Lecture 9: Introduction to ARCH/GARCH
Statistics 509 – Winter 2022
Reference: Chapter 14 of Ruppert/Matteson

Brian Thelen
258 West Hall
bjthelen@umich.edu

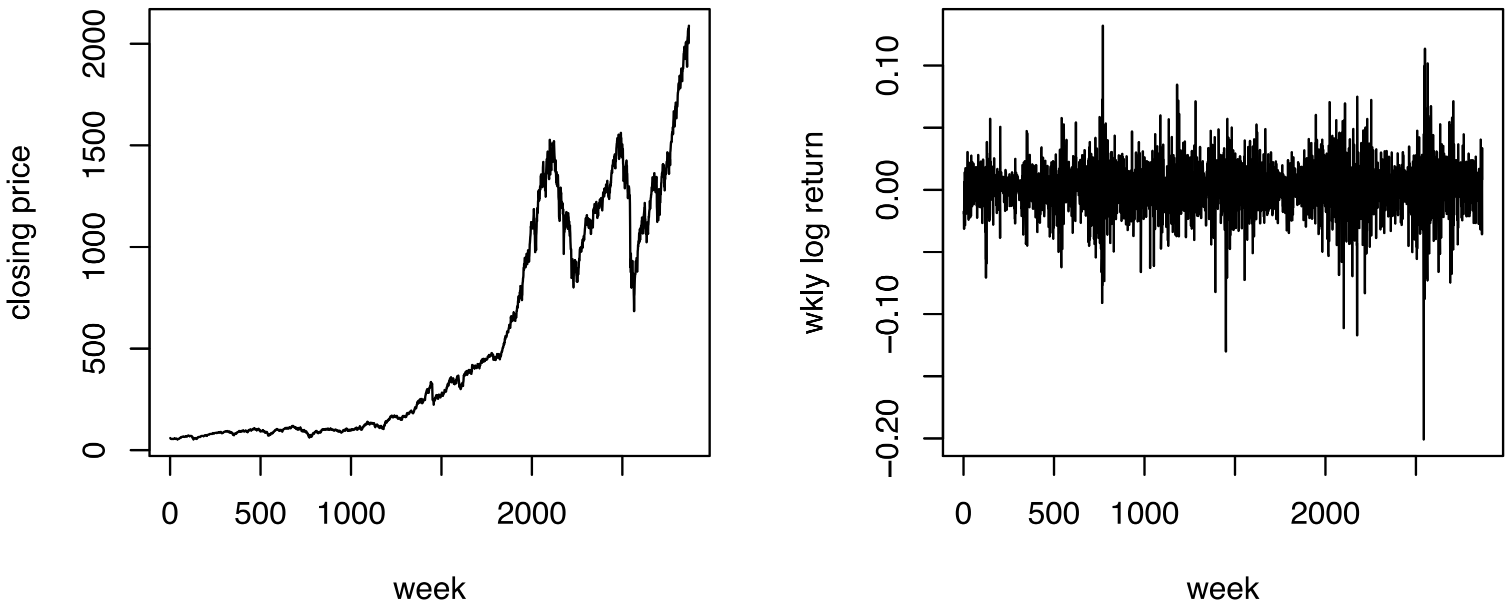
Overview

- Introduction
- ARCH Model
  - Auto-regressive conditional heteroskedasticity
- GARCH Model – Generalized ARCH
- Estimation Methods
- Examples

Motivation: ARCH

Background. In many financial data, the statistics actually are changing over time. None of the models presented previously can adapt to this. ARIMA model can't catch change in variation due to stationarity

Examples. SP500 data set over time.



Introduction: ARCH Model

Definition (ARCH): ARCH stands for **A** uto-**R** egressive **C** onditional **H** eteroskedasticity, and we say that $X = \{X_n\}$ corresponds to an ARCH(p) model if

$$X_n = \sigma_n \epsilon_n$$

是两个随机变量的乘积

where

- $\{\epsilon_n\}$ is an iid sequence of rvs with mean 0 and variance of 1 and
- σ_n is a function of $X_{n-1}, X_{n-2}, \ldots, X_{n-p}$ given by

volatility depends on the previous value

$$\sigma_n^2 = \alpha_o + \sum_{j=1}^{p} \alpha_j X_{n-j}^2$$

- where $\alpha_o > 0$ and $\alpha_j \geqslant 0$ for $j = 1, 2, \ldots, p$.

决定了过去数据的作用

Example. An ARCH(1) process assumes that $\alpha_1 > 0$ and the model is

$$X_n = \sqrt{\alpha_0 + \alpha_1 X_{n-1}^2} \quad \epsilon_n$$

Implications are

- α_0 is the floor variance (lower bound)
  - Variance of the $\{X_n\}$ process never goes below this
- $\sum_{j=1}^{p} \alpha_j X_{n-j}^2$ is the increase over the floor variance
  - depends on magnitude of X_{n-j}^2

ARCH Processes: Interpretation

Properties of ARCH Processes

- Self-exciting (increases in "current" variance tends to "stimulate" an increase in future variance)
- Variance depends on local (recent history) history

Recall-Definition. We say that a noise process $W = \{W_n\}$ is a **weak white noise process** if it has mean 0, constant variance of σ_W^2, and

不需要i.i.d只要cov是0

$$\text{Cov}(W_m, W_n) = 0 \qquad m \neq n.$$

ARMA只要是weak white noise

Remark. The model for $\{\epsilon_n\}$ in the AR/MA/ARMA and ARIMA models has been the **weak white noise process** model.

ARCH(1) Model

Example. Assume an ARCH(1) model – then

$$X_n^2 = \sigma_n^2 \epsilon_n^2 = \underbrace{(\alpha_o + \alpha_1 X_{n-1}^2)}_{\sigma_n^2} \epsilon_n^2$$

- Looks like AR process – but multiplicative

Note we can rewrite this model as

$$\begin{aligned} X_n^2 &= (\alpha_o + \alpha_1 X_{n-1}^2)\epsilon_n^2 \\ &= (\alpha_o + \alpha_1 X_{n-1}^2)\left(1 + \epsilon_n^2 - 1\right) \\ &= \alpha_o + \alpha_1 X_{n-1}^2 + \underline{\sigma_n^2(\epsilon_n^2 - 1)} \\ &= \alpha_o + \alpha_1 X_{n-1}^2 + V_n \end{aligned}$$

$E[\epsilon_n^2] = VAR(\epsilon_n)^2 = 1$

$E[V_n] = 0$

where

$$V_n = \sigma_n^2(\epsilon_n^2 - 1).$$

If V_n were weak white noise process , the above would suggests that X_n^2 follows an AR(1) process.

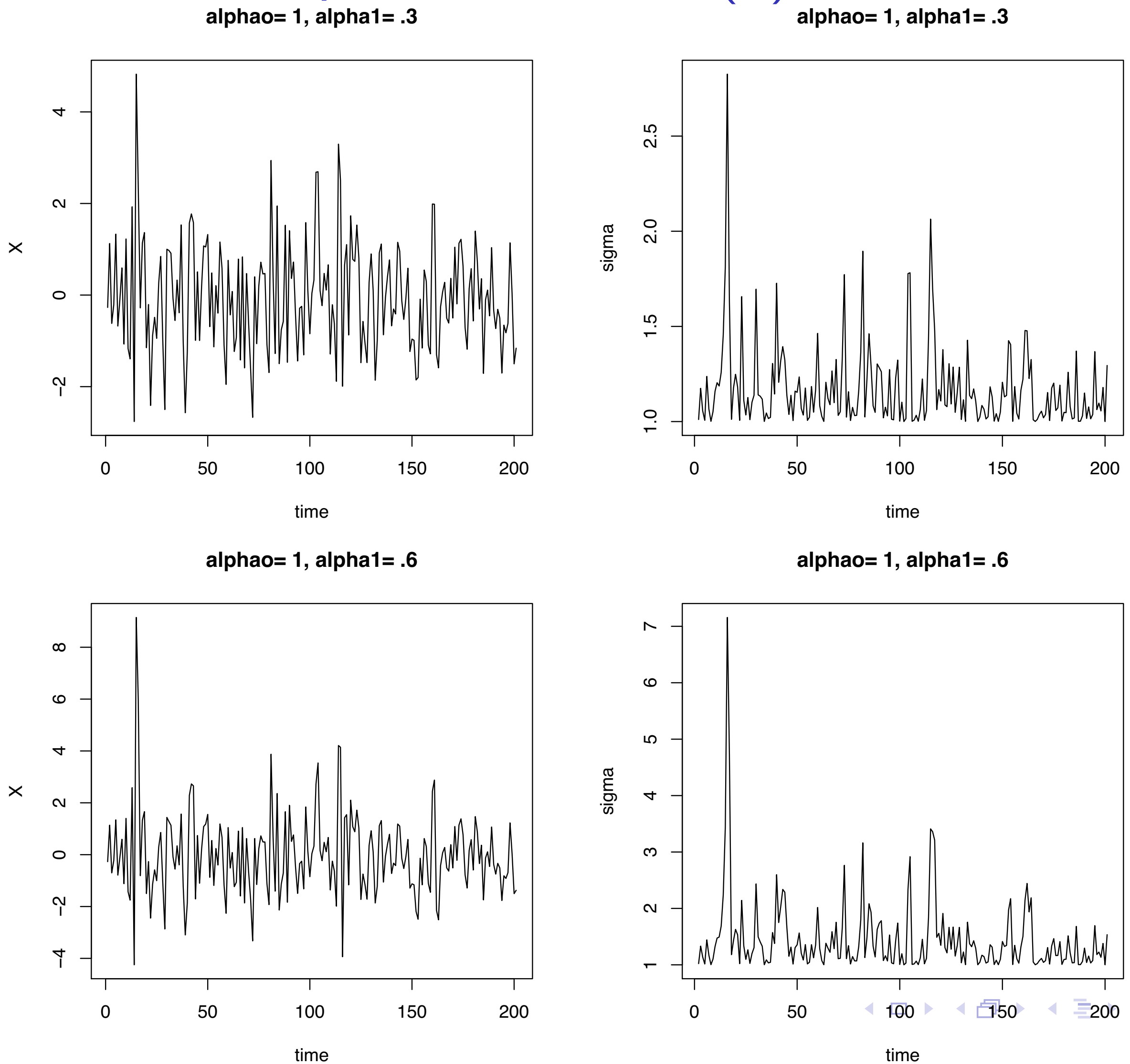
ARCH(p) Model

Remark. The generalization (relative to previous slide) for the case of ARCH(p) is that

