Notes on Multivariate Volatility Models

Jin-Lung LIn

Department of Finance, National DongHwa University

Why *Multivariate* GARCH (mgarch) Models?

- Volatilities across markets and assets often move together over time,
- Many useful applications include asset pricing models, portfolio selection, hedging, VaR, and volatility spillover among different assets and markets,
- Modeling the temporary dependence of second moments among variables is challenging in financial econometrics.

Illustrate the case for two series (Bivariate)

Two asset return series:

$$\boldsymbol{r}_t = \begin{bmatrix} r_{1t} \\ r_{2t} \end{bmatrix}.$$

Data: $(\boldsymbol{r}_1, \boldsymbol{r}_2, \cdots, \boldsymbol{r}_T)$.

Basic concept

Let F_{t-1} denote the information available at time t-1.

Partition the return as

$$\boldsymbol{r}_t = \boldsymbol{\mu}_t + \boldsymbol{a}_t, \quad \boldsymbol{a}_t = \boldsymbol{\Sigma}_t^{1/2} \boldsymbol{\epsilon}_t$$

where $\boldsymbol{\mu}_t = E(\boldsymbol{r}_t | F_{t-1})$ is the predictable component, and

$$\text{Cov}(\boldsymbol{a}_t | F_{t-1}) = \boldsymbol{\Sigma}_t = \begin{bmatrix} \sigma_{11,t} & \sigma_{12,t} \\ \sigma_{21,t} & \sigma_{22,t} \end{bmatrix},$$

$\{\boldsymbol{\epsilon}_t\}$ are iid 2-dimensional random vectors with mean zero and identity covariance matrix.

Multivariate volatility modeling study time evolution of $\{\boldsymbol{\Sigma}_t\}$.

$\boldsymbol{\Sigma}_t$ is symmetric, i.e. $\sigma_{12,t} = \sigma_{21,t}$

Requirement:

$\boldsymbol{\Sigma}_t$ must be positive definite for all t,

$$\sigma_{11,t} > 0, \quad \sigma_{22,t} > 0, \quad \sigma_{11,t}\sigma_{22,t} - \sigma_{12,t}^2 > 0.$$

For k asset returns, $\boldsymbol{\Sigma}_t$ has $k(k+1)/2$ variables.

The time-varying correlation between r_{1t} and r_{2t} is

$$\rho_{12,t} = \frac{\sigma_{12,t}}{\sqrt{\sigma_{11,t}\sigma_{22,t}}}.$$

Some complications

- Positiveness requirement is not easy to meet
- Too many variables to model
- Not easy to obtain the stationarity conditions for Var$(\boldsymbol{r_t}) = E(\Sigma_t)$

Three approaches of mgarch

- Direct generalization of univariate GARCH model:
  Exponentially weighted covariance, Diagonal VEC model, BEKK model
- Linear combinations of univariate GARCH model:
  Principal Component GARCH model, Factor GARCH model
- Nonlinear combinations of univariate GARCH models:
  Constant Conditional Correlation Model, Dynamic Conditional Correlation Model

Exponentially weighted model (EWMA)

$$\boldsymbol{\Sigma}_t = (1-\lambda)\boldsymbol{a}_{t-1}\boldsymbol{a}'_{t-1} + \lambda\boldsymbol{\Sigma}_{t-1},$$

where $0 < \lambda < 1$. That is,

$$\boldsymbol{\Sigma}_t = (1-\lambda)\sum_{i=1}^{\infty}\lambda^{i-1}\boldsymbol{a}_{t-i}\boldsymbol{a}'_{t-i}.$$

In practice, λ is chosen typically by the RiskMetrics proposal. However, one can estimate λ provided one assumes that a_t follows a multivariate normal distribution with mean 0 and variance $\Sigma_t = \text{Cov}_{t-1}(a_t)$.

Number of parameter = 1.

