

MATH 545: INTRODUCTION TO TIME SERIES ANALYSIS

WINTER 2025

CONDITIONAL HETEROSCEDASTIC MODELS

In a *conditional heteroscedastic model*, we have an observed data set with *time-varying variance*. Typically, these models can be considered as special forms of state space models. We typically assume that

$$Y_t = \sigma_t Z_t$$

where $\{Z_t\}$ is a Gaussian white noise process with variance 1; the volatility is therefore encapsulated in σ_t , and we adopt

- deterministic, observation-driven models,
- stochastic, parameter-driven models.

ARCH/GARCH MODELS: The ARCH and GARCH models use *deterministic* state space forms that allows the variance of the observed process to vary with time. The ARCH(1) model specifies $\sigma_t = \sqrt{X_t}$ so that

$$Y_t = \sqrt{X_t} Z_t$$

and then that

$$X_t = \alpha_0 + \alpha_1 Y_{t-1}^2$$

whereas the GARCH(1,1) model specifies

$$X_t = \alpha_0 + \alpha_1 Y_{t-1}^2 + \beta_1 X_{t-1}$$

where, in both cases, given $X_t = x_t$, Y_t has mean zero and variance x_t . The most general form of the GARCH model is the GARCH(p, q) model where

$$X_t = \alpha_0 + \sum_{j=1}^p \alpha_j Y_{t-j}^2 + \sum_{j=1}^q \beta_j X_{t-j}.$$

Moment properties: For GARCH(1, 1), by standard properties, we have that $\mathbb{E}[Y_t] = 0$, and using iterated expectation

$$\text{Var}[Y_t] = \mathbb{E}[Y_t^2] = \mathbb{E}[X_t Z_t^2] = \mathbb{E}[X_t] = \mathbb{E}[\alpha_0 + \alpha_1 Y_{t-1}^2 + \beta_1 X_{t-1}].$$

As $\text{Var}[Y_t] = \mathbb{E}[Y_t^2] = \mathbb{E}[X_t] = \mathbb{E}[X_{t-1}] = \mathbb{E}[Y_{t-1}^2]$ for all t , we may conclude (by analogy with results for AR(1)) that under stationarity

$$\text{Var}[Y_t] = \alpha_0 + \alpha_1 \text{Var}[Y_t] + \beta_1 \text{Var}[Y_t]$$

so that

$$\text{Var}[Y_t] = \frac{1}{1 - \alpha_1 - \beta_1} = \sigma_Y^2,$$

say, and that a condition for stationarity is that $\alpha_1 + \beta_1 < 1$. It is evident also that for $h = \pm 1, \pm 2, \dots$

$$\mathbb{E}[Y_t Y_{t+h}] = \mathbb{E}[\sqrt{X_t} Z_t \sqrt{X_{t+h}} Z_{t+h}] = \mathbb{E}[\sqrt{X_t X_{t+h}}] \mathbb{E}[Z_t Z_{t+h}] = 0$$

as $\{Z_t\}$ is white noise. However,

$$\text{Var}[Y_t^2] = \mathbb{E}[Y_t^4] - \{\mathbb{E}[Y_t^2]\}^2 = \mathbb{E}[X_t^2 Z_t^4] - \sigma_Y^4 = 3\mathbb{E}[X_t^2] - \sigma_Y^4$$

where

$$\mathbb{E}[X_t^2] = \text{Var}[X_t] + \{E[X_t]\}^2 = \frac{\sigma_W^2}{1 - \phi^2} + \mu_X^2 = \sigma_X^2 + \mu_X^2$$

say. Also, for $h = \pm 1, \pm 2, \dots$,

$$\mathbb{E}[Y_t^2 Y_{t+h}^2] = \mathbb{E}[X_t Z_t^2 X_{t+h} Z_{t+h}^2] = \mathbb{E}[X_t X_{t+h}] \mathbb{E}[Z_t^2 Z_{t+h}^2] = \mathbb{E}[X_t X_{t+h}] \neq 0$$

Then

$$\begin{aligned}
\mathbb{E}[Y_t^2 Y_{t+h}^2] &= \mathbb{E}[(\alpha_0 + \alpha_1 Y_{t-1}^2 + \beta_1 X_{t-1})(\alpha_0 + \alpha_1 Y_{t+h-1}^2 + \beta_1 X_{t+h-1})] \\
&= \alpha_0^2 + \alpha_0 \alpha_1 \mathbb{E}[Y_{t+h-1}^2] + \alpha_0 \beta_1 \mathbb{E}[X_{t+h-1}] \\
&\quad + \alpha_0 \alpha_1 \mathbb{E}[Y_{t-1}^2] + \alpha_1^2 \mathbb{E}[Y_{t-1}^2 Y_{t+h-1}^2] + \alpha_1 \beta_1 \mathbb{E}[X_{t+h-1} Y_{t-1}^2] \\
&\quad + \alpha_0 \beta_1 \mathbb{E}[X_{t-1}] + \alpha_1 \beta_1 \mathbb{E}[X_{t-1} Y_{t+h-1}^2] + \beta_1^2 \mathbb{E}[X_{t-1} X_{t+h-1}]
\end{aligned}$$

In this expression,

$$\begin{aligned}
\mathbb{E}[X_t X_{t+h}] &= \gamma_{Y^2}(h) \\
\mathbb{E}[Y_{t+h-1}^2] &= \sigma_Y^2 \\
\mathbb{E}[X_{t+h-1}] &= \alpha_0 + \alpha_1 \mathbb{E}[Y_{t+h-2}^2] = \alpha_0 + \alpha_1 \sigma_Y^2 \\
\mathbb{E}[Y_{t-1}^2 Y_{t+h-1}^2] &= \gamma_{Y^2}(h) \\
\mathbb{E}[X_{t+h-1} Y_{t-1}^2] &= \mathbb{E}[(\alpha_0 + \alpha_1 Y_{t+h-2}^2) Y_{t-1}^2] = \alpha_0 \sigma_Y^2 + \alpha_1 \gamma_{Y^2}(h-1) \\
\mathbb{E}[X_{t-1} Y_{t+h-1}^2] &= \mathbb{E}[(\alpha_0 + \alpha_1 Y_{t-2}^2) Y_{t+h-1}^2] = \alpha_0 \sigma_Y^2 + \alpha_1 \gamma_{Y^2}(h+1) \\
\mathbb{E}[X_{t-1} X_{t+h-1}] &= \gamma_{Y^2}(h)
\end{aligned}$$

so

$$\begin{aligned}
\mathbb{E}[Y_t^2 Y_{t+h}^2] &= \alpha_0^2 + \alpha_0 \alpha_1 \sigma_Y^2 + \alpha_0 \beta_1 (\alpha_0 + \alpha_1 \sigma_Y^2) \\
&\quad + \alpha_0 \alpha_1 \sigma_Y^2 + \alpha_1^2 \gamma_{Y^2}(h) + \alpha_1 \beta_1 (\alpha_0 \sigma_Y^2 + \alpha_1 \gamma_{Y^2}(h-1)) \\
&\quad + \alpha_0 \beta_1 (\alpha_0 + \alpha_1 \sigma_Y^2) + \alpha_1 \beta_1 (\alpha_0 \sigma_Y^2 + \alpha_1 \gamma_{Y^2}(h+1)) + \beta_1^2 \gamma_{Y^2}(h)
\end{aligned}$$

This ultimately gives a linear recursion of the form

$$\gamma_{Y^2}(h+1) = A + B \gamma_{Y^2}(h) + C \gamma_{Y^2}(h-1)$$

with initial conditions obtained by examining $h = 0, 1$.

STOCHASTIC VOLATILITY MODELS: The *stochastic volatility* model uses a stochastic state space form that allows the variance of the observed process to vary stochastically with time. The simplest form specifies that

$$Y_t = \exp\{X_t/2\} Z_t \quad \{Z_t\} \sim \text{WN}(0, \sigma_Z^2)$$

and then that

$$X_{t+1} = \phi X_t + W_t \quad \{W_t\} \sim \text{WN}(0, \sigma_W^2)$$

that is, given $X_t = x_t$, Y_t has mean zero and variance $\exp\{x_t\}$, and $\{X_t\} \sim \text{AR}(1)$. We can set $\sigma_Z^2 = 1$, and can reparameterize the model as follows:

$$Y_t = \exp\{X_t/2\} Z_t$$

with $\{Z_t\} \sim \text{WN}(0, 1)$, and

$$X_{t+1} = \phi_0 + \phi X_t + W_t$$

or equivalently

$$X_{t+1} = \mu_X + \phi(X_t - \mu_X) + W_t$$

where $\phi_0 = (1 - \phi)\mu_X$, and $\{W_t\} \sim \text{WN}(0, \sigma_W^2)$. For this model it is evident that $\mathbb{E}[Y_t] = 0$ for all t , and

$$\text{Var}[Y_t] = \mathbb{E}[Y_t^2] = \mathbb{E}[\exp\{X_t\} Z_t^2] = \mathbb{E}[\exp\{X_t\}],$$

whereas under stationarity, if $\{Z_t\}$ and $\{W_t\}$ are uncorrelated

$$\gamma_Y(h) = \mathbb{E}[Y_t Y_{t+h}] = \mathbb{E}[\exp\{X_t/2\} Z_t \exp\{X_{t+h}/2\} Z_{t+h}] = \mathbb{E}[\exp\{(X_t + X_{t+h})/2\}] \mathbb{E}[Z_t Z_{t+h}] = 0.$$

