# 83GIIN - Compresión y Recuperación de Información Multimedia

# Actividad 1 - Portafolio

Gagliardo Miguel Angel

09 de Marzo de 2025

## Introducción

En el contexto de la teoría de la información (TI) es importante entender como se distribuyen los datos en los archivos y sobre todo como esta distribución influye en la capacidad de comprimir los mismos.

En esta actividad, analizaremos 3 tipos diferentes de archivo: **Texto plano**, **FASTA** (que normalmente contienen secuencias de proteínas) e **imagen en formato RAW**, calcularemos la diversidad de símbolos que estos contienen, su frecuencia de aparición y finalmente la entropía de cada archivo, para finalmente demostrar de manera teórica que no todos los datos contenidos en ellos se comportan de la misma manera: algunos archivos contienen información con alta redundancia, mientras que otros presentan una distribución más uniforme de símbolos, y como esto tiene un impacto directo en la tasa de compresión **sin pérdida** que se puede alcanzar y en la cantidad mínima de bits necesarios para representar dicha información.

Para el **análisis de los archivos** se ha utilizado **python3** con las siguientes librerías:

- **numpy: 2.0.2**
- **pandas: 2.2.3**
- **matplotlib: 3.9.4**

Para el cálculo de símbolos, se han utilizado 2 estrategias:

- **En archivo de texto/FASTA:** Se lee el archivo linea por línea y caracter por caracter, si el caracter está contenido en un diccionario (dict type) se le suma un uno al contador de dicho caracter, caso contrario se agrega el caracter al diccionario y un 1 (dado que es su única aparición hasta el momento)
- **En archivo de imagen tipo RAW:** De manera similar al archivo de texto, en vez de leer

caracter por caracter, abrimos el archivo y leemos pixel por pixel, sabiendo que de antemano tiene 64 x 64 pixeles (ergo, 4096 en total) e incluyendolos en un diccionario de python.

Utilizando estos diccionarios tanto para los archivos de texto/FASTA/imagen en tipo RAW podemos mostrar los resultados en una tabla (para ver cuantas apariciones tiene cada símbolo), graficarlos utilizando matplotlib que tiene una función “built-in” para mostrar histogramas con sus frecuencias (ver siguiente apartado) o incluso calcular su entropía utilizando la fórmula de Shannon:

A la hora de teorizar sobre las potenciales tasas de compresión, utilizaremos la fórmula:

$$Tasa\ de\ compresión = \frac{H(s)}{L}$$

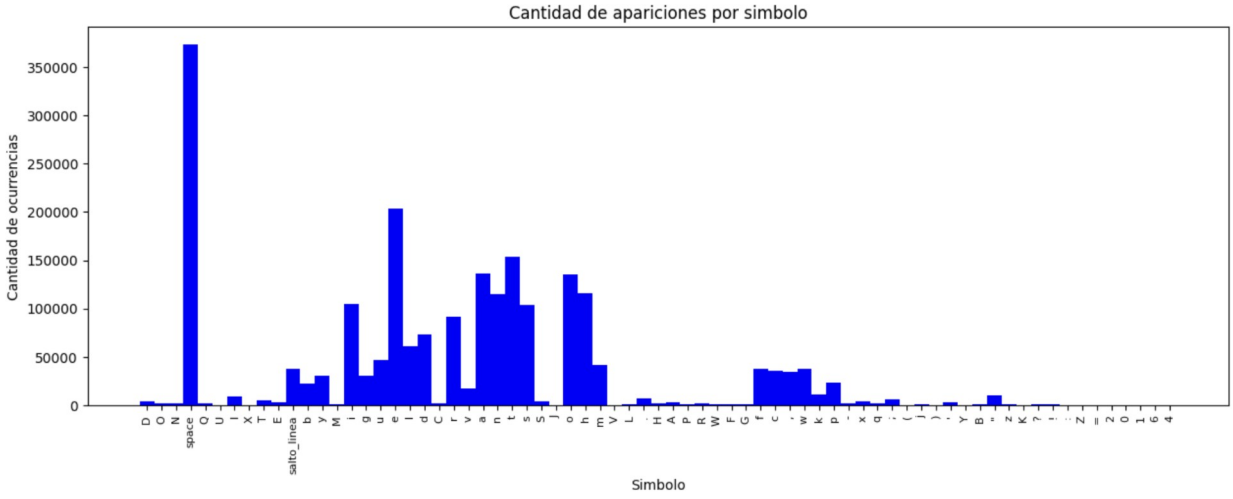
Donde:

- **H(S)** es la entropía del texto (ya calculada)
- L es la cantidad de bits por símbolo.