Splus command

```
 <== To compute covariance given lambda.
cov1 = EWMA.cov(rtn, lambda=0.96)
 <== Estimate lambda
cov2 = mgarch(rtn~1,~ewma1,trace=F)
```

VEC Model

$$\text{vec}(\Sigma_t) \;=\; C + A\text{vec}(a_t a_t') + B\text{vec}(\Sigma_{t-1})$$

where vec (or vech) stacks lower triangle of a $N \times N$ matrix as an $N(N+1)/2 \times 1$ vector:

$$\text{vec}(\Sigma_t) = (\Sigma_{11,t}, \Sigma_{21,t}, \Sigma_{22,t}, \Sigma_{23,t}, \ldots, \Sigma_{NN,t})$$

Problem for VEC models:

VEC is very flexible but number of parameters grows at the rate of $N(N+1)(N(N+1)+1)/2 \quad (21(N=2), 78(N=3), 210(N=4))$

Diagonal VEC model (DVEC)

Restrict A, B to be diagonal matrix or alternatively,

$$\Sigma_t = C + \alpha \otimes (a_t a_t') + \beta \otimes \Sigma_{t-1}$$

For instance, DVEC(1,1) model

$$\begin{aligned}
\sigma_{11,t} &= c_{11} + \alpha_{11} a_{1,t-1}^2 + \beta_{11}\sigma_{11,t-1} \\
\sigma_{12,t} &= c_{12} + \alpha_{12} a_{1,t-1} a_{2,t-1} + \beta_{21}\sigma_{21,t-1} \\
\sigma_{22,t} &= c_{22} + \alpha_{22} a_{2,t-1}^2 + \beta_{22}\sigma_{22,t-1}
\end{aligned}$$

Properties for a DVEC model,

- $\sigma_{ii,t}$ depends only on $a_{ii,t-1}$ and $h_{ii,t-1}$.
- $\sigma_{ij,t}$ depends only on $a_{i,t-1}, a_{j,t-1}$ and $\sigma_{ij,t-1}$.

Problems for DVEC model,

- Σ_t may not be positive definite,
- Model elements separately.

Variants for DVEC (to ensure positiveness):

$$\begin{aligned}
\Sigma_t &= C_0C_0' + (A_1A_1') \otimes (a_{t-1}a_{t-1}') + B_1B_1' \otimes \Sigma_{t-1} \\
\Sigma_t &= CC' + (\boldsymbol{a}_1\boldsymbol{a}_1') \otimes (a_{t-1}a_{t-1}') + \boldsymbol{b}_1\boldsymbol{b}_1' \otimes \Sigma_{t-1} \\
\Sigma_t &= C_0C_0' + (a_1a_1') \otimes (a_{t-1}a_{t-1}') + b_1b_1' \otimes \Sigma_{t-1}
\end{aligned}$$

where $A_1(\boldsymbol{B_1}), \boldsymbol{a_1}(\boldsymbol{b_1}), a_1(b_1)$ are matrix, vector and scalar respectively.

Splus command

```
 fit = mgarch(rtn~1,~dvec(1,1))
 summary(fit)
 names(fit)  <== to some what are stored.
 fit$R.t <== stores correlations
mgarch(rtn~1,~dvec.type.type(p,q), trace=F)
 <== where type could be either mat, vec or scalar.
```

BEKK model (Engle and Kroner (1995))

Simple BEKK(1,1) model

$$\boldsymbol{\Sigma}_t = C + \boldsymbol{\alpha}(\boldsymbol{a}_{t-1}\boldsymbol{a}'_{t-1})\boldsymbol{\alpha}' + \boldsymbol{\beta}\boldsymbol{\Sigma}_{t-1}\boldsymbol{\beta}'$$

where $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ are square matrices without restrictions.