Now

$$\text{Var}[Y_t^2] = \mathbb{E}[Y_t^4] - \{\mathbb{E}[Y_t^2]\}^2 = \mathbb{E}[\exp\{2X_t\} Z_t^4] - \{\mathbb{E}[\exp\{X_t\}]\}^2 = \mathbb{E}[\exp\{2X_t\}] \mathbb{E}[Z_t^4] - \{\mathbb{E}[\exp\{X_t\}]\}^2.$$

Under the assumption of Normality of the $\{Z_t\}$ process, we have $\mathbb{E}[Z_t^4] = 3$ by properties of the standard Normal. Similarly

$$\mathbb{E}[Y_t^2 Y_{t+h}^2] = \mathbb{E}[\exp\{X_t\} Z_t^2 \exp\{X_{t+h}\} Z_{t+h}^2] = \mathbb{E}[\exp\{(X_t + X_{t+h})\}] \mathbb{E}[Z_t^2 Z_{t+h}^2] = \mathbb{E}[\exp\{(X_t + X_{t+h})\}]$$

as $\mathbb{E}[Z_t^2] = 1$. So for $h \geq 1$,

$$\gamma_{Y^2}(h) = \mathbb{E}[Y_t^2 Y_{t+h}^2] - \{\mathbb{E}[Y_t^2]\}^2 = \mathbb{E}[\exp\{(X_t + X_{t+h})\}] - \{\mathbb{E}[\exp\{X_t\}]\}^2$$

To study properties of this model we use properties of the log-normal distribution that allow us to compute

$$\mathbb{E}[\exp\{X_t\}] \quad \mathbb{E}[\exp\{2X_t\}] \quad \mathbb{E}[\exp\{(X_t + X_{t+h})\}].$$

Specifically

$$\mathbb{E}[\exp(X_t)^j] = \exp\left\{j\mu_X + \frac{1}{2}j^2\sigma_X^2\right\},$$

so that, if $r = 2s$ is even and $\{X_t\}$ is stationary, all the even moments of Y_t exist and are given by the formula

$$\mathbb{E}[Y_t^r] = \mathbb{E}[(\exp(X_t))^s] \mathbb{E}[Z^r] = \exp\left\{s\mu_X + \frac{s^2}{2}\sigma_X^2\right\} \frac{(2s)!}{2^s s!}.$$

whereas all the odd moments are zero. Thus

$$\text{Var}[Y_t] = \mathbb{E}[Y_t^2] = \mathbb{E}[\exp\{X_t\}] = \exp\left\{\mu_X + \frac{1}{2}\sigma_X^2\right\}$$

$$\text{Var}[Y_t^2] = 3\mathbb{E}[\exp\{2X_t\}] - \{\mathbb{E}[\exp\{X_t\}]\}^2 = 3\exp\{2\mu_X + 2\sigma_X^2\} - \exp\{2\mu_X + \sigma_X^2\}.$$

Now, for this model, we know that $X_t + X_{t+h}$ is a Normal random variable with mean $2\mu_X$. We have by standard arguments that

$$\begin{aligned} \text{Cov}[X_t, X_{t+h}] &= \text{Cov}[X_t, \mu_X + \phi(X_{t+h-1} - \mu)] \\ &= \text{Cov}[X_t, \phi X_{t+h-1}] \\ &= \text{Cov}[X_t, \phi^h X_t] && \text{recursion} \\ &= \phi^h \text{Var}[X_t] \\ &= \phi^h \sigma_X^2. \end{aligned}$$

Therefore we have that

$$\text{Var}[X_t + X_{t+h}] = \text{Var}[X_t] + \text{Var}[X_{t+h}] + 2\text{Cov}[X_t, X_{t+h}] = 2\sigma_X^2 + 2\phi^h \sigma_X^2.$$

so that

$$X_t + X_{t+h} \sim \text{Normal}(2\mu_X, 2\sigma_X^2(1 + \phi^h)).$$

and hence for $h \geq 1$

$$\gamma_{Y^2}(h) = \mathbb{E}[\exp\{(X_t + X_{t+h})\}] - \{\mathbb{E}[\exp\{X_t\}]\}^2 = \exp\{2\mu_X + \sigma_X^2(1 + \phi^h)\} - \exp\{2\mu_X + \sigma_X^2\}$$

which shows that Y_t^2 is an autocorrelated process. We have that the ACF for $h = \pm 1, \pm 2, \dots$ is

$$\begin{aligned} \rho_{Y^2}(h) &= \frac{\exp\{2\mu_X + \sigma_X^2(1 + \phi^h)\} - \exp\{2\mu_X + \sigma_X^2\}}{3\exp\{2\mu_X + 2\sigma_X^2\} - \exp\{2\mu_X + \sigma_X^2\}} \\ &= \frac{\exp\{\sigma_X^2\phi^h\} - 1}{3\exp\{\sigma_X^2\} - 1} \end{aligned}$$

Simulation study: We begin with the GARCH models, and simulate from the ARCH(1) model with $\alpha_0 = 0.1$ and $\alpha_1 = 0.95$.

```
set.seed(3245)
n<-1000;al0<-0.01;al1<-0.95
X<-Y<-rep(0,n);X[1]<-al0;Y[1]<-rnorm(1)*sqrt(X[1]) #ARCH(1) MODEL
for(j in 2:n){
  X[j]<-al0+al1*Y[j-1]^2
  Y[j]<-rnorm(1)*sqrt(X[j])
}
```

GARCH models can be fitted using the fGarch library.

```
library(fGarch)
mydata<-data.frame(Y)

#Fitting the ARCH(1) model
fit.10<-garchFit(Y~garch(1,0),data=mydata,trace=F) #ARCH(1)
show(fit.10)

+
+ Title:
+   GARCH Modelling
+
+ Call:
+   garchFit(formula = Y ~ garch(1, 0), data = mydata, trace = F)
+
+ Mean and Variance Equation:
+   data ~ garch(1, 0)
+ <environment: 0x0000017d95b022b0>
+   [data = mydata]
+
+ Conditional Distribution:
+   norm
+
+ Coefficient(s):
+      mu      omega    alpha1
+ 0.0044008 0.0103478 0.9945946
+
+ Std. Errors:
+   based on Hessian
+
+ Error Analysis:
+      Estimate Std. Error t value      Pr(>|t|)
+ mu      0.0044008  0.0037231   1.182      0.237
+ omega   0.0103478  0.0008145  12.704 <0.0000000000000002 ***
+ alpha1  0.9945946  0.0799572  12.439 <0.0000000000000002 ***
+ ---
+ Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
+
+ Log Likelihood:
+   325.1824    normalized:  0.3251824
+
+ Description:
+   Fri Apr 11 08:55:20 2025 by user: David Stephens

#Fitting the GARCH(1,1) model
fit.11<-garchFit(Y~garch(1,1),data=mydata,trace=F) #GARCH(1,1)
show(fit.11)

+
+ Title:
```

```

+ GARCH Modelling
+
+ Call:
+ garchFit(formula = Y ~ garch(1, 1), data = mydata, trace = F)
+
+ Mean and Variance Equation:
+ data ~ garch(1, 1)
+ <environment: 0x0000017d96b190b0>
+ [data = mydata]
+
+ Conditional Distribution:
+ norm
+
+ Coefficient(s):
+      mu      omega      alpha1      beta1
+ 0.0045937 0.0100124 0.9878371 0.0098542
+
+ Std. Errors:
+ based on Hessian
+
+ Error Analysis:
+      Estimate Std. Error t value      Pr(>|t|)
+ mu      0.004594   0.003722   1.234      0.217
+ omega   0.010012   0.000869  11.521 <0.00000000000000002 ***
+ alpha1  0.987837   0.080238  12.311 <0.00000000000000002 ***
+ beta1   0.009854   0.011710   0.842      0.400
+ ---
+ Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
+
+ Log Likelihood:
+ 325.8299      normalized: 0.3258299
+
+ Description:
+ Fri Apr 11 08:55:21 2025 by user: David Stephens

c(fit.10@fit$ics[1],fit.11@fit$ics[1]) #AIC comparison

+      AIC      AIC
+ -0.6443647 -0.6436599

fit.10@fit$coef

+      mu      omega      alpha1
+ 0.00440077 0.01034776 0.99459456

fit.11@fit$coef

+      mu      omega      alpha1      beta1
+ 0.004593669 0.010012413 0.987837079 0.009854181

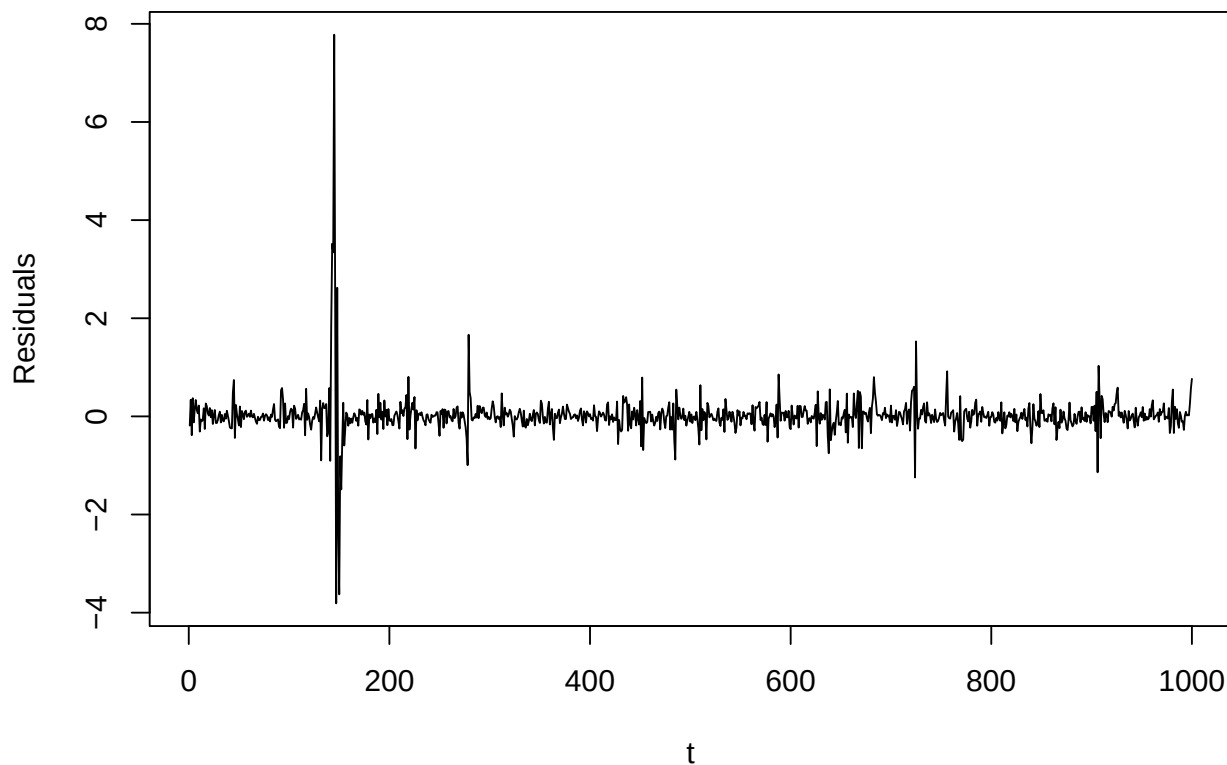
```

