CHAPTER 6

CONCLUSION

This paper aims to find the answer of the following question: how to forecast the stock price index in in Indonesian stock exchange? Whether the forecasted IDX through the learning procedure techniques of ANN model or not. Is it RWH and EMH theory true?. And comparing the results with some recent research using ANN model in this market.

Due to the issue of accurately forecasting the direction of movements of the stock market price levels is highly significant for formulating the best market trading solutions. It is fundamentally affecting financial trader’s decisions to buy or sell.

The research finding can be concluded as follows:

a. ANN technique can be applicable to forest stock price index in Indonesian stock market, The results of learning rate can be reached 99%, while the best result of forecasting rate is 66%, and the worst rate is 51%. This forecasting technique is considered as a new method which need to be improved in term of designing the model and find out its input variable that the market related to. So that it will enhance the better forecasting results.

b. The power of prediction may not really high in this research methodology as the author’s expectation. However, the prediction result
showed above is fairly good. So, it means that stock market price can be predictable.

c. In terms of comparing ANN performance with recently research. Putra and Kosala (2011)—using technical variables as input data, the highest accuracy rate is 80.48% and the worst one is 49.90%, but their forecasting was focused on individual company, not index prices. For Veri and Baba (2013) forecasting index price of the next trading day they’ve used daily prices as input variables, the empirical results showed that 95% of training accurate and for prediction value percentage varied from 95% to 5% of accuracy. Due to both research provided higher forecasting rate so this research model is not outperform their research methodology.

d. However, author believes this methodology can be applied along with other techniques to help a trading decision.

e. This research methodology has been proved very successful in other stock market research like LMY, TEPX, etc. there are some factors may affect the results of this research as to be mentioned in the limitation of this study.

f. A few of input indicators in the research may not enhance the accuracy rate (unnecessary), because, this stock market can be affected by many macro-economic factors such as political events, investors’ expectations, institutional investors’ choices, firms’ policies, general economic
conditions, interest rates, foreign exchange rates, movement of other stock market, psychology of investors etc.

The limitations of current study: Forecasting stock price index using artificial neural network is a new methodology applied in emerging market (Suchira Chaigusin, 2011) comparing to other methodology i.e. fundamental analysis, technical analysis etc. accessibility to this methodology have some limited.

The most important part of research work is to concentration on coding or programing to create the right forecasting model and find out the best training parameter combination in the goal of reaching the best forecasting results. Meanwhile the researcher himself has very less experience in the field of computer science and information technologies, as this knowledge/skill is required by the research itself. So, the forecasting model in this research may not perfectly done as same as IT expert does. However, author hope that this research will be the first step for other students who wish to continue improving this research methodology or other ANN model.

For further research, author would like to provide some such suggestion as following:

a. Improving this methodology through using matlab tool box, which will enhance more accurate prediction rate.

b. Each method has its own strengths and weaknesses. So, author suggest to use technical indicators of this study and other combining techniques
models by integrating ANN with other classification models such as Support Vector Machines (SVM), Genetics Algorithm (GA) etc. The weakness of one method can be balanced by the strengths of another by achieving a systematic effect.

c. To the best knowledge of the author, the prediction performance of this model can be improved by many ways i.e. adjusting the model parameters by conducting a more sensitive and comprehensive parameter setting. Otherwise, reduction of current variables and adding more different input variables i.e. macro-economic variables such as foreign exchange rates, interest rates and international stock indexes that related to IDX, etc.

**Benefit of this research:** author believes that this research method would benefits to other students in many ways for their further study about using artificial neural network as a tool to forecast stock prices:

a. For student who don’t know about ANN, this research can provide some information and idea on ANN, how its work and why it’s used in financial field.

b. It will be the basic idea for students who wants to try a new methodology of forecasting stock prices, especially, who are majoring finance or related fields.

c. Student can learn some part of its function used in this research and its code in the matlab program. So that they can create their own
methodology and may have more powerful prediction model than current research.

d. Author believes that this research methodology cannot be done without any mistake. However, this research provides student some basic understanding on how to do it and also benefits to other researcher more or less in someway.

e. As we are a student at present who want to be a successful investor in the future, we cannot reliance on the old ways of forecasting stock prices or using single analyzing technique for making investment decision. This technique (ANN) is highly supported for further study and author believes this research can be a reference or being the first step for other researcher who never used this forecasting technique before.
REFERENCES


Suchira Chaigusin (2011) An Investigation Into the Use of Neural Network for the Prediction of the Stock Exchange of Thailand. Edith Cowan University


Tuan Zea Tan, Chai Quek and Geok See Ng (2007). Biological brain-inspired genetic complementary learning for stock market and bank failure prediction. Computational Intelligence, 23(2), 236–261.


