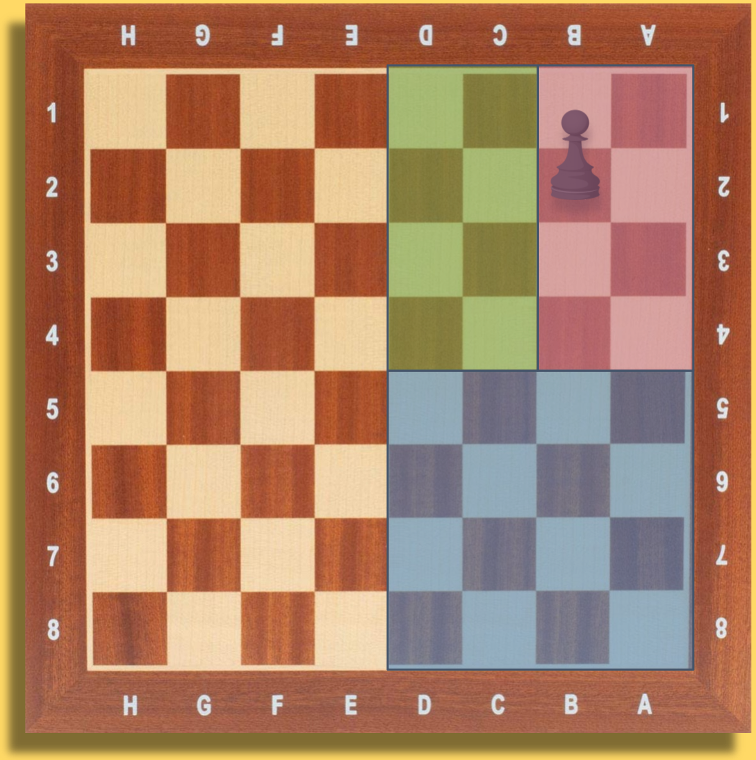
1. ¿Está en la mitad izquierda?

Ejercicio!



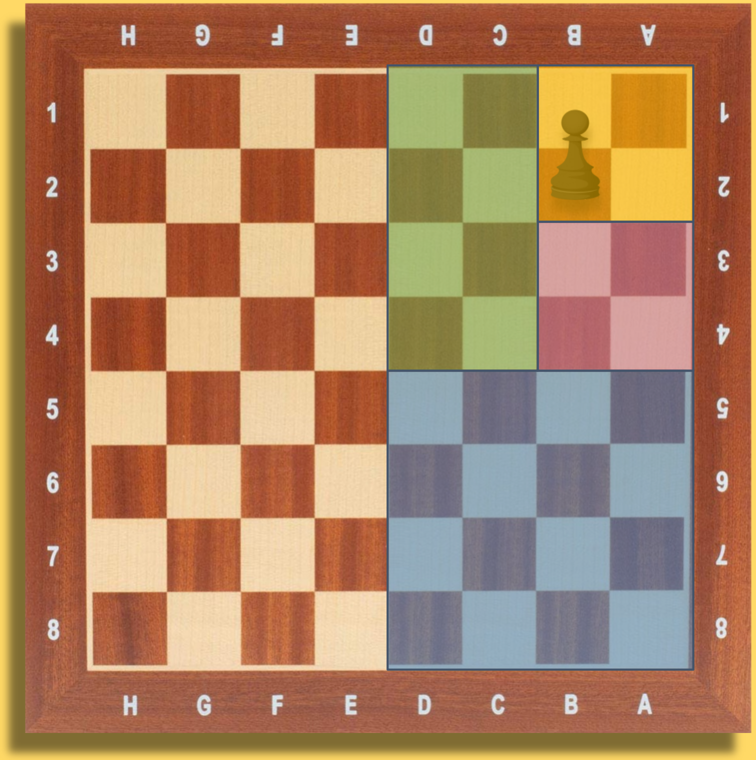
1. ¿Está en la mitad izquierda?
2. ¿Está en la mitad superior?

Ejercicio!



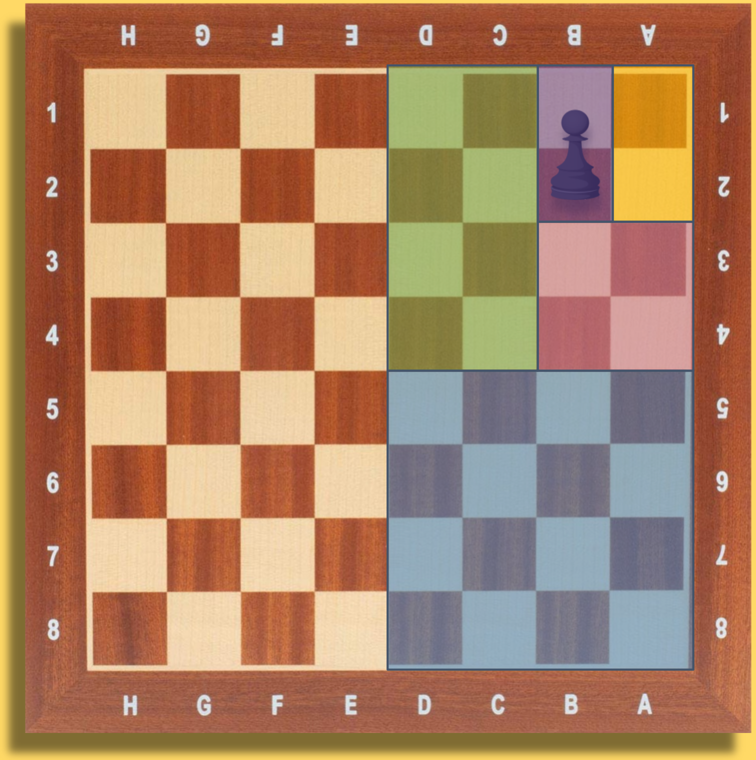
1. ¿Está en la mitad izquierda?
2. ¿Está en la mitad superior?
3. En ese cuadrante ¿Está en la mitad izquierda?