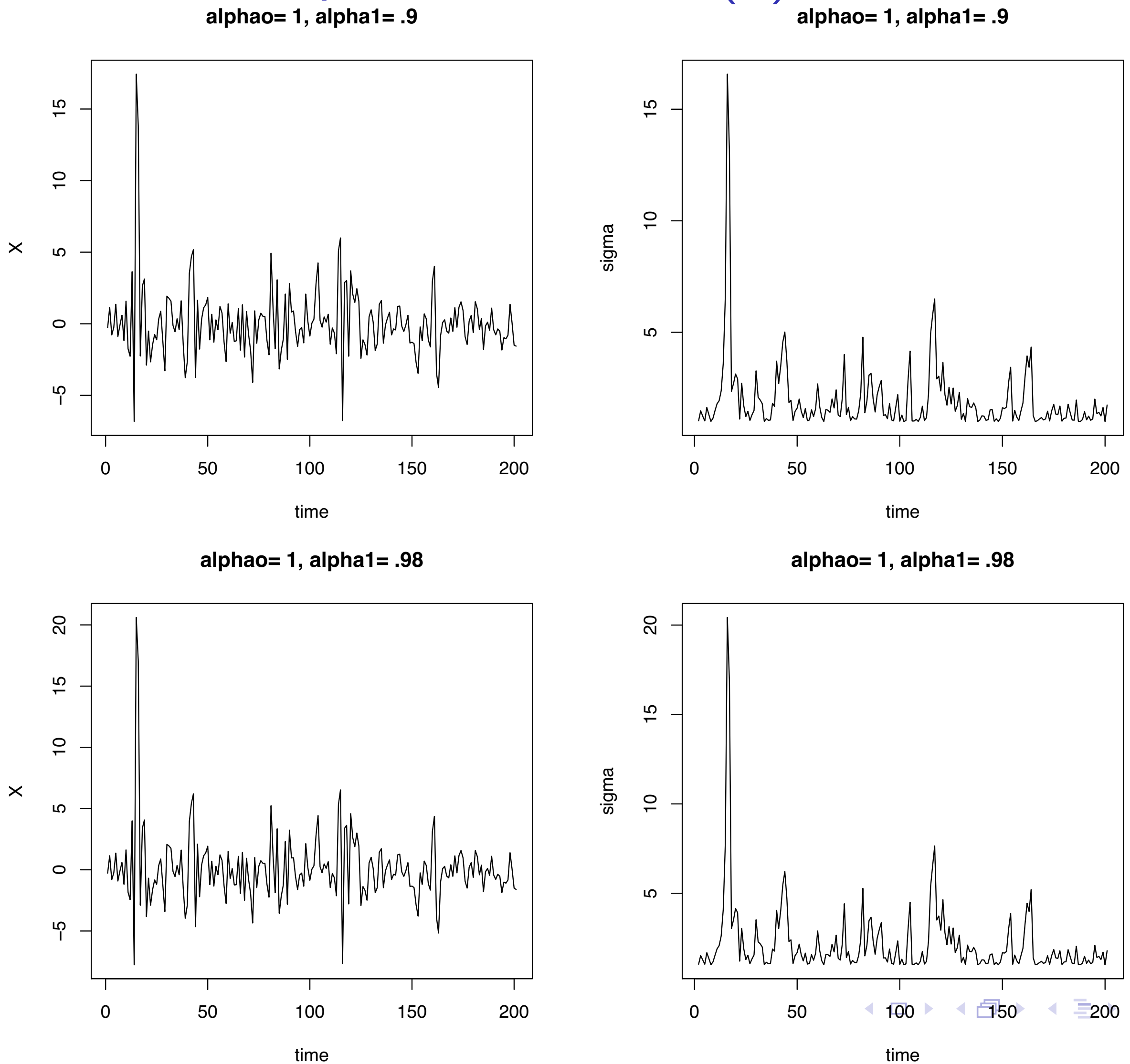
$$X_n^2 = \alpha_o + \sum_{j=1}^{p} \alpha_j X_{n-j}^2 + V_n$$

Interpretation.

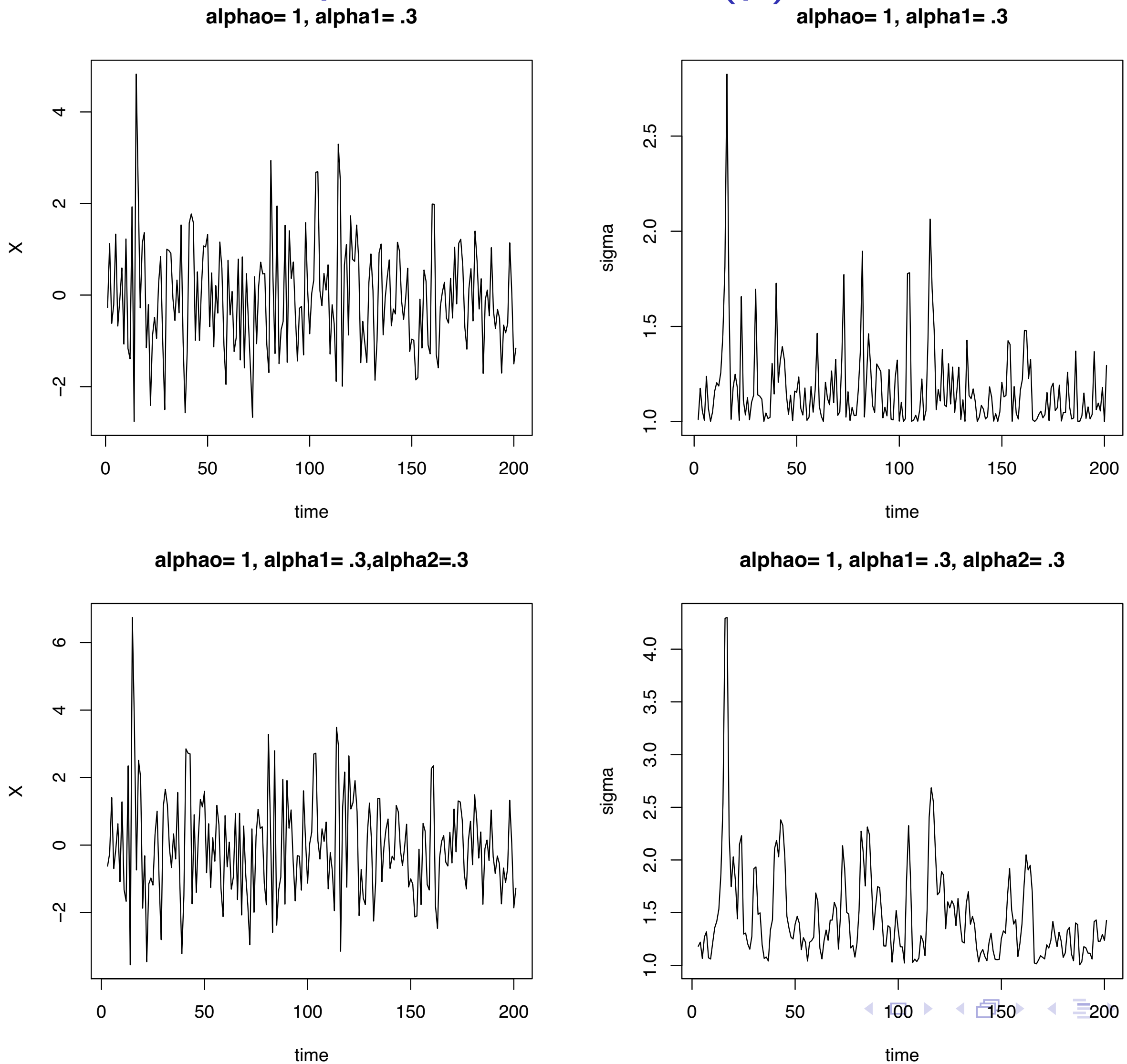
Simulation Examples 1-4: ARCH(1)



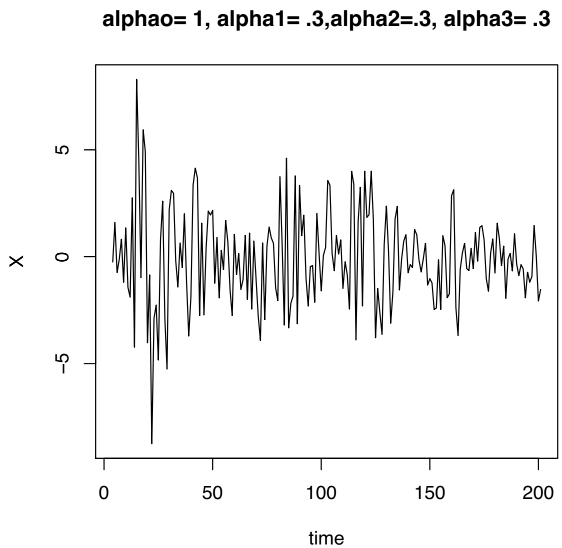
Simulation Examples 5-8: ARCH(1)

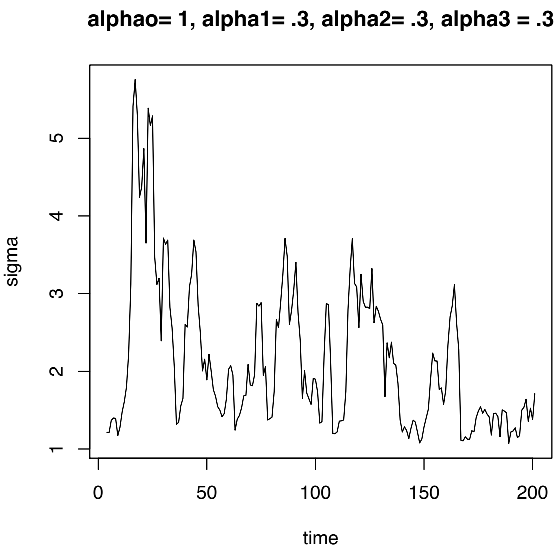


Simulation Examples 1-4: ARCH(p)



Simulation Examples 5-6: ARCH(p)





Theoretical Properties: ARCH

Example - ARCH(1) Suppose that X is an ARCH(1) process

$$X_n = \sigma_n \epsilon_n = \sqrt{\alpha_o + \alpha_1 X_{n-1}^2}\epsilon_n$$

with parameter α_o and $|\alpha_1| < 1$.

