

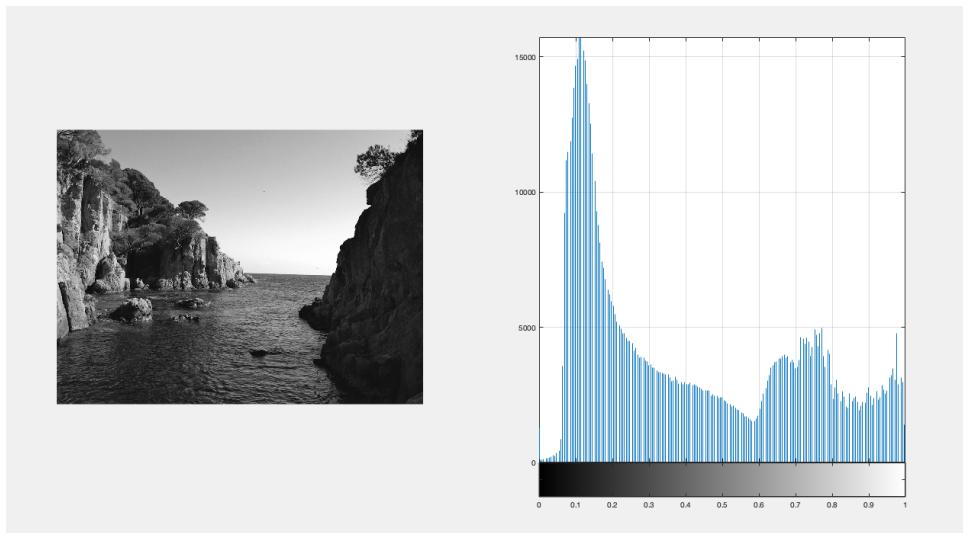
1. MEMORYLESS PROCESSES OF IMAGE SIGNALS

1.1. Image Histogram

En primer lloc analitzarem els histogrames de les cinc imatges per veure la freqüència dels diferents nivells de l'escala de grisos. Per fer-ho utilitzem la comanda:

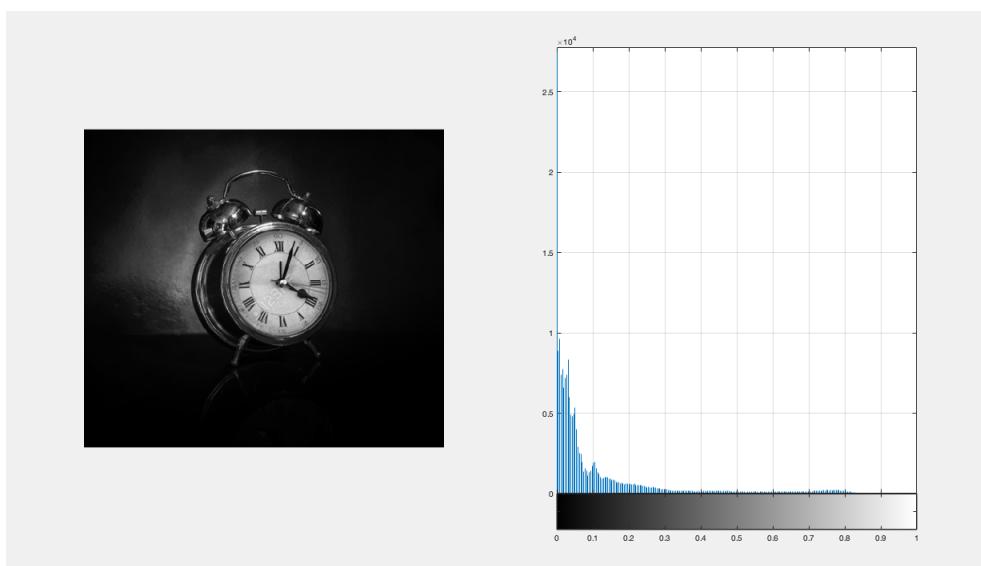
```
Image_Histogram(im1,ima2,ima3,ima4,ima5);
```

Imatge 1:



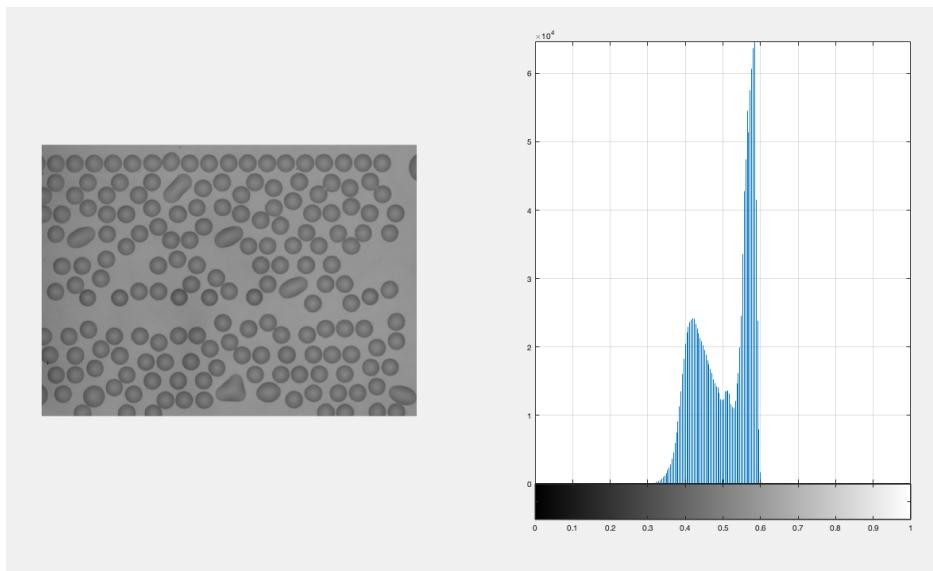
En l'histograma d'aquesta imatge, podem observar com hi ha molt contrast, ja que els valors de gris de la imatge es reparteixen en un rang molt llarg del gràfic. A més a més, s'observa com la imatge conté més negres que blancs, ja que hi ha més concentració de valors en el rang inferior.

Imatge 2:



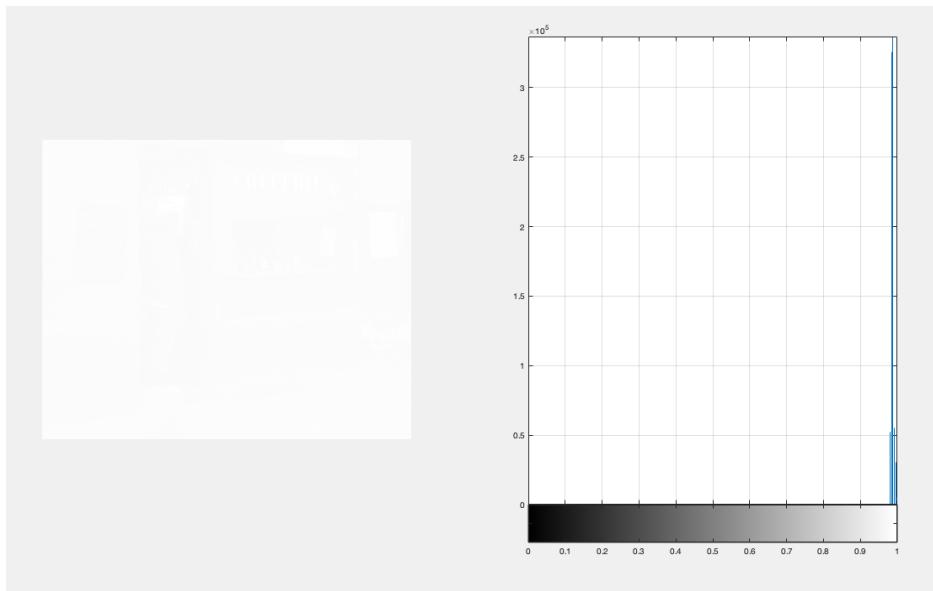
En aquesta imatge s'observa clarament que el color predominant és el negre, ja que quasi tots els valors es concentren en el rang inferior de la imatge, i al tenir els valors concentrats en un rang molt petit podem saber que la imatge té poc contrast.

Imatge 3:



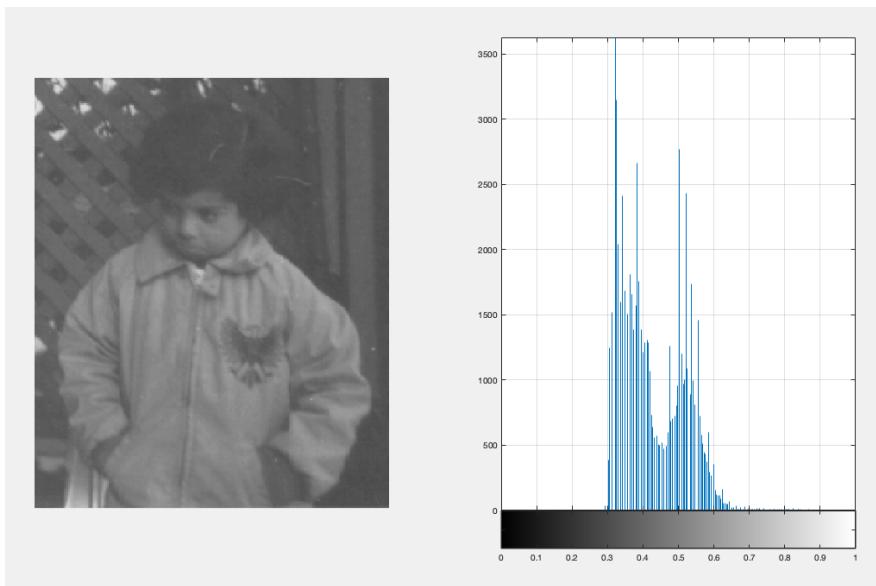
Aquesta imatge és una imatge amb poc contrast, ja que tots els seus valors estan concentrats en un rang molt petit de l'histograma. A més, els valors són intermitjós, és a dir, que predominen els grisos.

Imatge 4:



Aquesta imatge és la que té més blancs i amb menys contrast, com mostra l'histograma que conté tots els seus valors en el rang superior i en un rang extremadament petit.

