ΨΗΦΙΑΚΕΣ ΤΗΛΕΠΙΚΟΙΝΩΝΙΕΣ

Ιανουάριος 2024

1^{ο} σετ Εργαστηριακών Ασκήσεων

Ζήκος Σπύρος, 1084581, 4^{ο} έτος

Μέρος Α΄

1. a. Στον κώδικα, στην αρχή μετράμε τον αριθμό των εμφανίσεων όλων των πιθανών συμβόλων της πηγης (0-255) και μετά υπολογίζουμε τα σύμβολα που πραγματικά εμφανίζονται στην πηγη και την πιθανότητα εμφάνισης κάθε τέτοιου συμβόλου. Τα σύμβολα της πηγής είναι όλα τα πολλαπλάσια του 17. Είναι δηλαδή οι τιμές 0, 17, 34, …, 238, 255 με αντίστοιχες πιθανότητες:

0.0962	0.0814	0.0683	0.0625	0.0768	0.0905	0.1132	0.0906
0.0965	0.0671	0.0389	0.0329	0.0337	0.0259	0.0224	0.0031

b. Χρησιμοποιώντας την συνάρτηση huffmandict και υπολογίζοντας την εντροπία προκύπτουν:

i) Εντροπία κωδικοποίησης: 3.783082374204571

ii) Μέσο μήκος κώδικα: 3.837366666666667

iii) Αποδοτικότητα κώδικα: 3.783 / 3.8373 ≈ 0.98585

Βλέπουμε ότι το μέσος μήκος κώδικα είναι πολύ κοντά στην εντροπία κάτι που φαίνεται και από την αποδοτικότητα του κώδικα που σημαίνει ότι η κωδικοποίησή μας είναι καλή.

2. a. Ομαδοποιούμε τα σύμβολα της πηγής και κάνουμε ότι κάναμε και στο προηγούμενο ερώτημα. Τα σύμβολα της πηγής είναι πολλαπλάσια του 17 και είναι μεταξύ 0 και 65535. Φτιάχνονται προσθέτοντας δύο διαδοχικά σύμβολα της πηγής αφού πρώτα έχουμε πολλαπλασιάσει το πρώτο από τα δύο με το 256. Μπορούμε να ανακτήσουμε τα αρχικά σύμβολα με ακέραια διαίρεση και υπόλοιπο (mod(x,256)) του σύνθετου συμβόλου.

Σύμβολα (ο πίνακας προκύπτει από reshape, να αγνοηθούν τα 2 τελευταία στοιχεία)

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	0	4437	8806	13158	17476	21794	21998	26316	30651	34969	39304	43656	48059	52462	61081
2	17	4454	8823	13175	17493	21811	26129	26350	30668	34986	39321	43673	48076	52479	61115
3	34	4488	8840	13192	17510	21828	26146	26367	30685	35003	39338	43690	48093	56695	61132
4	51	4505	8857	13209	17527	21845	26163	30498	30702	35020	39355	43707	48110	56729	61149
5	68	4522	8874	13226	17544	21862	26180	30515	30719	35037	39372	43724	48127	56746	61166
6	85	4539	8891	13243	17561	21879	26197	30532	34850	35054	39389	43741	52326	56763	61183
7	102	8704	13056	13260	17578	21896	26214	30549	34867	35071	39423	43758	52360	56780	65433
8	4352	8721	13073	13277	17595	21913	26231	30566	34884	39219	43554	43775	52377	56797	65501
9	4369	8738	13090	13294	17612	21930	26248	30583	34901	39236	43588	47991	52394	56814	65518
10	4386	8755	13107	17425	17629	21947	26265	30600	34918	39253	43605	48008	52411	56831	65535
11	4403	8772	13124	17442	17646	21964	26282	30617	34935	39270	43622	48025	52428	61013	0
12	4420	8789	13141	17459	21777	21981	26299	30634	34952	39287	43639	48042	52445	61047	0

Πιθανότητες (ο πίνακας προκύπτει από reshape, να αγνοηθούν τα 2 τελευταία στοιχεία)

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	0.0821	2.0000e-04	7.3333e-04	0.0011	0.0413	0.0018	1.3333e-04	2.0000e-04	4.6667e-04	0.0128	0.0176	0.0022	0.0137	9.3333e-04	1.3333e-04
2	0.0138	3.3333e-04	3.3333e-04	5.3333e-04	0.0143	0.0036	6.6667e-05	6.6667e-05	2.0000e-04	0.0019	0.0345	0.0107	0.0063	2.6667e-04	3.3333e-04
3	0.0012	2.6667e-04	4.0000e-04	2.0000e-04	0.0025	0.0131	3.3333e-04	6.6667e-05	6.6667e-05	0.0011	0.0073	0.0167	0.0016	6.6667e-05	9.3333e-04
4	5.3333e-04	1.3333e-04	4.6667e-04	4.6667e-04	0.0012	0.0498	0.0017	2.0000e-04	2.6667e-04	4.0000e-04	0.0011	0.0059	4.6667e-04	2.0000e-04	0.0054
5	1.3333e-04	1.3333e-04	1.3333e-04	2.6667e-04	8.0000e-04	0.0159	0.0033	3.3333e-04	6.6667e-05	6.6667e-05	8.0000e-04	0.0013	1.3333e-04	6.6667e-04	0.0143
6	6.6667e-05	6.6667e-05	2.0000e-04	6.6667e-05	7.3333e-04	0.0021	0.0165	0.0019	6.6667e-05	1.3333e-04	1.3333e-04	3.3333e-04	1.3333e-04	0.0014	7.3333e-04
7	1.3333e-04	6.0000e-04	2.6667e-04	1.3333e-04	2.6667e-04	0.0012	0.0702	0.0047	2.0000e-04	2.0000e-04	6.6667e-05	1.3333e-04	6.6667e-04	0.0067	6.6667e-05
8	0.0113	0.0136	0.0028	6.6667e-05	1.3333e-04	5.3333e-04	0.0171	0.0150	4.0000e-04	6.6667e-05	6.6667e-05	1.3333e-04	1.0000e-03	0.0132	2.0000e-04
9	0.0501	0.0353	0.0125	6.6667e-05	2.0000e-04	3.3333e-04	0.0028	0.0493	0.0013	6.6667e-05	6.6667e-05	2.0000e-04	0.0024	0.0032	0.0030
10	0.0147	0.0129	0.0287	0.0012	2.6667e-04	3.3333e-04	0.0013	0.0143	0.0043	4.0000e-04	4.0000e-04	6.6667e-04	0.0073	2.6667e-04	5.3333e-04
11	0.0027	0.0023	0.0123	0.0023	1.3333e-04	1.3333e-04	7.3333e-04	0.0029	0.0161	8.0000e-04	6.6667e-05	0.0019	0.0164	6.6667e-05	0
12	9.3333e-04	0.0013	0.0024	0.0125	2.6667e-04	2.0000e-04	7.3333e-04	6.0000e-04	0.0565	0.0035	3.3333e-04	0.0079	0.0046	6.6667e-05	0

b. i) Εντροπία κωδικοποίησης: 5.614735314715528

ii) Μέσο μήκος κώδικα: 5.641199999999990

iii) Αποδοτικότητα κώδικα: 5.6147 / 5.6412 ≈ 0.9953

Βλέπουμε ότι το μέσος μήκος κώδικα είναι πολύ κοντά στην εντροπία κάτι που φαίνεται και από την αποδοτικότητα του κώδικα.

c.

Παρατηρούμε ότι η αποδοτικότητα του κώδικα βελτιώθηκε περίπου 1%. Επίσης, η εντροπία κωδικοποίησης αυξήθηκε αλλά και το μέσο μήκος κώδικα αυξήθηκε. Όμως, η αύξηση της αποδοτικότητας δείχνει ότι η εντροπία κωδικοποίησης αυξήθηκε (ποσοστιαία) περισσότερο απ' ότι αυξήθηκε το μέσο μήκος κώδικα. Άρα πετύχαμε καλύτερη κωδικοποίηση.

3. a. Ο τύπος αυτός δεν ισχύει για τον υπολογισμό της εντροπίας της δεύτερης τάξης επέκτασης πηγής επειδή η πηγή μας έχει μνήμη καθώς τα pixel της φωτογραφίας είναι συσχετισμένα μεταξύ τους. Ο τύπος αυτός ισχύει μόνο για πηγές χωρίς μνήμη.

b. $H(X^2) \leq L_2 < H(X^2) + 1 \Rightarrow 5.6147 \leq L_2 < 5.6147 + 1 \Rightarrow 5.6147 \leq L_2 < 6.6147$

4. Κωδικοποιούμε την πηγή (με huffmanenco) και μετά την αποκωδικοποιούμε (με huffmandeco) και συγκρίνουμε το αποτέλεσμα με το αρχικό σήμα (και είναι ίδιο).

Ύστερα υπολογίζουμε την δυαδική αναπαράσταση της πηγής και υπολογίζουμε τον λόγο:

$J = 115121 / 240000 = 0.479671$

5.

Πιθανότητα σωστής μετάβασης: $p=0.88$

Χωρητικότητα καναλιού:

$C = 1 + p*\log_2 p + (1-p)*\log_2(1-p) = 1 + 0.88* \log_2 0.88 + 0.12*\log_2(0.12) =$
$= 1 + 0.88*(-0.1844) + 0.12*(-3.059) = 1 - 0.162272 - 0.36708 = 0.47064$

- $x0 = 0$, $x1 = 1$, $y0=0$, $y1=1$,
- $p(x0)=0.5519$ (<u>από τον κώδικα</u>), $p(x1)=1-p(x0)=0.4481$,
- $p(yi|xi)=0.88$, $p(yi|xj)=0.12$ $(i!=j)$

- $p(y0) = p(y0|x0)*p(x0) + p(y0|x1)*p(x1) = 0.88*0.5519 + 0.12*0.4481 = 0.539444$
- $p(y1) = 1 - p(y0) = 0.460556$

- $p(x0,y0) = p(y0|x0)*p(x0) = 0.88*0.5519 = 0.485672$
- $p(x0,y1) = p(y1|x0)*p(x0) = 0.12*0.5519 = 0.066228$
- $p(x1,y0) = p(y0|x1)*p(x1) = 0.12*0.4481 = 0.053772$
- $p(x1,y1) = p(y1|x1)*p(x1) = 0.88*0.4481 = 0.394328$

- $H(X|Y) = \Sigma\Sigma p(xi,yj) *\log_2(p(xi,yj)/p(yj)) =$
  $= p(x0,y0)*\log_2(p(x0,y0)/p(y0)) + p(x0,y1)*\log_2(p(x0,y1)/p(y1)) +$
  $+ p(x1,y0)*\log_2(p(x1,y0)/p(y0)) + p(x1,y1)*\log_2(p(x1,y1)/p(y1)) =$
  $= 0.485672*\log_2(0.485672/0.539444) +$
  $+ 0.066228*\log_2(0.066228/0.460556) +$
  $+ 0.053772*\log_2(0.053772/0.539444) +$
  $+ 0.394328*\log_2(0.394328/0.460556) =$
  $= 0.485672*\log_2(0.90032) + 0.066228*\log_2(0.1438) +$
  $+ 0.053772*\log_2(0.09968) + 0.394328*\log_2(0.8562) =$
  $= 0.485672*(-0.15149) + 0.066228*(-2.7979) +$
  $+ 0.053772*(-3.3266) + 0.394328*(-0.224) =$
  $= - 0.073574 - 0.1853 - 0.178878 - 0.088329 = -0.526081 \simeq -0.526$

- $H(X) = -p(x0) *\log_2 p(x0) – p(x1)*\log_2 p(x1) =$
  $= -0.5519* \log_2(0.5519) - 0.4481*\log_2(0.4481) =$
  $= -0.5519*(-0.8575) -0.4481*(-1.1581) = 0.9922$

Αμοιβαία πληροφορία:

➔ $I(X;Y) = H(X) – H(X|Y) = 0.9922 - 0.526 = 0.4662$

Κώδικας

```
I = imread('parrot.png');
format short;

% count each possible pixel value 0-255
I_symbols = zeros(1,256);
for i = I
    for j = i'
        I_symbols(1,double(j)+1) = I_symbols(1,double(j)+1) + 1;
    end
end

% actual pixel values and their probabilities
symbols = zeros(1,nnz(I_symbols));
probabilities = zeros(1,nnz(I_symbols));
k=1;   % 1-16
for i = linspace(1,256,256)
    if (I_symbols(1,i) ~= 0)
        symbols(1,k) = i-1;
        % probability = symbol appearance / total appearances
        probabilities(1,k) = I_symbols(1,i)/(200*150);
        k = k + 1;
    end
end

[dict, avglen] = huffmandict(symbols, probabilities);

hf = 0;
for p = probabilities
    hf = hf - (p*log2(p));
end

fprintf("\tentropy   length    performance\n")
disp([hf, avglen, hf/avglen])

% ########### 2nd order ############
% count each possible combination of 2 pixel values 0 - 65535
I_symbols2 = zeros(1,65536);
subsum = 0;
even = 0;
for i = I
    for j = i'
        if (even ~= 0)
            I_symbols2(1,double(j)+subsum+1) = ...
            I_symbols2(1,double(j)+subsum+1) + 1;
            even = 0;
        else
            subsum = double(j)*256;
            even = 1;
        end
    end
end

% actual pixel values and their probabilities
symbols2 = zeros(1,nnz(I_symbols2));
probabilities2 = zeros(1,nnz(I_symbols2));
k=1;   % 1-186
```

```matlab
for i = linspace(1,65536,65536)
    if (I_symbols2(1,i) ~= 0)
        symbols2(1,k) = i-1;
        % probability = symbol appearance / total appearances
        probabilities2(1,k) = I_symbols2(1,i)/(100*150);
        k = k + 1;
    end
end

[dict2, avglen2] = huffmandict(symbols2, probabilities2);

hf2 = 0;
for p = probabilities2
    hf2 = hf2 - (p*log2(p));
end

fprintf("\tentropy   length    performance\n")
disp([hf2, avglen2, hf2/avglen2])

% ########### codify source ############
enc = huffmanenco(I(:)',dict);
deco = huffmandeco(enc, dict);

if (deco' == I(:))
    fprintf("\tDecoding the encoded source we get the initial source.\n")
end


bI = reshape((dec2bin(typecast(I(:),'uint8'),4)-'0').',1,[]);
J = size(enc,2)/size(bI,2);
fprintf("\tJ = %f\n\n", J)

% ########### channel ############
x=enc';

zeros_x = 0;
zeros_y_correct = 0;

many_times = 100;   % ideal: 10000+
for k=(1:many_times)
    y=binary_symmetric_channel(x);
    for i=(1:size(x))
        if x(i) == 0
            zeros_x = zeros_x + 1;
            if y(i) == 0
                zeros_y_correct = zeros_y_correct + 1;
            end
        end
    end
end
p_correct = round(zeros_y_correct/zeros_x,2);
fprintf("\tProbability p: %.2f\n", p_correct)

x_size = size(x);
p_x0 = (zeros_x/many_times)/x_size(1);
fprintf("\tProbability of x0=0: %.4f\n", p_x0)
```