Thus the maximum likelihood estimated model is

$$\begin{aligned}
 \text{ARCH}(1) \quad X_t &= 0.0103478 + 0.9945946Y_{t-1}^2 \\
 \text{GARCH}(1,1) \quad X_t &= 0.0100124 + 0.9878371Y_{t-1}^2 + 0.0098542X_{t-1}
 \end{aligned}$$

The ARCH(1) model is preferred under the AIC comparison. The parameter referred to as omega is α_0 , and mu is the mean of the $\{Y_t\}$ series. If we inspect the residual plot resulting from the ARCH(1) fit, the variance of the residuals seem to be adequate, apart from in one part of the plot.

```
plot(residuals(fit.10),type='l',xlab='t',ylab='Residuals')
```



Stochastic volatility simulation: For the SV simulation, in the original parameterization we have

$$Y_t = \exp\{X_t/2\}Z_t$$

with $\sigma_Z^2 = 0.01$, and

$$X_t = \phi X_{t-1} + W_t$$

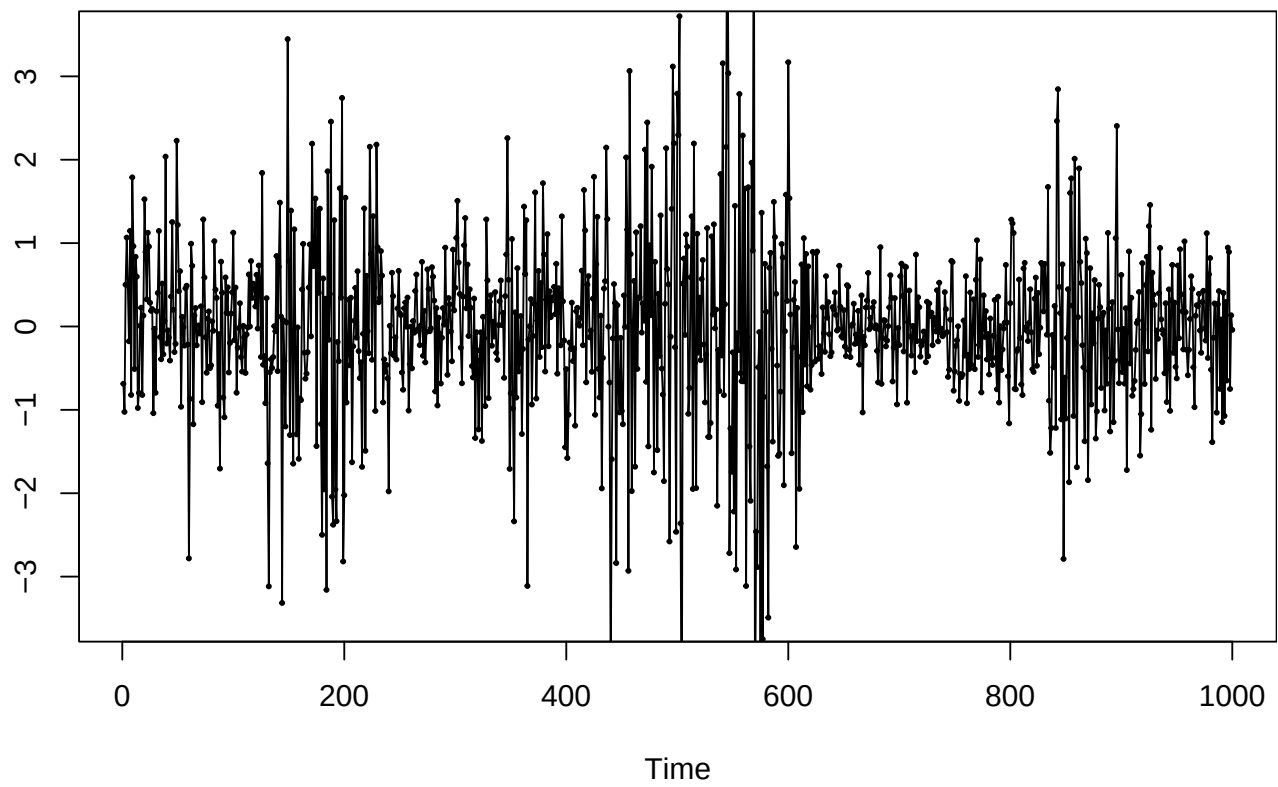
with $\sigma_W^2 = 0.05$. If we re-scale Z_t to be unit variance, then this the same model as obtained by re-writing the state equation as

$$X_t = \mu_X + \phi(X_{t-1} - \mu_X) + W_t$$

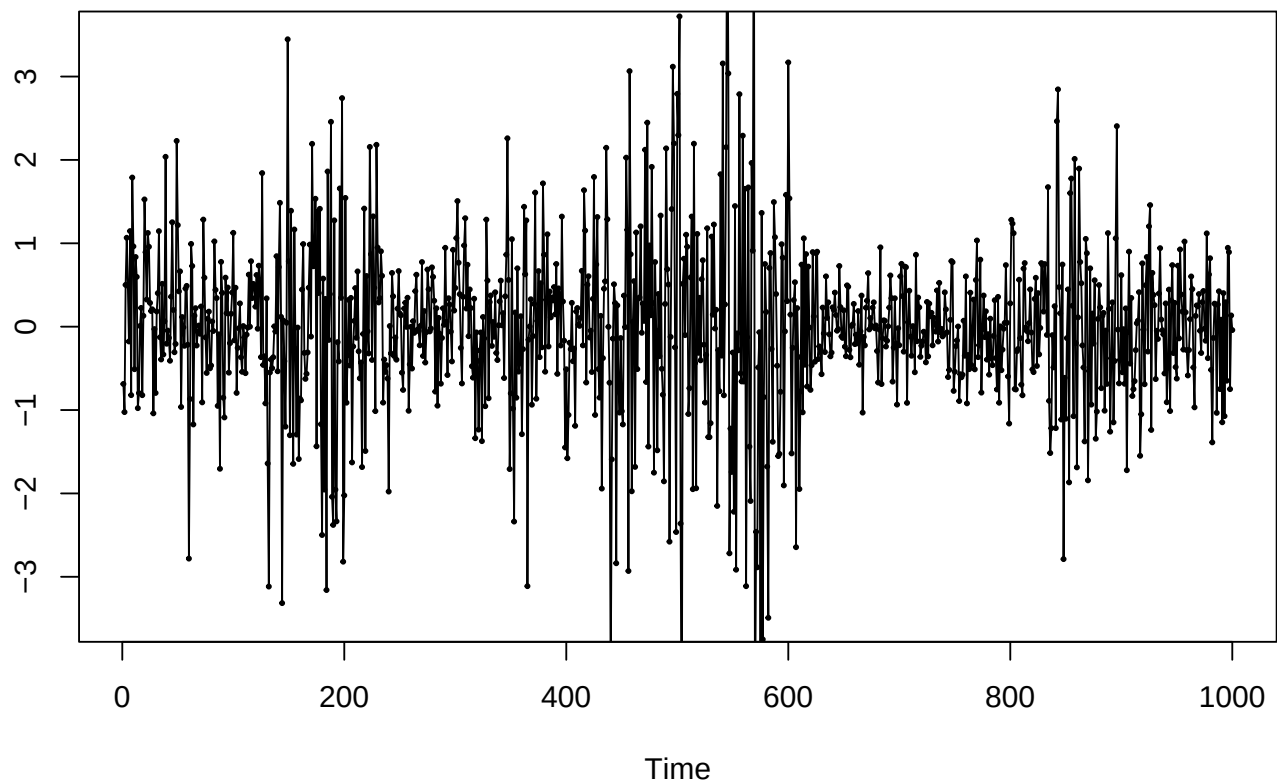
with $\mu_X = 2 \log \sigma_Z^2$.

```
set.seed(24)
n<-1000;phi<-0.98;sigZ<-0.01;sigW<-0.05
X<-arima.sim(n,model=list(ar=phi),rand.gen = function(n) sqrt(sigW) * rnorm(n))
Z<-rnorm(n)*sigZ
Y0<-exp(X/2)*Z

set.seed(24)
mu<-2*log(sigZ) #Reparameterization
X<-arima.sim(n,model=list(ar=phi),rand.gen = function(n) sqrt(sigW) * rnorm(n))
X<-X+mu
Z<-rnorm(n)
Y<-exp(X/2)*Z
par(mar=c(4,3,2,0));plot(100*Y0,type='l',ylim=range(-3.5,3.5));
points(100*Y0,pch=19,cex=0.3)
```



```
plot(100*Y,type='l',ylim=range(-3.5,3.5)); points(100*Y,pch=19,cex=0.3)
```

