Compresión de Datos

Compresión aritmética
Contextos

Compresión aritmética

Compresión aritmética

- Es un método estadístico
- Mejora los niveles de compresión de Huffman
- No asigna a cada símbolo un código en bits de longitud entera
- Conceptualmente codifica el archivo entero como un número real de 0 a 1
- Puede ser estático o dinámico

Aritmético estático

- Se divide el intervalo de 0 a 1 según la probabilidad de los símbolos
- En cada paso se elige un subintervalo según el símbolo leído
- En el siguiente paso se divide el subintervalo elegido de la misma forma
- Se repite hasta llegar al último carácter
- Finalmente se emite un número del subintervalo elegido en el último paso

Aritmético estático

- División de un intervalo:
  - Se divide el intervalo [$Piso_1$, $Techo_N$] en N subintervalos de diversas longitudes
  - Sea L = $Techo_N$ - $Piso_1$
  - $Techo_i$ = $Piso_i$ + $P(s_i).L$
  - $Piso_{i+1}$ = $Techo_i$
- En el siguiente paso, el subintervalo elegido es el que se subdivide

Aritmético estático

- Ejemplo: DIVIDIDOS

Carácter (mensaje)	Frecuencia de aparición	Probabilidad de aparición
D	3	1/3
I	3	1/3
O	1	1/9
S	1	1/9
V	1	1/9

Aritmético estático

	I	II	III	IV	V
-	**1,000**	1,0000	0,888889	0,790123	**0,786008**
	|	|	|	|	|
D	|	|	|	|	|
	|	|	|	|	|
-	**2/3**	**0,888889**	0,851852	**0,786008**	**0,784636**
	|	|	|	|	|
I	|	|	|	|	|
	|	|	|	|	|
-	1/3	**0,777778**	0,814815	**0,781893**	0,783265
O	|	|	|	|	|
-	2/9	0,740741	0,802469	0,780521	0,782807
S	|	|	|	|	|
-	1/9	0,703704	**0,790123**	0,779150	0,782350
V	|	|	|	|	|
-	0,000	0,666666	**0,777778**	0,777778	0,781893
	Leo la "D"	Leo la "I"	Leo la "V"	Leo la "I"	Leo la "D"

Aritmético estático

	VI	VII	VIII	IX	X
-	0,786008	**0,786008**	0,785551	0,785449	(0,785436)
	|	|	|	|	
D	|	|	|	|	
	|	|	|	|	
-	**0,785551**	**0,784636**	0,785500	0,785443	
	|	|	|	|	
I	|	|	|	|	
	|	|	|	|	**0,785435**
-	**0,785094**	0,783265	**0,785432**	0,785438	
O	|	|	|	|	
-	0,784941	0,782807	**0,785416**	**0,785436**	
S	|	|	|	|	
-	0,784789	0,782350	0,785416	**0,785434**	
V	|	|	|	|	
-	0,784636	0,781893	0,785399	0,785432	(0,785434)
	Leo la "I"	Leo la "D"	Leo la "O"	Leo la "S"	EMITIDO

Compresión aritmética

- Los intervalos de lectura se van achicando
- Cuanto más chico es el intervalo, más precisión hace falta para representarlo
- Cuanto más probable es un símbolo, menos bits se emiten cuando ocurre
- El método estático emite la tabla de frecuencias

Aritmético dinámico

- Se asigna frecuencia 1 a todos los símbolos
- Se divide el intervalo de 0 a 1 según la probabilidad dada por la tabla de frecuencias
- En cada paso se elige un subintervalo según el símbolo leído
- Luego se modifica la tabla de frecuencias
- En el siguiente paso se divide el subintervalo elegido según la nueva tabla
- Se repite hasta llegar al último carácter
- Finalmente se emite un número del subintervalo elegido en el último paso

Aritmético dinámico

	I	II	III	IV	V
-	1,000	1,0000	0,933333	0,904762	0,903571
D	|	|	|	|	|
-	4/5	0,933333	0,923810	0,903571	0,903307
I	|	|	|	|	|
-	3/5	0,900000	0,914286	0,902381	0,902910
O	|	|	|	|	|
-	2/5	0,866667	0,909524	0,901786	0,902778
S	|	|	|	|	|
-	1/5	0,833333	0,904762	0,901190	0,902646
V	|	|	|	|	|
-	0,000	0,800000	0,900000	0,900000	0,902381
	Leo la "D"	Leo la "I"	Leo la "V"	Leo la "I"	Leo la "D"

Aritmético dinámico

	VI	VII	VIII	IX	X
-	0,903571	0,903492	0,903492	0,9034782	(0,9034768)
D	|	|	|	|	
-	0,903492	0,903470	0,903485	0,9034776	
I	|	|	|	|	
-	0,904325	0,903442	0,903478	0,9034771	**0,9034767**
O	|	|	|	|	otra elección
-	0,903386	0,903435	0,903476	0,9034768	**0,90347675**
S	|	|	|	|	
-	0,903360	0,903427	0,903475	0,9034767	
V	|	|	|	|	
-	0,903307	0,903413	0,903471	0,9034764	(0,9034767)
	Leo la "I"	Leo la "D"	Leo la "O"	Leo la "S"	EMITIDO

