Πανεπιστήμιο Πατρών

Τμήμα Μηχ. Η/Υ & Πληροφορικής

Ψ Η Φ Ι Α Κ Ε Σ
Τ Η Λ Ε Π Ι Κ Ο Ι Ν Ω Ν Ι ΕΣ

Ακαδημαϊκό Έτος 2015-2016
1η Εργαστηριακή Άσκηση
Ερώτημα 2 – Κωδικοποίηση PCM

ΟΝΟΜΑΤΕΠΩΝΥΜΟ: ΚΑΨΑΛΗΣ ΠΑΝΑΓΙΩΤΗΣ

ΑΜ:4999

ΖΗΤΟΥΜΕΝΟ 1

a)

Σε αυτό το ερώτημα υπολογίζουμε πειραματικά την τιμή του SQNR της μέσης παραμόρφωσης μας ζητείται να συγκρίνουμε την πειραματική τιμή της μέσης παραμόρφωσης και την θεωρητική. Ο πειραματικός υπολογισμός του SQNR γίνεται με την χρήση της εντολής

`SQNR =10*log10(mean(xq.^2)/D);`

Δηλαδή υπολογίζουμε τα πηλίκο των κέντρων του κωδικοποιημένου διανύσματος εξόδου xq στο τετράγωνο με την μέση παραμόρφωση D. Το xq είναι έξοδος της συνάρτησης **`my_quantizer`** η οποία υλοποιεί τον ομοιόμορφο κβαντιστή , η μέση παραμόρφωση υπολογίζεται με την χρήση της εντολής **`D= mean((x - xq).^2);`**
Εκφράζεται δηλαδή ως το κέντρο του διανύσματος που προκύπτει αν από την διάνυσμα εισόδου x αφαιρέσουμε τους αντιπροσώπους που αντιστοιχούν για κάθε στοιχείο του χ, δηλαδή το διάνυσμα xq και υψώσουμε τα στοιχεία του προκύπτοντος διανύσματος στο τετράγωνο. Το SQNR για να υπολογιστεί σε db πρέπει να υπολογίσουμε το πηλίκο που προκύπτει **`(mean(xq.^2)/D)`** σε 10*log10.

Στην συνέχεια υπολογίζουμε την θεωρητική τιμή της μέσης παραμόρφωσης. Σύμφωνα με την θεωρία του PCM η μέση παραμόρφωση ισούται με την ισχύ του θορύβου κβαντισμού και υπολογίζεται από το εξής ολοκλήρωμα ∫(x-centers)PDF(x)dx όπου το διάνυσμα centers είναι η έξοδος της συνάρτησης **`my_quantizer`** και έχει ως στοιχεία τους αντιπρόσωπους κάθε περιοχής η συνάρτηση PDF(x) είναι η pdf της εκθετικής κατανομής η οποία δίνεται από την εκφώνηση. Για διευκόλυνσή δημιούργησα την συμβολική μεταβλητή z για τον υπολογισμό των ολοκληρωμάτων. Υπολογίζουμε τα εξής:

N (bits)	$SQNR_{db}$	D πειραματική	D θεωρητική
4	15.9382	0.0444	0.0466
6	16.8706	0.0360	0.0381

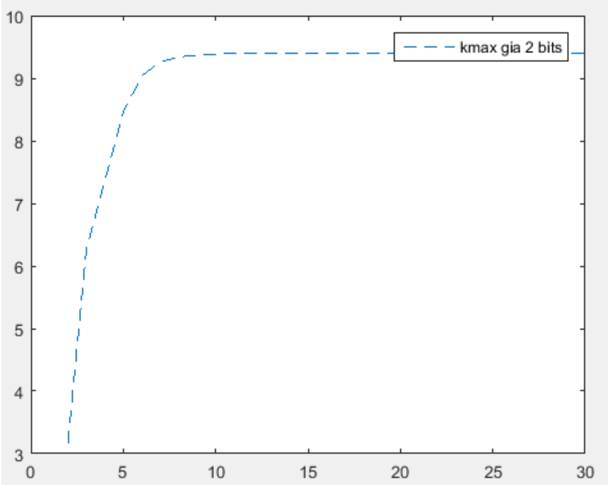
Παρατηρούμε ότι όσο αυξάνουμε τα bit της κωδικοποίησης πετυχαίνουμε όλο και καλύτερο SQNR, πιο συγκεκριμένα η ισχύς του σήματος γίνεται όλο και ισχυρότερη σε σχέση με την ισχύ του θορύβου όπως και μειώνεται και η παραμόρφωση. Καταλήγουμε λοιπόν στο συμπέρασμα ότι όσο αυξάνουμε τον αριθμό σταθμών κβάντισης έχουμε όλο και καλύτερη κωδικοποίηση πράγμα που αποδεικνύεται παρατηρώντας τον παραπάνω πίνακα για τις τιμές της μέσης παραμόρφωσης και του SQNR. Επίσης η πειραματική μέση παραμόρφωση και η θεωρητική έχουν πολύ μικρή απόκλιση.

