Financial Econometrics

Overview
Overview

Further packages for time series analysis

- **dse** – Multivariate time series modeling with state-space and vector ARMA (VARMA) models.
- **FinTS** – R companion to Tsay (2005).
- **forecast** – Univariate time series forecasting, including exponential smoothing, state space, and ARIMA models.
- **fracdiff** – ML estimation of ARFIMA models and semiparametric estimation of the fractional differencing parameter.
- **longmemo** – Convenience functions for long-memory models.
- **mFilter** – Time series filters, including Baxter-King, Butterworth, and Hodrick-Prescott.
- **Rmetrics** – Some 20 packages for financial engineering and computational finance, including GARCH modeling in **fGarch**.
- **tsDyn** – Nonlinear time series models: STAR, ESTAR, LSTAR.
- **vars** – (Structural) vector autoregressive (VAR) models.
Financial Econometrics

GARCH Modelling via tseries
GARCH models
GARCH models

`tseries` function `garch()` fits GARCH($p$, $q$) with Gaussian innovations. Default is GARCH(1, 1):

\[
y_t = \sigma_t \nu_t, \quad \nu_t \sim \mathcal{N}(0, 1) \text{ i.i.d.},
\]
\[
\sigma_t^2 = \omega + \alpha y_{t-1}^2 + \beta \sigma_{t-1}^2, \quad \omega > 0, \alpha > 0, \beta \geq 0.
\]

**Example:** DEM/GBP FX returns for 1984-01-03 through 1991-12-31

```r
R> library("tseries")
R> mp <- garch(MarkPound, grad = "numerical", trace = FALSE)
R> summary(mp)
```

Call:
```r
garch(x = MarkPound, grad = "numerical", trace = FALSE)
```

Model:
```
GARCH(1,1)
```

Residuals:
```
     Min      1Q  Median      3Q     Max
-6.79739 -0.53703 -0.00264  0.55233  5.24867
```
GARCH models

Coefficient(s):

| Estimate | Std. Error | t value | Pr(|t|) |
|----------|------------|---------|--------|
| a0       | 0.0109     | 0.0013  | 8.38   | <2e-16 |
| a1       | 0.1546     | 0.0139  | 11.14  | <2e-16 |
| b1       | 0.8044     | 0.0160  | 50.13  | <2e-16 |

Diagnostic Tests:

Jarque Bera Test

data: Residuals

X-squared = 1100, df = 2, p-value <2e-16

Box-Ljung test

data: Squared.Residuals

X-squared = 2.5, df = 1, p-value = 0.1

Remarks:

- **Warning**: OPG standard errors assuming Gaussian innovations.
- More flexible GARCH modeling via `garchFit()` in `fGarch`.

Christian Kleiber, Achim Zeileis © 2008–2017
Financial Econometrics

GARCH Modelling via Rmetrics
Rmetrics

- Initiated and mainly developed by D. Würtz (ETH, Dept. of Theoretical Physics).
- Environment for financial engineering and computational finance.
- Currently comprises some 20 packages: fArma, fAsianOptions, fAssets, fBasics, fBonds, fCalendar, fCopulae, fEcofin, fExoticOptions, fExtremes, fGarch, fImport, fMultivar, fNonlinear, fOptions, fPortfolio, fRegression, fSeries, fTrading, fUnitRoots, fUtilities.
- Unified framework, initially designed for teaching purposes.
- Unified naming conventions via standardized wrappers. For example, arima() from stats appears as armaFit().
- We consider GARCH modelling via garchFit() from fGarch.
GARCH modeling via `garchFit()`

**Example:** DEM/GBP FX returns for 1984-01-03 through 1991-12-31

R> library("fGarch")
R> mp_gf <- garchFit(~garch(1,1), data = MarkPound, trace = FALSE)
R> summary(mp_gf)

Title:
GARCH Modelling

Call:
garchFit(formula = ~garch(1, 1), data = MarkPound, trace = FALSE)

Mean and Variance Equation:
data ~ garch(1, 1)
<environment: 0x5613f8a63330>
[data = MarkPound]