## Resultados por Archivo

**A continuación, se presentará cada histograma por archivo y su análisis de tasa de compresión, para finalmente concluir con un razonamiento y aprendizajes.**

### 1. don-quixote.txt




Como vemos, exceptuando el outlier que representa el símbolo “space” (espacio) con ~350mil apariciones, tenemos una distribución bastante dispersa, y su entropía es **4.4286 bits/símbolo** en un texto que tiene **sólo 70 caracteres diferentes**.

Dado que la entropía del texto es **4.4286 bits/símbolo**, quiere decir que en promedio cada carácter del mismo podría representarse con ~4.42 bits en lugar de **8 bits que necesita un caracter en ASCII**.

**Nota:** Asumimos ASCII dado que es el código de caracteres (junto con UTF-8) mayormente utilizado para alfabeto latino y archivos de texto plano. Por tanto:

$$Tasa\ de\ compresión = \frac{H(s)}{L} = \frac{4.42\ bits/simbolo}{8\ bits/simbolo} = 0.5525$$

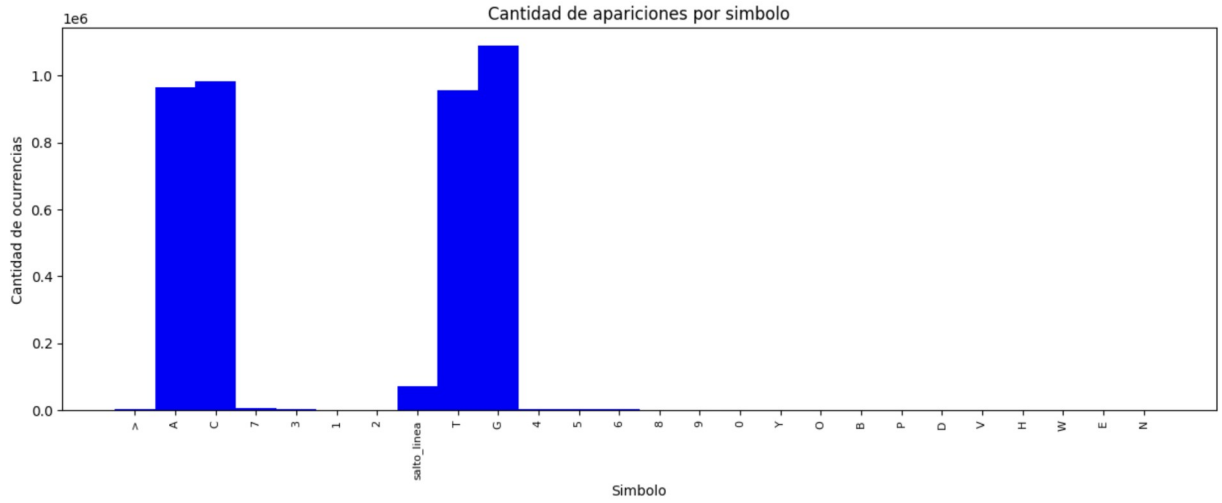
Con lo cual en un esquema de compresión óptimo **con el 55.25% de la información original podríamos representar el archivo sin pérdidas**.

Ahora bien, en realidad tenemos un texto con **70 caracteres, que se pueden presentar con en 7 bits, o sea $2^7$**, por lo cual nuestro cálculo más realista de una teórica tasa de compresión es de:

$$Tasa\ de\ compresión = \frac{H(s)}{L} = \frac{4.42\ bits/simbolo}{7\ bits/simbolo} = 0.6314$$

Con lo cual en un esquema de compresión óptimo **con el 63.14% de la información original podríamos representar el archivo sin pérdidas**.

## 2. ecoli.fa




Como vemos en este caso hay una distribución más uniforme dado que sólo hay cuatro grupos

de proteinas ACTG (Adenina, Citosina, Timina y Guanina).

En este caso la entropía del mismo es de **2.1545 bits/símbolo**, y dado que desconozco la cantidad diversa de símbolos que pueden aparecer, asumo ASCII o bien 8 bits para representar todos los símbolos como he hecho anteriormente, por tanto:

$$Tasa\,de\,compresión = \frac{H(s)}{L} = \frac{2.15\,bits/simbolo}{8\,bits/simbolo} = 0.26875$$

Con lo cual en un esquema de compresión óptimo **con el 26.8% de la información original podríamos representar el archivo sin pérdidas**.