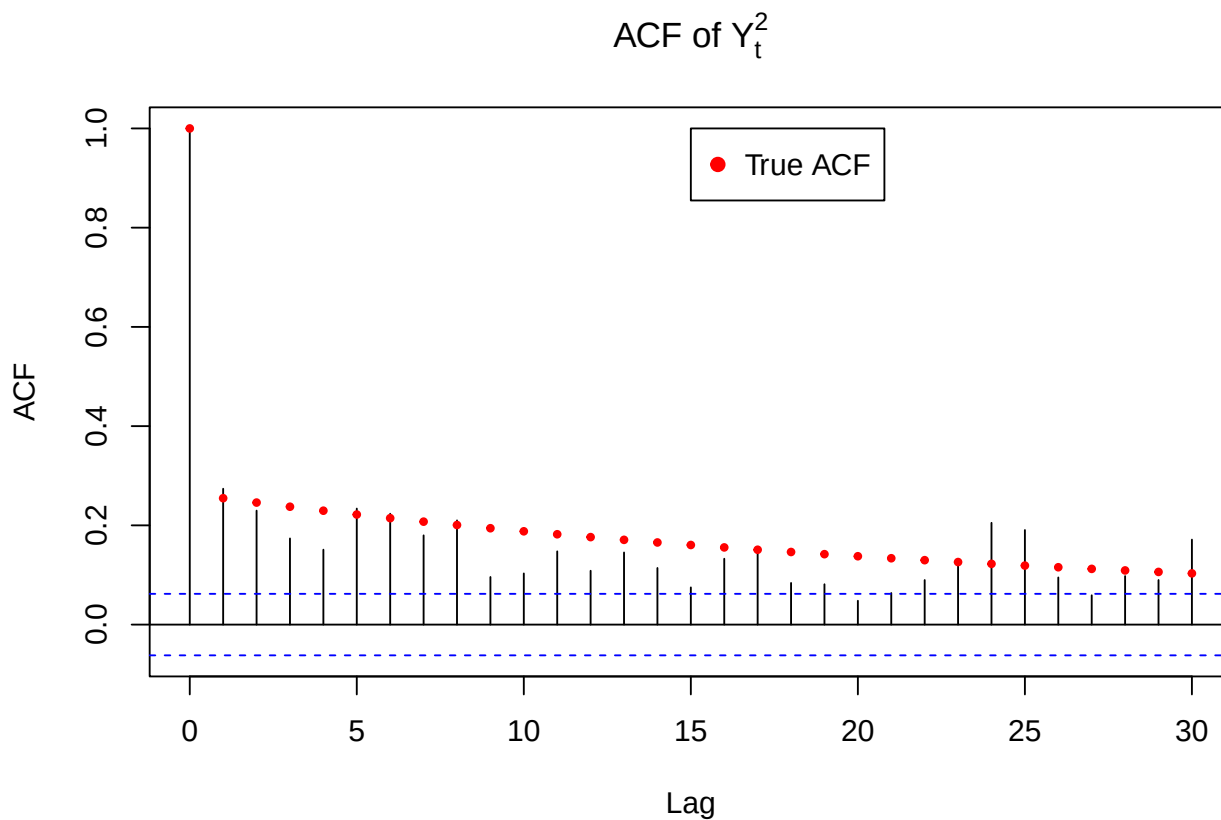


We can compare the theoretical ACF with the sample ACF

```
muX<-mu
sigX<-sigW/(1-phi^2)
gamma0<-3*exp(2*muX+2*sigX)-exp(2*muX+sigX)
hvec<-c(1:30)
gammah<-exp(2*muX+sigX*(1+phi^hvec))-exp(2*muX+sigX)
gammah/gamma0 #ACF

+ [1] 0.2547408 0.2459688 0.2375803 0.2295547 0.2218724 0.2145155 0.2074668
+ [8] 0.2007103 0.1942312 0.1880152 0.1820493 0.1763208 0.1708183 0.1655305
+ [15] 0.1604471 0.1555585 0.1508552 0.1463287 0.1419706 0.1377733 0.1337294
+ [22] 0.1298320 0.1260745 0.1224507 0.1189548 0.1155811 0.1123243 0.1091796
+ [29] 0.1061421 0.1032073

acf.vec<-c(gamma0,gammah)/gamma0
acf(Y^2,lag.max=30,,main=expression(paste('ACF of ',Y[t]^2)))
points(c(0,hvec),acf.vec,pch=19,cex=0.5,col='red')
legend(15,1.0,c('True ACF'),pch=19,col='red')
```

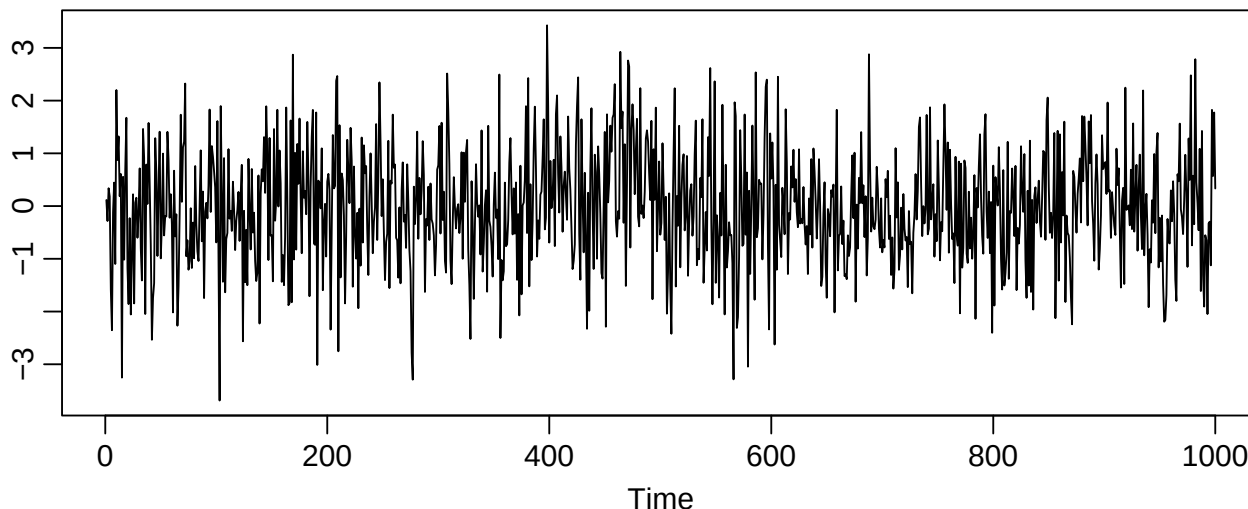


Inference for the SV model can be carried out using Bayesian methods and Markov chain Monte Carlo (MCMC) and the `stochvol` library.