Pros: positive definite

Cons: Many parameters, dynamic relations require further study

Compare BEKK(1,1) and DVEC(1,1)

$$\begin{aligned}
\text{BEKK:}\sigma_{22,t} &= c_{22} + (\alpha_{21}a_{1,t-1} + \alpha_{22}a_{2,t-1})^2 \\
&+ (\beta_{21}^2\sigma_{11,t-1} + 2\beta_{21}\beta_{22}\sigma_{21,t-1} + \beta_{22}^2\sigma_{22,t-1}) \\
\text{DVEC:}\sigma_{22,t} &= c_{22} + \alpha_{22}a_{2,t-1}^2 + \beta_{22}\sigma_{22,t-1}
\end{aligned}$$

Extra α_{12}, β_{12} are included in BEKK(1,1). BEKK is more general than DVEC but is also subject to the dimensional problem when number of variables and number of lags grow.

Splus command

```
fit2=mgarch(rtn~1,~bekk(1,1))
summary(fit2)
names(fit2)
```

Constant Conditional Correlation models (CCC)

$$\begin{aligned}\Sigma_t &= D_t R_t D_t \\ D_t &= diag(\sigma_{11,t}^{1/2} \ldots \sigma_{NN,t}^{1/2}), \\ R_t &= (\rho_{ij,t}), \text{with } \rho_{ii,t} = 1.\end{aligned}$$

R_t is the $N \times N$ matrix of conditional correlation:

$$\sigma_{ij,t} = \rho_{ij,t}\sqrt{\sigma_{ii,t}\sigma_{jj,t}}, \quad \forall i \neq j$$

For CCC,

$$\begin{aligned}R_t &= R = (\rho_{ij}), \\ \sigma_{ijt} &= \rho_{ij}\sqrt{\sigma_{ii,t}\sigma_{jj,t}}\end{aligned}$$

In other words, the dynamics of conditional covariance is determined solely by the dynamics of conditional variances.

CCC reduces the model complexity and number of parameters greatly but might be too restrictive.

Splus command

```
fit3=mgarch(rtn~1,~ccc(1,1))
summary(fit3)
```

```
names(fit3)

 or

fit3=mgarch(rtn~1,~ccc.type(1,1))
summary(fit3)
names(fit3)
<==where type could be any GARCH variants.
```

Dynamic Conditional Correlations (DCC)

Tse and Tsiu (2002): DCC(T)

$$
\begin{aligned}
R_t &= (1-\theta_1-\theta_2)R + \theta_1\Psi_{t-1} + \theta_2 R_{t-1} \\
\Psi_{ij,t-1} &= \frac{\sum_{m=1}^{M}\epsilon_{i,t-m}\epsilon_{j,t-m}}{\sqrt{(\sum_{m=1}^{M}\epsilon^2_{i,t-m})(\sum_{j=1}^{M}\epsilon^2_{j,t-m})}} \\
\epsilon_{it} &= a_{i,t}/\sqrt{\sigma_{ii,t}}
\end{aligned}
$$

where $\theta_1, \theta_2 > 0, \theta_1 + \theta_2 < 1, R$ is a symmetric $N \times N$ positive define matrix with $\rho_{ii} = 1$. Ψ_{t-1} is the sample correlation matrix for $(a_{t-M}, a_{t-M+1}, \ldots, a_{t-1})$ and R_t is a weighted average of correlation matrices (R, Ψ_{t-1}, R_{t-1}).

Engle (2001): DCC(E)

$$
\begin{aligned}
R_t &= (diagQ_t)^{-1/2}Q_t(diagQ_t)^{-1/2} \\
Q_t &= (1-\theta_1-\theta_2)\bar{Q} + \theta_1\epsilon_{t-1}\epsilon'_{t-1} + \theta_2 Q_{t-1} \\
\epsilon_{it} &= a_{it}/\sqrt{\sigma_{ii,t}}
\end{aligned}
$$