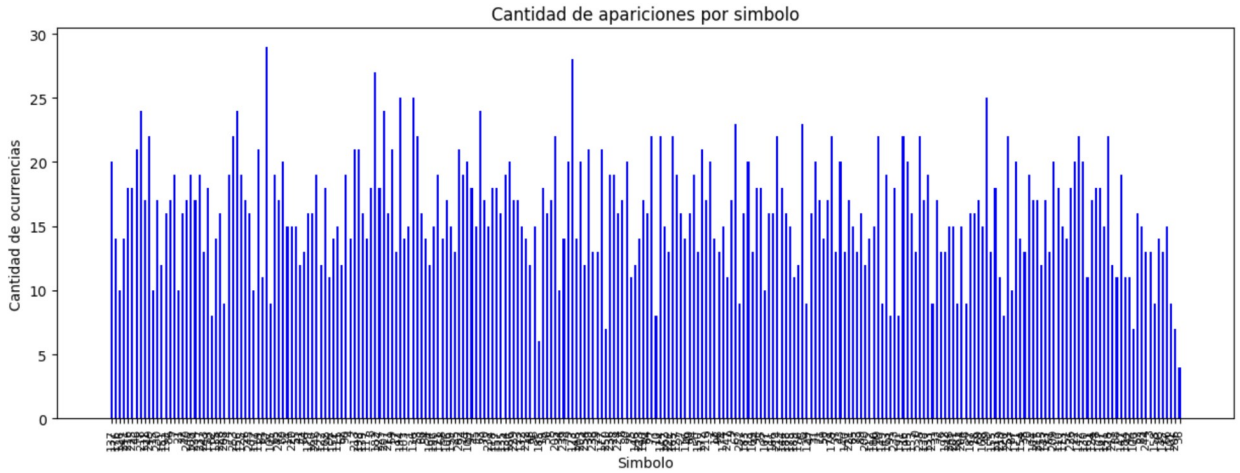
Ahora bien, dado que es un **archivo con 25 caracteres distintos** los podemos representar con 5 bits o bien $2^5$.

Por lo cual nuestro cálculo de una teórica tasa de compresión es de:

$$Tasa\,de\,compresión = \frac{H(s)}{L} = \frac{2.15\,bits/simbolo}{5\,bits/simbolo} = 0.43$$

Con lo cual en un esquema de compresión óptimo **con el 43% de la información original podríamos representar el archivo sin pérdidas**.

### 3. imagen_1.raw



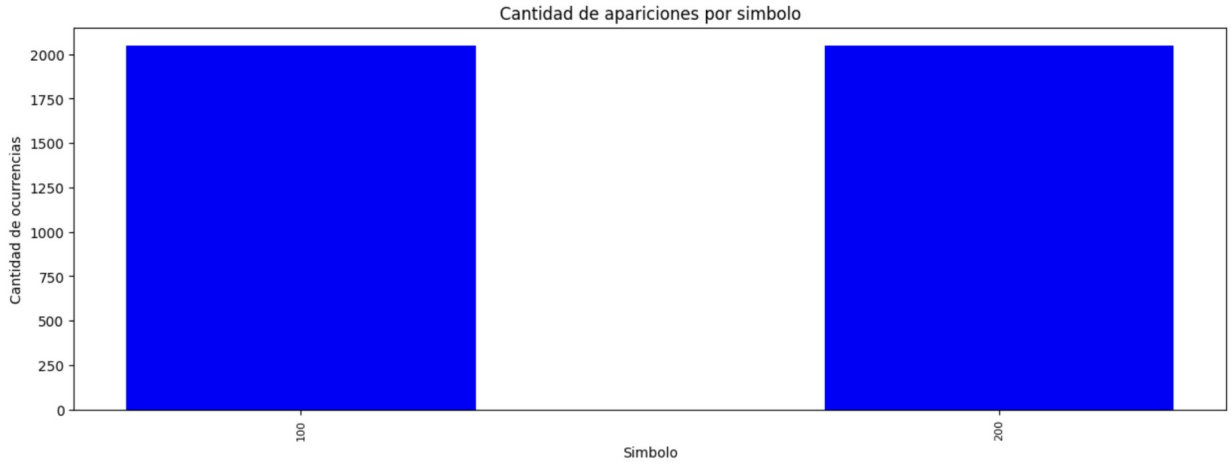

En este caso tenemos una distribución bastante uniforme entre todos los tipos diferentes de píxeles (de 0 a 255 o 256 pixeles en total).