(a) Show that $Y_n = X_n^2$ follows a first order AR process with a noise process V_n which is not $\{\epsilon_n\}$. $V_n = \sigma_n^2(\epsilon_n^2 - 1)$

(b) Via induction derive a representation for X_n^2 involving only the white noise process up to time n.

(c) Utilizing **(b)** , derive a representation for σ_n^2 which involves only the white noise process up to time $n-1$.

(d) Assuming that the variance of X_n is finite for all n, and using the representation in **(b)** , show that

$$E(X_n X_{n+m}) = \begin{cases} \frac{\alpha_o}{1-\alpha_1} & m = n \\ 0 & n \neq m \end{cases}$$

arch is weak white noise

(e) Show that $E(X_n | X_{n-1}, X_{n-2}, \ldots) = 0$ for all n and that this implies that $E(X_n) = 0$ for all n.

(f) Based on **(c)** , give an argument for why ϵ_n and $\sigma_{n'}$ are independent for $n' \leq n$.

(g) Show that the process V_n in **(a)** is weakly stationary second-order white noise process with mean 0, assuming that ϵ_n has a finite fourth moment, i.e., $E(\epsilon_n^4) < \infty$ and $\alpha_1^2 < \frac{1}{E(\epsilon_n^4)}$.

Note: For normal innovations, this would imply that $|\alpha_1| \leq \frac{1}{\sqrt{3}}$.

取决于分布,通常sigma是very heavy tail

Answer.

(b) $X_n^2 = \sigma_n^2 \epsilon_n^2$

$$= (\alpha_0 + \alpha_1 X_{n-1}^2)\,\epsilon_n^2$$

$$= \alpha_0 \epsilon_n^2 + \alpha_1 X_{n-1}^2 \epsilon_n^2$$

$$= \alpha_0 \epsilon_n^2 + \alpha_1 (\alpha_0 + \alpha_1 X_{n-2}^2)\,\epsilon_{n-1}^2 \epsilon_n^2$$

$$= \alpha_0 \sum_{j=0}^{\infty} \alpha_1^j \epsilon_n^2 \epsilon_{n-1}^2 \cdots \epsilon_{n-j}^2$$

(c) $\sigma_n^2 = \alpha_0 \sum_{j=0}^{\infty} \alpha_1^j \epsilon_{n-j}^2 = \alpha_0 + \alpha_0 \sum_{j=1}^{\infty} \alpha_1^j \epsilon_{n-1}^2 \cdots \epsilon_{n-j}^2$

(d) $E[X_n^2] = E\left[\alpha_0 \sum_{j=0}^{\infty} \alpha_1^j \epsilon_n^2 \epsilon_{n-j}^2\right] = \alpha_0 \sum_{j=0}^{\infty} \alpha_1^j E[\underbrace{\epsilon_n^2}_{1} \underbrace{\epsilon_{n-j}^2}_{1}] = \frac{\alpha_0}{1-\alpha_1^j}$

(e) $E[X_n \mid X_{n-1}, X_{n-2} \cdots] = E[\sigma_n \epsilon_n \mid X_{n-1} \cdots X_{n-k}] = 0$

Answer.

(f) $n' < n$, σ_n^2 depends on $\epsilon_{n'-1}, \epsilon_{n'-2} \cdots$

But ϵ_n is independent of those $n' \leq n$

(g) $V_n = \sigma_n^2 (\epsilon_n^2 - 1)$

$\Rightarrow E(V_n) = E(\sigma_n^2) E(\epsilon_n^2 - 1) = 0$

$\mathrm{Var}(V_n) = E(V_n^2) = E(\sigma_n^4) E[(\epsilon_n^2 - 1)^2]$

Answer.

Answer.

Estimation for ARCH: Overview

Motivation

- Want to carry out estimation of the parameters based on real data
- Natural to want to do something approaching MLE
- Turns out that "true MLE" is not easy to carry out
- An alternative known as conditional ML is relatively straightforward

使用national ML替代true MLE来对参数进行估计

Conditional MLE for ARCH

Objective Derive conditional MLE for ARCH(1) – details for ARCH(p) ($p \geq 2$) is similar.

Recall. Suppose that $X_1, X_2, X_3, \ldots$ followed an AR(1) process with parameters α and σ^2, then assuming that $\{\epsilon_n\}$ are iid $\mathcal{N}(0, \sigma^2)$, the conditional log-likelihood function of $X_2, X_3, \ldots, X_n$ given $X_1 = x_1$ can be derived by noting that

$$(\underbrace{X_2 - \alpha X_1}_{\epsilon_2}, X_3 - \alpha X_2, \ldots, \underbrace{X_n - \alpha X_{n-1}}_{\epsilon_n}) \text{ are iid } N(0, \sigma^2)$$

so that

累乘在一起，指数上就是相加

$$\begin{aligned}
\mathcal{L}(\alpha, \sigma^2 | x_1) &= \log\left(f(x_2, x_3, \ldots, x_n | x_1)\right) \\
&= \log\left\{\prod_{i=2}^{n} \frac{1}{\sqrt{2\pi\sigma}} e^{-\frac{1}{2}\left(\frac{X_i - \alpha X_{i-1}}{\sigma}\right)^2}\right\} \\
&= -\frac{n-1}{2}\log(2\pi\sigma^2) - \frac{1}{2}\sum_{i=2}^{n}\left(\frac{X_i - \alpha X_{i-1}}{\sigma}\right)^2
\end{aligned}$$

Conditional MLE for ARCH - cont'd

$E(X_n)=0 \quad Var(X_n)=E(X_n^2)=E[(\alpha_0+\alpha_1 X_{n-1}^2)\epsilon_n^2]=\alpha_0+\alpha_1 X_{n-1}^2$

Note for the ARCH(1) model and again assuming that $\{\epsilon_n\}$ are iid $\mathcal{N}(0,\sigma^2)$,

$$X_n = \underline{\sqrt{\alpha_o + \alpha_1 X_{n-1}^2}}\,\epsilon_n$$

因为是条件分布所以还是正态分布

so that the conditional distribution of X_n given $X_{n-1} = x_{n-1}$ is $\mathcal{N}(0, \alpha_0+\alpha_1 X_{n-1}^2)$, and so

$$\begin{aligned} f(x_n|x_1,x_2,\ldots,x_{n-1}) &= f(x_n|x_{n-1}) \\ &= \frac{1}{\sqrt{2\pi(\alpha_0+\alpha_1 X_{n-1}^2)}} e^{-\frac{X_n}{2(\alpha_0+\alpha_1 X_{n-1}^2)}} \end{aligned}$$

只由上一个数决定

Conditional MLE for ARCH - cont'd

Recalling the telescoping rule for pdfs, we have that

$$
\begin{aligned}
& f(x_2, x_3, \ldots, x_n | x_1) \\
&= f(x_2|x_1) \cdot f(x_3|x_1, x_2) \cdots f(x_n|x_1, x_2, \ldots, x_{n-1}) \\
&= f(x_2|x_1) f(x_3|x_2) \cdots f(x_n|x_{n-1})
\end{aligned}
$$

Substituting in the conditional pdfs and taking logs, we have that the conditional ML is given by

$$\mathcal{L}(\alpha_o, \alpha_1 | x_1) = \boxed{-\frac{1}{2}\sum_{i=2}^{n} \log\left[2\pi\left(\alpha_o + \alpha_1 x_{i-1}^2\right)\right]} - \boxed{\frac{1}{2}\sum_{i=2}^{n} \frac{x_i^2}{\alpha_o + \alpha_1 x_{i-1}^2}}$$

The conditional ML is derived as

$$(\hat{\alpha}_o, \hat{\alpha}_1) = \operatorname*{arg\,max}_{\alpha_o, \alpha_1} \mathcal{L}(\alpha_o, \alpha_1 | x_1)$$

R-Functions for ARCH Estimation

Instructions. Need to install "tseries" package and then do the usual command of `library(tseries)`. The estimation function is

```
garch(x, order = c(1, 1), coef = NULL, itmax = 200, eps = NULL,
      grad = c("analytical","numerical"), series = NULL,
      trace = TRUE, ...)

Arguments
x -- a numeric vector or time series.

order -- a two dimensional integer vector giving the orders of the
model to fit. order[2] corresponds to the ARCH part and order[1] to
the GARCH part.

coef -- If given this numeric vector is used as the initial estimate of
the GARCH coefficients. Default initialization is to set the GARCH
parameters to slightly positive values and to initialize the intercept
such that the unconditional variance of the initial GARCH is equal to
the variance of x.

itmax -- gives the maximum number of log-likelihood function evaluations
itmax and the maximum number of iterations 2*itmax the optimizer is
allowed to compute.
```

```
eps -- defines the absolute (max(1e-20, eps^2)) and relative function
convergence tolerance (max(1e-10, eps^(2/3))), the coefficient-
convergence tolerance (sqrt(eps)), and the false convergence tolerance
(1e2*eps). Default value is the machine epsilon, see Machine.

grad -- indicates if the analytical gradient or a numerical approximation
is used for the optimization.

Details
garch uses a Quasi-Newton optimizer to find the maximum likelihood
estimates of the conditionally normal model. The first max(p, q) values
are assumed to be fixed. The optimizer uses a hessian approximation
computed from the BFGS update. Only a Cholesky factor of the Hessian
approximation is stored.  For more details see Dennis et al. (1981),
Dennis and Mei (1979), Dennis and More (1977), and Goldfarb (1976).
The gradient is either computed analytically or using a numerical
approximation.
```

Values (out)
A list of class "garch" with the following elements:

order -- the order of the fitted model.

coef -- estimated GARCH coefficients for the fitted model.

n.likeli -- the negative log-likelihood function evaluated at the coefficient estimates (apart from some constant).

n.used -- the number of observations of x.

residuals -- the series of residuals. $\hat{\varepsilon}_n$

$\hat{\sigma}$

fitted.values -- the bivariate series of conditional standard deviation predictions for x.

series -- the name of the series x.

frequency -- the frequency of the series x.

call -- the call of the garch function.

asy.se.coef -- the asymptotic-theory standard errors of the coefficient estimates.