Imatge 5:



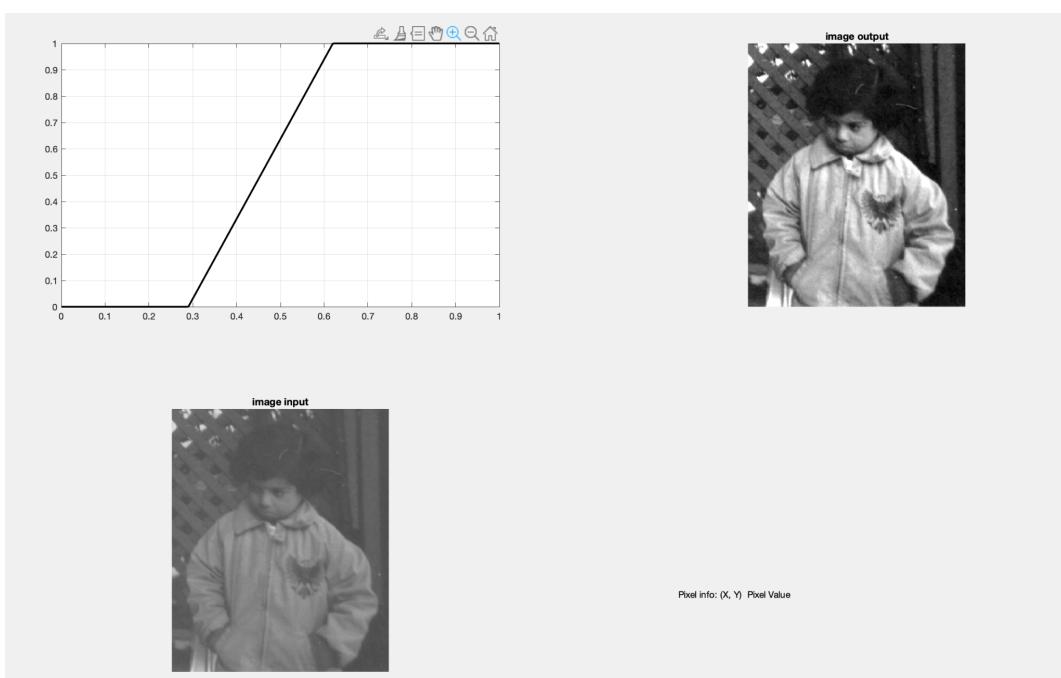
En aquesta imatge, no predomina ni el negre ni el blanc, però sí que hi ha poc contrast, ja que els valors de l'histograma es concentra en un rang petit.

1.2 Image contrast (linear and non-linear)

a) Enhance the contrast

Ara hem analitzat el contrast de la imatge 5 i l'hem augmentat mapejant els valors més petits de 0.29 a 0 i els més grans de 0.62 a 1. Hem perdut una mica d'informació ja que hem decidit agafar el valor 0.62 encara que hi havia algun valor més gran que hem considerat que no era rellevant. Al fer això estem mapejant tots els valors que queden per sobre de 0.62 a 1 junts, així que és una operació no reversible, aquests valors ja no els podrem separar. La comanda utilitzada és:

```
ima5_c=Image_Contrast(ima5,[0.29 0.62],[0.0 1.0]);
```

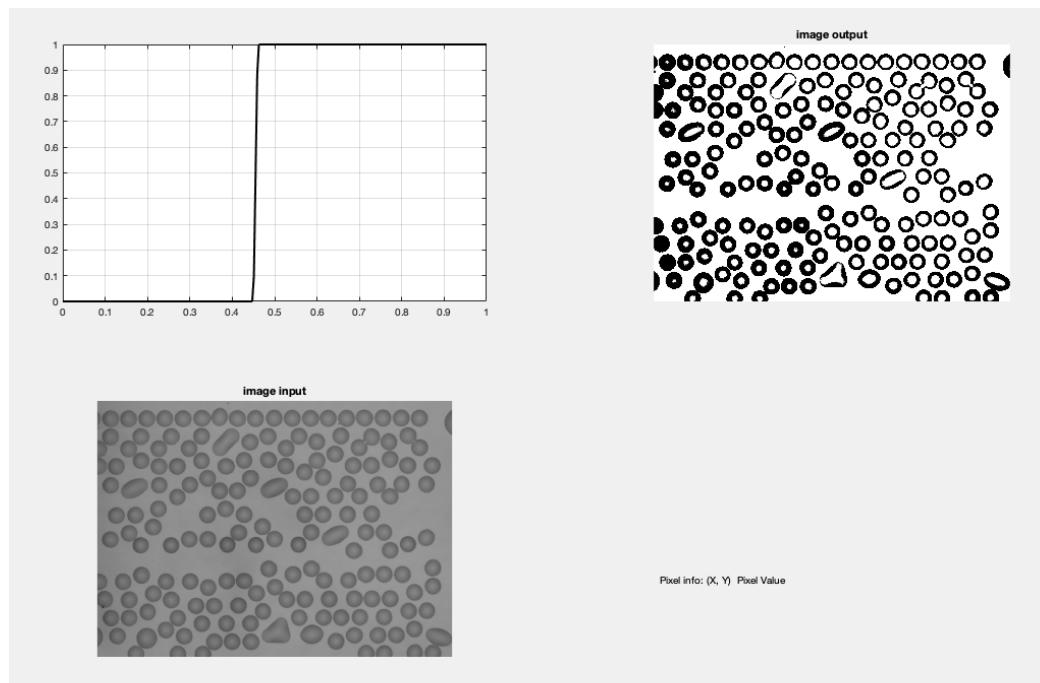


b) Before counting drops

Hem binaritzat la imatge per tal de distingir millor els cercles del fons de la imatge. Els cercles els hem clipejat a 0 (negres) i el fons a 1 (blanc). El llindar que hem escollit és el 0.45, el qual ens sembla que s'aproxima bastant a la imatge que volem conseguir.

La comanda utilitzada és:

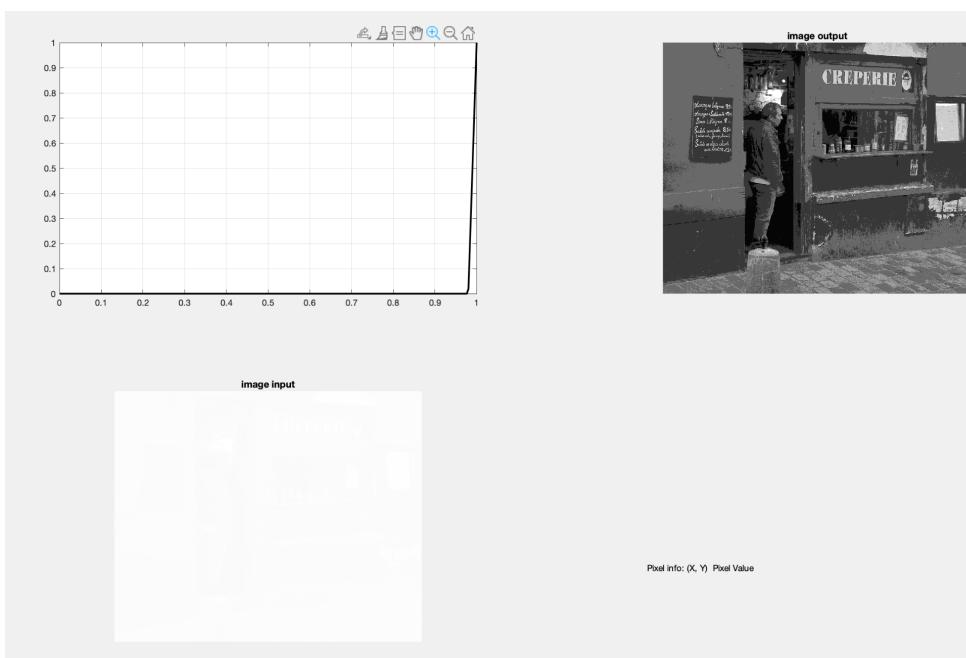
```
ima3_c=Image_Contrast(ima3, [0.45 0.46], [0.0 1.0]);
```



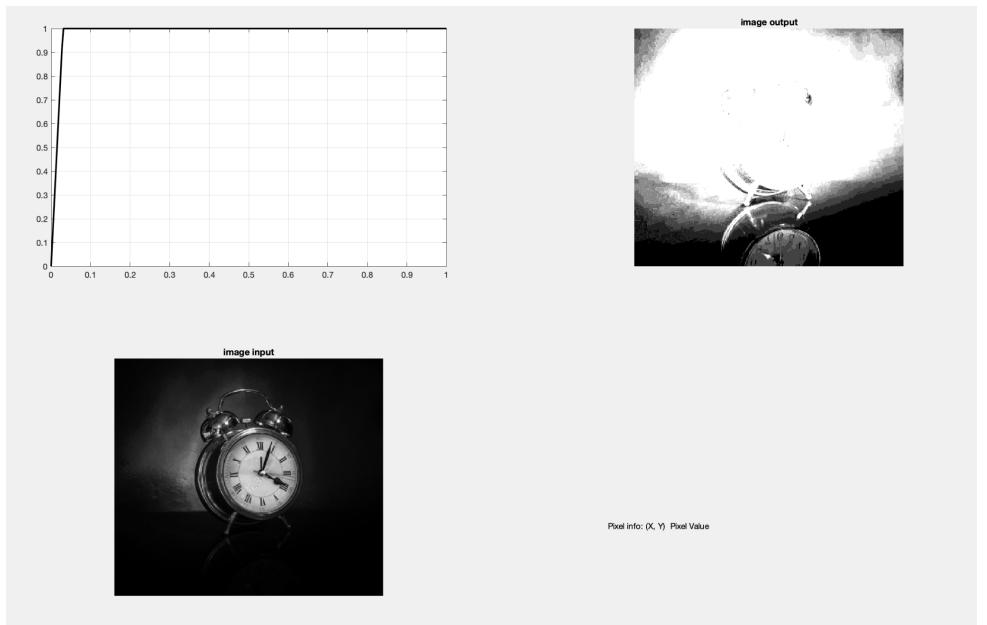
c) A white image?

