ΤΕΧΝΟΛΟΓΙΚΗ ΠΡΟΒΛΕΨΗ

Σκοπός της εργασίας είναι να ελεγχθούν 7 μοντέλα πρόβλεψης μέσω των οποίων θα καταλήξουμε στο βέλτιστο , για το συγκεκριμένο σύνολο δεδομένων που μας δωθηκε. Η σύγκριση θα γίνει μέσω των σφαλμάτων που προκύπτουν από την εφαρμογή του κάθε μοντέλου στο πρόγραμμα matlab.

1^{Ο} ΚΕΦΑΛΑΙΟ

- **ΜΟΝΤΕΛΟ AR**

  Το AR είναι ένα μοντέλο αυτοπαλινδρόμησης. Οι παράμετροι του AR μοντέλου με τη χρήση μεθόδων ελαχίστων τετραγώνων.
  Στο συγκεκριμένο μοντέλο χρησιμοποιήσαμε τα πρώτα 200 για να κάνουμε «training» στα δεδομένα. Στα υπόλοιπα δεδομένα έγινε η πρόβλεψη .
  Δοκιμάσαμε να γίνει training σε 130 αλλά προέκυπταν μεγαλύτερα σφάλματα.

Κώδικας στη matlab:

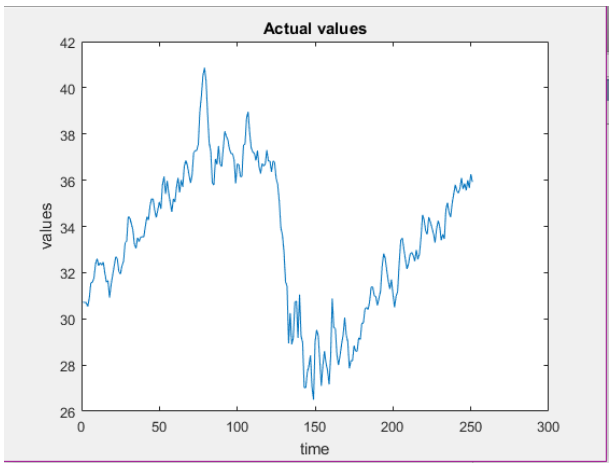
```
%ΜΠΕΡΜΠΑΤΗ ΑΛΕΞΑΝΔΡΑ 2013010143
%ΚΑΡΑΜΠΕΛΑ ΚΩΝΣΤΑΝΤΙΝΑ 2013010014
% AR  MODEL

close all  %clean the workspace
clear
clc

mydata=xlsread ('MPERMPATI.xlsx', 'E1:E252'); % retrieves the E column data of range 1 to 252

figure(1) % a view of data
plot (mydata)
xlabel('time'); ylabel('values')
title('Actual values')

%  Estimation of an AR model
y=mydata

y1=mydata(1:200) % TRAINING DATA
y2=mydata(201:end)% TESTING DATA

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

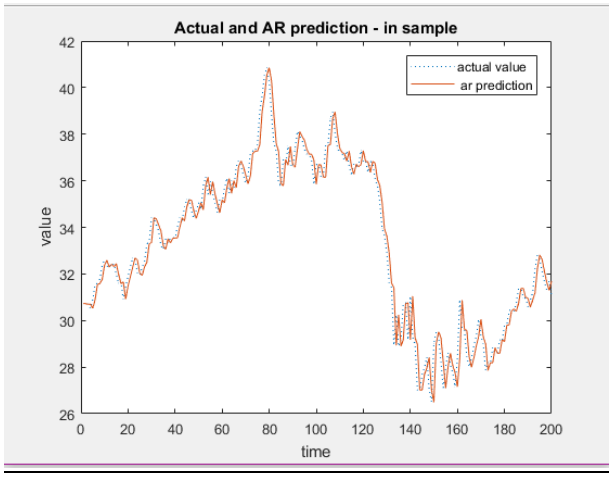
```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
m_ar=ar(y1,1, 'ls') % estimation of the model

fpe_ar=fpe(m_ar) % estimation of Akaike's Final prediction Error (FPE)
aic_ar=aic(m_ar) % estimation of Akaike's Information criterion (AIC)

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% in sample evaluation
yhat_ar_insampleIP=predict(m_ar, y1,1) %predicts the in sample data

%AA=yhat_ar_insampleIP{1,:} % it reads the 'cell' type

%yhat_ar_insample=AA
yhat_ar_insample=yhat_ar_insampleIP

t1=1:(length(yhat_ar_insample))
figure(2)
plot(t1,y1,':', t1,yhat_ar_insample,'-');
legend('actual value',' ar prediction')
xlabel('time')
ylabel('value')
title('Actual and AR prediction - in sample')
```

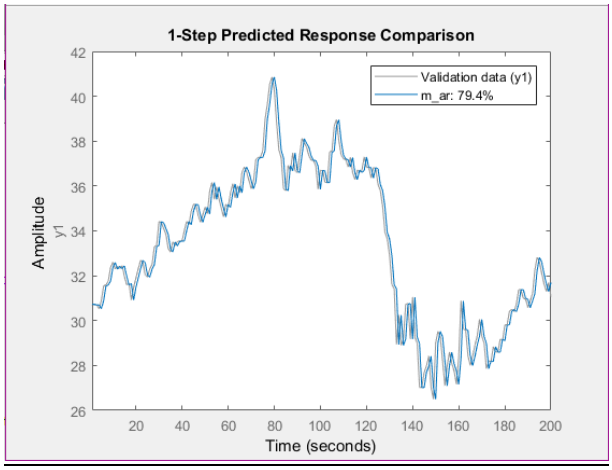
```
figure (3) % in sample prediction and plotting
compare(y1, m_ar, 1)% it is another way to predict and plot directly

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%prediction by AR model out of sample

yhat_arIP=predict(m_ar, y2,1) %predicts the unseen data

%AAA=yhat_arIP{1,:}

%yhat_ar=AAA

yhat_ar=yhat_arIP
t=1:(length(yhat_ar))

figure(4)
plot(t,y2,'b-s', t,yhat_ar,'r-x');
legend('actual value',' ar prediction')
xlabel('time')
ylabel('value')
title('Actual and AR prediction -out of sample')
```

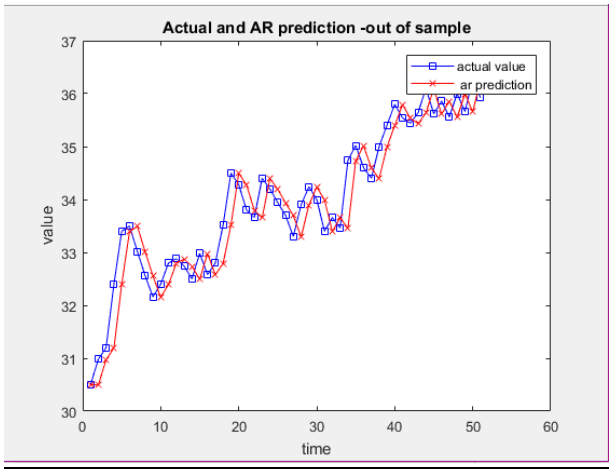
```
figure (5)
compare(y2, m_ar, 1)% it is another way to predict and plot directly

[y2 yhat_ar]% prints the  actual and predicted value

get(m_ar) % gives model information
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% error measures calculation

% Mean Square Error (MSE)
MSE_ar=(1/length(yhat_ar))*norm(y2-yhat_ar)^2

% Root Mean Square Error (RMSE)
RMSE_ar=sqrt(norm(y2-yhat_ar)^2/length(y2-yhat_ar))

% Mean Absolute Error (MAE)
MAE_ar=(1/length(y2-yhat_ar))*sum(abs(y2-yhat_ar))

% Mean Absolute Percentage Error (MAPE)
MAPE_ar=(100/length(y2-yhat_ar))*sum(abs(y2-yhat_ar)./abs(y2))

%end
```

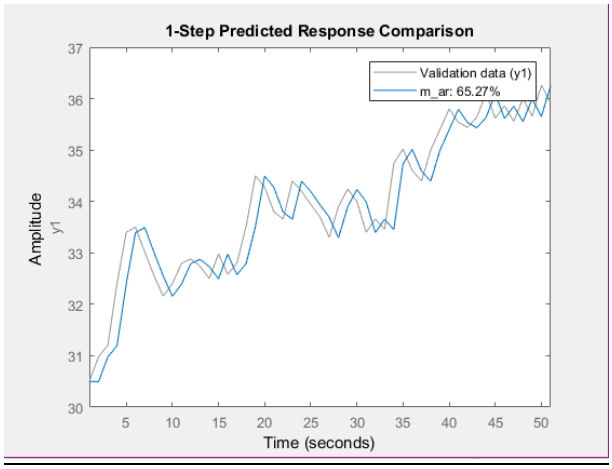
<u>Διαγράμματα μοντέλου AR :</u>




Actual and AR prediction - in sample
actual value
ar prediction
value
time


1-Step Predicted Response Comparison
Validation data (y1)
m_ar: 79.4%
Amplitude
y1
Time (seconds)


Actual and AR prediction -out of sample
actual value
ar prediction
value
time


1-Step Predicted Response Comparison
Validation data (y1)
m_ar: 65.27%
Amplitude
y1
Time (seconds)

2^{ο} ΚΕΦΑΛΑΙΟ

- **ΜΟΝΤΕΛΟ ARX**
  Το ARX είναι ένα δυναμικό μοντέλο , αφού εξαρτάται από τα αποτελέσματα της προηγούμενης στιγμής , όπως και το Ar . Χρησιμοποιείται όταν έχουμε πολυμεταβλητότητα.

Κώδικας στη matlab:

```
%ΜΠΕΡΜΠΑΤΗ ΑΛΕΞΑΝΔΡΑ 2013010143
%ΚΑΡΑΜΠΕΛΑ ΚΩΝΣΤΑΝΤΙΝΑ 2013010014

close all
clear
clc

mydata=xlsread ('MPERMPATI.xlsx', 'e1:e252') % retrieves the E column data of range 1 to 252 fr

figure(1) % a view of data
plot (mydata)
xlabel('time'); ylabel('values')
title('Actual values')
```

```
%  Estimation of an ARX model
y=mydata

y1=mydata(1:200) % TRAINING DATA
y2=mydata(201:end)% TESTING DATA


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
m_arx=arx(y1,1) % estimation of the model

fpe_arx=fpe(m_arx) % estimation of Akaike's Final prediction Error (FPE)
aic_arx=aic(m_arx) % estimation of Akaike's Information criterion (AIC)


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% in sample evaluation
```

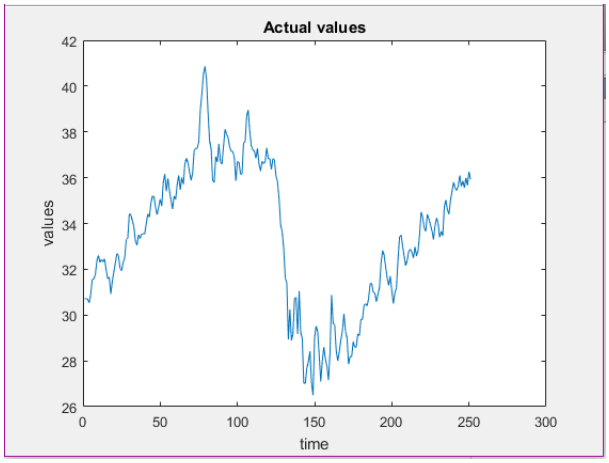
```
yhat_arx_insampleIP=predict(m_arx, y1,1) %predicts the in sample data
%yhat_arx_insampleIP=predict(nb, y1) %predicts the in sample data

%AA=yhat_arx_insampleIP{1,:};
AA=yhat_arx_insampleIP;

yhat_arx_insample=AA

t1=1:(length(yhat_arx_insample))
figure(2)
plot(t1,y1,':', t1,yhat_arx_insample,'-');
legend('actual value',' arx prediction')
xlabel('time')
ylabel('value')
title('Actual and ARX prediction - in sample')
```

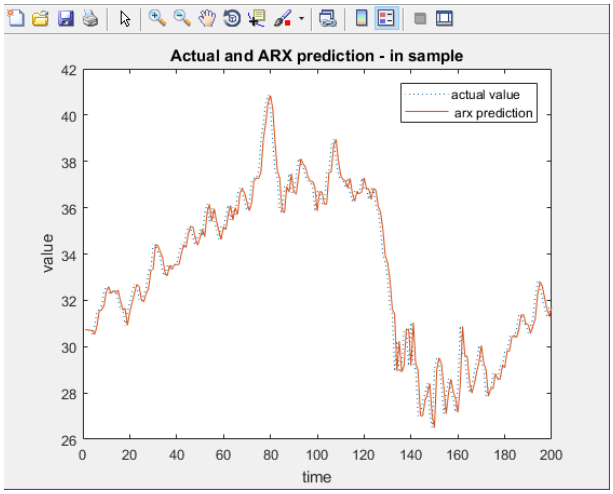
```
figure (3) % in sample prediction and plotting
compare(y1, m_arx, 1)% it is another way to predict and plot directly




%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%prediction by ARX model out of sample

yhat_arxIP=predict(m_arx, y2,1) %predicts the unseen data
%yhat_arxIP=predict(nb,y2,1) %predicts the unseen data

%BBB=yhat_arxIP{1,:};
BBB=yhat_arxIP;
```

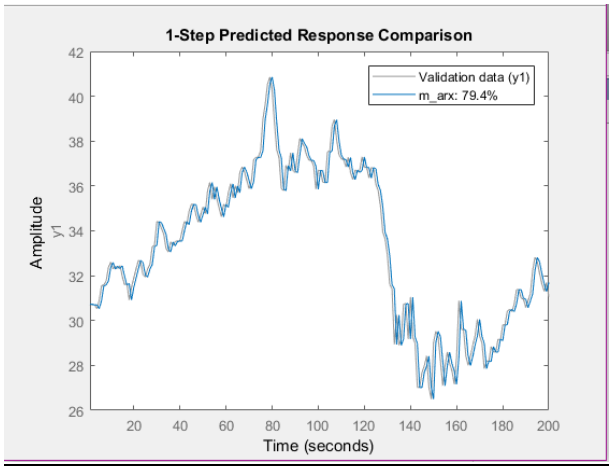
```
65
66 -   yhat_arx=BBB
67
68 -   t=1:(length(yhat_arx))
69 -   figure(4)
70 -   plot(t,y2,'b-s', t,yhat_arx,'r-x');
71 -   legend('actual value',' arx prediction')
72 -   xlabel('time')
73 -   ylabel('value')
74 -   title('Actual and ARX prediction -out of sample')
75
76
77 -   figure (5)
78 -   compare(y2, m_arx, 1)% it is another way to predict and plot directly
79
80 -   [y2 yhat_arx]% prints the  actual and predicted value
```

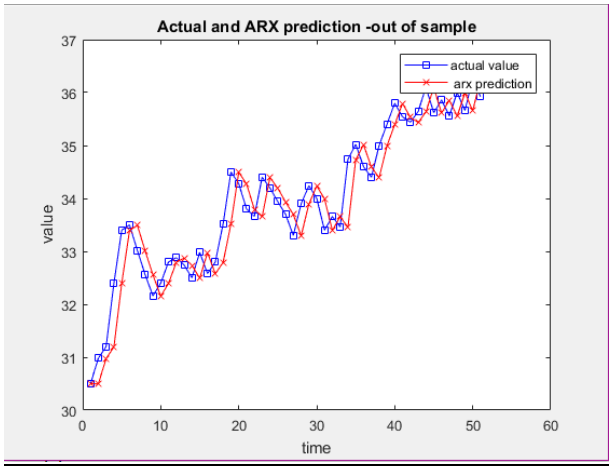
```
81 -   get(m_arx) % gives model information
82    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
83    % error measures calculation
84    % Mean Square Error (MSE)
85 -   MSE_arx=(1/length(yhat_arx))*norm(y2-yhat_arx)^2
86    % Root Mean Square Error (RMSE)
87 -   RMSE_arx=sqrt(norm(y2-yhat_arx)^2/length(y2-yhat_arx))
88    % Mean Absolute Error (MAE)
89 -   MAE_arx=(1/length(y2-yhat_arx))*sum(abs(y2-yhat_arx))
90    % Mean Absolute Percentage Error (MAPE)
91 -   MAPE_arx=(100/length(y2-yhat_arx))*sum(abs(y2-yhat_arx)./abs(y2))
92    %end
93
```

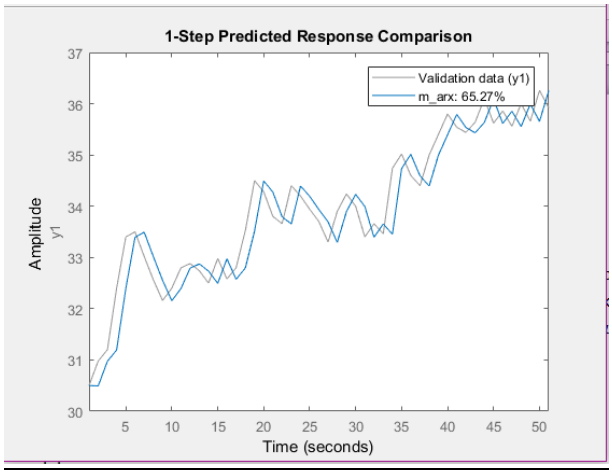
Διαγράμματα μοντέλου ARX :











3^{ο} ΚΕΦΑΛΑΙΟ

- **<u>ΜΟΝΤΕΛΟ ARMA</u>**

Είναι μοντέλο αυτοπαλινδρόμησης κινούμενου μέσου. Επηρεάζεται από εξωγενείς παράγοντες που μπορούν να αλλάξουν την χρονοσειρά.

<u>Κώδικας στη matlab:</u>

```
%ΜΠΕΡΜΠΑΤΗ ΑΛΕΞΑΝΔΡΑ 2013010143
%ΚΑΡΑΜΠΕΛΑ ΚΩΝΣΤΑΝΤΙΝΑ 2013010014

%The ARMA model is calculated for a scalar time series m0 =[na nc] .

