CODE FROM STATA. I COULD NOT UPLOAD IT AS DO-FILE. I COPIED THE CODE IN THIS WORD DOCUMENT.

\*Working Directory\*

cd ""

import excel using DATABASE\_final.xlsx, firstrow clear

\*DESTRING VARIABLES\*

destring REC\_FR REC\_SP REC\_GM, replace

\*RENAME\*

rename (importexcelusingDATABASExlsx)(YEARS)

tsset YEARS // For the three countries, Series cover the 1986-2019 period

\* Preliminary Statistics

sum Y\_FR K\_FR L\_FR EXP\_FR NEC\_FR REC\_FR Y\_SP K\_SP L\_SP EXP\_SP NEC\_SP REC\_SP Y\_GM K\_GM L\_GM EXP\_GM NEC\_GM REC\_GM, detail

tabstat Y\_FR K\_FR L\_FR EXP\_FR NEC\_FR REC\_FR Y\_SP K\_SP L\_SP EXP\_SP NEC\_SP REC\_SP Y\_GM K\_GM L\_GM EXP\_GM NEC\_GM REC\_GM, stat(mean median sd skewness kurtosis range iqr cv)

\* Jbera

vec Y\_FR K\_FR L\_FR EXP\_FR NEC\_FR REC\_FR Y\_SP K\_SP L\_SP EXP\_SP NEC\_SP REC\_SP Y\_GM K\_GM L\_GM EXP\_GM NEC\_GM REC\_GM

vecnorm

\*GENERATING LOG, FIRST DIFFERENCE AND EXP SERIES\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

gen LY\_FR= ln(Y\_FR)

gen LK\_FR= ln(K\_FR)

gen LL\_FR= ln(L\_FR)

gen LEXP\_FR= ln(EXP\_FR)

gen LNEC\_FR= ln(NEC\_FR)

gen LREC\_FR= ln(REC\_FR)

gen LY\_SP= ln(Y\_SP)

gen LK\_SP= ln(K\_SP)

gen LL\_SP= ln(L\_SP)

gen LEXP\_SP= ln(EXP\_SP)

gen LNEC\_SP= ln(NEC\_SP)

gen LREC\_SP= ln(REC\_SP)

gen LY\_GM= ln(Y\_GM)

gen LK\_GM= ln(K\_GM)

gen LL\_GM= ln(L\_GM)

gen LEXP\_GM= ln(EXP\_GM)

gen LNEC\_GM= ln(NEC\_GM)

gen LREC\_GM= ln(REC\_GM)

gen Y\_FR1= exp(Y\_FR)

gen K\_FR1= exp(K\_FR)

gen L\_FR1= exp(L\_FR)

gen EXP\_FR1= exp(EXP\_FR)

gen NEC\_FR1= exp(NEC\_FR)

gen REC\_FR1= exp(REC\_FR)

gen Y\_SP1= exp(Y\_SP)

gen K\_SP1= exp(K\_SP)

gen L\_SP1= exp(L\_SP)

gen EXP\_SP1= exp(EXP\_SP)

gen NEC\_SP1= exp(NEC\_SP)

gen REC\_SP1= exp(REC\_SP)

gen Y\_GM1= exp(Y\_GM)

gen K\_GM1= exp(K\_GM)

gen L\_GM1= exp(L\_GM)

gen EXP\_GM1= exp(EXP\_GM)

gen NEC\_GM1= exp(NEC\_GM)

gen REC\_GM1= exp(REC\_GM)