Com tots els valors de la imatge estan concentrats al final de l'histograma, tota la informació rellevant esta “amagada” en els blancs de la imatge. Al clipear els valors inferiors a 0.98 a 0 aconseguim eliminar els blancs innecessaris de la imatge i podem veure la seva informació i sabem que la Lasagne Bolognaise ha costat 9,5. La comanda utilitzada és:

```
ima5_c=Image_Contrast(ima5, [0.98 1.0], [0.0 1.0]);
```



d) The image was manipulated

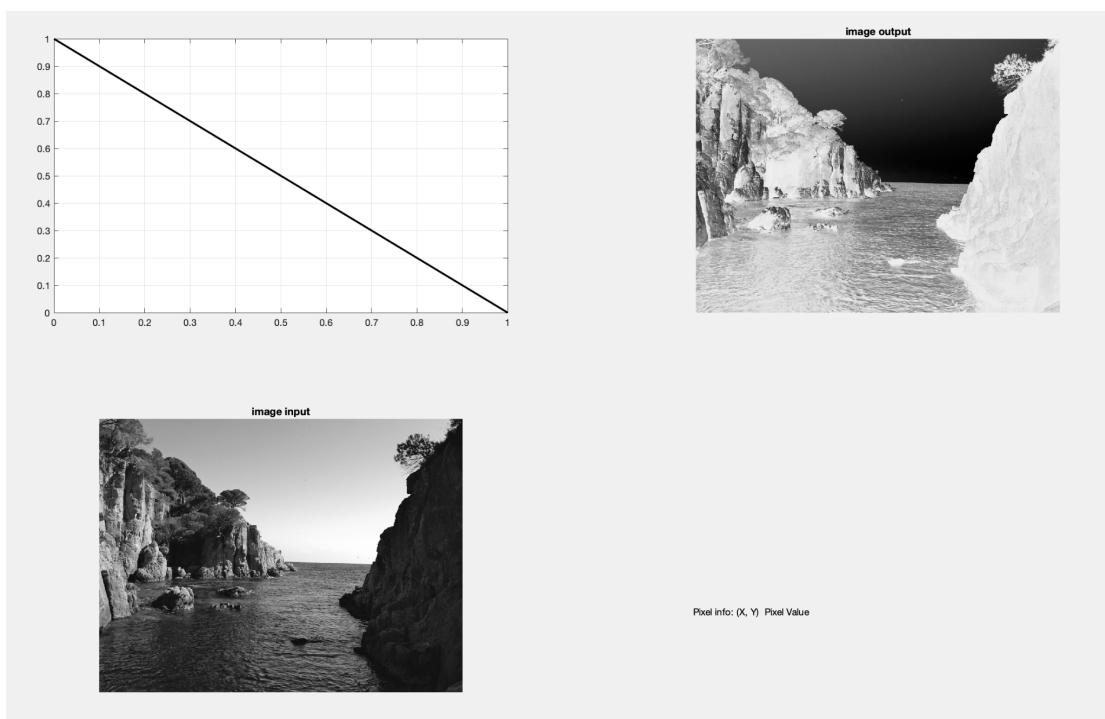


Per descobrir que la imatge estava manipulada ens hem de fixar en el fons, el qual estava molt negre i ocultava informació. Si expandim els primers valors de l'histograma (els més foscos) descobrim que hi ha l'ombra del rellotge, on es veu l'hora que era de veritat al fer-se la fotografia. L'hora és les 8.30.

La comanda utilitzada és:

```
ima2_c=Image_Contrast(ima2, [0.0 0.03], [0.0 1.0]);
```

e) Negative image



Per invertir el rang de valors de la imatge mapegem de 0 a 1 i de 1 a 0, d'aquesta manera creem el negatiu de la imatge. La comanda utilitzada és:

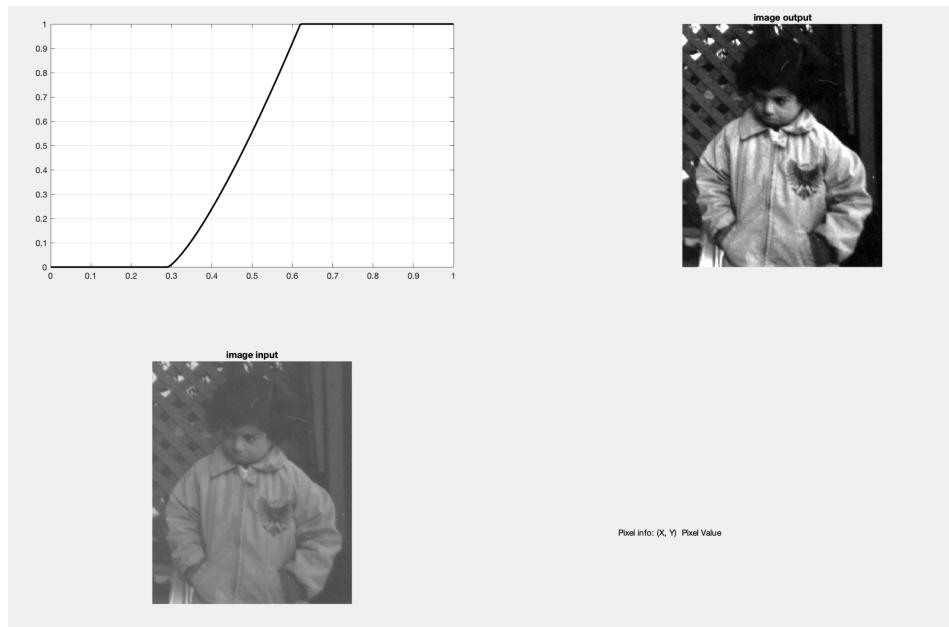
```
ima1_c=Image_Contrast(ima1,[0.0 1.0],[1.0 0.0]);
```

f) Non-linear mapping

Ara editem la imatge amb un altre tipus de mapejat, afegint l'argument gamma. Hem escollit un valor per sobre de 1 (1.3) i un per sota de 1 (0.7).

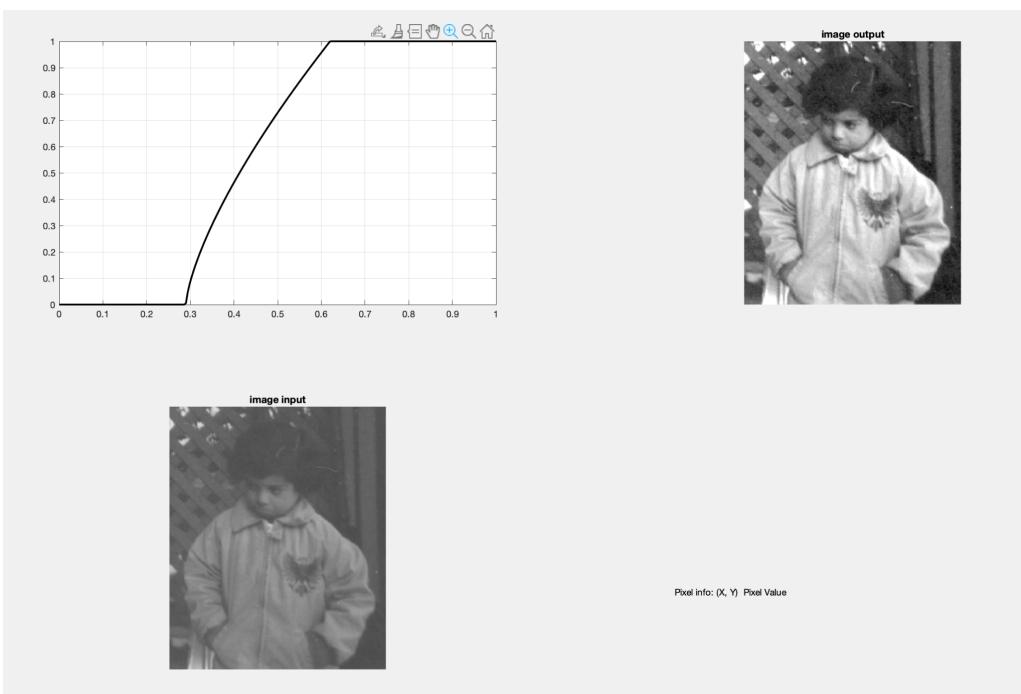
Les comandes utilitzades són:

```
ima5_c=Image_Contrast(ima5,[0.29 0.62],[0.0 1.0],1.3);
```



Si utilitzem un valor de gamma per sobre de 1, expandim més els valors foscos i comprimim els clars. També observem com augmenta el contrast.

```
ima5_c=Image_Contrast(ima5,[0.29 0.62],[0.0 1.0],0.7);
```

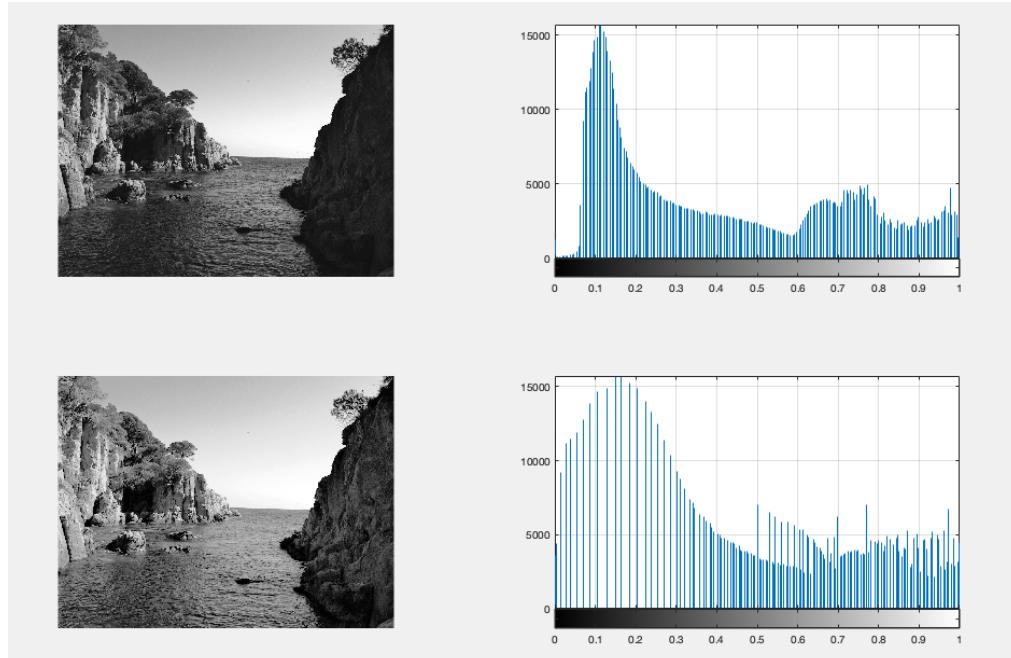


Si utilitzem un valor de gamma per sota de 1, expandim els valors clars i comprimim els foscos. Amb aquesta opció es veu millor la imatge.