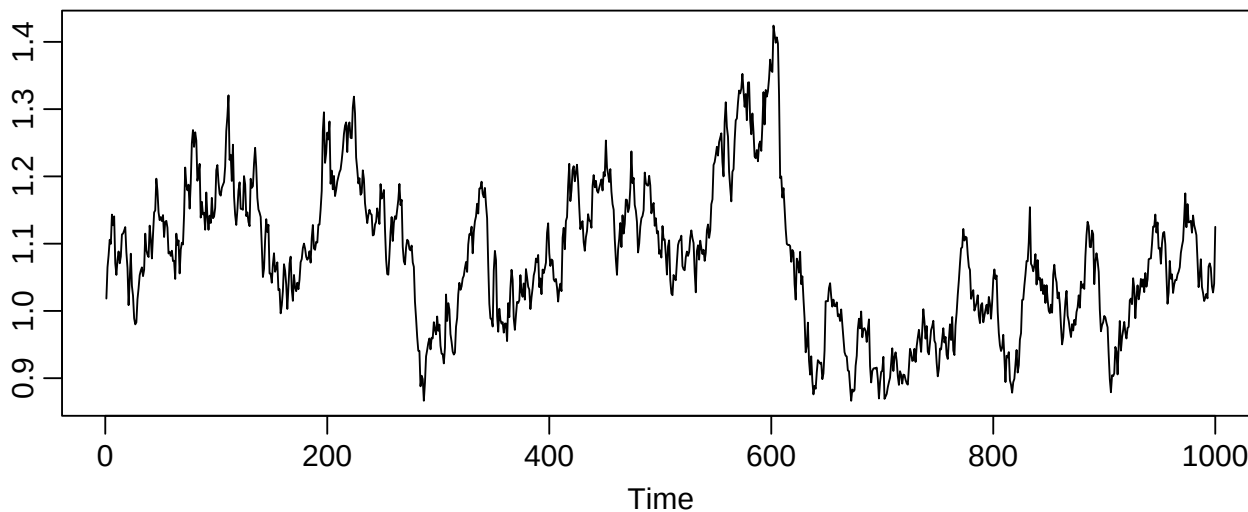
- `svsim` simulates a SV model
- `svsample` carries out MCMC analysis

```
library(stochvol)
y.sim <- svsim(1000, mu = 2*log(sigZ), phi = phi, sigma = sigW)
plot(y.sim)
```

Simulated data: 'log-returns' (in %)



Simulated volatilities (in %)



The MCMC algorithm draws 5000 samples from the *posterior distribution* for the three parameters of the model μ , ϕ and σ_W .

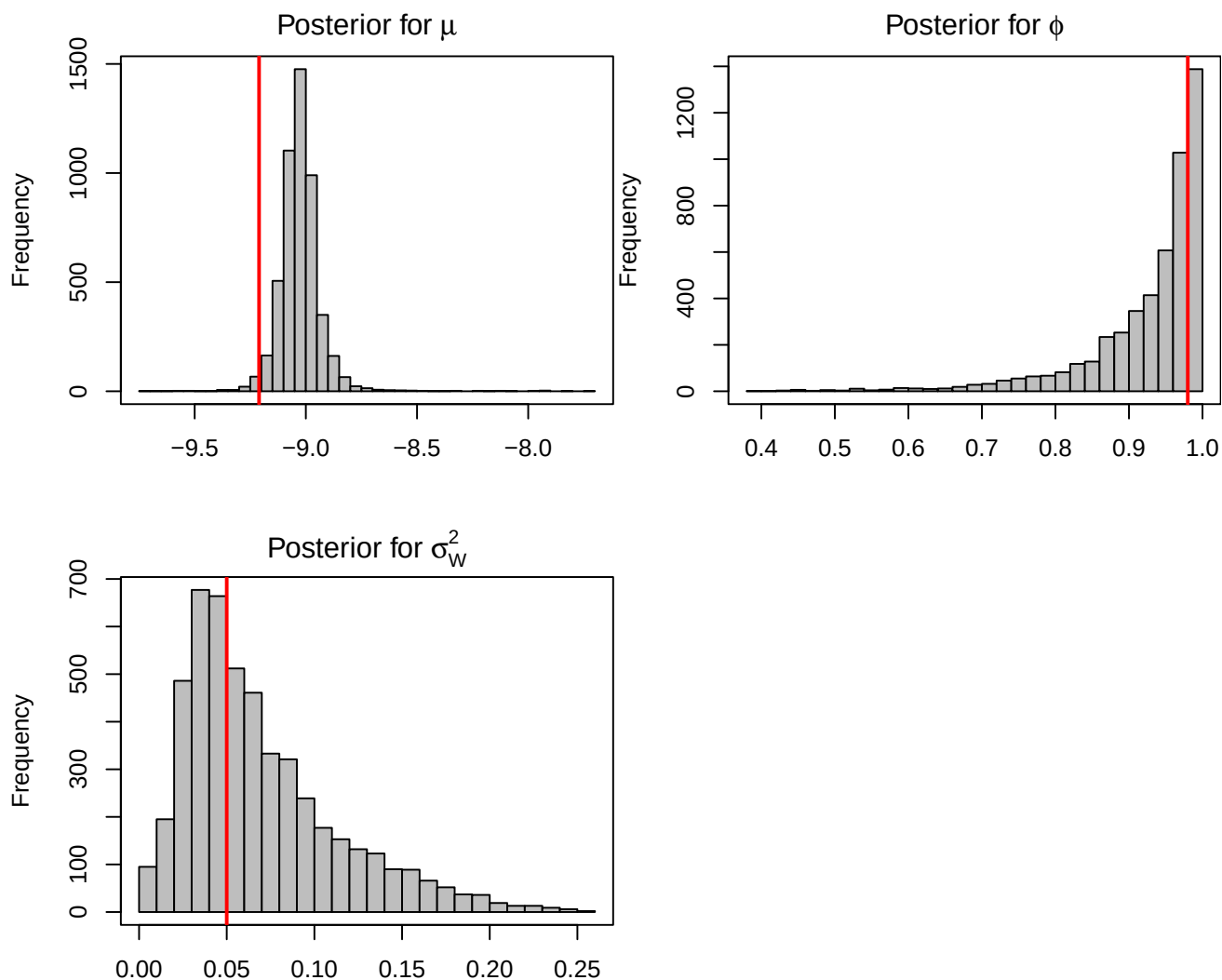
```
mcmc.draws <- svsample(y.sim$y, draws = 5000, burnin = 1000, priormu = c(0, 10),
  priorphi = c(20, 1.2), priorsigma = 100)
```

```
+ Done!
+ Summarizing posterior draws...
```

```

para<-data.matrix(mcmc.draws$para)
par(mfrow=c(2,2),mar=c(4,4,2,0))
hist(para[,1],col='gray',nclass=30,xlab='',
      main=expression(paste('Posterior for ',mu)))
box();abline(v=mu,col='red',lwd=2) #Mark the true values with red lines.
hist(para[,2],col='gray',nclass=30,xlab='',
      main=expression(paste('Posterior for ',phi)))
box();abline(v=phi,col='red',lwd=2)
hist(para[,3],col='gray',nclass=30,xlab='',
      main=expression(paste('Posterior for ',sigma[W]^2)))
box();abline(v=sigW,col='red',lwd=2)

```



Parameter estimates can be obtained from the posterior distribution by summarizing the posterior samples:

```

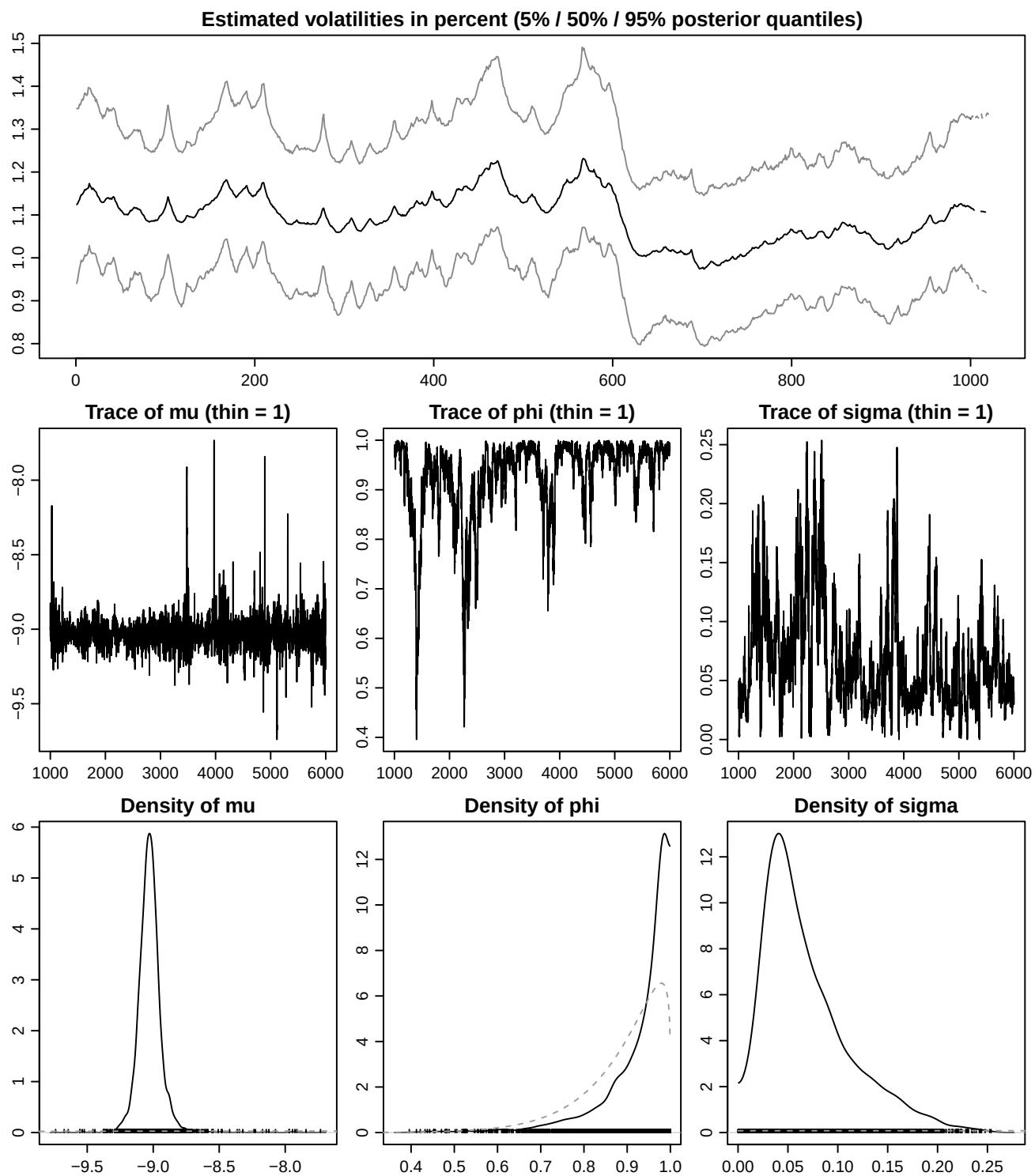
apply(para,2,summary)

