MATLAB PROJECT

Financial computing techniques
Track in Econometrics of banking and financial market

Professor:
Costin PROTOPOPESCU

Presented by:
George Michell Vargas Contreras

SUMMARIZE

This work is about financial and econometrics techniques specially in time series using the MATLAB software.

Table of contents

Table of figures

1. Introduction and objectives

This project is part of the financial computing techniques subject. It aims to apply all the different topics learnt in class and some extra resources to develop in MATLAB a project related to econometrics techniques and financial theory.

Additionally, the matlab project and this document is available in the following web page: https://georgevargascontreras.github.io/Matlabproject/.

In order to develop this project, I execute the following code in 7 steps:

I. Download data from yahoo finance and upload in Matlab software.
II. Preliminary analysis of the time series.
III. Ordinary linear square and non-parametric regression
IV. Autocorrelogram of the time series
V. The best ARMA model.
VI. Arch effect
VII. ARMAX
VIII. Garch Model
IX. Results from the Ibex35 and CAC40 index

Normally, the code is efficient because it uses a matrix form to compute the estimations, the time of execution depends on the access of internet to download the data from yahoo finance but if you previously take the data, it will spend around 1 or 2 minutes to execute.

Additionally, the code presents some boucle’s to compute the best ARMA, ARMAX and GARCH model and usually it displays some messages with the best results and some conclusions of the most important tests. Firstly, I will present the code with some comments and afterward I will do the analysis of the results.

2. Download data from yahoo finance and upload in Matlab software

In this code you can see a procedure to download the data from yahoo finance with an auxiliary function to connect from MATLAB. It is necessary to write the securities' tickers as they are in yahoo finance, then write the initial, final date and periodicity.

For this project, I analyze two European index IBEX35 and CAC40 which are the Madrid and Paris Market Index over a period of 2 years between 26 august 2015 and 26 august 2017. Afterward, you can load the excel or csv file in Matlab.

It is necessary to verify the figures in the excel file because sometimes yahoo finance throw missing values depending on the stock exchange trading days.

```
%% Initialize-Clean comand window, workspace and figures
clc
clear all
clf
% Download data
Asset = {
    '^IBEX','';
    '^FCHI',''
    };
Date1 = '26-Aug-2015 08:33:51'; %Initial date
Date2 = '26-Aug-2017 08:33:51'; %End date
interval = '1d'; %Periodicty
downloadStocksData(Asset,Date1,Date2,interval)
%% Display
disp('-----------------------')
disp('Time series Project')
disp('By:George Vargas')
disp('-----------------------')
%% Load excel data
filename='^IBEX.xls';filename2='^FCHI.xls'; %Select your file name remenber is posible xls and xlsx
Data = xlsread(filename); Data2 = xlsread(filename2);%Select sheet to work
format shortg; x=Data(1:end,5);volumex=Data(1:end,6); %Precise format
format shortg; y=Data2(1:end,5); volumey=Data2(1:end,6); %Only close price and volume colum
```

FIGURE 1 DOWNLOAD DATA FROM YAHOO FINANCE AND LOAD EXCEL FILE IN MATLAB.

3. Preliminary analysis of the time series

First, the time series analysis required a basic graph and some statistics to study the behavior of the variables.

Normally, the financial time series are nonstationary because they show a bearish or bullish trend that you can graphically see. However, it is necessary to confirm it, applying the Unit Root test of Nonstationary either the Dickey-Fuller or Philips Perrot test. Afterward, the variables should be transformed in their differences, logarithms or returns.

The following code shows how to compute the main statistics in a matrix form, indexes graphs and their transformation and the Dicker Fuller and Phillips Perrot test of unit root of non-stationary.

```
%% Descriptive statistic
[statsx,statsy]=datastats(x,y);%Cell array
[statsvolumex,statsvolumey]=datastats(volumex,volumey);%Cell array
resultsstats{1}=statsx;resultsstats{2}=statsy;
[h,p,statsres(1,1),critval] = jbtest(x);[h,p,statsres(2,1),critval] = jbtest(y);
[h,p,statsres(3,1),critval] = jbtest(volumex);[h,p,statsres(4,1),critval] = jbtest(volumey);
statsres(1,2) = kurtosis(x);statsres(2,2) = kurtosis(y);
statsres(3,2) = kurtosis(volumex);statsres(4,2) = kurtosis(volumey);
statsres(1,3) = skewness(x);statsres(2,3) = skewness(y);
statsres(3,3) = skewness(volumex);statsres(4,3) = skewness(volumey);
resultsstats{3}=statsvolumex;resultsstats{4}=statsvolumey;
descriptivestats=cell2mat(resultsstats(:));%Transform a cell array in array
```

FIGURE 2 DESCRIPTIVE STATISTIC IN MATRIX FORM.

```
%% Graph of both index and their diferences
figure(1)
plot(x,'r-','linewidth',2); hold on
plot(y,'b-','linewidth',2); hold off
set(gca,'xtick',[1 257 514]);
set(gca,'xticklabel',{'August 2015','August 2016','August 2017'});
ylabel('Index Points');xlabel('Time');
title('Price close for CAC40 and IBEX35 from 26-Aug-2015 to 26-Aug-2017')
legend('CAC40','IBEX35','Location','northwest','Orientation','horizontal')
%Diferences and returns estimations of time series
difx=x(2:end)- x(1:end-1);dify=y(2:end)- y(1:end-1);
log_difx=log(x(2:end))- log(x(1:end-1)); %price2ret(x) another way
log_dify=log(y(2:end))- log(y(1:end-1));
figure(2)
subplot(2,1,1)
plot(difx,'r-','linewidth',2)
set(gca,'xtick',[1 257 514]);
set(gca,'xticklabel',{'August 2015','August 2016','August 2017'});
title('Diff and logdif of Price close for CAC40 from 26-Aug-2015 to 26-Aug-2017')
legend('Difx','Location','southeast','Orientation','horizontal')
subplot(2,1,2)
plot(log_difx,'b-','linewidth',2)
set(gca,'xtick',[1 257 514]);
set(gca,'xticklabel',{'August 2015','August 2016','August 2017'});
legend('Log-difx','Location','southeast','Orientation','horizontal')

figure(3)
subplot(2,1,1)
plot(dify,'r-','linewidth',2)
set(gca,'xtick',[1 257 514]);
set(gca,'xticklabel',{'August 2015','August 2016','August 2017'});
title('Diff and logdif of Price close for IBEX35 from 26-Aug-2015 to 26-Aug-2017')
legend('Dify','Location','southeast','Orientation','horizontal')
subplot(2,1,2)
plot(log_dify,'b-','linewidth',2)
set(gca,'xtick',[1 257 514]);
set(gca,'xticklabel',{'August 2015','August 2016','August 2017'});
legend('Log-dify','Location','southeast','Orientation','horizontal')
```

FIGURE 3 BASIC GRAPH OF THE INDEX TIME SERIES AND THEIR TRANSFORMATIONS.

```
%% Test unit root of Nonstationary Dickey-Fuller and Philips Perrot
[hdf,pValuedfx,statdfx,cValuedfx] = adftest(x);[hpp,pValueppy,statppx,cValueppx] = pptest(y);
if pValuedfx>0.05 && pValueppy>0.05
    disp('------------------------')
    disp('Both Time series have a unit root and are Nonstationary')
    disp('------------------------')
else
    disp('------------------------')
    disp('One or both time series not have unit roots and are stationary')
    disp('------------------------')
end
```

FIGURE 4 DICKEY-FULLER AND PHILIPS PERROT UNIT ROOT TEST OF NON-STATIONARY.

4. Ordinary Linear Square and non-parametric regression

Additionally, it is interesting to study the estimation of ordinary linear square, polynomials of degree 5 and degree 10 and the non-parametric estimation:

```
%% Ordinary Linear Square, polynomials and non-parametric regression
c=1;Tx=size(x,1);Ty=size(y,1);
for i=1:Tx
 trendx(i,1)=i;
 constantx(i,1)=c;
end
for i=1:Ty
 trendy(i,1)=i;
 constanty(i,1)=c;
end
vectorx=[constantx trendx];vectory=[constanty trendy];
%B is a vector with the coeficients and R is the resial sample
[Bx,Bintx,Rx,Rintx,statsx]=regress(x,vectorx);[By,Binty,Ry,Rinty,statsy]=regress(y,vectory);
prevtrendx=x-Rx;
prevtrendy=y-Ry;
d5=5;d10=10;
 %Find the coeficients of degree n
polynomial5=polyfit(trendx,x,d5);polynomial10=polyfit(trendx,x,d10);
polynomial5y=polyfit(trendy,y,d5);polynomial10y=polyfit(trendy,y,d10);
%valuation of polynomial acording value of coeficients and values x
p5=polyval(polynomial5,trendx);p10=polyval(polynomial10,trendx);
p5y=polyval(polynomial5y,trendy);p10y=polyval(polynomial10y,trendy);
```

```
%Non parametric estimation with a kernel regression
c1=0.15;c2=0.7;h_s=1.06*std(x)*Tx^(-0.2);yh_s=1.06*std(x)*Ty^(-0.2); %silverman rule for chosing optimal h
xnp1=zeros(Tx,1); xnp2=zeros(Tx,1); ynp1=zeros(Tx,1);ynp2=zeros(Tx,1); %colum vector (pre-initiation)
kernel=@(u)exp(-u.^2/2)/sqrt(2*pi); %Gausian kernel as anonymous function
%small bandwidth
for k=1:Tx;
arg1_x=kernel((trendx(k)-trendx)/c1/h_s);
den=sum(arg1_x);
nom=sum(x.*arg1_x);
xnp1(k)=nom/den;
end
for k=1:Ty;
arg1_y=kernel((trendy(k)-trendy)/c1/yh_s);
deny=sum(arg1_y);
nomy=sum(y.*arg1_y);
ynp1(k)=nomy/deny;
end
%big bandwidth
for k=1:Tx
arg_x2=kernel((trendx(k)-trendx)/c2/h_s);
den=sum(arg_x2);
nom=sum(x.*arg_x2);
xnp2(k)=nom/den;
end
```

FIGURE 5 ORDINARY LINEAR SQUARE, POLYNOMIALS AND NON-PARAMETRIC REGRESSION.