gen DLY\_FR= d.Y\_FR

gen DLK\_FR= d.K\_FR

gen DLL\_FR= d.L\_FR

gen DLEXP\_FR= d.EXP\_FR

gen DLNEC\_FR= d.NEC\_FR

gen DLREC\_FR= d.REC\_FR

gen DLY\_SP= d.Y\_SP

gen DLK\_SP= d.K\_SP

gen DLL\_SP= d.L\_SP

gen DLEXP\_SP= d.EXP\_SP

gen DLNEC\_SP= d.NEC\_SP

gen DLREC\_SP= d.REC\_SP

gen DLY\_GM= d.Y\_GM

gen DLK\_GM= d.K\_GM

gen DLL\_GM= d.L\_GM

gen DLEXP\_GM= d.EXP\_GM

gen DLNEC\_GM= d.NEC\_GM

gen DLREC\_GM= d.REC\_GM

gen DY\_FR1= d.Y\_FR1

gen DK\_FR1= d.K\_FR1

gen DL\_FR1= d.L\_FR1

gen DEXP\_FR1= d.EXP\_FR1

gen DNEC\_FR1= d.NEC\_FR1

gen DREC\_FR1= d.REC\_FR1

gen DY\_SP1= d.Y\_SP1

gen DK\_SP1= d.K\_SP1

gen DL\_SP1= d.L\_SP1

gen DEXP\_SP1= d.EXP\_SP1

gen DNEC\_SP1= d.NEC\_SP1

gen DREC\_SP1= d.REC\_SP1

gen DY\_GM1= d.Y\_GM1

gen DK\_GM1= d.K\_GM1

gen DL\_GM1= d.L\_GM1

gen DEXP\_GM1= d.EXP\_GM1

gen DNEC\_GM1= d.NEC\_GM1

gen DREC\_GM1= d.REC\_GM1

\* LAG LENGTH SELECTION \*

varsoc LY\_FR

varsoc LK\_FR

varsoc LL\_FR

varsoc LEXP\_FR

varsoc LNEC\_FR

varsoc LREC\_FR

varsoc LY\_SP

varsoc LK\_SP

varsoc LL\_SP

varsoc LEXP\_SP

varsoc LNEC\_SP

varsoc LREC\_SP

varsoc LY\_GM

varsoc LK\_GM

varsoc LL\_GM

varsoc LEXP\_GM

varsoc LNEC\_GM

varsoc LREC\_GM

\*\*\*VECTOR ERROR CORRECTION MODEL\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*FR\*

vec LY\_FR LNEC\_FR LREC\_FR LK\_FR LL\_FR LEXP\_FR, rank(1)

\*SP\*

vec LY\_SP LNEC\_SP DLREC\_SP LK\_SP LL\_SP LEXP\_SP, rank(2) lags(3)

\*GM\*

vec LY\_GM LNEC\_GM LREC\_GM LK\_GM LL\_GM LEXP\_GM, rank(1)

vecnorm

 \*IMPULSE RESPONSE FUNCTIONS\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*FRANCE\*

var Y\_FR NEC\_FR REC\_FR, lags(1)

irf set irf1

irf create irf1, step(10)

irf graph irf

\*SPAIN\*

var DLY\_SP DLNEC\_SP REC\_SP1, lags(1)

irf set irf2

irf create irf2, step(10)

irf graph irf

\*GERMANY\*

var Y\_GM NEC\_GM REC\_GM, lags(1)

irf set irf3

irf create irf3, step(10)

irf graph irf

\*GREGORY HANSEN COINTEGRATION TEST: BREAK LEVEL TREND REGIME REGIMETREND\*\*\*\*\*\*\*\*

ghansen LL\_FR LNEC\_FR LREC\_FR LY\_FR, break(level) lagmethod(downt)

ghansen LL\_FR LNEC\_FR LREC\_FR LY\_FR, break(trend) lagmethod(downt)

ghansen LL\_FR LNEC\_FR LREC\_FR LY\_FR, break(regime) lagmethod(downt)

ghansen LL\_FR LNEC\_FR LREC\_FR LY\_FR, break(regimetrend) lagmethod(downt)

ghansen LREC\_SP LY\_SP, break(level) lagmethod(downt)

ghansen LNEC\_SP LY\_SP, break(trend) lagmethod(downt)

ghansen LNEC\_SP LY\_SP, break(regime) lagmethod(downt)

ghansen LNEC\_SP LY\_SP, break(regimetrend) lagmethod(downt)

ghansen LL\_GM LNEC\_GM, break(level) lagmethod(downt)

ghansen LL\_GM LNEC\_GM LREC\_GM LY\_GM, break(trend) lagmethod(downt)

ghansen LL\_GM LNEC\_GM, break(regime) lagmethod(downt)

ghansen LL\_GM LNEC\_GM, break(regimetrend) lagmethod(downt)

\* GRAPH\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

tsline NEC\_FR1 NEC\_SP1 NEC\_GM1, scheme(s2mono) name(NEC)

tsline REC\_FR1 REC\_SP1 REC\_GM1, scheme(s2mono) name(REC)

tsline Y\_FR1 Y\_SP1 Y\_GM1, scheme(s2mono) name(Y)

graph combine NEC REC Y, name(leftright2)

\* ROBUSTNESS: Correlation matrix\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*FRANCE\*

sktest LY\_FR LNEC\_FR LREC\_FR LK\_FR LL\_FR LEXP\_FR

\*SPAIN\*

sktest LY\_SP LNEC\_SP LREC\_SP LK\_SP LL\_SP LEXP\_SP

\*GERMANY\*

sktest LY\_GM LNEC\_GM LREC\_GM LK\_GM LL\_GM LEXP\_GM

\*FRANCE\*

correlate LY\_FR LNEC\_FR LREC\_FR LK\_FR LL\_FR LEXP\_FR

\*SPAIN\*

correlate LY\_SP LNEC\_SP LREC\_SP LK\_SP LL\_SP LEXP\_SP

\*GERMANY\*

correlate LY\_GM LNEC\_GM LREC\_GM LK\_GM LL\_GM LEXP\_GM

\*FRANCE

pwcorr LY\_FR LNEC\_FR LREC\_FR LK\_FR LL\_FR LEXP\_FR, sig bon star(.05)