ARCH: More on Statistical Tests

Estimated Residuals Note that with ARCH model, have a recursive scheme for estimating σ_n and ϵ_n. For ARCH(p), it follows the simple form:

$$\hat{\sigma}_n^2 = \hat{\alpha}_0 + \sum_{j=1}^{\infty} \hat{\alpha}_j X_{n-j}^2$$

$$\hat{\epsilon}_n = \frac{X_n}{\hat{\sigma}_n}$$

Test of Normality: Jarque-Berra Test

$$\boxed{JB = \frac{n}{6}\left(S^2 + \frac{(K-3)^2}{4}\right)}$$

where

- S is empirical skewness parameter (of est residuals)
- K is empirical kurtosis parameter (of est residuals)
- Under the null distribution (residuals are normally distributed), the approximate distribution of JB is approximately chi-square with 2 degrees of freedom

Reject Normality (of errors) if JB is large

$JB > \chi^2_{2,1-\alpha} = 1-\alpha$ quantile of χ^2_2

Test of Zero auto-correlation: Box-Ljung

- Summary of GARCH puts out the p-value of test of the auto-correlation of the squared residuals.
- Only being carried out for lag $h = 1$ (degree of freedom $= 1$)

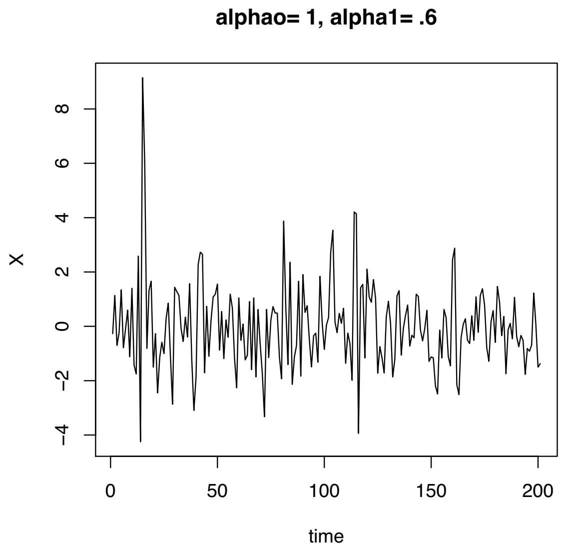
Question: Why is the Box-Ljung test done on the squared residuals as opposed to the residuals?

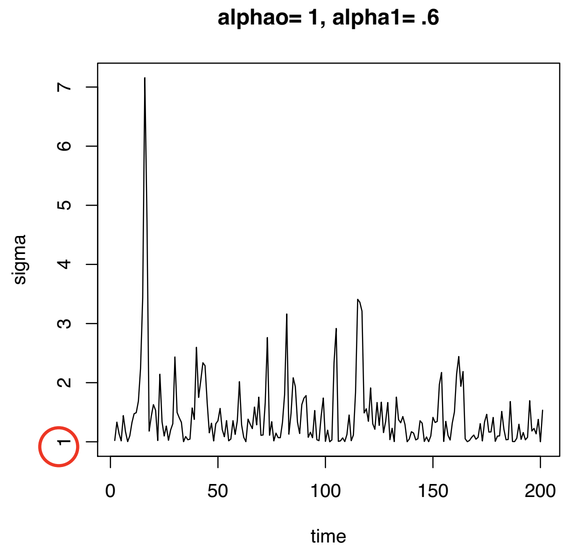
Simulation Example: ARCH Estimation

Background: Simulated an ARCH(1) process with parameters of

$$\alpha_o = 1, \quad \alpha_1 = .6$$

and did $n = 201$. The data looks like





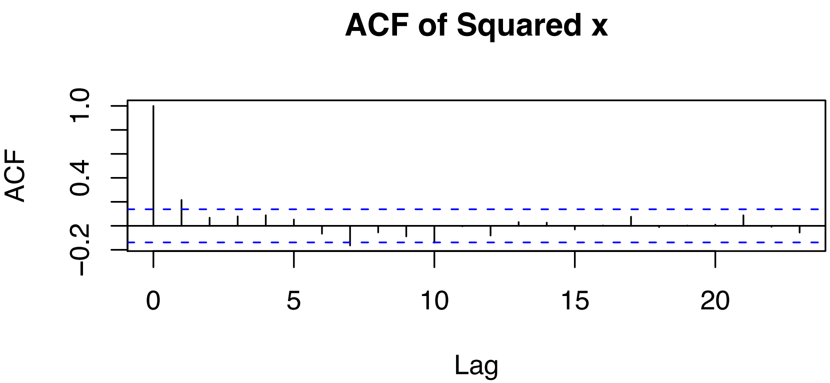
R-commands for Simulation

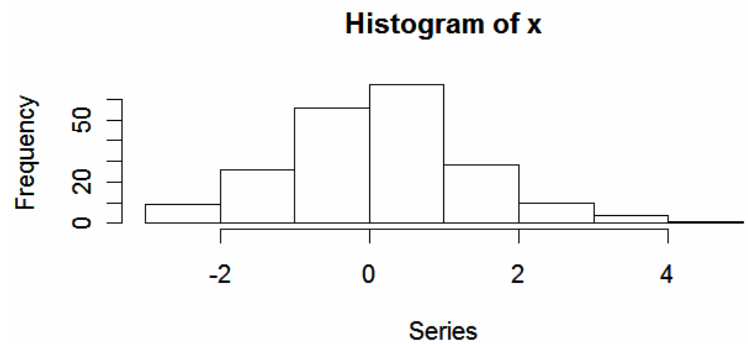
Carried out estimation via

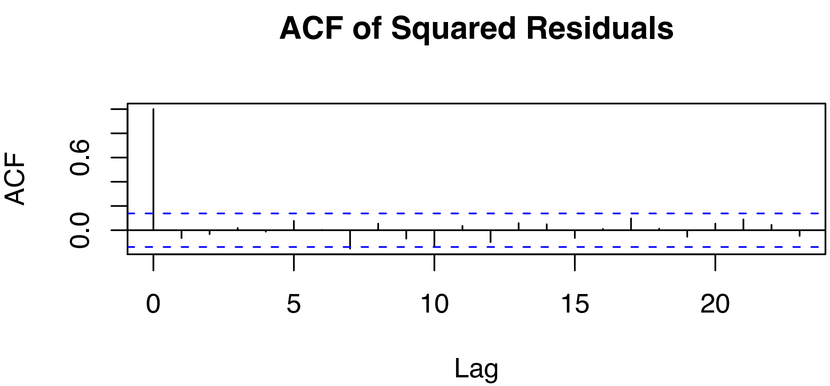
```
> n = 201
> w = rnorm(n,0,1)
> x = 0*w
> x[1] = w[1]
> sigma = 0*w
> alphao = 1
> alpha1 = .6
> for (k in c(2:n)){
+   sigma[k] = (alphao + alpha1*(x[k-1]^2))^.5
+   x[k] = sigma[k]*w[k]
+ }
>
> garchest <- garch(x, order = c(0, 1), coef = NULL, itmax = 200,
+ eps = NULL, grad = c("analytic"))
```

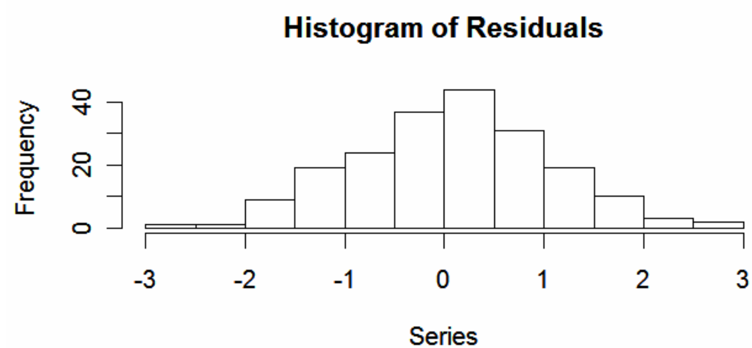
```
 ***** ESTIMATION WITH ANALYTICAL GRADIENT *****
> summary(garchest)

Residuals:
     Min       1Q   Median       3Q      Max
-2.77706 -0.57450  0.06261  0.70491  2.70170

Coefficient(s):
    Estimate  Std. Error  t value Pr(>|t|)
a0    0.9740      0.1625    5.995 2.03e-09 ***
a1    0.4012      0.1495    2.684  0.00727 **
---

Diagnostic Tests:
        Jarque Bera Test
data:  Residuals
X-squared = 0.1106, df = 2, p-value = 0.9462

        Box-Ljung test
data:  Squared.Residuals
X-squared = 0.8631, df = 1, p-value = 0.3529

> windows()
> plot(garchest)
```