Subsequently, you can plot in the same graph the different estimations by OLS, polynomials functions and non-parametric regression as you can see in the following code:

```
%% Graph estimation by Ordinary Linear Square, polynomials and non-parametric regression
figure(5)
subplot(2,1,1)
plot(x,'r-','linewidth',2); hold on
plot(prevtrendx,'b-','linewidth',2); hold on
plot(p5,'c-','linewidth',2); hold on
plot(p10,'m-','linewidth',2); hold off
set(gca,'xtick',[1 257 514]);
set(gca,'xticklabel',{'August 2015','August 2016','August 2017'});
ylabel('Index Points');xlabel('Time');
title('Price close for CAC4,ols and polinomyal d=1 and d=5 from 26-Aug-2015 to 26-Aug-2017')
legend('CAC40','OLSCAC40','P5','P10','Location','northwest','Orientation','horizontal')
subplot(2,1,2)
plot(x,'r-','linewidth',2); hold on
plot(prevtrendx,'b-','linewidth',2); hold on
plot(xnp1,'y-','linewidth',2); hold on
plot(xnp2,'g-','linewidth',2); hold off
set(gca,'xtick',[1 257 514]);
set(gca,'xticklabel',{'August 2015','August 2016','August 2017'});
ylabel('Index Points');xlabel('Time');
title('Price close for CAC40,ols and non-parametric regression from 26-Aug-2015 to 26-Aug-2017')
legend('CAC40','OLSCAC40','np1','np2','Location','northwest','Orientation','horizontal')
```

```
figure(6)
subplot(2,1,1)
plot(y,'r-','linewidth',2); hold on
plot(prevtrendy,'b-','linewidth',2); hold on
plot(p5y,'c-','linewidth',2); hold on
plot(p10y,'m-','linewidth',2); hold off
set(gca,'xtick',[1 257 514]);
set(gca,'xticklabel',{'August 2015','August 2016','August 2017'});
ylabel('Index Points');xlabel('Time');
title('Price close for IBEX35, ols and polinomyal d=1 and d=5 from 26-Aug-2015 to 26-Aug-2017')
legend('IBEX35','OLSIBEX35','P5','P10','Location','northwest','Orientation','horizontal')
subplot(2,1,2)
plot(y,'r-','linewidth',2); hold on
plot(prevtrendy,'b-','linewidth',2); hold on
plot(ynp1,'y-','linewidth',2); hold on
plot(ynp2,'g-','linewidth',2); hold off
set(gca,'xtick',[1 257 514]);
set(gca,'xticklabel',{'August 2015','August 2016','August 2017'});
ylabel('Index Points');xlabel('Time');
title('Price close for IBEX35, ols and non-parametric regression from 26-Aug-2015 to 26-Aug-2017')
legend('IBEX35','OLSIBEX35','np1','np2','Location','northwest','Orientation','horizontal')
```

FIGURE 6 GRAPH ORDINARY LINEAR SQUARE, POLYNOMIALS DEGREE 5,10 AND NON-PARAMETRIC REGRESSION.

5. Autocorrelogram of the time series

Another important component to study a time series is compute the correlogram to see autocorrelations function for data values and vary time lags.

Normally the financial time series present a structure in the correlogram which can help to identify if the time series are serially correlated between them.

Therefore, I want to test the following hypotheses:

H0: Autocorrelation of order k is equal to zero: r(i)=0 for all i.

H1: Autocorrelation of order k is different from zero: r(i)≠0 for all i.

I perform the correlogram with 20 lags because we are considering daily returns, therefore 20 lags represent approximately 1 month of trading activity, which is an appropriate period to use in the autocorrelation test.

```
%% Compute autocorrelogram
 H=20;
 gamma_emp=autocovariance_empiric(log_difx,H);
 rho_emp=gamma_emp/var(log_difx);
 gamma_empy=autocovariance_empiric(log_dify,H);
 rho_empy=gamma_empy/var(log_dify);

  figure(4)
    subplot(2,1,1)
    bar([0:H],[1 rho_emp],'r'); hold on
    autocorr(log_difx); hold off
    title('Autocorrelogram for serie x')
    axis tight
    subplot(2,1,2)
    bar([0:H],[1 rho_empy],'b'); hold on
    autocorr(log_dify); hold off
        title('Autocorrelogram for serie Y')
    axis tight
    axis tight
```

FIGURE 7 AUTOCORRELATION FUNCTION AND AUTOCORRELOGRAM PLOT.

Time series present a structure in the correlogram, as consequence it is recommended to use an ARMA model and use a criterium such as AIC (Akaike information criterion), which is useful to decide the best model, as you can see in the bellow code there are two examples to analyze an ARMA model with the arma reg function and arima and estimate function of econometrics toolbox of matlab:

```matlab
%% The best ARMA MODEL for the series returns with armareg function
disp('-----------------------')
disp('The best ARMA MODEL')
disp('-----------------------')
p=[1 1 1 1 1 2 2 2 2 2 3 3 3 3 3 4 4 4 4 4];q=[0 1 2 3 4 0 1 2 3 4 0 1 2 3 4 0 1 2 3 4];
ARMA_models=zeros(size(p',1),7);
for i=1:size(p',1);
    ARMA_models(i,1)=p(i)';ARMA_models(i,2)=q(i)';
    ARMA_models(i,3)=ARMA_models(i,1)+ARMA_models(i,2);
    [result] = arma_reg(log_difx, ARMA_models(i,1), ARMA_models(i,2));
    phi{i}=1:p(i);phi{i}=result.ar;theta{i}=1:p(i);theta{i}=result.ma; %cell array
    serieresiduals{i}=result.residuals;
    [resulty] = arma_reg(log_dify, ARMA_models(i,1), ARMA_models(i,2));
    phiy{i}=1:p(i);phiy{i}=resulty.ar;thetay{i}=1:p(i);thetay{i}=resulty.ma; %cell array
    serieresidualsy{i}=resulty.residuals;
    ARMA_models(i,4)=RSS2(serieresiduals{i});ARMA_models(i,6)=RSS2(serieresidualsy{i});
    ARMA_models(i,5)=aic_criterion(ARMA_models(i,3),Tx-1,ARMA_models(i,4));
    ARMA_models(i,7)=aic_criterion(ARMA_models(i,3),Ty-1,ARMA_models(i,6));
end
[m,ind]=min(ARMA_models(:,5));[my,indy]=min(ARMA_models(:,7)); %Optimal AIC m=value ind=row
msgx = sprintf('ARMA P= %d and q= %d .',ARMA_models(ind,1),ARMA_models(ind,2));
msgy = sprintf('ARMA P= %d and q= %d .',ARMA_models(indy,1),ARMA_models(indy,2));
disp('-----------------------')
disp('The best ARMA MODEL for CAC40')
disp(msgx)
disp('The best ARMA MODEL for IBEX35')
disp(msgy)
```

FIGURE 8 THE BEST ARMA MODEL SERIES RETURN WITH ARMAREG FUNCTION.

```matlab
%% The best ARMA for the series return whit matlab functions
p=[1 1 1 2 2 2 3 3 3 4 4 4];
q=[0 1 2 0 1 2 0 1 2 0 1 2];
Tp=size(p',1);
for i=1:Tp
Mdlx = arima(p(i)',0,q(i)'); %Generate the model arima(p,D,q)
resultx = estimate(Mdlx,log_difx,'display','off');
resx(i,1) =RSS2(infer(resultx,log_difx));
aicx(i,1)=aic_criterion(p(i)'+q(i)',Tx-1,resx(i,1));
Mdly = arima(p(i)',0,q(i)'); %Generate the model arima(p,D,q)
resulty = estimate(Mdly,log_dify,'display','off');
resy(i,1) = RSS2(infer(resulty,log_dify));
aicy(i,1)=aic_criterion(p(i)'+q(i)',Ty-1,resy(i,1));
end
[m,indarimax]=min(aicx(:,1));[my,indarimay]=min(aicy(:,1)); %Optimal AIC m=value ind=row
msgx = sprintf('ARMA P= %d and q= %d .',p(indarimax)',q(indarimax)');
msgy = sprintf('ARMA P= %d and q= %d .',p(indarimay)',q(indarimay)');
disp('-----------------------')
disp('The best ARMA MODEL for IBEX35')
disp(msgx)
disp('The best ARMA MODEL for CAC40')
disp(msgy)
disp('-----------------------')
```

FIGURE 9 THE BEST ARMA MODEL SERIES RETURN WITH MATLAB FUNCTIONS.

After obtaining the best ARMA model for the series returns, you can compute and plot the results and compare them with the initial index time series. Bellow I present one method to do it:

```
%% Compute the best ARMA(4,2) for the series return of IBEX35 and ARMA(4,2) for CAC40 with matlab function
Armamodelibex = arima(4,0,2); %Generate the best model arima(p,D,q)
resultArmamodelibex = estimate(Armamodelibex,log_difx);
resoArmamodelibex=infer(resultArmamodelibex,log_difx);
phiibex35=cell2mat(resultArmamodelibex.AR);
thethaibex35=cell2mat(resultArmamodelibex.MA);
armaibex2=log_difx-resoArmamodelibex; Tarmaibex2=size(armaibex2,1);
closepricex(1,1)=x(1,1);
closepricex(2:Tx,1)=x(1:Tx-1,1).*exp(armaibex2(1:Tarmaibex2,1));

Armamodelcac = arima(4,0,2); %Generate the best model arima(p,D,q)
resultArmamodelcac = estimate(Armamodelcac,log_dify);
resoArmamodelcac=infer(resultArmamodelcac,log_dify);
phicac=cell2mat(resultArmamodelcac.AR);
thethacac=cell2mat(resultArmamodelcac.MA);
armacac2=log_dify-resoArmamodelcac; Tarmacac2=size(armacac2,1);
closepricey(1,1)=y(1,1);
closepricey(2:Ty,1)=y(1:Ty-1,1).*exp(armacac2(1:Tarmacac2,1));
```

