Rstudio Do-File

Mr. Ahmad Ilu

February 5, 2020

Commnents

- Option 1

Title: ***How to Conduct GARCH (1,1)***

- load: library(e1071)
- load: library(zoo)
- load: library(rugarch)

How to Declare Dataframe as Time Series

- COMMAND: library(tidyquant)
- COMMAND: library(timetk)
- COMMAND: q <-xts::xts(GARCH_1_1[,-1], order.by = GARCH_1_1$Date)
- COMMAND: Myts=ts(DCC_DATA,start = c(2004,1),end = c(2019,12), frequency = 12)
- COMMAND: str(q)
- COMMAND: head(q)
- COMMAND: chart.TimeSeries(TSR)
- COMMAND: chart.TimeSeries(TSR$ASIRTN)

How to Conduct GARCH (1,1)

GARCH(1,1)

Identify volatility clustering

- command: plot.ts(`ASI RTN`)

Conduct ARCH LM Test

- command: ArchTest(`ASI RTN`)

Ugarch specification

- command: x=ugarchspec(variance.model = list(garchorder = c(1,1)),mean.model = list(armaorder= c(0,0)))
- command: x_fit=ugarchfit(x,data = `ASI RTN`)
- command: x_fit

Forecasting

- command: `ASI RTN`=ugarchforecast(x_fit,n.ahead = 20)
- command: `ASI RTN`

How to Conduct EGARCH

Egarch specification

Y=ugarchspec(variance.model = list(model="eGARCH",garchorder = c(1,1)),mean.model = list(armaorder= c(0,0)))

- command: Y_fit=ugarchfit(Y,data = `ASI RTN`)
- command: Y_fit

Forecasting

- command: `ASI RTN`=ugarchforecast(Y_fit,n.ahead = 20)
- command: `ASI RTN`

POST ESTIMATION

- command: x_fit@fit
- command: Y_fit@fit\protect\T1\textdollarcoef
- command: Y_fit@fit\protect\T1\textdollarresiduals
- command: plot(Y_fit)

UNITROOT TEST

ADF

- COMMAND: adf.test(`ASI RTN`) TO DIFFERCENCE IT

- COMMAND: dasi=diff(`ASI RTN`)

pp

- command: pp.test(`ASI RTN`)
- command: dasi=diff(`ASI RTN`)

DESCRIPTIVE STASTICS & NORMALITY

- COMMAND: FinTS.stats(`EXCH RTN`)
- COMMAND: jarque.bera.test(`EXCH RTN`)

ARDL MODELLING

ARDL MODEL

- COMMAND: Model1=ardlDlm(formula = formula , data = GARCH_1_1 , x = ASIRTN , y = EXCHRTN + ASI + ASI , p = 1 , q = 1 , remove = NULL)
- COMMAND: summary(Model1)

ARDL BOUNDS TESTING

- COMMAND: ardlBound(data = GARCH_1_1, formula = formula, case = 3, p = 3, remove = NULL, autoOrder = FALSE, ic = c("AIC", "BIC", "MASE", "GMRAE"), max.p = 3, max.q = 3, ECM = TRUE, stability = TRUE)

ARDL BOUNDS ORDER

- COMMAND: ardlBoundOrders(data = GARCH_1_1 , formula = formula, ic = c("AIC"), max.p = 15, max.q = 15, FullSearch = FALSE)

NARDL MODELLING

PRE-TEST DIAGNOSTICS

dim.data.frame(THESIS_DATA)

lapply(d,diff)

apply(d, 1, diff)

sapply(d,diff)

library(nardl)

NARDL MODEL FIT

- COMMAND: reg<-nardl(EXCH.R~INF.R,p=4,q=4,d,ic="aic",maxlags = FALSE,graph = FALSE,case=3)
- COMMAND: summary(reg)

- COMMAND: reg<-nardl(EXCH.R~`OIL PRICE*`,p=4,q=1,d,ic="aic",maxlags = TRUE,graph = TRUE,case=3)
- COMMAND: summary(reg)

DYNAMIC MULTILPIER

- COMMAND: plotmplier(reg,reg$np,1,10)

CUSUMQ

e<-reg$rece

k<-reg$k

n<-reg$n

cusum(e=e,k=k,n=n)

cumsq(e=e,k=k,n=n)

#How to Conduct DCC MODEL

To conduct a DCC Model one must compute a standard GARCH/Univariate GARCH Model first in a bid to capture volatility and subsequently a DCC Model to identify and observe the dynamic correlations between the variables.

Recall a Univariate GARCH Model

- command: x=ugarchspec(variance.model = list(garchorder = c(1,1)),mean.model = list(armaorder= c(0,0)))
- command: x_fit=ugarchfit(x,data = `ASI RTN`)
- command: x_fit

TRANSTING to DCC

- command:

x.n<-multispec(replicate(5,ugarchspec(mean.model = list(armaorder= c(0,0)))))

multf=multifit(x.n,TSR)

summary(multf)

DCCspec1=dccspec(uspec = x.n, dccOrder = c(1, 1), distribution = "mvnorm")

DCCfit1=dccfit(DCCspec1, data = TSR, fit.control = list(eval.se = FALSE), fit = multf)

DCCfit1

cov1=rcov(DCCfit1)

cor1=rcoR(fit1)

dim(cov1)

dim(cor1)

DCCF2=dccforecast(DCCfit1,n.ahead = 10)

