Cuantización

Fernando Martínez
fernando.martinez@upc.edu

Departament de Matemàtiques • Universitat Politècnica de Catalunya

12 de abril de 2021

Capítulos 9 y 10 de Introduction to Data Compression **Sayood, Khalid** Morgan Kaufmann, 2012, 4th ed.

Cuantización

Ejemplo

Imagen de grises 8 bits [0,256), codificamos con 2 bits:

0 | 64 | 128 | 192 | 256

[32)[96)[160)[224)

00 | 01 | 10 | 11

Original

Uniforme: 2 bits por píxel

Cuantización escalar

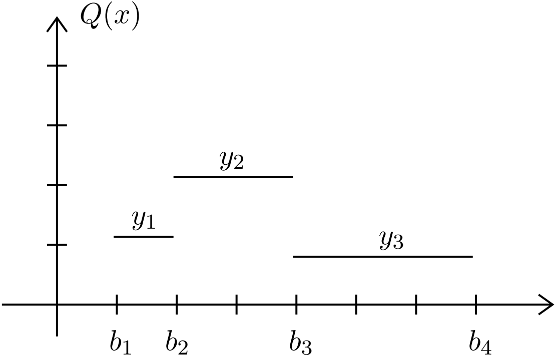

Figura: Función de cuantización $Q(x)$.

$\{b_i\}_{i=0,\ldots,M}$ *límites de decisión*, $\{y_i\}_{i=1,\ldots,M}$ *niveles de reconstrucción*,

$$Q(x) = y_i \text{ si } b_{i-1} \leq x < b_i$$

Cuantización escalar: Problema general (I)

Problema: Modelizamos nuestras entradas con una variable aleatoria X con densidad de probabilidad $f(x)$.

Deseamos cuantizar una fuente dando los *límites de decisión* $\{b_i\}_{i=0,...,M}$ y los *niveles de reconstrucción* $\{y_i\}_{i=1,...,M}$ que determinan la función de cuantización $Q(x) = y_i$ si $b_{i-1} \leq x < b_i$.

Definimos*:

$$\sigma^2 \equiv \int_{-\infty}^{\infty} [x - Q(x)]^2 f(x)dx = \sum_{i=1}^{M} \int_{b_{i-1}}^{b_i} [x - Q(x)]^2 f(x)dx \qquad (1)$$

$\mathcal{C}$ código con M palabra de longitudes $\{l_1, ..., l_M\}$. Si $\tilde{l} = \sum_{i=1}^{M} l_i p(y_i)$, $p(y_i) = \int_{b_{i-1}}^{b_i} f(x)dx$, entonces

$$\tilde{l} = \sum_{i=1}^{M} \int_{b_{i-1}}^{b_i} l_i f(x)dx \qquad (2)$$

*Hay otras *distancias* o *errores*, σ, que usan $|x - Q(x)|$, $\sup |x - Q(x)|$...

Cuantización escalar: Problema general (II)

Problema A Dado $\epsilon > 0$, una cota al *error*, hallar los límites de decisión $\{b_i\}$, los niveles de reconstrucción $\{y_i\}$ y las longitudes de las palabras del código $\{l_i\}$ que minimizan (2) y satisfacen $\sigma < \epsilon$.

Problema B Dado $\epsilon > 0$, una cota a la longitud media, hallar los límites de decisión $\{b_i\}$, los niveles de reconstrucción $\{y_i\}$ y las longitudes de las palabras del código $\{l_i\}$ que minimizan (1) y satisfacen $\tilde{l} < \epsilon$.

Cuantización escalar uniforme

Todos los intervalos son de la misma longitud, salvo tal vez los extremos:

$$x \in [-X_{\max}, X_{\max}], \quad b_i - b_{i-1} = \Delta, \quad \Delta = \frac{2\,X_{\max}}{M}$$

Distribución uniforme $f(x) = \frac{1}{2\,X_{\max}}$

Minimizar σ: $y_i = \frac{b_i + b_{i-1}}{2}$, $y_i - y_{i-1} = \Delta$ niveles de reconstrucción equiespaciados.

Al ser una distribución uniforme podemos utilizar un código de bloque de palabras de longitud n si $M = 2^n$:

$$\sigma^2 = \sum_{i=1}^{M} \int_{(i-1)\Delta}^{i\Delta} \left[x - \frac{2\,i-1}{2}\Delta\right]^2 \frac{1}{2\,X_{\max}} dx = \frac{\Delta^2}{12} = \frac{1}{3}\frac{X_{\max}^2}{M^2}.$$

Duplicar el número de intervalos disminuye el error a la mitad. En códigos de bloques, aumentar en 1 bit el tamaño de las palabras reduce el error a la mitad.