```

	mu	phi	sigma	nu	rho
+ Min.	-9.739574	0.3958921	0.00006361955	Inf	0
+ 1st Qu.	-9.075656	0.9023020	0.03732497758	Inf	0
+ Median	-9.029274	0.9576573	0.05677469285	Inf	0
+ Mean	-9.025288	0.9266686	0.06913500867	Inf	0
+ 3rd Qu.	-8.984027	0.9817176	0.09039690730	Inf	0
+ Max.	-7.732702	0.9998031	0.25367060995	Inf	0

We can also forecast using the model, with commands

```
fore <- predict(mcmc.draws, 20)
plot(mcmc.draws, forecast = fore)
```



Real Data Analysis: For an analysis of a real data set, we consider the S & P 500 index over the period 2007-2017.

```
SP.2017<-read.csv('GSPC.csv',header=TRUE)
X<-SP.2017$Close
Tv<-as.Date(as.character(SP.2017$Date), '%d/%m/%Y')
par(mar=c(4,4,2,0))
plot(Tv,X,type='l',main='S & P 500 index',xlab='Date')
```

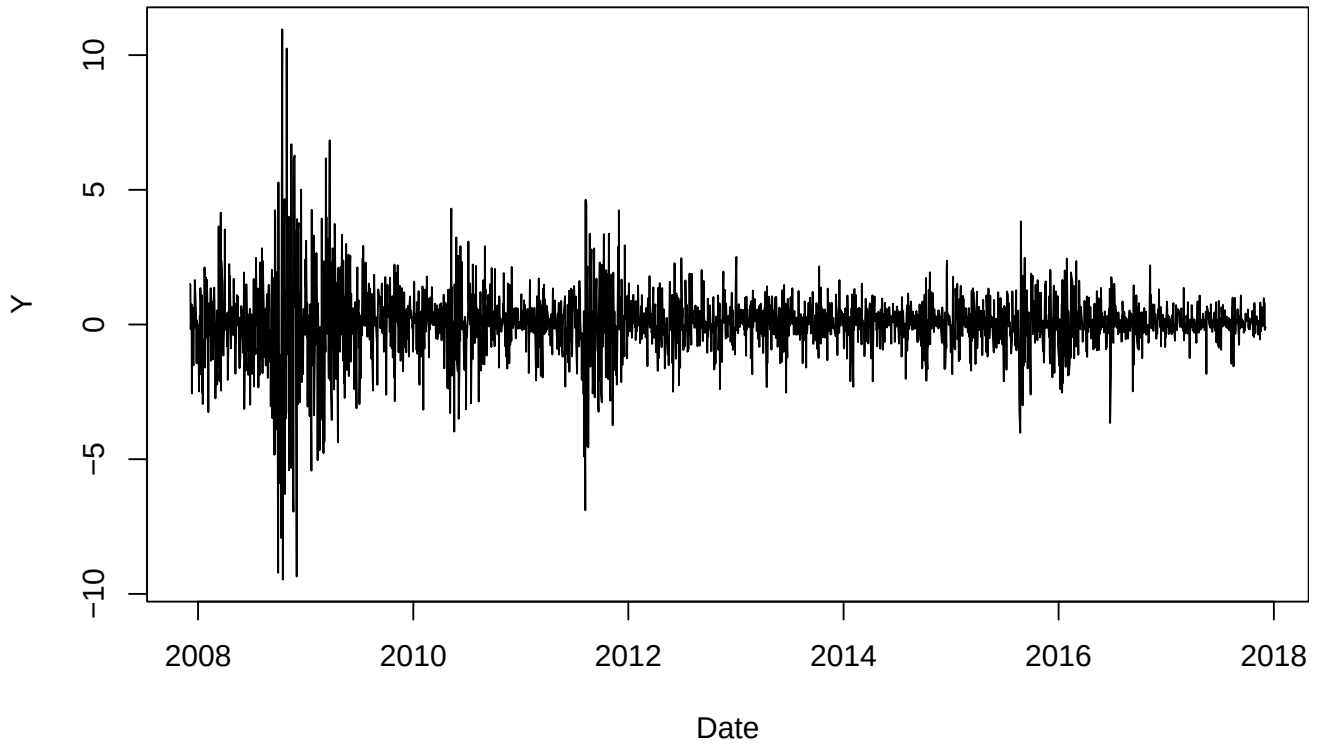


If we let X_t denote the index value, we then transform to the percentage log returns series

$$Y_t = 100 \log(X_{t+1}/X_t) = 100(\log X_{t+1} - \log X_t)$$

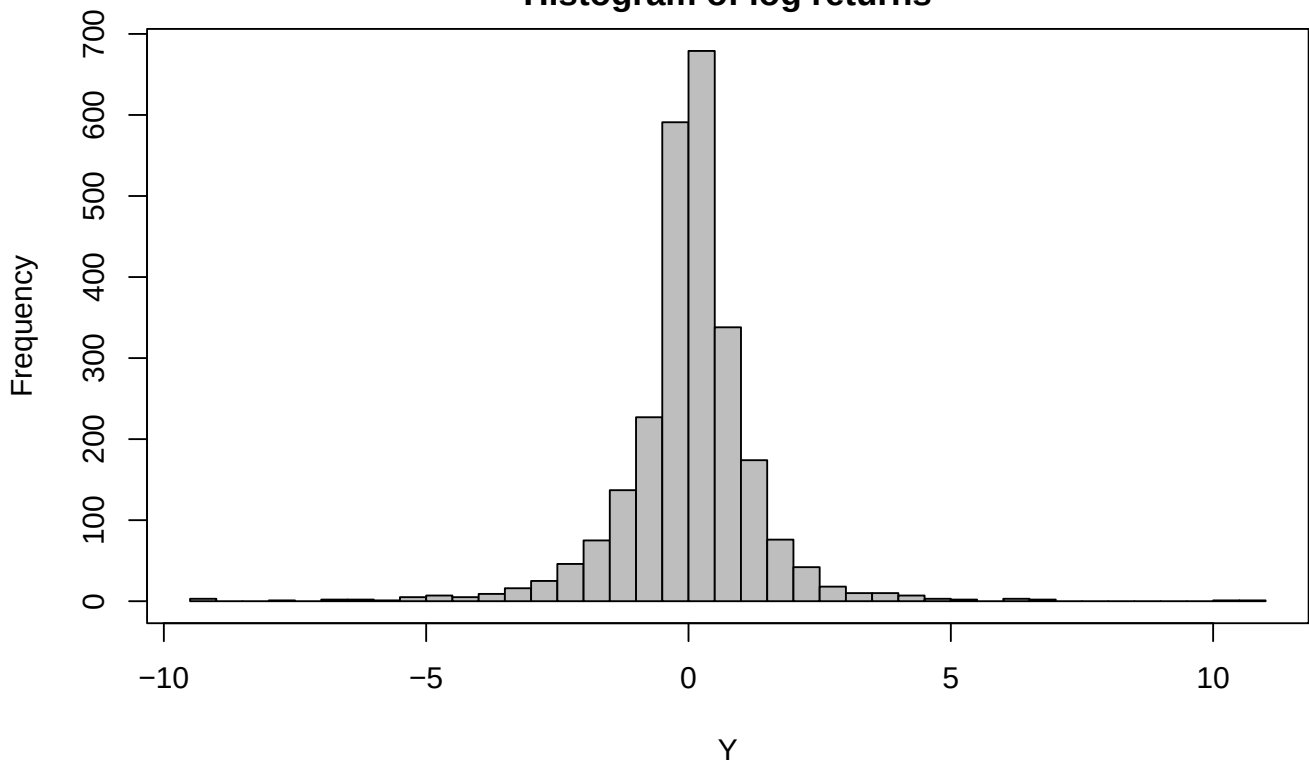
```
Y<-100*diff(log(X))
par(mar=c(4,4,2,0))
plot(Tv[-1],Y,type='l',main='S & P 500 log returns',xlab='Date')
```