Apendix A: Matlab code

The code below are some of the main part of this research methodology

A. Preprocess code

```matlab
function varargout = preproses(varargin)
% PREPROSES M file for preproses.fig
% PREPROSES, by itself, creates a new PREPROSES or
% raises the existing
% singleton.*
% H = PREPROSES returns the handle to a new PREPROSES
% or the handle to
% the existing singleton*.
% PREPROSES('CALLBACK',hObject,eventData,handles,...)
calls the local
% function named CALLBACK in PREPROSES.M with the
given input arguments.
% PREPROSES('Property','Value',...) creates a new
% existing singleton*. Starting from the left, property
% value pairs are
% applied to the GUI before preproses_OpeningFcn gets
called. An
% unrecognized property name or invalid value makes
% property application
% stop. All inputs are passed to preproses_OpeningFcn
% via varargin.
% *See GUI Options on GUIDE's Tools menu. Choose "GUI
% allows only one
% instance to run (singleton)".
% See also: GUIDE, GUIDATA, GUIDATA
% Edit the above text to modify the response to help
% preproses

% Last Modified by GUIDE v2.5 23-Jul-2014 08:15:08

% Begin initialization code - DO NOT EDIT
gui_Singleton = 1;
gui_State = struct('gui_Name', mfilename, ...
    'gui_Singleton', gui_Singleton, ...
    'gui_OpeningFcn', @preproses_OpeningFcn, ...
    'gui_OutputFcn', @preproses_OutputFcn, ...
    'gui_LayoutFcn', [], ...
    'gui_Callback', []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end
```
if nargin
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end

% End initialization code  DO NOT EDIT

% - Executes just before preproses is made visible.
function preproses_OpeningFcn(hObject, eventdata, handles, varargin)
% This function has no output args, see OutputFcn.
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
% varargin    command line arguments to preproses (see VARARGIN)

% Choose default command line output for preproses
handles.output = hObject;

% Update handles structure
guidata(hObject, handles);

% UIWAIT makes preproses wait for user response (see UIRESUME)
% uisave(handles.figure1);

% --- Outputs from this function are returned to the
% command line.
function varargout = preproses_OutputFcn(hObject, eventdata, handles)
% varargout    cell array for returning output args (see VARARGOUT);
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Get default command line output from handles structure
varargout{1} = handles.output;

% --- Executes on button press in pushbutton1.
function pushbutton1_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton1 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% --- Executes just before preproses is made visible.
function preproses_OpeningFcn(hObject, eventdata, handles, varargin)
% This function has no output args, see OutputFcn.
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
global stock1 stock
format short g, format compact
stock=xlsread('table.xls','sheet1','mydata1');
t=handles.uitable1;
set(t,'Data',stock);
set(t,'Columnname',{'AD' 'CCI' 'LW%R' 'MACD' 'Mom' 'Roc'
'RSI' 'SMA' 'Stoch.K%' 'MA%k-D%' 'MA%D-%D' 'WMA'});
t=handles.uitable2;
stock1=[];
for i=1:size(stock,2)-1
    stock1=[stock1 normalisasi(stock(:,i),-1,1)];
end
stock1=[stock1 stock(:,end)];
set(t,'Data',stock1);
set(t,'Columnname',{'AD' 'CCI' 'LW%R' 'MACD' 'Mom' 'Roc'
'RSI' 'SMA' 'Stoch.K%' 'MA%k-D%' 'MA%D-%D' 'WMA'});
save mydata stock stock1
% --- Executes on button press in pushbutton2.
function pushbutton2_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton2 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
close(preproses)

% --- Executes on button press in pushbutton3.
function pushbutton3_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton3 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

B. Training code

function varargout = training(varargin)
% TRAINING M-file for training.fig
%   TRAINING, by itself, creates a new TRAINING or raises the existing singleton*.
%       H = TRAINING returns the handle to a new TRAINING or the handle to the existing singleton*.
%   TRAINING('CALLBACK',hObject,eventData,handles,...) calls the local function named CALLBACK in TRAINING.M with the given input arguments.
% TRAINING('Property','Value',...) creates a new
% TRAINING or
% raises the existing singleton*. Starting from the
% left, 
% property value pairs are applied to the GUI before
% training_OpeningFcn gets called. An
% unrecognized property name or invalid value makes
% property application
% stop. All inputs are passed to training_OpeningFcn
% via varargin.
% *
% *See GUI Options on GUIDE's Tools menu. Choose "GUI
% allows only one
% instance to run (singleton)".
% *
% See also: GUIDE, GUIDATA, GUIHANDLES
%
% Edit the above text to modify the response to help
% training
%
% Last Modified by GUIDE v2.5 12-Aug-2014 06:57:57
%
% Begin initialization code - DO NOT EDIT
% gui_Singleton = 1;
gui_State = struct('gui_Name', mfilename, ...
    'gui_Singleton', gui_Singleton, ...
    'gui_OpeningFcn', @training_OpeningFcn,
    ...
    'gui_OutputFcn', @training_OutputFcn,
    ...
    'gui_LayoutFcn', [], ...
    'gui_Callback', []);
if nargin && ischar(varargin{1})
gui_State.gui_Callback = str2func(varargin{1});
end
if nargin
 [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
 gui_mainfcn(gui_State, varargin{:});
end
% End initialization code - DO NOT EDIT
%
% --- Executes just before training is made visible.
% function training_OpeningFcn(hObject, eventdata, handles, varargin)
% This function has no output args, see OutputFcn.
% hObject handle to figure
% eventdata reserved - to be defined in a future version of
% MATLAB
% handles structure with handles and user data (see
% GUIDATA)
% varargin command line arguments to training (see
% VARARGIN)
% Choose default command line output for training
handles.output = hObject;
global stock stock1
load mydata
% Update handles structure
guidata(hObject)=t, handles);