Ejercicio!



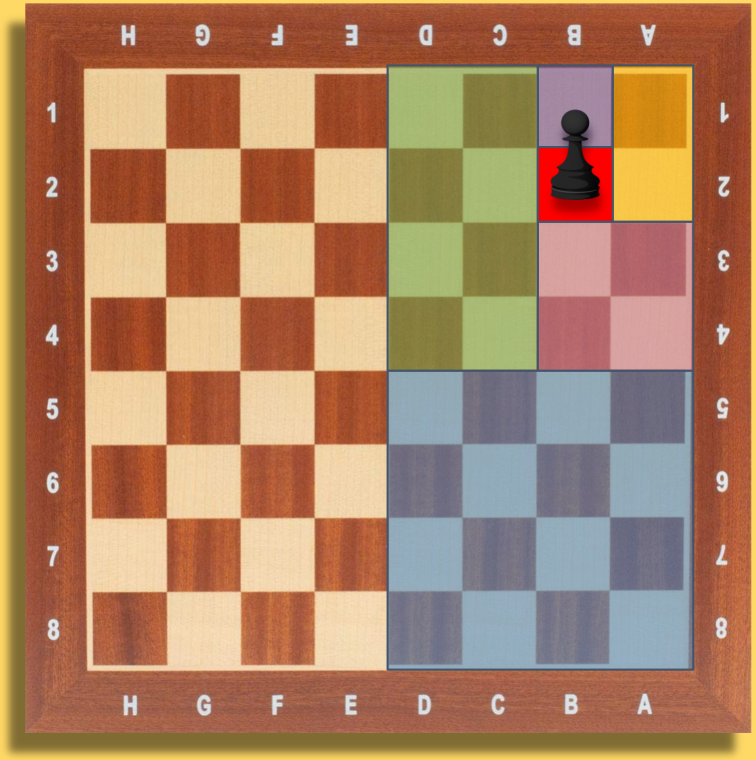
1. ¿Está en la mitad izquierda?
2. ¿Está en la mitad superior?
3. En ese cuadrante ¿Está en la mitad izquierda?
4. En ese cuadrante ¿Está en la mitad superior?

Ejercicio!



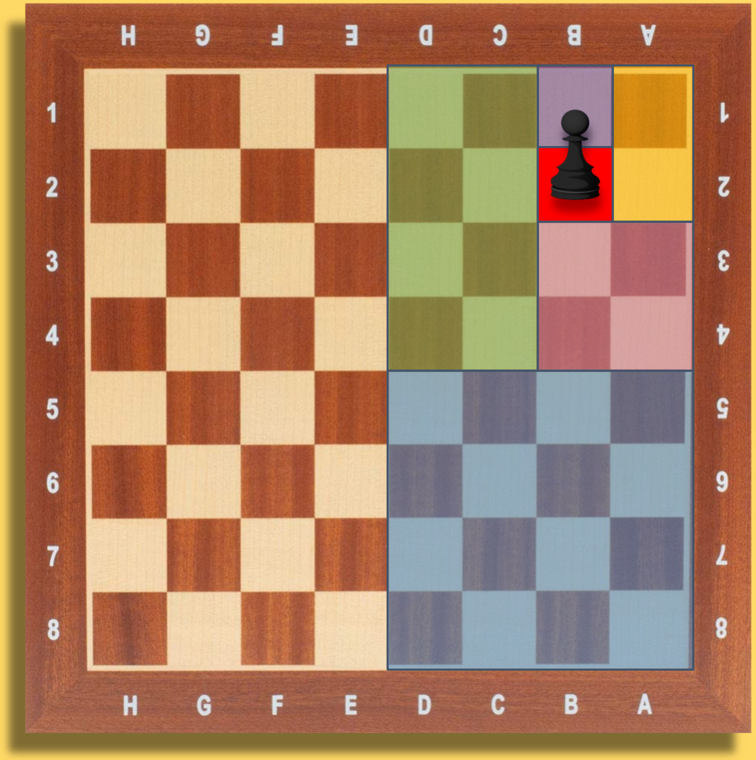
1. ¿Está en la mitad izquierda?
2. ¿Está en la mitad superior?
3. En ese cuadrante ¿Está en la mitad izquierda?
4. En ese cuadrante ¿Está en la mitad superior?
5. En ese cuadrante ¿Está en la mitad izquierda?

Ejercicio!