FIGURE 10 COMPUTE THE BEST ARMA FOR THE SERIES RETURN

```
%% Plot the best arma model IBEX35 and CAC40
figure(7)
subplot(2,1,1)
plot(log_difx,'c-','linewidth',2); hold on
plot(armaibex2,'m-','linewidth',2); hold off
set(gca,'xtick',[1 257 514]);
set(gca,'xticklabel',{'August 2015','August 2016','August 2017'});
ylabel('Index Points');xlabel('Time');
title('Return IBEX35 and arma model(4,2)ibex35 from 26-Aug-2015 to 26-Aug-2017')
legend('Return IBEX35','ARMAIBEX35(4,2)','Location','southwest','Orientation','horizontal')
subplot(2,1,2)
plot(x,'r-','linewidth',2); hold on
plot(closepricex,'c-','linewidth',2); hold off
set(gca,'xtick',[1 257 514]);
set(gca,'xticklabel',{'August 2015','August 2016','August 2017'});
ylabel('Index Points');xlabel('Time');
title('Close price for IBEX35 and close price estimate with returns of ARMAIBEX35(4,2)from 26-Aug-2015 to 26-Aug-2017')
legend('IBEX35 price','Close price ARMAIBEX35','Location','northwest','Orientation','horizontal')
```

```
figure(8)
subplot(2,1,1)
plot(log_dify,'c-','linewidth',2); hold on
plot(armacac2,'m-','linewidth',2); hold off
set(gca,'xtick',[1 257 514]);
set(gca,'xticklabel',{'August 2015','August 2016','August 2017'});
ylabel('Index Points');xlabel('Time');
title('Return CAC40 and arma model(4,2)CAC40 from 26-Aug-2015 to 26-Aug-2017')
legend('Return CAC40','ARMACAC40(4,2)','Location','southwest','Orientation','horizontal')
subplot(2,1,2)
plot(y,'r-','linewidth',2); hold on
plot(closepricey,'m-','linewidth',2); hold off
set(gca,'xtick',[1 257 514]);
set(gca,'xticklabel',{'August 2015','August 2016','August 2017'});
ylabel('Index Points');xlabel('Time');
title('Close price for CAC40 and close price estimate with returns of CAC40(4,2)from 26-Aug-2015 to 26-Aug-2017')
legend('CAC40 price','Close price ARMACAC40','Location','northwest','Orientation','horizontal')
```

FIGURE 11 PLOT OF THE SERIES RETURN AND CLOSE PRICE WITH ARMA AND THE ORIGINAL VALUES.

6. Arch effect

It is important to compute and analyze the residual sample, moreover estimate the arch effect test to verify if the variance is constant or no:

```
%% Archeffect to verify heterockedasticty/Homoscedasticity
[h,pValuearchx,stat,cValue] = archtest(resoArmamodelibex,'Alpha',0.05);
[h,pValuearchy,stat,cValue] = archtest(resoArmamodelcac,'Alpha',0.05);
msghomo = sprintf('Pvaluex= %d and Pvaluey= %d >0.05.',pValuearchx,pValuearchy);
msghete = sprintf('Pvaluex= %d and Pvaluey= %d <0.05.',pValuearchx,pValuearchy);
if pValuearchx>0.05 && pValuearchy>0.05
    disp('-----------------------')
    disp('Both Time series do not have ARCH Effects-Homoskedasticity')
    disp(msghomo)
    disp('-----------------------')
else
    disp('-----------------------')
    disp('One or both time series have ARCH Effects-Heteroskedasticity')
    disp(msghete)
    disp('-----------------------')
end
```

FIGURE 12 ARCH EFFECT TEST

7. ARMAX

With the previews arch test, I identify that the index's time series do no present problems of Heteroskedasticity meaning that the variance is constant.

Now, I introduce an exogeneous variable in the model to compute an ARMAX model, in this case the new time series are the index's volume uploaded in the figure 1. The following code shows one method to compute the best ARMAX with ordinary least square and plot the results to compare them with the initial time series:

```
%% The best ARIMAX for IBEX35 whit OLS estimation
for i=4:Tx
 volx(i,1)=volumex(i-1,1);
 xt1(i,1)=x(i-1,1);
 xt2(i,1)=x(i-2,1);
 xt3(i,1)=x(i-3,1);
 epsilonxt1(i,1)=utx(i-1,1);
 epsilonxt2(i,1)=utx(i-2,1);
end
vectorx=[ones(size(volx)) volx xt1 xt2 xt3 epsilonxt1 epsilonxt2];
[B,Bint,Rarimaxx,Rintarimaxx,stats]=regress(x(),vectorx);
Estimationx=x-Rarimaxx;
figure(8)
plot(x(4:Tx),'r-','linewidth',2); hold on
plot(Estimationx(4:Tx),'b-','linewidth',2); hold off
set(gca,'xtick',[1 257 514]);
set(gca,'xticklabel',{'August 2015','August 2016','August 2017'});
ylabel('Index points');xlabel('Time');
title('IBEX35 and ARIMAX(x=volume)from 26-Aug-2015 to 26-Aug-2017')
legend('IBEX35','ARIMAXIBEX35','Location','northwest','Orientation','horizontal')
```

```
%% The best ARIMAX for CAC40 whit OLS estimation
for i=5:Ty
 voly(i,1)=volumey(i-1,1);
 yt1(i,1)=y(i-1,1);
 yt2(i,1)=y(i-2,1);
 yt3(i,1)=y(i-3,1);
 yt4(i,1)=y(i-4,1);
 epsilonyt1(i,1)=uty(i-1,1);
 epsilonyt2(i,1)=uty(i-2,1);
end
vectory=[ones(size(voly)) voly yt1 yt2 yt3 yt4 epsilonyt1 epsilonyt2];
[B,Bint,Rarimaxy,Rintarimaxy,stats]=regress(y(),vectory);
Estimationy=y-Rarimaxy;
figure(9)
plot(y(5:Ty),'r-','linewidth',2); hold on
plot(Estimationy(5:Ty),'b-','linewidth',2); hold off
set(gca,'xtick',[1 257 514]);
set(gca,'xticklabel',{'August 2015','August 2016','August 2017'});
ylabel('Index points');xlabel('Time');
title('CAC40 and ARIMAX(x=volume)from 26-Aug-2015 to 26-Aug-2017')
legend('CAC40','ARIMAXCAC40','Location','northwest','Orientation','horizontal')
```

FIGURE 13 THE BEST ARMAX FOR THE INDEX TIME SERIES WITH OLS AND VOLUME AS EXOGENOUS VARIABLE

8. Garch model

In addition, it is important to study the exogeneous time series and verify if they are stationary or not as well as the arch effect. The follow figure displays the analysis of the exogeneous time series:

```
%% Analysis of the exogeneous time series
figure(10)
plot(volumex,'r-','linewidth',2); hold on
plot(volumey,'b-','linewidth',2); hold off
set(gca,'xtick',[1 257 514]);
set(gca,'xticklabel',{'August 2015','August 2016','August 2017'});
ylabel('Number of transactions');xlabel('Time');
title('Volume for CAC40 and IBEX35 from 26-Aug-2015 to 26-Aug-2017')
legend('volumeCAC40','volumeIBEX35','Location','northwest','Orientation','horizontal')
% Advande dicker fuller test
[h,pValuedvolumex,stat,cValue]= adftest(volumex);[h,pValuedvolumey,stat,cValue]= adftest(volumey);
msgNonstationaryvol = sprintf('Pvaluevolumex= %d and Pvaluevolumey= %d >0.05.',pValuedvolumex,pValuedvolumey);
msgstationaryvol = sprintf('Pvaluevolumex= %d and/or Pvaluevolumey= %d <0.05.',pValuedvolumex,pValuedvolumey);
if pValuedvolumex>0.05 && pValuedvolumey>0.05
    disp('-----------------------')
    disp('Both Time series have a unit root and are Nonstationary')
    disp(msgNonstationaryvol)
    disp('-----------------------')
else
    disp('-----------------------')
    disp('One or both time series not have unit roots and are stationary')
    disp(msgstationaryvol)
    disp('-----------------------')
end
```

```
%% Archeffect to verify heterockedasticty
[h,pValuearchvolumex,stat,cValue] = archtest(volumex,'Alpha',0.05);[h,pValuearchvolumey,stat,cValue] = archtest(volumey,'Alpha',0.05);
msghomovol = sprintf('Pvaluex= %d and Pvaluey= %d >0.05.',pValuearchvolumex,pValuearchvolumey);
msghetevol = sprintf('Pvaluex= %d and Pvaluey= %d <0.05.',pValuearchvolumex,pValuearchvolumey);
if pValuearchvolumex>0.05 && pValuearchvolumey>0.05
    disp('----------------------')
    disp('Both Time series do not have ARCH Effects-Homoskedasticity')
    disp(msghomovol)
    disp('----------------------')
else
    disp('----------------------')
    disp('One or both time series have ARCH Effects-Heteroskedasticity')
    disp(msghetevol)
    disp('----------------------')
end
```

FIGURE 14 ANALYSIS OF THE EXOGENEOUS TIME SERIES PLOT, ADVANCED DICKER FULLER AND ARCH EFFECT TEST

I can conclude that the time series are stationary. However, they present problems of Heteroskedasticity (Arch effect), then it is recommended to use a family Garch model.

For this reason, I apply Garch model for exogenous variables to estimate the conditional variance of the time series model and later compute the best garch model for both indexes and plot them as you can see in the below code:

```
%% Garch model exogenous variables index volume-conditional variance time series model
p=[1 1 1 1 2 2 2 2 3 3 3 3 4 4 4 4];
q=[1 2 3 4 1 2 3 4 1 2 3 4 1 2 3 4];
Tp=size(p',1);Tvolx=size(volumex,1);Tvoly=size(volumey,1);
for i=1:Tp
garchmodelvolx = garch(p(i)',q(i)');%Generate the garch model(p,q)
resultvolx = estimate(garchmodelvolx,volumex,'display','off');%,'display','off');
resvolx(i,1) =RSS2(infer(resultvolx,volumex));
aicvolx(i,1)=aic_criterion(p(i)'+q(i)',Tvolx,resvolx(i,1));
garchmodelvoly = garch(p(i)',q(i)');%Generate the garch model(p,q)
resultvoly = estimate(garchmodelvoly,volumey,'display','off');%,'display','off');
resvoly(i,1) =RSS2(infer(resultvoly,volumey));
aicvoly(i,1)=aic_criterion(p(i)'+q(i)',Tvoly,resvoly(i,1));
end
[m,indgarchvolx]=min(aicvolx(:,1));[my,indgarchvoly]=min(aicvoly(:,1)); %Optimal AIC m=value ind=row
msggarchvolx = sprintf('ARMA P= %d and q= %d .',p(indgarchvolx)',q(indgarchvolx)');
msggarchvoly = sprintf('ARMA P= %d and q= %d .',p(indgarchvoly)',q(indgarchvoly)');
disp('-----------------------')
disp('The best ARMA MODEL for IBEX35')
disp(msggarchvolx)
disp('The best ARMA MODEL for CAC40')
disp(msggarchvoly)
disp('-----------------------')
```

