Introduction to ARCH and GARCH Models

• ARCH (AutoRegressive Conditional Heteroskedasticity) models were proposed by Engle in 1982.
• GARCH (Generalized ARCH) models proposed by Bollerslev in 1986.
• Engle received the Nobel price in 2003. The GARCH model framework is considered as one of the most important contributions in empirical finance over the last 20 years.
• Engle currently resides at NYU and heads the volatility institute.
Robert Engle, NYU

Nobel Prize citation: "for methods of analyzing economic time series with time-varying volatility (ARCH)"
Champion Pairs Skater Too!
The ARCH Family

Bolerslev (2008) identified over 150 different ARCH models. Here are some of the most common:

- GJR-GARCH
- TARCH
- STARCH
- AARCH
- NARCH
- MARCH
- SWARCH
- SNPARCH
- APARCH
- TAYLOR-SCHWERT
- FIGARCH
- FIEGARCH
- Component
- Asymmetric Component
- SQGARCH
- CESGARCH
- Student t
- GED
- SPARCH
MSFT Daily Returns

Returns

Returns^2

dBln(Returns)
Return Autocorrelations

MSFT Returns

GSPC Returns

MSFT Returns^2

GSPC Returns^2

MSFT abs(Returns)

GSPC abs(Returns)
## Summary Statistics

```r
> table.Stats(MSFT.GSPC.ret)

<table>
<thead>
<tr>
<th></th>
<th>MSFT</th>
<th>GSPC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Observations</td>
<td>3082.0000</td>
<td>3082.0000</td>
</tr>
<tr>
<td>NAs</td>
<td>0.0000</td>
<td>0.0000</td>
</tr>
<tr>
<td>Minimum</td>
<td>-0.1560</td>
<td>-0.0903</td>
</tr>
<tr>
<td>Quartile 1</td>
<td>-0.0093</td>
<td>-0.0061</td>
</tr>
<tr>
<td>Median</td>
<td>0.0000</td>
<td>0.0006</td>
</tr>
<tr>
<td>Arithmetic Mean</td>
<td>0.0001</td>
<td>0.0001</td>
</tr>
<tr>
<td>Geometric Mean</td>
<td>-0.0001</td>
<td>0.0000</td>
</tr>
<tr>
<td>Quartile 3</td>
<td>0.0095</td>
<td>0.0063</td>
</tr>
<tr>
<td>Maximum</td>
<td>0.1955</td>
<td>0.1158</td>
</tr>
<tr>
<td>SE Mean</td>
<td>0.0004</td>
<td>0.0002</td>
</tr>
<tr>
<td>LCL Mean (0.95)</td>
<td>-0.0006</td>
<td>-0.0004</td>
</tr>
<tr>
<td>UCL Mean (0.95)</td>
<td>0.0009</td>
<td>0.0006</td>
</tr>
<tr>
<td>Variance</td>
<td>0.0005</td>
<td>0.0002</td>
</tr>
<tr>
<td>Stdev</td>
<td>0.0214</td>
<td>0.0137</td>
</tr>
<tr>
<td>Skewness</td>
<td>0.2500</td>
<td>0.0298</td>
</tr>
<tr>
<td>Kurtosis</td>
<td>9.0241</td>
<td>7.3286</td>
</tr>
</tbody>
</table>
```

© Eric Zivot 2012
Specify ARCH(1) Process in rugarch

\[ r_t = \sigma_t z_t \]
\[ z_t \sim iid \ N(0,1) \]
\[ \sigma_t^2 = 0.1 + 0.8 \varepsilon_{t-1}^2 \]

# Use functions from rugarch package

> arch1.spec <- ugarchspec(variance.model = list(garchOrder = c(1,0)),
+                           mean.model = list(armaOrder = c(0,0)),
+                           fixed.pars = list(mu = 0, omega = 0.1,
+                                               alpha1 = 0.8))

> class(arch1.spec)
[1] "uGARCHspec"
attr(,"package")
[1] "rugarch"
Specify ARCH(1) Process

> show(arch1.spec)

*---------------------------------*
|       GARCH Model Spec         |
*---------------------------------*

Conditional Variance Dynamics

GARCH Model : sGARCH(1,0)
Variance Targeting : FALSE

Conditional Mean Dynamics

Mean Model : ARFIMA(0,0,0)
Include Mean : TRUE
GARCH-in-Mean : FALSE

Conditional Distribution

Distribution : norm
Includes Skew : FALSE
Includes Shape : FALSE
Includes Lambda : FALSE