close all
clear
clc

mydata=xlsread ('MPERMPATI.xlsx', 'e1:e252');


figure(1) % a view of data
plot (mydata)
xlabel('time'); ylabel('values')
title('Actual values')
```

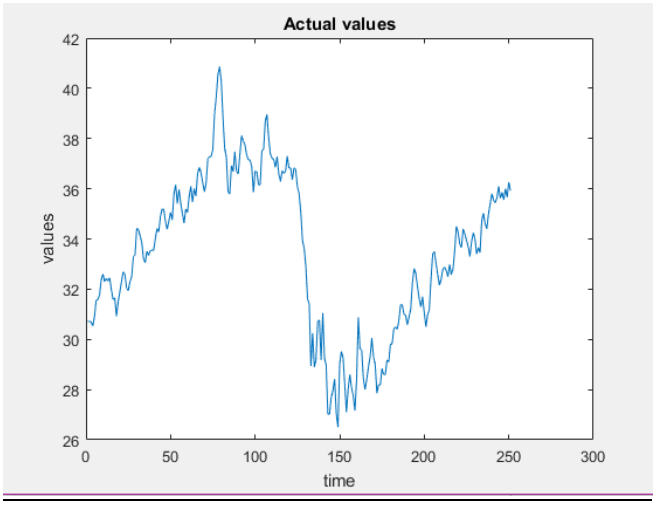
```
y=mydata;

y1=mydata(1:200); % TRAINING DATA
y2=mydata(201:end);% TESTING DATA


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
m_arma=armax(y1,[1,1]); % estimation of the model

fpe_arma=fpe(m_arma); % estimation of Akaike's Final prediction Error (FPE)
aic_arma=aic(m_arma); % estimation of Akaike's Information criterion (AIC)

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% in sample evaluation
yhat_arma_insampleIP=predict(m_arma, y1,1); %predicts the in sample data
```

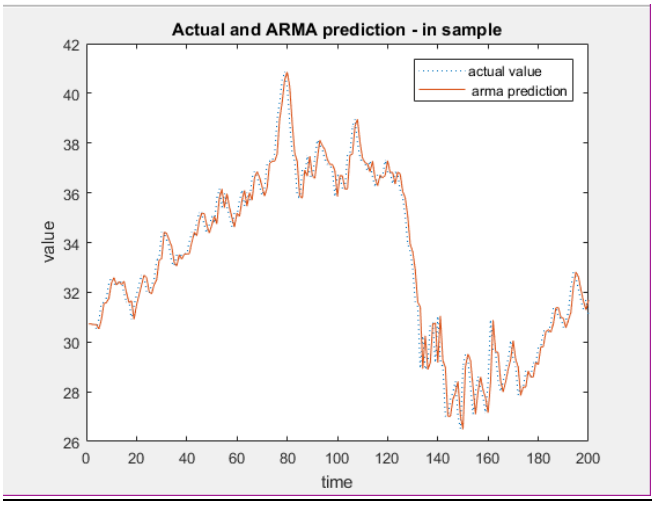
```
%AA=yhat_arma_insampleIP{1,:}
%yhat_arma_insample=AA
yhat_arma_insample=yhat_arma_insampleIP;


t1=1:(length(yhat_arma_insample));
figure(2)
plot(t1,y1,':', t1,yhat_arma_insample,'-');
legend('actual value',' arma prediction')
xlabel('time')
ylabel('value')
title('Actual and ARMA prediction - in sample')

figure (3) % in sample prediction and plotting
compare(y1, m_arma, 1)% it is another way to predict and plot directly
```

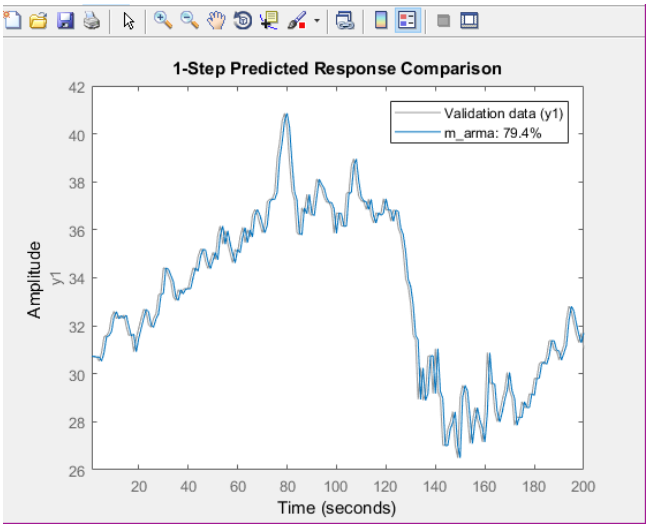
```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%prediction by ARMA model out of sample

yhat_armaIP=predict(m_arma, y2,1,1); %predicts the unseen data

%BB=yhat_armaIP{1,:};
%yhat_arma=BB
yhat_arma=yhat_armaIP;

t=1:(length(yhat_arma));
figure(4)
plot(t,y2,'b-s', t,yhat_arma,'r-x');
legend('actual value',' arma prediction')
xlabel('time')
ylabel('value')
title('Actual and ARMA prediction -out of sample')
```

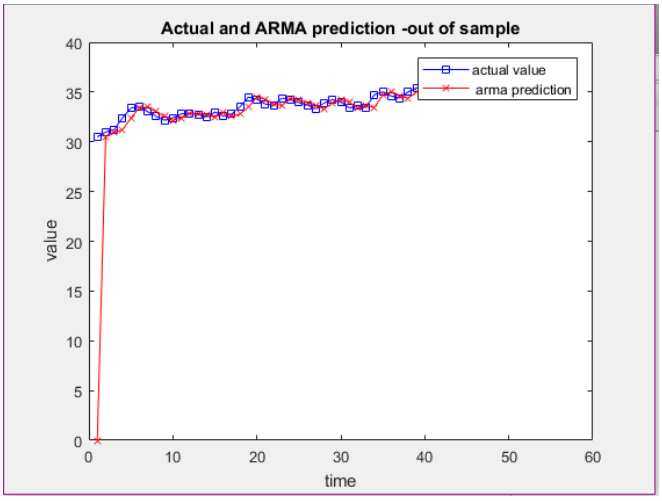
```
figure (5)
compare(y2, m_arma, 1)% it is another way to predict and plot directly
[y2 yhat_arma];% prints the  actual and predicted value
get(m_arma) % gives model information
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% error measures calculation
% Mean Square Error (MSE)
MSE_arma=(1/length(yhat_arma))*norm(y2-yhat_arma)^2;
% Root Mean Square Error (RMSE)
RMSE_arma=sqrt(norm(y2-yhat_arma)^2/length(y2-yhat_arma));
% Mean Absolute Error (MAE)
MAE_arma=(1/length(y2-yhat_arma))*sum(abs(y2-yhat_arma));
% Mean Absolute Percentage Error (MAPE)
MAPE_arma=(100/length(y2-yhat_arma))*sum(abs(y2-yhat_arma)./abs(y2));
%end
```

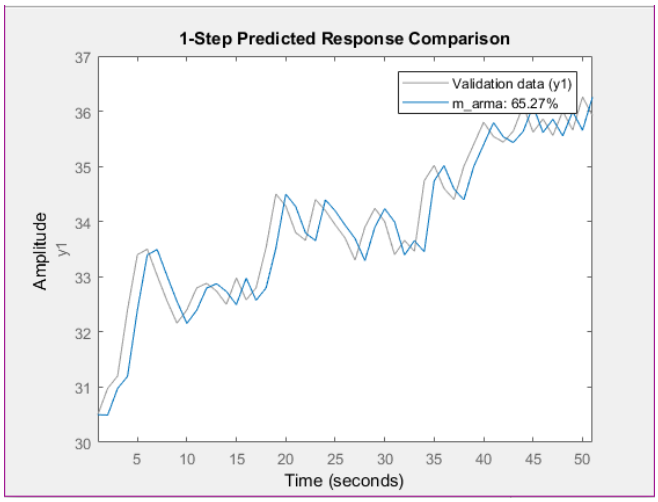
Διαγράμματα μοντέλου ARMA :


Actual values
values
time
42
40
38
36
34
32
30
28
26
0
50
100
150
200
250
300


Actual and ARMA prediction - in sample
actual value
arma prediction
value
time
42
40
38
36
34
32
30
28
26
0
20
40
60
80
100
120
140
160
180
200


1-Step Predicted Response Comparison
Validation data (y1)
m_arma: 79.4%
Amplitude
y1
Time (seconds)
42
40
38
36
34
32
30
28
26
20
40
60
80
100
120
140
160
180
200


Actual and ARMA prediction -out of sample
actual value
arma prediction
value
time
40
35
30
25
20
15
10
5
0
0
10
20
30
40
50
60



4^{ο} ΚΕΦΑΛΑΙΟ

- **ΜΟΝΤΕΛΟ OUTPUT ERROR**

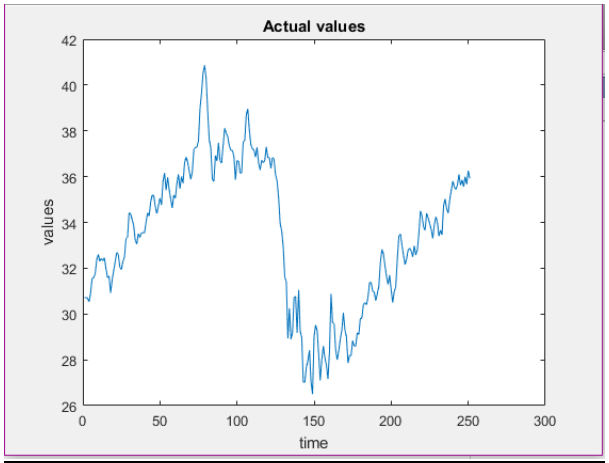
  Το συγκεκριμένο μοντέλο δεν λαμβάει υπόψην χρονικές τιμές t-1, κλπ. Δεν εξαρτάται από προηγούμενες τιμές-χρονοσειρές, αλλά δημουργεί τυχαίες τιμές και δουλεύει με έναν σύνθετο κώδικα. Η συγκεκριμένη μέθοδος μπορεί να κάνει training γεννώντας τυχαίες τιμές και δεν ενδείκνιται για δεδομένα που δεν εξαρτώνται από την τύχη .

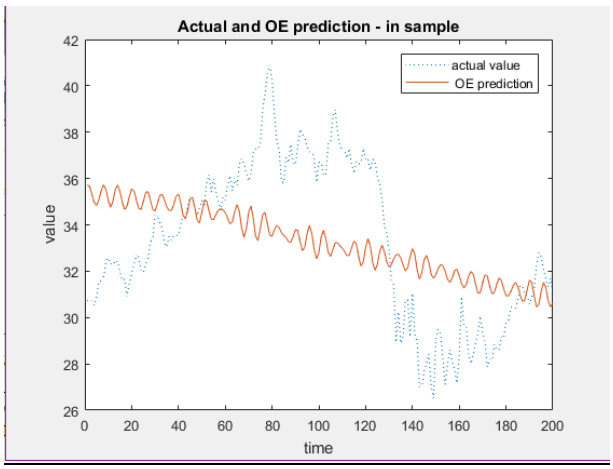
  Κώδικας στη matlab:

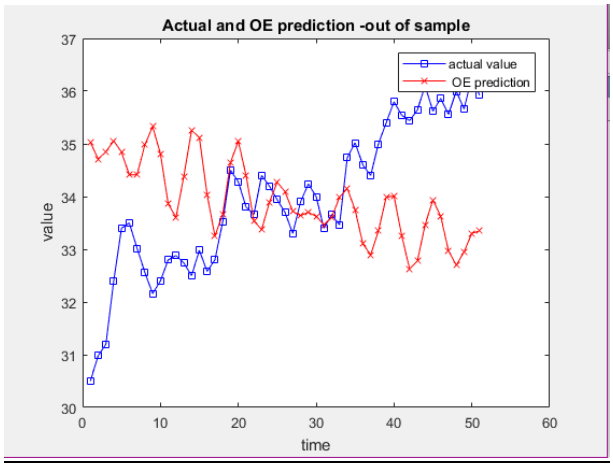
```
%ΜΠΕΡΜΠΑΤΗ ΑΛΕΞΑΝΔΡΑ 2013010143
%ΚΑΡΑΜΠΕΛΑ ΚΩΝΣΤΑΝΤΙΝΑ 2013010014
% PREDICTION USING AN ---OE--- MODEL :PROJECT 4