Figure 15 The best GARCH model for the exogeneous variable.

```
%% Compute the best garch model for the volume of ibex35
garchmodelvolx = garch(p(indgarchvolx)',q(indgarchvolx)');%Generate the best garch model(p,q)
resultvolx = estimate(garchmodelvolx,volumex,'display','off');
conditvarvolumex=infer(resultvolx,volumex);
conditvolvolumex=sqrt(conditvarvolumex);
figure(12)
plot(volumex,'c--','linewidth',2); hold on
plot(conditvolvolumex,'b-','linewidth',2); hold off
set(gca,'xtick',[1 257 514]);
set(gca,'xticklabel',{'August 2015','August 2016','August 2017'});
ylabel('Number of transactions');xlabel('Time');
title('Volume for IBEX35 and conditional volatility from 26-Aug-2015 to 26-Aug-2017')
legend('volume IBEX35','conditional volatility volume IBEX35','Location','northwest','Orientation','horizontal')
% Simulation conditional variance using presample data.
Modelgarchxoneone = garch('Constant',resultvolx.Constant,'GARCH',resultvolx.GARCH,'ARCH',resultvolx.ARCH);
rng default; % For reproducibility
[Vn,Yn] = simulate(Modelgarchxoneone,100,'NumPaths',10);
figure(13)
plot(Vn);set(gca,'xtick',[1 257 514]);
set(gca,'xticklabel',{'August 2015','August 2016','August 2017'});
ylabel('Number of transactions');xlabel('Time');
title('Conditional variance for IBEX35 volume with 10 paths from 26-Aug-2015 to 26-Aug-2017')
legend('conditional variance','Location','northwest','Orientation','horizontal')
```

Figure 16 Estimation of the best GARCH model for the volume of the IBEX35

```
%% Compute the best garch model for the volume of cac40
garchmodelvoly = garch(p(indgarchvoly)',q(indgarchvoly)');%Generate the best garch model(p,q)
resultvoly = estimate(garchmodelvoly,volumey,'display','off');
conditvarvolumey=infer(resultvoly,volumey);
conditvolvolumey=sqrt(conditvarvolumey);
figure(14)
plot(volumey,'c--','linewidth',2); hold on
plot(conditvolvolumey,'b-','linewidth',2); hold off
set(gca,'xtick',[1 257 514]);
set(gca,'xticklabel',{'August 2015','August 2016','August 2017'});
ylabel('Number of transactions');xlabel('Time');
title('Volume for CAC40 and conditional volatility from 26-Aug-2015 to 26-Aug-2017')
legend('volume CAC40','conditional volatility volume CAC40','Location','northwest','Orientation','horizontal')
% Simulation conditional variance using presample data.
Modelgarchyoneone = garch('Constant',resultvoly.Constant,'GARCH',resultvoly.GARCH,'ARCH',resultvoly.ARCH);
rng default; % For reproducibility
[Vny,Yny] = simulate(Modelgarchyoneone,100,'NumPaths',10);
figure(15)
plot(Vny);set(gca,'xtick',[1 257 514]);
set(gca,'xticklabel',{'August 2015','August 2016','August 2017'});
ylabel('Number of transactions');xlabel('Time');
title('Conditional variance for CAC40 volume with 10 paths from 26-Aug-2015 to 26-Aug-2017')
legend('conditional variance','Location','northwest','Orientation','horizontal')
```

FIGURE 17 ESTIMATION OF THE BEST GARCH MODEL FOR THE VOLUME OF CAC40

9. Results IBEX35 and CAC40

In this part, I will analyze and present the graphs and tables generated from the preview code.

First, the descriptive statistics are showed in the following tables:

	num	max	min	mean	median	range	std
IBEX 35	514,00	11.135,40	7.645,50	9.451,86	9.348,50	3.489,90	846,02
CAC 40	514,00	5.432,40	3.896,71	4.688,59	4.571,99	1.535,69	345,79
Volume IBEX 35	514,00	768.018.900,00	70.619.000,00	263.713.337,74	260.821.500,00	697.399.900,00	90.757.212,30
Volume CAC 40	514,00	415.775.700,00	-	105.171.014,01	100.830.050,00	415.775.700,00	36.536.697,15

	Jarque-Bera Test	Kurtosis	Skewness
IBEX 35	34,6241	1,8014	0,2121
CAC 40	29,2701	2,0712	0,3550
Volume IBEX 35	323,8317	6,3272	1,0062
Volume CAC 40	2.909,4867	14,0837	1,8030

According to some statistic measures, the number of observations are 514, the mean and the standard deviation are lower for the ibex 35 regarding to the CAC40. It is evidenced in figure 18, in which the CAC40 fluctuates with superior units and the range of its movement is higher than IBEX35.

The Jarque Bera(JB) statistical test of normality shows that all variables don't follow a normal distribution because the statistical test is superior to 6 (JB>=6), then I reject the null hypothesis.

H0: JB<6 Time series follow a normal distribution.

H1: JB>=6 Time series don't follow a normal distribution

Additionally, I see that the volume of the index time series is leptokurtic because the Kurtosis is higher than three, meaning that series are more peaked than normal distribution. On the other hand, the index price is platykurtic because the Kurtosis is lower than three, therefore the series are less peaked than normal distribution.

Moreover, the skewness shows that all-time series distributions are skewed to the right because the values are higher than 0.

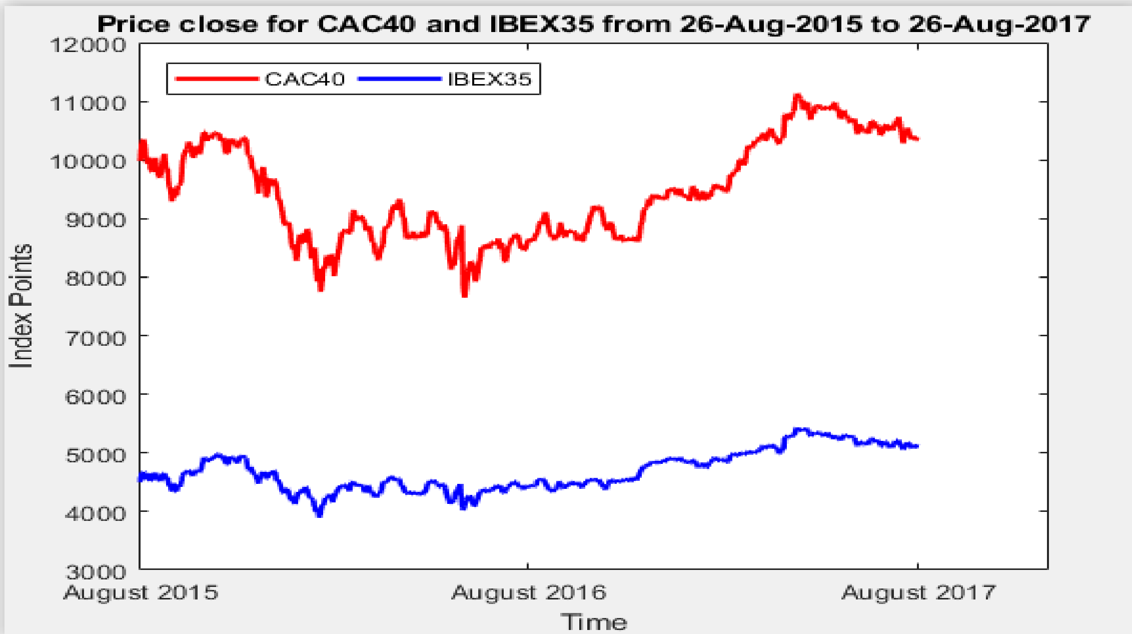

FIGURE 18 PLOT OF CAC40 AND IBEX35 CLOSE PRICE.

According to the basic graph of the close price for both indexes over a period of 2 years between 2015 and 2016, you can see that the time series are nonstationary because they show a bullish trend. However, if you want to confirm it, you should apply the Dickey-Fuller or Philips Perrot test unit root of Nonstationary.

H0:g1=0-Time series has a unit root and is Nonstationary.

H1: g1<0-Time series does not have a unit root and is stationary.

After estimating the Dickey-Fuller and Philips Perrot test, I can conclude that the time series is nonstationary because the hypothesis null cannot be rejected. In other words, the p-values are higher than 5%.

```
-----------------------
Both Time series have a unit root and are Nonstationary
Pvalueibex35= 6.487120e-01 and Pvaluecac40= 7.904210e-01 >0.05.
-----------------------
```

As consequence, I transform both indexes in their differences and the first differences with logarithms or returns. As you can see in the following graphs, the series are stationaries:

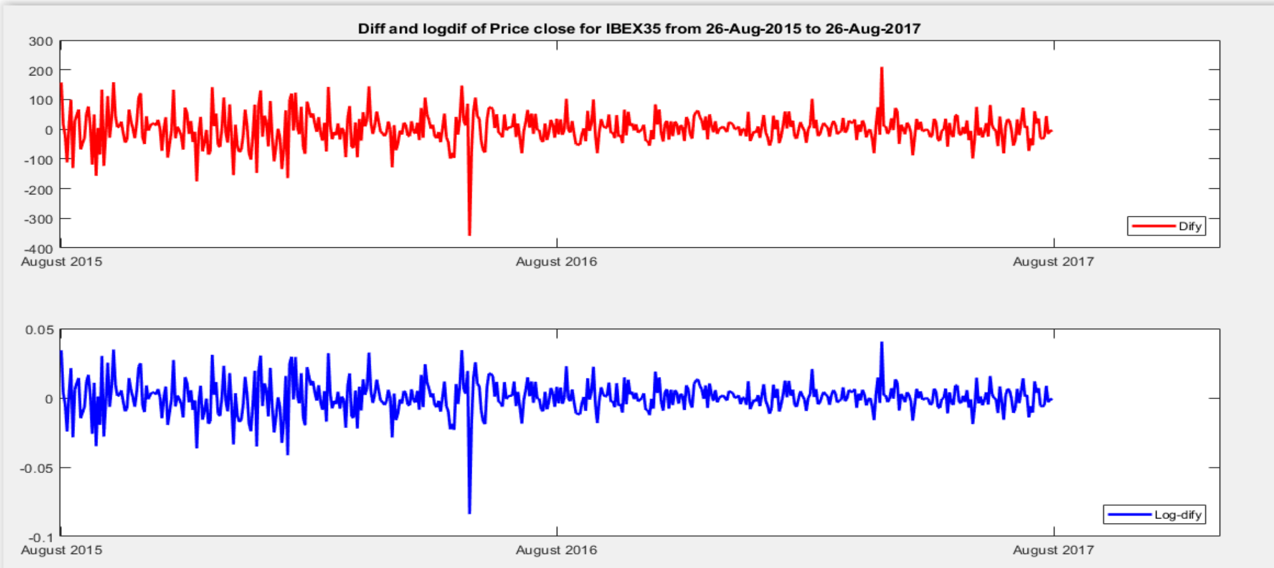

FIGURE 19 TRANFORMATION IBEX 35

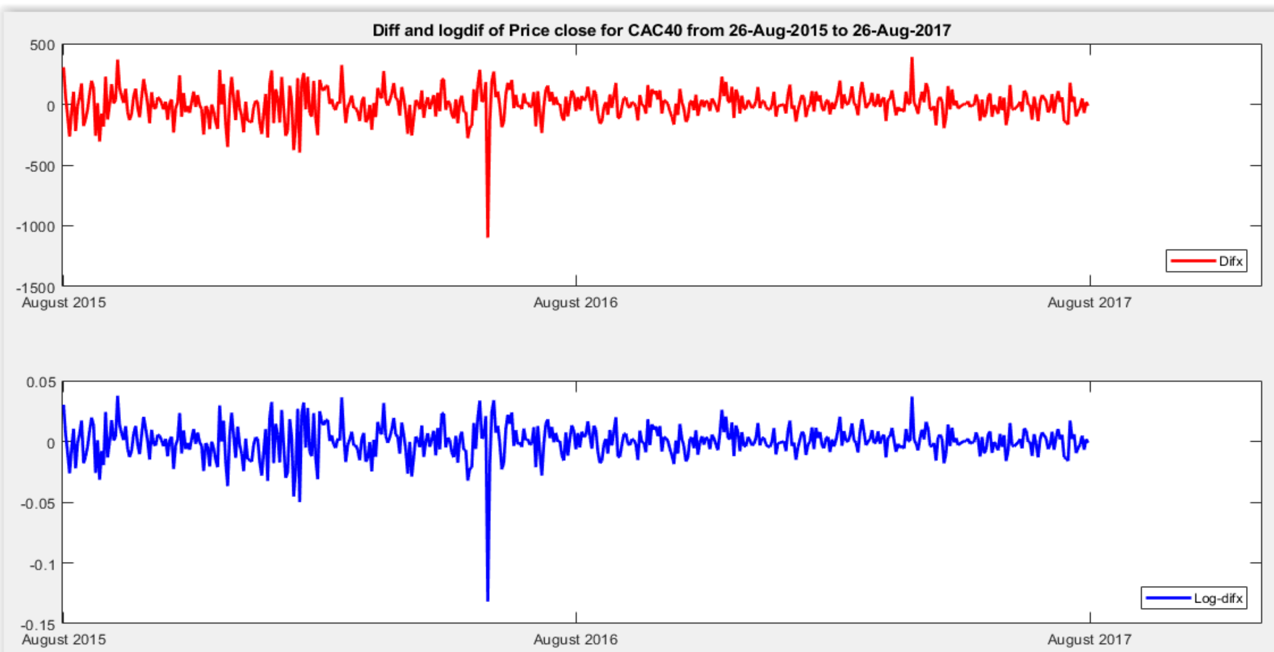

FIGURE 20 TRANSFORMATION CAC40

The correlograms of autocorrelation function and their figures show that return series don't have persistence. It means that they have short memory process because the fluctuation of the ACF decreases quickly, the bars' size is small and most of autocorrelations aren't representative.

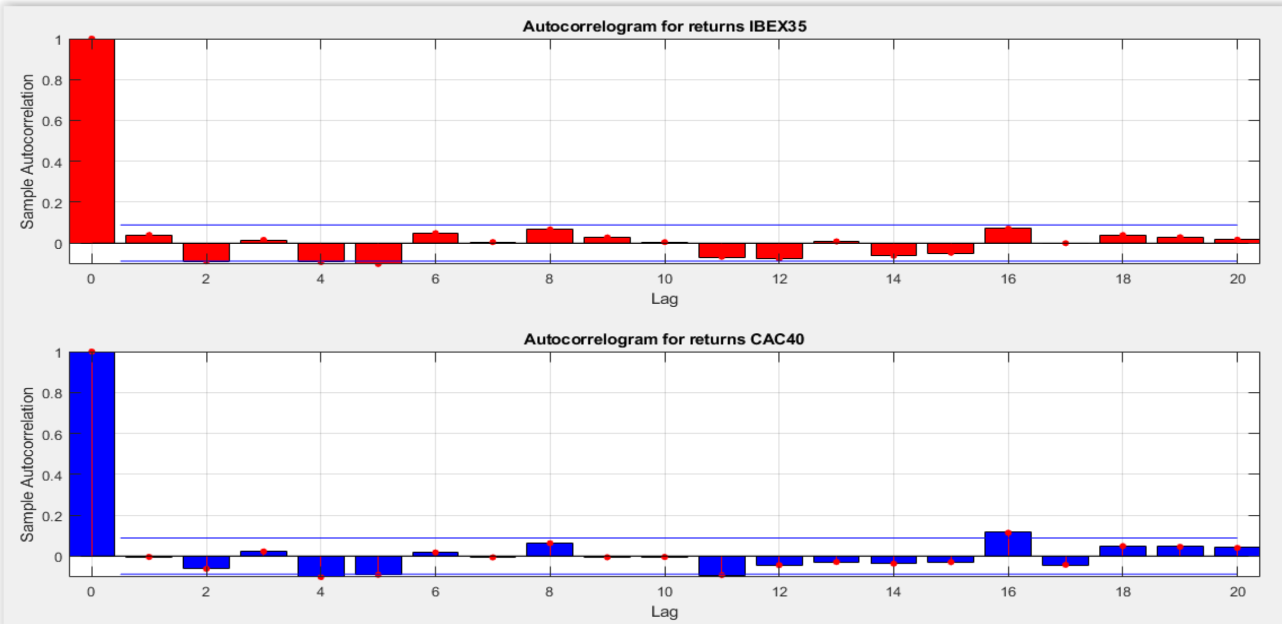

FIGURE 21 CORRELOGRAM OF TIME SERIES RETURN IBEX35 AND CAC40

Additionally, it is interesting to study the close price for the IBEX35 and the CAC40 with ordinary linear square and polynomial functions. For instance, the OLS estimation do not apply because evidently there is not a lineal relation.

In case of polynomial function with degree 5 and 10 the approximation is more suitable, in the following figures 22 and 23 you can see that the pink line is a polynomial of degree 10 which is a good estimation.

The polynomial of degree n is a function with this form and Coefficients a_{n-t}:

$$f(x) = a_n x^n + a_{n-1}x^{n-1} + a_{n-2}x^{n-2} + \cdots .... + a_2x^2 + a_1x^1 + a_0$$

The non-parametric estimation is another way to approximate the real value, first it is necessary to compute h using the Silverman's rule, estimate the Gaussian Kernel as anonymous function and after compute the small and big bandwidth with Nadayara Watson.

$$\text{Silverman's rule:} h = 1.06\sigma_x n^{-\frac{1}{5}}$$

$$\text{Gaussian kernel: k(x)=} \frac{e^{-\frac{u^2}{2}}}{\sqrt{2\pi}}$$

$$\text{Nadaya Watson:} \frac{\sum_{i=1}^{n} y_i k(\frac{x-x_i}{\frac{c1}{h}})}{\sum_{i=1}^{n} k(\frac{x-x_i}{\frac{c1}{h}})} < \hat{\varphi}(x,h) < \frac{\sum_{i=1}^{n} y_i k(\frac{x-x_i}{\frac{c2}{h}})}{\sum_{i=1}^{n} k(\frac{x-x_i}{\frac{c2}{h}})}$$

In the figures 22 and 23 the yellow line and the green line represent the non-parametric estimation to estimate the small and big bandwidth with c1=0.15 and c2=0.7, therefore you can conclude that it is also an appropriate estimation.

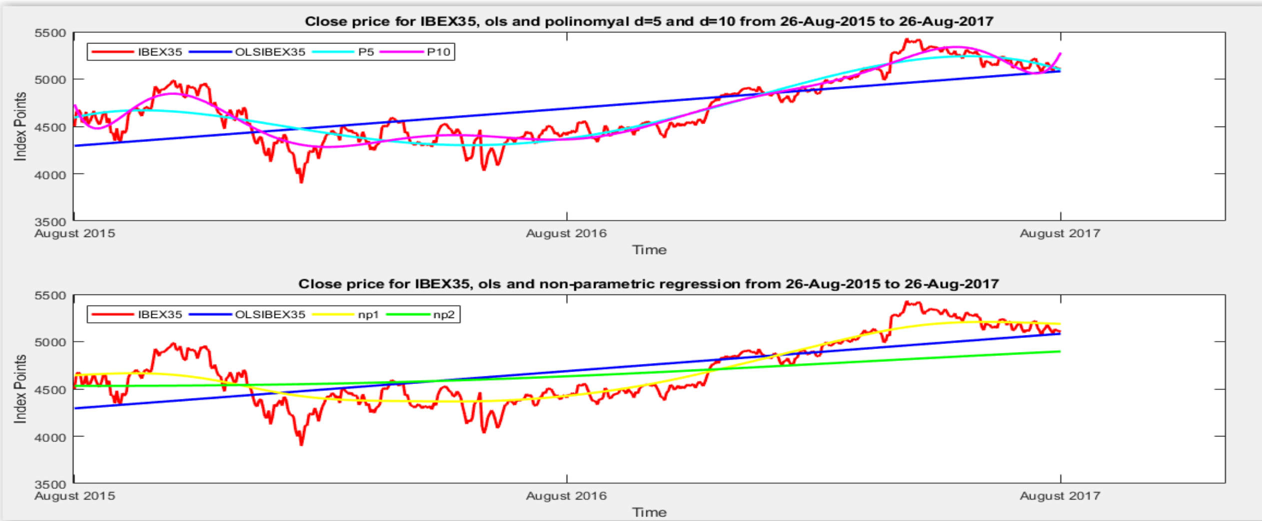

FIGURE 22 GRAPH ORDINARY LINEAR SQUARE, POLYNOMIALS DEGREE 5,10 AND NON-PARAMETRIC REGRESSION FOR IBEX 35.