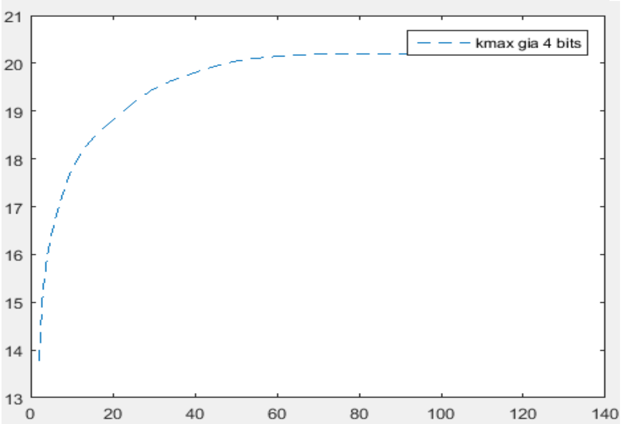
b)
Στο δεύτερο ερώτημα του πρώτου ζητήματος μας ζητείται να βρεθεί η πιθανότητα όπου η είσοδος του κβαντιστή βρίσκεται εκτός δυναμικής περιοχής. Η δυναμική περιοχή είναι το διάστημα [min_value,max_value]. Συνεπώς για τον υπολογισμό αυτής της πιθανότητας παίρνουμε το διάνυσμα εισόδου x και υπολογίζουμε το πλήθος των στοιχείων που είναι εκτός του διαστήματος [min_value,max_value] και το διαιρούμε με το πλήθος όλων των στοιχείων του διανύσματος x για να βρούμε την πιθανότητα αυτή. Υπολογίζουμε ότι:
`distortion_overload`=0.0167

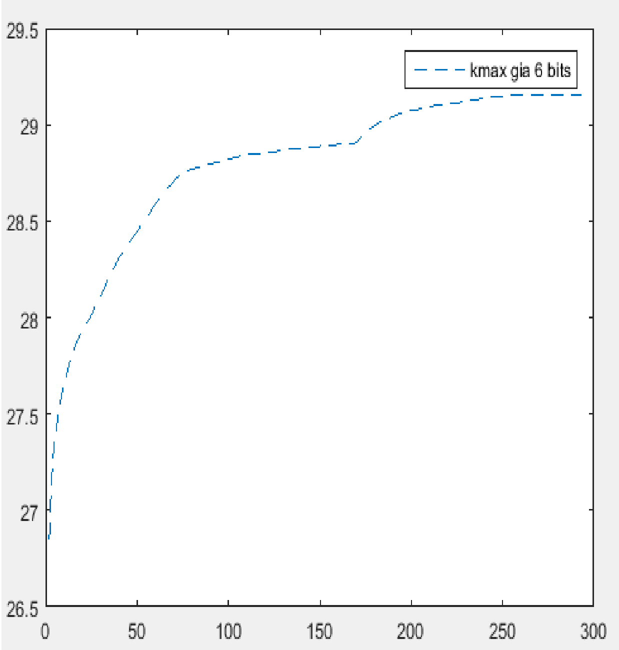
Το script του ζητήματος 1 υλοποιείται από το αρχείο **zhtoumeno1.m**

ΖΗΤΟΥΜΕΝΟ 2

a)Το script για το τρέχον ερώτημα είναι το **zhtoumeno2a.m**
Εκτελώντας το script παίρνουμε τις εξής κυματομορφές:







Παρατηρούμε ότι όσο αυξάνονται τα bits, δηλαδή αυξάνεται ο αριθμός των σταθμών κβάντισης το SQNR αυξάνεται δηλαδή ο θόρυβος μειώνεται.

b) Το script για το τρέχον ερώτημα είναι το **zhtoumeno2b.m**

Εφαρμόζοντας τον αλγόριθμο **Lloyd Max** έπειτα από Kmax επαναλήψεις υπολογίζουμε ότι το $SQNR_{db}$ έχει ως εξής:

N	$SQNR_{db}$ μετά από Kmax επαναλήψεις
2	9.4055
4	20.2002
6	29.1582

Όπως σχολιάσαμε και στο προηγούμενο ερώτημα παρατηρούμε ότι όσο αυξάνεται ο αριθμός των bit τόσο και το SQNR αυξάνεται δηλαδή έχουμε καλύτερη κωδικοποίηση και ο θόρυβος μειώνεται.

Εκτελώντας το script **zhtoumeno2b.m** χρησιμοποιούμε την συνάρτηση του **ομοιόμορφου κβαντιστή** για την πηγή Β παρατηρούμε τα εξής

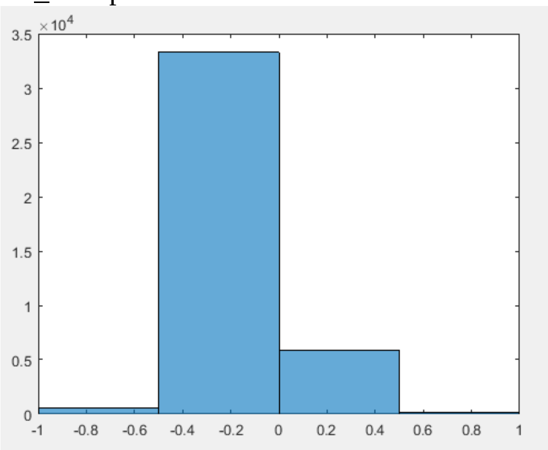
N	$SQNR_{db}$
2	5.7104
4	13.8674
6	26.8606