Compare DCC(T) with DCC(E)

$$
\begin{aligned}
\text{DCC(T)} \quad \rho_{12,t} &= (1-\theta_1-\theta_2)\rho_{12} + \theta_2\rho_{12,t-1} + \theta_1\frac{\sum_{m=1}^{M}\epsilon_{1,t-m}\epsilon_{2,t-m}}{\sqrt{(\sum_{m=1}^{M}\epsilon^2_{1,t-m})(\sum_{m=1}^{M}\epsilon^2_{2,t-m})}} \\
\text{DCC(E)} \quad \rho_{12,t} &= \frac{(1-\theta_1-\theta_2)\bar{q}_{12} + \alpha\epsilon_{1,t-1}\epsilon_{2,t-1} + \beta q_{12,t-1}}{\sqrt{(1-\theta_1-\theta_2)\bar{q}_{11} + \theta_1\epsilon^2 1, t-1 + \theta_2 q_{11,t-1})((1-\theta_1-\theta_2)\bar{q}_{22} + \theta_1\epsilon^2_{2,t-1} + \theta_2 q_{22,t-1})}}
\end{aligned}
$$

Principal component GARCH models (PGARCH or OGARCH)

Alexander (2000)

Procedure

1. Compute the principal component of the bivariate series
2. Fit specified univariate GARCH model to each component

The procedure can be further revised by

1. Fitting univariate GARCH model to each series to obtain the standardized residuals,
2. Compute principal component of the standardized residuals,
3. Fit univariate GARCH to each component series.

Alternative Models

- Multivariate stochastic volatility models

  Harvey, Ruiz and Shephard (1994)

$$
\begin{aligned}
a_{it} &= \sigma_i \epsilon_{it} exp(0.5 h_{it}) \\
h_t &= \Phi h_{t-1} + \eta_t
\end{aligned}
$$

- Factor GARCH model

  Engle, Ng and Rothschild (1990).

Splus command

```
fit4=mgarch(rtn~1,~prcomp.type(1,1))
summary(fit4)
names(fit4)
```

Leverage Effect

Negative shocks might have larger impact on volatility than positive shocks. In their words, asymmetry exist. For BEKK models, leverage effect can be written as:

$$\sigma_{ij,t} = c_{ij} + \boldsymbol{\alpha}_i' a_{t-1} a_{t-1}' \boldsymbol{\alpha}_i + \boldsymbol{\beta}_i' \Sigma_{t-1} \boldsymbol{\beta}_j + g_i' b_{t-1} b_{t-1}' g_j$$

where $b_t = max(0, -a_t)$.

Splus command:

```
fit5=mgarch(rtn~1,~bekk(1,1), leverage=TRUE)
summary(fit5)
names(fit5)
```

Estimation

Maximum Likelihood

$$\begin{aligned} L_T(\theta, \Sigma) &= \sum_{t=1}^{T} log f(y_t, \theta, \Sigma, I_{t-1}) \\ f(y_t|\theta, \Sigma, I_{t-1}) &= |\Sigma_t|^{-1/2} g(\Sigma_t^{-1/2}(y_t - \mu_t)) \end{aligned}$$

where g is the specified density function like, multivariate Gaussian, multivariate student, GED, etc.

Diagnostic Checking

Diagnostically check:

- iid about $\epsilon_t = a_t/\sigma_t$,
- distribution assumption about ϵ_t,
- functional form of Σ_t.

Prediction

Take DVEC(1,1) for example,

$$\begin{aligned} E_T(\Sigma_{T+1}) &= A_0 + A_1 \otimes (\epsilon_T \epsilon_T') + B_1 \otimes \Sigma_T \\ E_T(\Sigma_{T+2}) &= A_0 + A_1 \otimes E(\epsilon_{T+1} \epsilon_{T+1}') + B_1 \otimes E_T(\Sigma_{T+1}) \\ &= A_0 + (A_1 + B_1) \otimes E_T(\Sigma_{T+1}) \end{aligned}$$

Splus command

```
rtn.predict=predict(rtn.bekk,10)
class(rtn.predict)
names(rtn.predict)
<== [1] "series.pred", "sigma.pred" "R.pred"
```