Diagnostic Plots from Simulation Example

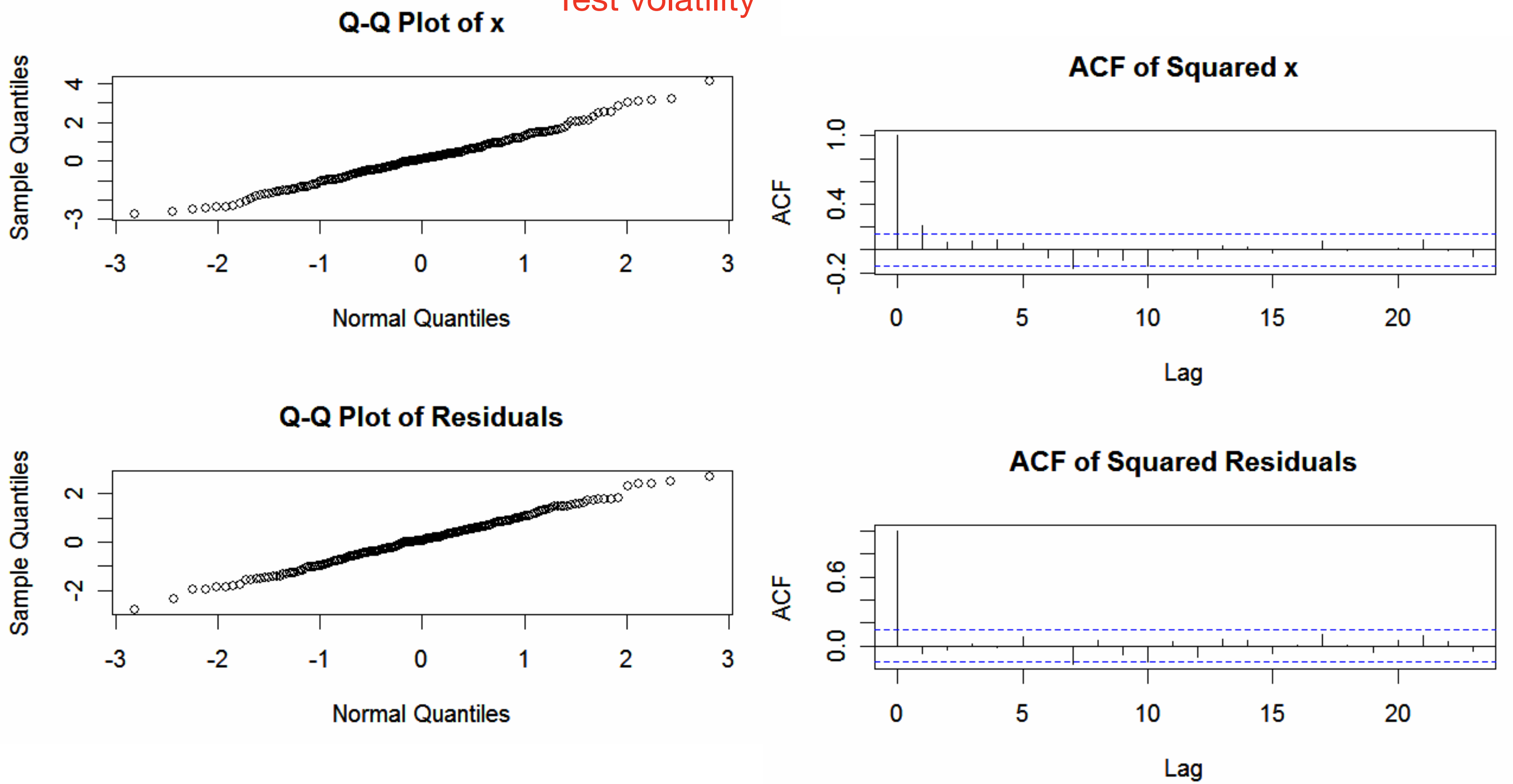








Test volatility



Interpretation

GARCH: Motivation

Remark. It has been obverved in a number of applications that

- behavior of ARCH processes does not match the financial time series
- One main discrepany has to do with "persistence" of *outliers regions/values*. 可以处理连续的heavy tail

One remedy is the **Generalized ARCH (GARCH)** model.

Definition (GARCH) We say a process $X = \{X_n\}$ is GARCH(p,q) if

$$X_n = \mu_n + \sigma_n \epsilon_n$$

where $\{\epsilon_n\}$ is an iid white noise process with mean 0 and variance of 1, and the model for σ_n is

$$\sigma_n^2 = \alpha_o + \sum_{i=1}^{p} \beta_i \sigma_{n-i}^2 + \sum_{j=1}^{q} \alpha_j \tilde{X}_{n-j}^2$$

with

$$\tilde{X}_n = X_n - \mu_n$$

GARCH Models

Remark. A widely used model is GARCH(1,1) –

$$X_n = \mu_n + \sigma_n \epsilon_n$$

where

$$\sigma_n^2 = \alpha_o + \beta_1 \sigma_{n-1}^2 + \alpha_1 \tilde{X}_n^2$$

Interpretation/Comparison with ARCH

GARCH: Some Properties

Example. Show that if $X = \{X_n\}$ is a GARCH(1,1) process then $Y_n = \{X_n^2\}$ is an ARMA(1,1) with a weak white-noise process $V_n = \sigma_n^2(\epsilon_n^2 - 1)$ (i.e., mean 0, constant variance, and uncorrelated).

Hint. We showed that X_n being an ARCH(1) process implied that X_n^2 is an AR(1) process with the same weak white-noise process V_n.

Answer.

$$X_n^2 = \sigma_n^2 \epsilon_n^2$$
$$= \sigma_n^2 + \sigma_n^2(\epsilon_n^2 - 1)$$
$$= \alpha_0 + \beta_1 \sigma_{n-1}^2 + \alpha X_{n-1}^2 + V_n$$
$$= \alpha_0 + (\alpha+\beta_1) X_{n-1}^2 - \beta_1 (X_{n-1}^2 - \sigma_{n-1}^2) + V_n$$

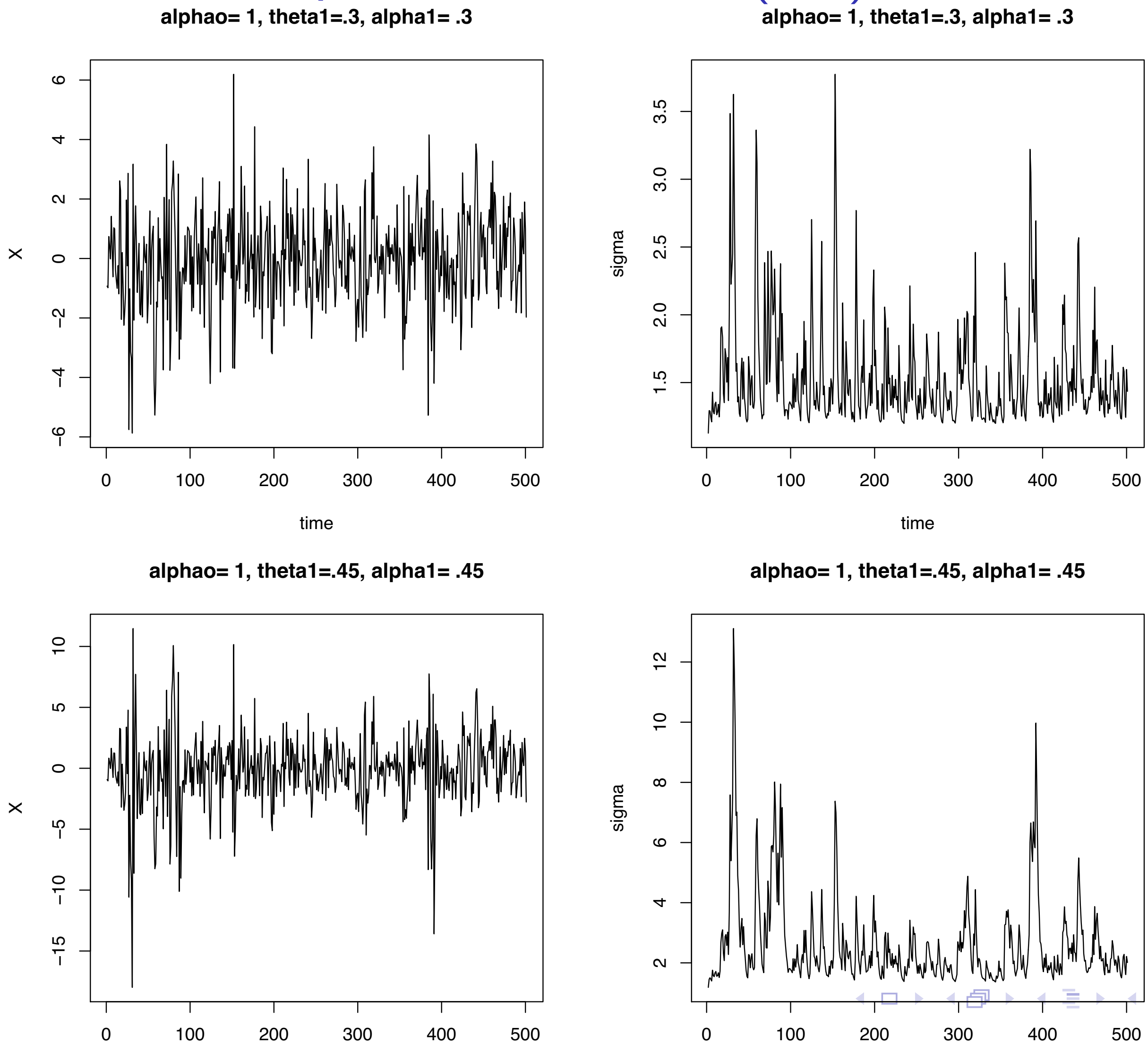
$$X_n^2 - (\alpha+\beta_1) X_{n-1}^2 = \alpha_0 + V_n - \beta_1 V_{n-1}$$

ARMA模型需要进行单位根检验

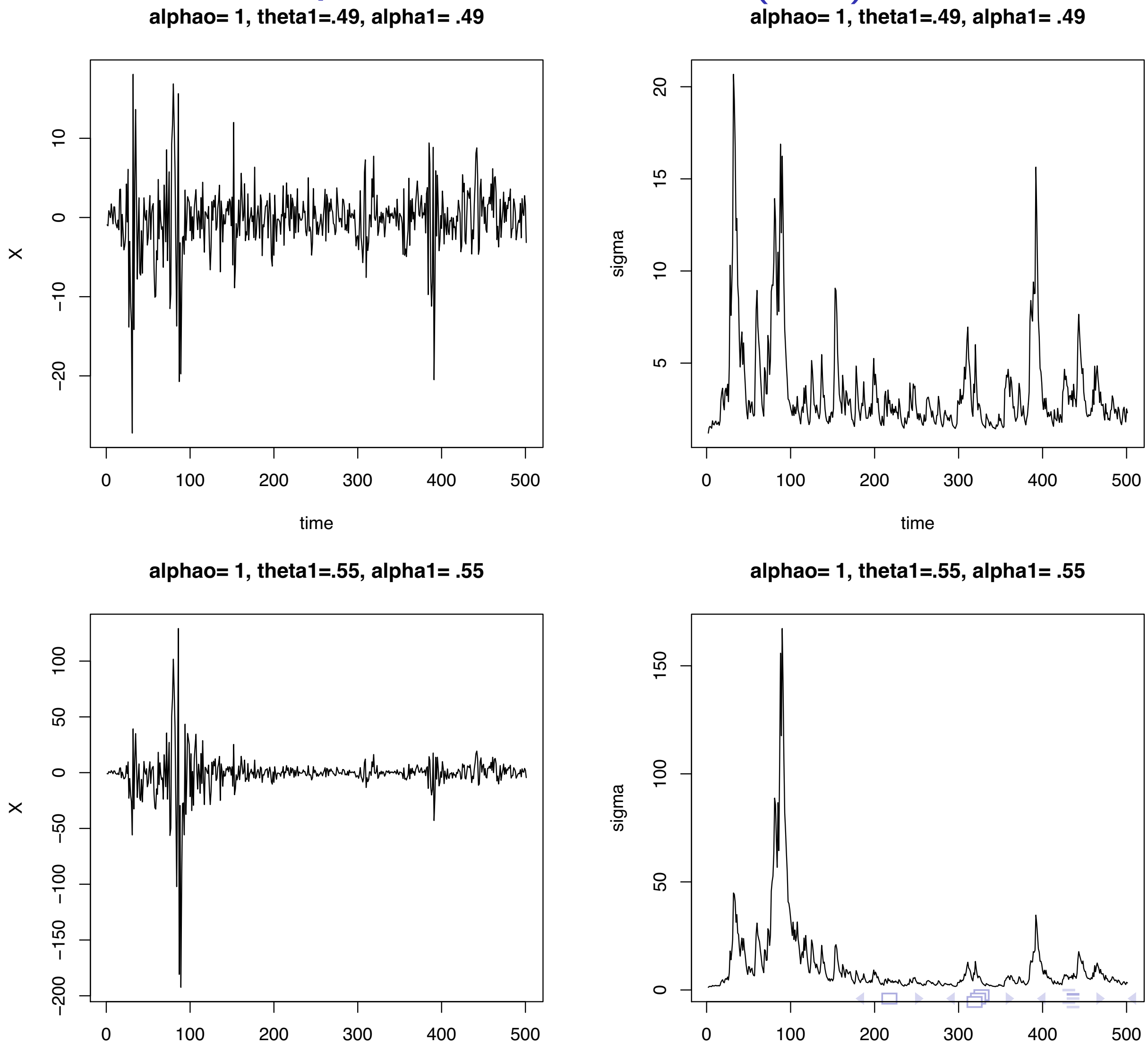
$$\alpha + \beta_1 < 1$$

Answer.