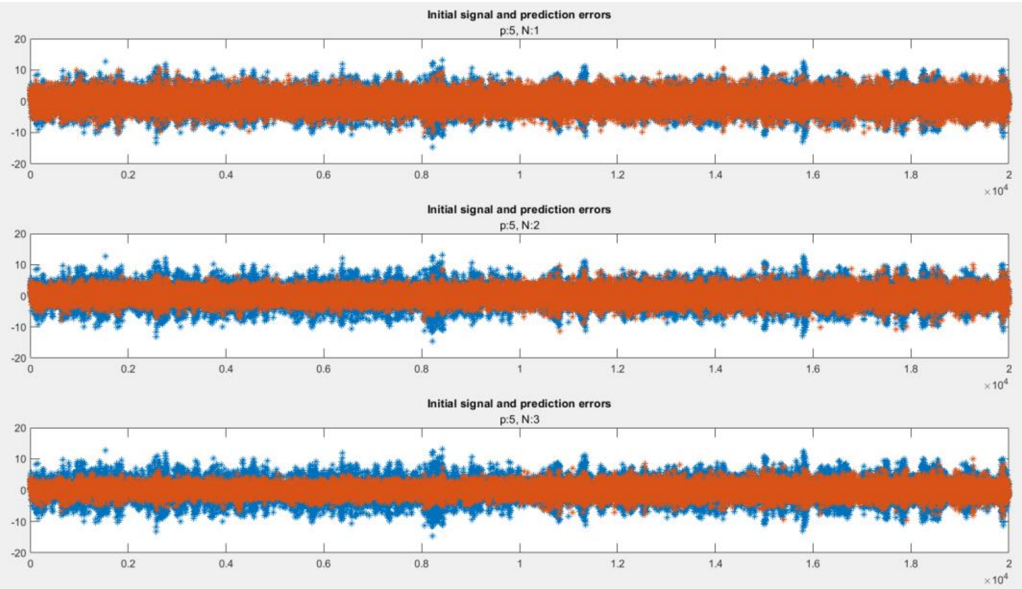
Μέρος Β΄

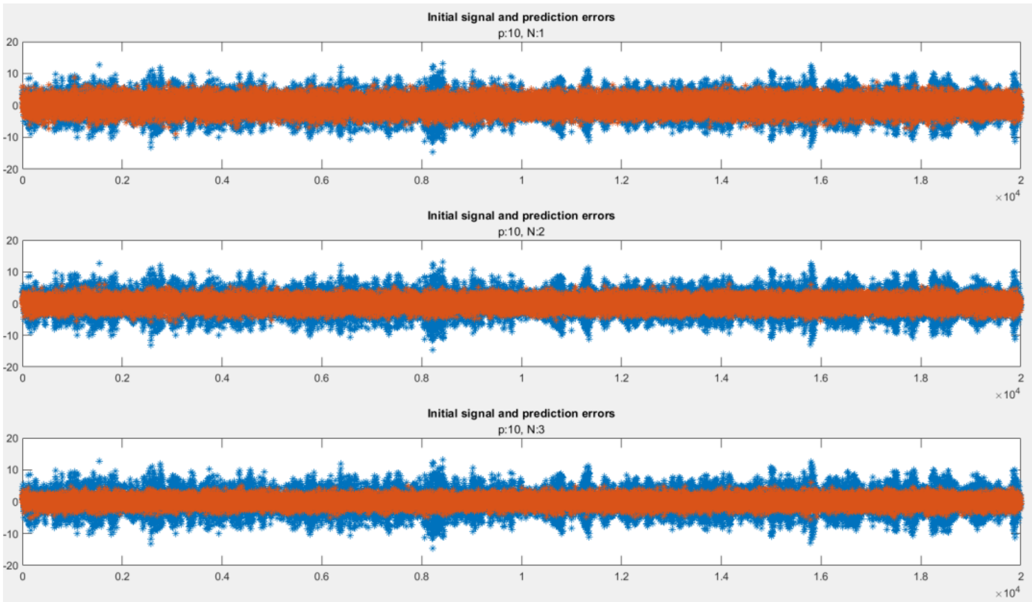
Για τον κβαντιστή υποθέτω ότι αν y=3 και οι γειτονικές περιοχές κβάντισης είναι οι [1,3] και [3,5] τότε το y κβαντίζεται στην περιοχή [3,5] στην τιμή 4. Δηλαδή οι τιμές που βρίσκονται στα όρια περιοχών παίρνουν την μεγαλύτερη τιμή κβάντισης.

20 ΧΙΛΙΑΔΕΣ ΔΕΙΓΜΑΤΑ

1. Οκ.

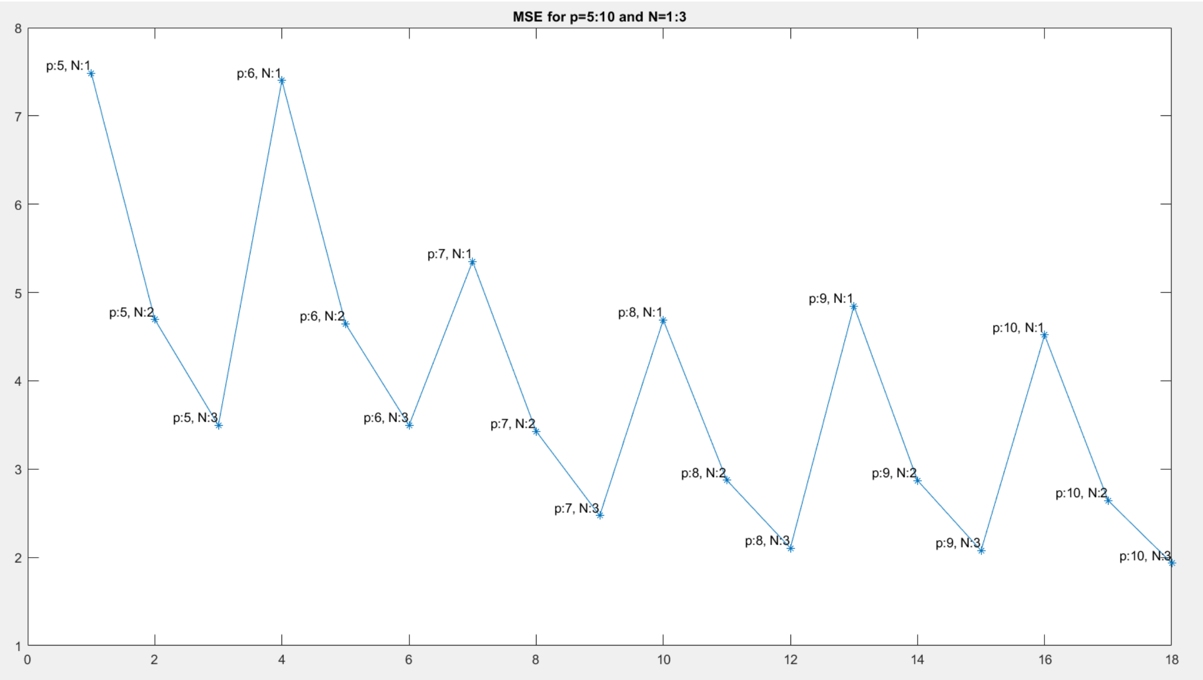
2. Επιλέγω p=5 και p=10.



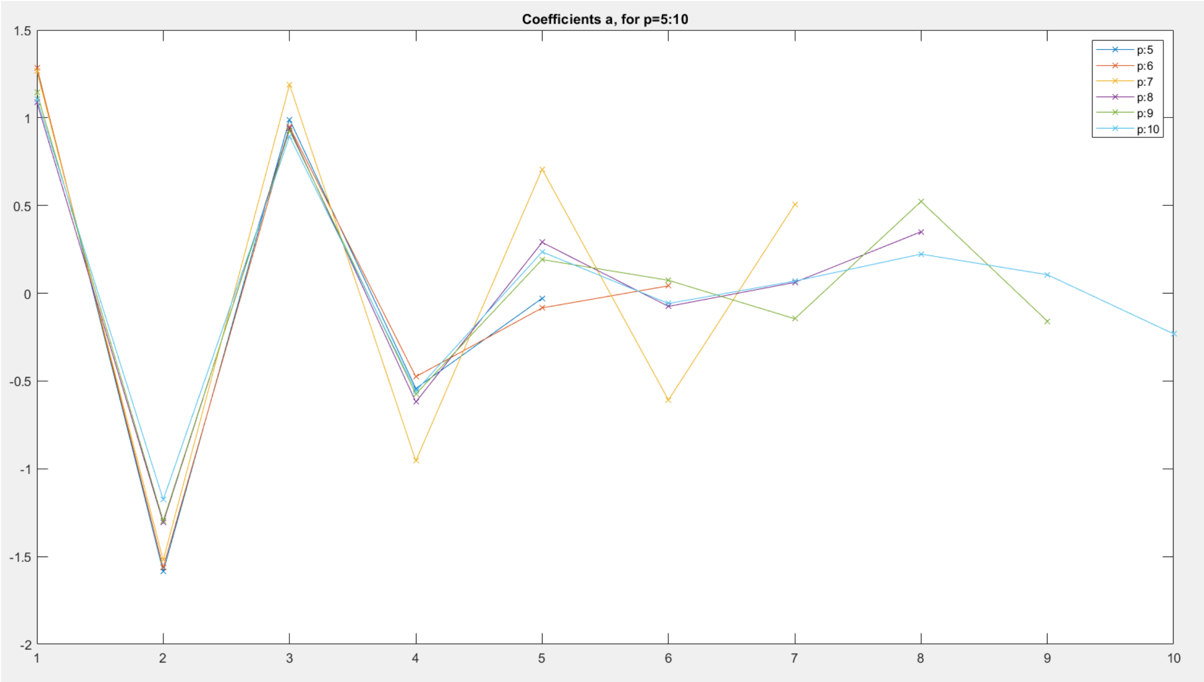


Τα γραφήματα δείχνουν ότι η δυναμική περιοχή του σφάλματος πρόβλεψης είναι μικρότερη από την δυναμική περιοχή του σήματος και έτσι θα υπάρχει μικρότερο σφάλμα κατά την κβάντηση. Παρατηρώ ότι το σφάλμα πρόβλεψης κατά απόλυτη τιμή είναι πιο πυκνό στις χαμηλές τιμές απ’ ότι ψηλές τιμές. Επίσης, όσο αυξάνεται το N τόσο μικραίνει και η δυναμική περιοχή του σφάλματος πρόβλεψης επειδή έχουμε καλύτερη ακρίβεια κατά την κβάντιση του κάθε σφάλματος πρόβλεψης το οποίο επηρεάζει τις τιμές των επόμενων σφαλμάτων πρόβλεψης. Ακόμα, όσο αυξάνεται το p τόσο μειώνεται η πυκνότητα του σφάλματος πρόβλεψης στις ψηλότερες τιμές το οποίο οφείλεται στο ότι έχουμε περισσότερους όρους και άρα περισσότερο ακριβής πρόβλεψη.

3.



Η απόδοση του συστήματος είναι καλή διότι το μέσο τετραγωνικό σφάλμα μειώνεται όταν αυξάνουμε το N κρατώντας το p σταθερό και μειώνεται όταν αυξάνουμε το p κρατώντας το N σταθερό.