% UIWAIT makes training wait for user response (see
Uiresume)
% uiswait(handles.figure1);

% --- Outputs from this function are returned to the command
line.
function varargout = training_OutputFcn(hObject, eventdata, handles)
% varargout  cell array for returning output args (see
varargout); % Create architecture
% hObject  handle to pushbutton1 (see GCBO)
% eventdata reserved - to be defined in a future version of
Matlab
% handles  structure with handles and user data (see
Guidata)

% Get default command line output from handles structure
varargout(1) = handles.output;

% --- Executes on button press in pushbutton1.
function pushbutton1_Callback(hObject, eventdata, handles)
% hObject  handle to pushbutton1 (see GCBO)
% eventdata reserved to be defined in a future version of
Matlab
% handles  structure with handles and user data (see
Guidata)
% load mydata
% %create architecture
% PT=P;
global xet y Pn T stock1
nil=str2num(get(handles.edit13,'string')) 2004;
dataall=stock1;
tabel=('[37-36:245-36,:'
'245-36+1:490-36,;'
'490 36+1:740 36,;'
'740-36+1:983 36,;'
'983-36+1:1226-36,;'
'1226-36+1:1471 36,;'
'1471-36+1:1718-36,;'
'1718-36+1:1863-36,;'
'1863-36+1:2202-36,;'
'2202-36+1:2300-36,;
]);
% cmd=['data=dataall( tabel(nil) ');]
% eval(cmd);
% P=data(:,1:end-1)';
% T=data(:,end)';
Pn=data(1:end-1,1:end-1)';
function pushbutton2_Callback(hObject, eventdata, handles)
%
% --- Executes on button press in pushbutton2.

T=data(2:end,end)';
save data Pn T data
nh=str2num(get(handles.edit4,'string'));
ett = newff(minmax(Pn),[nh 1], { 'tansig' 'purelin'},'traindx');
%inialize before train
nett.LW(2,1) = nett.LW(2,1)*0.01;
ett.b(2) = nett.b(2)*0.01;
ett.performFcn = 'sse';
EPOCH=str2num(get(handles.edit7,'string'));
nett.trainParam.epochs =EPOCH ;%10000; %parameter epoch
nett.trainParam.goal = str2num(get(handles.edit9,'string'));
lR=str2num(get(handles.edit5,'string'));
nett.trainParam.lr=Lr;
nett.trainParam.show = 1000;
MC=str2num(get(handles.edit6,'string'));
nett.trainParam.mc = MC; %momentum koefisien
nett.trainParam.time=20*60;
nett.trainParam.min_grad=1e-50;
% do train
[nett,Tr,Y,E,Pf,Af] = train(nett,Pn,T);
y=sim(nett,Pn);
yt=[];
for i=1:length(y)
    yt(i)=fth(y(i));
end
hasil=[(1:length(T))' T' 'y' 'yt'];
t=handles.uitable1;
set(t,'Data',hasil);
set(t,'Columnname',{'Day' 'Actual' 'Predicted' 'Up/Down'});
set(t,'Columnwidth',{30 50 50 50});
MAE=sum(abs(T'-yt'))/length(T');
set(handles.edit1,'string',MAE);
RMSE=sqrt(mse(T'-yt')); %versi one
set(handles.edit2,'string',RMSE);
MAPE=sum(abs(T'-yt'))/length(T');
set(handles.edit8,'string',MAPE);
SSE=sse(T,yt);
SST=sse(T mean(T));
R2=1 {(SSE/SST);
set(handles.edit3,'string',R2);
tot=0;
for i=1:length(y)
    if T(i)==yt(i)
        tot=tot+1;
    end
end
PR=tot/length(yt);
set(handles.edit10,'string',PR);
Pstat=ranksum(T,yt);
set(handles.edit15,'string',Pstat);