Simulation Examples 1-4: GARCH(1,1)



Simulation Examples 5-8: GARCH(1,1)



GARCH Estimation

Remark. Estimation for GARCH is again based on a conditional MLE approach

- Assumes distribution for the innovations ϵ_n
  - Default is Normal distribution
  - In other packages, a common alternative is t-distribution with degrees of freedom
    - Alternative for heavier tails
    - Degrees of freedom can be estimated as well

- Numerical optimization is done – must specify technique and parameters of search

Remark. GARCH(1,1) is most popular model

- Not common go to higher order in practice

order增加了会unstable

GARCH: More on Statistical Tests

Estimated Residuals Note that with GARCH model, have a recursive scheme for estimating σ_n and ϵ_n. For mean-zero GARCH(1,1) process, it follows the following form:

$$\hat{\sigma}_0^2 = \alpha_0, \quad X_0 = 0$$

$$\hat{\sigma}_1^2 = \alpha_0 + \beta_1 \sigma_0^2 + \alpha_1 X_0^2 \qquad \hat{\epsilon}_1 = \frac{X_1}{\hat{\sigma}_1}$$

$$\hat{\sigma}_n^2 = \alpha_0 + \beta_1 \sigma_{n-1}^2 + \alpha_1 X_{n-1}^2$$

$$\hat{\epsilon}_n = \frac{X_n}{\hat{\sigma}_n}$$

GARCH Estimation: Simulation Example

Background Time series $\{X_n\}_1^{650}$ and coefficients of

$$\alpha_0 = 1, \ \ \alpha_1 = .3, \quad \beta_1 = .3$$

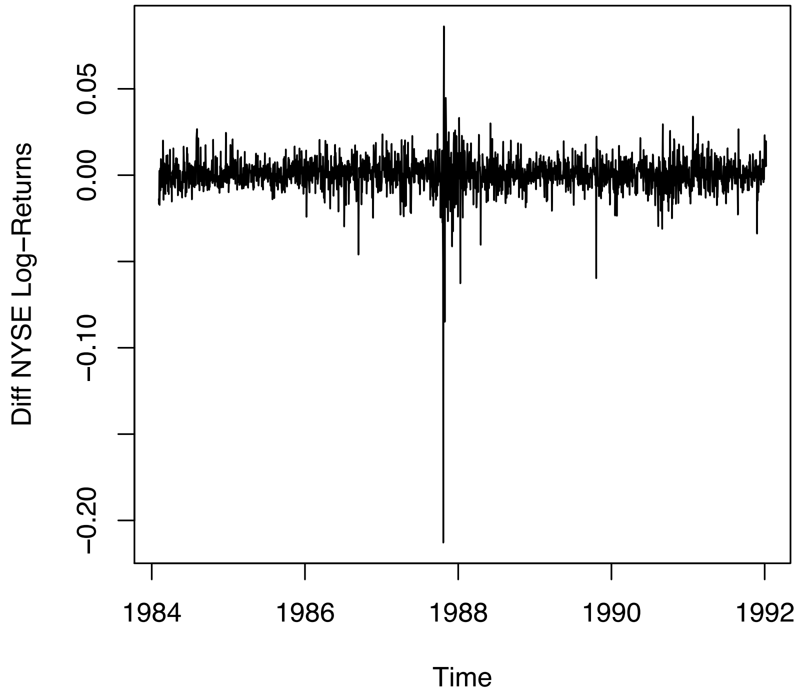
```
> n = 701
> w = rnorm(n,0,1)
> x = 0*w
> x[1] = w[1]
> sigma = 0*w+.2
>
> alphao = 1
> alpha1 = .3
> beta1 = .3
> for (k in c(2:n)){
+   sigma[k] = (alphao + beta1*sigma[k-1]^2+alpha1*(x[k-1]^2))^.5
+   x[k] = sigma[k]*w[k]
+ }
> x1 = x[c(50:n)] # allow the ‘‘burn-in’’ process
>
> garchest11 <- garch(x1, order = c(1, 1), coef = c(1.5,.2,.2),
itmax = 200, eps = NULL, grad = c("analytic"))
```

```
 ***** ESTIMATION WITH ANALYTICAL GRADIENT *****

> summary(garchest11)
Model:
GARCH(1,1)

Residuals:
     Min       1Q   Median       3Q      Max
-3.16705 -0.64892  0.05336  0.70817  3.03722

Coefficient(s):
    Estimate  Std. Error  t value Pr(>|t|)
a0    1.1976      0.3206    3.735 0.000187 ***
a1    0.2514      0.0693    3.628 0.000285 ***
b1    0.2825      0.1440    1.961 0.049858 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Diagnostic Tests:
        Jarque Bera Test
data:  Residuals
X-squared = 0.4775, df = 2, p-value = 0.7876

        Box-Ljung test
data:  Squared.Residuals
X-squared = 0.0103, df = 1, p-value = 0.9191
```

both p-value means we succeed

GARCH Time Series Analysis

NYSE Example. Have data corresponding to the NYSE from Feb 1984 to Dec of 1991 (went through crash of Oct 19 1987).



Approach

- Traditional ARIMA Analysis
- GARCH Analysis

R-Code: GARCH

R-Code: Installing the TSA package gives

- `garch.sim` function
- slightly different (expanded) `arima` function

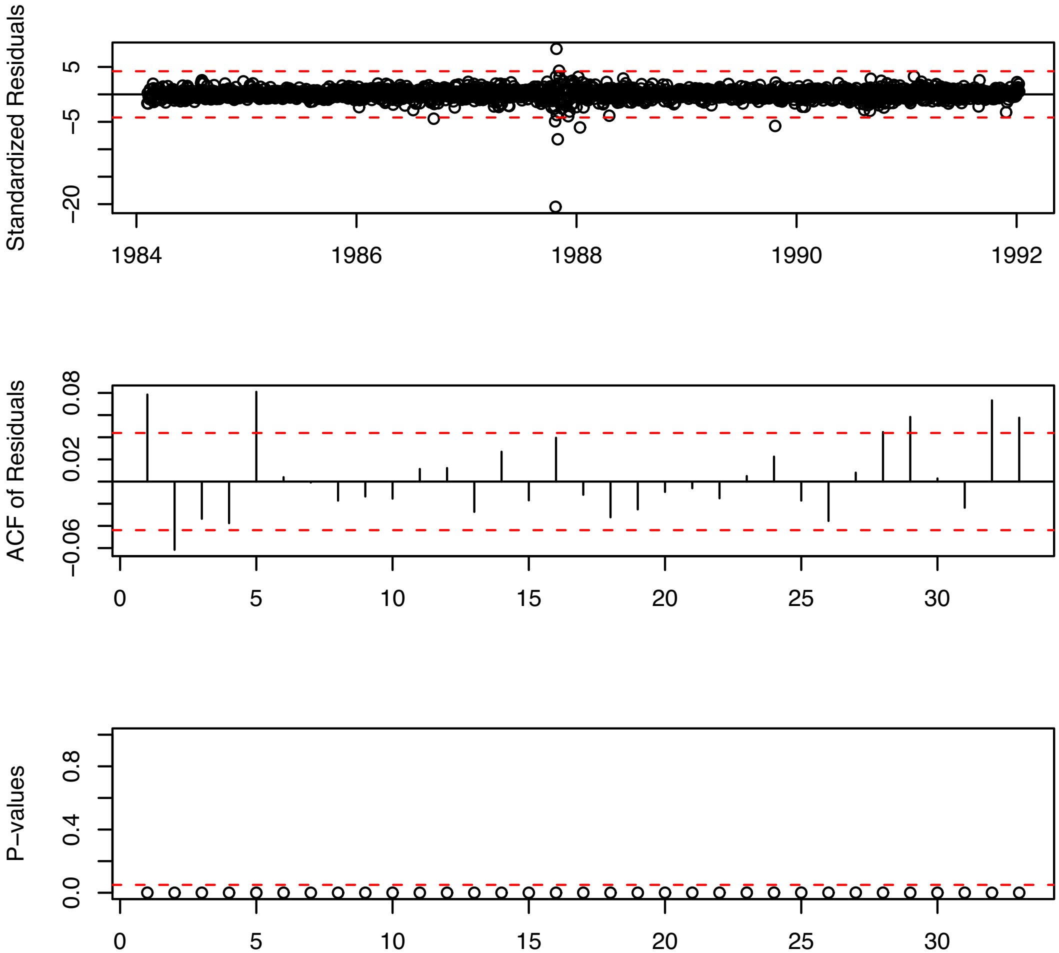
Examples on the next few slides utilized this arima function

NYSE Data - ARIMA(0,1,0)

R-Code:

```
> X = read.csv("Data\\NYSE-Feb84-Dec91.csv",header=TRUE)
> NYSE_lret <- log(X$AdjClose)
> NYSE_difflret <- diff(log(X$AdjClose))
> NYSE_lret.ts <- ts(data=NYSE_lret,start=c(1984,24),frequency=252,
+ names=c("NYSE_LRet"))
> NYSE_difflret.ts <-
> ts(data=NYSE_difflret,start=c(1984,24),frequency=252,
+ names=c("NYSE_diffLRet"))

> NYSE_est010 =  arima(NYSE_lret.ts, order = c(0,1,0), method = "ML")
> NYSE_est010

Call:
arima(x = NYSE_lret.ts, order = c(0, 1, 0), method = "ML")


sigma^2 estimated as 0.0001079:  log likelihood = 6293.55,  aic = -12585.1
> windows()
> tsdiag(NYSE_est010)
```