Conditional Distribution:
norm

Coefficient(s):
mu omega alpha1 beta1
GARCH modeling via `garchFit()`

\[-0.0061903 \quad 0.0107614 \quad 0.1531341 \quad 0.8059737\]

Std. Errors:

based on Hessian

Error Analysis:

|          | Estimate | Std. Error | t value | Pr(>|t|) |
|----------|----------|------------|---------|----------|
| mu       | -0.006190 | 0.008462   | -0.732  | 0.464447 |
| omega    | 0.010761  | 0.002838   | 3.793   | 0.000149 |
| alpha1   | 0.153134  | 0.026422   | 5.796   | 6.8e-09  |
| beta1    | 0.805974  | 0.033381   | 24.144  | < 2e-16  |

Log Likelihood:

-1107 normalized: -0.5606

Description:

Thu Oct 12 13:41:42 2017 by user: zeileis

Standardised Residuals Tests:

<table>
<thead>
<tr>
<th>Test</th>
<th>Statistic</th>
<th>p-Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Jarque-Bera Test</td>
<td>R</td>
<td>1060</td>
</tr>
</tbody>
</table>
GARCH modeling via \texttt{garchFit()}

\begin{verbatim}
Shapiro-Wilk Test R W 0.9623 0
Ljung-Box Test R Q(10) 10.12 0.4299
Ljung-Box Test R Q(15) 17.04 0.3163
Ljung-Box Test R Q(20) 19.3 0.5026
Ljung-Box Test R^2 Q(10) 9.063 0.5262
Ljung-Box Test R^2 Q(15) 16.08 0.3769
Ljung-Box Test R^2 Q(20) 17.51 0.6198
LM Arch Test R TR^2 9.771 0.636
\end{verbatim}

Information Criterion Statistics:
\begin{verbatim}
AIC BIC SIC HQIC
1.125 1.137 1.125 1.129
\end{verbatim}

Remarks:

- Benchmark data set for GARCH(1, 1), see McCullough and Renfro \textit{(J. Economic and Social Measurement 1998)}. \texttt{garchFit()} hits the benchmark.
- Note that constant included by default (not possible with \texttt{tseries}).
- Standard errors are from the Hessian.
More on `garchFit()`

`garchFit()` provides

- ARMA models with GARCH-type innovations
- Various innovation distributions: Gaussian, \( t \), GED, including skewed generalizations.
- Several algorithms for maximizing log-likelihood, default is `nlminb`.
- Two methods for initializing recursions.
ARMA models with GARCH components

Mean equation is ARMA

\[ y_t = \mu + \sum_{t-i}^{m} \phi_i y_{t-i} + \sum_{t-j}^{n} \theta_j \varepsilon_{t-j} + \varepsilon_t \]

Variance equation for GARCH\((p, q)\) is

\[ \varepsilon_t = \sigma_t \nu_t, \]
\[ \nu_t \sim \mathcal{D}_\vartheta(0, 1) \text{ i.i.d.,} \]
\[ \sigma_t^2 = \omega + \sum_{i=1}^{p} \alpha_i \varepsilon_{t-i}^2 + \sum_{t-j}^{q} \beta_j \sigma_{t-j}^2. \]
ARMA models with APARCH components

Mean equation is ARMA

\[ y_t = \mu + \sum_{i=1}^{m} \phi_i y_{t-i} + \sum_{j=1}^{n} \theta_j \varepsilon_{t-j} + \varepsilon_t \]

Variance equation for APARCH\((p, q)\) is

\[ \varepsilon_t = \sigma_t \nu_t, \]
\[ \nu_t \sim \mathcal{D}_\vartheta(0, 1) \text{ i.i.d.}, \]
\[ \sigma_t^\delta = \omega + \sum_{i=1}^{p} \alpha_i (|\varepsilon_{t-i}| - \gamma_i \varepsilon_{t-i})^\delta + \sum_{t-j}^{q} \beta_j \sigma_{t-j}^\delta. \]

where \( \delta > 0 \) and the leverage parameters \(-1 < \gamma_i < 1\).