close all
clear
clc

mydata=xlsread ('MPERMPATI.xlsx', 'e1:e252');

figure(1) % a view of data
plot (mydata)
xlabel('time'); ylabel('values')
title('Actual values')

%  Estimation of an OE model


y=mydata;

y1=mydata(1:200); % TRAINING DATA
y2=mydata(201:end);% TESTING DATA

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%The Output Error Method does not make sense for a time series (no input).
%Here we create data u by a random numbers
u=sin([1:length(y1)]')+0.2*randn(length(y1),1);
m_oe=oe([y1 u],[1,3,0]); % estimation of the model

fpe_oe=fpe(m_oe); % estimation of Akaike's Final prediction Error (FPE)
aic_oe=aic(m_oe); % estimation of Akaike's Information criterion (AIC)

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% in sample evaluation
yhat_oe_insampleIP=predict(m_oe, [y1 u],1); %predicts the in sample data

%AA=yhat_oe_insampleIP(1,:);
%yhat_oe_insample=AA;
yhat_oe_insample=yhat_oe_insampleIP ;

t1=1:(length(yhat_oe_insample));

figure(2)
plot(t1,y1,':', t1,yhat_oe_insample,'-');
legend('actual value',' OE prediction')
xlabel('time')
ylabel('value')
title('Actual and OE prediction - in sample')

%figure (3) % in sample prediction and plotting
%compare(y1, m_oe, 1)% it is another way to predict and plot directly

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%prediction by OE model out of sample

u2=sin([1:length(y2)]')+0.2*randn(length(y2),1);

yhat_oeIP=predict(m_oe, [y2 u2],1); %predicts the unseen data

%BB=yhat_oeIP{1,:};
%yhat_oe=BB;
yhat_oe=yhat_oeIP;

t=1:(length(yhat_oe));
figure(4)

plot(t,y2,'b-s', t,yhat_oe,'r-x');
legend('actual value',' OE prediction')
xlabel('time')
ylabel('value')
title('Actual and OE prediction -out of sample')

%figure (5)
%compare(y2, m_oe, 1)% it is another way to predict and plot directly

[y2 yhat_oe];% prints the  actual and predicted value

get(m_oe) % gives model information

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% error measures calculation

```

```
% Mean Square Error (MSE)
MSE_oe=(1/length(yhat_oe))*norm(y2-yhat_oe)^2;
% Root Mean Square Error (RMSE)
RMSE_oe=sqrt(norm(y2-yhat_oe)^2/length(y2-yhat_oe));
% Mean Absolute Error (MAE)
MAE_oe=(1/length(y2-yhat_oe))*sum(abs(y2-yhat_oe));
% Mean Absolute Percentage Error (MAPE)
MAPE_oe=(100/length(y2-yhat_oe))*sum(abs(y2-yhat_oe)./abs(y2));
%end
```

Διαγράμματα μοντέλου OUTPUT ERROR :







5ο ΚΕΦΑΛΑΙΟ

- **ΜΟΝΤΕΛΟ BOX- JENKINS**

Είναι μία μέθοδος η οποία εφαρμόζει μοντέλα αυτορυθμιζόμενου κινούμενου μέσου (ARMA) ή αυτόματης αντιστροφής κινητού μέσου (ARIMA) για να βρει την καλύτερη προσαρμογή μιας χρονοσειράς-μοντέλο σε προηγούμενες τιμές μιας χρονοσειράς .

Κώδικας στη matlab:

```
%ΜΠΕΡΜΠΑΤΗ ΑΛΕΞΑΝΔΡΑ 2013010143
%ΚΑΡΑΜΠΕΛΑ ΚΩΝΣΤΑΝΤΙΝΑ 2013010014

% PREDICTION USING AN ---BoxJenkins--- MODEL  :PROJECT 5

close all
clear
clc

mydata=xlsread ('MPERMPATI.xlsx', 'e1:e252') % retrieves the E column data of range 1 to 252 fr

figure(1) % a view of data
plot (mydata)
xlabel('time'); ylabel('values');
title('Actual values');
```

```
%  Estimation of an BJ model
y=mydata;

y1=mydata(1:200); % TRAINING DATA
y2=mydata(201:end);% TESTING DATA

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%Here we create data u by a random numbers
u=sin([1:length(y1)]')+0.2*randn(length(y1),1);
m_bj=bj([y1 u],[2 2 2 2 1]); % estimation of the model

fpe_bj=fpe(m_bj); % estimation of Akaike's Final prediction Error (FPE)
aic_bj=aic(m_bj); % estimation of Akaike's Information criterion (AIC)
```

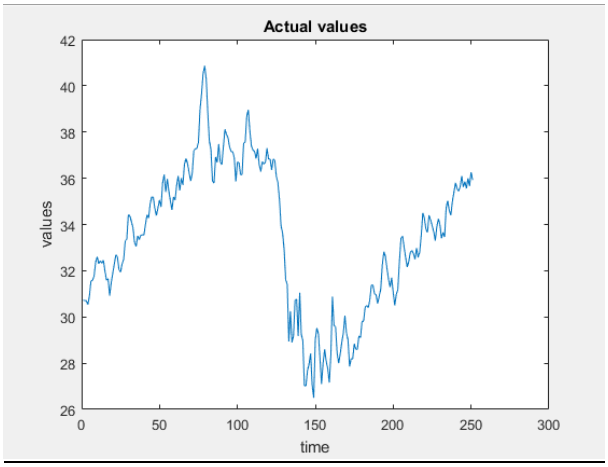
```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% in sample evaluation
yhat_bj_insampleIP=predict(m_bj, [y1 u],1); %predicts the in sample data

%AA=yhat_bj_insampleIP{1,:};
%yhat_bj_insample=AA
yhat_bj_insample=yhat_bj_insampleIP;


t1=1:(length(yhat_bj_insample));
figure(2)
plot(t1,y1,':', t1,yhat_bj_insample,'-');
legend('actual value',' BJ prediction')
xlabel('time')
```

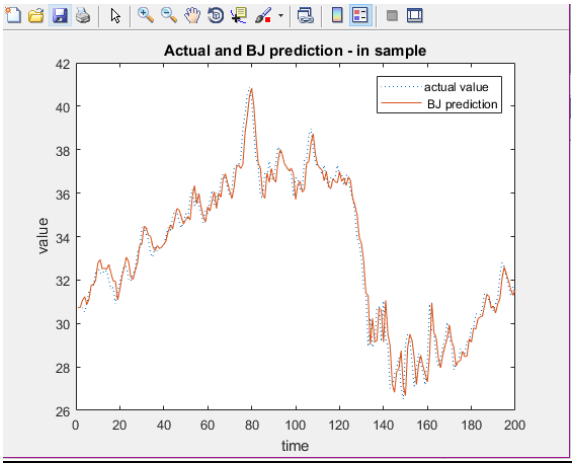
```
ylabel('value')
title('Actual and BJ prediction - in sample')

%figure (3) % in sample prediction and plotting
%compare(y1, m_bj, 1)% it is another way to predict and plot directly