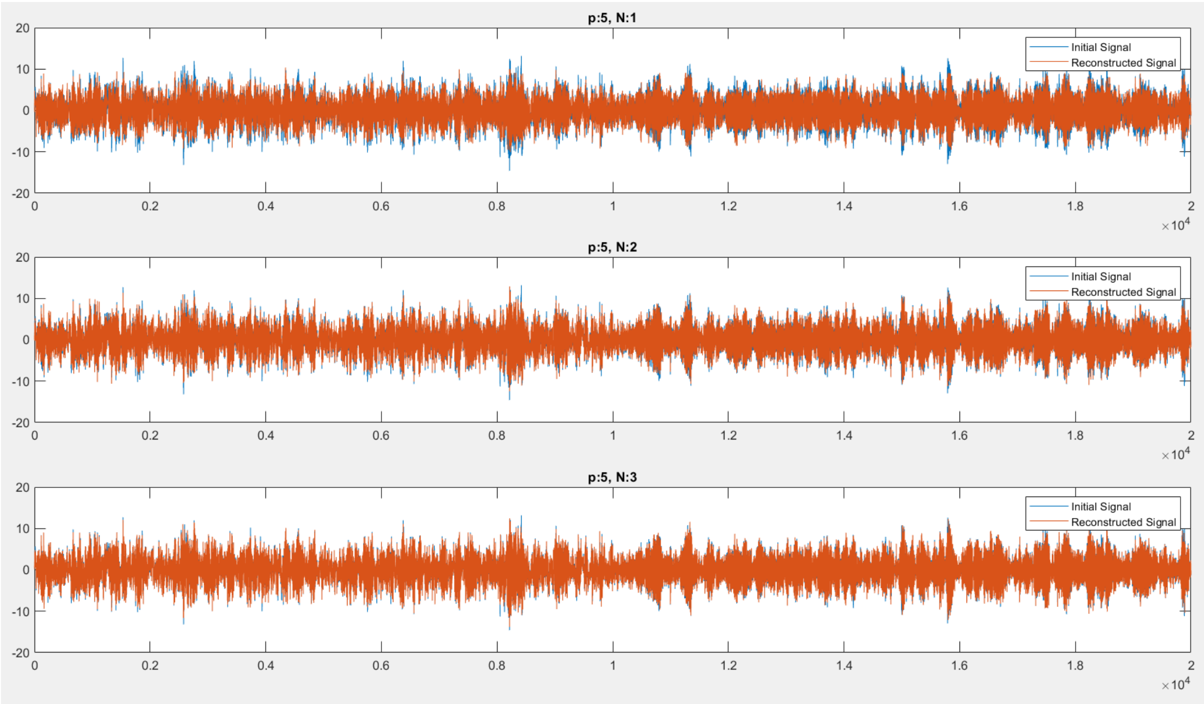


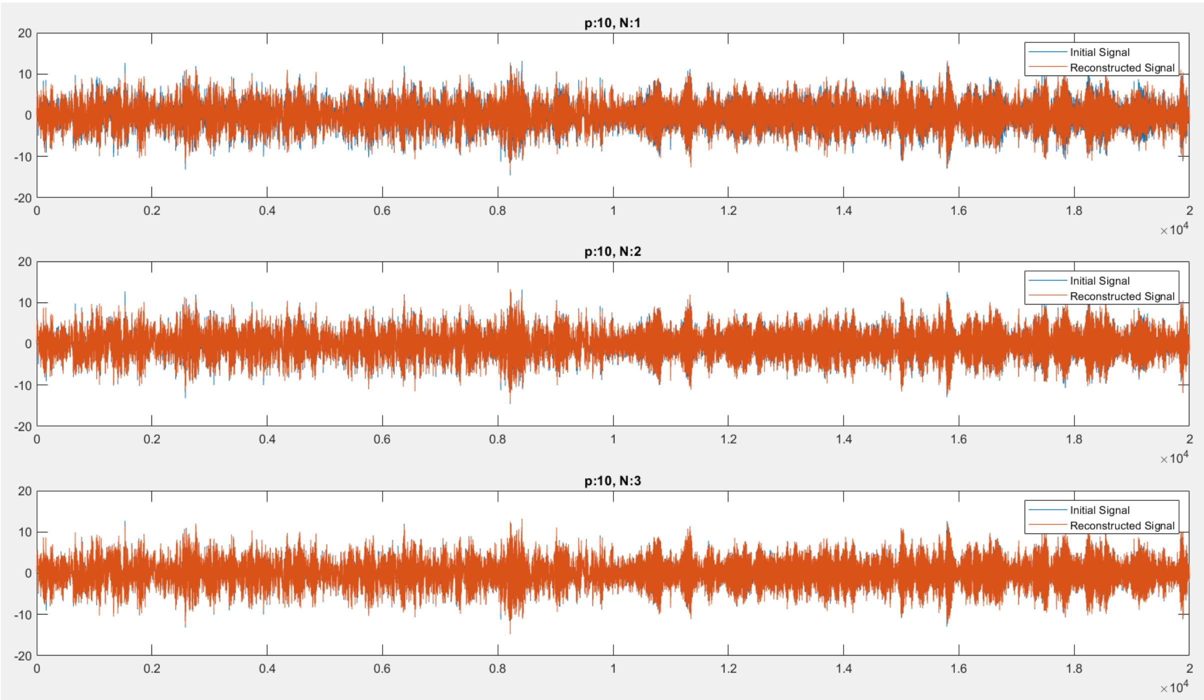
Συντελεστές του προβλέπτη για p=5:10 (Κάθε γραμμή αντιστοιχεί σε μία τιμή του p)

1.2855	-1.5859	0.9905	-0.5425	-0.0286	0	0	0	0	0
1.2865	-1.5627	0.9482	-0.4751	-0.0833	0.0425	0	0	0	0
1.2651	-1.5209	1.1888	-0.9549	0.7071	-0.6081	0.5056	0	0	0
1.0877	-1.3073	0.9402	-0.6195	0.2899	-0.0748	0.0621	0.3505	0	0
1.1433	-1.2975	0.9287	-0.5742	0.1922	0.0740	-0.1453	0.5233	-0.1589	0
1.1063	-1.1763	0.8947	-0.5566	0.2360	-0.0583	0.0692	0.2231	0.1058	-0.2314

Οι συντελεστές του προβλέπτη αρχίζουν από αριθμούς κατά απόλυτη τιμή μεγαλύτερους της μονάδας και σταδιακά πλησιάζουν όλο και πιο κοντά στο 0. Αυτό γίνεται λόγω της μεγαλύτερης επίδρασης που έχουν τα γειτονικά δείγματα στην πρόβλεψη. Οι συντελεστές του δείγματος 5 και πιο πριν χρονικά είναι κοντά στο μηδέν που σημαίνει ότι τα δείγματα αυτά επηρεάζουν λίγο την τιμή του δείγματος που θέλουμε να προβλέψουμε.

4.





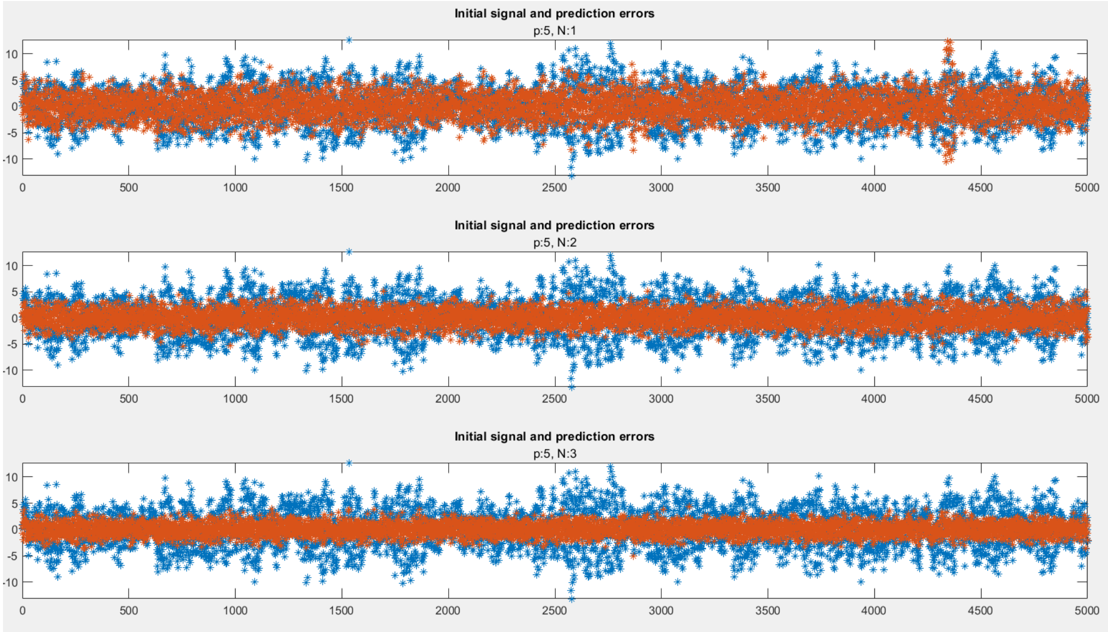
Παρατηρούμε ότι όσο μεγαλώνει το N τόσο μοιάζει περισσότερο το αρχικό σήμα με το ανακατασκευασμένο.

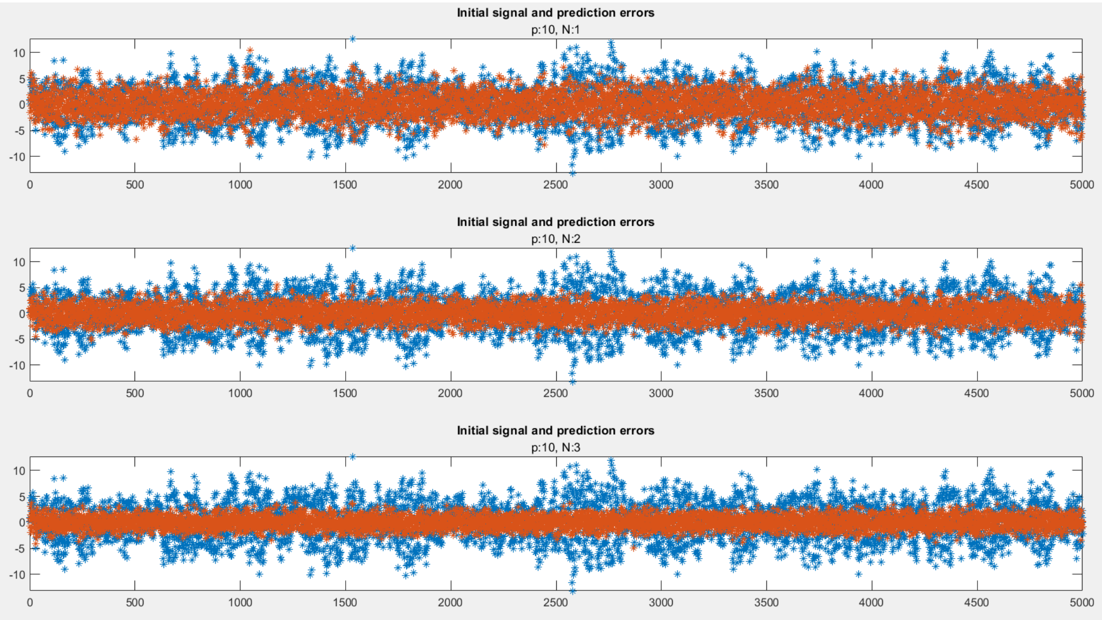
ΠΡΩΤΑ 5 ΧΙΛΙΑΔΕΣ ΔΕΙΓΜΑΤΑ

1.

Οκ.

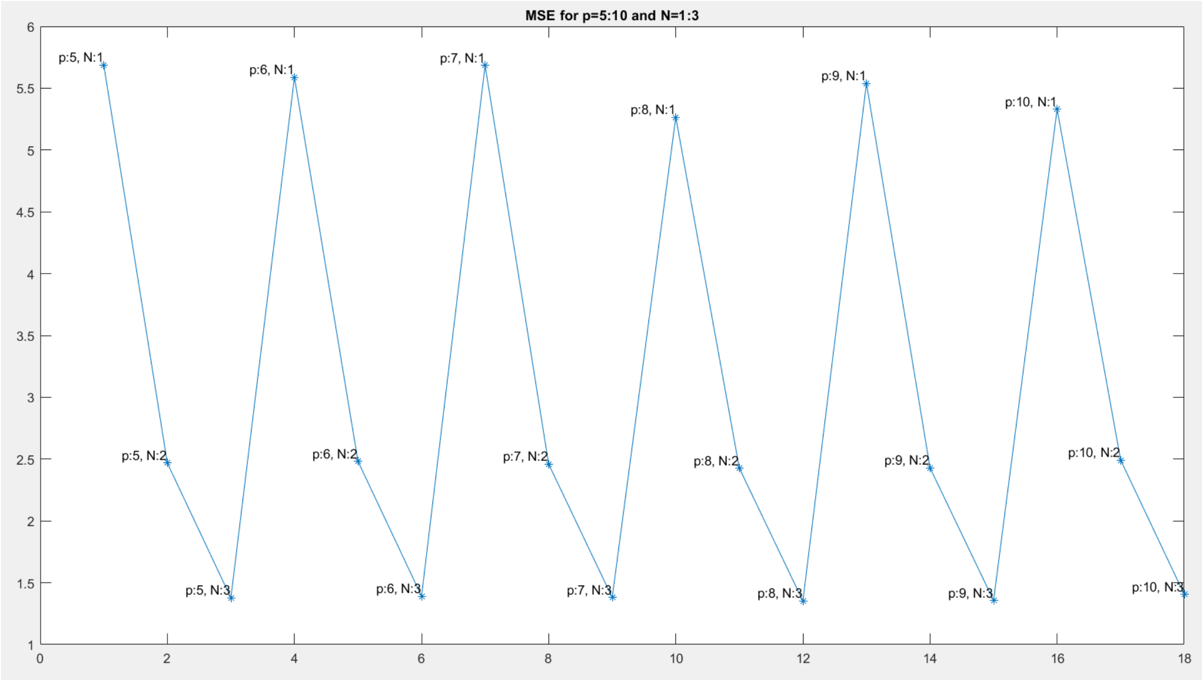
2.