© Eric Zivot 2012
Simulate ARCH(1) Process

```r
# Use functions from rugarch package
> set.seed(123)
> arch1.sim = ugarchpath(arch1.spec, n.sim=1000)

> class(arch1.sim)
[1] "uGARCHpath"
attr(,"package")
[1] "rugarch"

> slotNames(arch1.sim)
[1] "path"   "model"   "seed"

> names(arch1.sim@path)
[1] "sigmaSim" "seriesSim" "residSim"

> plot(arch1.sim)
```

© Eric Zivot 2012
Simulated ARCH(1) Process

Simulated Returns

Simulated Volatility
ARCH(1) Autocorrelations

ACF of Returns

ACF of Returns^2

ACF of |Returns|
ARCH(1) Normal QQ-Plot

ARCH(1) with normal errors has fat tails!
Simulate GARCH(1,1) Process

```r
> garch11.spec = ugarchspec(variance.model=list(garchOrder=c(1,1)),
+ mean.model = list(armaOrder=c(0,0)),
+ fixed.pars=list(mu = 0, omega=0.1,
+ alphal=0.1,
+ betal = 0.7))

> set.seed(123)
> garch11.sim = ugarchpath(garch11.spec, n.sim=1000)
```

Note: \( \alpha_1 + \beta_1 = 0.8 \), same as for the ARCH(1)
Simulated GARCH(1,1) Process

Simulated Returns

Simulated Volatility

© Eric Zivot 2012
GARCH(1,1) Autocorrelations

- Returns
- Returns^2
- abs(Returns)
Simulated GARCH(1,1) returns are not far from normal.
Testing for ARCH/GARCH Effects

# use Box.test from stats package
> Box.test(coredata(MSFT.ret^2), type="Ljung-Box", lag = 12)

Box-Ljung test

Q-stat on squared returns

data:  coredata(MSFT.ret^2)
X-squared = 503.4529, df = 12, p-value < 2.2e-16

> Box.test(coredata(GSPC.ret^2), type="Ljung-Box", lag = 12)

Box-Ljung test

Q-stat on squared returns

data:  coredata(GSPC.ret^2)
X-squared = 2973.828, df = 12, p-value < 2.2e-16

© Eric Zivot 2012
Testing for ARCH/GARCH Effects

# Engle’s LM ArchTest() function from FinTS package

```r
> ArchTest(MSFT.ret)

ARCH LM-test; Null hypothesis: no ARCH effects
data:  MSFT.ret
Chi-squared = 246.8778, df = 12, p-value < 2.2e-16

> ArchTest(GSPC.ret)

ARCH LM-test; Null hypothesis: no ARCH effects
data:  GSPC.ret
Chi-squared = 879.794, df = 12, p-value < 2.2e-16
```
Fit GARCH(1,1) to MSFT Returns

\[ r_t = \mu + \varepsilon_t, \quad \varepsilon_t = \sigma_t z_t \]

\[ \sigma_t^2 = \omega + \alpha_1 \varepsilon_{t-1}^2 + \beta_1 \sigma_{t-1}^2 \]

\[ z_t \sim iid \ N(0,1) \]

```r
# specify GARCH(1,1) with constant in mean equation
> garch11.spec = ugarchspec(variance.model=list(garchOrder=c(1,1)),
+                           mean.model = list(armaOrder=c(0,0)))

# estimate GARCH(1,1) by MLE
> MSFT.garch11.fit = ugarchfit(spec=garch11.spec, data=MSFT.ret,
+                              solver.control=list(trace = 1))

Iter: 1 fn: -8042.0206    Pars:  0.0004893676 0.000004681  
  0.07196160  0.918437848
Iter: 2 fn: -8042.0206    Pars:  0.0004893476 0.000004681  
  0.071962506  0.918437308
solnp--> Completed in 2 iterations
```

© Eric Zivot 2012
uGARCHfit Object

> class(MSFT.garch11.fit)
[1] "uGARCHfit"
attr(,"package")
[1] "rugarch"

> slotNames(MSFT.garch11.fit)
[1] "fit" "model"