1. ¿Está en la mitad izquierda?
2. ¿Está en la mitad superior?
3. En ese cuadrante ¿Está en la mitad izquierda?
4. En ese cuadrante ¿Está en la mitad superior?
5. En ese cuadrante ¿Está en la mitad izquierda?
6. En ese cuadrante ¿Está en la mitad superior?

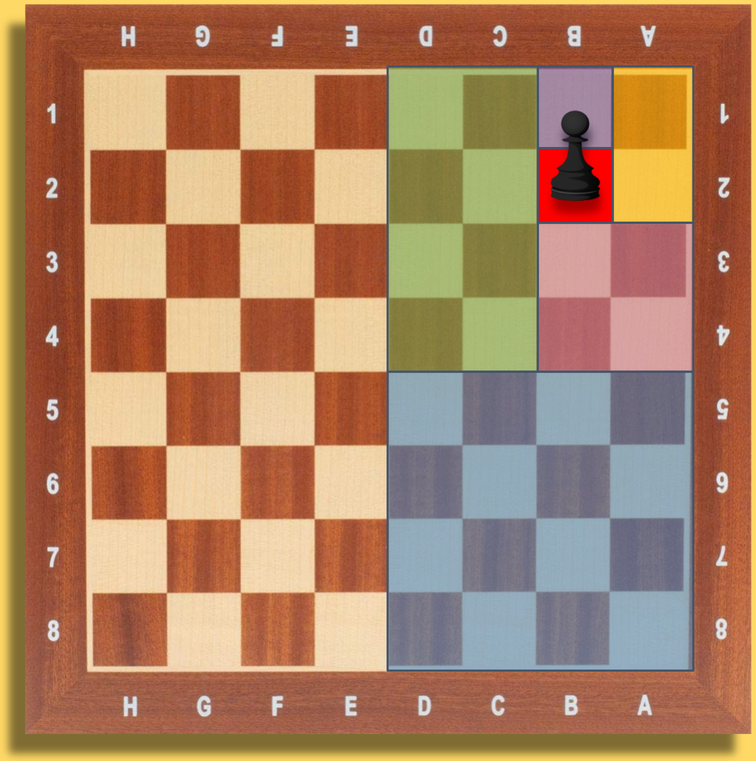
Ejercicio!



Podemos modelar este problema mediante una variable aleatoria:

$$A_c = \{A1, A2, \dots, H8\}$$

Ejercicio!

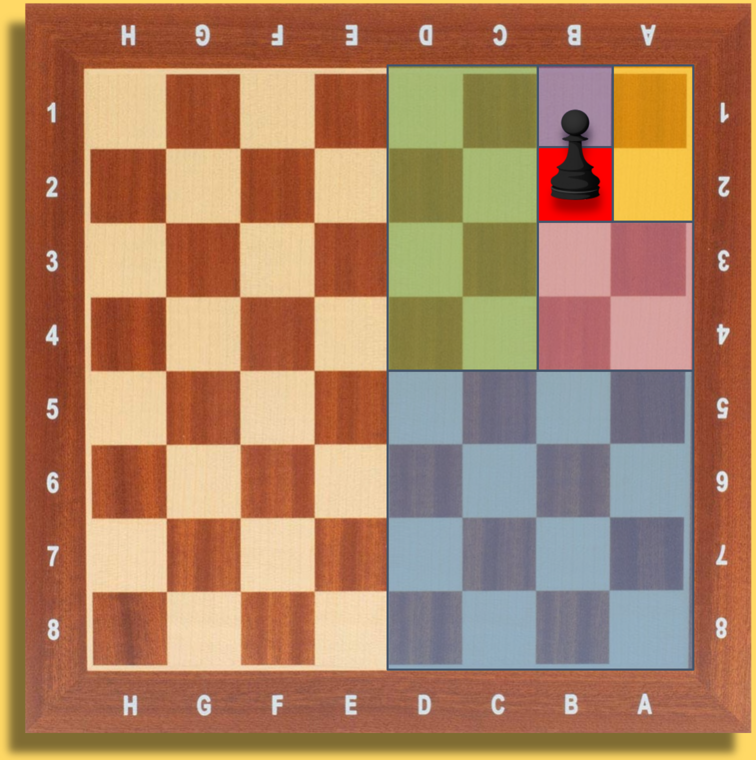


Podemos modelar este problema mediante una variable aleatoria:

$$A_c = \{A1, A2, \ldots, H8\}$$

$$p(C) = \{1/64, 1/64, \ldots, 1/64\}$$

Ejercicio!