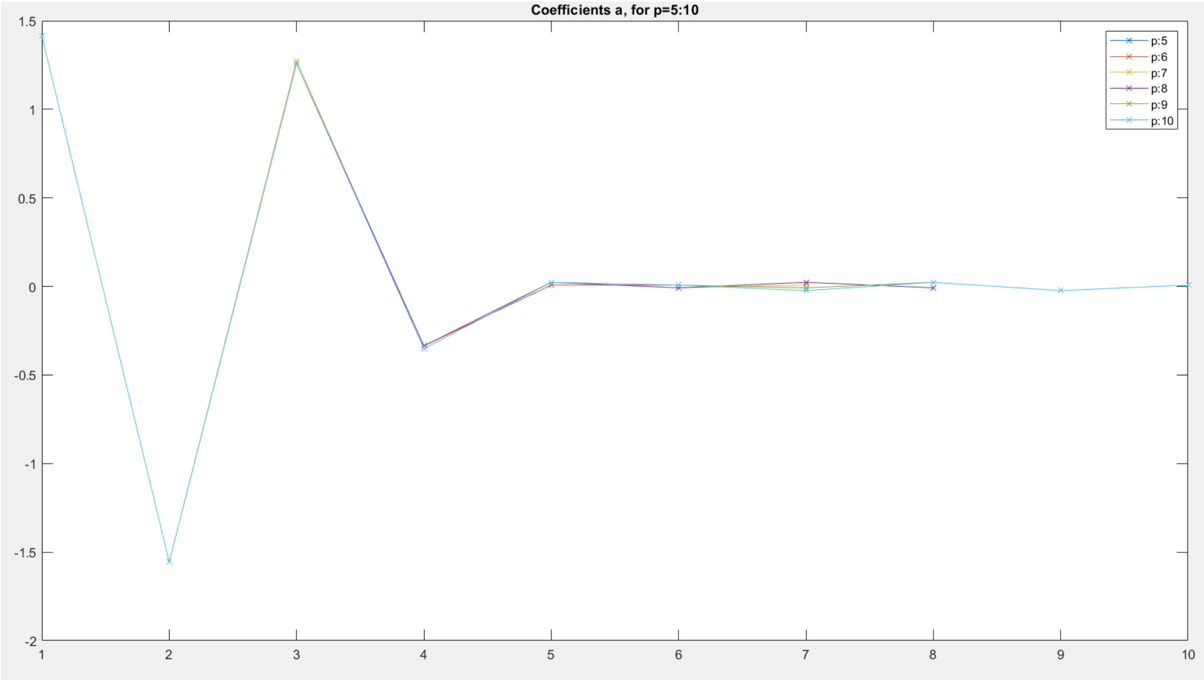


Παρόμοια αποτελέσματα με τα 20 χιλιάδες δείγματα. Οι δυναμικές περιοχές του σφάλματος πρόβλεψης φαίνονται λίγο μικρότερες απ΄ότι στα 20 χιλιάδες δείγματα.

3.



Το μέσο τετραγωνικό σφάλμα μειώνεται όταν αυξάνουμε το N κρατώντας το p σταθερό αλλά η απόδοση του συστήματος δεν είναι πολύ καλή διότι το μέσο τετραγωνικό σφάλμα δεν μειώνεται πάντα όταν αυξάνουμε το p κρατώντας το N σταθερό. Σε σχέση με τα 20 χιλιάδες δείγματα το σφάλμα για N=1 και p μεγαλύτερο ίσο του 7 είναι μεγαλύτερο σε αυτά τα 5 χιλ. δείγματα. Όμως, για μεγαλύτερα N το σφάλμα είναι μικρότερο στα 5 χιλ. δείγματα.



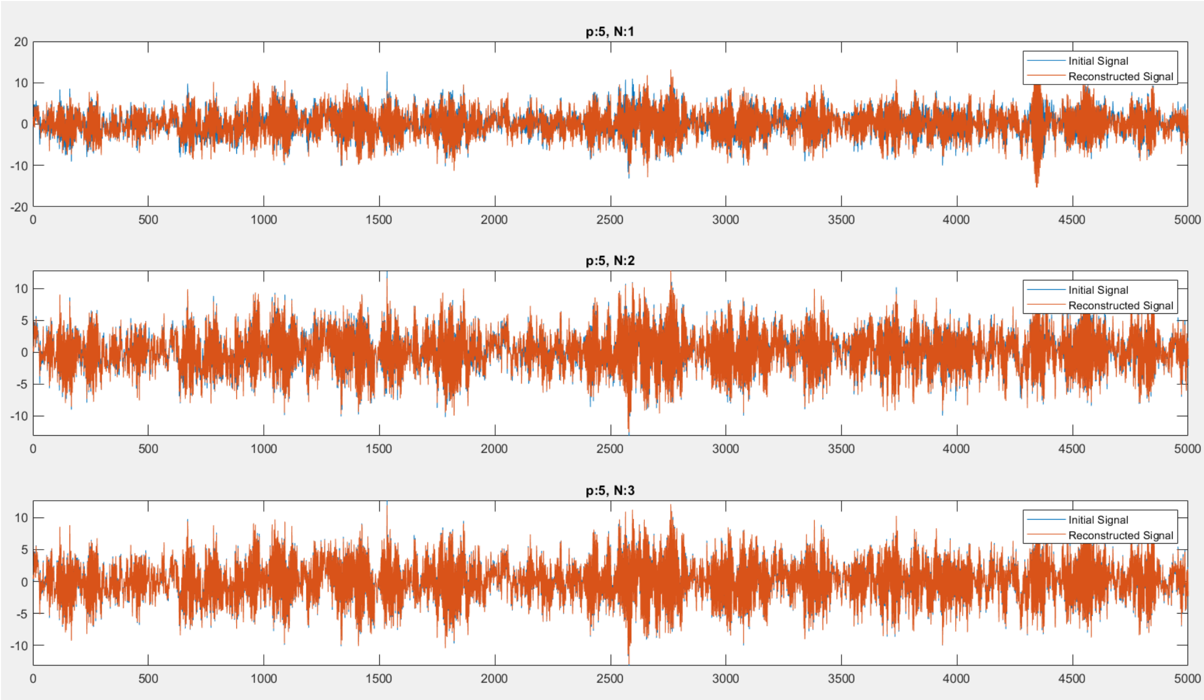
Συντελεστές του προβλέπτη για p=5:10 (Κάθε γραμμή αντιστοιχεί σε μία τιμή του p)

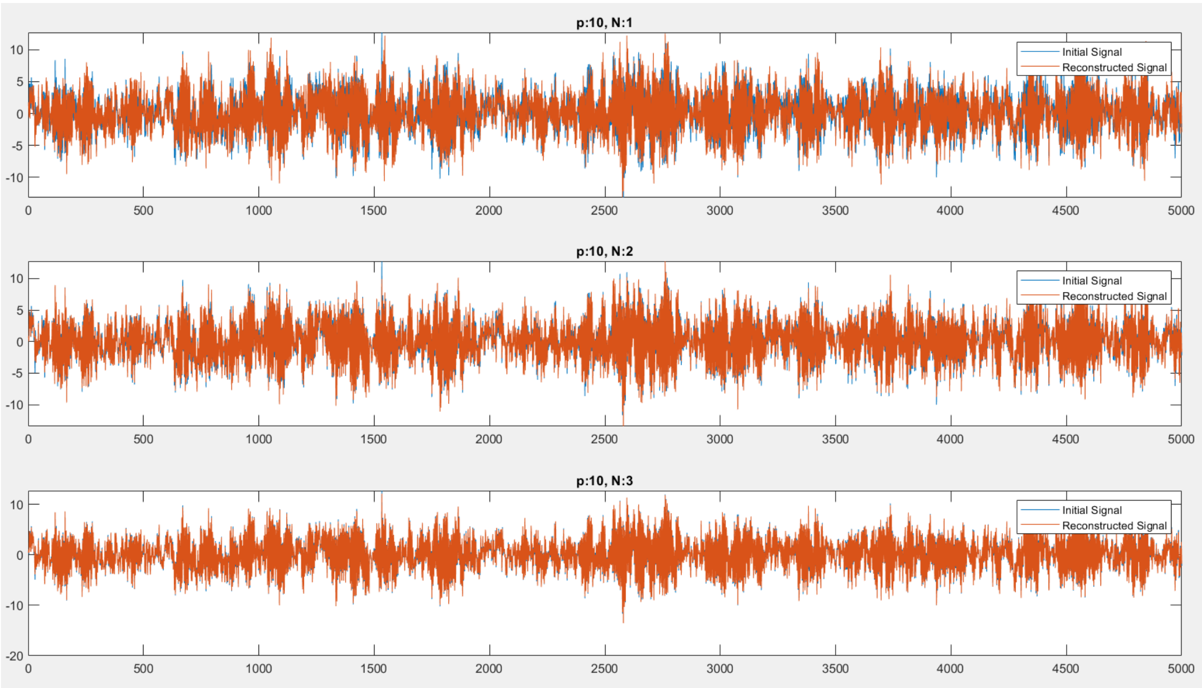
1.4141	-1.5547	1.2578	-0.3359	0.0078	0	0	0	0	0
1.4141	-1.5547	1.2578	-0.3359	0.0078	0.0078	0	0	0	0
1.4141	-1.5547	1.2734	-0.3516	0.0234	-0.0078	0.0078	0	0	0
1.4141	-1.5547	1.2578	-0.3359	0.0234	-0.0078	0.0234	-0.0078	0	0
1.4141	-1.5547	1.2578	-0.3516	0.0234	0.0078	-0.0078	0.0234	-0.0234	0
1.4141	-1.5547	1.2578	-0.3516	0.0234	0.0078	-0.0234	0.0234	-0.0234	0.0078

Οι συντελεστές του προβλέπτη αρχίζουν από αριθμούς κατά απόλυτη τιμή μεγαλύτερους της μονάδας και σταδιακά πλησιάζουν όλο και πιο κοντά στο 0. Αυτό γίνεται λόγω της μεγαλύτερης επίδρασης που έχουν τα γειτονικά δείγματα στην πρόβλεψη. Οι συντελεστές του δείγματος 5 και πιο πριν χρονικά είναι κοντά στο μηδέν που σημαίνει ότι τα δείγματα αυτά επηρεάζουν λίγο την τιμή του δείγματος που θέλουμε να προβλέψουμε.

Σε σχέση με τα 20 χιλιάδες δείγματα, για τα πρώτα 5 χιλιάδες δείγματα οι συντελεστές μετά το 5^{ο} δείγμα είναι πιο κοντά στο 0 απ’ ότι για τα 20 χιλιάδες δείγματα και οι πρώτοι συντελεστές είναι πιο μεγάλοι από αυτούς που βρήκαμε στα 20 χιλιάδες δείγματα.

4.





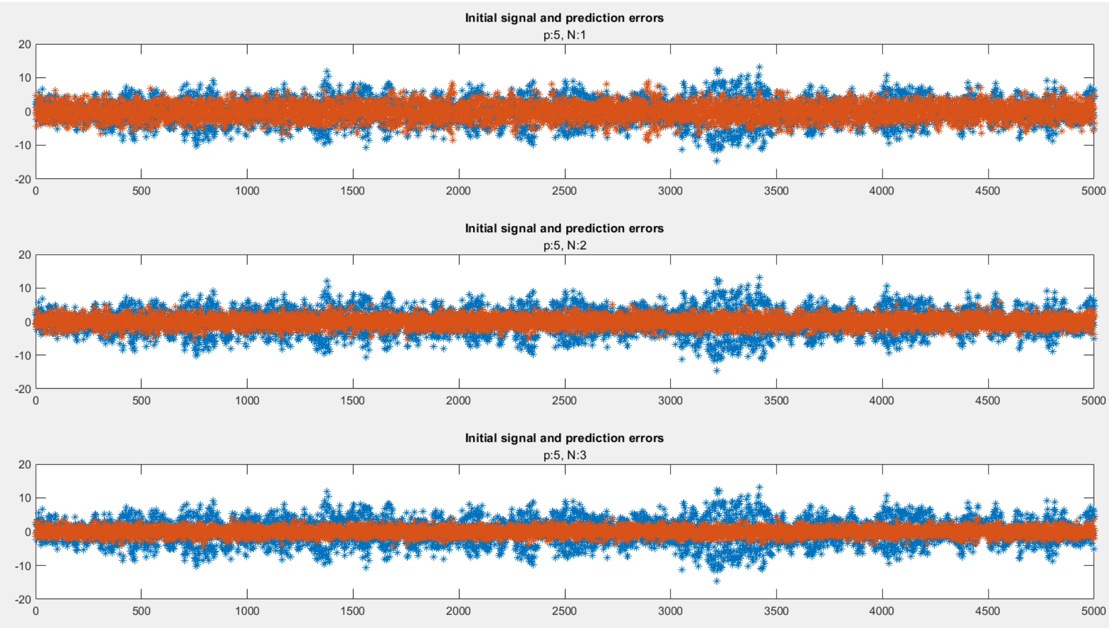
Παρατηρούμε ότι όσο μεγαλώνει το N τόσο μοιάζει περισσότερο το αρχικό σήμα με το ανακατασκευασμένο. Τα αποτελέσματα αυτά είναι παρόμοια με τα 20 χιλιάδες δείγματα.

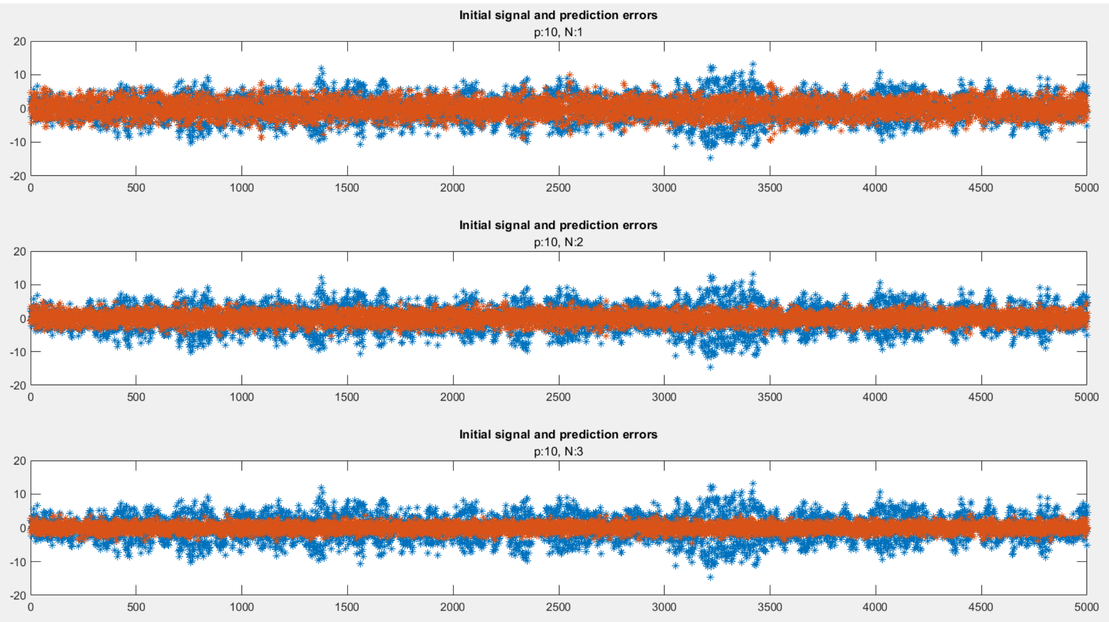
ΔΕΥΤΕΡΑ 5 ΧΙΛΙΑΔΕΣ ΔΕΙΓΜΑΤΑ

1.