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%prediction by BJ model out of sample

u2=sin([1:length(y2)]')+0.2*randn(length(y2),1);

yhat_bjIP=predict(m_bj, [y2 u2],1); %predicts the unseen data

% BB=yhat_bjIP{1,:};
```

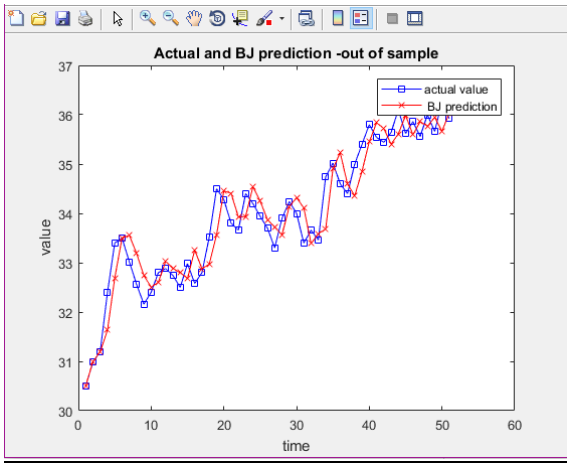
```
% yhat_bj=BB
yhat_bj=yhat_bjIP;


t=1:(length(yhat_bj));
figure(4)
plot(t,y2,'b-s', t,yhat_bj,'r-x');
legend('actual value',' BJ prediction')
xlabel('time')
ylabel('value')
title('Actual and BJ prediction -out of sample')
%
%figure (5)
%compare(y2, m_bj, 1)% it is another way to predict and plot directly

```

```
[y2 yhat_bj];% prints the  actual and predicted value

get(m_bj) % gives model information

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% error measures calculation
% Mean Square Error (MSE)
MSE_bj=(1/length(yhat_bj))*norm(y2-yhat_bj)^2;
% Root Mean Square Error (RMSE)
RMSE_bj=sqrt(norm(y2-yhat_bj)^2/length(y2-yhat_bj));
% Mean Absolute Error (MAE)
MAE_bj=(1/length(y2-yhat_bj))*sum(abs(y2-yhat_bj));
% Mean Absolute Percentage Error (MAPE)
MAPE_bj=(100/length(y2-yhat_bj))*sum(abs(y2-yhat_bj)./abs(y2));
%end
```

Διαγράμματα μοντέλου BJ :







6^{Ο} ΚΕΦΑΛΑΙΟ

ΝΕΥΡΩΝΙΚΑ ΔΙΚΤΥΑ

Μας βοηθούν να παρουμε μια απόφαση, αντγράφοντας τον ανθρώπινο εγκέφαλο. Είναι το πιο εξελιγμένο είδος πρόβλεψης. Οι πληροφορίες καταλήγουν στους νευρώνες από το νευρικό σύστημα, το οποίο είναι ένα κύκλωμα διασυνδεδεμένων νευρόνων. Στόχος των νευρονικών δικτύων είναι η επίλυση υπολογιστικών προβλημάτων. Η βασική τους χρήση είναι σε χρηματιστηριακές προβλέψεις.

- **ΜΟΝΤΕΛΟ ANFIS**

Κώδικας στη matlab:

```
%ΜΠΕΡΜΠΑΤΗ ΑΛΕΞΑΝΔΡΑ Α.Μ. 2013010143
%ΚΑΡΑΜΠΕΛΑ ΚΩΝΣΤΑΝΤΙΝΑ Α.Μ. 2013010014

% PREDICTION USING AN ---ANFIS--- MODEL  :PROJECT 6

close all
clear all
clc

mydata=xlsread ('MPERMPATI.xlsx', 'e1:e252');

%load agrasf
%mydata

tic
figure(1) % a view of data
plot (mydata)
xlabel('time'); ylabel('values')
title('Actual values')
```

```

%  Estimation of an ANFIS model
% prepare training data
%input data
tr=mydata(1:200); % TRAINING DATA
% input (k-2)
train=tr;
train(length(train))=[];% removes the last row
train(length(train))=[]; %removes the second last row
length(train)


%input k-1

train1=tr; % first input
train1(length(train1))=[];% removes the last row
train1(1)=[]; % removes the first row
length(train1)

```

```
% output k
train2=tr; %second input
train2(1)=[];  % removes the first row
train2(1)=[];  %removes the second row
length(train2)



trn_data=[train train1 train2 ]; %(k-2) (k-1) (k) training data



%%%%%% Ploting TRAINING data as a scater plot%%%%%%%
figure(10)
subplot(1,2,1)
plot (train, train2, 'o')
xlabel ('input (k-2) ')
ylabel('ouput (k)')
```

```
title('Training data')
axis equal; axis square

subplot(1,2,2)
plot (train1, train2, 'o')
xlabel ('input (k-1) ')
ylabel('ouput (k)')
title('Training data')
axis equal; axis square




%preparing the evaluation(test) data
ev=mydata(201:end);% TESTING DATA

% input (k-2)
eval=ev;
eval(length(eval))=[]; eval(length(eval))=[];
```

```
length(eval)

% input (k-1)
eval1=ev;
eval1(length(eval))=[];
eval1(1)=[];
length(eval)

% input (k)
eval2=ev;
eval2(1)=[];
eval2(1)=[];
length(eval2)


evaldata=[eval eval1];   %input (k-2) and  (k-1)

y2=eval2; %(k) output data for testing
```

```


% generate FIS matrix
epoch_n=400;
mf_n=[4 4];

%mf_type='gbellmf';   %type of membership function
%mf_type='trimf';   %the parameter b>c
%mf_type='gauss2mf'
%mf_type='gaussmf' %
%mf_type='smf' % unsupported
mf_type='trapmf'; % the parameter b>c
%mf_type='zmf'  %unsupported
%mf_type='pimf' % run problem

in_fismat=genfis1(trn_data, mf_n, mf_type);

% start training
ss=0.1;

[m_anfis, trn_error, step_size ] = ...
    anfis(trn_data, in_fismat, [epoch_n nan ss nan nan], [1,1,1,1]);
```

```
117
118 -   figure('name', ('ANFIS: time series prediction'))
119
120 -   subplot(211);
121 -   tmp=[trn_error ];
122 -   plot(tmp);
123 -   title('Error Curves');
124 -   axis([0 epoch_n min(tmp(:)) max(tmp(:))]);
125 -   xlabel('epochs')
126 -   ylabel('RMSE')
127 -   legend('Training Error');
128
129
130 -   subplot(212);
131 -   plot(step_size);
132 -   xlabel('epochs')
133 -   title('Step Size');
134
135
136
137     % plot the initial membership functons
138 -   figure (20)
```

```
139 -   subplot(2,1,1)
140 -   [mfx, mfy]=plotmf(in_fismat, 'input', 1);
141 -   plot(mfx, mfy);
142 -   title('(a) Initial MFs on input')
143 -   axis([-inf inf -inf inf]);
144 -   subplot(2,1,2)
145 -   [mfx, mfy]=plotmf(in_fismat, 'input', 2);
146 -   plot(mfx, mfy);
147 -   title('(b) Initial MFs on input')
148 -   axis([-inf inf -inf inf]);
149
150
151     % plot final MF's on x,y,z,u
152
153 -   figure (30)
154 -   subplot(2,1,1)
155 -   [mfx, mfy]=plotmf(m_anfis, 'input', 1);
156 -   plot(mfx, mfy);
157 -   title('(a) Final MFs on input')
158 -   axis([-inf inf -inf inf]);
159 -   subplot(2,1,2)
160 -   [mfx, mfy]=plotmf(m_anfis, 'input', 2);
```