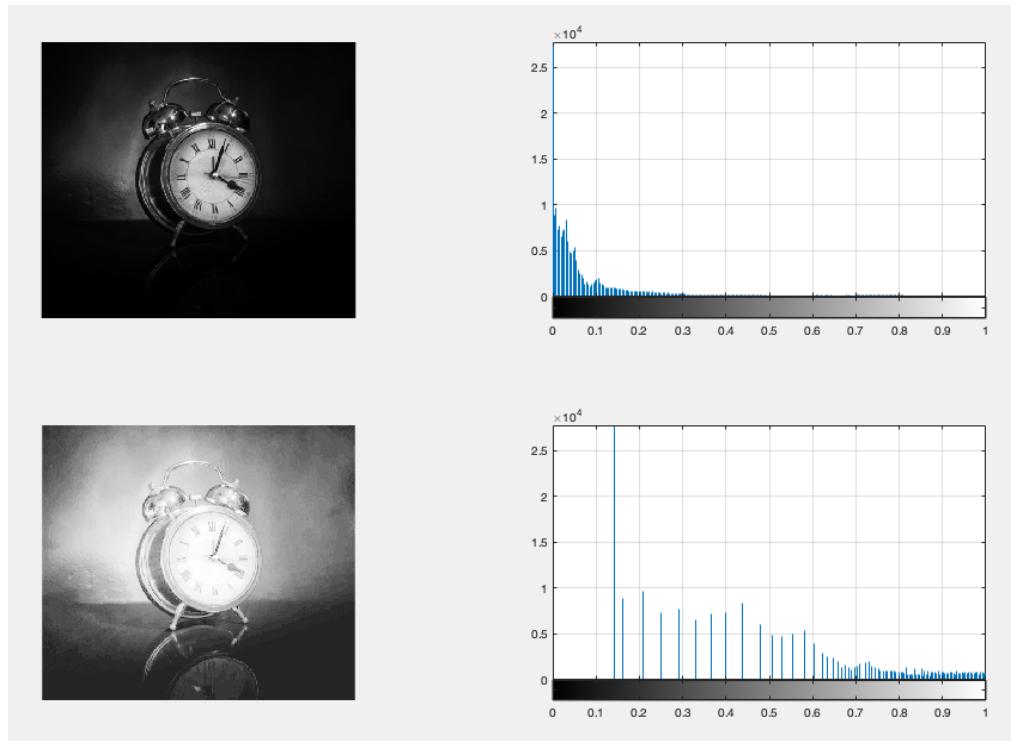
1.3. Image histogram equalization

Imatge 1:

```
ima1_eq = Image_EqHistogram(ima1);  
Image_Histogram(ima1,ima1_eq);
```



Imatge 2:



En la imatge 1 observem com amb l'equalització hi ha un aclariment general de la imatge, ja que els valors concentrats a les zones fosques s'expandeixen i es divideixen millor per l'histograma. Això també causa un increment del contrast, el qual s'aprecia en la imatge equalitzada. En la imatge 2 veiem un gran canvi entre la imatge original i l'equalitzada, això és degut a que els valors estan concentrats en un rang molt petit, concretament als foscos. Quan equalitzem la imatge hi ha una expansió del rang dinàmic, per això passen a haver moltes zones molt més clares. L'equalització és molt més perceptible en la imatge 2, degut a que el seu rang dinàmic és més petit i per tant hi ha un major canvi quan l'expandim. És en aquest cas que és interessant equalitzar una imatge, és a dir, quan el rang dinàmic de la imatge és petit. Llavors quan equalitzem la imatge s'expandeix el rang i podem observar millor els detalls de la imatge.

No és possible equalitzar perfectament una imatge, mai aconseguirem un histograma completament pla.

1.4. Uniform and optimum quantizer

- Quantització uniforme de la imatge 1 amb $N=\{256,50,10\}$:



$N=10$:

$\text{mse} = 9.4152\text{e-}04$

$\text{psnr} = 30.2617$

```
[imal_quant10, mse, psnr] = Image_UniformQ(imal,10);
```

N=50:

```
mse = 3.5083e-05  
psnr = 44.5490  
[imal_quant50, mse, psnr]= Image_UniformQ(imal,50);
```

N=256:

```
mse = 0  
psnr = Inf  
[imal_quant256, mse, psnr]= Image_UniformQ(imal,256);
```

Quan quantitzem la imatge amb valors baixos de N, observem que apareixen uns contorns que en la realitat no estan allà (“false contours”). Aquest efecte es veu sobretot en el cel, que hi ha unes línies que no estan en la imatge original. Amb N=10 s’aprecien molt aquestes línies i amb N=50 encara les podem veure una mica, però molt menys marcades. Quants menys nivells utilitzem, més contorns falsos apareixen.

- Nombre de nivells N necessaris per aconseguir “quantització transparent”?

Provem diferents valors de N fins aconseguir que no es vegin falsos contorns:



N=32

```
mse = 8.7119e-05  
psnr = 40.5989
```



$\text{mse} = 1.9897\text{e-}05$

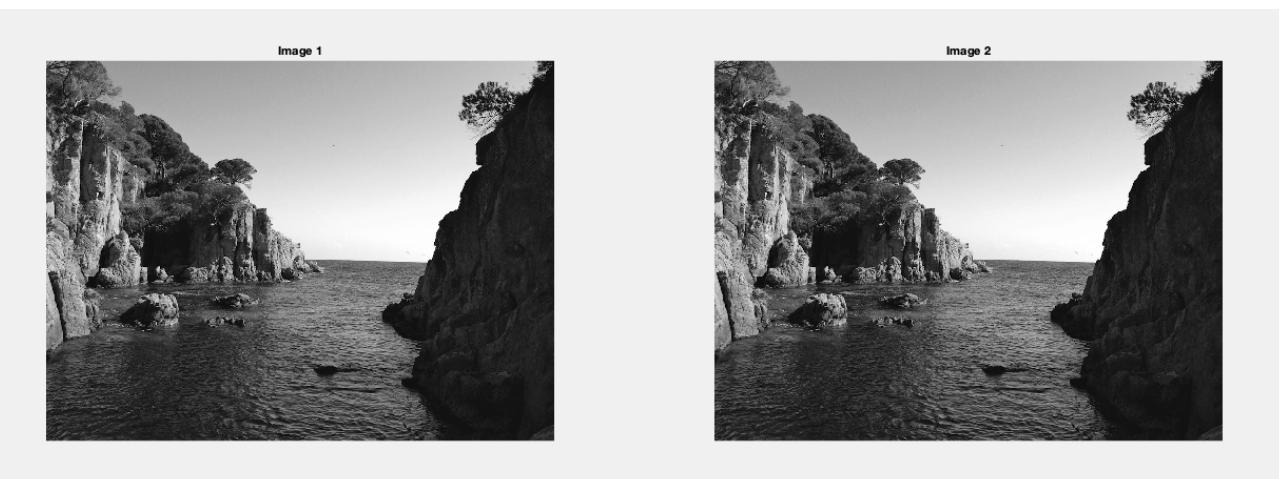
$\text{psnr} = 47.0121$



$N=100$

$\text{mse} = 8.6890\text{e-}06$

$\text{psnr} = 50.6103$



N=128

mse = 5.3228e-06

psnr = 52.7386

Hem quantitzat la imatge amb els valors de N={32,64,100,128} i hem arribat a la conclusió que a partir del 128 (7 bits) els efectes de la quantització són imperceptibles, ja que no veiem cap diferència entre la imatge original i la quantitzada. A partir de N=64 (6 bits) desapareixen pràcticament els falsos contorns, però hem considerat que és en N=128 quan s'aconsegueix que sigui “transparent”. Observant els valors del mse i del psnr, concluïm que la qualitat perceptual de la imatge cau quan el mse augmenta (ja que hi ha més error) o quan el psnr decreix (implica més soroll en la senyal).

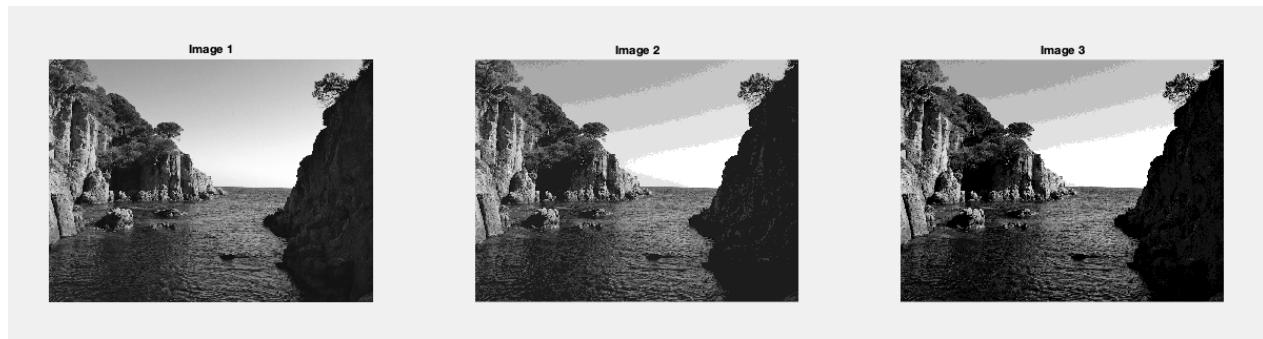
- Comparació quantització uniforme i quantització òptima:

Quantitzem les imatges 1 i 5 de manera uniforme i amb l'algorisme Max Lloyd òptim i les comparem, ho fem pels valors de N={256,50,20}.