Οκ.

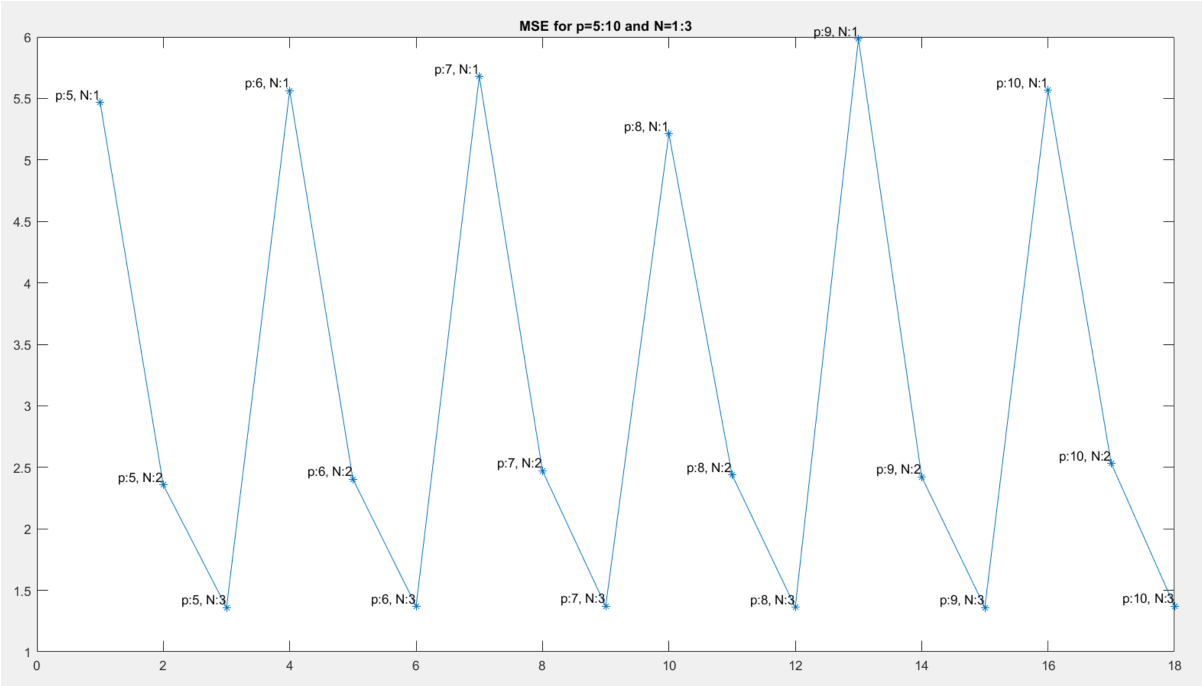
2.



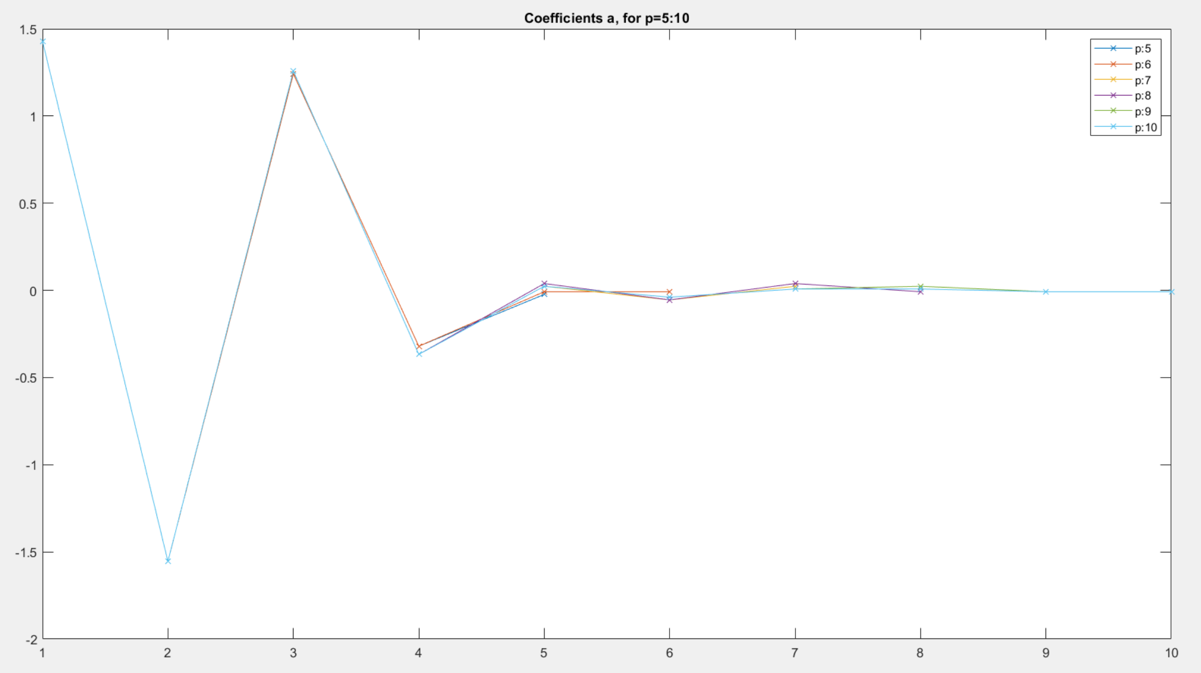


Παρόμοια αποτελέσματα με τα 20 χιλιάδες δείγματα. Οι δυναμικές περιοχές του σφάλματος πρόβλεψης φαίνονται λίγο μικρότερες απ΄ότι στα 20 χιλιάδες δείγματα.

3.



Το μέσο τετραγωνικό σφάλμα μειώνεται όταν αυξάνουμε το N κρατώντας το p σταθερό αλλά η απόδοση του συστήματος δεν είναι πολύ καλή διότι το μέσο τετραγωνικό σφάλμα δεν μειώνεται πάντα όταν αυξάνουμε το p κρατώντας το N σταθερό. Σε σχέση με τα 20 χιλιάδες δείγματα το σφάλμα για N=1 και p μεγαλύτερο ίσο του 7 είναι μεγαλύτερο σε αυτά τα 5 χιλ. δείγματα. Όμως, για μεγαλύτερα N το σφάλμα είναι μικρότερο στα 5 χιλ. δείγματα.



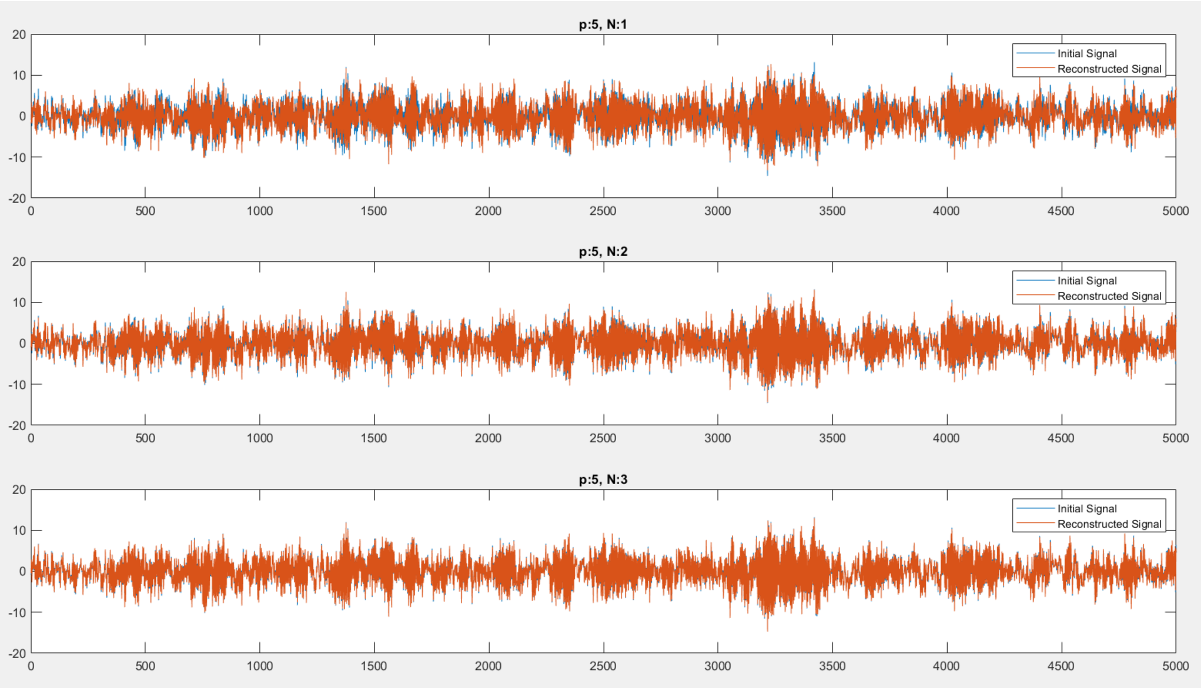
Συντελεστές του προβλέπτη για p=5:10 (Κάθε γραμμή αντιστοιχεί σε μία τιμή του p)

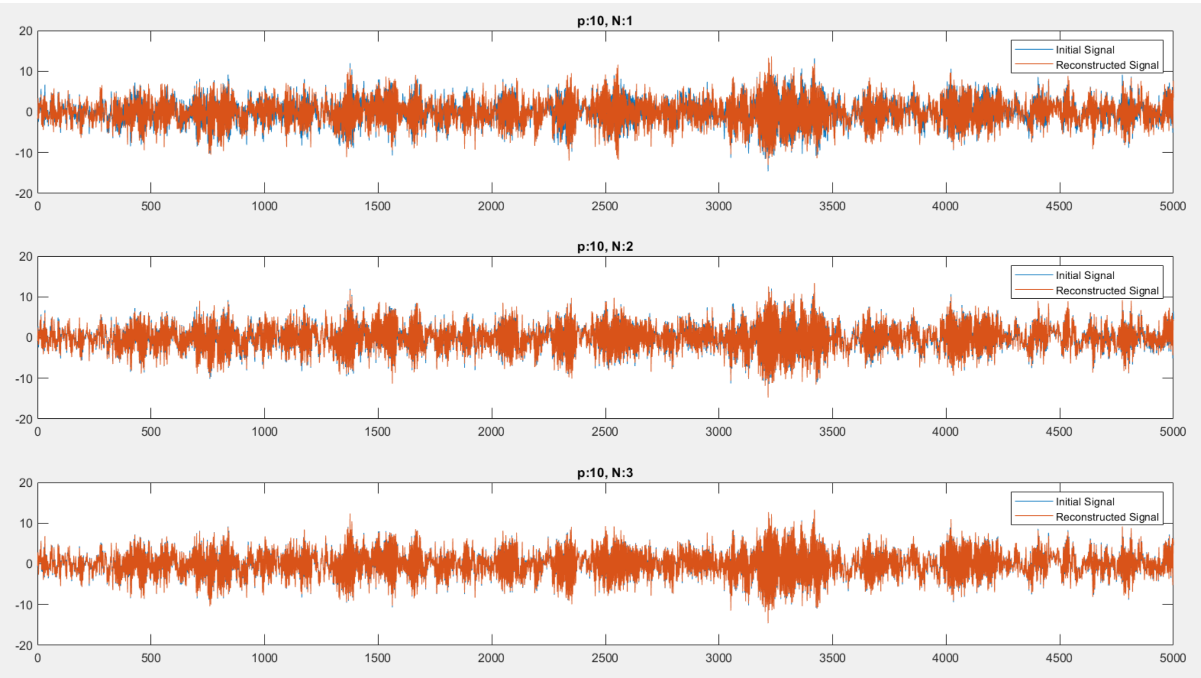
1.4297	-1.5547	1.2422	-0.3203	-0.0234	0	0	0	0	0
1.4297	-1.5547	1.2422	-0.3203	-0.0078	-0.0078	0	0	0	0
1.4297	-1.5547	1.2578	-0.3672	0.0234	-0.0547	0.0234	0	0	0
1.4297	-1.5547	1.2578	-0.3672	0.0391	-0.0547	0.0391	-0.0078	0	0
1.4297	-1.5547	1.2578	-0.3672	0.0234	-0.0391	0.0078	0.0234	-0.0078	0
1.4297	-1.5547	1.2578	-0.3672	0.0234	-0.0391	0.0078	0.0078	-0.0078	-0.0078

Οι συντελεστές του προβλέπτη αρχίζουν από αριθμούς κατά απόλυτη τιμή μεγαλύτερους της μονάδας και σταδιακά πλησιάζουν όλο και πιο κοντά στο 0. Αυτό γίνεται λόγω της μεγαλύτερης επίδρασης που έχουν τα γειτονικά δείγματα στην πρόβλεψη. Οι συντελεστές του δείγματος 5 και πιο πριν χρονικά είναι κοντά στο μηδέν που σημαίνει ότι τα δείγματα αυτά επηρεάζουν λίγο την τιμή του δείγματος που θέλουμε να προβλέψουμε.