> names(MSFT.garch11.fit@fit)
[1] "hessian" "cvar" "var"
[4] "sigma" "z" "LLH"
[7] "log.likelihoods" "residuals" "coef"
[10] "robust.cvar" "scores" "se.coef"
[13] "tval" "matcoef" "robust.se.coef"
[16] "robust.tval" "robust.matcoef" "fitted.values"
[19] "convergence" "kappa" "persistence"
[22] "timer" "ipars" "solver"

> names(MSFT.garch11.fit@model)
[1] "modelinc" "modeldesc" "modeldata" "pars" "start.pars"
[6] "fixed.pars" "maxOrder" "pos.matrix" "fmodel" "pidx"
[11] "n.start"
<table>
<thead>
<tr>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>coef()</td>
<td>Extract estimated coefficients</td>
</tr>
<tr>
<td>infocriteria()</td>
<td>Calculate information criteria for fit</td>
</tr>
<tr>
<td>likelihood()</td>
<td>Extract likelihood</td>
</tr>
<tr>
<td>nyblom()</td>
<td>Calculate Hansen-Nyblom coefficient stability test</td>
</tr>
<tr>
<td>signbias()</td>
<td>Calculate Engle-Ng sign bias test</td>
</tr>
<tr>
<td>newsimpact()</td>
<td>Calculate news impact curve</td>
</tr>
<tr>
<td>as.data.frame()</td>
<td>Extract data, fitted data, residuals and conditional vol</td>
</tr>
<tr>
<td>sigma()</td>
<td>Extract conditional volatility estimates</td>
</tr>
<tr>
<td>residuals()</td>
<td>Extract residuals</td>
</tr>
<tr>
<td>fitted()</td>
<td>Extract fitted values</td>
</tr>
<tr>
<td>getspec()</td>
<td>Extract model specification</td>
</tr>
<tr>
<td>gof()</td>
<td>Compute goodness-of-fit statistics</td>
</tr>
<tr>
<td>uncmean()</td>
<td>Extract unconditional mean</td>
</tr>
<tr>
<td>uncvariance()</td>
<td>Extract unconditional variance</td>
</tr>
<tr>
<td>plot()</td>
<td>Produce various plots</td>
</tr>
<tr>
<td>persistence()</td>
<td>Calculate persistence of fitted model</td>
</tr>
<tr>
<td>halflife()</td>
<td>Calculate half-life of fitted model</td>
</tr>
</tbody>
</table>

© Eric Zivot 2008
Summary of GARCH(1,1) Fit

> MSFT.garch11.fit

*---------------------------------*
*          GARCH Model Fit        *
*---------------------------------*

Conditional Variance Dynamics
-----------------------------------
GARCH Model : sGARCH(1,1)
Mean Model : ARFIMA(0,0,0)
Distribution : norm

Optimal Parameters
------------------------------------

| Parameter | Estimate  | Std. Error | t value | Pr(>|t|) |
|-----------|-----------|------------|---------|---------|
| mu        | 0.000489  | 0.000273   | 1.7894  | 0.073557|
| omega     | 0.000005  | 0.000001   | 4.6888  | 0.000003|
| alpha1    | 0.071963  | 0.010254   | 7.0177  | 0.000000|
| beta1     | 0.918437  | 0.011102   | 82.7242 | 0.000000|

Robust Standard Errors:

| Parameter | Estimate  | Std. Error | t value | Pr(>|t|) |
|-----------|-----------|------------|---------|---------|
| mu        | 0.000489  | 0.000298   | 1.6407  | 0.100856|
| omega     | 0.000005  | 0.000003   | 1.7766  | 0.075641|
| alpha1    | 0.071963  | 0.025959   | 2.7722  | 0.005568|
| beta1     | 0.918437  | 0.025206   | 36.4367 | 0.000000|

© Eric Zivot 2012
Summary of GARCH(1,1) Fit

LogLikelihood : 8042

Information Criteria
------------------------------------
Akaike -5.2161  
Bayes -5.2083  
Shibata -5.2161  
Hannan-Quinn -5.2133

Q-Statistics on Standardized Residuals
------------------------------------

<table>
<thead>
<tr>
<th>statistic</th>
<th>p-value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lag10</td>
<td>11.19</td>
</tr>
<tr>
<td>Lag15</td>
<td>17.78</td>
</tr>
<tr>
<td>Lag20</td>
<td>26.32</td>
</tr>
</tbody>
</table>