\*SPAIN\*

pwcorr LY\_SP LNEC\_SP LREC\_SP LK\_SP LL\_SP LEXP\_SP, sig bon star(.05)

\*GERMANY\*

pwcorr LY\_GM LNEC\_GM LREC\_GM LK\_GM LL\_GM LEXP\_GM, sig bon star(.05)

\*We would like to thank the reviewers and the editor for their valuable comments\*

\*Working Directory\*

cd "C:\Users"

import excel using Database.xlsx, firstrow clear

tsset Years

\*DESTRING VARIABLES\*

destring Y\_FR K\_FR L\_FR EXP\_FR NEC\_FR REC\_FR Y\_SP K\_SP L\_SP EXP\_SP NEC\_SP REC\_SP Y\_GM K\_GM L\_GM EXP\_GM NEC\_GM REC\_GM, replace

gen DY\_FR =d.Y\_FR

gen DK\_FR =d.K\_FR

gen DL\_FR =d.L\_FR

gen DEXP\_FR =d.EXP\_FR

gen DNEC\_FR =d.NEC\_FR

gen DREC\_FR =d.REC\_FR

gen DY\_SP =d.Y\_SP

gen DK\_SP =d.K\_SP

gen DL\_SP =d.L\_SP

gen DEXP\_SP =d.EXP\_SP

gen DNEC\_SP =d.NEC\_SP

gen DREC\_SP =d.REC\_SP

gen DY\_GM =d.Y\_GM

gen DK\_GM =d.K\_GM

gen DL\_GM =d.L\_GM

gen DEXP\_GM =d.EXP\_GM

gen DNEC\_GM =d.NEC\_GM

gen DREC\_GM =d.REC\_GM

\*Zivot-Andrews Unit Root Test with Structural Breaks\*

zandrews DY\_FR, break(intercept) lagmethod(AIC)

zandrews DY\_FR, break(both) lagmethod(AIC)

zandrews DK\_FR, break(intercept) lagmethod(AIC)

zandrews DK\_FR, break(both) lagmethod(AIC)

zandrews DL\_FR, break(intercept) lagmethod(AIC)

zandrews DL\_FR, break(both) lagmethod(AIC)

zandrews DEXP\_FR, break(intercept) lagmethod(AIC)

zandrews DEXP\_FR, break(both) lagmethod(AIC)

zandrews DNEC\_FR, break(intercept) lagmethod(AIC)

zandrews DNEC\_FR, break(both) lagmethod(AIC)

zandrews DREC\_FR, break(intercept) lagmethod(AIC)

zandrews DREC\_FR , break(both) lagmethod(AIC)

zandrews DY\_SP, break(intercept) lagmethod(AIC)

zandrews DY\_SP, break(both) lagmethod(AIC)

zandrews DK\_SP, break(intercept) lagmethod(AIC)

zandrews DK\_SP, break(both) lagmethod(AIC)

zandrews DL\_SP, break(intercept) lagmethod(AIC)

zandrews DL\_SP, break(both) lagmethod(AIC)

zandrews DEXP\_SP, break(intercept) lagmethod(AIC)

zandrews DEXP\_SP, break(both) lagmethod(AIC)

zandrews DNEC\_SP, break(intercept) lagmethod(AIC)

zandrews DNEC\_SP, break(both) lagmethod(AIC)

zandrews DREC\_SP, break(intercept) lagmethod(AIC)

zandrews DREC\_SP , break(both) lagmethod(AIC)

zandrews DY\_GM, break(intercept) lagmethod(AIC)

zandrews DY\_GM, break(both) lagmethod(AIC)

zandrews DK\_GM, break(intercept) lagmethod(AIC)

zandrews DK\_GM, break(both) lagmethod(AIC)

zandrews DL\_GM, break(intercept) lagmethod(AIC)

zandrews DL\_GM, break(both) lagmethod(AIC)

zandrews DEXP\_GM, break(intercept) lagmethod(AIC)

zandrews DEXP\_GM, break(both) lagmethod(AIC)

zandrews DNEC\_GM, break(intercept) lagmethod(AIC)

zandrews DNEC\_GM, break(both) lagmethod(AIC)

zandrews DREC\_GM, break(intercept) lagmethod(AIC)

zandrews DREC\_GM , break(both) lagmethod(AIC)