En este caso la entropía del mismo es de 7.9474 bits/símbolo, y dado que necesitamos 8 bits para representar todos los diferentes valores de los píxeles $2^8$ = 256:

$$Tasa\,de\,compresión = \frac{H(s)}{L} = \frac{7.9474\,bits/simbolo}{8\,bits/simbolo} = 0.993$$

Con lo cual en un esquema de compresión óptimo **con el 99.3% de la información original podríamos representar el archivo sin pérdidas**.

## 4. imagen_2.raw



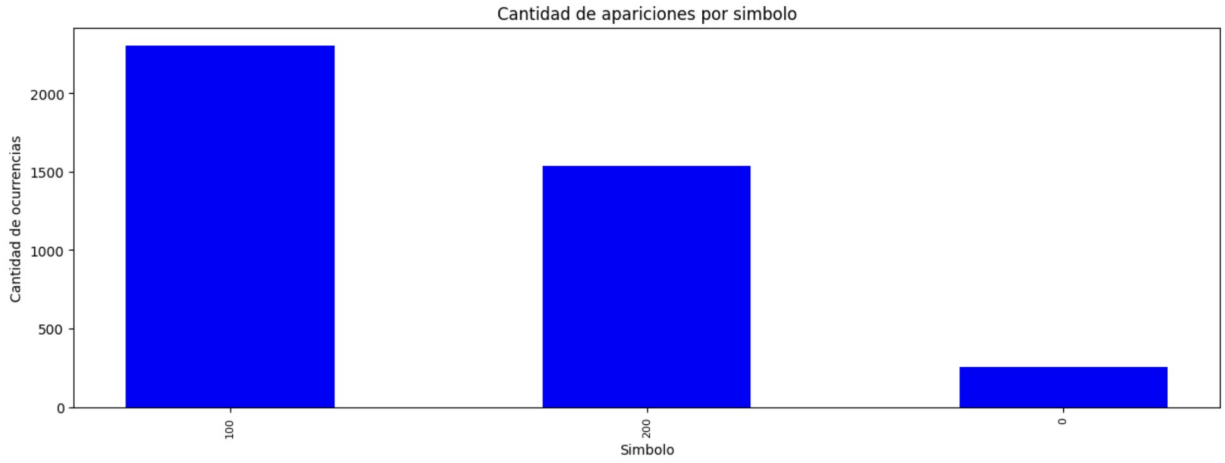

En este archivo tenemos una distribución uniforme entre **2 tipos de píxeles (**100 y 200).

La entropía del mismo es de 1.0 bits/símbolo, y dado que necesitamos 8 bits para representar todos los diferentes valores de los píxeles:

$$Tasa\,de\,compresión = \frac{H(s)}{L} = \frac{1.0\,bits/simbolo}{8\,bits/simbolo} = 0.125$$

Con lo cual en un esquema de compresión óptimo **con el 12.5% de la información original podríamos representar el archivo sin pérdidas.**