H0 : No serial correlation

Q-Statistics on Standardized Squared Residuals
------------------------------------

<table>
<thead>
<tr>
<th>statistic</th>
<th>p-value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lag10</td>
<td>1.081</td>
</tr>
<tr>
<td>Lag15</td>
<td>2.300</td>
</tr>
<tr>
<td>Lag20</td>
<td>2.930</td>
</tr>
</tbody>
</table>

Tests for ARCH/GARCH behavior in standardized residuals.
No evidence of serial correlation in squared residuals

© Eric Zivot 2012
## Summary of GARCH(1,1) Fit

### ARCH LM Tests

<table>
<thead>
<tr>
<th>Statistic</th>
<th>DoF</th>
<th>P-Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>ARCH Lag[2]</td>
<td>2</td>
<td>0.8611</td>
</tr>
<tr>
<td>ARCH Lag[5]</td>
<td>5</td>
<td>0.9826</td>
</tr>
<tr>
<td>ARCH Lag[10]</td>
<td>10</td>
<td>0.9997</td>
</tr>
</tbody>
</table>

### Nyblom stability test

- Joint Statistic: 0.9803
- Individual Statistics:
  - mu: 0.09264
  - omega: 0.06068
  - alpha1: 0.33424
  - beta1: 0.12796

### Tests for coefficient stability (structural change)

No evidence for unstable parameters.

### Asymptotic Critical Values (10% 5% 1%)

- Joint Statistic: 1.07 1.24 1.6
- Individual Statistic: 0.35 0.47 0.75
Summary of GARCH(1,1) Fit

<table>
<thead>
<tr>
<th>Sign Bias Test</th>
<th>t-value</th>
<th>prob</th>
<th>sig</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sign Bias</td>
<td>2.1124</td>
<td>0.03473</td>
<td>**</td>
</tr>
<tr>
<td>Negative Sign Bias</td>
<td>0.8984</td>
<td>0.36904</td>
<td></td>
</tr>
<tr>
<td>Positive Sign Bias</td>
<td>0.2570</td>
<td>0.79721</td>
<td></td>
</tr>
<tr>
<td>Joint Effect</td>
<td>5.2995</td>
<td>0.15114</td>
<td></td>
</tr>
</tbody>
</table>

Tests for leverage effects (discuss later)

Some evidence of asymmetric effects

Adjusted Pearson Goodness-of-Fit Test:

<table>
<thead>
<tr>
<th>group</th>
<th>statistic</th>
<th>p-value(g-1)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>125.1</td>
<td>1.233e-17</td>
</tr>
<tr>
<td>2</td>
<td>133.7</td>
<td>2.195e-15</td>
</tr>
<tr>
<td>3</td>
<td>156.9</td>
<td>4.420e-16</td>
</tr>
<tr>
<td>4</td>
<td>165.8</td>
<td>1.351e-14</td>
</tr>
</tbody>
</table>

Tests for Distribution goodness-of-fit
Normal distribution assumption is strongly rejected!

Elapsed time : 0.4252
Extractor Functions

# estimated coefficients
> coef(MSFT.garch11.fit)

mu     omega    alpha1     beta1

# unconditional mean in mean equation
> uncmean(MSFT.garch11.fit)

mu
0.0004893

# unconditional variance: omega/(alpha1 + beta1)
> uncvariance(MSFT.garch11.fit)

unconditional
 0.0004876

# persistence: alpha1 + beta1
> persistence(MSFT.garch11.fit)

persistence
 0.9904

# half-life: ln(0.5)/(ln(alpha1 + beta1))
> halflife(MSFT.garch11.fit)

Half-Life
 71.85

© Eric Zivot 2012
Conditional Volatility: $\sigma_t$

```r
> plot.ts(sigma(MSFT.garch11.fit), ylab="sigma(t)", col="blue")
```
> plot(MSFT.garch11.fit)

Make a plot selection (or 0 to exit):

1: Series with 2 Conditional SD Superimposed
2: Series with 2.5% VaR Limits (with unconditional mean)
3: Conditional SD
4: ACF of Observations
5: ACF of Squared Observations
6: ACF of Absolute Observations
7: Cross Correlation
8: Empirical Density of Standardized Residuals
9: QQ-Plot of Standardized Residuals
10: ACF of Standardized Residuals
11: ACF of Squared Standardized Residuals
12: News-Impact Curve