S & P 500 log returns



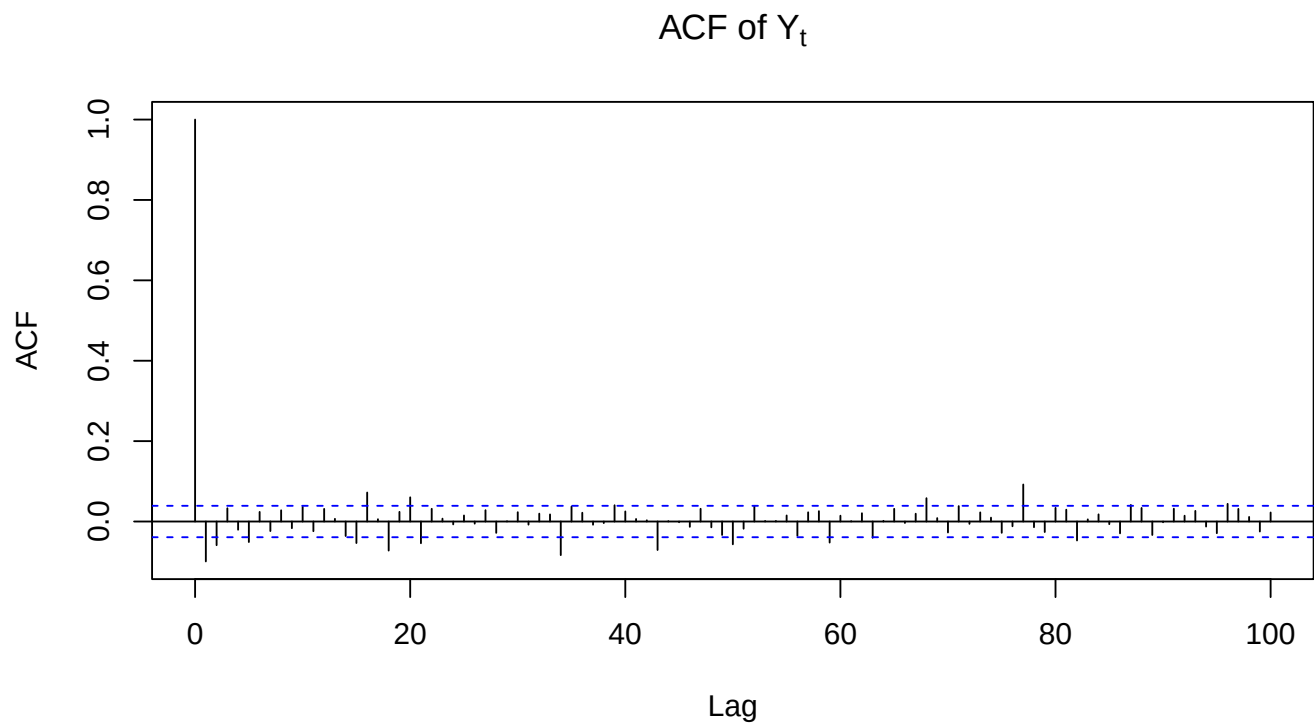
```
hist(Y,nclass=50,col='gray',main='Histogram of log returns');box()
```

Histogram of log returns

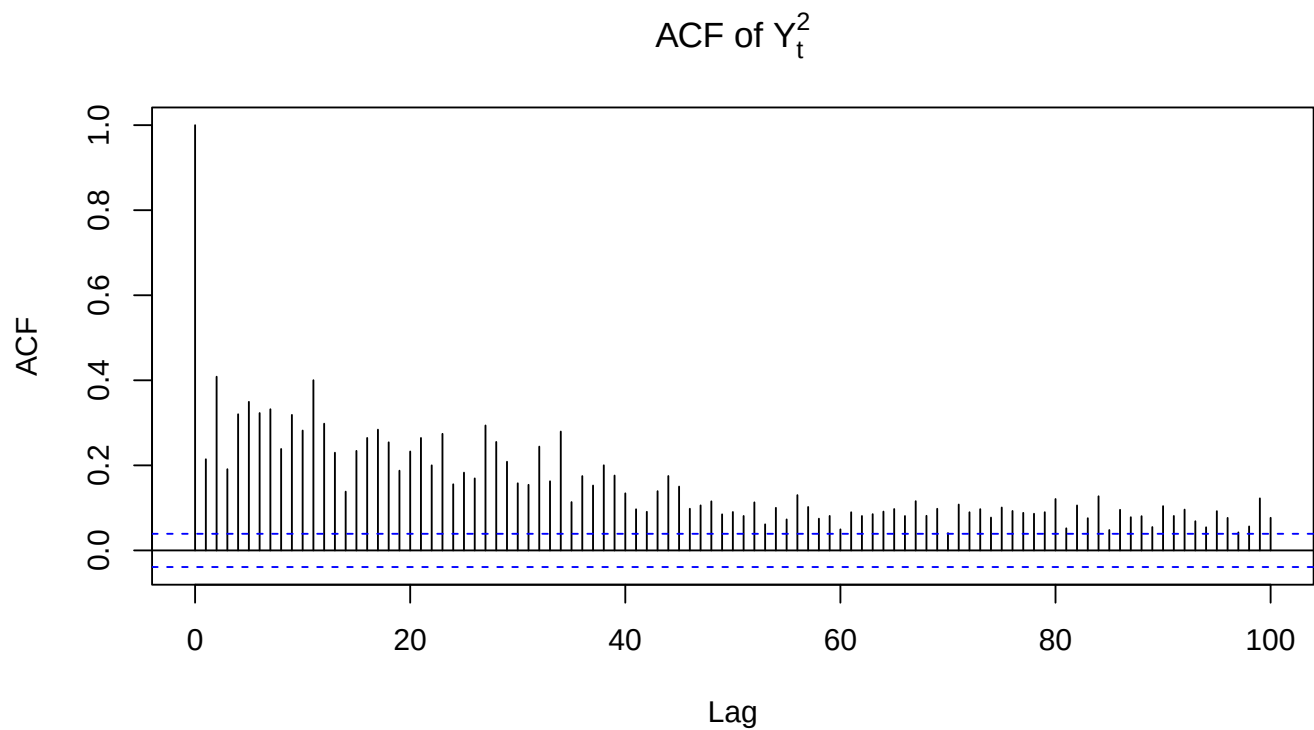


The histogram indicates a *leptokurtic* (or heavy-tailed) marginal distribution for the log returns.

```
par(mar=c(4,4,4,0))
acf(Y,lag.max=100,main=expression(paste('ACF of ',Y[t])))
```



```
acf(Y^2,lag.max=100,main=expression(paste('ACF of ',Y[t]^2)))
```



The log returns series is uncorrelated, but the squared series has significant autocorrelation.

The ARCH(1) and the GARCH(1,1) models can be used.

```
mydata<-data.frame(Y)
fit.10<-garchFit(Y~garch(1,0),data=mydata,trace=F) #ARCH(1)
show(fit.10)

+
+ Title:
+   GARCH Modelling
+
+ Call:
+   garchFit(formula = Y ~ garch(1, 0), data = mydata, trace = F)
+
+ Mean and Variance Equation:
+   data ~ garch(1, 0)
+ <environment: 0x000001f1e0a576d8>
+   [data = mydata]
+
+ Conditional Distribution:
+   norm
+
+ Coefficient(s):
+       mu      omega    alpha1
+ 0.052625  1.049464  0.454685
+
+ Std. Errors:
+   based on Hessian
+
+ Error Analysis:
+
+      Estimate  Std. Error  t value      Pr(>|t|)
+ mu      0.05262    0.02105    2.500      0.0124 *
+ omega   1.04946    0.04259   24.639 <0.0000000000000002 ***
+ alpha1  0.45469    0.05340    8.514 <0.0000000000000002 ***
+ ---
+ Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
+
+ Log Likelihood:
+  -4029.679      normalized:  -1.600349
+
+ Description:
+   Fri Apr 11 09:54:18 2025 by user: David Stephens

fit.11<-garchFit(Y~garch(1,1),data=mydata,trace=F) #GARCH(1,1)
show(fit.11)

+
+ Title:
+   GARCH Modelling
+
+ Call:
+   garchFit(formula = Y ~ garch(1, 1), data = mydata, trace = F)
+
+ Mean and Variance Equation:
+   data ~ garch(1, 1)
+ <environment: 0x000001f1dc376990>
+   [data = mydata]
+
+ Conditional Distribution:
+   norm
+
+ Coefficient(s):
+       mu      omega    alpha1    beta1
```

```

+ 0.062678 0.021065 0.128384 0.856552
+
+ Std. Errors:
+ based on Hessian
+
+ Error Analysis:
+      Estimate Std. Error t value      Pr(>|t|)
+ mu      0.062678  0.015057   4.163    0.0000314478 ***
+ omega   0.021065  0.003832   5.497    0.0000000387 ***
+ alpha1  0.128384  0.014623   8.780 < 0.0000000000000002 ***
+ beta1   0.856552  0.014474  59.180 < 0.0000000000000002 ***
+ ---
+ Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
+
+ Log Likelihood:
+ -3417.046      normalized: -1.357048
+
+ Description:
+ Fri Apr 11 09:54:18 2025 by user: David Stephens

c(fit.10@fit$ics[1],fit.11@fit$ics[1]) #AIC comparison

+      AIC      AIC
+ 3.203081 2.717272

```

Here the GARCH(1,1) model is preferred, and we have

$$X_t = 0.0210647 + 0.1283842Y_{t-1}^2 + 0.8565519X_{t-1}$$

as the maximum likelihood fitted model. Here we have

$$\hat{\alpha}_1 + \hat{\beta}_1 = 0.1283842 + 0.8565519 = 0.9849361 < 1$$

We can carry out an analysis using the stochastic volatility model. Here we use the MCMC functionality in the stochvol package.