```
161 -   plot(mfx, mfy);
162 -   title('(b) Final MFs on input')
163 -   axis([-inf inf -inf inf]);
164
165
166
167
168     %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
169
170
171 -   figure(40)
172 -   plotfis(in_fismat)
173
174 -   figure(50)
175 -   plotfis(m_anfis)
176
177 -   showrule(m_anfis)
178
179
180 -   anfisedit (m_anfis)
181 -   surfview(m_anfis)
182
```

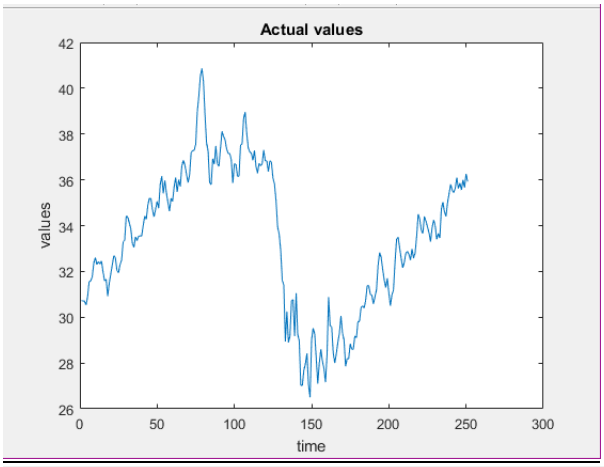
```
183
184     %%%%%%%%%%%%%%%%%%%%%%%%%%%%%% IN SAMPLE EVALAUTION %%%%%%%%%%%%%%
185 -   insample_data=trn_data(:,1:2);
186 -   insaple_output=trn_data(:,3);
187 -   yhat_anfis_insample=evalfis(insample_data, m_anfis);
188
189 -   figure (60)
190 -   plot(insaple_output(end-30:end), 'b-s'), hold, plot(yhat_anfis_insample(end-30:end), 'r-x');
191 -   legend('actual values','ANFIS prediction values')
192 -   xlabel('time')
193 -   ylabel('values')
194 -   title('Actual values and ANFIS prediction in sample')
195
196
197 -   figure (70)
198 -   plot(insaple_output-yhat_anfis_insample)
199 -   xlabel('time')
200 -   ylabel('error')
201 -   title('Prediction in sample errors')
202
203
204
```

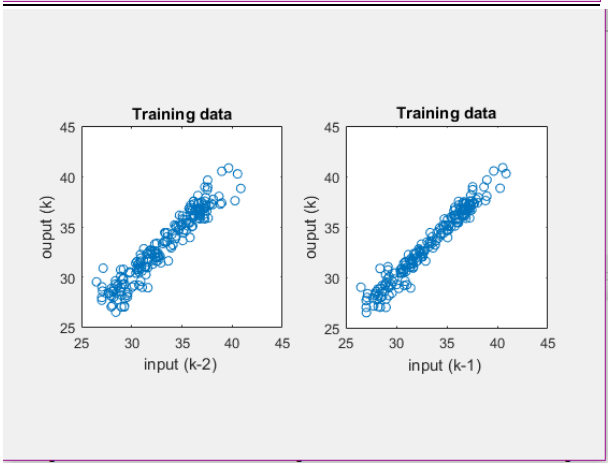
```
%Root Mean Square Error (RMSE)
RMSE_anfis_insample=sqrt(norm(insaple_output-yhat_anfis_insample)^2/length(insaple_output-yhat_
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%prediction by ANFIS model out of sample
%%%%%%%%%%%%%%%%%%%%%%%%%%%% ANFIS EVALUATION %%%%%%%%%%%%%%%%%%
input=evaldata;
%%%%%% Ploting EVALUATION data as a scater plot%%%%%%%
figure(90)
subplot(1,2,1)
plot (eval, eval2, 'o')
xlabel ('input (k-2)')
ylabel('ouput (k)')
title('Evaluating data')
axis equal; axis square
subplot(1,2,2)
plot (eval1, eval2, 'o')
xlabel ('input (k-1)')
```

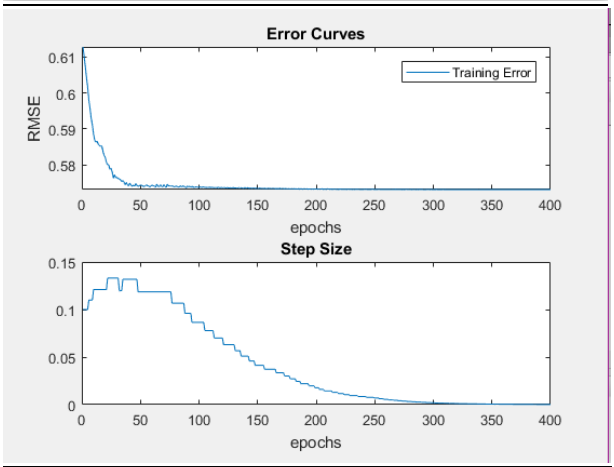
```
ylabel('ouput (k)')
title('Evaluating data')
axis equal; axis square
%%%%%%%%% Evaluation of Anfis out of sample
yhat_anfis=evalfis(input, m_anfis);
adapt_input=y2;
length(adapt_input);
length(yhat_anfis);
result=[adapt_input yhat_anfis (adapt_input-yhat_anfis)];
er_anfis=adapt_input-yhat_anfis; %error
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
figure (100)