Podemos modelar este problema mediante una variable aleatoria:

$$A_c = \{A1, A2, \ldots, H8\}$$

$$p(C) = \{1/64, 1/64, \ldots, 1/64\}$$

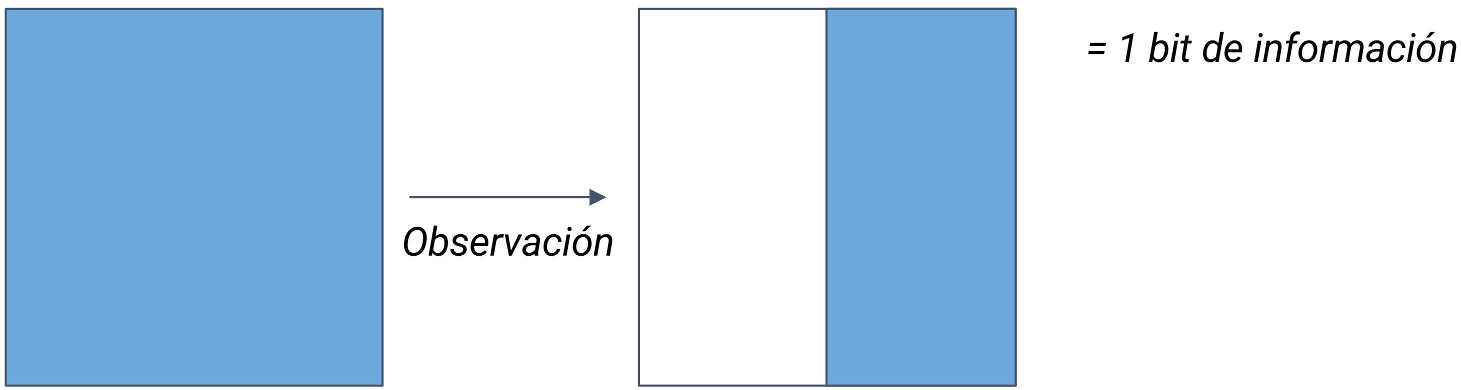
$$H(C) = \sum_{i=1}^{64} \frac{1}{64} \log_2 \frac{1}{1/64} = 6$$

Interpretando la Entropía

Podemos pensar a la entropía como la cantidad mínima de preguntas binarias (si trabajamos con log en base 2) para identificar el valor de una variable aleatoria discreta

Interpretando la Entropía

Podemos pensar a la entropía como la cantidad mínima de preguntas binarias (si trabajamos con log en base 2) para identificar el valor de una variable aleatoria discreta.

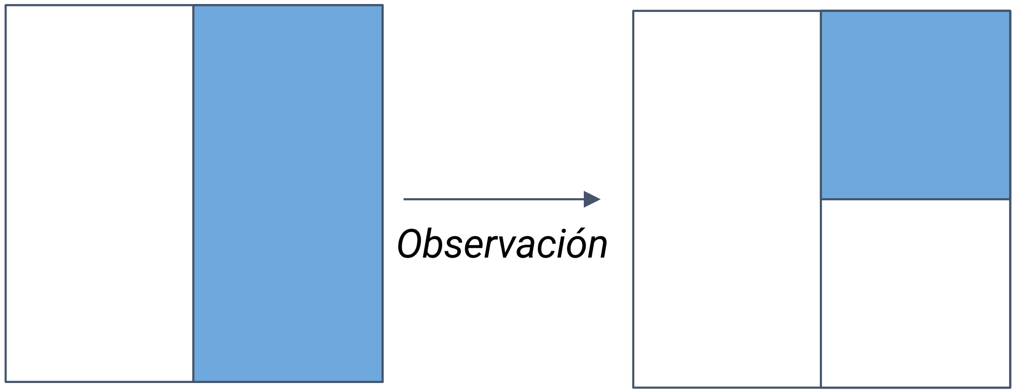


Espacio de posibilidades

$$p = \frac{1}{2}$$

Interpretando la Entropía

Podemos pensar a la entropía como la cantidad mínima de preguntas binarias (si trabajamos con log en base 2) para identificar el valor de una variable aleatoria discreta.



= 1 bit de información

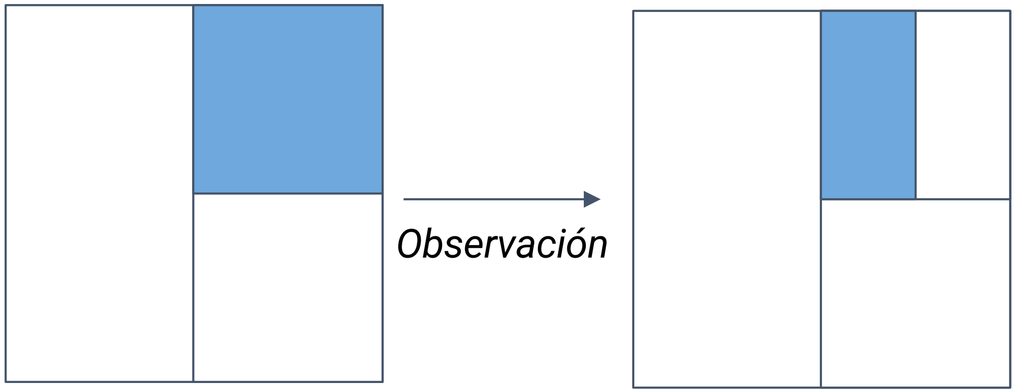
+ 1 bit de información

Espacio de posibilidades

$$p = \frac{1}{2}$$

Interpretando la Entropía

Podemos pensar a la entropía como la cantidad mínima de preguntas binarias (si trabajamos con log en base 2) para identificar el valor de una variable aleatoria discreta.



Espacio de posibilidades

$$p = \frac{1}{2}$$

= 1 bit de información

+ 1 bit de información

+ 1 bit de información

…

Interpretando la Entropía

¿Cuánta información contiene la posición del peón en un tablero de ajedrez?