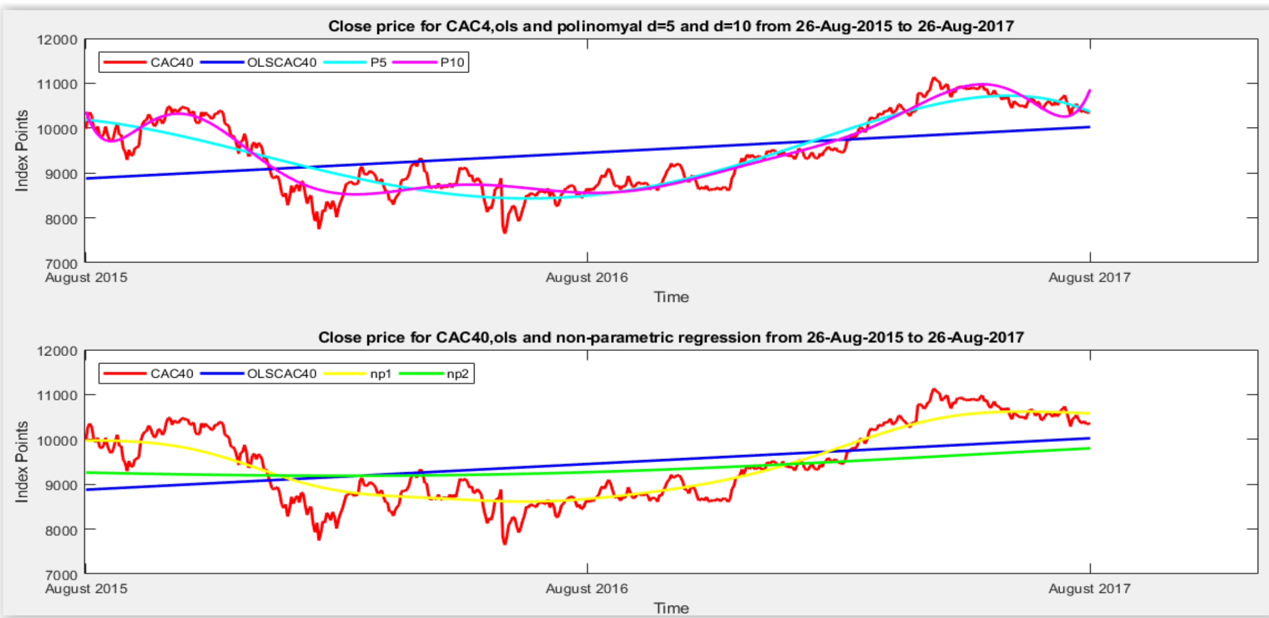

FIGURE 23 GRAPH ORDINARY LINEAR SQUARE, POLYNOMIALS DEGREE 5,10 AND NON-PARAMETRIC REGRESSION FOR CAC40.

In order to apply a better approach, I estimate the best arma model for return of IBEX 35 and CAC 40, which is the lowest Akaike criterium, afterward I compute the price with the following equation:

$$Indexreturn's_t = c + \varepsilon_t + \sum_{i=1}^{p=4} \varphi_i \, Indexreturn's_{t_{t-i}} + \sum_{i=1}^{q=2} \theta_i \, \varepsilon_{t_{t-i}}$$

$$ClosePrice_t = ClosePrice_{t-1} * e^{Indexreturn's_t}$$

The following figures show the best arma model and the close price estimation:

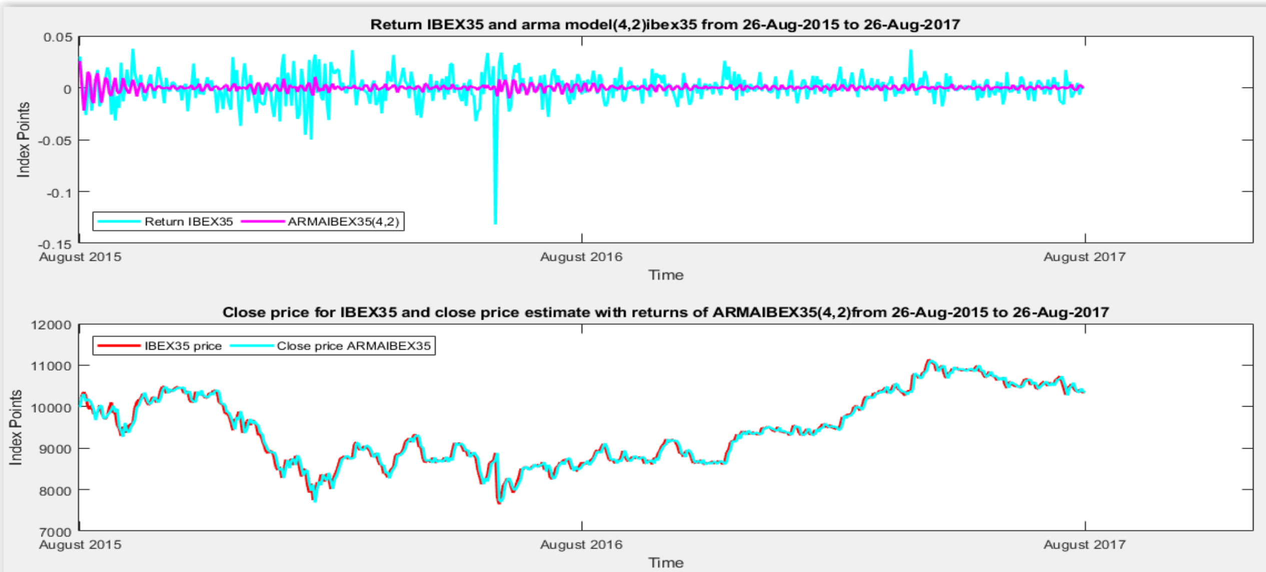

FIGURE 24 ARMA MODEL (4,2) FOR THE IBEX35 RETURN AND CLOSE PRICE

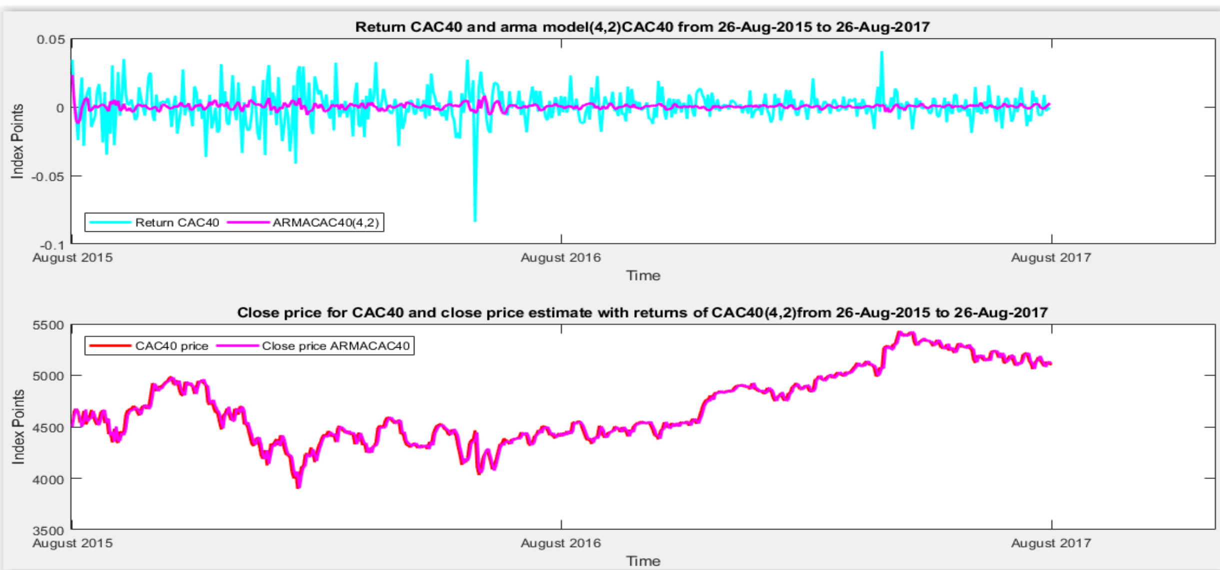

FIGURE 25 ARMA MODEL (4,2) FOR THE CAC40 RETURN AND CLOSE PRICE

I can conclude that the arma model p=4 and q=2 is a good forecast, however it is required to estimate if the sample residual present arch effect. In this case, the p-value is higher than 5%, for this reason I do not reject the null hypothesis about the constant variance.

```
------------------------
Both Time series do not have ARCH Effects-Homoskedasticity
Pvaluex= 6.286393e-01 and Pvaluey= 6.815260e-02 >0.05.
------------------------
```

Now, I introduce an exogeneous variable to propose another kind of model. This type of model is called ARIMAX which has the same structure of the ARMA model, but you can introduce an independent variable. In the bellow charts, you can see in the ARIMAX's estimation in the blue line and the turnover as exogeneous variable.