APARCH comprises various GARCH-type models, including ARCH, GARCH, Taylor/Schwert-GARCH, GJR-GARCH, TARCH, NARCH, log-ARCH, . . .
ARMA models with APARCH components

More complex example: Ding, Granger, Engle (J. Emp. Fin. 1993) MA(1)-APARCH(1,1) model for S&P 500 returns (17055 observations)

R> sp_ap <- garchFit(~ arma(0,1) + aparch(1,1),
+   data = ts(100 * sp500dge), trace = FALSE)

Excerpt from summary(sp_ap):

Std. Errors:
   based on Hessian

Error Analysis:

| Parameter | Estimate  | Std. Error | t value | Pr(>|t|) |
|-----------|-----------|------------|---------|----------|
| mu        | 0.020595  | 0.006342   | 3.247   | 0.00116  |
| ma1       | 0.144709  | 0.008346   | 17.338  | < 2e-16  |
| omega     | 0.009991  | 0.001066   | 9.373   | < 2e-16  |
| alpha1    | 0.083792  | 0.004343   | 19.293  | < 2e-16  |
| gamma1    | 0.374182  | 0.028027   | 13.351  | < 2e-16  |

Results broadly agree with original paper (p. 99, eq. (19)), where algorithm was BHHH. (Note: percentage returns!)
ARMA models with APARCH components

Further ARCH-type models:

Taylor-Schwert ARCH (compare Ding, Granger, Engle, eq. (16))

```r
R> sp_tsarch <- garchFit(~ arma(0,1) + garch(1,1), delta = 1,
+ data = ts(100 * sp500dge), trace = FALSE)
```

Threshold ARCH (TARCH)

```r
R> sp_tarch <- garchFit(~ arma(0,1) + garch(1,1), delta = 1,
+ leverage = TRUE, data = ts(100 * sp500dge), trace = FALSE)
```

GJR-GARCH

```r
R> sp_tarch <- garchFit(~ arma(0,1) + garch(1,1), delta = 2,
+ leverage = TRUE, data = ts(100 * sp500dge), trace = FALSE)
```
ARMA models with APARCH components

Specifying innovation distributions:

cond.dist – specification of conditional distributions allowing for "dnorm", "dged", "dstd", "dsnorm", "dsged", "dsstd". Three of these ("dsnorm", "dsged", "dsstd") are skewed. – Thus

GARCH(1,1) with Student-\(t\) (shape parameter estimated)

R> sp_garch_std <- garchFit(~ garch(1,1), cond.dist = "dstd", + data = ts(100 * sp500dge), trace = FALSE)

GARCH(1,1) with Student-\(t_3\) (shape parameter fixed at 3)

R> sp_garch_std3 <- garchFit(~ garch(1,1), + cond.dist = "dstd", shape = 3, include.shape = FALSE, + data = ts(100 * sp500dge), trace = FALSE)

GARCH(1,1) with Laplace (a GED with shape fixed at 1)

R> sp_garch_ged <- garchFit(~ garch(1,1), + cond.dist = "dged", shape = 1, include.shape = FALSE, + data = ts(100 * sp500dge), trace = FALSE)
ARMA models with APARCH components

Further remarks:

- More details regarding fitting process, defaults, etc. upon setting `trace = TRUE`
- `plot()` method offers 12 types of plots: time series, conditional std. dev., ACF of obs. and squared obs., residuals, ACF of residuals and squared residuals, etc.

Example: (ARMA-APARCH cont’d)
Series with superimposed conditional std. dev. is

```R
R> plot(sp_ap, which = 3)
```
ARMA models with APARCH components

Series with 2 Conditional SD Superimposed
Financial Econometrics

Extensions
Additional tools for financial engineering

- Portfolio management: fPortfolio, portfolio offer portfolio selection and optimization.
- Risk management:
  - Classical Value-at-Risk: VaR.
  - Extreme Value Theory models: evd, evdbayes, evir, extRremes, ismec, POT.
  - Multivariate modeling: fCopulae, copula, fgac
- High-frequency data: realized.

More complete overview in CRAN Task View Empirical Finance at
http://CRAN.R-project.org/view=Finance