## 5. imagen_3.raw



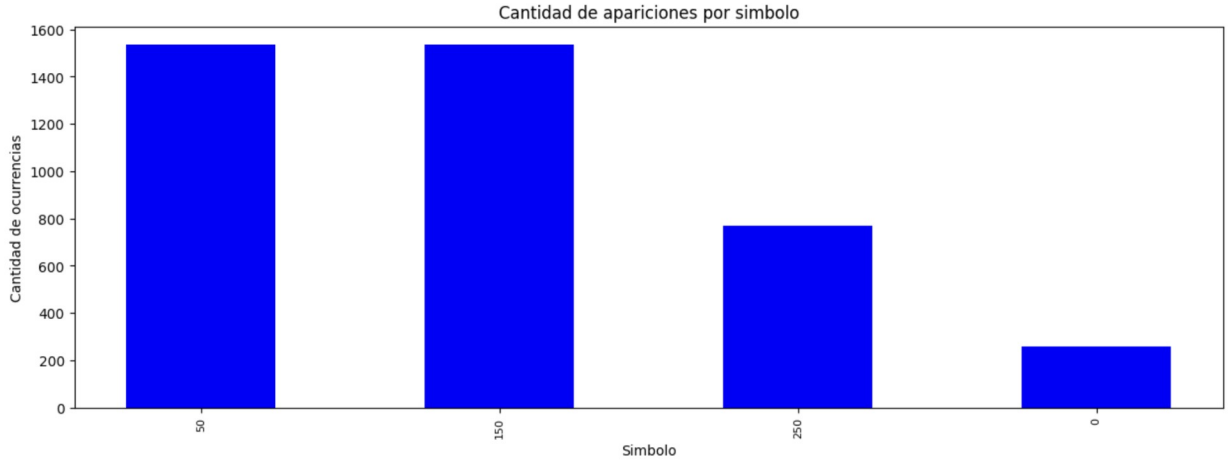

En este archivo tenemos una distribución es **menos uniforme** que en el caso anterior, donde claramente hay más cantidad de píxeles de tipo “100” que “200” y “0”.

La entropía del mismo es de 1.2476 bits/símbolo, y dado que necesitamos 8 bits para representar todos los diferentes valores de los píxeles:

$$Tasa\,de\,compresión = \frac{H(s)}{L} = \frac{1.2476\,bits/simbolo}{8\,bits/simbolo} = 0.156$$

Con lo cual en un esquema de compresión óptimo **con el 15.6% de la información original podríamos representar el archivo sin pérdidas**.

## 6. imagen_4.raw



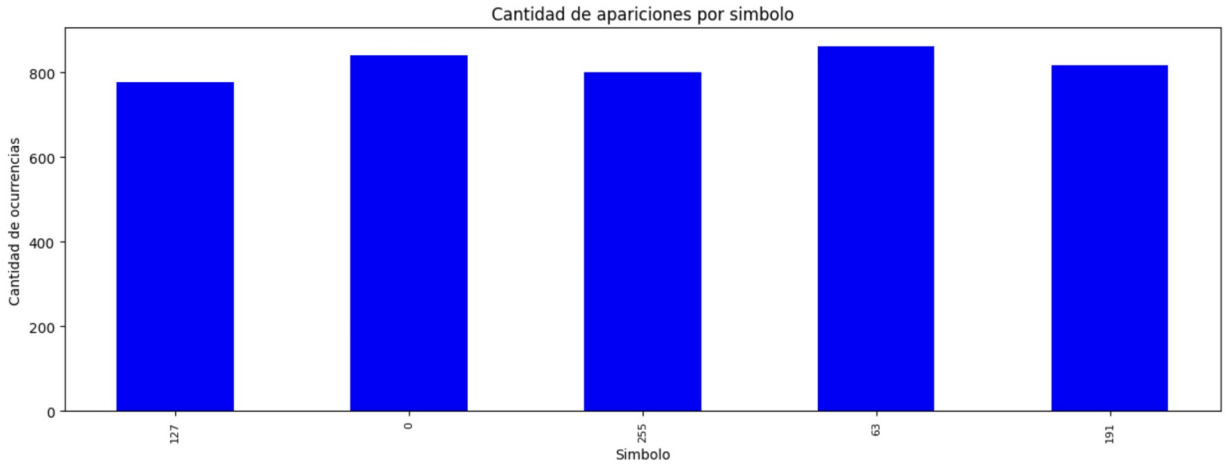

En este archivo tenemos una distribución cada uniforme entre 2 tipos de píxeles (50 y 150) y más dispar con respecto a los píxeles “250” y “0”.

La entropía del mismo es de 1.7641 bits/símbolo, y dado que necesitamos 8 bits para representar todos los diferentes valores de los píxeles:

$$Tasa\,de\,compresión = \frac{H(s)}{L} = \frac{1.7641\,bits/simbolo}{8\,bits/simbolo} = 0.22$$

Con lo cual en un esquema de compresión óptimo **con el 22% de la información original podríamos representar el archivo sin pérdidas**.