% --- Executes on button press in pushbutton2.
function pushbutton2_Callback(hObject, eventdata, handles)
% hObject handle to pushbutton2 (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles structure with handles and user data (see GUIDATA)
close(training)

function edit1_Callback(hObject, eventdata, handles)
% hObject handle to edit1 (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of edit1 as text
% str2double(get(hObject,'String')) returns contents of edit1 as a double

% --- Executes during object creation, after setting all properties.
function edit1_CreateFcn(hObject, eventdata, handles)
% hObject handle to edit1 (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
% See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function edit2_Callback(hObject, eventdata, handles)
% hObject handle to edit2 (see GCBO)
% eventdata reserved to be defined in a future version of MATLAB
% handles structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of edit2 as text
% str2double(get(hObject,'String')) returns contents of edit2 as a double

% --- Executes during object creation, after setting all properties.
function edit2_CreateFcn(hObject, eventdata, handles)
% hObject handle to edit2 (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles empty - handles not created until after all CreateFcns called
% Hint: edit controls usually have a white background on Windows.
% See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function edit3_Callback(hObject, eventdata, handles)
% hObject    handle to edit3 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of edit3 as text
%       str2double(get(hObject,'String')) returns contents of edit3 as a double

% --- Executes during object creation, after setting all properties.
function edit3_CreateFcn(hObject, eventdata, handles)
% hObject    handle to edit3 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
% See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function edit4_Callback(hObject, eventdata, handles)
% hObject    handle to edit4 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of edit4 as text
%       str2double(get(hObject,'String')) returns contents of edit4 as a double

% --- Executes during object creation, after setting all properties.
function edit4_CreateFcn(hObject, eventdata, handles)
% hObject    handle to edit4 (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
% See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
    get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function edit5_Callback(hObject, eventdata, handles)
% hObject    handle to edit5 (see GCBO)
% eventdata reserved to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of edit5 as text
%        str2double(get(hObject,'String')) returns contents of edit5 as a double

% --- Executes during object creation, after setting all properties.
function edit5_CreateFcn(hObject, eventdata, handles)
% hObject    handle to edit5 (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
% See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
    get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function edit6_Callback(hObject, eventdata, handles)
% hObject    handle to edit6 (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of edit6 as text
%        str2double(get(hObject,'String')) returns contents of edit6 as a double
% --- Executes during object creation, after setting all properties.
function edit6_CreateFcn(hObject, eventdata, handles)
    % hObject    handle to edit6 (see GCBO)
    % eventdata  reserved - to be defined in a future version of MATLAB
    % handles    empty - handles not created until after all CreateFcsns called

    % Hint: edit controls usually have a white background on Windows.
    % See ISPC and COMPUTER.
    if ispc && isequal(get(hObject,'BackgroundColor'),
        get(0,'defaultUicontrolBackgroundColor'))
        set(hObject,'BackgroundColor','white');
    end

function edit7_Callback(hObject, eventdata, handles)
    % hObject    handle to edit7 (see GCBO)
    % eventdata  reserved - to be defined in a future version of MATLAB
    % handles    structure with handles and user data (see GUIDATA)

    % Hints: get(hObject,'String') returns contents of edit7 as text
    % str2double(get(hObject,'String')) returns contents of edit7 as a double

% --- Executes during object creation, after setting all properties.
function edit7_CreateFcn(hObject, eventdata, handles)
    % hObject    handle to edit7 (see GCBO)
    % eventdata  reserved - to be defined in a future version of MATLAB
    % handles    empty - handles not created until after all CreateFcsns called

    % Hint: edit controls usually have a white background on Windows.
    % See ISPC and COMPUTER.
    if ispc && isequal(get(hObject,'BackgroundColor'),
        get(0,'defaultUicontrolBackgroundColor'))
        set(hObject,'BackgroundColor','white');
    end

function edit8_Callback(hObject, eventdata, handles)
    % hObject    handle to edit8 (see GCBO)
    % eventdata  reserved - to be defined in a future version of MATLAB
    % handles    structure with handles and user data (see GUIDATA)

    % Hints: get(hObject,'String') returns contents of edit8 as text

% str2double(get(hObject,'String')) returns contents of edit8 as a double

% --- Executes during object creation, after setting all properties.
function edit8_CreateFcn(hObject, eventdata, handles)
% hObject handle to edit8 (see GCBO)
% eventdata reserved to be defined in a future version of MATLAB
% handles empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
    get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function edit9_Callback(hObject, eventdata, handles)
% hObject handle to edit9 (see GCBO)
% eventdata reserved to be defined in a future version of MATLAB
% handles structure with handles and user data (see GUIDATA)

% Hint: get(hObject,'String') returns contents of edit9 as text
%       str2double(get(hObject,'String')) returns contents of edit9 as a double