Estático vs. Dinámico

- Las diferencias son las mismas que para Huffman:
  - El compresor estático debe hacer dos pasadas, el dinámico sólo una
  - El compresor estático comprime de forma óptima, el dinámico no ya que va aprendiendo gradualmente la tabla de frecuencias
  - El estático emite la tabla de frecuencias, el dinámico no

Aritmética de enteros

- Las computadoras no trabajan nativamente con números reales de precisión arbitraria
- En vez de utilizar un intervalo real para trabajar, se elige un intervalo entero de 0 a 2^N. Los números de ese intervalo tendrán N bits de precisión

Aritmética de enteros

- División de un intervalo:
  - Se divide el intervalo [$Piso_1$, $Techo_N$] en N subintervalos de diversas longitudes
  - Sea L = $Techo_N$ - $Piso_1$
  - $Techo_i$ = floor($Piso_i$ + $P(s_i)$.L) - 1
  - $Piso_{i+1}$ = $Techo_i$ + 1
- En el siguiente paso, el subintervalo elegido es el que se subdivide

Aritmética de enteros

- Renormalización: overflow

Si el piso y el techo del intervalo comparten el/los dígito(s) más significativo(s), éste se emite y se lo elimina del piso y el techo. El dígito menos significativo se completa, en el techo con un 1 y en el piso con un 0:

Techo: 11110101 → 11010111

Piso: 11001010 → 00101000

Aritmética de enteros

- Renormalización: underflow

Si el piso es de la forma (01…) y el techo es de la forma (10…), se normaliza de la siguiente forma:

Techo: 10110101 → 11101011
Piso: 01001010 → 00010100 (U=1)

Techo: 10001101 → 11101111
Piso: 01111010 → 01010000 (U=3)

Aritmética de enteros

- Es posible que tras normalizar un overflow, se descubra un underflow

Techo: 11011001 → 11100111
Piso: 10101110 → 00111000 (U=1)

Aritmética de enteros

- Al detectar un overflow y realizar la emisión, se emiten tras el primer dígito tantos dígitos negados como sea el contador de underflow y se pone el contador en 0

(U=2)

Techo: 11101101 → 11101111

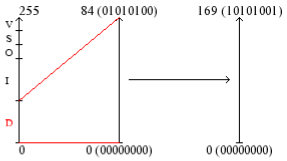
Piso: 11010110 → 00110000

Emisión: 1-00-1, nuevo valor de U=1

Aritmética de enteros
Ejemplo estático: DIVIDIDOS

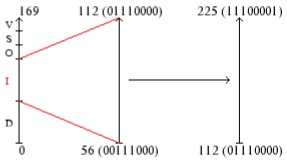
Compresión del caracter 'D' - (Posición: 0)
Piso inicial: 0 - Techo Inicial 255
Intervalo: 256
Nuevo Piso = 0 + 256 * 0 / 9 = **0**
Nuevo Techo = 0 + 256 * 3 / 9 - 1 = **84**
Resultado de la normalización:

- Piso: 00000000 (0) -> 00000000 (0)
- Techo: 01010100 (84) -> 10101001 (169)
- Emisión: **0**
- Contador de underflow: 0



Compresión del caracter 'I' - (Posición: 1)
Piso inicial: 0 - Techo Inicial 169
Intervalo: 170
Nuevo Piso = 0 + 170 * 3 / 9 = **56**
Nuevo Techo = 0 + 170 * 6 / 9 - 1 = **112**
Resultado de la normalización:

- Piso: 00111000 (56) -> 01110000 (112)
- Techo: 01110000 (112) -> 11100001 (225)
- Emisión: **0**
- Contador de underflow: 0



Compresión del caracter 'V' - (Posición: 2)
Piso inicial: 112 - Techo Inicial 225
Intervalo: 114
Nuevo Piso = 112 + 114 * 8 / 9 = **213**
Nuevo Techo = 112 + 114 * 9 / 9 - 1 = **225**
Resultado de la normalización:

- Piso: 11010101 (213) -> 00101000 (40)
- Techo: 11100001 (225) -> 10001111 (143)
- Emisión: **11**
- Contador de underflow: 1

Aritmética de enteros
Ejemplo estático: DIVIDIDOS

Compresión del caracter 'I' - (Posición: 3)
Piso inicial: 40 - Techo Inicial 143
Intervalo: 104
Nuevo Piso = 40 + 104 * 3 / 9 = **74**
Nuevo Techo = 40 + 104 * 6 / 9 - 1 = **108**
Resultado de la normalización:

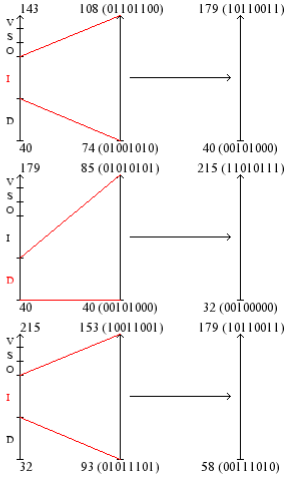
- Piso: 01001010 (74) -> 00101000 (40)
- Techo: 01101100 (108) -> 10110011 (179)
- Emisión: **011**
- Contador de underflow: 0

Compresión del caracter 'D' - (Posición: 4)
Piso inicial: 40 - Techo Inicial 179
Intervalo: 140
Nuevo Piso = 40 + 140 * 0 / 9 = **40**
Nuevo Techo = 40 + 140 * 3 / 9 - 1 = **85**
Resultado de la normalización:

- Piso: 00101000 (40) -> 00100000 (32)
- Techo: 01010101 (85) -> 11010111 (215)
- Emisión: **0**
- Contador de underflow: 1

Compresión del caracter 'I' - (Posición: 5)
Piso inicial: 32 - Techo Inicial 215
Intervalo: 184
Nuevo Piso = 32 + 184 * 3 / 9 = **93**
Nuevo Techo = 32 + 184 * 6 / 9 - 1 = **153**
Resultado de la normalización:

- Piso: 01011101 (93) -> 00111010 (58)
- Techo: 10011001 (153) -> 10110011 (179)
- Contador de underflow: 2



Aritmética de enteros
Ejemplo estático: DIVIDIDOS

Compresión del caracter 'D' - (Posición: 6)
Piso inicial: 58 - Techo Inicial 179
Intervalo: 122
Nuevo Piso = 58 + 122 * 0 / 9 = **58**
Nuevo Techo = 58 + 122 * 3 / 9 - 1 = **97**
Resultado de la normalización:

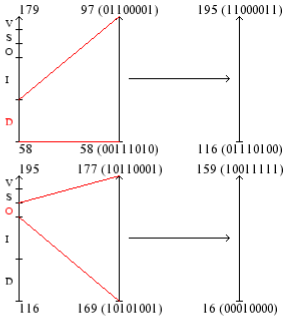
- Piso: 00111010 (58) -> 01110100 (116)
- Techo: 01100001 (97) -> 11000011 (195)
- Emisión: **011**
- Contador de underflow: 0

Compresión del caracter 'O' - (Posición: 7)
Piso inicial: 116 - Techo Inicial 195
Intervalo: 80
Nuevo Piso = 116 + 80 * 6 / 9 = **169**
Nuevo Techo = 116 + 80 * 7 / 9 - 1 = **177**
Resultado de la normalización:

- Piso: 10101001 (169) -> 00010000 (16)
- Techo: 10110001 (177) -> 10011111 (159)
- Emisión: **101**
- Contador de underflow: 1

Compresión del caracter 'S' - (Posición: 8)
Piso inicial: 16 - Techo Inicial 159
Intervalo: 144
Nuevo Piso = 16 + 144 * 7 / 9 = **128**
Nuevo Techo = 16 + 144 * 8 / 9 - 1 = **143**
Al ser el ultimo caracter, emito el piso, incluyendo los underflows:

- Piso: 10000000 (128)
- Emisión: **100000000**



Aritmética de enteros

- Para comprimir con aritmético dinámico y aritmética de enteros, combinar los conceptos vistos en dinámico+codificación real y estático+codificación entera, y tener en cuenta que:
- En el caso dinámico, se debe utilizar un símbolo para señalar el EOF, ya que no se emite una tabla de frecuencias que permita inferir la cantidad de caracteres

Descompresión aritmética

- Leer el número (o un número de la cantidad de dígitos de trabajo) y fijarse en qué parte del intervalo cae
- Emitir el símbolo al que le corresponde el subintervalo que contiene al valor
- Actualizar los límites del intervalo
- Actualizar la tabla de frecuencias si el método es dinámico
- Repetir el paso para los siguientes símbolos hasta llegar al fin de archivo.

Contextos

Contextos

- Se intenta predecir el símbolo siguiente. Para cada carácter, el o los caracteres precedentes son su contexto
- Se parte de la base de que un texto tiene secuencias que se repiten
  - En el contexto de “Marado” es habitual encontrar una ‘n’

Contextos

- El orden de un contexto es la cantidad de caracteres precedentes que se toman en cuenta para predecir
- Para cada carácter o conjunto de ellos, se almacena una tabla de probabilidades asociada. Si se utilizan contextos de orden 1, la tablas asociadas a ‘Q’ y a ‘P’ serán muy distintas

Contextos

- Utilizar distintas tablas de probabilidades según el contexto ayuda a predecir mejor
- Una mejor predicción equivale a una menor emisión en bits
- Si los datos no tienen patrones repetidos, utilizar contextos no mejora la compresión