DCCF2

DCCF2@mforecast

To Ascertain the correlations in the last PERIOD

rcor(DCCfit1, type="R")[„'2018-12-01']

plot(rcor(DCCfit1, type="R")[ASIRTN,ASIRTN,], type='l')

plot(rcor(fit1)[ASIRTN, EXCHRTN,], type='l')

plot(rcov(fit1)[ASIRTN, EXCHRTN,], type='l')

CorBG=covP[4,5,]

diag(rcov(DCCfit1)[„'2018-12-01'])

as.xts(CorBG)

plot(CorBG, type='l')

How to RUN A VAR

DCC_DATA <- read_excel("C:/Users/ACCER/Desktop/RESEARCH COLLECTION/DCC DATA.xlsx")
* View(DCC_DATA)

- q <-xts::xts(DCC_DATA[,-1], order.by = DCC_DATA$Date)
- View(q)
- library(vars)
- plot(DCC_DATA, nc = 2, xlab = " ")
- plot(q, nc = 2, xlab = " ")
- plot(q$OILPRICE, nc = 2, xlab = " ")
- attach(q)
- VARselect(q, lag.max = 8, type = "both")
- q111 <- DCC_DATA[, c("ASIRTN","EXCHRTN","OILPRICE")]
- p1ct <- VAR(q111, p = 1, type = "both")
- p1ct
- summary(p1ct, equation = OILPRICE)
- plot(p1ct, names = "OILPRICE")
- ser11 <- serial.test(p1ct, lags.pt = 16, type = "PT.asymptotic")
- ser11
- ser11$serial
- norm1 <- normality.test(p1ct)
- norm1$jb.mul
- plot(arch1, names = "EXCHRTN")
- plot(stability(p1ct), nc = 2)

JOHANSEN COINTEGRATION

- summary(ca.jo(q111, type = “trace”, ecdet = “trend”, K = 3, spec = “transitory”))
- cointest=summary(ca.jo(q111, type = “trace”, ecdet = “trend”, K = 3, spec = “transitory”))
- cointest
- cointest@teststat
- vecm <- ca.jo(q111[, c(“ASIRTN”, “OILPRICE”,“EXCHRTN”)], type = “trace”, ecdet = “trend”, K = 3, spec = “transitory”)
- vecm.r1 <- cajorls(vecm, r = 1)
- vecm.r1
- vecm.r1$beta

How to plot a Barplot

- ggplot(CORO, aes(x=STATES,y=`CONFIRMED CASES`, fill= STATES)) + geom_bar(stat = “identity”) PIE
- ggplot(CORO, aes(x=STATES,y=`CONFIRMED CASES`, fill= STATES)) + geom_bar(stat = “identity”) + scale_fill_manual(values=c(“red”,“yellow2”,“slateblue4”,“green3”,“orange”, “purple”))
- ggplot(CORO, aes(x=STATES,y=`CONFIRMED CASES`, fill= STATES)) + geom_bar(stat = “identity”) + scale_fill_brewer(palette = “Oranges”)
- ggplot(CORO, aes(x=STATES,y=`CONFIRMED CASES`, fill= STATES)) + geom_bar(stat = “identity”) + scale_fill_brewer(palette = “Oranges”) + ggtitle(“Top 6 COVID Outbreak States”)
- ggplot(CORO, aes(x=STATES,y=`CONFIRMED CASES`, fill= STATES)) + geom_bar(stat = “identity”) + scale_fill_brewer(palette = “Oranges”) + ggtitle(“Top 6 COVID Outbreak States”) + geom_text(aes(label= `CONFIRMED CASES`), vjust= -0.3, size = 3.5)
- ggplot(CORO, aes(x=STATES,y=`CONFIRMED CASES`, fill= STATES)) + geom_bar(stat = “identity”) + scale_fill_brewer(palette = “Reds”) + ggtitle(“Top 6 COVID Outbreak States”) + geom_text(aes(label= `CONFIRMED CASES`), vjust= 1.5, colour= “black”, size = 3.5)

How to plot a PIE CHART

- BARPLOT=ggplot(Book1, aes(x=““,y= VALUE, fill= GROUP)) + geom_bar(stat =”identity”)
- PIECHART=BARPLOT + coord_polar (“y”, start=0)
- PIECHART

How to Condcut a Probit Model

- VAR1=as.factor(CORO$REGION)
- contrasts(VAR1)
- MODEL=glm(VAR1~CCASES+STATES,family = binomial(link = "probit"), data = CORO)
- MODEL
- pnorm(-1.311)

How to Condcut a Logit Model

- Zone=factor(CORO$REGION)
- levels(CORO$REGION)=0:1
- MODEL=glm(VAR1~CCASES+STATES,family = binomial(link = "logit"), data = CORO)
- exp(-4.913e+00)
- chis=MODEL$null.deviance - MODEL$deviance
- dfdiff=MODEL$df.null - MODEL$df.residual
- pchisq(chis,dfdiff,lower.tail = F)
- library(BaylorEdPsych)
- PseudoR2(MODEL)
- library(mfx)
- logitmfx(VAR1 ~ CCASES, data = CORO)

CONFUSION MATRIX

- PMODEL=predict(MODEL,CORO)
- TAB=table(PMODEL>0.5,CORO$REGION)
- TAB
- sum(diag(TAB))/sum(TAB)*100
- 4/(4+2)*100