Σε σχέση με τα 20 χιλιάδες δείγματα, για τα πρώτα 5 χιλιάδες δείγματα οι συντελεστές μετά το 5^{ο} δείγμα είναι πιο κοντά στο 0 απ’ ότι για τα 20 χιλιάδες δείγματα και οι πρώτοι συντελεστές είναι πιο μεγάλοι από αυτούς που βρήκαμε στα 20 χιλιάδες δείγματα.

4.





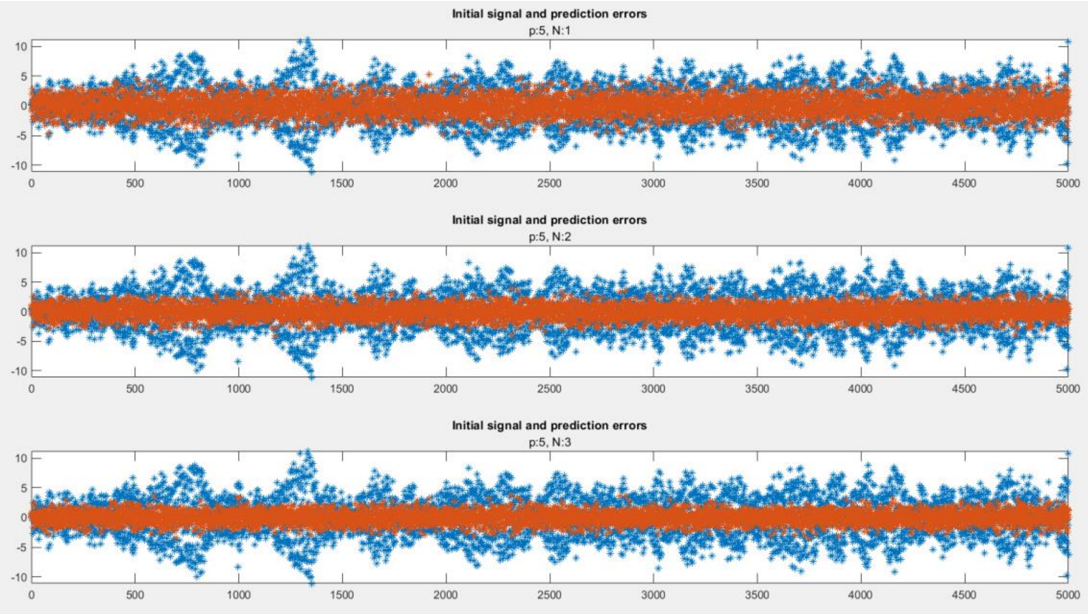
Παρατηρούμε ότι όσο μεγαλώνει το N τόσο μοιάζει περισσότερο το αρχικό σήμα με το ανακατασκευασμένο. Τα αποτελέσματα αυτά είναι παρόμοια με τα 20 χιλιάδες δείγματα.

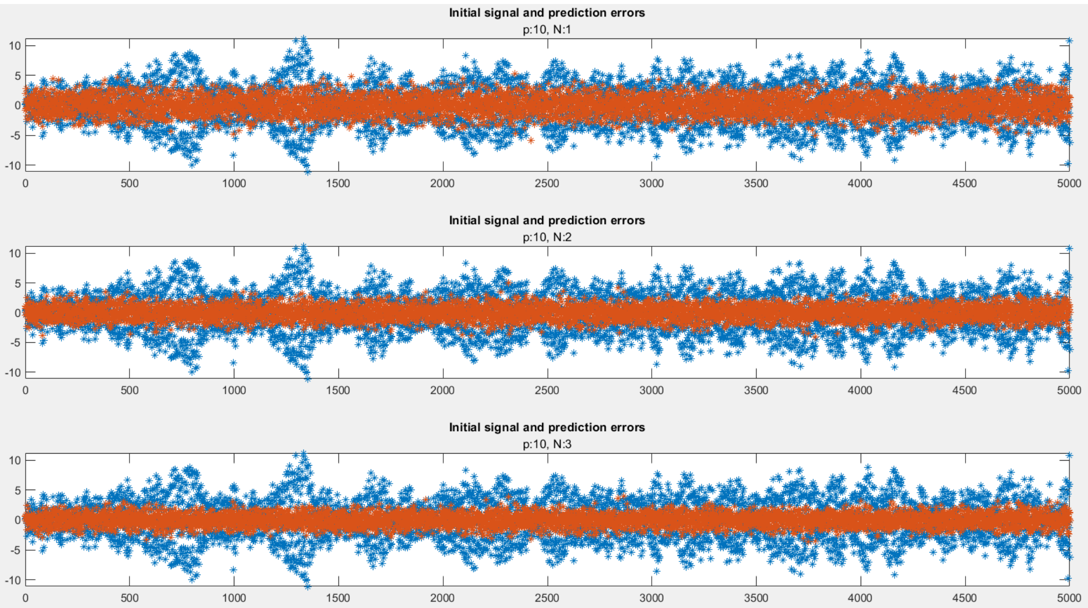
ΤΡΙΤΑ 5 ΧΙΛΙΑΔΕΣ ΔΕΙΓΜΑΤΑ

1.

Οκ.

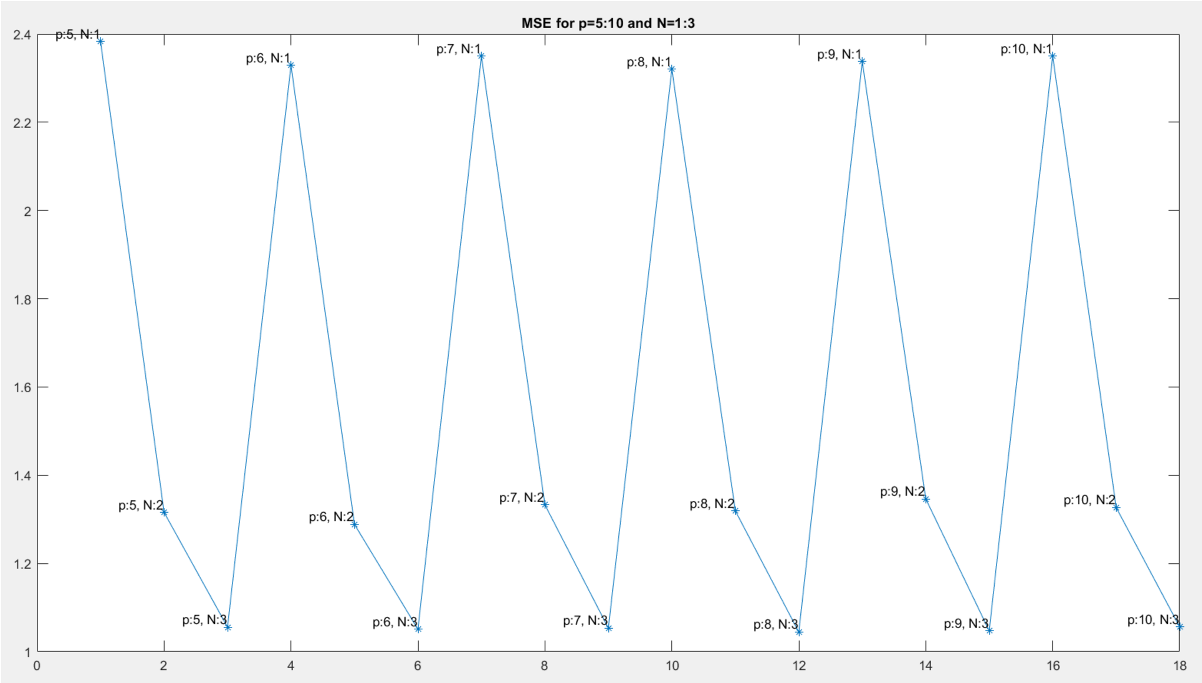
2.



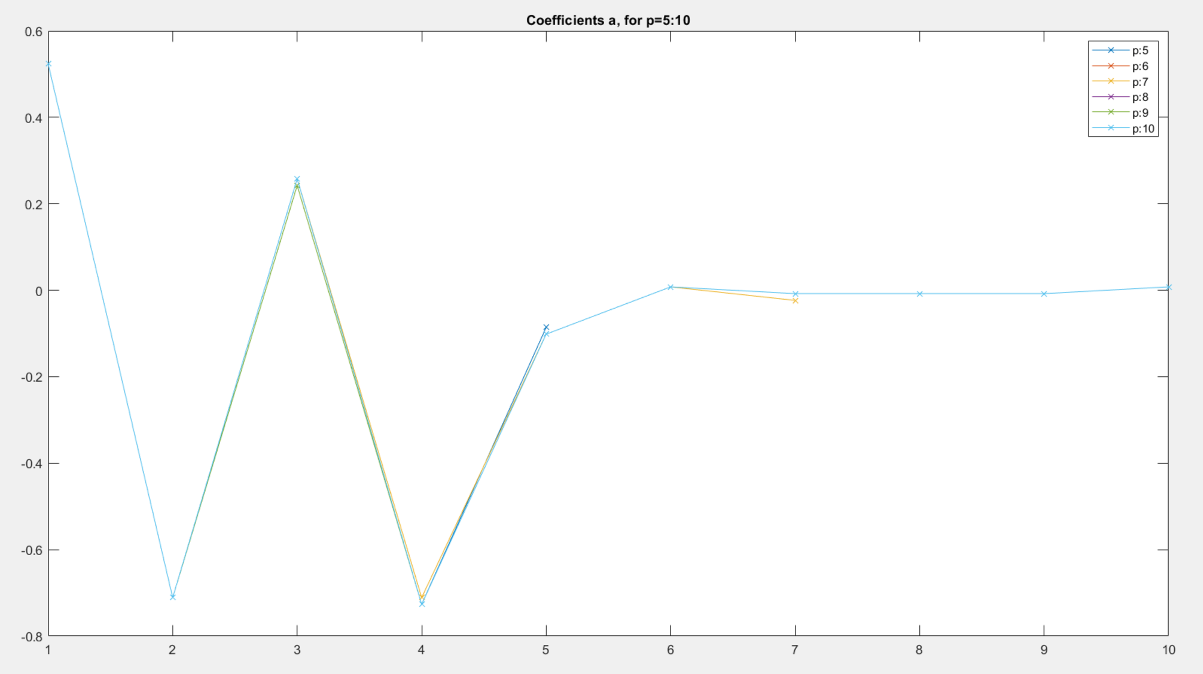


Παρόμοια αποτελέσματα με τα 20 χιλιάδες δείγματα. Οι δυναμικές περιοχές του σφάλματος πρόβλεψης φαίνονται λίγο μικρότερες απ΄ότι στα 20 χιλιάδες δείγματα.

3.



Το μέσο τετραγωνικό σφάλμα μειώνεται όταν αυξάνουμε το N κρατώντας το p σταθερό αλλά η απόδοση του συστήματος δεν είναι πολύ καλή διότι το μέσο τετραγωνικό σφάλμα δεν μειώνεται πάντα όταν αυξάνουμε το p κρατώντας το N σταθερό. Σε σχέση με τα 20 χιλιάδες δείγματα το σφάλμα είναι πάντα μικρότερο σε αυτά τα 5 χιλ. δείγματα.



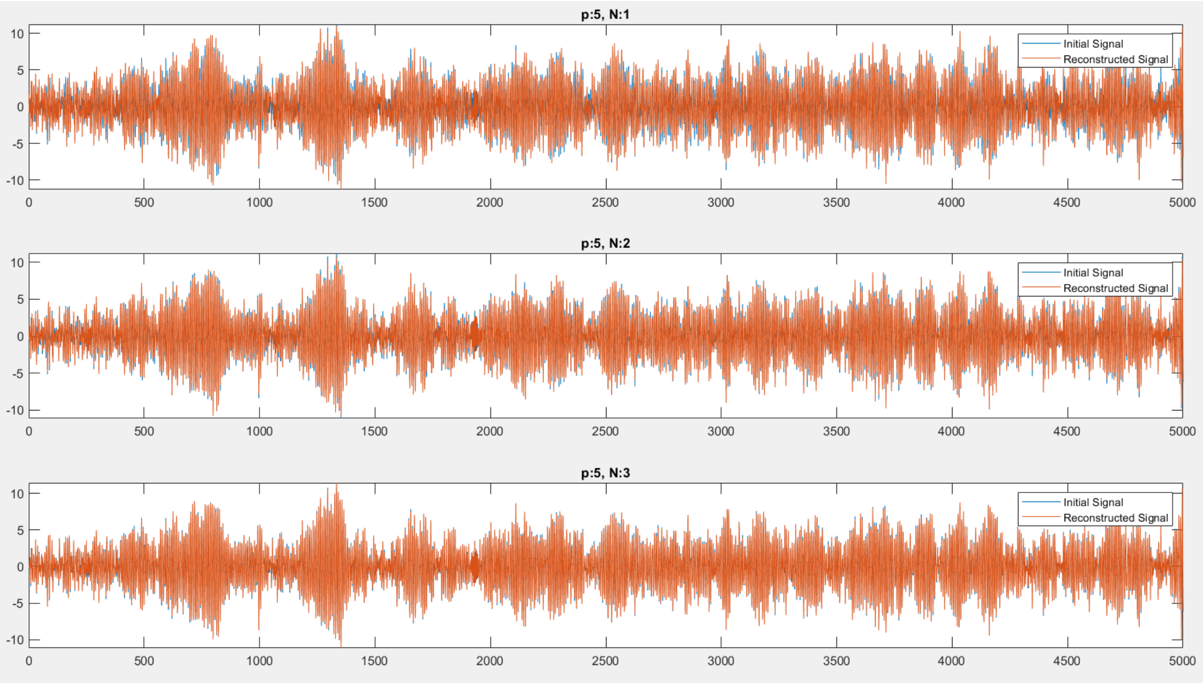
Συντελεστές του προβλέπτη για p=5:10 (Κάθε γραμμή αντιστοιχεί σε μία τιμή του p)