plot(adapt_input(end-30:end), 'b-s'), hold, plot(yhat_anfis(end-30:end), 'r-x');
legend('actual values','ANFIS prediction values')
xlabel('time')
ylabel('values')
title('Actual values and ANFIS out of sample prediction')
```

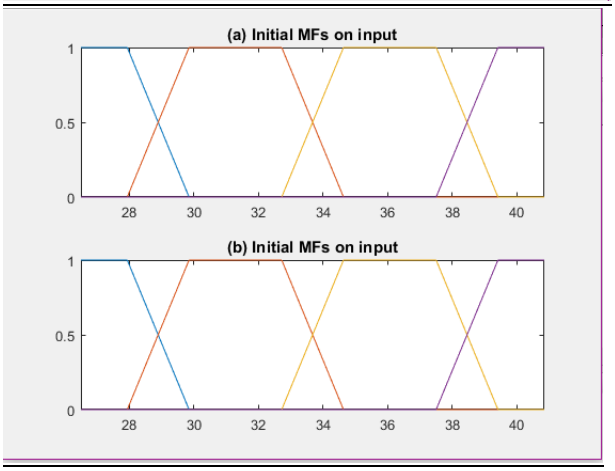
```
figure (120)
plot(adapt_input-yhat_anfis)
xlabel('time')
ylabel('error')
title('Prediction errors')
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% error measures calculation
MSE_anfis=(1/length(yhat_anfis))*norm(er_anfis)^2;
%Root Mean Square Error (RMSE)
RMSE_anfis=sqrt(norm(er_anfis)^2/length(er_anfis));
%Mean Absolute Error (MAE)
MAE_anfis=(1/length(er_anfis))*sum(abs(er_anfis));
%Mean Absolute percentage Error (MAPE)
MAPE_anfis=(100/length(er_anfis))*sum(abs(er_anfis)./abs(adapt_input));
toc  % the calculation time in seconds
runing_minute_time=toc/60;
%end
```

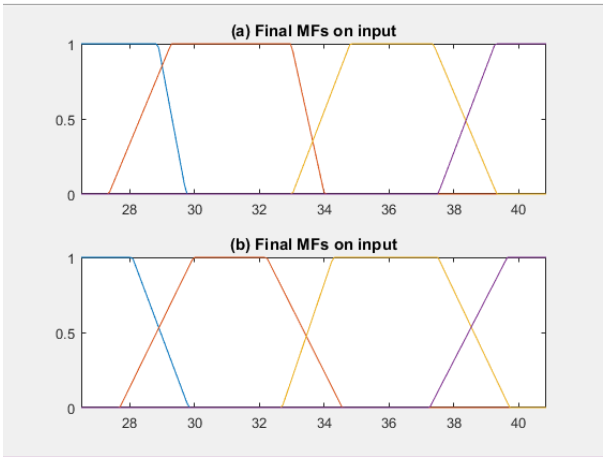
Διαγράμματα μοντέλου ANFIS :

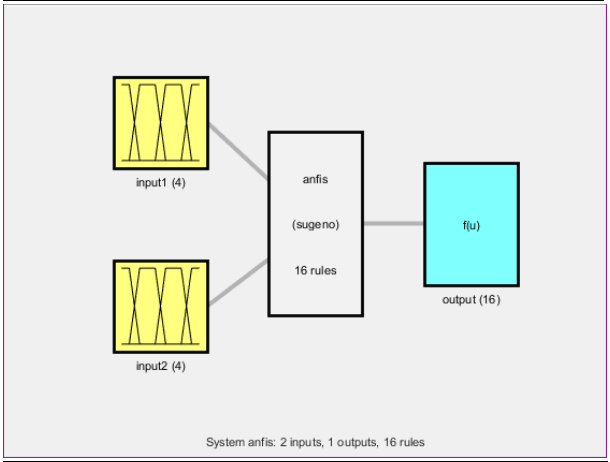

Actual values
values
time

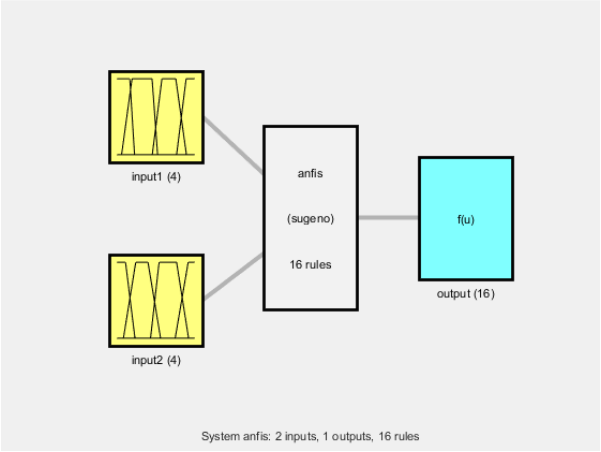

Training data
ouput (k)
input (k-2)
Training data
ouput (k)
input (k-1)

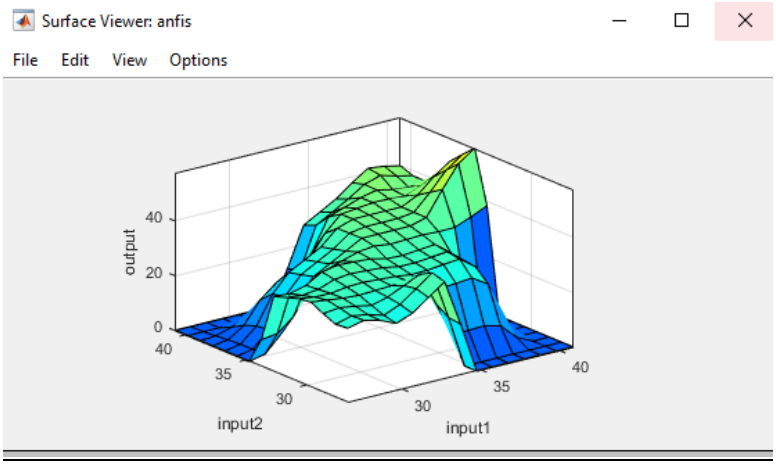

Error Curves
Training Error
RMSE
epochs
Step Size
epochs

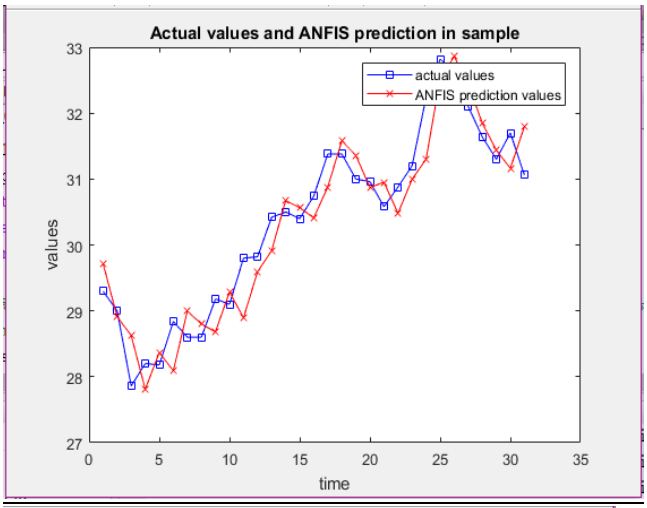

(a) Initial MFs on input
(b) Initial MFs on input

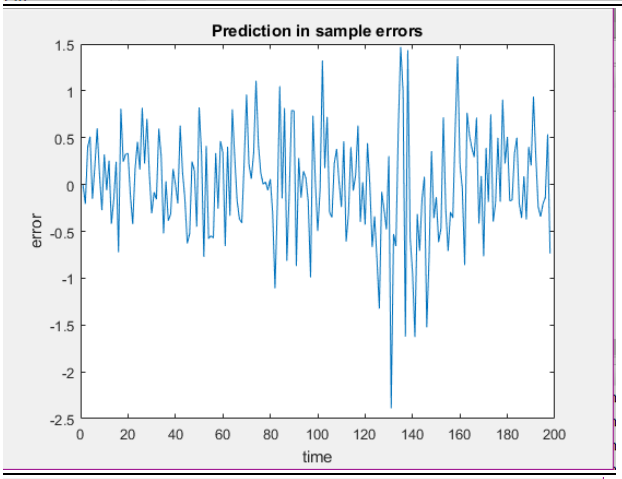

(a) Final MFs on input
1
0.5
0
28
30
32
34
36
38
40
(b) Final MFs on input
1
0.5
0
28
30
32
34
36
38
40

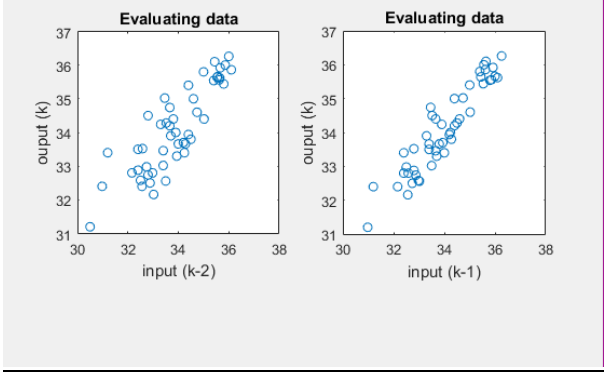

input1 (4)
input2 (4)
anfis
(sugeno)
16 rules
f(u)
output (16)
System anfis: 2 inputs, 1 outputs, 16 rules

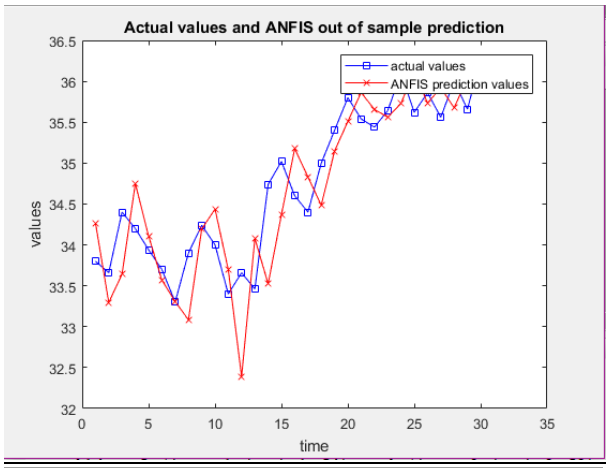

input1 (4)
input2 (4)
anfis
(sugeno)
16 rules
f(u)
output (16)
System anfis: 2 inputs, 1 outputs, 16 rules

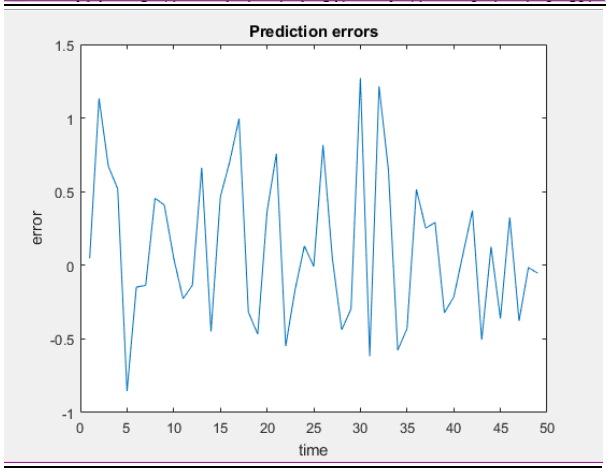

Surface Viewer: anfis
File
Edit
View
Options
output
40
20
0
40
35
30
input2
30
35
40
input1


Actual values and ANFIS prediction in sample
actual values
ANFIS prediction values
values
time


Prediction in sample errors
error
time


Evaluating data
ouput (k)
input (k-2)
Evaluating data
ouput (k)
input (k-1)





- **<u>ΜΟΝΤΕΛΟ NN</u>**

  <u>Κώδικας στη matlab:</u>