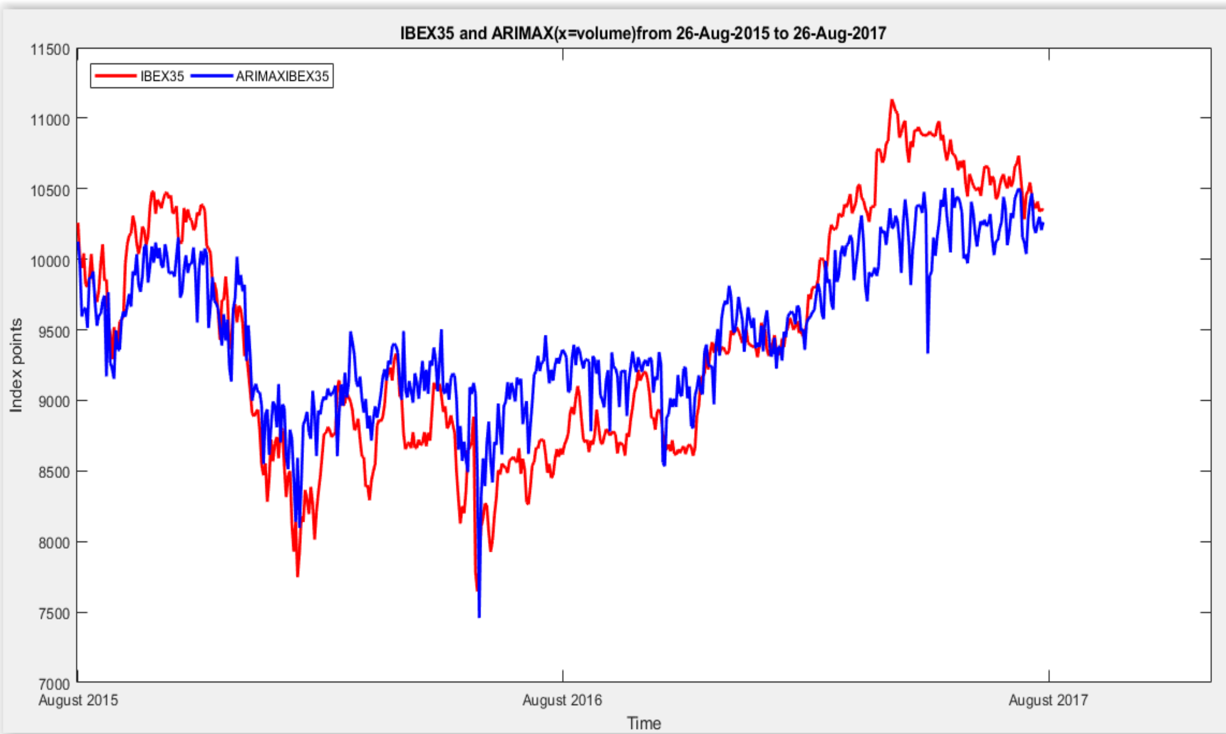

FIGURE 26 ARMAX MODEL (4,2) FOR THE IBEX35 CLOSE PRICE

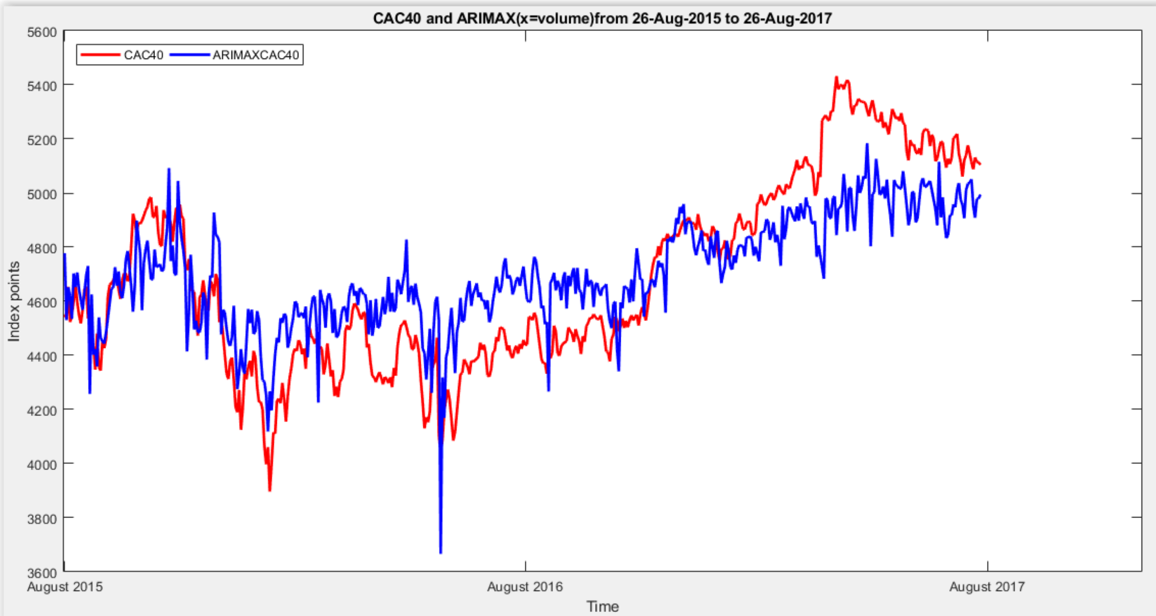

FIGURE 27 ARMA MODEL (4,2) FOR THE CAC40 CLOSE PRICE.

Moreover, it is important to analyze the index's volumes to predict or forecast the dependent variable. In the following chart I show the CAC 40 and IBEX 35 daily volumes.

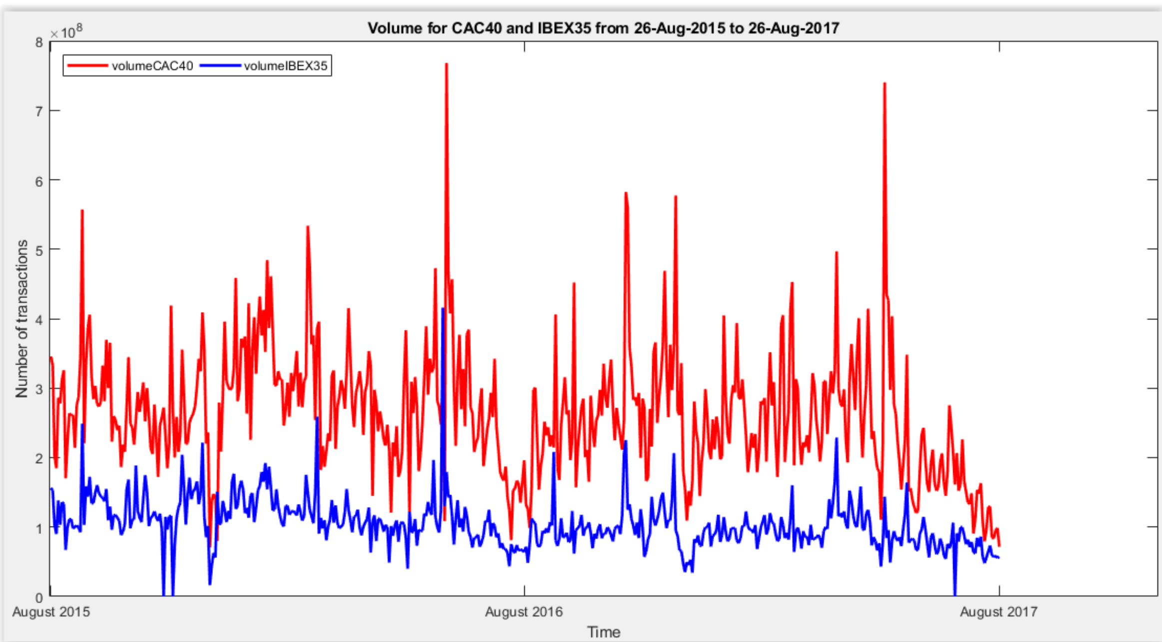

FIGURE 28 PLOT OF THE VOLUME FOR CAC40 AND IBEX35.

After estimating the Dickey-Fuller test, I can conclude that the volume's time series are stationary because I cannot reject the null hypothesis. In other words, the critical value is lower than the t-statistic.

For the arch effect, the p-value is lower than 5%. For this reason, I rejecte the null hypothesis about the constant variance because the series presents a Heteroskedasticity's problem. Then, it is recommend to use a family Garch model.

```
-----------------------
The best GARCH MODEL for the volume of IBEX35 volume
ARMA P= 1 and q= 1 .
The best GARCH MODEL for the volume of CAC40 volume
ARMA P= 1 and q= 1 .
-----------------------
```

The best garch model is estimated using a boucle with the arima and estimate matlab's functions, next I choose the model with the lowest Akaike criterium. The bellow graphs exhibit the conditional volatility for IBEX 35 and CAC 40 and the generation of 10 paths or samples.

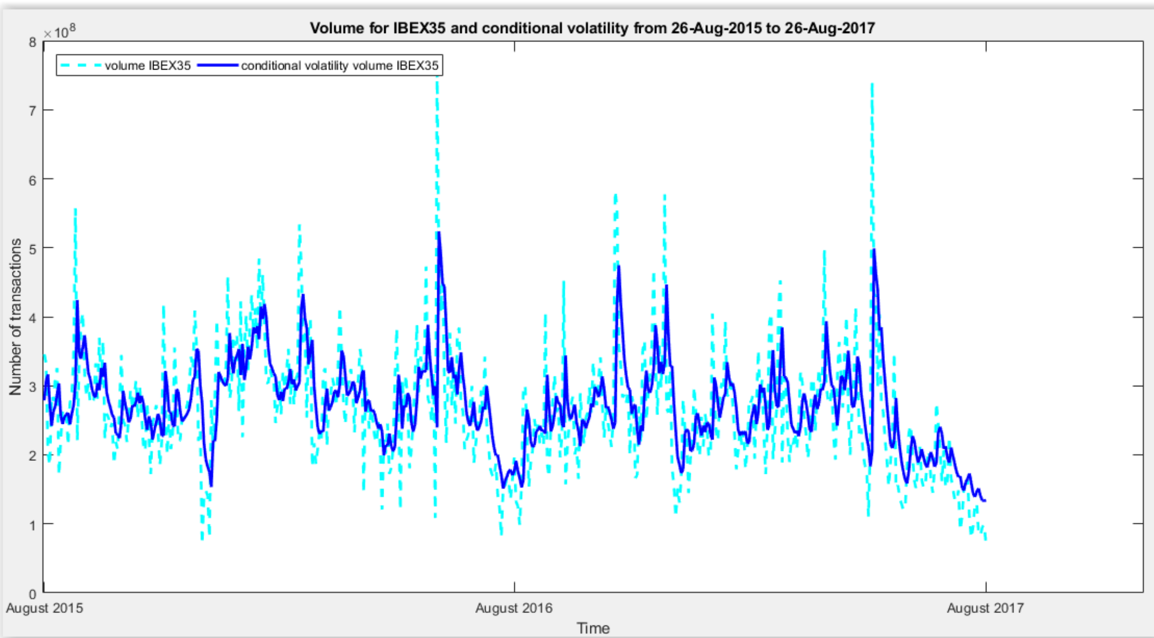

FIGURE 29 PLOT OF THE VOLUME FOR IBEX35 AND THE CONDITIONAL VOLATILITY.