Cuantización escalar uniforme adaptativa

Aunque $\Delta = b_i - b_{i-1}$ es constante, puede variar con el *tiempo* dependiendo de los valores que va tomando la variable aleatoria.

- **FAQ** Forward Adaptative Quantization: Se leen bloques de N datos y se realiza la cuantización en base a ellos.
- **BAQ** Backward Adaptative Quantization: Se actualiza Δ a medida que se van leyendo los datos.

Ejemplo

FAQ: De una imagen se leen bloques de 8×8 píxeles, se calcula su máximo y su mínimo y se cuantiza en dicho rango $[\text{mín}, \text{máx}]$ (se han de guardar dichos valores al codificar).

Ejemplo

BAQ. Jayant Quantizer: se varía Δ según si los datos van cayendo en los niveles *internos* o *externos*; si es en los *internos* se disminuye Δ, si es en los *externos* se aumenta Δ.

Cuantización escalar no uniforme

μ-law

Es un método de cuantificación *logarítmica* que se usa para señales de audio: la entrada se representa en complemento a 2 con 14 bits y se realizan una serie de transformaciones para obtener un código de 14 bits en los que s es el signo. Después se le aplica una cuantización escalar no uniforme según la tabla:

14 bit Binary input code	Compressed code	14 bit Binary Output Code
s00000001abcdx	s000abcd	s00000001abcd1
s0000001abcdxx	s001abcd	s0000001abcd10
s000001abcdxxx	s010abcd	s000001abcd100
s00001abcdxxxx	s011abcd	s00001abcd1000
s0001abcdxxxxx	s100abcd	s0001abcd10000
s001abcdxxxxxx	s101abcd	s001abcd100000
s01abcdxxxxxxx	s110abcd	s01abcd1000000
s1abcdxxxxxxxx	s111abcd	s1abcd10000000

Cuantización escalar no uniforme

μ-law

Input value	8 bit Compressed code	Output value
+8158 to +4063 in 16 intervals of 256	0x80 + interval number	8031-256k
+4062 to +2015 in 16 intervals of 128	0x90 + interval number	3999-128k
+2014 to +991 in 16 intervals of 64	0xA0 + interval number	1983-64k
+990 to +479 in 16 intervals of 32	0xB0 + interval number	975-32k
+478 to +223 in 16 intervals of 16	0xC0 + interval number	471-16k
+222 to +95 in 16 intervals of 8	0xD0 + interval number	119-8k
+94 to +31 in 16 intervals of 4	0xE0 + interval number	93-4k
+30 to +1 in 15 intervals of 2	0xF0 + interval number	30-2k
0	0xFF	0
-1	0x7F	-1
-31 to -2 in 15 intervals of 2	0x70 + interval number	-30+2k
-95 to -32 in 16 intervals of 4	0x60 + interval number	-93+4k
-223 to -96 in 16 intervals of 8	0x50 + interval number	-119+8k
-479 to -224 in 16 intervals of 16	0x40 + interval number	-471+16k
-991 to -480 in 16 intervals of 32	0x30 + interval number	-975+32k
-2015 to -992 in 16 intervals of 64	0x20 + interval number	-1983+64kk
-4063 to -2016 in 16 intervals of 128	0x10 + interval number	-3999+128k
-8159 to -4064 in 16 intervals of 256	0x00 + interval number	-8131+256k

Cuantización escalar no uniforme (I)

Volvamos al caso general consideremos los problemas:

(A) Fijamos $\{b_i\}$, minimizar σ^2 respecto $\{y_i\}$. Se ha de cumplir:

$$y_i = \frac{\int_{b_{i-1}}^{b_i} x\, f(x)\, dx}{\int_{b_{i-1}}^{b_i} f(x)\, dx}$$

y_i *centro de masas* del intervalo.

(B) Fijamos $\{y_i\}$, minimizar σ^2 respecto $\{b_i\}$. Se ha de cumplir:

$$b_i = \frac{y_i + y_{i+1}}{2}$$

Los límites de decisión son el punto medio de los niveles de reconstrucción.

Cuantización escalar no uniforme (II)

El problema general es hallar $\{b_i\}$ e $\{y_i\}$.