Zivot an Wang's example

```
# Prepared for the MGARCH Chapter # Date: April 5, 2002
# Two series analyzed:

hp.ibm=seriesMerge(hp.s,ibm.s)
tmp=acf(hp.ibm^2)
hp.ibm.cov=EWMA.cov(hp.ibm,lambda=0.9672375)
seriesPlot(cbind(hp.ibm.cov[,1,1],hp.ibm.cov[,2,2],hp.ibm.cov[,1,2]),
one.plot=F,strip.text=c("HP Vol.","IBM Vol.","Cov."))

hp.ibm.ewma=mgarch(hp.ibm~1,~ewma1,trace=F)
hp.ibm.ewma

mgarch(hp.ibm~1,~ewma2,trace=F)
hp.ibm.dvec=mgarch(hp.ibm~1,~dvec(1,1),trace=F)
class(hp.ibm.dvec)
hp.ibm.dvec
names(hp.ibm.dvec)
coef(hp.ibm.dvec)
sqrt(diag(vcov(hp.ibm.dvec,method="qmle")))
residuals(hp.ibm.dvec,
standardize=T)
summary(hp.ibm.dvec)

autocorTest(residuals(hp.ibm.dvec,standardize=T)^2,lag=12)
archTest(residuals(hp.ibm.dvec,standardize=T),lag=12)
autocorTest(residuals(hp.ibm.dvec,standardize=T)^2,lag=12,bycol=F)
plot(hp.ibm.dvec, ask=F)

hp.ibm.cross=hp.ibm.dvec$R.t[,1,2]
hp.ibm.cross=timeSeries(hp.ibm.cross,pos=positions(hp.ibm))
```

```
seriesPlot(hp.ibm.cross,strip="Conditional Cross Corr.")

mgarch(hp.ibm~1,~dvec.mat.scalar(1,1),trace=F)
hp.ibm.bekk=mgarch(hp.ibm~1,~bekk(1,1))
hp.ibm.bekk

seriesPlot(cbind(hp.ibm.dvec$R.t[,1,2],hp.ibm.bekk$R.t[,1,2]),
          strip=c("DVEC Corr.","BEKK Corr."),one.plot=F,layout=c(1,2,1))

mgarch(hp.ibm~1,~ccc.two.comp(1,1),trace=F)
mgarch(hp.ibm~1,~prcomp.pgarch(1,1,1),trace=F)
mgarch(hp.ibm~1,~egarch(1,1),leverage=T,trace=F)

hp.ibm.beta=mgarch(hp.ibm~seriesData(nyse.s),~dvec(1,1),xlag=1)
summary(hp.ibm.beta)
plot(hp.ibm.beta, ask=F)

weekdaysVec=as.integer(weekdays(positions(hp.ibm)))
MonFriDummy=(weekdaysVec==2|weekdaysVec==6)
hp.ibm.dummy=mgarch(hp.ibm~1,~dvec(1,1)+MonFriDummy )
summary(hp.ibm.dummy)

hp.ibm.dvec.t=mgarch(hp.ibm~1,~dvec(1,1),cond.dist="t")
hp.ibm.dvec.t$cond.dist
hp.ibm.comp=compare.mgarch(hp.ibm.dvec,hp.ibm.dvec.t)
hp.ibm.comp
plot(hp.ibm.comp,qq=T)

class(hp.ibm.dvec$model)
hp.ibm.dvec$model
names(hp.ibm.dvec$model)
```

```
hp.ibm.dvec$model$arch

bekk.mod=hp.ibm.bekk$model
bekk.mod$a.value[2,1]=0
hp.ibm.bekk2=mgarch(series=hp.ibm,model=bekk.mod)

bekk.mod=hp.ibm.bekk$model
bekk.mod$c.value=rep(0,2)
bekk.mod$c.which=rep(F,2)
hp.ibm.bekk3=mgarch(series=hp.ibm,model=bekk.mod)
LR.stat=-2*(hp.ibm.bekk3$likelihood -hp.ibm.bekk$likelihood)
LR.stat
```