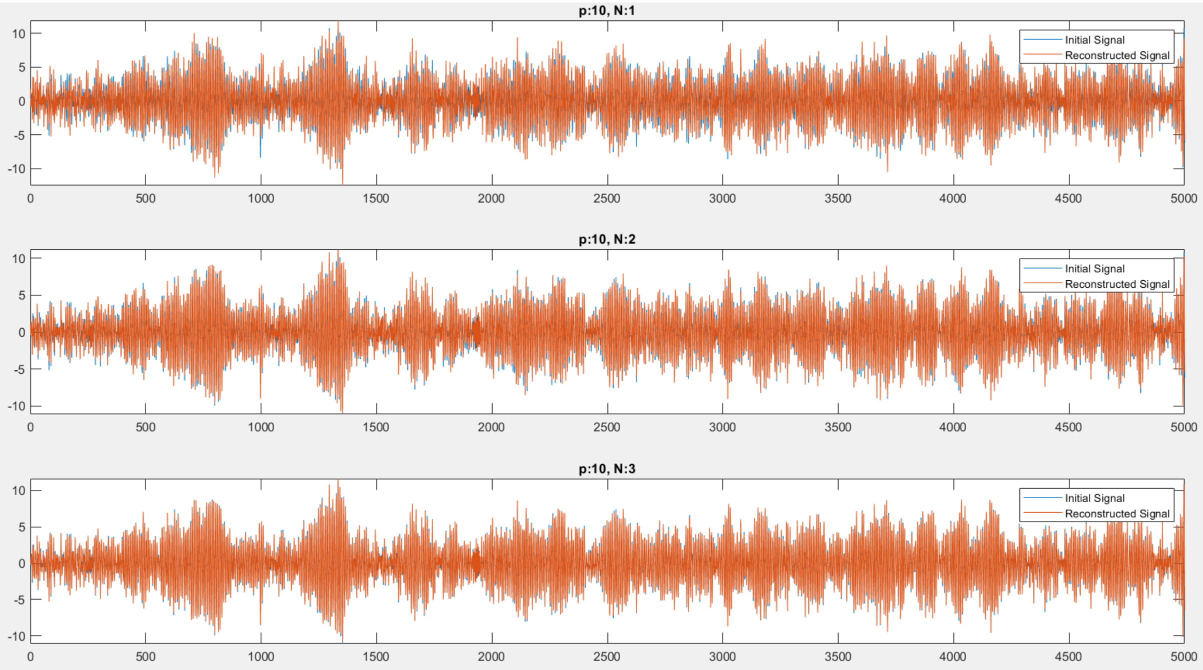
0.5234	-0.7109	0.2578	-0.7266	-0.0859	0	0	0	0	0
0.5234	-0.7109	0.2578	-0.7109	-0.1016	0.0078	0	0	0	0
0.5234	-0.7109	0.2578	-0.7109	-0.1016	0.0078	-0.0234	0	0	0
0.5234	-0.7109	0.2422	-0.7266	-0.1016	0.0078	-0.0078	-0.0078	0	0
0.5234	-0.7109	0.2422	-0.7266	-0.1016	0.0078	-0.0078	-0.0078	-0.0078	0
0.5234	-0.7109	0.2578	-0.7266	-0.1016	0.0078	-0.0078	-0.0078	-0.0078	0.0078

Οι συντελεστές του προβλέπτη αρχίζουν από αριθμούς κατά απόλυτη τιμή μεγαλύτερους της μονάδας και σταδιακά πλησιάζουν όλο και πιο κοντά στο 0. Αυτό γίνεται λόγω της μεγαλύτερης επίδρασης που έχουν τα γειτονικά δείγματα στην πρόβλεψη. Οι συντελεστές του δείγματος 5 και πιο πριν χρονικά είναι κοντά στο μηδέν που σημαίνει ότι τα δείγματα αυτά επηρεάζουν λίγο την τιμή του δείγματος που θέλουμε να προβλέψουμε.

Σε σχέση με τα 20 χιλιάδες δείγματα, για τα πρώτα 5 χιλιάδες δείγματα οι συντελεστές μετά το $5^{ο}$ δείγμα είναι πιο κοντά στο 0 απ’ ότι για τα 20 χιλιάδες δείγματα και οι πρώτοι συντελεστές είναι πιο μεγάλοι από αυτούς που βρήκαμε στα 20 χιλιάδες δείγματα.

4.





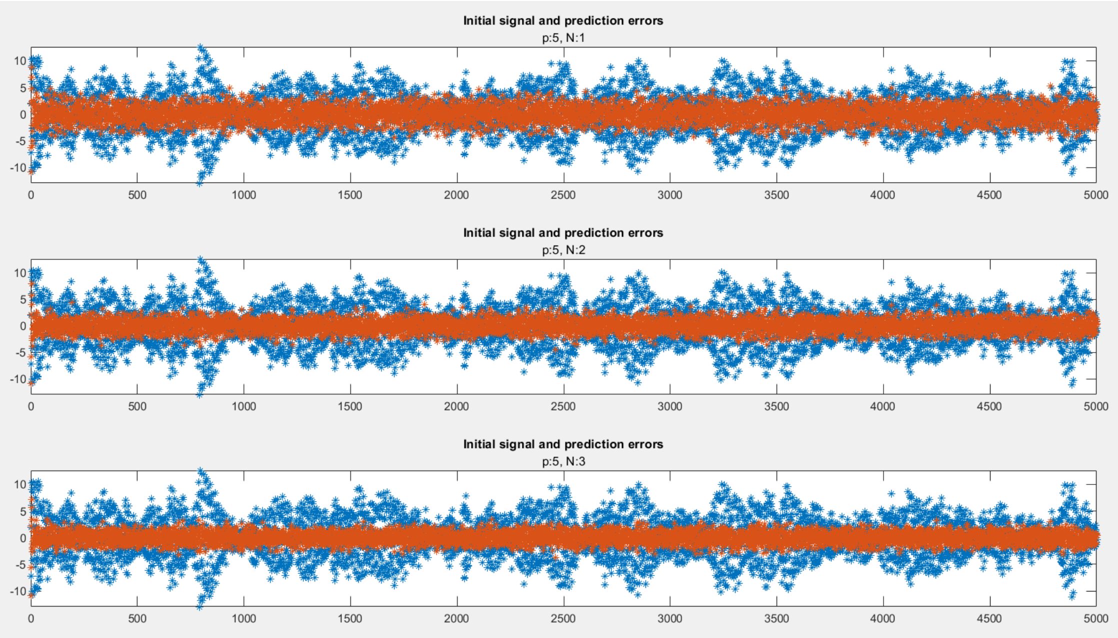
Παρατηρούμε ότι όσο μεγαλώνει το N τόσο μοιάζει περισσότερο το αρχικό σήμα με το ανακατασκευασμένο. Τα αποτελέσματα αυτά είναι παρόμοια με τα 20 χιλιάδες δείγματα.

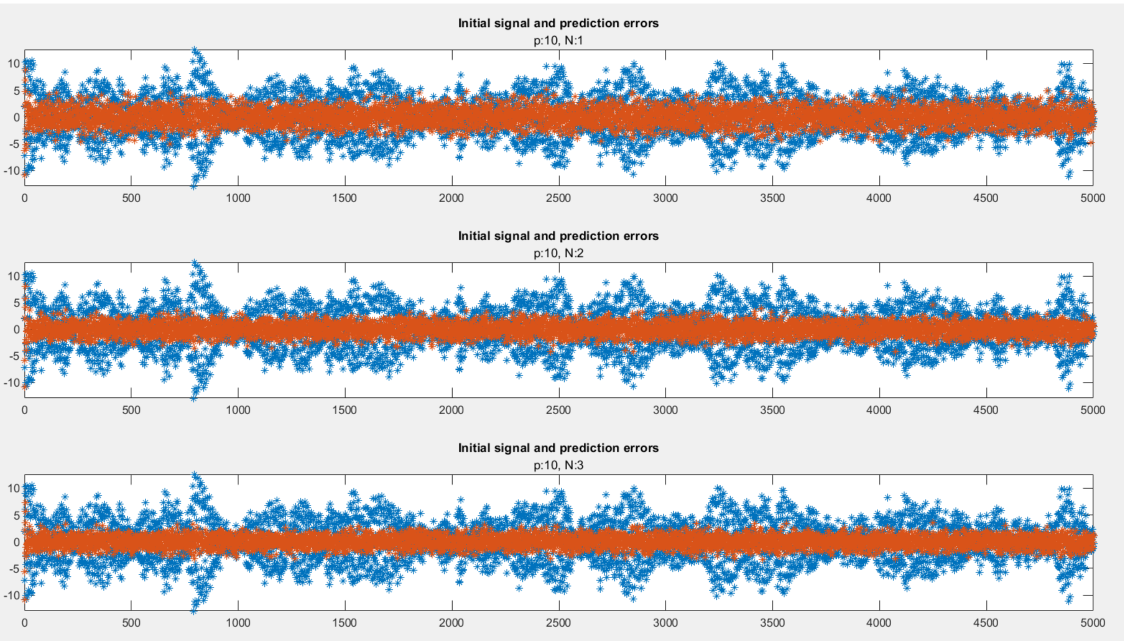
ΤΕΛΕΥΤΑΙΑ 5 ΧΙΛΙΑΔΕΣ ΔΕΙΓΜΑΤΑ

1.

Οκ.

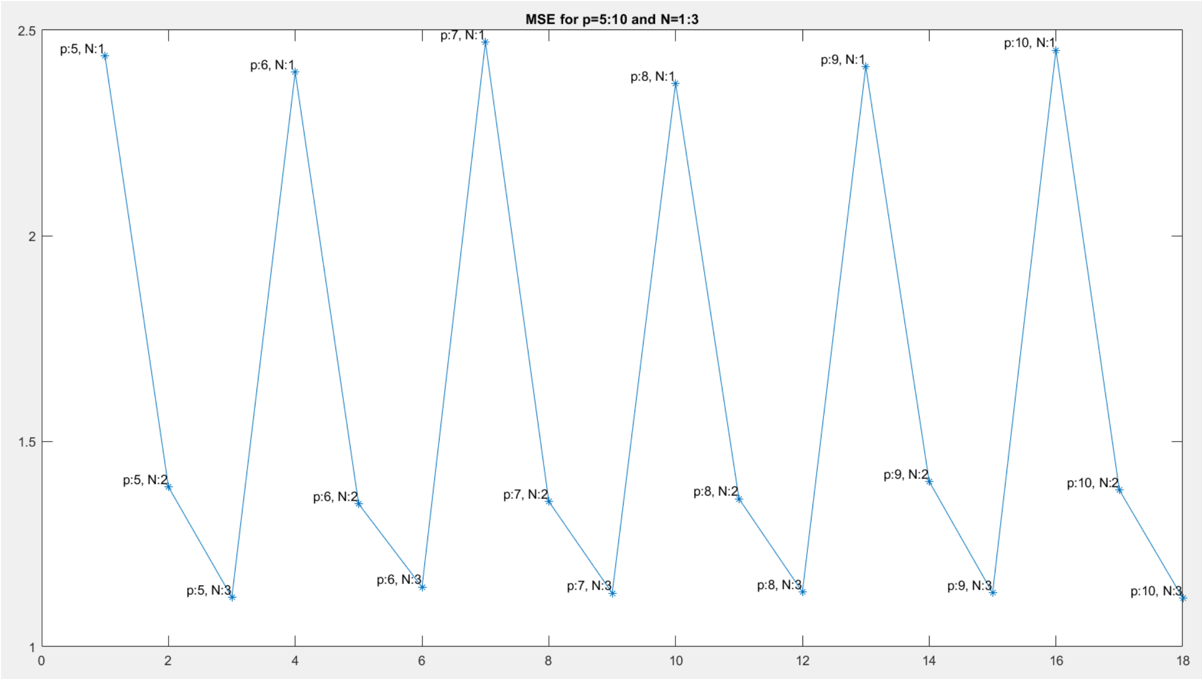
2.



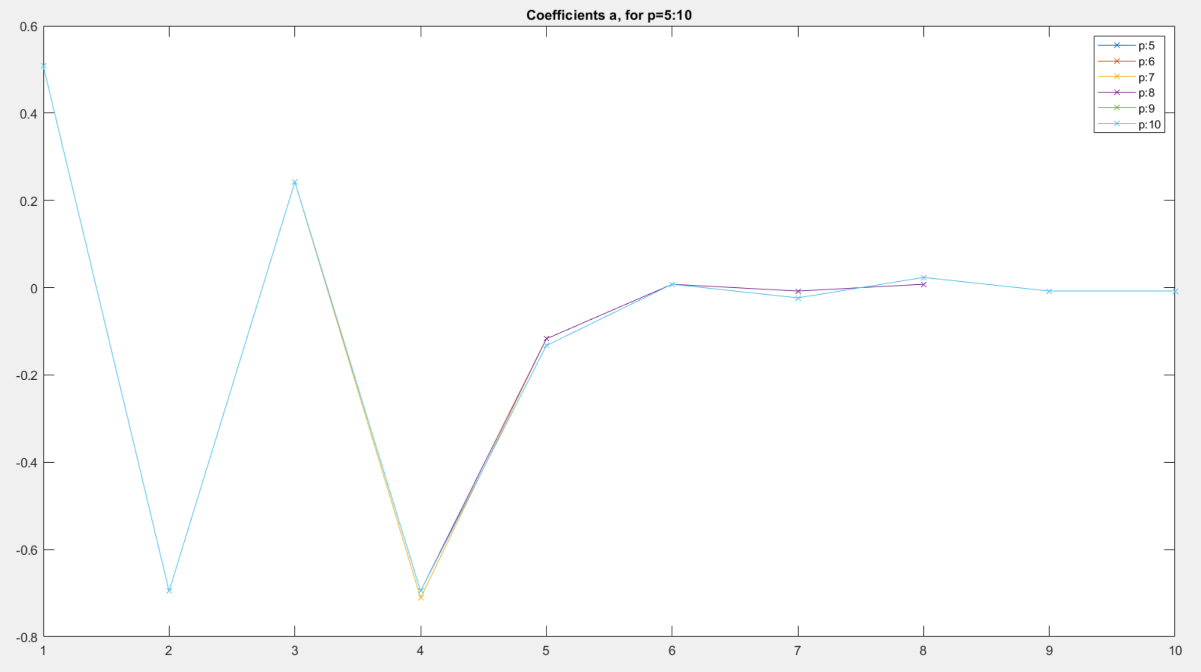


Παρόμοια αποτελέσματα με τα 20 χιλιάδες δείγματα. Οι δυναμικές περιοχές του σφάλματος πρόβλεψης φαίνονται λίγο μικρότερες απ΄ότι στα 20 χιλιάδες δείγματα.