1. Fijemos los valores iniciales de $\{y_i\}$. Por **(B)** podemos calcular la primera aproximación de los valores $\{b_i\}$. Ahora usamos **(A)** para actualizar los valores $\{y_i\}$ y así sucesivamente.
2. Supongamos que deseamos hallar $b_0 = 0$, b_1, b_2,..., y_1, y_2,... Demos un valor tentativo a y_1**, resolvemos numéricamente **(A)** $y_1 = \frac{\int_{b_0}^{b_1} x\, f(x)\, dx}{\int_{b_0}^{b_1} f(x)\, dx}$ siendo y_1 la incógnita. Usando **(B)** obtenemos b_1, ahora podemos calcular y_2 resolviendo numéricamente **(A)**...

** El resultado dependerá del valor tomado

Cuantización vectorial (I)

Tener un diccionario compuesto por vectores $\{\overrightarrow{y_i}\}$ y codificar nuestros vectores según indica el diccionario.

Ejemplo

Códigos correctores de errores:
(0 0 0 0 0 0 0) 0000
(0 0 0 0 0 1 0) $\longrightarrow$ (0 0 0 0 0 0 0) 0000
(1 0 0 0 0 1 0) $\longrightarrow$ (0 0 0 0 0 0 0) 0000
(0 0 0 0 0 1 1) $\longrightarrow$ (1 0 0 0 0 1 1) 1001
(0 0 0 0 1 0 1) $\longrightarrow$ (0 1 0 0 1 0 1) 0101
(0 0 0 0 1 1 0) $\longrightarrow$ (0 0 1 0 1 1 0) 0010
(0 0 0 1 0 1 1) $\longrightarrow$ (0 0 0 1 1 1 1) 0001
(0 0 0 1 1 0 0) $\longrightarrow$ (1 0 0 1 1 0 0) 1000
(0 1 0 0 1 0 1) 0101
(1 0 1 1 0 1 0) 1010

Cuantización vectorial (II)

K-means – Linde-Buzo-Gray (LGB)

1. Inicializamos un conjunto de vectores de reconstrucción $\{\overrightarrow{y_i}^{(1)}\}$, $k = 1$, $D^{(0)} = 0$, ϵ lindar.
2. Hallamos las regiones de cuantización
$$V_i^{(k)} = \left\{ \overrightarrow{x},\ d(\overrightarrow{x}, \overrightarrow{y_i}^{(k)}) < d(\overrightarrow{x}, \overrightarrow{y_j}^{(k)})\ \forall j \neq i \right\}$$
3. Calculamos la distorsión
$$D^{(k)} = \sum_{i=1}^{M} \int_{V_i^{(k)}} \|\overrightarrow{x} - \overrightarrow{y_i}^{(k)}\| f(\overrightarrow{x})\, d\overrightarrow{x}$$
4. Si $\dfrac{D^{(k)} - D^{(k-1)}}{D^{(k)}} < \epsilon$ entonces `stop`.
5. $K = k + 1$, calculamos los nuevos valores de $\{\overrightarrow{y_i}^{(k)}\}$, que son los centroides de $\left\{V_i^{(k)}\right\}$ y volvemos a 2.

Cuantización vectorial (III)

Observaciones:

I) Cuando se usa para clasificar objetos se utiliza un conjunto para entrenar y otro de test.

II) No siempre converge a la solución óptima

III) Depende de la elección inicial de $\{\overrightarrow{y_i}^{(1)}\}$.
Una forma de elegirlos es empezar con un vector que sea el centroide. A continuación se perturba para obtener dos vectores, se aplica el algoritmo, los nuevos centroides se vuelven a ”dividirz así hasta obtener el número de vectores deseado.

IV) Depende de las definiciones de $d(\overrightarrow{x}, \overrightarrow{y})$ y $\|\overrightarrow{x}\|$ y de cómo se actualiza.

Cuantización de métodos predictivos (I)

Ejemplo

Original	0	1,2	2,4	4,9	7,1	6,3	4,8	3,4	0,9	-0,9	-2,4	-4,1
Diferencias		1,2	1,2	2,5	2,2	-0,8	-1,5	-1,4	-2,5	-1,8	-1,5	-1,7
Cuantizado		1,5	1,5	2,5	2,5	-0,5	-1,5	-1,5	-2,5	-1,5	-1,5	-1,5
Recuperado		1,5	3	5,5	8	7,5	6	4,5	2	0,5	-1	-2,5
Error		0,3	0,6	0,6	0,9	1,2	1,2	1,1	1,1	1,4	1,4	1,6

Límites: ...-3,-2,-1,0,1,2,3...