Selection:
Plot Method: `plot(x, which=1)`
Plot Method: `plot(x, which="all")`
Normality Assumption is Bad

> plot(MSFT.garch11.fit, which=9)

© Eric Zivot 2012
Convergence Problems

# Fit ARCH(1) to MSFT
> arch1.spec = ugarchspec(variance.model = list(garchOrder=c(1,0)),
+                           mean.model = list(armaOrder=c(0,0)))
> MSFT.arch1.fit = ugarchfit(spec=arch1.spec, data=MSFT.ret,
+                            solver.control=list(trace = 1))

Iter: 1 fn: -7659.3402 Pars: -0.0004496  0.0002867  0.4711622
solnp--> Solution not reliable....Problem Inverting Hessian.

Convergence problems could be due to some extreme observations in the data. Sometime “cleaning” the data of “outliers” can help with convergence
Cleaned Data

> MSFT.ret.clean = Return.clean(MSFT.ret, method="boudt")
ARCH(1) on Cleaned Data

```r
> MSFT.clean.arch1.fit = ugarchfit(spec=arch1.spec, data=MSFT.ret.clean,
+                                    solver.control=list(trace = 1))
```

Iter: 1 fn: -7845.5508  Pars:  -0.0001778  0.0002673  0.3586366
Iter: 2 fn: -7845.5508  Pars:  -0.0001777  0.0002673  0.3586415

solnp--> Completed in 2 iterations

```r
> MSFT.clean.arch1.fit
```

*---------------------------------------------*
*          GARCH Model Fit                    *
*---------------------------------------------*

Conditional Variance Dynamics

GARCH Model : sGARCH(1,0)
Mean Model : ARFIMA(0,0,0)
Distribution : norm

Optimal Parameters

| Parameter | Estimate | Std. Error | t value | Pr(>|t|) |
|-----------|----------|------------|---------|----------|
| mu        | -0.000178| 0.000309   | -0.57434| 0.56574  |
| omega     | 0.000267 | 0.000009   | 28.64710| 0.00000  |
| alpha1    | 0.358641 | 0.034496   | 10.39666| 0.00000  |

© Eric Zivot 2012
Model Selection

```r
> arch.order = 1:5
> arch.names = paste("arch", arch.order, sep="")

# fit all arch models with p <= 5
> arch.list = list()
> for (p in arch.order) {
+   arch.spec = ugarchspec(variance.model = list(garchOrder=c(p,0)),
+                           mean.model = list(armaOrder=c(0,0)))
+   arch.fit = ugarchfit(spec=arch.spec, data=MSFT.ret.clean,
+                       solver.control=list(trace = 0))
+   arch.list[[p]] = arch.fit
+ }

> names(arch.list) = arch.names

# Add GARCH(1,1) refit to cleaned data to list
> arch.list$garch11 = garch11.fit
```

© Eric Zivot 2012
Model Selection

# Compute information criteria using infocriteria() function
> info.mat = sapply(arch.list, infocriteria)
> rownames(info.mat) = rownames(infocriteria(arch.list[[1]]))
> info.mat

<table>
<thead>
<tr>
<th></th>
<th>arch1</th>
<th>arch2</th>
<th>arch3</th>
<th>arch4</th>
<th>arch5</th>
<th>garch11</th>
</tr>
</thead>
<tbody>
<tr>
<td>Akaike</td>
<td>-5.089</td>
<td>-5.140</td>
<td>-5.180</td>
<td>-5.218</td>
<td>-5.243</td>
<td>-5.319</td>
</tr>
<tr>
<td>Bayes</td>
<td>-5.083</td>
<td>-5.132</td>
<td>-5.170</td>
<td>-5.206</td>
<td>-5.230</td>
<td>-5.311</td>
</tr>
<tr>
<td>Shibata</td>
<td>-5.089</td>
<td>-5.140</td>
<td>-5.180</td>
<td>-5.218</td>
<td>-5.243</td>
<td>-5.319</td>
</tr>
<tr>
<td>Hannan-Quinn</td>
<td>-5.087</td>
<td>-5.137</td>
<td>-5.177</td>
<td>-5.213</td>
<td>-5.238</td>
<td>-5.316</td>
</tr>
</tbody>
</table>

GARCH(1,1) has the best fit – smallest values of info criteria
ARCH(5) vs. GARCH(1,1)
GARCH(1,1) Forecasts