% --- Executes during object creation, after setting all properties.
function edit9_CreateFcn(hObject, eventdata, handles)
% hObject handle to edit9 (see GCBO)
% eventdata reserved to be defined in a future version of MATLAB
% handles empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
    get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function edit10_Callback(hObject, eventdata, handles)
% hObject handle to edit10 (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles structure with handles and user data (see GUIDATA)
% Hints: get(hObject,'String') returns contents of edit10 as text
%        str2double(get(hObject,'String')) returns contents of edit10 as a double

% --- Executes during object creation, after setting all properties.
function edit10_CreateFcn(hObject, eventdata, handles)
    hObject    handle to edit10 (see GCBO)
    eventdata  reserved - to be defined in a future version of MATLAB
    handles    empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
    get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function edit11_CreateFcn(hObject, eventdata, handles)
    hObject    handle to edit11 (see GCBO)
    eventdata  reserved - to be defined in a future version of MATLAB
    handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of edit11 as text
%        str2double(get(hObject,'String')) returns contents of edit11 as a double

% --- Executes during object creation, after setting all properties.
function edit11_CreateFcn(hObject, eventdata, handles)
    hObject    handle to edit11 (see GCBO)
    eventdata  reserved - to be defined in a future version of MATLAB
    handles    empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
    get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

% --- Executes on selection change in popupmenu1.
function popupmenu1_Callback(hObject, eventdata, handles)
    hObject    handle to popupmenu1 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: contents - get(hObject,'String') returns popupmenul contents as cell array
% contents{get(hObject,'Value')} returns selected item from popupmenul

% --- Executes during object creation, after setting all properties.
function popupmenul_CreateFcn(hObject, eventdata, handles)
% hObject    handle to popupmenul (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: popupmenu controls usually have a white background on Windows.
% See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function edit12_Callback(hObject, eventdata, handles)
% hObject    handle to edit12 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of edit12 as text
% str2double(get(hObject,'String')) returns contents of edit12 as a double

% --- Executes during object creation, after setting all properties.
function edit12_CreateFcn(hObject, eventdata, handles)
% hObject    handle to edit12 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
% See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end
function edit13_Callback(hObject, eventdata, handles)
    % hObject    handle to edit13 (see GCBO)
    % eventdata reserved - to be defined in a future version of MATLAB
    % handles    structure with handles and user data (see GUIDATA)

    % Hints: get(hObject,'String') returns contents of edit13 as text
    % str2double(get(hObject,'String')) returns contents of edit13 as a double

    % --- Executes during object creation, after setting all properties.
    function edit13_CreateFcn(hObject, eventdata, handles)
        % hObject    handle to edit13 (see GCBO)
        % eventdata reserved - to be defined in a future version of MATLAB
        % handles    empty - handles not created until after all CreateFcns called

        % Hint: edit controls usually have a white background on Windows.
        % See ISPC and COMPUTER.
        if ispc && isequal(get(hObject,'BackgroundColor'),
            get(0,'defaultUicontrolBackgroundColor'))
            set(hObject,'BackgroundColor','white');
        end

    function edit14_Callback(hObject, eventdata, handles)
        % hObject    handle to edit14 (see GCBO)
        % eventdata reserved - to be defined in a future version of MATLAB
        % handles    structure with handles and user data (see GUIDATA)

        % Hints: get(hObject,'String') returns contents of edit14 as text
        % str2double(get(hObject,'String')) returns contents of edit14 as a double

        % --- Executes during object creation, after setting all properties.
        function edit14_CreateFcn(hObject, eventdata, handles)
            % hObject    handle to edit14 (see GCBO)
            % eventdata reserved - to be defined in a future version of MATLAB
            % handles    empty - handles not created until after all CreateFcns called

            % Hint: edit controls usually have a white background on Windows.
            % See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

% --- Executes on button press in pushbutton3.
function pushbutton3_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton3 (see GCBO)
% eventdata  reserved - to be defined in a future version of
% MATLAB
% handles    structure with handles and user data (see
% GUIDATA)
    global net
    net=newff([p n],
    [nmf=[p n];
    cmd=['save ' nmf ' net'];
    eval(cmd);

% --- Executes on button press in pushbutton4.
function pushbutton4_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton4 (see GCBO)
% eventdata  reserved - to be defined in a future version of
% MATLAB
% handles    structure with handles and user data (see
% GUIDATA)
    global data y T
    figure,
    plot(T,'b');
    hold on;
    plot(y,'r');
    title('Actual/Predicted graph');
    xlabel('Period (day)');
    ylabel('1=up/0=down');
    legend('Actual','Predicted',1);
    axis([1 length(y) 0 1.5]);

% --- Executes on button press in pushbutton5.
function pushbutton5_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton5 (see GCBO)
% eventdata  reserved - to be defined in a future version of
% MATLAB
% handles    structure with handles and user data (see
% GUIDATA)

function edit15_Callback(hObject, eventdata, handles)
% hObject    handle to edit15 (see GCBO)
% eventdata  reserved - to be defined in a future version of
% MATLAB
% handles    structure with handles and user data (see
% GUIDATA)