Αυτή την φορά καλούμαστε να σχολιάσουμε την αποδοτικότητα των δύο κβάντιστών του ομοιόμορφου και του μη-ομοιόμορφου.
Παρατηρώντας τις μετρήσεις των SQNR βλέπουμε ότι για τα αντίστοιχα bit το SQNR του μη-ομοιόμορφου είναι πιο μεγάλο σε σχέση με αυτό που προκύπτει έπειτα από την κωδικοποίηση με ομοιόμορφο κβαντιστή. Πράγμα που είναι αναμενόμενο καθώς ο αλγόριθμος Lloyd Max μας δίνει την δυνατότητα να σχεδιάσουμε τον βέλτιστο μη-ομοιόμορφο κβαντιστή για οποιαδήποτε πηγή.

c) Το script για το τρέχον ερώτημα είναι το **zhtoumeno2c.m** το οποίο καλεί την συνάρτηση για τον υπολογισμό των πιθανοτήτων εμφάνισης κάθε στάθμης την οποία την έχω ονομάσει **upologismos_pi8anothtwn.**

Για τον ομοιόμορφο κβαντιστή για N=2,4,6 bits

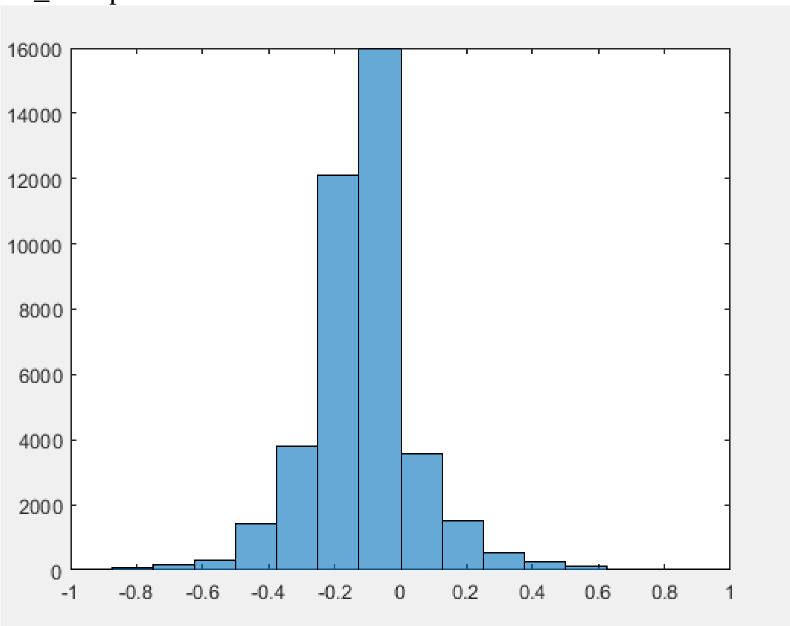
N=2
uni_theoritikes_pi8anothtes2=[0.0141, 0.8344, 0.1473,0.004]
uni_peiramatikes_pi8anothtes2=[0.0141, 0.8344,0.1473,0.0042]
uni_entropia2 = 0.7450 bits/s



Ιστόγραμμα πως κατανέμονται τα στοιχεία της εισόδου για στις περιοχές που ορίζονται από τον κβαντιστή για N=2.

N=4
uni_theoritikes_pi8anothtes4 =[0.0007,0.0021,0.0043,0.0071, 0.0359,0.0955,0.3032,0.3997 ,0.0894,0.0383,0.0133,0.0063]

uni_peiramatikes_pi8anothtes4 =[0.0007,0.0021,0.0043,0.0071,0.0359,0.0955,0.3032,0.3997 ,0.0894,0.0383,0.0133,0.0063]
uni_entropia4= 2.3175 bits/s

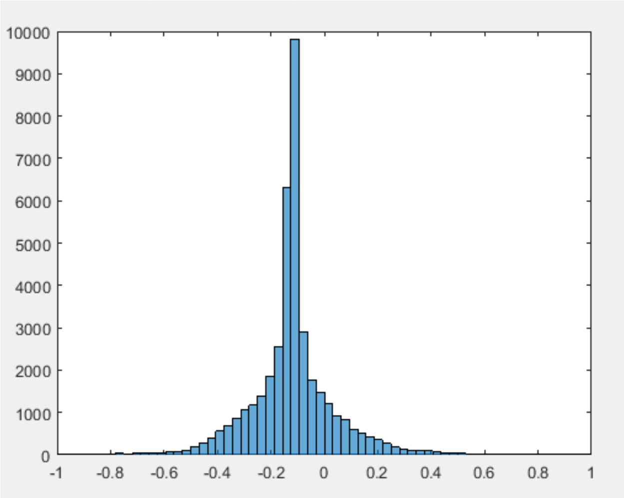


Ιστόγραμμα πως κατανέμονται τα στοιχεία της εισόδου για στις περιοχές που ορίζονται από τον κβαντιστή για N=4.

N=6
uni_entropia6= 4.0430 bits/s

```
uni_theoritikes_pi8anothtes6 =

  Columns 1 through 12

    0.0001    0.0002    0.0001    0.0004    0.0004    0.0005    0.0005    0.0008    0.0007    0.0010    0.0014    0.0012

  Columns 13 through 24

    0.0014    0.0015    0.0016    0.0026    0.0044    0.0073    0.0102    0.0141    0.0175    0.0219    0.0267    0.0294

  Columns 25 through 36

    0.0349    0.0465    0.0638    0.1580    0.2458    0.0724    0.0444    0.0371    0.0304    0.0229    0.0211    0.0150

  Columns 37 through 48

    0.0126    0.0103    0.0088    0.0066    0.0045    0.0035    0.0029    0.0025    0.0024    0.0018    0.0012    0.0010

  Columns 49 through 60

    0.0008    0.0006    0.0005    0.0006    0.0003    0.0002    0.0001    0.0002    0.0001    0.0002    0.0001    0.0001

  Columns 61 through 64

    0.0002    0.0001    0.0002    0.0001

>> |
```

Και τα πειραματικά δεδομένα είναι ίδια



Ιστόγραμμα πως κατανέμονται τα στοιχεία της εισόδου για στις περιοχές που ορίζονται από τον κβαντιστή για N=6.