Imatge 1:

N = 10

```
[ima1_uniform, mse1u, psnr1u]= Image_UniformQ(ima1,10);  
[ima1_maxlloyd, mse1ml, psnr1ml]= Image_MaxLloydQ(ima1,10);
```



Quantització uniforme:

mse = 9.4152e-04

psnr = 30.2617

Quantització amb Max Lloyd:

mse = 6.6349e-04

psnr = 31.7816

N = 50

```
[ima1_uniform, mse1u, psnr1u]= Image_UniformQ(ima1,50);
```

```
[ima1_maxlloyd, mse1ml, psnr1ml]= Image_MaxLloydQ(ima1,50);
```



Quantització uniforme:

mse = 3.5083e-05

psnr = 44.5490

Quantització amb Max Lloyd:

mse = 2.9376e-05

psnr = 45.3201

N = 256

```
[ima1_uniform, mse1u, psnr1u]= Image_UniformQ(ima1,256);
```

```
[ima1_maxlloyd, mse1ml, psnr1ml]= Image_MaxLloydQ(ima1,256);
```



Quantització uniforme:

mse = 0

psnr = Inf

Quantització amb Max Lloyd:

mse = 2.4216e-05

psnr = 46.1590

Imatge 5:

N=10

```
[ima5_uniform, mse5u, psnr5u]= Image_UniformQ(ima5,10);  
[ima5_maxlloyd, mse5ml, psnr5ml]= Image_MaxLloydQ(ima5,10);
```



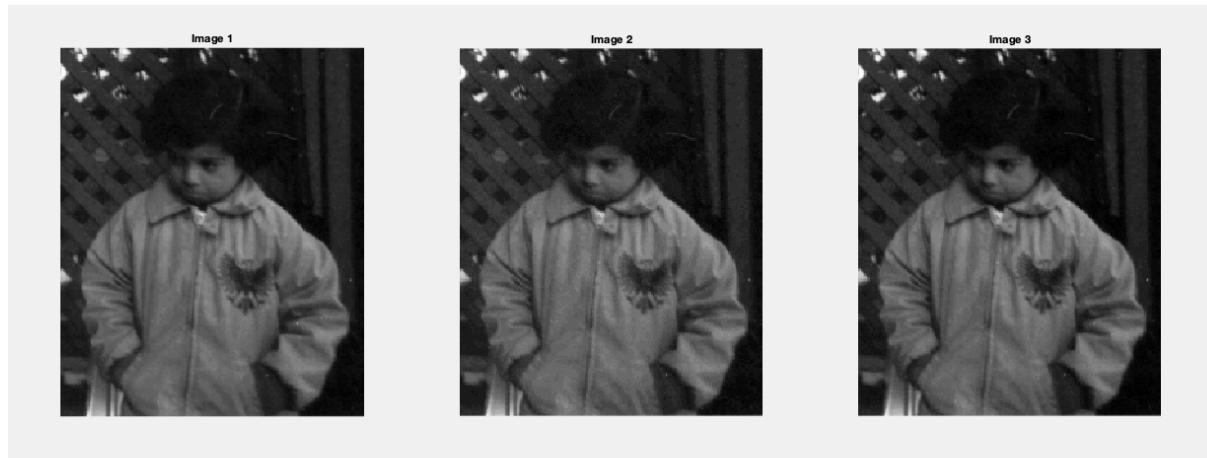
psnr = 29.9179

Quantització amb Max Lloyd:

mse = 1.6920e-04

psnr = 37.7159

N=50



Quantització uniforme:

mse = 3.5656e-05

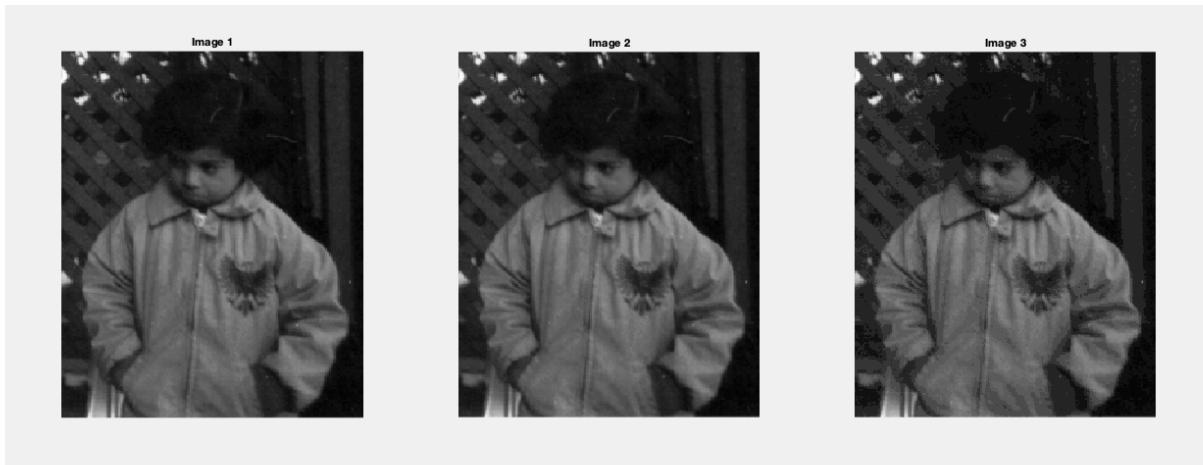
psnr = 44.4787

Quantització amb Max Lloyd:

mse = 8.2585e-06

psnr = 50.8310

N=256



Quantització uniforme:

$\text{mse} = 0$

$\text{psnr} = \text{Inf}$

Quantització amb Max Lloyd:

$\text{mse} = 8.3632\text{e-}05$

$\text{psnr} = 40.7763$

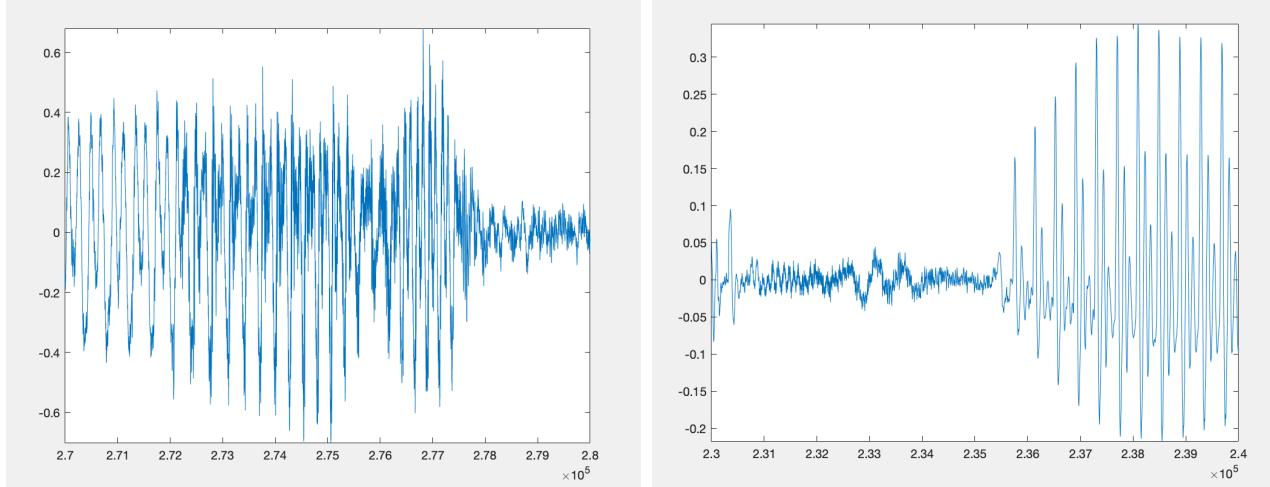
Observant els resultats de les quantitzacions uniformes i òptimes podem dir que en general la principal diferència a simple vista sembla ser que amb l'algorisme de Max-Lloyd òptim els efectes de la quantització són menors i menys visibles, encara que són bastant semblants els dos quantitzadors. Veiem com menor sigui la N, més diferència hi ha entre els quantitzadors (amb N=10 apreciem un gran canvi).

Si ens fixem en les mesures de qualitat, concluïm que el quantitzador òptim té un mse més petit i un psnr major (és a dir, una millor quantització) pels valors petits de N, però per valors grans és el quantitzador uniforme el que funciona millor. Amb N=256, el quantitzador uniforme no produeix cap error, mentre que el quantitzador òptim segueix tenint error.

Hi ha un major increment del psnr entre el quantitzador uniforme i l'òptim en la imatge 5. Això ho podem explicar amb el rang dinàmic de la imatge, si mirem l'histograma veiem que és un rang petit, per tant el quantitzador uniforme no funciona tant bé i hi ha una major diferència amb l'òptim. En canvi, en la imatge 1 l'histograma s'assembla més a una distribució uniforme per tot el rang [0,1] i per tant el quantitzador uniforme funciona millor i té menys diferències amb l'òptim.

2. MEMORYLESS PROCESSES OF AUDIO SIGNALS

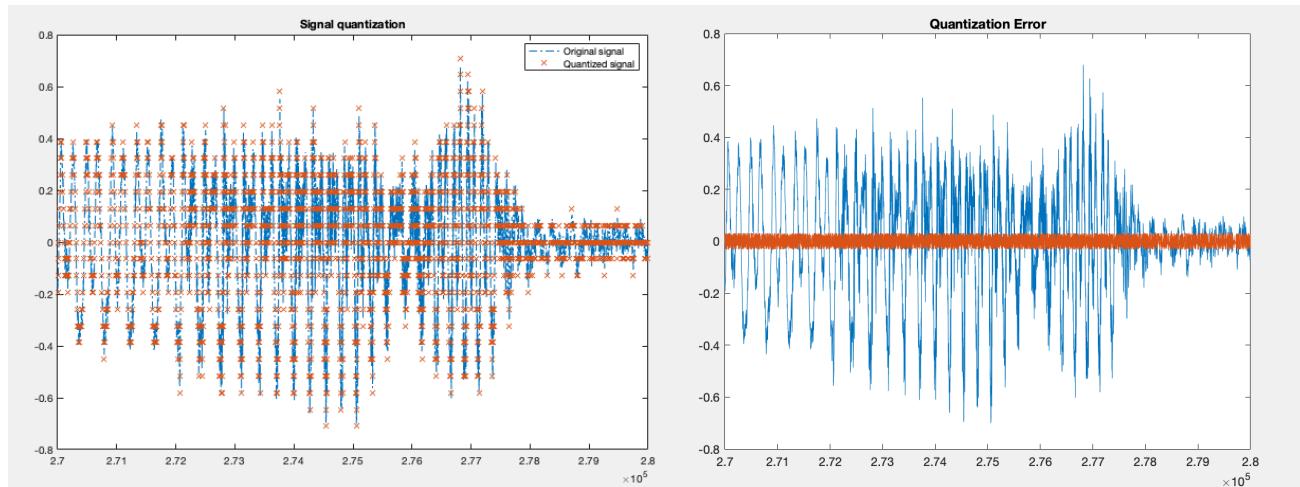
Senyals Audio 1 i Audio 2:



2.1. Uniform quantization of audio signals

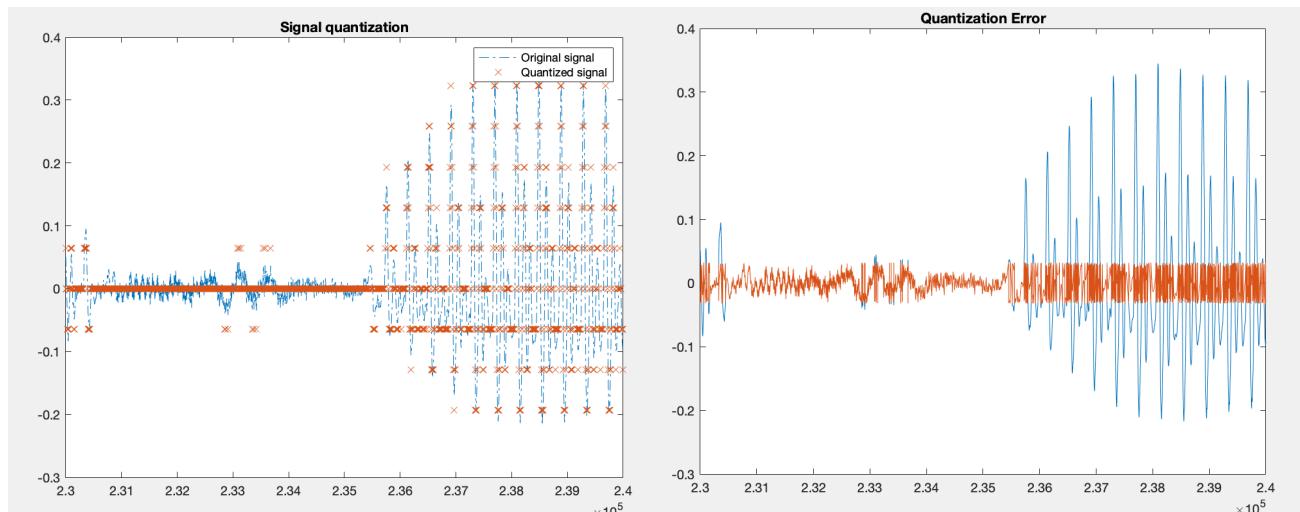
Àudio 1:

```
[x1_quant5, SNR, Qerr] = Audio_UniformQ(x1,5,48000); SNR = 19.1752
```



Àudio 2:

```
[x2_quant5, SNR, Qerr] = Audio_UniformQ(x2,5,48000); SNR = 16.0121
```



Per representar l'interval $[-1,1]$ s'utilitzen 31 nivells de quantització, això equival a $2^5 - 1$. Això és perquè la quantització la hem fet amb 5 bits, i hem utilitzat mid-tread, per tant tenim un nivell a zero i a canvi perdem un nivell (tenim 31 nivells i no 32).

- Rang dinàmic de l'error de quantització:

El rang dinàmic de l'error de quantització dels àudios és $[-0.0323, 0.0323]$. Teòricament, el rang dinàmic de l'error és $[-\Delta/2, \Delta/2]$, on Δ és el pas de quantització, és a dir, l'amplada dels nivells. Com tenim 31 nivells a l'interval $[-1,1]$, dividim $2/31$ i ens dona $\Delta=0.0645$. Si dividim $\Delta/2$, ens dona exactament els límits del nostre rang dinàmic. Per tant, coincideix el valor teòric amb l'experimental.

- Nivell de representació de les observacions properes a 0:

Utilitzem una quantització mid-tread, el que significa que tenim un nivell de representació a 0. Per tant, les observacions properes a 0 les representarem com 0.

- Distorsió i qualitat de les senyals d'àudio:

Al fer la quantització del Audio1.wav, l'audio perd qualitat però no és molest, s'escolta bé encara mentre que al fer el mateix per l'Audio2.wav es crea un soroll al fons que distorsiona l'àudio i a moments el fa intel·ligible. En el primer audio el SNR = 19,1752 mentre que al segon SNR = 16,0121 el que fa pensar que el 2 té més distorsió perquè té el SNR més baix i que per tant la qualitat depèn d'aquest factor.

Observant els histogrames podem veure com en el Audio1.wav la senyal quantitzada es reparteix per tots els valors de la original mentre que en el Audio2.wav tots els valors que eren propers a 0 passen a ser 0 el que fa que es creï aquesta distorsió en el audio resultant.

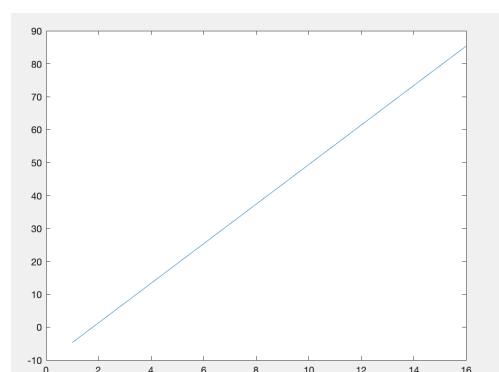
- SNR experimental i teòric:

Primer calculem el SNR teòric per Nb en l'interval d'1 a 16 amb la fórmula:

$$SNR_t(N_b) = 6 \cdot N_b + 4.77 - 20 \log_{10}(1/\sigma_x)$$

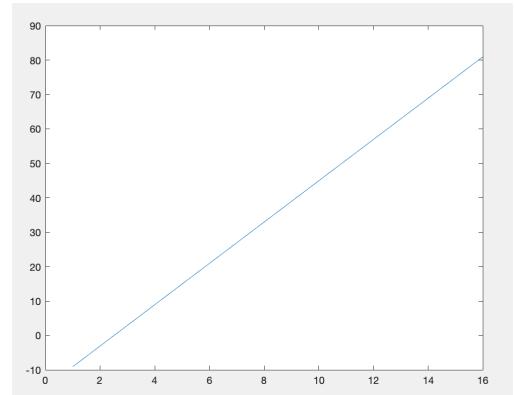
AUDIO 1:

```
N=1;
SNRt1=[];
while N<=16
    SNRt1(N)=6*N+4.77-20*log10(1/std2(x1));
    N=N+1;
end
plot(SNRt);
```



AUDIO 2:

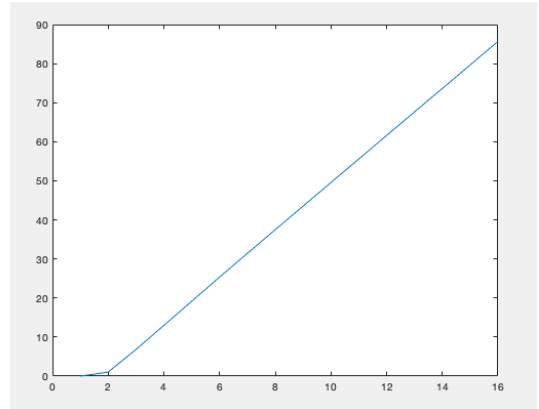
```
N=1;  
SNRt=[];  
while N<=16  
    SNRt(N)=6*N+4.77-20*log10(1/std2(x2));  
    N=N+1;  
end  
plot(SNRt);
```



Ara calculem el SNR experimental per Nb en l'interval d'1 a 16:

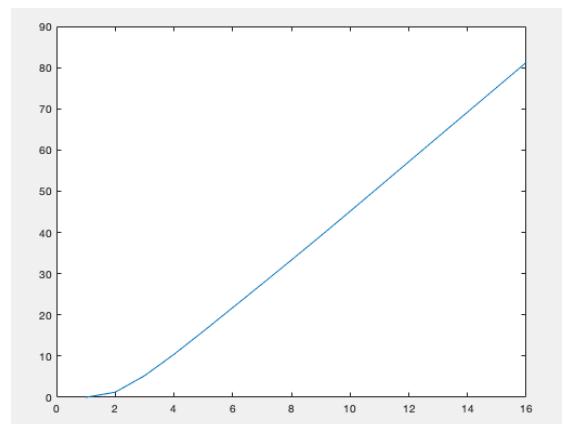
AUDIO 1:

```
N=1;  
SNRe1=[];  
while N<=16  
    [x1_quantN, SNRN, QerrN] = Audio_UniformQ(x1,N,48000);  
    SNRe1(N)=SNRN;  
    N=N+1;  
end  
plot(SNRe1);
```



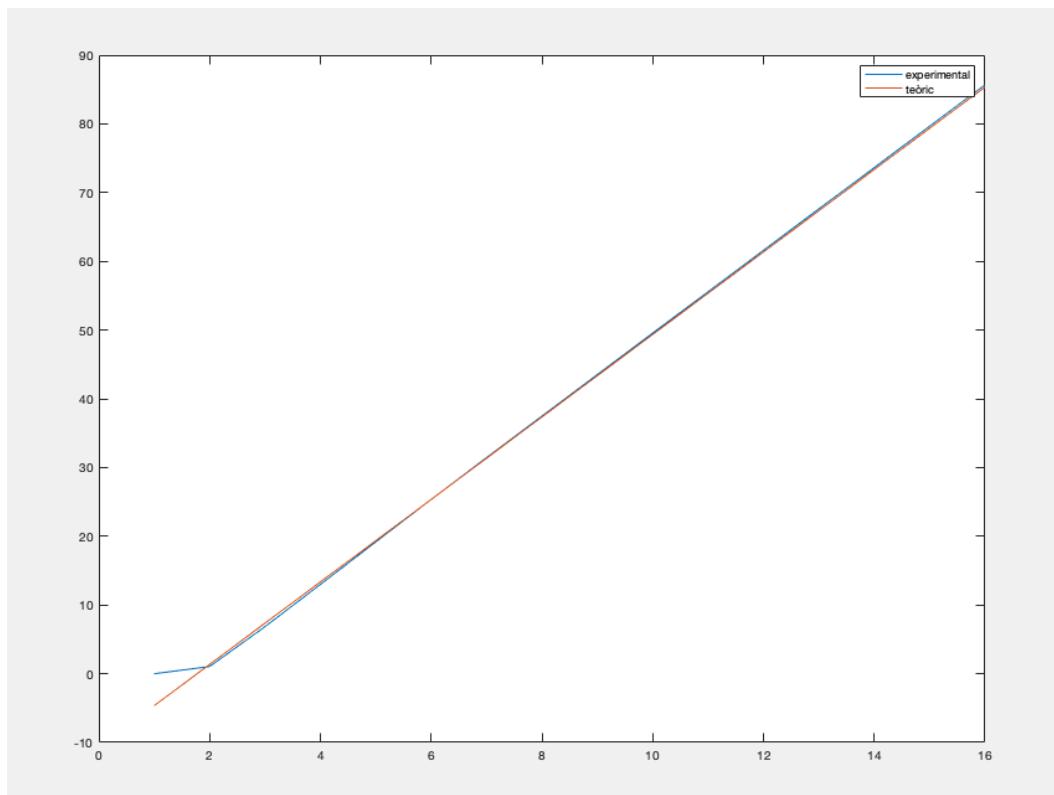
AUDIO 2:

```
N=1;  
SNRe2=[];  
while N<=16  
    [x1_quantN, SNRN, QerrN] = Audio_UniformQ(x2,N,48000);  
    SNRe2(N)=SNRN;  
    N=N+1  
end  
plot(SNRe2);
```