Interpretando la Entropía

¿Cuánta información contiene la posición del peón en un tablero de ajedrez? → *6 bits*

Interpretando la Entropía

¿Cuánta información contiene la posición del peón en un tablero de ajedrez? → *6 bits*

El valor de entropía depende **solo** de la distribución de probabilidades, no de los símbolos de la variable aleatoria

Si duplicamos la cantidad de casillas en el tablero de ajedrez, o en las caras de un dado, la entropía se incrementa en 1 bit

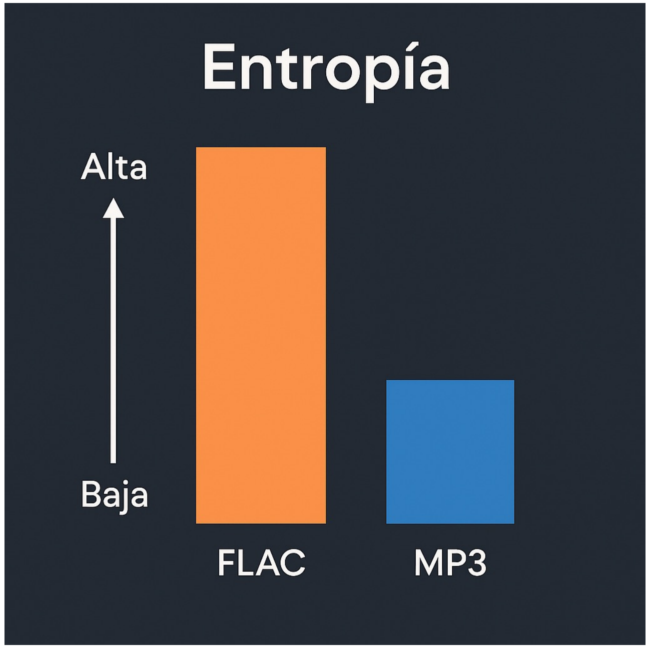
Interpretando la Entropía

- **Noticias y titulares**
- **Conversación**
- **Juegos de Azar**
- **Lotería**
- **Pronóstico del tiempo**

Interpretando la Entropía

Interpretando la Entropía

¿Qué tiene mayor entropía un archivo de música en .flac o en .mp3



Bits, Shannons y Bans

La unidad de medición de información depende de la base del logaritmo que estemos utilizando

- Es similar a lo que sucede cuando medimos agua en una receta y usamos *litros* o *tazas*

Bits, Shannons y Bans

La unidad de medición de información depende de la base del logaritmo que estemos utilizando

- Es similar a lo que sucede cuando medimos agua en una receta y usamos *litros* o *tazas*.

Si medimos información usando logaritmo natural (base *e* = 2.72), entonces las unidades se llaman *nats* en vez de *bits* (base 2)

Si usamos logaritmos en base 10, las unidades se llaman *bans*

Si usamos logaritmos en base 2, las unidades son los *bits* o *Shannons* (*Sh*)

Primer teorema de Shannon (1948)

Si deseamos comunicar muestras extraídas de alguna distribución, en promedio necesitaremos al menos tantos símbolos como la entropía H de esa distribución para comunicar sin ambigüedades esas muestras

Primer teorema de Shannon (1948)

Si deseamos comunicar muestras extraídas de alguna distribución, en promedio necesitaremos al menos tantos símbolos como la entropía *H* de esa distribución para comunicar sin ambigüedades esas muestras

- *Source-coding theorem*

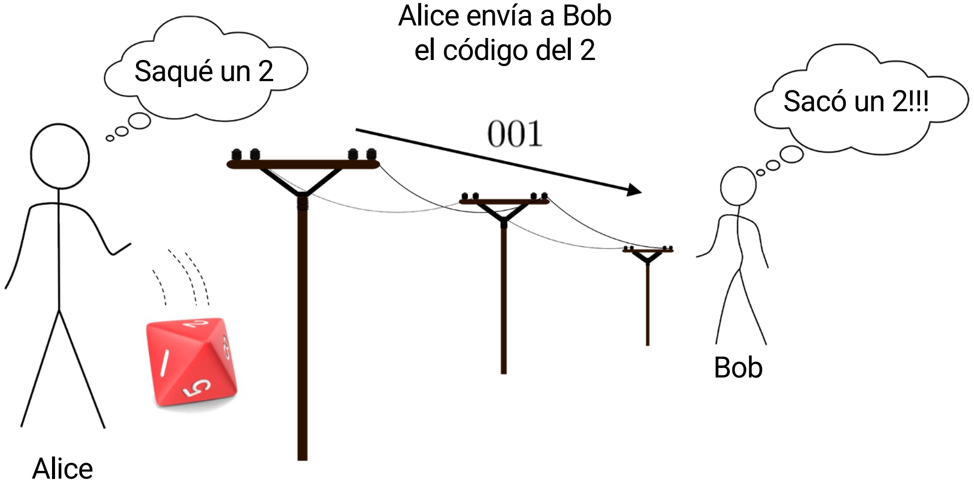
La entropía proporciona un límite inferior sobre cuánto podemos comprimir nuestra descripción de las muestras de la distribución antes de que inevitablemente perdamos información

$$L(S) \geq H(X)$$

Eficiencia en la codificación de datos

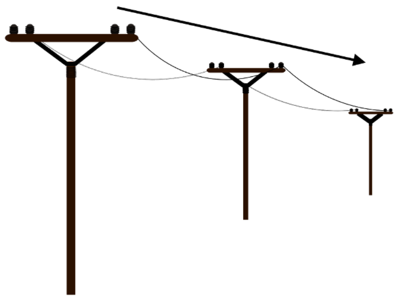
Dado de 8 caras:

Símbolo	Código
$s_1 = 1$	$x_1 = 000$
$s_2 = 2$	$x_2 = 001$
$s_3 = 3$	$x_3 = 010$
$s_4 = 4$	$x_4 = 011$
$s_5 = 5$	$x_5 = 100$
$s_6 = 6$	$x_6 = 101$
$s_7 = 7$	$x_7 = 110$
$s_8 = 8$	$x_8 = 111$



Fuente: dado de 8 caras

Símbolo	Código
$s_1 = 1$	$x_1 = 000$
$s_2 = 2$	$x_2 = 001$
$s_3 = 3$	$x_3 = 010$
$s_4 = 4$	$x_4 = 011$
$s_5 = 5$	$x_5 = 100$
$s_6 = 6$	$x_6 = 101$
$s_7 = 7$	$x_7 = 110$
$s_8 = 8$	$x_8 = 111$



Entropía (asumiendo equiprobabilidad)

$H = 3$ bits/símbolo

Fuente: dado de 8 caras

Símbolo	Código
$s_1 = 1$	$x_1 = 000$
$s_2 = 2$	$x_2 = 001$
$s_3 = 3$	$x_3 = 010$
$s_4 = 4$	$x_4 = 011$
$s_5 = 5$	$x_5 = 100$
$s_6 = 6$	$x_6 = 101$
$s_7 = 7$	$x_7 = 110$
$s_8 = 8$	$x_8 = 111$

La longitud media del código es

L = 3 dígitos binarios/símbolo

Según el Primer Teorema de Shannon, se trata de un **código eficiente** (*L* = *H*(*X*))

Entropía (asumiendo equiprobabilidad)

$H = 3$ bits/símbolo

Eficiencia en la codificación de datos

Eficiencia de un código: cociente entre la cantidad promedio *L(S)* de dígitos binarios por símbolo en el código *S* y la entropía *H(X)* de la fuente de información:

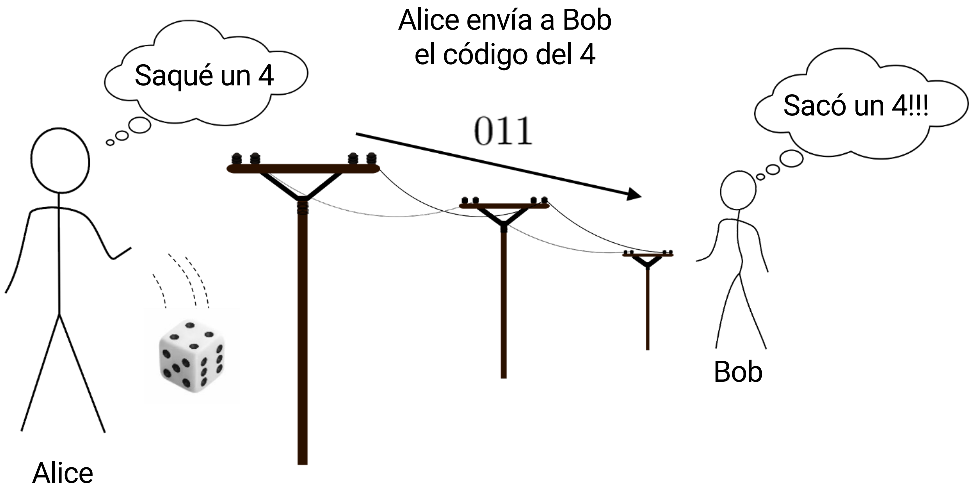
$$\frac{H(X)}{L(S)}$$

Para el ejemplo anterior, este cociente es de **1 bit por dígito binario**

Otra fuente: dado de 6 caras

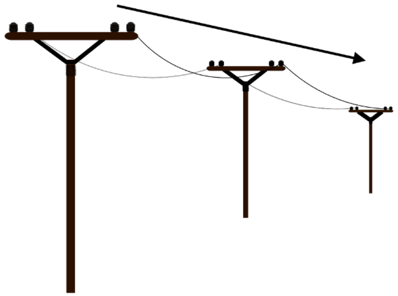
Dado de 6 caras:

Símbolo	Código
$s_1 = 1$	$x_1 = 000$
$s_2 = 2$	$x_2 = 001$
$s_3 = 3$	$x_3 = 010$
$s_4 = 4$	$x_4 = 011$
$s_5 = 5$	$x_5 = 100$
$s_6 = 6$	$x_6 = 101$