## 7. imagen_5.raw



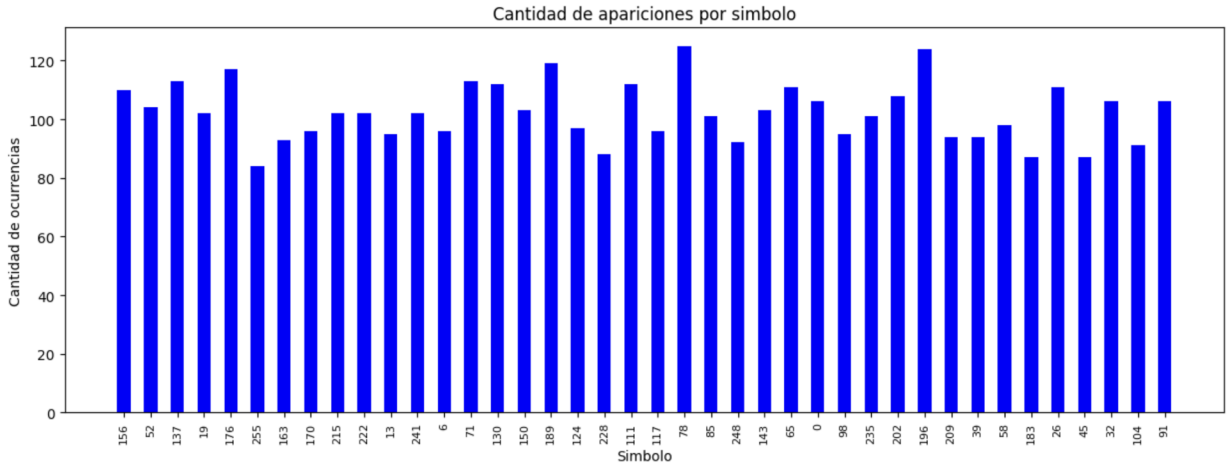

En este archivo tenemos una distribución bastante más uniforme entre 5 tipos de píxeles (127, 0, 255, 63 y 191).

La entropía del mismo es de 2.321 bits/símbolo, y dado que necesitamos 8 bits para representar todos los diferentes valores de los píxeles:

$$Tasa\,de\,compresión = \frac{H(s)}{L} = \frac{2.321\,bits/simbolo}{8\,bits/simbolo} = 0.29$$

Con lo cual en un esquema de compresión óptimo **con el 29% de la información original podríamos representar el archivo sin pérdidas.**

## 8. imagen_6.raw




En este archivo tenemos una distribución uniforme entre 40 tipos de píxeles.

La entropía del mismo es de 5.3152 bits/símbolo, y dado que necesitamos 8 bits para representar todos los diferentes valores de los píxeles:

$$Tasa\,de\,compresión = \frac{H(s)}{L} = \frac{5.3152\,bits/simbolo}{8\,bits/simbolo} = 0.66$$

Con lo cual en un esquema de compresión óptimo **con el 66% de la información original podríamos representar el archivo sin pérdidas**.

## Conclusiones

La teoría nos muestra que partiendo de la fórmula:

$$Tasa\ de\ compresión = \frac{H(s)}{L}$$

Si T (Tasa de compresión) **es cercana a 1**, significa que el archivo ya está representado de manera casi óptima y **la compresión adicional es mínima** y por contrario si T es menor que 1, indica que **existe redundancia** y el archivo **puede comprimirse** aún más.

Como podemos ver en los histogramas y cálculos demostrados al teorizar sobre la compresión óptima de los archivos, podemos inferir que lo que afecta su capacidad de compresión es:

- **La dispersión en su distribución:** A mayor dispersión, esto es,mayor diferencia entre la cantidad de apariciones por símbolo, **menor capacidad de compresión.**
- **La cantidad de símbolos en el archivo: A mayor cantidad de símbolos diferentes** en el archivo, **menor capacidad de compresión.**

En conclusión, la capacidad de compresión de un archivo depende tanto de la distribución de sus símbolos como de la cantidad de símbolos distintos que lo componen. Los archivos con una **distribución más sesgada** (donde algunos símbolos aparecen mucho más que otros) y una **menor variedad de símbolos tienden a ser más compresibles**, mientras que aquellos con una **distribución más uniforme** y **mayor cantidad de símbolos distintos presentan menor redundancia** y, por ende, una **menor capacidad de compresión**.

Esto se observa claramente en el archivo **imagen_2.raw**, que tiene una baja dispersión y un

conjunto reducido de símbolos, lo que permite una alta tasa de compresión (hasta un ~90%).

Por el contrario, en **imagen_1.raw**, la mayor cantidad de símbolos distintos y su distribución más uniforme hacen que la capacidad de compresión sea casi nula (~0.7%).

Este análisis confirma que la modelización adecuada de la fuente es clave para optimizar la representación de los datos y predecir su compresibilidad teórica.