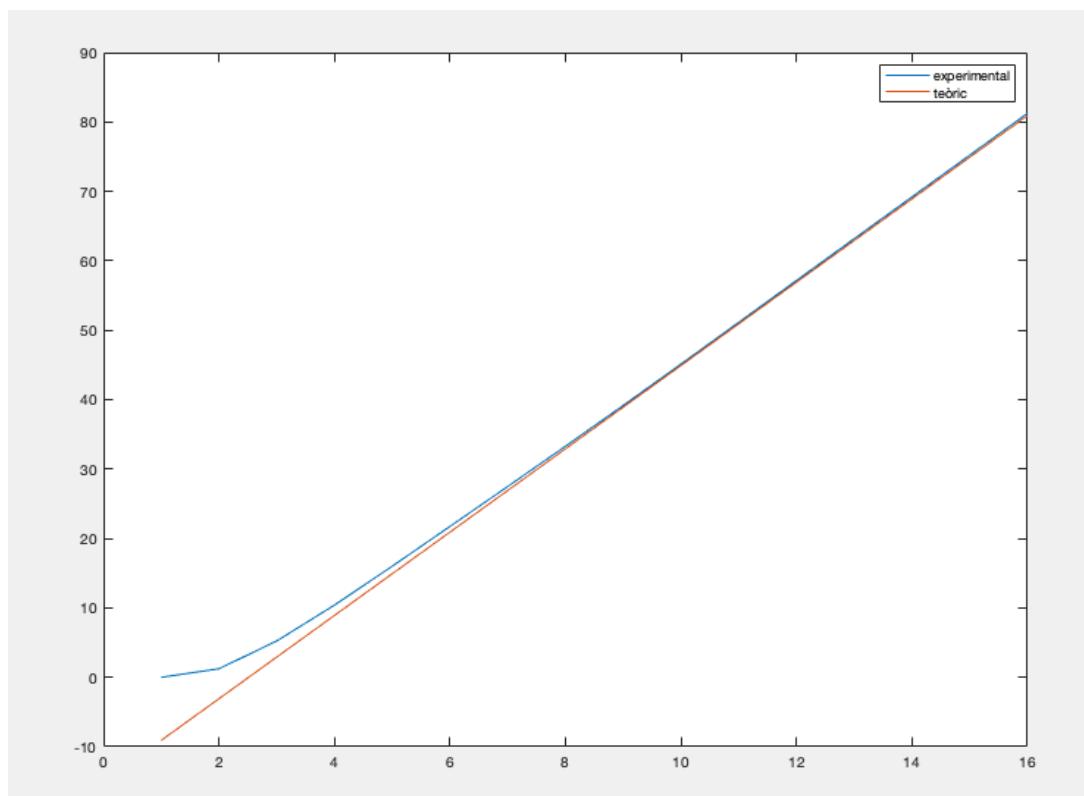


Solapament gràfiques SNR teòric i SNR experimental:

AUDIO 1:



AUDIO 2:

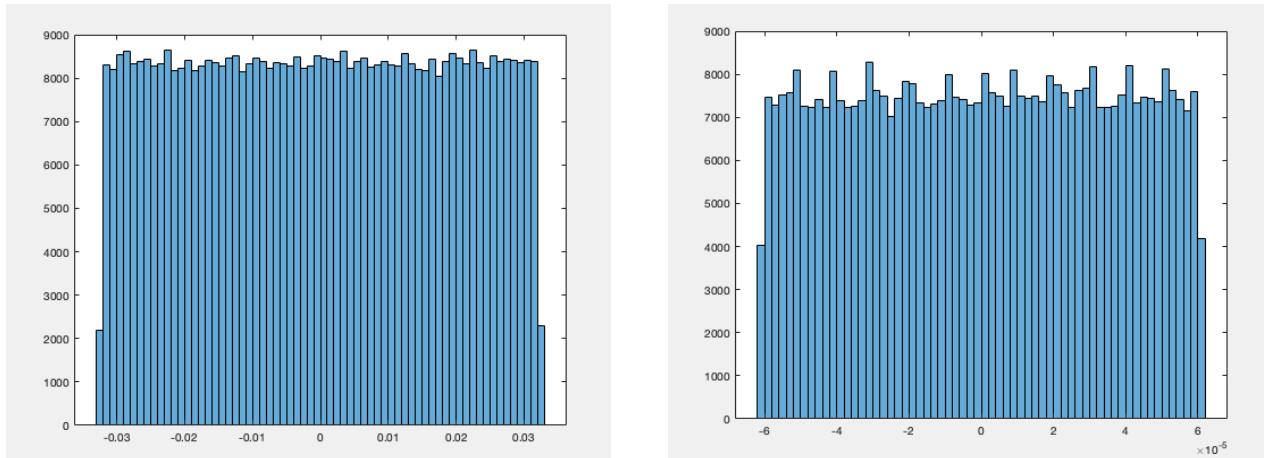


- Diferències resultat teòric i experimental:

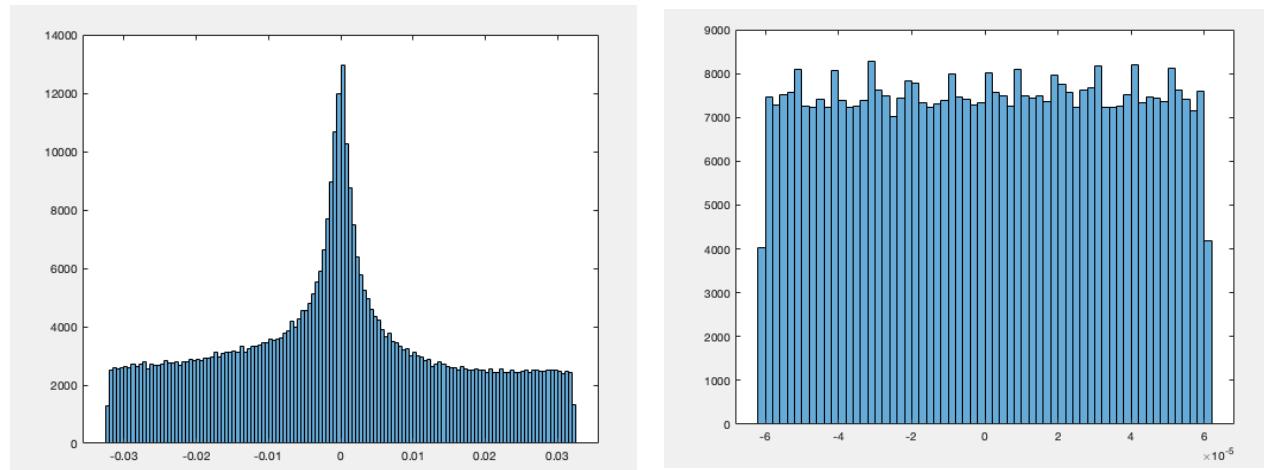
La principal diferència entre el SNR teòric i l'experimental es troba en els nivells baixos de N_b , on veiem una distorsió clara entre les dues gràfiques. En l'àudio 1, les funcions teòrica i experimental es solapen en $N_b=2$, i a partir d'aquell valor es mantenen iguals. D'altra banda, en l'àudio 2 això no passa fins als nivells de bits 7/8.

Histogrames de l'error de quantització per $N_b=5$ i $N_b=14$:

AUDIO 1:



AUDIO 2:



Els histogrames d'error també ens ajuden a comparar els resultats teòrics amb els experimentals. Teòricament, s'assumeix que l'error de quantització segueix una pdf uniforme. En la pràctica, veiem com en el audio 2 amb els nivells baixos de N_b , l'error té una pdf gaussiana centrada en el 0, el que significa que l'error al voltant del 0 és molt major. Quan anem augmentat el nombre de bits amb els que quantitzem, la pdf de l'error es va aproximant a una uniforme, com en el cas teòric.

Per tant, arribem a la conclusió de que els resultats teòrics i experimentals aproximadament coincideixen per valors elevats de N_b , però hi ha una distorsió pels valors baixos.

- Percepció de la distorsió:

Si anem baixant el nombre de bits amb els que quantitzem, per l'Audio 1 la distorsió es comença a percebre amb $N_b=5$, amb $SNR=19.1752$. Considerem que la distorsió és completament inacceptabile amb $N_b=2$, amb $SNR=1.0422$.

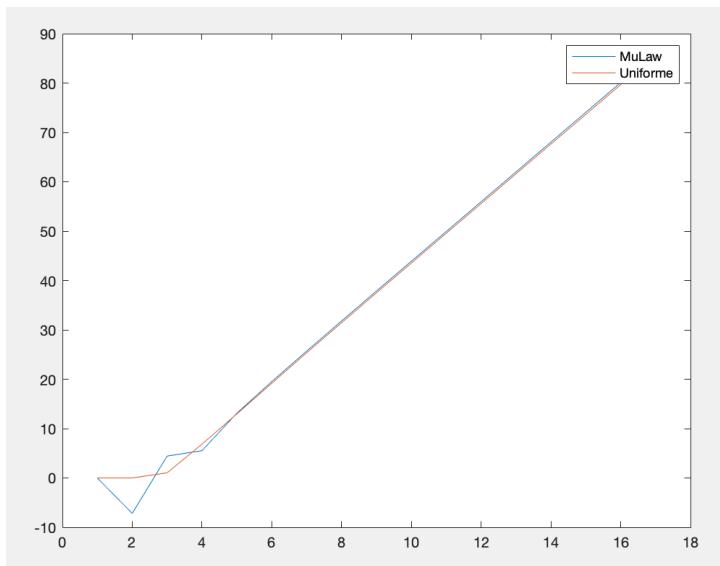
Per l'Audio 2 la distorsió es pot percebre quan quantitzem amb $N_b=8$, amb $SNR=33.3000$. I considerem que és inacceptabile quan $N_b=3$, amb $SNR=5.2381$.