Otra fuente: dado de 6 caras

Símbolo	Código
$s_1 = 1$	$x_1 = 000$
$s_2 = 2$	$x_2 = 001$
$s_3 = 3$	$x_3 = 010$
$s_4 = 4$	$x_4 = 011$
$s_5 = 5$	$x_5 = 100$
$s_6 = 6$	$x_6 = 101$

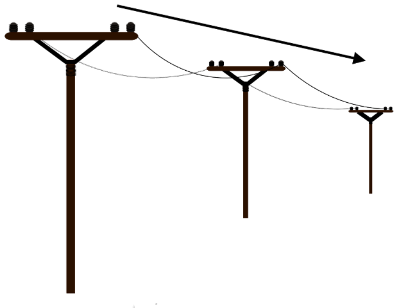


Entropía (asumiendo equiprobabilidad)

$H = 2.58$ bits/símbolo

Otra fuente: dado de 6 caras

Símbolo	Código
$s_1 = 1$	$x_1 = 000$
$s_2 = 2$	$x_2 = 001$
$s_3 = 3$	$x_3 = 010$
$s_4 = 4$	$x_4 = 011$
$s_5 = 5$	$x_5 = 100$
$s_6 = 6$	$x_6 = 101$



La longitud media del código es

L = 3 dígitos binarios/símbolo

En este caso, la eficiencia del código es de:

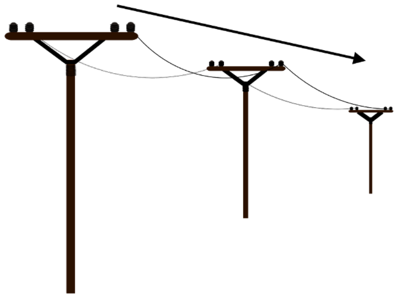
$$\frac{2.58}{3} = 0.86 \text{ bits/dígito binario}$$

Entropía (asumiendo equiprobabilidad)

$$H = 2.58 \text{ bits/símbolo}$$

Otra fuente: dado de 6 caras

Símbolo	Código
$s_1 = 1$	$x_1 = 000$
$s_2 = 2$	$x_2 = 001$
$s_3 = 3$	$x_3 = 010$
$s_4 = 4$	$x_4 = 011$
$s_5 = 5$	$x_5 = 100$
$s_6 = 6$	$x_6 = 101$



La longitud media del código es

L = 3 dígitos binarios/símbolo

En este caso, la eficiencia del código es de

$$\frac{2.58}{3} = 0.86 \text{ bits/dígito binario}$$

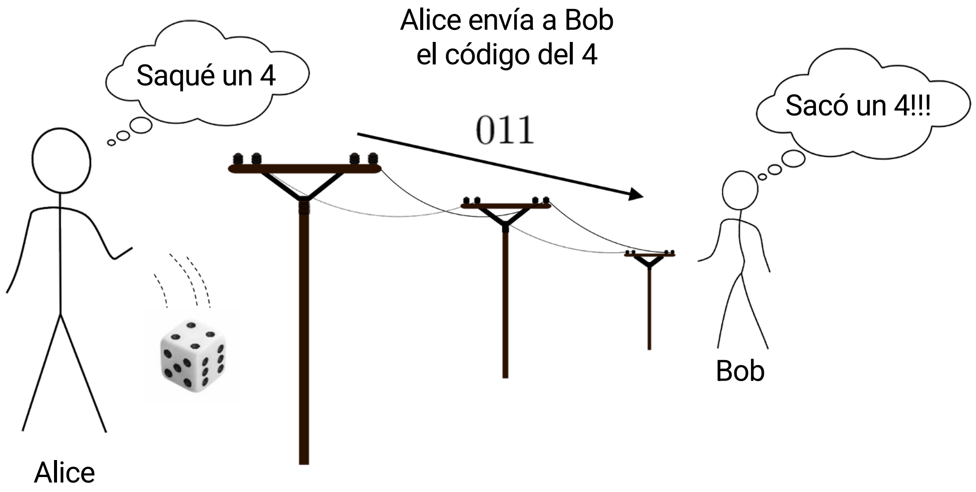
Entropía (asumiendo equiprobabilidad)

$$H = 2.58 \text{ bits/símbolo}$$

Qué salió mal?

Eficiencia en la codificación de datos

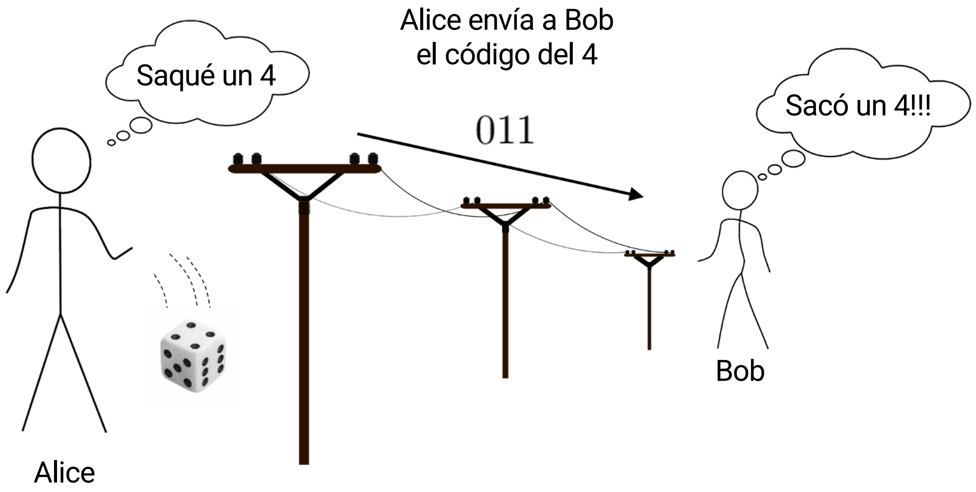
No siempre que transmitimos un dígito binario estamos transmitiendo un bit de información!!