3.



Το μέσο τετραγωνικό σφάλμα μειώνεται όταν αυξάνουμε το N κρατώντας το p σταθερό αλλά η απόδοση του συστήματος δεν είναι πολύ καλή διότι το μέσο τετραγωνικό σφάλμα δεν μειώνεται πάντα όταν αυξάνουμε το p κρατώντας το N σταθερό. Σε σχέση με τα 20 χιλιάδες δείγματα το σφάλμα είναι πάντα μικρότερο σε αυτά τα 5 χιλ. δείγματα.



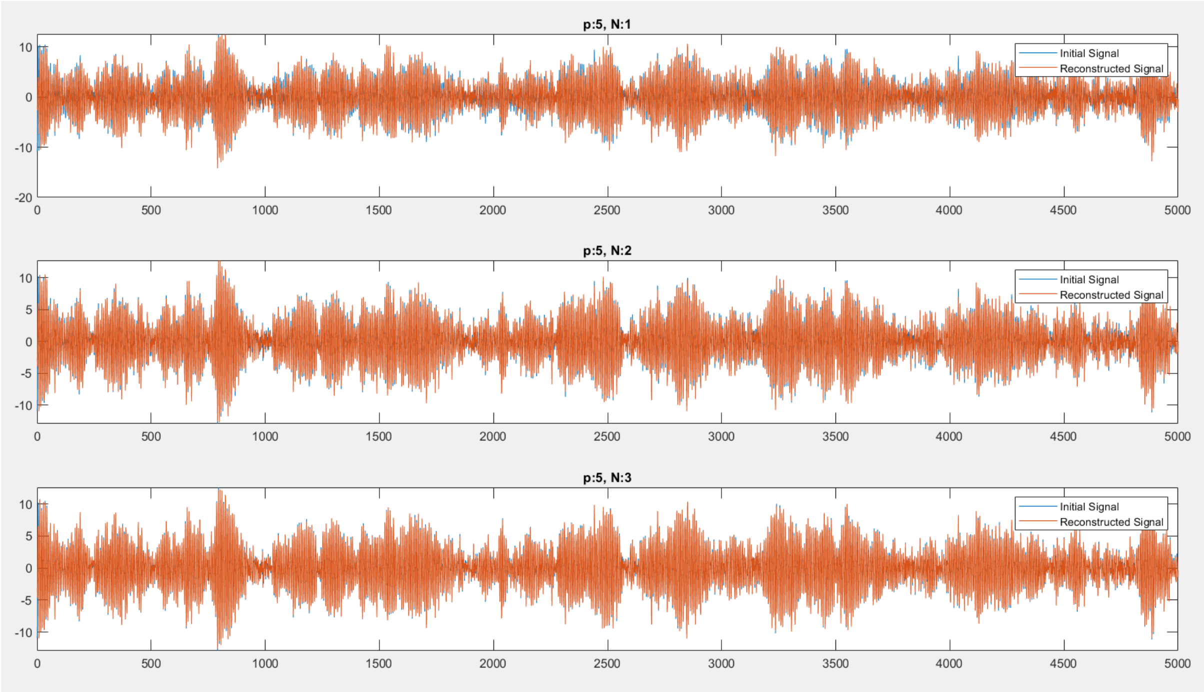
Συντελεστές του προβλέπτη για p=5:10 (Κάθε γραμμή αντιστοιχεί σε μία τιμή του p)

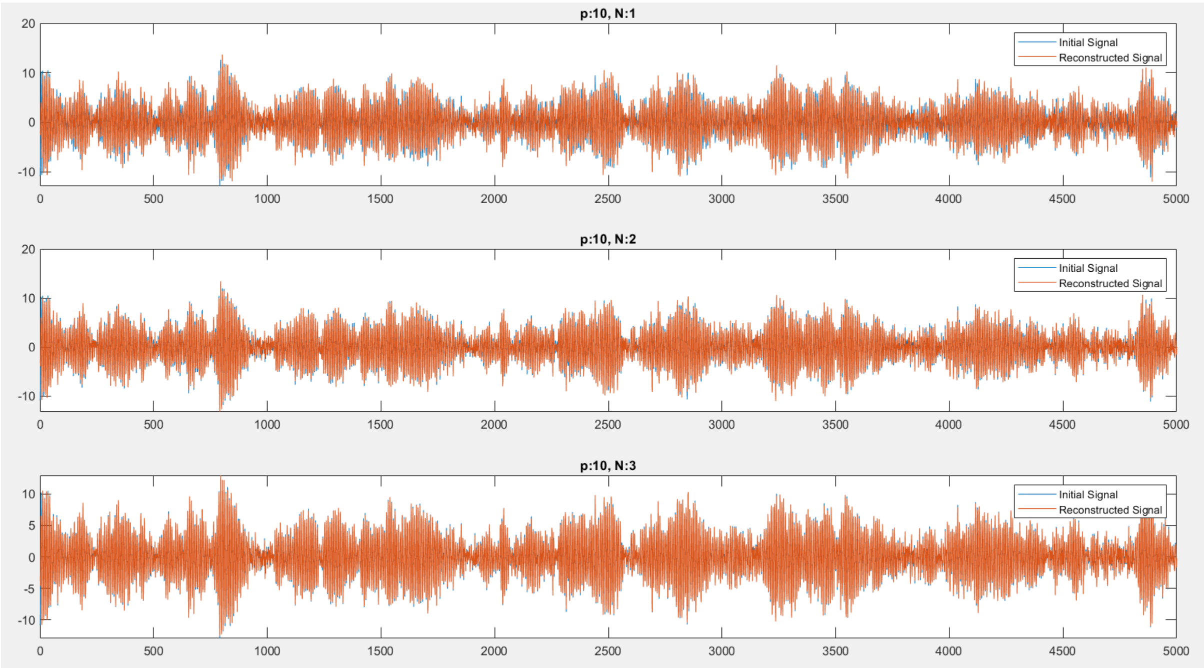
0.5078	-0.6953	0.2422	-0.7109	-0.1172	0	0	0	0	0
0.5078	-0.6953	0.2422	-0.7109	-0.1172	0.0078	0	0	0	0
0.5078	-0.6953	0.2422	-0.7109	-0.1172	0.0078	-0.0078	0	0	0
0.5078	-0.6953	0.2422	-0.6953	-0.1172	0.0078	-0.0078	0.0078	0	0
0.5078	-0.6953	0.2422	-0.6953	-0.1328	0.0078	-0.0234	0.0234	-0.0078	0
0.5078	-0.6953	0.2422	-0.6953	-0.1328	0.0078	-0.0234	0.0234	-0.0078	-0.0078

Οι συντελεστές του προβλέπτη αρχίζουν από αριθμούς κατά απόλυτη τιμή μεγαλύτερους της μονάδας και σταδιακά πλησιάζουν όλο και πιο κοντά στο 0. Αυτό γίνεται λόγω της μεγαλύτερης επίδρασης που έχουν τα γειτονικά δείγματα στην πρόβλεψη. Οι συντελεστές του δείγματος 5 και πιο πριν χρονικά είναι κοντά στο μηδέν που σημαίνει ότι τα δείγματα αυτά επηρεάζουν λίγο την τιμή του δείγματος που θέλουμε να προβλέψουμε.

Σε σχέση με τα 20 χιλιάδες δείγματα, για τα πρώτα 5 χιλιάδες δείγματα οι συντελεστές μετά το $5^{ο}$ δείγμα είναι πιο κοντά στο 0 απ’ ότι για τα 20 χιλιάδες δείγματα και οι πρώτοι συντελεστές είναι πιο μεγάλοι από αυτούς που βρήκαμε στα 20 χιλιάδες δείγματα.

4.





Παρατηρούμε ότι όσο μεγαλώνει το N τόσο μοιάζει περισσότερο το αρχικό σήμα με το ανακατασκευασμένο. Τα αποτελέσματα αυτά είναι παρόμοια με τα 20 χιλιάδες δείγματα.

Κώδικας

```
% Change arguments according to do_everything documentation (comments)
do_everything(2,1)


function do_everything(question_num, source_mode)
% question_num: 2 for question 2, 3 for question 3 and 4 for question 4
% source_mode: 1 for the initial source, 2 for first 5000 samples, 3 for
% second 5000 samples, 4 for third 5000 samples, 5 for last 5000 samples
    S = load('source.mat').t;

    if source_mode==2
        S = S(1:5000);
    elseif source_mode==3
        S = S(5001:10000);
    elseif source_mode==4
        S = S(10001:15000);
    elseif source_mode==5
        S = S(15001:20000);
    end
    s = size(S,1);

    if question_num==2
        for p=[5,10]
            figure()
            for N=[1,2,3]
                [~, ~, y] = transmitter(S, p, N);
                subplot(3,1,N)
                plot(1:s, S, '*')
                hold on
                plot(1:s, y, '*')
                hold off
                title('Initial signal and prediction errors',...
                    ['p:',int2str(p), ', N:', int2str(N)])
            end
        end
    elseif question_num==3
        pc = 1;    % point counter
        points_y = zeros(6*3, 1);
        labels = '';
        for p=5:10
            for N=1:3
                [~, ~, y] = transmitter(S, p, N);
                y = y.^2;
                y = sum(y)/size(y,1);

                points_y(pc)=y;
                labels = [labels 'p:' int2str(p) ', N:' int2str(N),'-'];
                pc = pc + 1;
            end
        end
        figure()
        plot(1:pc-1, points_y', "-*")
        labels = split(labels,'-');
        text(1:pc-1,points_y',labels(1:18),'VerticalAlignment', ...
```

```
            'bottom','HorizontalAlignment','right')
        title('MSE for p=5:10 and N=1:3')

        as = zeros(6,10);
        legend_txt = '';
        figure()
        for p=5:10
            [~, a, ~] = transmitter(S,p,3);
            as(p-4,1:p) = a';
            plot(1:p,a,'-x')
            legend_txt = [legend_txt,'p:',int2str(p),'-'];
            hold on
        end
        hold off

        legend_txt = split(legend_txt, '-');
        legend(legend_txt(1:end-1))
        title('Coefficients a, for p=5:10')
        disp('Coefficients:')
        disp(as)
    elseif question_num==4
        for p=[5,10]
            figure()
            for N=1:3
                [yc, aq, ~] = transmitter(S, p, N);
                S_tonos = receiver(yc, aq, p);
                subplot(3,1,N)
                plot(S,'-')
                hold on
                subplot(3,1,N)
                plot(S_tonos,'-')
                hold off
                legend('Initial Signal','Reconstructed Signal')
                title(['p:' int2str(p) ', N:' int2str(N)])
            end
        end
    end
end


function [y_caps,aq,ys] = transmitter(x, p, N)
    a = coef(x,p);
    aq = zeros(size(a,1),1);
    for i=1:p
        [aq_pos,aq_centers] = my_quantizer(a(i),8,-2,2);
        aq(i) = aq_centers(aq_pos);
    end

    memory = zeros(p, 1);
    y_caps = zeros(size(x,1),1);
    ys = zeros(size(x,1),1);

    for i=1:size(x,1)

        y_tonos = prediction(memory, aq, p);

        y = x(i)-y_tonos;
        ys(i)=y;
```

```
        [y_cap_pos, y_cap_centers] = my_quantizer(y, N, -3.5, 3.5);
        y_cap = y_cap_centers(y_cap_pos);
        y_caps(i) = y_cap;

        y_cap_tonos = y_cap + y_tonos;

        memory = [memory(2:p); y_cap_tonos];
    end
end


function [y_cap_tonos] = receiver(y_caps, aq, p)
    y_cap_tonos = zeros(size(y_caps,1), 1);
    memory = zeros(p, 1);

    for i=1:size(y_caps,1)
        y_tonos = prediction(memory, aq, p);

        y_cap_tonos(i) = y_tonos + y_caps(i);

        memory = [memory(2:p); y_cap_tonos(i)];
    end
end


function [a,R,r] = coef(x, p)
    r = zeros(p,1);
    for i=1:p
        r(i) = autocorrelation(x,size(x,1),p,i,0);
    end

    R = zeros(p,p);
    for i=0:p-1
        for j=0:p-1
            R(i+1,j+1) = autocorrelation(x,size(x,1),p,i,j);
        end
    end

    a = R \ r;
end


function [Rij] = autocorrelation(x,N,p,i,j)
    Rij = 0;
    for n=p+1:N
        Rij = Rij + x(n-i)*x(n-j);
    end
    Rij = Rij/(N-p);
end


function [y_tonos] = prediction(y_cap_tonos, aq, p)
    y_tonos = 0;
    for i=1:p
        y_tonos = y_tonos + aq(i)*y_cap_tonos(p-i+1);
    end
end
```

```
function [pos, centers] = my_quantizer(yn,N,min_value,max_value)
    if yn>max_value
        yn=max_value;
    elseif yn<min_value
        yn=min_value;
    end

    levels = 2^N;
    delta = (max_value-min_value)/levels;

    centers(1) = max_value - (delta/2);
    for i=2:levels
        centers(i) = centers(i-1) - delta;
    end

    for i=1:size(centers')
        if (centers(i)+(delta/2)>=yn) && (centers(i)-(delta/2)<=yn)
            pos=i;
            break;
        end
    end
end
```