Συμπερασματικά η πειραματική εκτίμηση γίνεται με το να βρούμε πόσα στοιχεία της εισόδου βρίσκονται στις αντίστοιχες περιοχές και το πλήθος των στοιχείων που βρίσκονται σε κάθε περιοχή το διαιρούμε με το πλήθος του διανύσματος εισόδου.

Η θεωρητική εκτίμηση γίνεται χρησιμοποιώντας ιστογράμματα. Δίνουμε ως είσοδο στην εντολή histogram το διάνυσμα εισόδου και σαν δεύτερο όρισμα τις περιοχές κβάντισης.

```
h=histogram(x,space);
theoritikes_pi8anothtes=h.Values./(length(x));
```

όπου h.Values το διάνυσμα που περιέχει το πλήθος κάθε περιοχής κβάντισης.

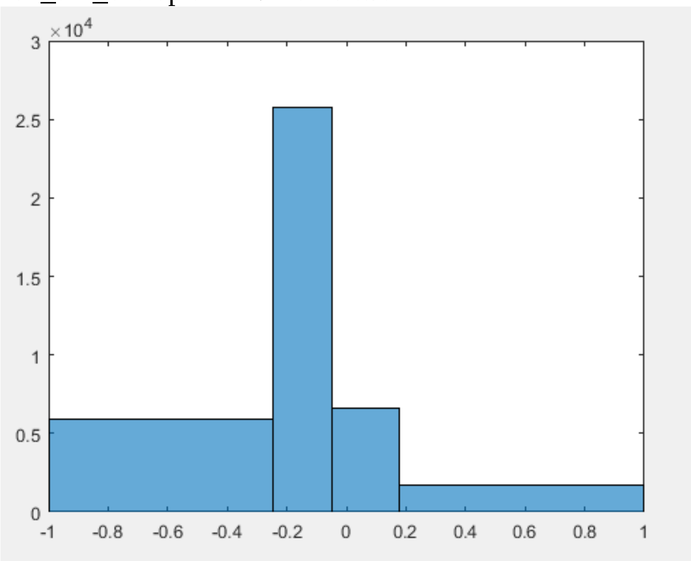
Για τον μη-ομοιόμορφο κβαντιστή για N=2,4,6 bits

N=2

non_uni_theoritikes_pi8anothtes2 =[0.1473,0.6442,0.1656,0.0429]

`non_uni_peiramatikes_pi8anothtes2=[`0.1473,0.6442,0.1656,0.0429]

non_uni_entropia2 =1.4401 bits/s



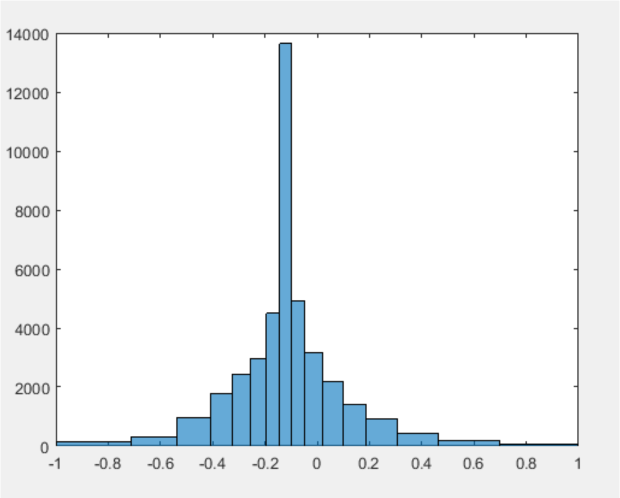
Ιστόγραμμα πως κατανέμονται τα στοιχεία της εισόδου για στις περιοχές που ορίζονται από τον μη-ομοιόμορφο κβαντιστή για N=2.

N=4

non_uni_entropia4 = 3.1206 bits/s

```
>> non_uni_theoritikes_pi8anothtes4

non_uni_theoritikes_pi8anothtes4 =

  Columns 1 through 12

    0.0036    0.0075    0.0237    0.0443    0.0609    0.0740    0.1125    0.3418    0.1227    0.0789    0.0550    0.0353

  Columns 13 through 16

    0.0231    0.0112    0.0042    0.0011
```

```
non_uni_peiramatikes_pi8anothtes4 =

    0.0036
    0.0075
    0.0237
    0.0443
    0.0609
    0.0740
    0.1125
    0.3418
    0.1227
    0.0789
    0.0550
    0.0353
    0.0231
    0.0112
    0.0042
    0.0011
```