# Compute h-step ahead forecasts for h=1,...,100
> MSFT.garch11.fcst = ugarchforecast(MSFT.garch11.fit, 
+ n.ahead=100)

> class(MSFT.garch11.fcst)
[1] "uGARCHforecast"
attr(,"package")
[1] "rugarch"

> slotNames(MSFT.garch11.fcst)
[1] "forecast" "model"

> names(MSFT.garch11.fcst@forecast)
[1] "n.ahead" "N" "n.start" "n.roll" "forecasts"
[6] "fdates"
GARCH(1,1) Forecasts

> MSFT.garch11.fcst

*------------------------------------*
* GARCH Model Forecast            *
*------------------------------------*
Model: sGARCH
Horizon: 100
Roll Steps: 0
Out of Sample: 0

0-roll forecast:

<table>
<thead>
<tr>
<th></th>
<th>sigma</th>
<th>series</th>
</tr>
</thead>
<tbody>
<tr>
<td>2012-04-04</td>
<td>0.01136</td>
<td>0.0003397</td>
</tr>
<tr>
<td>2012-04-05</td>
<td>0.01151</td>
<td>0.0003397</td>
</tr>
<tr>
<td>2012-04-06</td>
<td>0.01166</td>
<td>0.0003397</td>
</tr>
<tr>
<td>2012-04-09</td>
<td>0.01180</td>
<td>0.0003397</td>
</tr>
<tr>
<td>2012-04-10</td>
<td>0.01194</td>
<td>0.0003397</td>
</tr>
</tbody>
</table>

...
<table>
<thead>
<tr>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>as.array</code></td>
<td>Extracts the forecast array</td>
</tr>
<tr>
<td><code>as.data.frame</code></td>
<td>Extracts the forecasts</td>
</tr>
<tr>
<td><code>as.list</code></td>
<td>Extracts the forecast list will all rollframes</td>
</tr>
<tr>
<td><code>plot</code></td>
<td>Forecasts plots</td>
</tr>
<tr>
<td><code>fpm</code></td>
<td>Forecast performance measures</td>
</tr>
<tr>
<td><code>show</code></td>
<td>Forecast summary</td>
</tr>
</tbody>
</table>
GARCH(1,1) Forecasts

> plot(MSFT.garch11.fcst)

Make a plot selection (or 0 to exit):

1: Time Series Prediction (unconditional)
2: Time Series Prediction (rolling)
3: Conditional SD Prediction

Selection:
GARCH(1,1) Forecasts

h-step ahead return forecast

h-step ahead volatility forecast

© Eric Zivot 2012
# Extract forecasts into data.frame
> MSFT.fcst.df = as.data.frame(MSFT.garch11.fcst)
> head(MSFT.fcst.df)

<table>
<thead>
<tr>
<th>sigma</th>
<th>series</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.01136</td>
<td>0.0003397</td>
</tr>
<tr>
<td>0.01151</td>
<td>0.0003397</td>
</tr>
<tr>
<td>0.01166</td>
<td>0.0003397</td>
</tr>
<tr>
<td>0.01180</td>
<td>0.0003397</td>
</tr>
<tr>
<td>0.01194</td>
<td>0.0003397</td>
</tr>
<tr>
<td>0.01208</td>
<td>0.0003397</td>
</tr>
</tbody>
</table>

# h-day return variance forecast = sum of h-day ahead
# variance forecasts
> fcst.var.hDay = cumsum(MSFT.fcst.df$sigma^2)
> fcst.vol.hDay = sqrt(MSFT.fcst.var.hDay)
Forecasts of h-day Return Vol

GARCH(1,1) Forecast of h-day Return Vol

© Eric Zivot 2012
Conditional VaR Forecasts

# h step-ahead conditional normal GARCH(1,1) VaR
> VaR.95.garch11 = MSFT.fcst.df$series[1] +
> MSFT.fcst.df$sigma[1]*qnorm(0.05)
> VaR.95.garch11
[1] -0.01835

# compute 20-day vol forecast from fitted GARCH(1,1)
> sigma.20day = sqrt(sum(MSFT.fcst.df$sigma[1:20]^2))
> VaR.95.garch11.20day = 20*MSFT.fcst.df$series[1] +
> sigma.20day*qnorm(0.05)
> VaR.95.garch11.20day
[1] -0.08621

© Eric Zivot 2012