Niveles -3,5, -2,5, -1,5, -0,5, 0,5, 1,5, 2,5, 3,5...

El error va creciendo poco a poco.

Cuantización de métodos predictivos (II)

$\{x_n\}$ la secuencia original
$\{d_n\}$ la secuencia de diferencias, $d_n = x_n - x_{n-1}$
$\{\hat{d}_n\}$ la secuencia de diferencias cuantizadas, $\hat{d}_n = Q(d_n) = d_n + e_n$
$\{\hat{x}_n\}$ la secuencia reconstruida
Calculemos $\hat{x}_n = \hat{x}_{n-1} + \hat{d}_n$, tomando $\hat{x}_0 = x_0$:

$$\hat{x}_1 = \hat{x}_0 + \hat{d}_1 = x_0 + d_1 + e_1 = x_0 + (x_1 - x_0) + e_1 = x_1 + e_1$$

$$\begin{aligned}\hat{x}_2 &= \hat{x}_1 + \hat{d}_2 = (x_1 + e_1) + (d_2 + e_2) = \\ &= x_1 + (x_2 - x_1) + e_1 + e_2 = x_2 + e_1 + e_2\end{aligned}$$

$$\ldots$$

$$\hat{x}_k = x_k + \sum_{i=1}^{k} e_i$$

Como la media del e_i será 0, se puede pensar que el error se cancelará a medida que recuperamos más elementos de la secuencia pero en la práctica no acostumbra a ser así (paseo aleatorio o del borracho).

Cuantización de métodos predictivos (III)

Para evitar este problema la secuencia de diferencias se construye

$$d_n = x_n - \hat{x}_{n-1}$$

Ejemplo

Original	0	1,2	2,4	4,9	7,1	6,3	4,8	3,4	0,9	-0,9	-2,4	-4,1
Diferencias		1,2	0,9	2,9	2,6	-0,7	-1,7	-1,6	-2,6	-1.9	-1,9	-2,1
Cuantizado		1,5	0,5	2,5	2,5	-0,5	-1,5	-1,5	-2,5	-1,5	-1,5	-2,5
Recuperado		1,5	2	4,5	7	6,5	5	3,5	1	-0.5	-2	-4,5
Error		0,3	-0,4	-0,4	-0,1	0,2	0,2	0,1	0,1	0,4	0.4	-0.4

Límites: ...-3,-2,-1,0,1,2,3...

Niveles -3,5, -2,5, -1,5, -0,5, 0,5, 1,5, 2,5, 3,5...

El error es el error producido en la cuantización.

Cuantización de métodos predictivos (IV)

$\{x_n\}$ la secuencia original
$\{d_n\}$ la secuencia de diferencias, $d_n = x_n - \hat{\mathbf{x}}_{\mathbf{n-1}}$
$\{\hat{d}_n\}$ la secuencia de diferencias cuantizadas, $\hat{d}_n = Q(d_n) = d_n + e_n$
$\{\hat{x}_n\}$ la secuencia reconstruida

$$\hat{x}_0 = x_0$$
$$\hat{x}_1 = \hat{x}_0 + \hat{d}_1 = \hat{x}_0 + d_1 + e_1 = \hat{x}_0 + (x_1 - \hat{x}_0) + e_1 = x_1 + e_1$$
$$\hat{x}_2 = \hat{x}_1 + \hat{d}_2 = \hat{x}_1 + (d_2 + e_2) = \hat{x}_1 + (x_2 - \hat{x}_1) + e_2 = x_2 + e_2$$
$$\ldots$$
$$\hat{x}_k = x_k + e_k$$

Cuantización de métodos predictivos (V)

En este proceso estamos asumiendo que el siguiente valor es igual al anterior y codificamos la diferencia:

$$p_n = f(\hat{x}_{n-1}) = \hat{x}_{n-1}, \quad d_n = x_n - p_n, \quad \hat{d}_n = Q(d_n) \quad \hat{x}_n = p_n + \hat{d}_n$$

En general podemos suponer:

$$p_n = f(\hat{x}_{n-1}, \hat{x}_{n-2}, ..., \hat{x}_{n-N})$$

f depende de $\hat{x}_i$ (no de x_i) porque el decodificador sólo tiene acceso a $\hat{x}_i$

Ejemplo

El valor de un píxel Θ se puede predecir a partir de los valores A, B, C,... ya leídos,

B	C	D
A	Θ	

- filtro lineal (media ponderada de los píxeles),
- filtro no lineal (mediana, moda, máximo, mínimo...)