% Hints: get(hObject,'String') returns contents of edit15 as text
% str2double(get(hObject,'String')) returns contents of edit15 as a double

% --- Executes during object creation, after setting all properties.
function edit15_CreateFcn(hObject, eventdata, handles)
% hObject    handle to edit15 (see GCBO)
% eventdata  reserved to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
% See ispc and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
    get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

C. Testing code

function varargout = testing(varargin)
% TESTING M file for testing.fig
% TESTING, by itself, creates a new TESTING or raises the existing singleton*. % H = TESTING returns the handle to a new TESTING or the handle to the existing singleton*. % TESTING('CALLBACK',hObject,eventData,handles,...) calls the local % function named CALLBACK in TESTING.M with the given input arguments. % TESTING('Property','Value',...) creates a new TESTING or raises the existing singleton*. Starting from the left, property value pairs are applied to the GUI before testing_OpeningFcn gets called. An unrecognized property name or invalid value makes property application stop. All inputs are passed to testing_OpeningFcn via varargin.
% *See GUI Options on GUIDE's Tools menu. Choose "GUI allows only one instance to run (singleton)".
% See also: GUIDE, GUIDATA, GUIDATA
% Edit the above text to modify the response to help testing

% Last Modified by GUIDE v2.5 12-Aug-2014 19:45:41

% Begin initialization code - DO NOT EDIT
gui_Singleton = 1;
gui_State = struct('gui_Name', mfilename, ...
    'gui_Singleton', gui_Singleton, ...
    'gui_OpeningFcn', @testing_OpeningFcn, ...
    'gui_OutputFcn', @testing_OutputFcn, ...
    'gui_LayoutFcn', [], ...
    'gui_Callback', []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end
if nargin
    varargin{1:nargin} = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end
% End initialization code - DO NOT EDIT

% --- Executes just before testing is made visible.
function testing_OpeningFcn(hObject, eventdata, handles, varargin)
% This function has no output args, see OutputFcn.
% hObject    handle to figure
% eventdata reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
% varargin    command line arguments to testing (see VARARGIN)

% Choose default command line output for testing
handles.output = hObject;
global stock stock1
load mydata
% Update handles structure
guidata(hObject, handles);

% UIWAIT makes testing wait for user response (see UIRESUME)
% uiwait(handles.figure1);

% --- Outputs from this function are returned to the command line.
function varargout = testing_OutputFcn(hObject, eventdata, handles)
% varargout  cell array for returning output args (see
VARARGOUT);
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version
of MATLAB
% handles    structure with handles and user data (see
GUIDATA)

% Get default command line output from handles structure
varargout{1} = handles.output;

function edit5_Callback(hObject, eventdata, handles)
% hObject    handle to edit5 (see GCBO)
% eventdata  reserved - to be defined in a future version
of MATLAB
% handles    structure with handles and user data (see
GUIDATA)

% Hints: get(hObject,'String') returns contents of edit5 as
text
%        str2double(get(hObject,'String')) returns contents
of edit5 as a double

% --- Executes during object creation, after setting all
properties.
function edit5_CreateFcn(hObject, eventdata, handles)
% hObject    handle to edit5 (see GCBO)
% eventdata  reserved - to be defined in a future version
of MATLAB
% handles    empty - handles not created until after all
CreateFcns called

% Hint: edit controls usually have a white background on
Windows.
%        See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function edit6_Callback(hObject, eventdata, handles)
% hObject    handle to edit6 (see GCBO)
% eventdata  reserved - to be defined in a future version
of MATLAB
% handles    structure with handles and user data (see
GUIDATA)

% Hints: get(hObject,'String') returns contents of edit6 as
text
%        str2double(get(hObject,'String')) returns contents
of edit6 as a double

% --- Executes during object creation, after setting all
properties.
function edit6_CreateFcn(hObject, eventdata, handles)
% hObject    handle to edit6 (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all
CreateFcns called

% Hint: edit controls usually have a white background on
Windows.
% See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function edit7_CreateFcn(hObject, eventdata, handles)
% hObject    handle to edit7 (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of edit7 as text
%        str2double(get(hObject,'String')) returns contents of edit7 as a double

% --- Executes during object creation, after setting all
properties.
function edit7_Callback(hObject, eventdata, handles)
% hObject    handle to edit7 (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles empty - handles not created until after all
CreateFcns called

% Hint: edit controls usually have a white background on
Windows.
% See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function edit8_Callback(hObject, eventdata, handles)
% hObject    handle to edit8 (see GCBO)
% eventdata reserved to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of edit8 as text
%        str2double(get(hObject,'String')) returns contents of edit8 as a double
% --- Executes during object creation, after setting all properties.
function edit8_CreateFcn(hObject, eventdata, handles)
% hObject    handle to edit8 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
% See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
    get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