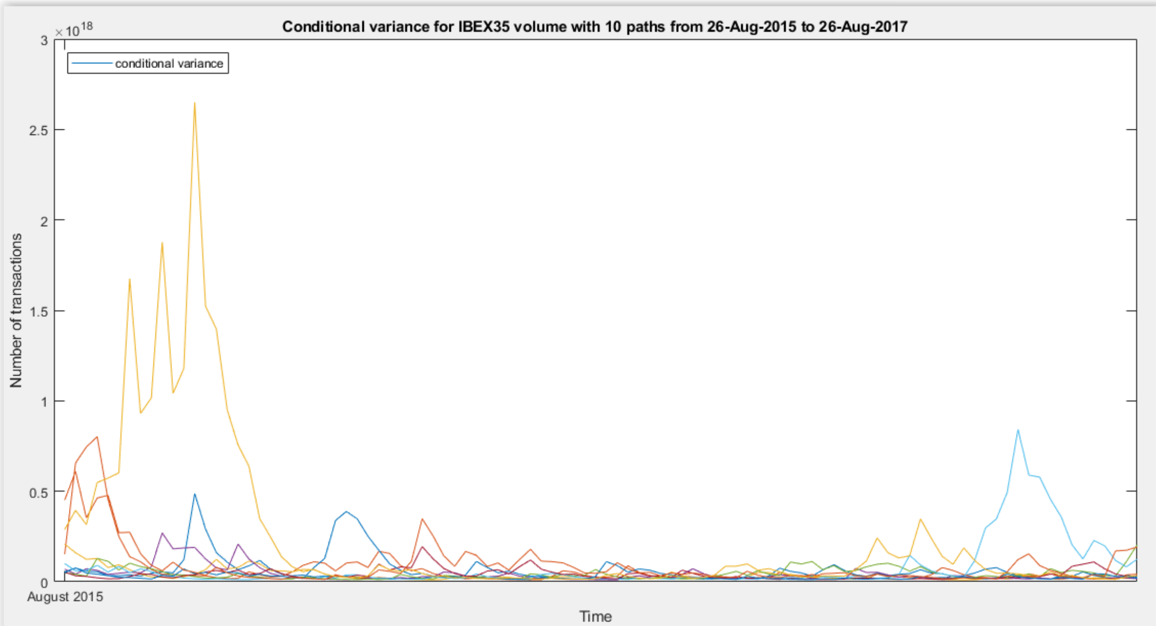

FIGURE 30 PLOT OF THE CONDITIONAL VOLATILITY FOR IBEX35 WITH 10 PATHS.

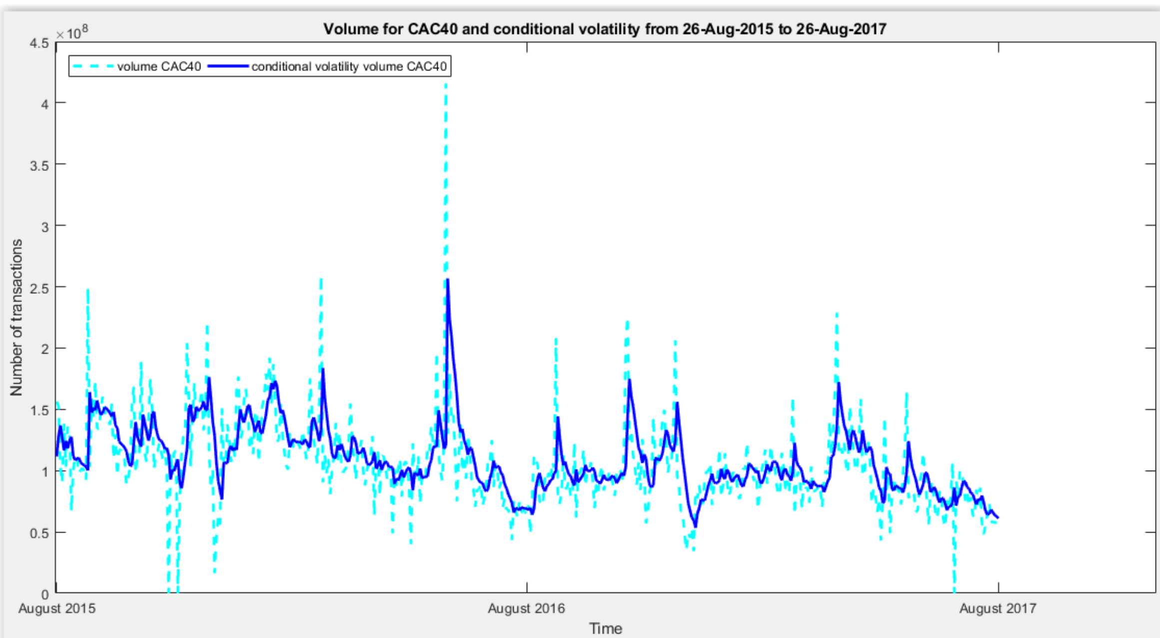

FIGURE 31 PLOT OF THE VOLUME FOR CAC40 AND THE CONDITIONAL VOLATILITY.

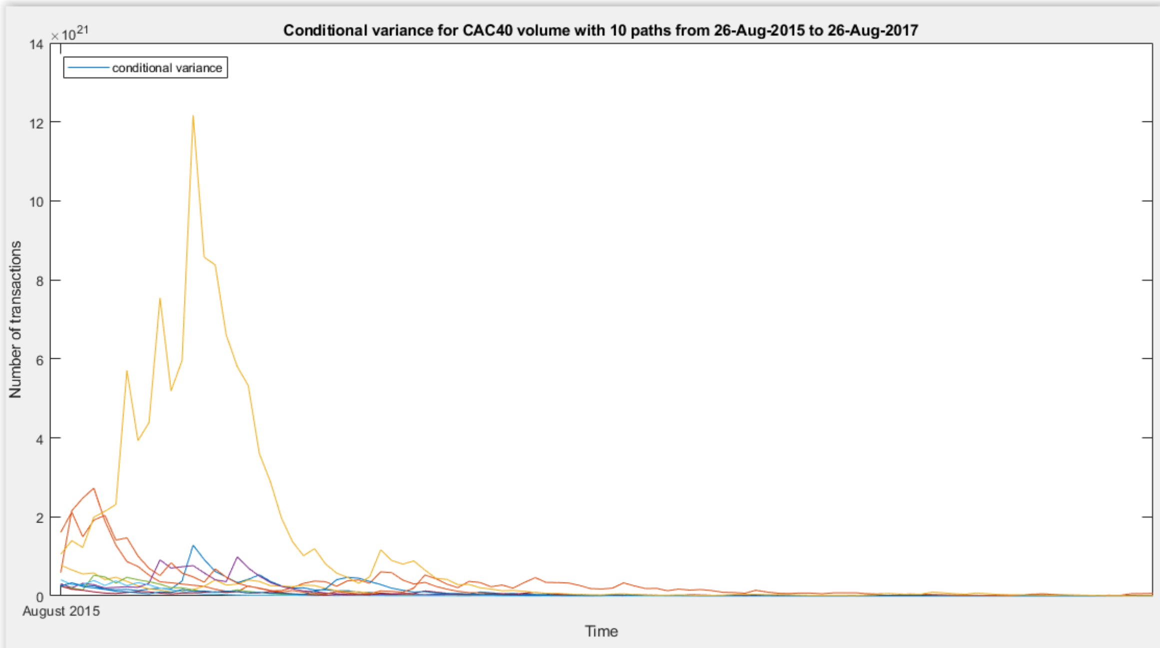

FIGURE 32 PLOT OF THE CONDITIONAL VOLATILITY FOR CAC40 WITH 10 PATHS

10. Auxiliary Functions

In this project, I use different tools to study the index time series. First, I present the auxiliary Matlab's functions, functions created by other authors and finally some functions created in this project.

- Matlab Functions: In general, I use the statistic and econometrics tool box, these are some functions that I apply:
  - Datastats(XDATA,YDATA): returns data statistics for XDATA and YDATA in the structures XDS and YDS.
  - Jbtest(X): Jarque bera Test of normality.
  - Kurtosis and skewness functions.
  - Cell2mat(C): converts a multidimensional cell array with contents of the same data type into a single matrix.
  - Adftest(y): Advance Dickey-Fuller tests assess the null hypothesis of a unit root in univariate time series y.
  - Regress(Y,X): returns a vector STATS containing, in the following order, the R-square statistic, the F-statistic and p-value for the full model, and an estimate of the error variance.
  - Polyfit(X,Y,N): finds the coefficients of a polynomial P(X) of degree N.
  - Polyval(P,X): returns the value of a polynomial P evaluated at X.
  - Arima(p,D,q): model by specifying either the degrees p, D, and q and allows the creation of an arima model of the general form.
  - Estimate(Mdl,Y): estimate the parameters of a regression model with ARIMA time series errors. The estimation process infers the disturbances of the underlying response series and then fits the model to the response data via maximum likelihood.
  - Archtest(res): The ARCH test of Engle.
  - Garch(P,Q): Create a garch(P,Q) conditional variance model by specifying either the degrees P and Q.
  - Infer(Mdl,Y): Infer the innovations and unconditional disturbances of a univariate regression model with ARIMA time series errors.
  - Simulate(Mdl,numObs): Simulate sample paths of responses, unconditional disturbances, and innovations of a univariate regression model with ARIMA errors.

- Functions created by other authors:
    - DownloadStocksData(Asset,Date1,Date2,interval)
      Author: Aitor Roca
      Source: https://fr.mathworks.com/matlabcentral/fileexchange/64218-download-historical-financial-data-from-yahoo-?focused=8095300&tab=function
    - Arma_reg(y, p, q): This function uses a regression approach to initialize the coefficients of the ARMA process.
      Author: Protopopescu 2010
      Source: Aix-Marseille University

- Functions creates in this project:
    - Autocovariance_empiric(x,H): Estimate autocovariance function (Yule–Walker equations).
    - RSS2(residuals): This function computes residual sum squared.
    - Aic_criterion (d,N,RSS2):This function compute the Akaike criterium.

11. Conclusions

Finally, I can conclude that the both indexes do not show a lineal relation and the polynomial function is a better approximation than OLS, as well as, the non-parametric regression is an appropriate estimation.

However, the Arma model presents a better approach because it studies the preview data and the indexes’ time series do not present an arch effect. For this reason, it is not necessary to use a Garch family model.

In addition, I apply an Armax model to introduce another variable since it produces a forecast with an independent variable and the historical data. Nevertheless, in some periods the forecast presents a small deviation from the real value. Moreover, it is necessary to analyze the exogenous variable within its own model.

In other words, the independent variables present heteroskedasticity, as a result I use a Garch model to predict the conditional variance.

The Jarque Bera’s test reflects that indexes and their volumes do not follow a normal distribution and they are more or lower peaked than normal distribution. As consequence, it is important to mmention that the gaussian law is a strong hypothesis because normally the financial time series do not follow this kind of movements.

Another important conclusion is that CAC40 fluctuates with superior units and the range of its movement is higher than IBEX35, for this reason the CAC40 presents a higher volatility than the Spain’s index.

12. References

- Analysis of Financial Time Series, 2nd Edition, Ruey S. Tsay, University of Chicago, Graduate School of Business.
- Handbook of volatility models and their aplications. Bauwens Luc, Hafner Christian and Laurent Sebastien. Wiley & Sons Inc. Publication.
- A Review of Kernel Density Estimation with Applications to Econometrics. Adriano Z. Zambom and Ronaldo Dias.
- Econometrics Toolbox user's Guide. The MathWorks, Inc.