

Comandos do SPLUS para Modelagem GARCH

1. Módulo GARCH

- File → Load Module → GARCH
ou
- na janela de comandos,
> module(garch)

2. Ajuste de um Modelo GARCH

A partir de agora, chamemos a série a ser modelada por *rt*.

```
> rt.mod <- garch(rt ~ 1, ~ garch(1,1))
```

Ajusta um modelo GARCH(1,1) com constante.

```
> rt.mod
```

Fornece estimativas dos coeficientes do modelo ajustado. Uma saída mais detalhada é dada por

```
> summary(rt.mod)
```

```
> coefficients(rt.mod)
```

Extrai estimativas dos coeficientes do modelo.

```
> cov(rt.mod)
```

Extrai a matriz de covariâncias das estimativas dos coeficientes. Uma saída melhor é dada por

```
> cov(rt.mod, method="QMLE")
```

```
> residuals(rt.mod) [1:10]
```

Este comando fornece os 10 primeiros resíduos do modelo ajustado

```
> residuals(rt.mod, standardize=T)
```

Este comando fornece os resíduos padronizados.

```
> rt.mod$ df.residual
```

Dá o número de graus de liberdade associados aos resíduos

```
> rt.mod1 <- garch(rt ~ ar(2), ~ garch(1,1))
```

Este comando ajusta um modelo AR(2) a *rt* e depois um modelo GARCH(1,1) aos resíduos do modelo AR.

```
> modelo <- garch(serie ~ arma(2,1), ~ garch(1,1), trace=F)
> modelo$model$AR$value[1]=0
> modelo$model$AR$which[1]=F
> garch(series=serie,model=modelo$model)
```

Ajusta um modelo ARMA(2,1)+GARCH(1,1), com AR(1)=0.

```
> compare(rt.mod,rt.mod1)
```

Compara os ajustes de dois modelos em termos de AIC,BIC, etc

```
> rt.pred <- predict(rt.mod, 200)
```

Calcula 200 previsões do modelo ajustado a *rt*.

```
> plot(rt.pred, series.pred=T)
```

Obtém gráficos das previsões do d.p. condicional e da série.

3. Erros Não- Gaussianos

Os comandos anteriores supõem que a distribuição condicional dos erros é normal (gaussiana). O SPLUS também estima modelos com as seguintes distribuições condicionais dos erros:

(a) *t* de Student, com ν graus de liberdade.

(b) distribuição gaussiana generalizada:

$$f(x) = c. \exp\left\{-\frac{1}{2}|x/\lambda|^\nu\right\},$$

onde

$$c = \frac{\nu}{\lambda 2^{(1+1/\nu)} \Gamma(1/\nu)},$$

$$\lambda = \left[2^{(-2/\nu)} \frac{\Gamma(1/\nu)}{\Gamma(3/\nu)}\right]^{1/2}.$$

(c) A distribuição exponencial dupla é um caso especial de (a), com

$$f(x) = \frac{1}{\sqrt{2}} e^{-\sqrt{2}|x|}.$$

Para especificar uma dessas distribuições use:

```
cond.dist="t"
cond.dist="ged"
cond.dist="double.exp"
```

```
> rt.mod.t <- garch(rt ~ 1, ~ garch(1,1), cond.dist="t")
```

Os parâmetros das distribuições são estimados conjuntamente por MV.

Para especificar o número de g.l. = 3:

```
> rt.mod.t3 <- garch(rt ~ 1, ~ garch(1,1), cond.dist="t", dist.est=F, dist.par=3)
```

No exemplo, para saber o valor de ν estimado, pedir:

```
> rt.mod$ cond.dist
```

4. Extensões do Modelo GARCH

```
> garch(rt ~ 1, ~ egarch(1,1), leverage=T)
```

Ajusta um modelo EGARCH(1,1). Formulação diferente do EViews.

```
> garch(rt ~ 1, ~ tgarch(1,1))
```

Ajusta um modelo TGARCH(1,1).

5. Restrição nos Parâmetros

```
> rt.mod <- revise(rt.fit)
```

rt.fit : nome do modelo ajustado anteriormente.

Fazer a seleção desejada.

Depois Save e Exit.

```
> garch(series=rt, model=rt.mod)
```

Aqui, `rt` é o nome da série e `rt.mod` é o modelo especificado.

Exemplo:

```
> ibv.modt <- garch(ibv ~ 1 + ar(10), ~ garch(1,1), cond.dist="t")
```

```
ibv.modt
```

```
summary(ibv.modt)
```

```
> ibv.revmodt <- revise(ibv.modt)
```

Estipular os parâmetros

```
> ibv.b <- garch(series=ibv, model=ibv.revmodt, cond.dist="t")
```

```
ibv.b
```

```
plot(ibv.b)
```