2.2. μ -law quantization of speech signals

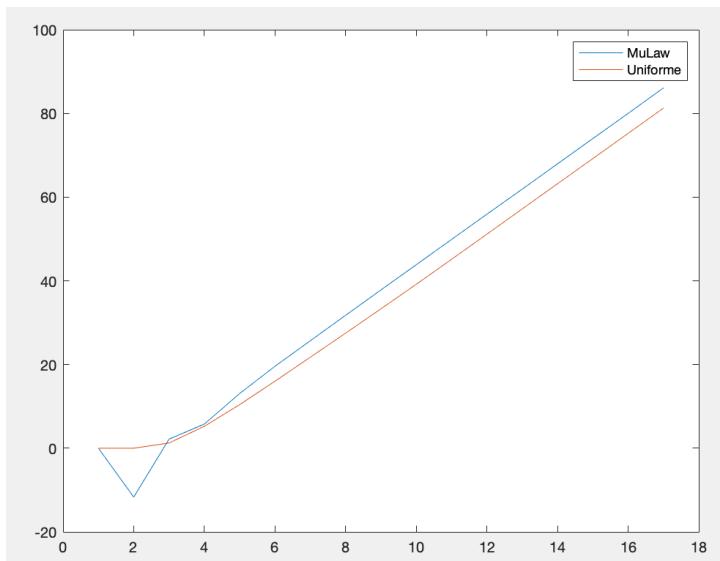
Ara quantitzem els audios amb un quantitzador μ -law:

```
[x_quantN, SNR, Qerr] = Audio_MuLawQ (x,Nbits,Fs)
```

AUDIO 1:



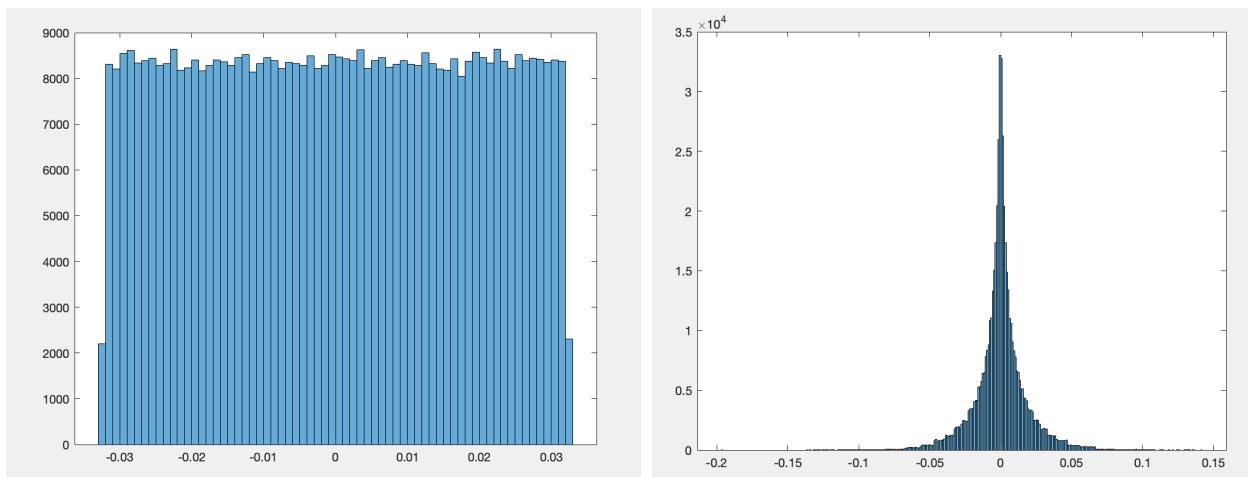
AUDIO 2:



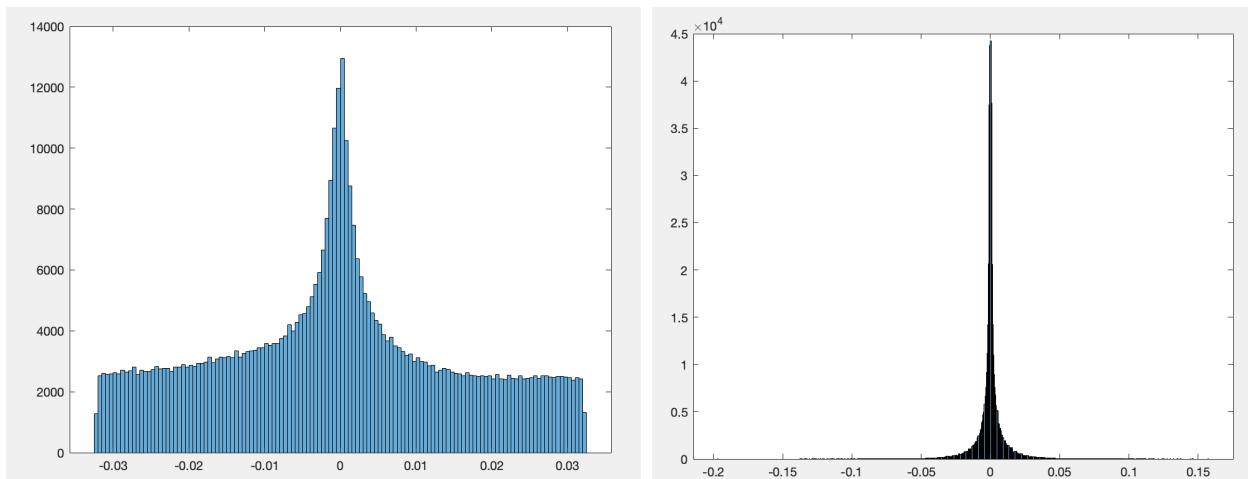
En l'Audio 1, la quantització μ -law comença a ser millor que la uniforme a partir de Nbits=5. En els nombres de bits baixos no queda molt clar quin és millor, però a partir del 5 veiem com la gràfica va una mica per sobre. En l'Audio 2 aquesta millora s'aprecia més, observem com a partir de Nbits=4, la quantització μ -law té una millora considerable respecte la uniforme, com es pot observar a l'histograma de l'error, on tot i que te un pic molt gran a 0, la resta del histograma és casi nul.

Si ens fixem només en el SNR les millors són sutils, però si analitzem els histogrames de l'error de quantificació per Nbits=5 dels 2 mètodes, podem observar com clarament la quantificació uniforme té un error constant mentre que la μ -law té un error de quantització casi nul excepte al voltant del 0.

AUDIO 1:

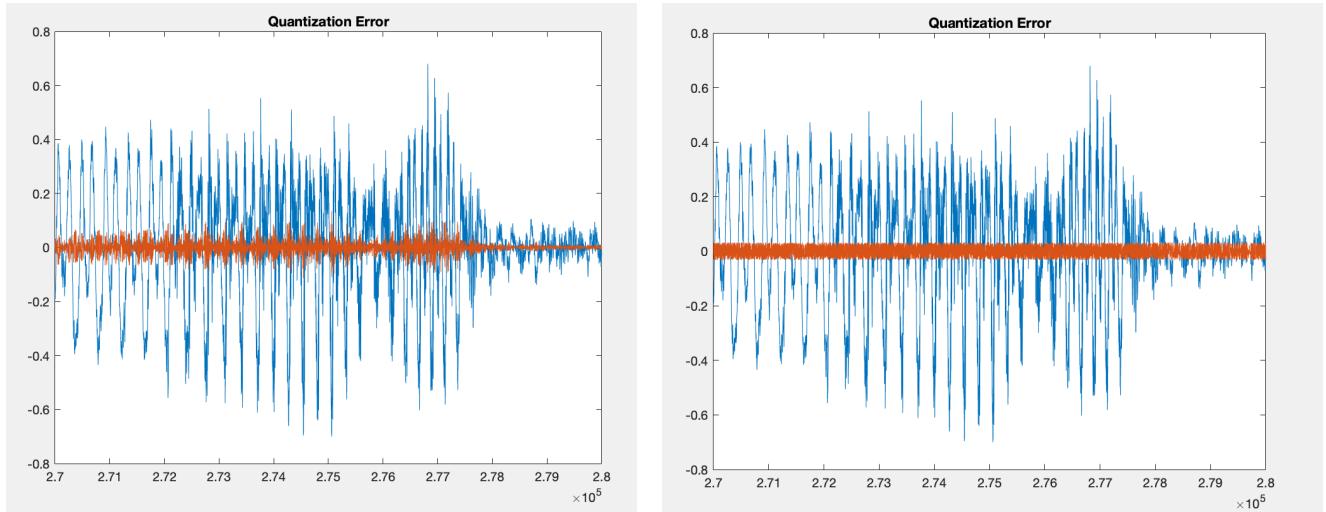


AUDIO 2:

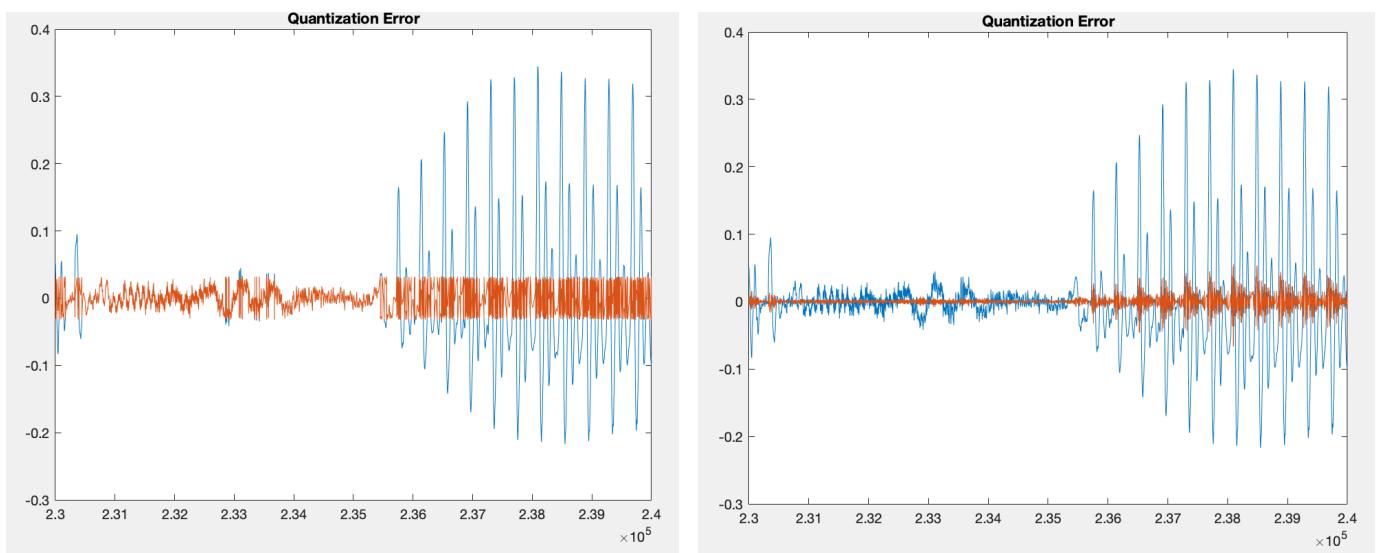


Quan escoltem les senyals quantitzades de les dues maneres amb Nbits=5, tot i que l'audio s'entén amb les dues quantitzacions, la uniforme té un soroll de fons constant mentre que la es sent neta.

AUDIO 1:



AUDIO 2:



Comparant l'error de quantització amb els dos mètodes diferents, veiem que amb el μ -law l'error té un rang dinàmic més petit i una potència menor, per tant, en general, és un millor mètode de quantització.