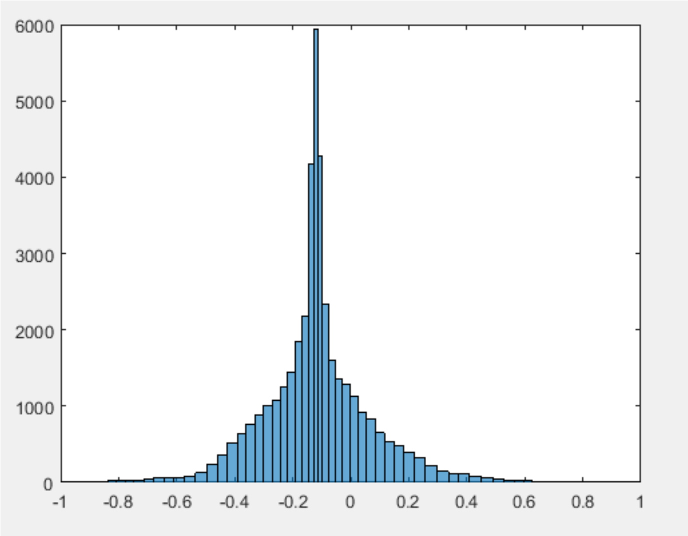
Ιστόγραμμα πως κατανέμονται τα στοιχεία της εισόδου για στις περιοχές που ορίζονται από τον μη-ομοιόμορφο κβαντιστή για N=4

. **N=6**

```
non_uni_theoritikes_pi8anothtes6 =

  Columns 1 through 12

    0.0001    0.0002    0.0001    0.0004    0.0004    0.0005    0.0005    0.0008    0.0008    0.0012    0.0015    0.0016

  Columns 13 through 24

    0.0015    0.0020    0.0031    0.0057    0.0089    0.0131    0.0158    0.0192    0.0222    0.0254    0.0269    0.0315

  Columns 25 through 36

    0.0363    0.0463    0.0546    0.1044    0.1487    0.1071    0.0587    0.0401    0.0340    0.0323    0.0282    0.0230

  Columns 37 through 48

    0.0210    0.0163    0.0135    0.0120    0.0100    0.0080    0.0053    0.0035    0.0027    0.0028    0.0022    0.0014

  Columns 49 through 60

    0.0010    0.0006    0.0005    0.0006    0.0003    0.0002    0.0001    0.0002    0.0001    0.0002    0.0001    0.0001

  Columns 61 through 64

    0.0002    0.0001    0.0002    0.0001

>>
```

non_uni_entropia6 = 4.4997 bits/s

Και τα πειραματικά δεδομένα είναι ίδια



Ιστόγραμμα πως κατανέμονται τα στοιχεία της εισόδου για στις περιοχές που ορίζονται από τον μη-ομοιόμορφο κβαντιστή για N=6

Συμπερασματικά η πειραματική εκτίμηση γίνεται με το να βρούμε πόσα στοιχεία της εισόδου βρίσκονται στις αντίστοιχες περιοχές και το πλήθος των στοιχείων που βρίσκονται σε κάθε περιοχή το διαιρούμε με το πλήθος του διανύσματος εισόδου.

Η θεωρητική εκτίμηση γίνεται χρησιμοποιώντας ιστογράμματα. Δίνουμε ως είσοδο στην εντολή histogram το διάνυσμα εισόδου και σαν δεύτερο όρισμα τις περιοχές κβάντισης. Το διάνυσμα space είναι έξοδος της συνάρτησης Lloyd_Max και είναι το διάνυσμα με τα άκρα των περιοχών κβάντισης.

```
h=histogram(x,space);
theoritikes_pi8anothtes=h.Values./(length(x));
```

όπου h.Values το διάνυσμα που περιέχει το πλήθος κάθε περιοχής κβάντισης.

Παρατηρούμε ότι και στις 2 περιπτώσεις δηλαδή στην ομοιόμορφη και στην μη-ομοιόμορφη κβάντιση οι θεωρητικές μετρήσεις και οι πειραματικές συμπίπτουν.

d) Το script για το τρέχον ερώτημα είναι το **zhtoumeno2d.m**

Για τον υπολογισμό του μέσου τετραγωνικού σφάλματος χρησιμοποιείται ο εξής τύπος:

$$\mathrm{MSE} = \frac{1}{n}\sum_{i=1}^{n}(\hat{Y_i} - Y_i)^2$$

Πιο συγκεκριμένα υπολογίζουμε το άθροισμα των διαφορών των στοιχείων των διανυσμάτων εισόδου με τα στοιχεία των στοιχείων του κωδικοποιημένων εξόδων των συναρτήσεων Lloyd_Max ή my_quantizer

Πιο συγκεκριμένα για τον **ομοιόμορφο κβαντιστή** λαμβάνουμε τα εξής αποτελέσματα

N	MSE
2	0.0192
4	0.0017
6	0.0001

Παρατηρούμε ότι το σφάλμα ολοένα και μικραίνει πράγμα που μας δείχνει ότι όσο αυξάνουν τα bits η κωδικοποιημένη έξοδος προσεγγίζει την είσοδο που δίνεται στις συναρτήσεις κβάντισης συνεπώς έχουμε ολοένα και καλύτερη κωδικοποίηση οπότε το σφάλμα ,όσο αυξάνεται ο αριθμός των bit, μειώνεται.