NYSE Data - ARIMA(0,1,0) Diagnostics

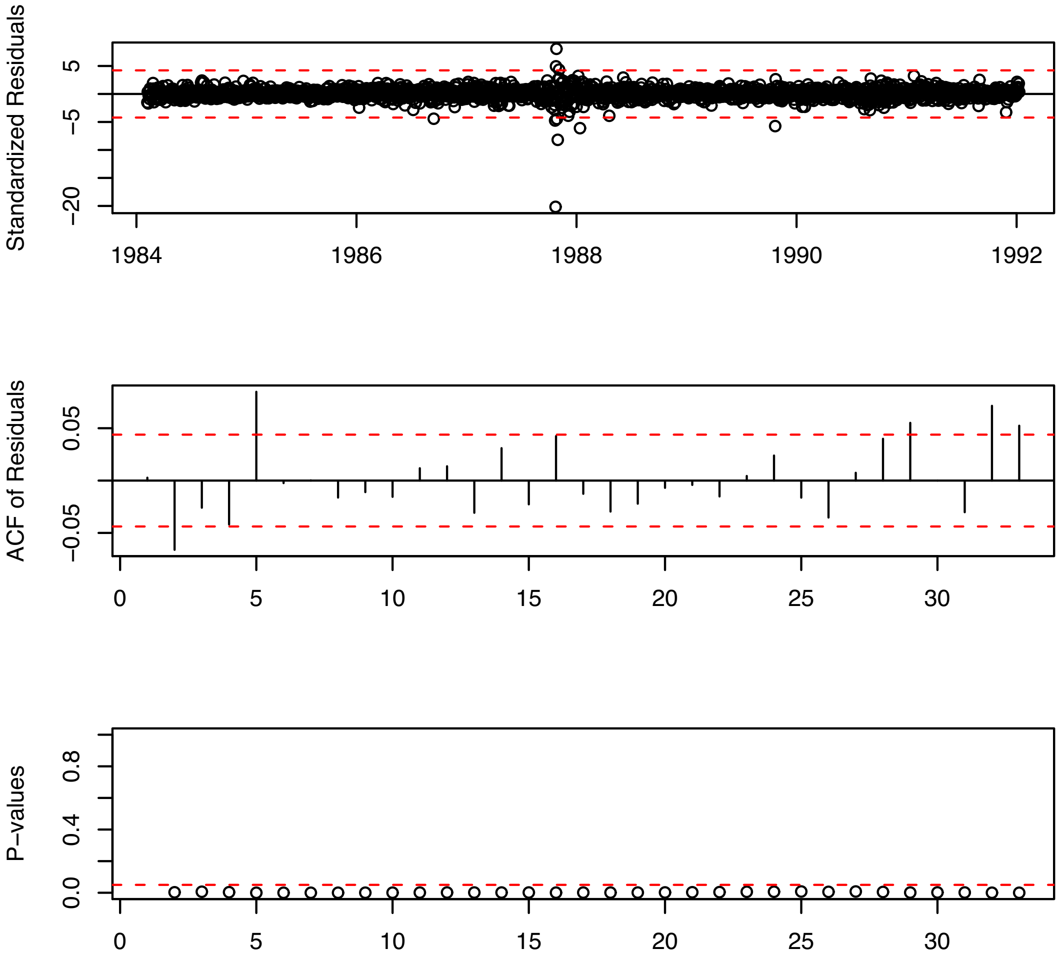


NYSE Data - ARIMA (1,1,0)

R-code

```
> NYSE_est110 =  arima(NYSE_lret.ts, order = c(1,1,0), method = "ML")
> NYSE_est110

Call:
arima(x = NYSE_lret.ts, order = c(1, 1, 0), method = "ML")

Coefficients:
         ar1
      0.0814
s.e.  0.0223

sigma^2 estimated as 0.0001072:  log likelihood = 6300.19,  aic = -12596.37
> windows()
> tsdiag(NYSE_est110)
```

NYSE Data - ARIMA(1,1,0) Diagnostics

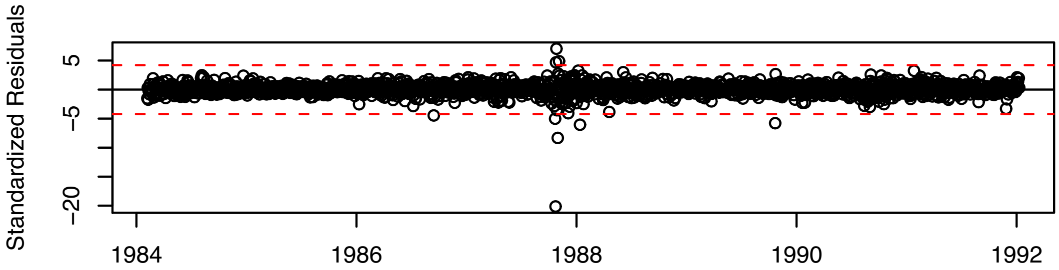


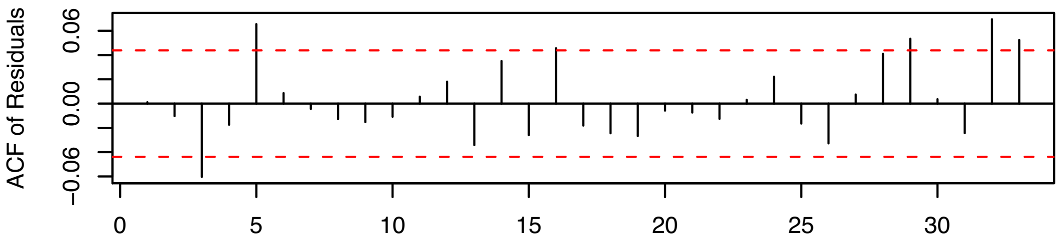
NYSE Data - ARIMA (1,1,1)

R-code

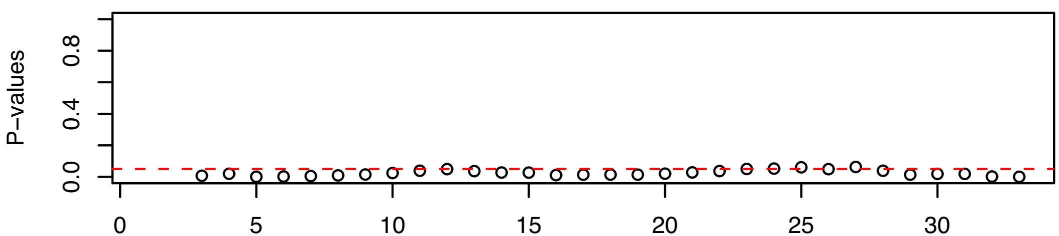
```
> NYSE_est111 =  arima(NYSE_lret.ts, order = c(1,1,1), method = "ML")
> NYSE_est111

Call:
arima(x = NYSE_lret.ts, order = c(1, 1, 1), method = "ML")

Coefficients:
          ar1     ma1
      -0.6428  0.7287
s.e.   0.0916  0.0817

sigma^2 estimated as 0.0001065:  log likelihood = 6306,  aic = -12606
> windows()
> tsdiag(NYSE_est111)
```

NYSE Data - ARIMA(1,1,1) Diagnostics





The residuals are not normal



NYSE - GARCH Analysis/Model

Apply GARCH modeling to NYSE Data

```
> NYSE_garchest11 = garch(NYSE_difflret.ts, order = c(1, 1), coef = NULL,
+ itmax = 200, eps = NULL, grad = c("analytic"))

 ***** ESTIMATION WITH ANALYTICAL GRADIENT *****

> summary(NYSE_garchest11)

Model:
GARCH(1,1)

Residuals:
    Min      1Q  Median      3Q     Max
-9.4156 -0.4476  0.0843  0.6039  4.0724

Coefficient(s):
    Estimate  Std. Error  t value Pr(>|t|)
a0 7.597e-06   7.309e-07    10.39   <2e-16 ***
a1 1.256e-01   3.907e-03    32.15   <2e-16 ***
b1 7.904e-01   1.313e-02    60.20   <2e-16 ***
---
```

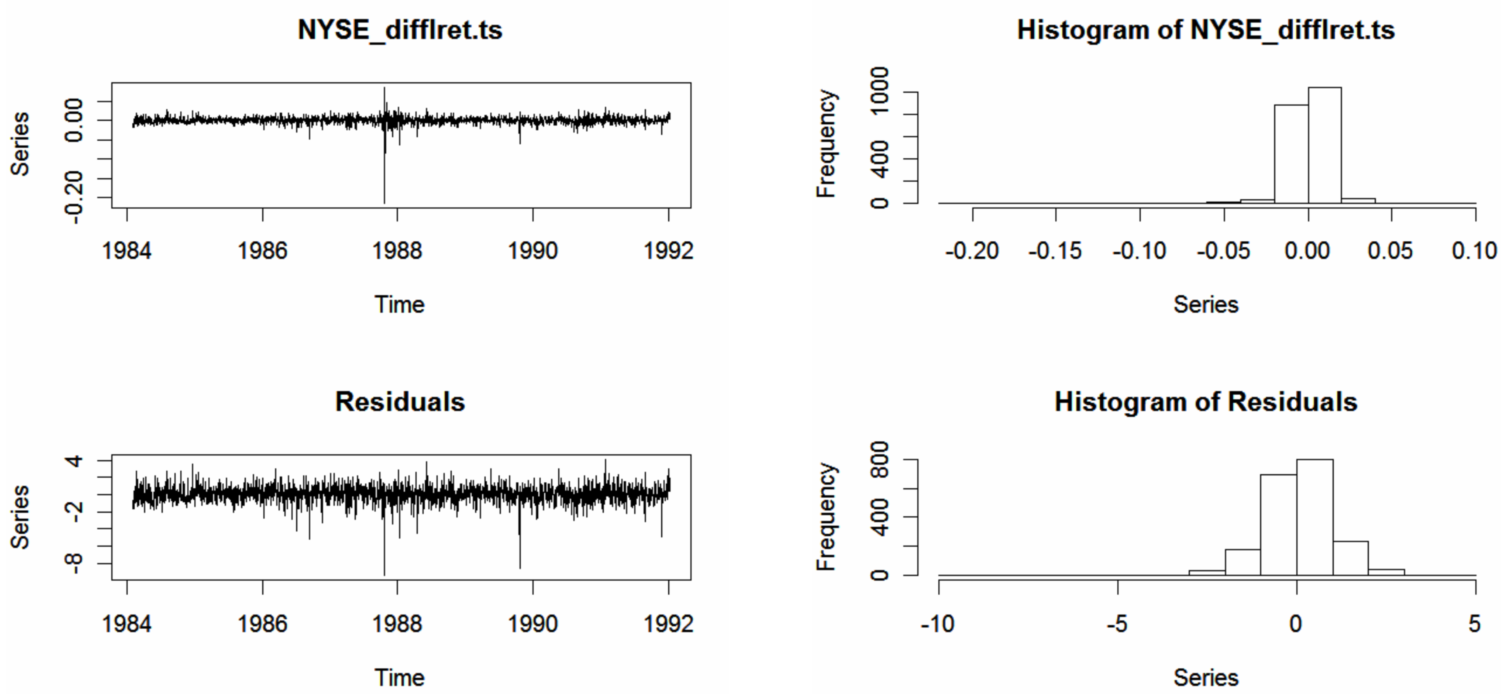
heavy tail

$\alpha_1+\beta_1 \approx 0.92$

```
Diagnostic Tests:
        Jarque Bera Test
data:  Residuals
X-squared = 5876.294, df = 2, p-value < 2.2e-16

        Box-Ljung test
data:  Squared.Residuals
X-squared = 1.408, df = 1, p-value = 0.2354
```