% --- Executes on button press in pushbutton1.
function pushbutton1_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton1 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
close(testing);

% --- Executes on button press in pushbutton2.
function pushbutton2_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton2 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
data=xlsread('table.xls','sheet1','testing_data');
% P=rand(12,365*5);
% T=randint(1,365*5,[0 1]);
% y=data(:,1:12)';
% T=data(:,end)';
Pn=[];
for i=1:size(P,1)
    Pn=[Pn; normalisasi(P(i,:), 1,1)];
end
Pn=Pn(:,1:end-1);
T=T(:,1:end-1);
y=sim(net,Pn);
hasil=[(1:length(T))' y' zeros(length(y),1)];
for i=1:size(hasil,1)
    if y(i)<0
        hasil(i,4)=0;
    elseif y(i)>=0 && y(i)<.5
        hasil(i,4)=1;
    else
        hasil(i,4)=2;
    end
end
%xlswrite('table.xls',hasil,'validasi')
t=handles.uitable1;
set(t,'Data',hasil);
set(t,'Columnname',['Day' 'Actual' 'Predicted' '0=V;1=-';2='^']);
set(t,'Columnwidth',{30 50 50 75});
MAE=mean(abs(T'-yt')/length(T'));
set(handles.edit5,'string',MAE);
RMSE=sqrt(mean((T'-yt')^2));
set(handles.edit6,'string',RMSE);
MAPE=mean(abs(T'-yt')/100);
set(handles.edit7,'string',MAPE);
SSE=sse(T-y);
R2=1-(SSE/SST);
set(t,'Columnname',{'Day' 'Actual' 'Predicted' 'Up/Down'});
set(t,'Data',hasil);
t=handles.uitable1;
hasil=[(1:length(T))' T' y' yt'
yt(i)=fth(y(i));
for i=1:length(y)
    yt(i)=fth(y(i));
end
hasil=[(1:length(T))' T' y' yt'];
t=handles.uitable1;
set(t,'Data',hasil);
set(t,'Columnname',['Day' 'Actual' 'Predicted' 'Up/Down']);
set(t,'Columnwidth',{30 50 50 75});
MAE=sum(abs(T'-yt')/length(T'));

set(handles.edit5,'string',MAE);
RMSE=sqrt(mse(T'-yt'));
set(handles.edit6,'string',RMSE);
MAPE=abs(T'-yt')/length(T);
set(handles.edit7,'string',MAPE);
SSE=sse(T-yt);
SST=sse(T-mean(T));
R2=1-(SSE/SST);
set(handles.edit8,'string',R2);
tot=0;
for i=1:length(y)
  if T(i)==yt(i)
    tot=tot+1;
end
PR=tot/length(yt);
set(handles.edit12,'string',PR);
Pstat=ranksum(T,yt);
set(handles.edit13,'string',Pstat);

% --- Executes on button press in pushbutton4.
function pushbutton4_Callback(hObject, eventdata, handles)

% hObject    handle to pushbutton4 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
close(testing);

function edit9_Callback(hObject, eventdata, handles)

% hObject    handle to edit9 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of edit9 as text
% str2double(get(hObject,'String')) returns contents of edit9 as a double

% --- Executes during object creation, after setting all properties.
function edit9_CreateFcn(hObject, eventdata, handles)

% hObject    handle to edit9 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
% See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
ge(0,'defaultUicontrolBackgroundColor'))
set(hObject,'BackgroundColor','white');
end

function edit10_Callback(hObject, eventdata, handles)
% hObject handle to edit10 (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of edit10 as text
%       str2double(get(hObject,'String')) returns contents of edit10 as a double

% --- Executes during object creation, after setting all properties.
function edit10_CreateFcn(hObject, eventdata, handles)
% hObject handle to edit10 (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
    get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function edit11_Callback(hObject, eventdata, handles)
% hObject handle to edit11 (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of edit11 as text
%       str2double(get(hObject,'String')) returns contents of edit11 as a double

% --- Executes during object creation, after setting all properties.
function edit11_CreateFcn(hObject, eventdata, handles)
% hObject handle to edit11 (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
% See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
    get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

% --- Executes on button press in pushbutton5.
function pushbutton5_Callback(hObject, eventdata, handles)
    % hObject    handle to pushbutton5 (see GCBO)
    % eventdata  reserved - to be defined in a future version
    % handles    structure with handles and user data (see GUIDATA)
    global net
    [n p]=uigetfile('*.*');
    nmf=[p n];
    cmd=['load ' nmf ' '];
    eval(cmd);