Για τον **μη-ομοιόμορφο κβαντιστή** λαμβάνουμε τα εξής αποτελέσματα

N	MSE
2	0.0046
4	0.0004
6	0.0000

Παρατηρούμε ότι το σφάλμα ολοένα και μικραίνει πράγμα που μας δείχνει ότι όσο αυξάνουν τα bits η κωδικοποιημένη έξοδος προσεγγίζει την είσοδο που δίνεται στις συναρτήσεις κβάντισης συνεπώς έχουμε ολοένα και καλύτερη κωδικοποίηση οπότε το σφάλμα ,όσο αυξάνεται ο αριθμός των bit, μειώνεται.

Σε σχέση με την ομοιόμορφη κβάντιση , η μη-ομοιόμορφη κβάντιση όπως παρατηρούμε και από τους παραπάνω πίνακες επιτυγχάνει μεγαλύτερη απόδοση καθώς τα αντίστοιχα σφάλματα είναι πιο μικρά σε σχέση με αυτά της ομοιόμορφης ιδιαίτερα για την περίπτωση των N=6 bits το μέσο τετραγωνικό σφάλμα είναι μηδέν.

Από την επόμενη σελίδα παρατίθενται οι κώδικες των συναρτήσεων και των ερωτημάτων.

Ομοιόμορφος Κβαντιστής (my_quantizer.m)

```
function [xq,centers] = my_quantizer(x,N,min_value,max_value)
    epipeda=2^N;
    delta=(max_value-min_value)/epipeda;
    anw_akro=[];
    katw_akro=[];
    akro=min_value;

    while akro<max_value
        katw_akro=[katw_akro akro];
        anw_akro=[anw_akro akro+delta];
        akro=akro+delta;
    end
    perioxes=cell(1,length(katw_akro));
    centers=zeros(1,length(katw_akro))
    for i=1:length(katw_akro)
        k=[katw_akro(i) anw_akro(i)];
        perioxes{i}=[k];
        centers(i)=mean(k);
    end
    xq=zeros(1,length(x));
    for i=1:length(perioxes)
        space=perioxes{i};
        for j=1:length(x)
          if (x(j)>=space(length(space)-1))&&(x(j)<space(length(space)))
            xq(j)=centers(i);
          end
          if x(j)<=min_value
               xq(j)=centers(1);
          end
           if x(j)>=max_value
             xq(j)=centers(length(centers));
           end
        end
    end

 end
```

Μη-Ομοιόμορφος Κβαντιστής (Lloyd_Max.m)

```
function [xq, centers, D,spaces] = Lloyd_Max(x, N, min_value, max_value)
    %spaces:ta akra twn perioxwn kvantishs
    epipeda=2^N;
    delta=(max_value-min_value)/epipeda;
    anw_akro=[];
    katw_akro=[];
    akro=min_value;
    %to arxiko sunolo perioxwn 8a einai omoiomorfes perioxes(isapexontes)
     while akro<max_value
            katw_akro=[katw_akro akro];
            anw_akro=[anw_akro akro+delta];
            akro=akro+delta;
     end
     D=[0];
     xq=zeros(1,length(x));
     perioxes=cell(1,length(katw_akro));
     centers=zeros(1,length(katw_akro));
     for i=1:length(katw_akro)
            perioxes{i}=[katw_akro(i) anw_akro(i)];
            centers(i)=mean(perioxes{i});
     end
     xq=zeros(1,length(x));
        for k=1:length(perioxes)
            s=perioxes{k};
            for j=1:length(x)
              if (x(j)>=s(length(s)-1))&&(x(j)<s(length(s)))
                xq(j)=centers(k);
              end
              if x(j)<=min_value
                   xq(j)=centers(1);
              end
               if x(j)>=max_value
                 xq(j)=centers(length(centers));
               end
```

```
            end
        end
      D=[D mean((x'-xq).^2)];
      i=2;
      while abs(D(i)-D(i-1))>=eps
          temp=[];
          for j=1:length(centers)-1
              temp=[temp (centers(j)+centers(j+1))/2];
          end
          spaces=[min_value,[temp],max_value]
          for p=1:length(spaces)-1
              perioxes{p}=[spaces(p) spaces(p+1)];
          end
          for j=1:length(spaces)-1
              centers(j)=mean(x(x>=spaces(j) & x<=spaces(j+1)));
          end
          xq=zeros(1,length(x));
         for k=1:length(perioxes)
             s=perioxes{k};
             for j=1:length(x)
               if (x(j)>=s(length(s)-1))&&(x(j)<s(length(s)))
                 xq(j)=centers(k);
               end
               if x(j)<=min_value
                    xq(j)=centers(1);
               end
                if x(j)>=max_value
                  xq(j)=centers(length(centers));
                end
             end
         end
         D=[D mean((x'-xq).^2)];
         i=i+1;
      end
end
```