残差分布假设不成立

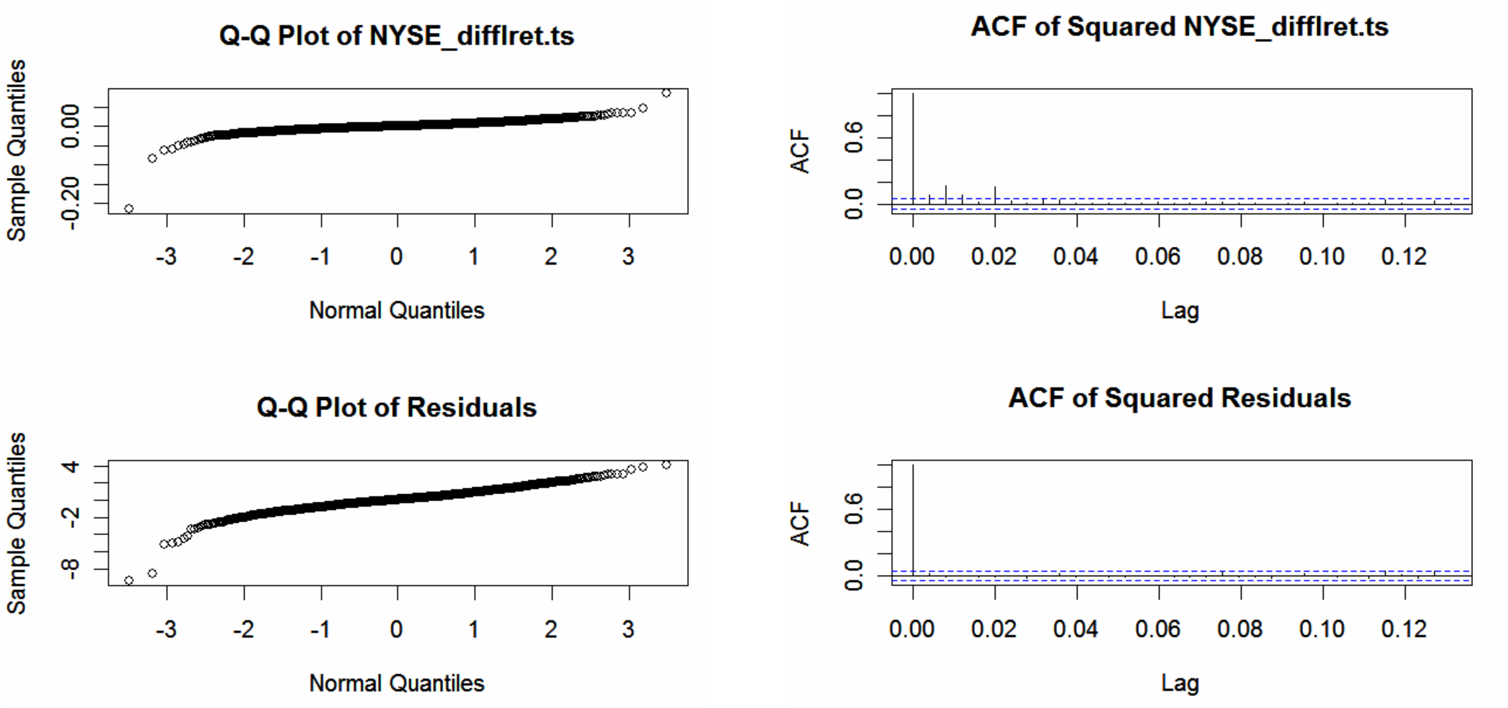
NYSE GARCH(1,1) - Diagnostic Plots



使用garch残差最低值从-20变成-9

NYSE GARCH(1,1) - Diagnostic Plots

QQ plot不对称



GARCH Variance Prediction

Remark. Based on estimated parameters in GARCH model, it is straightforward to do prediction.

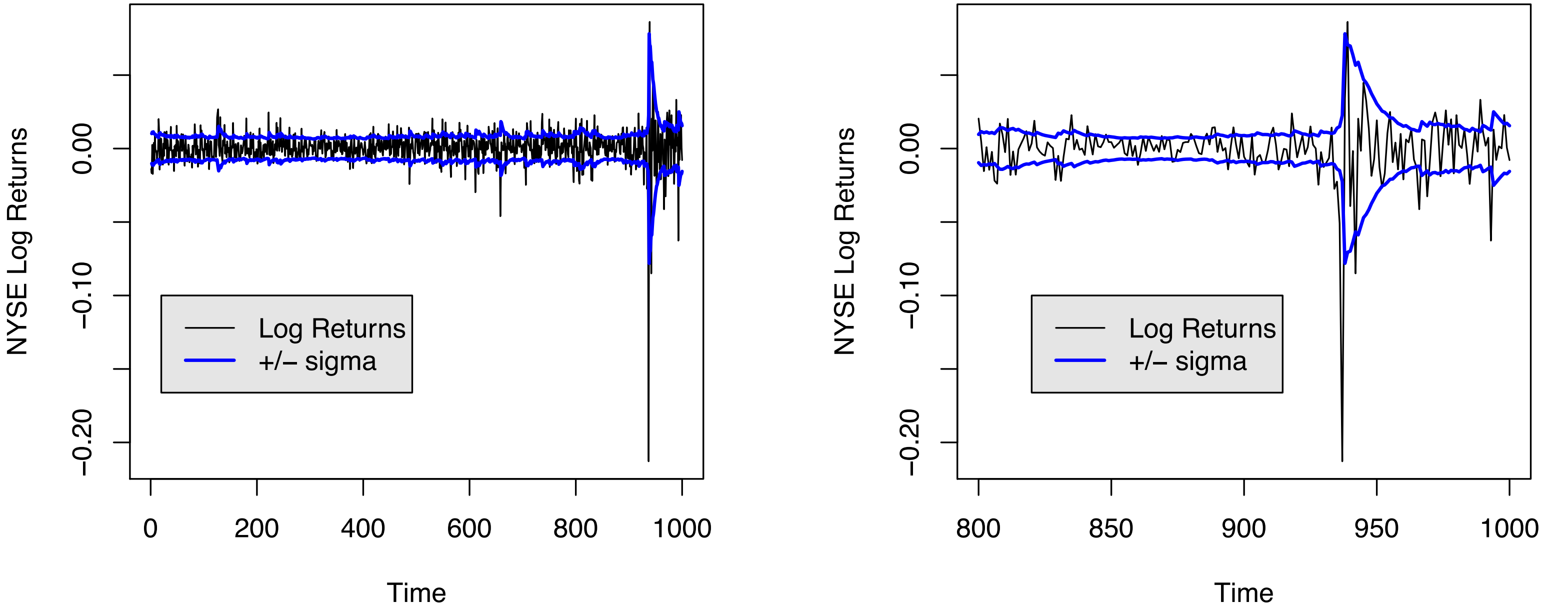
Example. Based on estimated parameters of GARCH model, can do one-step ahead estimates of σ_n^2 – for the NYSE, this corresponds to

$$\sigma_n^2 = 7.597e - 06 + (7.904e - 01) \cdot \sigma_{n-1}^2 + (1.256e - 01) \cdot X_{n-1}^2$$

GARCH: Plot of $\hat{\sigma}_n$ vs. n

Remark. For NYSE GARCH estimation, plot $\hat{\sigma}_n$ vs. n.

收敛过程拟合的很准



R-commands

```
> u = predict(NYSE_garchest11)
> windows()
> plot(c(1:1000),NYSE_difflret.ts[c(1:1000)],type='l',xlab='Time',
+ ylab='NYSE Log Returns')
> lines(c(1:1000),u[c(1:1000),1],lwd=2,lty=1,col='blue')
> lines(c(1:1000),u[c(1:1000),2],lwd=2,lty=1,col='blue')
> windows()
> plot(c(800:1000),NYSE_difflret.ts[c(800:1000)],type='l',xlab='Time',
+ ylab='NYSE Log Returns')
> lines(c(800:1000),u[c(800:1000),1],lwd=2,lty=1,col='blue')
> lines(c(800:1000),u[c(800:1000),2],lwd=2,lty=1,col='blue')
```

Additional Comments on ARCH and GARCH

Remark. Many variations on ARIMA/GARCH "time-series" theme (and used in financial data analysis) not covered in this course

- Robust (non-normal white-noise process) estimation
  - Heavier-tailed distributions on the white-noise process
- Mixing GARCH/ARCH with AR/ARMA/ARIMA (apply to the error process)
- APARCH and EGARCH (exponential GARCH) – different dynamics depending whether returns are negative or positive
- IGARCH (integrated GARCH) – time series is the integrated version of GARCH
  - Applicable for cases where have (even more) persistent volatility
- More sophisticated means of using one time series to predict/estimate another ($\{X_n\}$ used in predicting $\{Y_n\}$).

Additional Comments on ARCH and GARCH

Remark. In all cases for GARCH and its extensions

- common to have issues with relatively flat likelihood functions
- resulting in fragile convergence of iterative optimization algorithms (in deriving estimates)
- recommend rerunning from different initial values and look at diagnostics