% --- Executes on button press in pushbutton6.
function pushbutton6_Callback(hObject, eventdata, handles)
    % hObject    handle to pushbutton6 (see GCBO)
    % eventdata  reserved - to be defined in a future version
    % handles    structure with handles and user data (see GUIDATA)
    global data y Pn
    figure,
    plot(Pn(3,:),'b');
    hold on;
    plot(y,'r');
    title('Price index comparison');
    xlabel('Period (day)');
    ylabel('Closing price index');
    legend('Closing price','predicted price',1);

% --- Executes on button press in pushbutton7.
function pushbutton7_Callback(hObject, eventdata, handles)
    % hObject    handle to pushbutton7 (see GCBO)
    % eventdata  reserved - to be defined in a future version
    % handles    structure with handles and user data (see GUIDATA)

function edit12_Callback(hObject, eventdata, handles)
    % hObject    handle to edit12 (see GCBO)
    % eventdata  reserved - to be defined in a future version
    % handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of edit12
% as text
% str2double(get(hObject,'String')) returns contents
% of edit12 as a double
% --- Executes during object creation, after setting all properties.
function edit12_CreateFcn(hObject, eventdata, handles)
    % hObject    handle to edit12 (see GCBO)
    % eventdata  reserved - to be defined in a future version of MATLAB
    % handles    empty - handles not created until after all CreateFcsns called

% Hint: edit controls usually have a white background on Windows.
% See ISPC and COMPUTER.
if ispc & isequal(get(hObject,'BackgroundColor'),
    get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

% --- Executes on button press in pushbutton8.
function pushbutton8_Callback(hObject, eventdata, handles)
    % hObject    handle to pushbutton8 (see GCBO)
    % eventdata  reserved - to be defined in a future version of MATLAB
    % handles    structure with handles and user data (see GUIDATA)

% --- Executes on selection change in popupmenu1.
function popupmenu1_Callback(hObject, eventdata, handles)
    % hObject    handle to popupmenu1 (see GCBO)
    % eventdata  reserved - to be defined in a future version of MATLAB
    % handles    structure with handles and user data (see GUIDATA)

    % Hints: contents = get(hObject,'String') returns popupmenu1 contents as cell array
    % contents{get(hObject,'Value')} returns selected item from popupmenu1
    global ¶n F T stock1 stock1 dataall=stock1;
tabel=('37-36:245-36,';
    '245-36:490-36,';
    '490 36:740 36,';
    '740-36:983-36,';
    '983-36:1226-36,';
    '1226-36:1471-36,';
    '1471-36:1718-36,';
    '1718-36:1863-36,';
    '1863-36:2202-36,';
    '2202-36:2300-36,');
    pil=get(handles.popupmenu1,'value');
    switch pil
        case 1
            nil=str2num(get(handles.edit10,'string'));
            nil=nil-2004;
cmd=['data=dataall(' tabel{nil} ');'];
eval(cmd);
Pn=datau(:,1:end-1)';
T=datau(:,end)';
case 2
xlsread('table.xls','-1);
P=xlsread('table.xls','predict','topredict');
P=P(:,1:end-1)';
Pn=P;
T=datau(:,end)';
end

% --- Executes during object creation, after setting all properties.
function popupmenu1_CreateFcn(hObject, eventdata, handles)
% hObject    handle to popupmenu1 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: popupmenu controls usually have a white background on Windows.
%       See ISPC and COMPUTER.
if ispc & isequal(get(hObject,'BackgroundColor'),
    get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function edit13_Callback(hObject, eventdata, handles)
% hObject    handle to edit13 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of edit13 as text
%        str2double(get(hObject,'String')) returns contents of edit13 as a double

% --- Executes during object creation, after setting all properties.
function edit13_CreateFcn(hObject, eventdata, handles)
% hObject    handle to edit13 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%       See ISPC and COMPUTER.
if ispc & isequal(get(hObject,'BackgroundColor'),
    get(0,'defaultUicontrolBackgroundColor'))

set(hObject,'BackgroundColor','white');
end

function edit14_Callback(hObject, eventdata, handles)
% hObject    handle to edit14 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of edit14 as text
% str2double(get(hObject,'String')) returns contents of edit14 as a double

% --- Executes during object creation, after setting all properties.
function edit14_CreateFcn(hObject, eventdata, handles)
% hObject    handle to edit14 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
% See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

% --- Executes on button press in pushbutton9.
function pushbutton9_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton9 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
global Pn P net T stock1 stock
xlsread('table.xls','-1');
P=xlsread('table.xls','predict','topredict');
Pn=[];
for i=1:12
    Pn(i)=normalisasip(P(i),stock(:,i),-1,1);
end
Pn=Pn';
y=sim(net,Pn);
yt=[];
for i=1:length(y)
    yt(i)=fth(y(i));
end
set(handles.edit14,'string',yt(1));
hasil=Pn;
set(handles.uitable1,'Data',hasil)