Eficiencia en la codificación de datos

No siempre que transmitimos un dígito binario estamos transmitiendo un bit de información

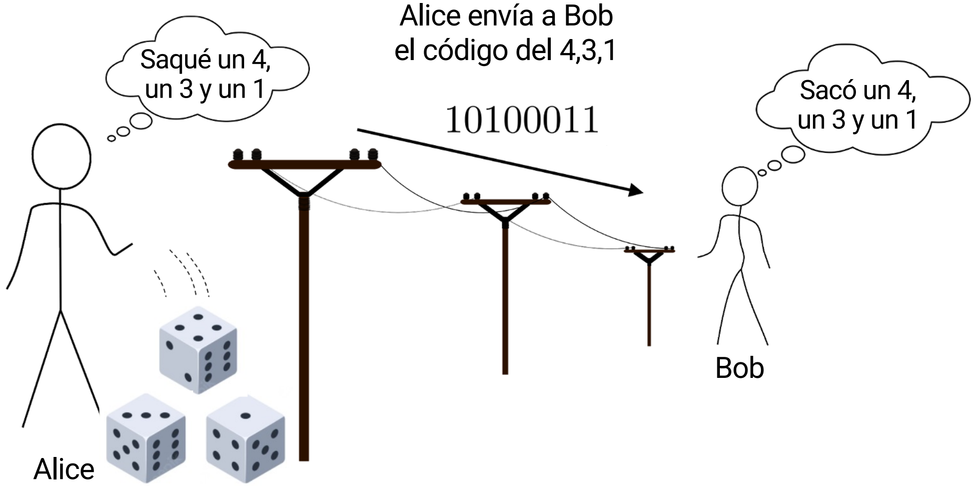
Se maximiza la eficiencia cuando cada dígito binario transporta la mayor cantidad de información posible



Recodificación de la fuente

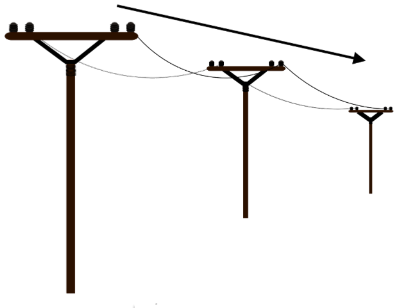
Idea! Codificar más de un símbolo a la vez. Por ejemplo, 3 símbolos seguidos:

Símbolo	Código
$s_1 = 1, 1, 1$	$x_1 = 00000000$
$s_2 = 1, 1, 2$	$x_2 = 00000001$
$s_3 = 1, 1, 3$	$x_3 = 00000010$
$s_4 = 1, 1, 4$	$x_4 = 00000011$
...	...
$s_{163} = 4, 3, 1$	$x_4 = 10100011$
...	...
$s_{216} = 6, 6, 6$	$x_{216} = 11011000$



Recodificación de la fuente

Símbolo	Código
$s_1 = 1, 1, 1$	$x_1 = 00000000$
$s_2 = 1, 1, 2$	$x_2 = 00000001$
$s_3 = 1, 1, 3$	$x_3 = 00000010$
$s_4 = 1, 1, 4$	$x_4 = 00000011$
...	...
$s_{163} = 4, 3, 1$	$x_4 = 10100011$
...	...
$s_{216} = 6, 6, 6$	$x_{216} = 11011000$

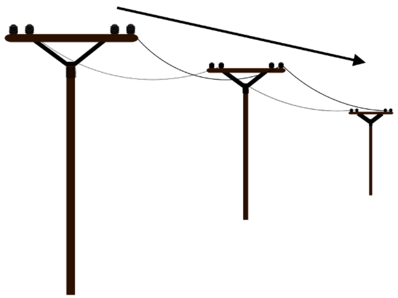


Eficiencia:

$$\frac{2.58}{3} = 0.86 \text{ bits/dígito binario}$$

Recodificación de la fuente

Símbolo	Código
$s_1 = 1,1,1$	$x_1 = 00000000$
$s_2 = 1,1,2$	$x_2 = 00000001$
$s_3 = 1,1,3$	$x_3 = 00000010$
$s_4 = 1,1,4$	$x_4 = 00000011$
...	...
$s_{163} = 4,3,1$	$x_4 = 10100011$
...	...
$s_{216} = 6,6,6$	$x_{216} = 11011000$



Si los valores de la variable no representan un número entero de bits, se puede mejorar la codificación combinando varios símbolos en el código

Eficiencia:

$$\frac{2.58}{\cancel{3}\; 8/3} = \cancel{0.86}\; 0.970$$ bits/dígito binario