```
%ΜΠΕΡΜΠΑΤΗ ΑΛΕΞΑΝΔΡΑ Α.Μ. 2013010143
%ΚΑΡΑΜΠΕΛΑ ΚΩΝΣΤΑΝΤΙΝΑ Α.Μ. 2013010014

%ONE-STEP AHEAD PREDICTION  % By Neural Network

clear all
close all
clc

mydata=xlsread ('MPERMPATI.xlsx', 'e2:e252');

%output (k)
data_for_output= xlsread ('MPERMPATI.xlsx','e3:e252');
%input one step delay (k-1)
dataDealyed_for_input=xlsread ('MPERMPATI.xlsx', 'e2:e251');

data=[dataDealyed_for_input data_for_output];

% prepare training data
% input (k-1)
N2=length(data_for_output);
```

```
N1=floor((N2/5)*4); % 80%

%training data
train_data_input=dataDealyed_for_input(1:N1); %training data-input
train_data_output=data_for_output(1:N1); %training data-input

%testing data
test_data_input=dataDealyed_for_input(N1+1:N2); %testing data
test_data_output=data_for_output(N1+1:N2); %testing data

x=train_data_input;
y=train_data_output;

[x y];

trainX =x;
trainY = y;

% Create test set
testX = test_data_input;
testY = test_data_output;
```

```
43
44 -  t=1:(length(x));
45
46    % figure of data
47 -  figure('name', 'Training data');
48 -  subplot(211); plot(t, x,'-', t, x, 'go');
49 -  xlabel('Time'); ylabel('x'); axis([-inf inf -inf inf]);
50 -  title('Training data x')
51
52 -  subplot(212); plot(t, y, '-', t, y, 'go');
53 -  xlabel('Time'); ylabel('y'); axis([-inf inf -inf inf]);
54 -  title('Training data y')
55
56
57    %%%%%%PLOTING TRAINING DATA AS A SCATTER PLOT%%%%%%%%
58 -  figure('name', ['TRAINING DATA AS A SCATTER PLOT 2D']);
59 -  plot (x, y, 'o')
60 -  xlabel ('x')
61 -  ylabel('y')
62 -  title('Training data')
63 -  axis equal; axis square
```

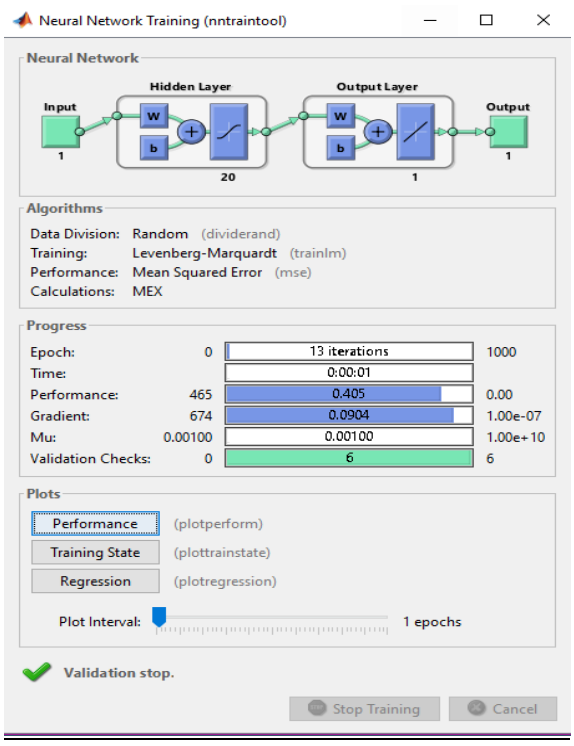
```
64
65
66 -  figure('name', ['TRAINING DATA AS A SCATTER PLOT 3D'])
67 -  plot(x, y, 'o');
68 -  axis([-inf inf -inf inf -inf inf]);
69 -  set(gca, 'box', 'on');
70 -  xlabel('x'); ylabel('y');% zlabel('y(k+1)'); title('Training Data');
71
72 -  net = newff(trainX', trainY', 20); %create a feed-forward backprobagation network
73    %net = newfit(trainX', trainY', 20); %create a fiting network
74
75 -  net.performFcn = 'mae'; % calculates the MAE (mean absolute eror)
76 -  net = train(net, trainX', trainY');
77
78
79    %% Forecast using Neural Network Model
80    % Once the model is built, perform a forecast on the independent test set.
81
82 -  forecastLoad = sim(net, testX')';
83
84    %% Compare Forecast Load and Actual Load
```

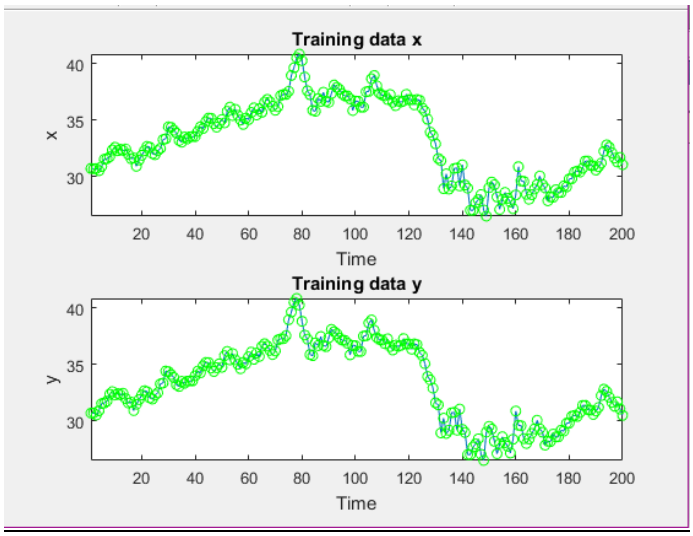
```
85    % Create a plot to compare the actual load and the predicted load as well
86    % as compute the forecast error. In addition to the visualization, quantify
87    % the performance of the forecaster using metrics such as mean absolute
88    % error (MAE), mean absolute percent error (MAPE) and daily peak forecast
89    % error.
90
91 -  err = testY-forecastLoad;
92    %fitPlot(testDates, [testY forecastLoad], err);
93
94 -  errpct = abs(err)./testY*100;
95
96    %fL = reshape(forecastLoad, 24, length(forecastLoad)/24)'; %μετατρέπει τις γραμμες σε στήλες
97    %tY = reshape(testY, 24, length(testY)/24)';
98    %peakerrpct = abs(max(tY,[],2) - max(fL,[],2))./max(tY,[],2) * 100;
99
100 -  MAE = mean(abs(err));
101 -  MAPE = mean(errpct(~isinf(errpct)));
102
103 -  MSE_NN=(1/length(forecastLoad))*norm(err)^2;
104
105   %Root Mean Square Error (RMSE)
106 -  RMSE_NN=sqrt(norm(err)^2/length(err));
```

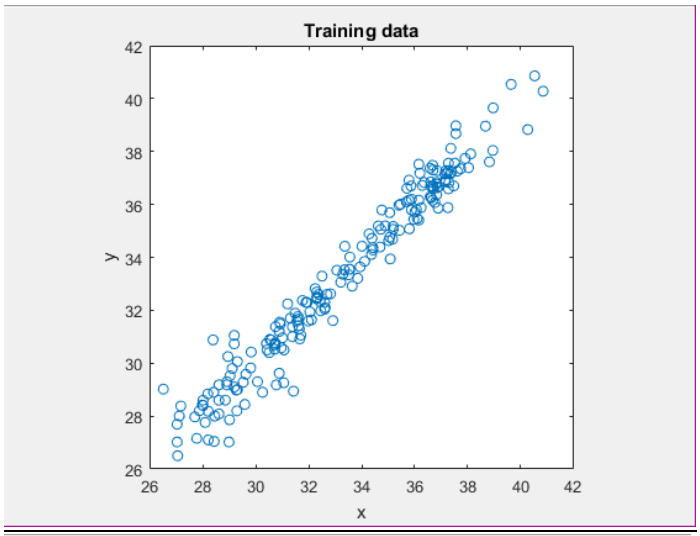
```
107
108     %Mean Absolute Error (MAE)
109 -   MAE_NN=(1/length(err))*sum(abs(err));
110
111     %Mean Absolute percentage Error (MAPE)
112 -   MAPE_NN=(100/length(err))*sum(abs(err)./abs(testY));
113
114
115     %% Examine Distribution of Errors
116     % In addition to reporting scalar error metrics such as MAE and MAPE, the
117     % plot of the distribution of the error and absolute error can help build
118     % intuition around the performance of the forecaster
119
120 -   figure (3);
121 -   subplot(3,1,1); hist(err,100); title('Error distribution');
122 -   subplot(3,1,2); hist(abs(err),100); title('Absolute error distribution');
123 -   line([MAE MAE], ylim); legend('Errors', 'MAE');
124 -   subplot(3,1,3); hist(errpct,100); title('Absolute percent error distribution');
125 -   line([MAPE MAPE], ylim); legend('Errors', 'MAPE');
126
127
```

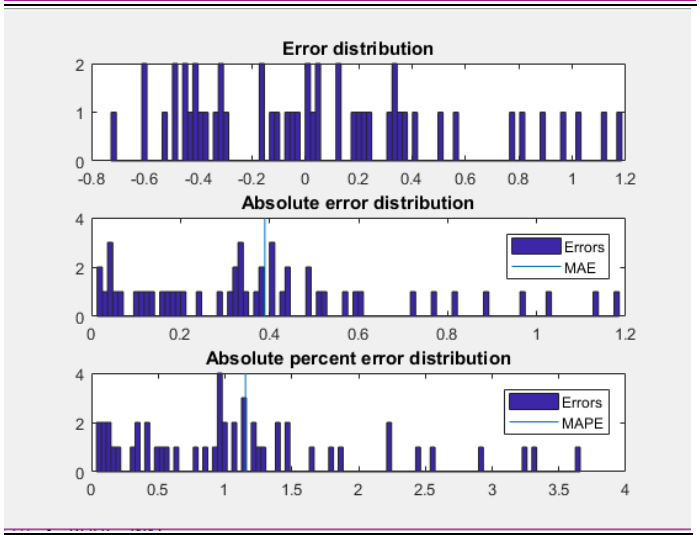
```
128
129
130 -   figure (200)
131 -   plot(forecastLoad(end-40:end), 'b-s'), hold, plot(testY(end-40:end), 'r-x');
132 -   legend('actual values','NN forecasted values')
133 -   xlabel('time')
134 -   ylabel('values')
135 -   title('Actual values and NN forecasts')
136
137     % four erros
138 -   MAPE_NN;
139 -   MAE_NN;
140 -   MSE_NN;
141 -   RMSE_NN;
142
143 -   fprintf('RMSE = %d\n', RMSE_NN)
144 -   fprintf('MAE = %d\n', MAE_NN)
145 -   fprintf('MAPE = %d\n', MAPE_NN)
146 -   fprintf('MSE = %d\n', MSE_NN)
147
148     %end
```

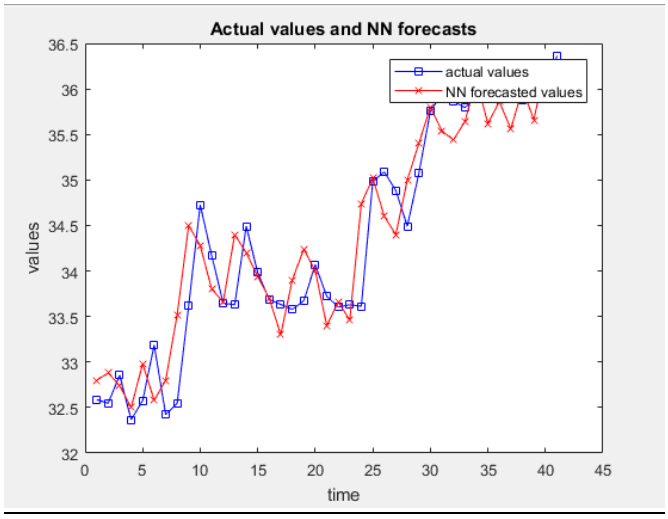
Διαγράμματα μοντέλου NN :




Training data x
x
Time
Training data y
y
Time


Training data
y
x


Error distribution
Absolute error distribution
Errors
MAE
Absolute percent error distribution
Errors
MAPE


Actual values and NN forecasts
actual values
NN forecasted values
values
time

Τα σφάλματα των παραπάνω μοντέλων βρίκονται στον συγκεντρωτικό πίνακα που ακολουθεί.

ΣΦΑΛΜΑΤΑ	AR	ARX	ARMA	OE	BJ	ANFIS	NN
FRE	0,5026	0,5076	0,5127	10,3976	0,5223		
AIC	-0,6880	-0,6780	-0,6680	2,3415	-0,6496		
MSE	0,2331	0,2331	18,4723	3,8120	0,2192	0,2790	0,2387
RMSE	0,4829	0,4829	4,2979	1,9524	0,4682	0,5282	0,4886
MAE	0,4021	0,4021	0,9991	1,5879	0,3808	0,4285	0,3897
MAPE	1,1834	1,1834	3,1407	4,7000	1,1176	1,2614	1,1532

Όπως μπορούμε να δούμε από τον παραπάνω πίνακα , το μοντέλο με τα μικρότερα σφάλματα, επομένως και το πιο αποτελεσματικό, είναι το BOX-JENKINS. Τα σφάλματα με τη μεγαλύερη βαρύτητα είναι το FRE και το AIC. Το συγκεκριμένο μοντέλο έχει το μικρότερο AIC σφάλμα, όμως το FRE σφάλμα, δεν είναι το μικρότερο που παρουσιάστηκε . Παρ'όλα αυτά , όλα τα υπόλοιπα σφάλματα του συγκεκριμένου μοντέλου είναι τα μικρότερα που παρουσιάστηκαν. Επομένως , το μοντέλο που επιλέγεται είναι το BOX-JENKINS.