Ζητούμενο 1 a & b (zhtoumeno1.m)

```
%dhmiourgia phghs A
M=10000;
t = (randn(M,1)+j*randn(M,1))/sqrt(2);
x= abs(t).^2;
min_value=0;
max_value=4;
N1=4;
N2=6;
[xq1,centers1] =  my_quantizer(x,N1,min_value,max_value);
[xq2,centers2] =  my_quantizer(x,N2,min_value,max_value);

%zhtoumeno 1.a
xq1=transpose(xq1);
xq2=transpose(xq2);
D1=mean((x - xq1).^2);
D2=mean((x - xq2).^2);
SQNR1=10*log10(mean(xq1.^2)/D1);
SQNR2=10*log10(mean(xq2.^2)/D2);
%arxika dhmiourgoume mia nea metavlhth, thn metavlhth z
syms z
%h phgh A akolou8ei thn ek8etikh katanomh h opoia apo to z(-oo ,0] einai
%mhden kai apo to [0,+oo) einai exp(-z)
%h 8ewrhtikh paramorfwsh upologizetai apo ta oloklhrwmata stis omoiomorfe
%perioxes twn x-xq
y=(z -centers1).^2;
theoritiko_D= zeros(length(y)+2, 1);
%epeidh h pdf einai mhden sto (-oo,0] tote kai h diamorfwsh einai mhden sto
%mhden
delta1=(max_value-min_value)/2^N1;
diasthmata=[min_value:delta1:max_value];
theoritiko_D(1)=0;
theoritiko_D(end)=int(y(end)*exp(-z), diasthmata(end), Inf);
for k = 1:length(y)
    theoritiko_D(k+1) = int(y(k)*exp(-z),diasthmata(k), diasthmata(k+1));
end
theoritiki_D=sum(theoritiko_D);
%gia ton deutero kvantisth exoume ws ekshs
y2=(z -centers2).^2;
```

```
theoritiko_D2= zeros(length(y2)+2, 1);
delta2=(max_value-min_value)/2^N2;
diasthmata2=[min_value:delta2:max_value];
theoritiko_D2(1)=0;
theoritiko_D2(end)=int(y2(end)*exp(-z), diasthmata2(end), Inf);
for j = 1:length(y2)
    theoritiko_D2(j+1) = int(y2(j)*exp(-z),diasthmata2(j), diasthmata2(j+1));
end
theoritiki_D2=sum(theoritiko_D2);
%zhtoumeno 1.b
sum=0;
for i=1:length(x)
    if x(i)<min_value || x(i)>max_value
        sum=sum+1;
    end
end

distortion_overload=sum/length(x);
```

Ζητούμενο 2

a) script zhtoumeno2a.m

```
%zhtoumeno 2.a
[y,fs]=audioread('speech.wav');
N=[2,4,6];
min_value=-1;
max_value=1;
%kanonikopoihsh ths phghs etsi wste na vrisketai sto epitrepto diasthma
%[min_value,max_value]
y=(y-min(y));
f=(min(y)+max(y))/2;
y_c=(y-f)/f;
[xq1,centers1,D1]=Lloyd_Max(y_c,N(1),min_value,max_value);
[xq2,centers2,D2]=Lloyd_Max(y_c,N(2),min_value,max_value);
[xq3,centers3,D3]=Lloyd_Max(y_c,N(3),min_value,max_value);

sqnr_2=zeros(length(D1),1);
sqnr_4=zeros(length(D2),1);
sqnr_6=zeros(length(D3),1);

for i=1:length(D1)
    sqnr_2(i)=10*log10(mean(y_c.^2)/D1(i));
end
kmax1=[1:1:length(D1)];
figure(1)
plot(kmax1,sqnr_2,'--');
legend('kmax gia 2 bits ','sqnr ');

for i=1:length(D2)
    sqnr_4(i)=10*log10(mean(y_c.^2)/D2(i));
end
kmax2=[1:1:length(D2)];
figure(2)
plot(kmax2,sqnr_4,'--');
legend('kmax gia 4 bits ','sqnr');

for i=1:length(D3)
    sqnr_6(i)=10*log10(mean(y_c.^2)/D3(i));
end
kmax3=[1:1:length(D3)];
figure(3)
plot(kmax3,sqnr_6,'--');
legend('kmax gia 6 bits','sqnr');
%gia to epomeno erwthma, sqnr epeita apo kmax epanalhpseis
max_timi2=sqnr_2(end);
max_timi4=sqnr_4(end);
max_timi6=sqnr_6(end);
```

b) script zhtoumeno2b.m

```
[y,fs]=audioread('speech.wav');
N=[2,4,6];
min_value=-1;
max_value=1;
%kanonikopoihsh ths phghs etsi wste na vrisketai sto epitrepto diasthma
%[min_value,max_value]
y=(y-min(y));
f=(min(y)+max(y))/2;
y_c=(y-f)/f;

[xq2b,centers2b] = my_quantizer(y_c,N(1),min_value,max_value);
[xq4b,centers4b] = my_quantizer(y_c,N(2),min_value,max_value);
[xq6b,centers6b] = my_quantizer(y_c,N(3),min_value,max_value);

xq2b=transpose(xq2b);
D2b=mean((y_c - xq2b).^2);
SQNR2b=10*log10(mean(xq2b.^2)/D2b);

xq4b=transpose(xq4b);
D4b=mean((y_c - xq4b).^2);
SQNR4b=10*log10(mean(xq4b.^2)/D4b);

xq6b=transpose(xq6b);
D6b=mean((y_c - xq6b).^2);
SQNR6b=10*log10(mean(xq6b.^2)/D6b);
```

c) συνάρτηση που ζητείται στο ερώτημα (upologismos_pi8anothtwn.m)

```
function [ theoritikes_pi8anothtes,peiramatikes_pi8anothtes,entropia,space] =
upologismos_pi8anothtwn( x,N,min_value,max_value,eidos_kvantisth)

    if eidos_kvantisth==1
        [xq,centers] = my_quantizer(x,N,min_value,max_value);
        sum=zeros(length(centers),1);
        for i=1:length(xq)
            for z=1:length(centers)
                if(xq(i)==centers(z))
                    sum(z)=sum(z)+1;
                end
            end
        end
        peiramatikes_pi8anothtes=sum/length(xq);
        entropia=0;
        for j=1:length(peiramatikes_pi8anothtes)

entropia=entropia+peiramatikes_pi8anothtes(j)*log2(1/peiramatikes_pi8anothtes(j));
        end
        delta=(max_value-min_value)/2^N;
        space=[min_value:delta:max_value];
        h=histogram(x,space);
        theoritikes_pi8anothtes=h.Values./(length(x));
    end
    if eidos_kvantisth==2
        [xq, centers,~, spaces]=Lloyd_Max(x, N, min_value, max_value);
        sum=zeros(length(centers),1);
        for i=1:length(xq)
            for z=1:length(centers)
                if(xq(i)==centers(z))
                    sum(z)=sum(z)+1;
                end
            end
        end
        peiramatikes_pi8anothtes=sum./length(xq);
        entropia=0;
        for j=1:length(peiramatikes_pi8anothtes)

entropia=entropia+peiramatikes_pi8anothtes(j)*log2(1/peiramatikes_pi8anothtes(j));
        end
        space=spaces;
        h=histogram(x,space);
        theoritikes_pi8anothtes=h.Values./(length(x));
    end
end
```

c) το script που εκτελεί την συνάρτηση που ζητείται

```
[y,fs]=audioread('speech.wav');
N=[2,4,6];
min_value=-1;
max_value=1;
%kanonikopoihsh ths phghs etsi wste na vrisketai sto epitrepto diasthma
%[min_value,max_value]
y=(y-min(y));
f=(min(y)+max(y))/2;
y_c=(y-f)/f;
%eidos_kvantishs=1 gia omoiomorfh kvantish
[ uni_theoritikes_pi8anothtes2,uni_peiramatikes_pi8anothtes2,uni_entropia2,space2] =
upologismos_pi8anothtwn( y_c,N(1),min_value,max_value,1);

[ uni_theoritikes_pi8anothtes4,uni_peiramatikes_pi8anothtes4,uni_entropia4,space4] =
upologismos_pi8anothtwn( y_c,N(2),min_value,max_value,1);

[ uni_theoritikes_pi8anothtes6,uni_peiramatikes_pi8anothtes6,uni_entropia6,space6] =
upologismos_pi8anothtwn( y_c,N(3),min_value,max_value,1);

%eidos kvantishs=2 gia mh-omoiomorfh kvantish
[non_uni_theoritikes_pi8anothtes2,non_uni_peiramatikes_pi8anothtes2,non_uni_entropia2,
space2n] = upologismos_pi8anothtwn( y_c,N(1),min_value,max_value,2);

[non_uni_theoritikes_pi8anothtes4,non_uni_peiramatikes_pi8anothtes4,non_uni_entropia4,
space4n] = upologismos_pi8anothtwn( y_c,N(2),min_value,max_value,2);

[non_uni_theoritikes_pi8anothtes6,non_uni_peiramatikes_pi8anothtes6,non_uni_entropia6,
space6n] = upologismos_pi8anothtwn( y_c,N(3),min_value,max_value,2);
```

d) script zhtoumeno2d.m

```
%vasizomaste gia thn apodosh me thn xrhsh tou mean-squared error
%8a eksetasoume thn kwdikopoihsh gia min_value=-1, max_value=1, N=2,4,6
[y,fs]=audioread('speech.wav');
min_value=-1;
max_value=1;
N=[2,4,6];
y=(y-min(y));
f=(min(y)+max(y))/2;
y_c=(y-f)/f;
[xq2d,centers2d] = my_quantizer(y_c,N(1),min_value,max_value);
[xq4d,centers4d] = my_quantizer(y_c,N(2),min_value,max_value);
[xq6d,centers6d] = my_quantizer(y_c,N(3),min_value,max_value);

[xq1l,centers1,D1]=Lloyd_Max(y_c,N(1),min_value,max_value);
[xq2l,centers2,D2]=Lloyd_Max(y_c,N(2),min_value,max_value);
[xq3l,centers3,D3]=Lloyd_Max(y_c,N(3),min_value,max_value);
%dianusma p ferei ta mse gia ka8e N omoiomorfhs kwdikopoihshs
mse1=zeros(length(3),1);
%dianusma p ferei ta mse gia ka8e N mh-omoiomorfhs kwdikopoihshs
mse=zeros(length(3),1);
m1=0;
m2=0;
m3=0;
m4=0;
m5=0;
m6=0;
for i=1:length(y_c)
    m1=m1+(y_c(i)-xq2d(i)).^2;
    m2=m2+(y_c(i)-xq4d(i)).^2;
    m3=m3+(y_c(i)-xq6d(i)).^2;
    m4=m4+(y_c(i)-xq1l(i)).^2;
    m5=m5+(y_c(i)-xq2l(i)).^2;
    m6=m6+(y_c(i)-xq3l(i)).^2;
end
mse(1)=m1/length(y_c);
mse(2)=m2/length(y_c);
mse(3)=m3/length(y_c);
```

```
mse1(1)=m4/length(y_c);
mse1(2)=m5/length(y_c);
mse1(3)=m6/length(y_c);
```

ΒΙΒΛΙΟΓΡΑΦΙΑ

- **«Συστήματα Τηλεπικοινωνιών», John G.Proakis , Masoud Salehi**
- **«http://www.mathworks.com/»**
- **«https://eclass.upatras.gr/»**