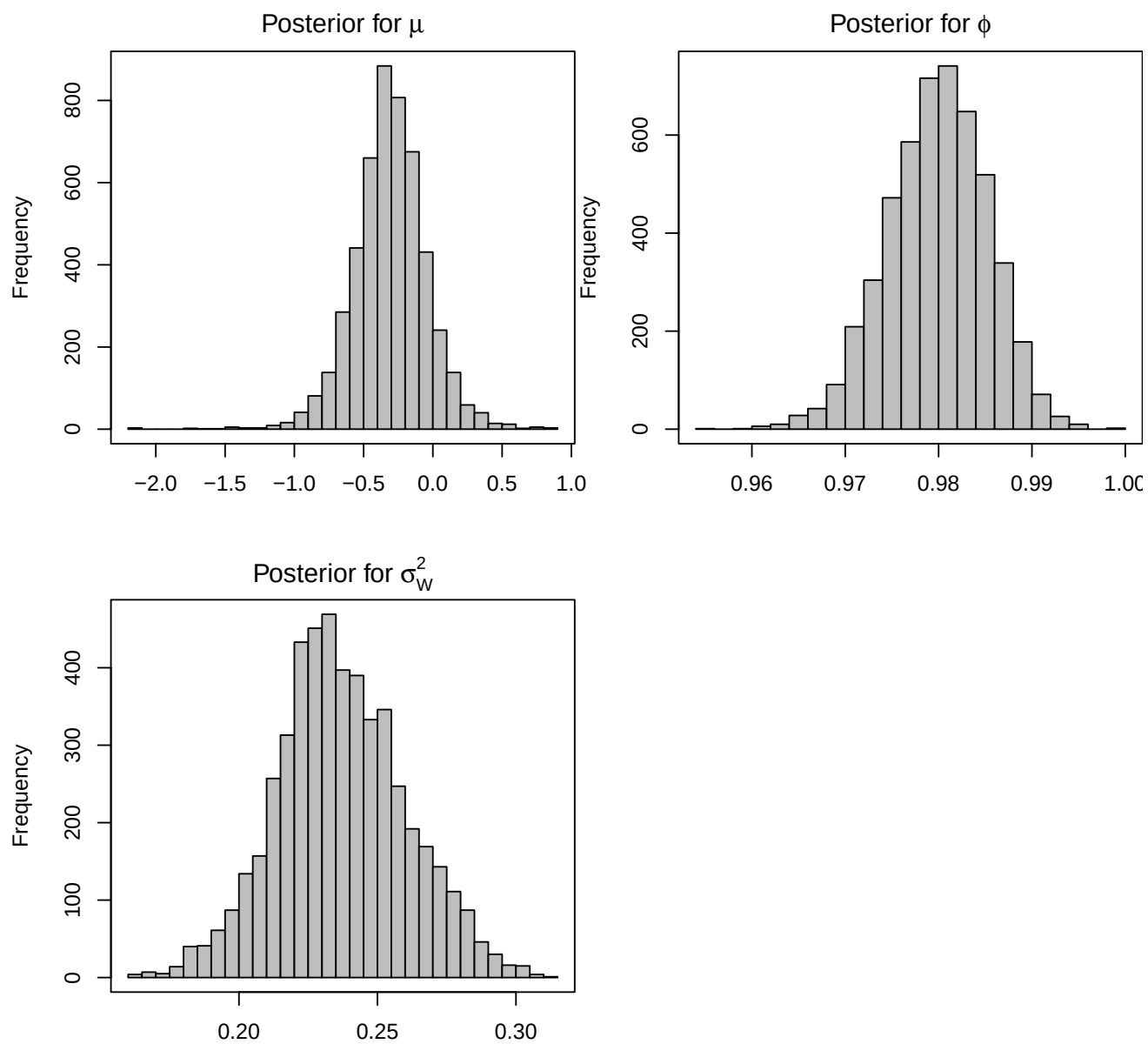
```

mcmc.draws <- svsample(Y, draws = 5000, burnin = 1000, priormu = c(0, 10),
                      priorphi = c(20, 1.2), priorsigma = 100)

+ Argument 'y' (data vector) contains values very close to zero. I am applying an offset constant
+ of size 0.000129092224403547 to do the auxiliary mixture sampling. If you want to avoid this,
+ you might consider de-meaning the returns before calling this function.
+ Done!
+ Summarizing posterior draws...

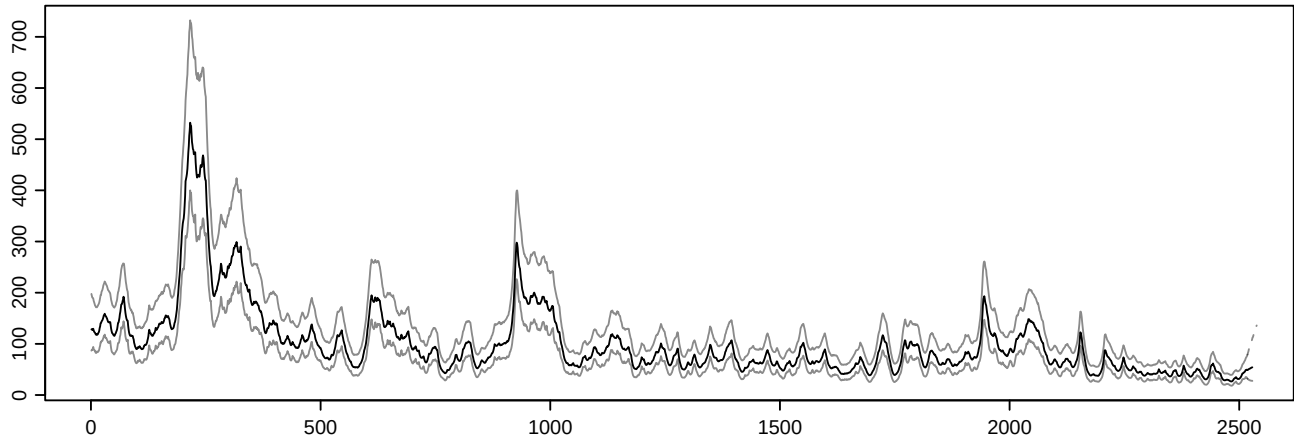
para<-data.matrix(mcmc.draws$para)
par(mfrow=c(2,2),mar=c(4,4,2,0))
hist(para[,1],col='gray',nclass=30,xlab='',
      main=expression(paste('Posterior for ',mu)))
box()
hist(para[,2],col='gray',nclass=30,xlab='',
      main=expression(paste('Posterior for ',phi)))
box()
hist(para[,3],col='gray',nclass=30,xlab='',
      main=expression(paste('Posterior for ',sigma[W]^2)))
box()

```

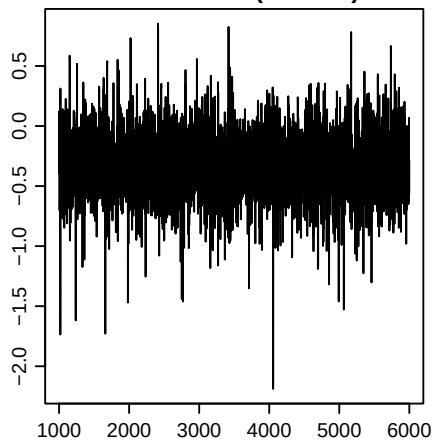


```
fore <- predict(mcmc.draws, 20)
plot(mcmc.draws, forecast = fore)
```

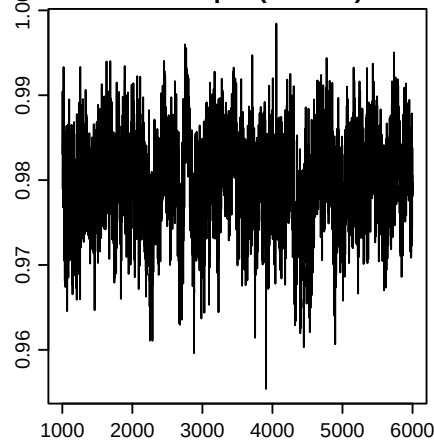
Estimated volatilities in percent (5% / 50% / 95% posterior quantiles)



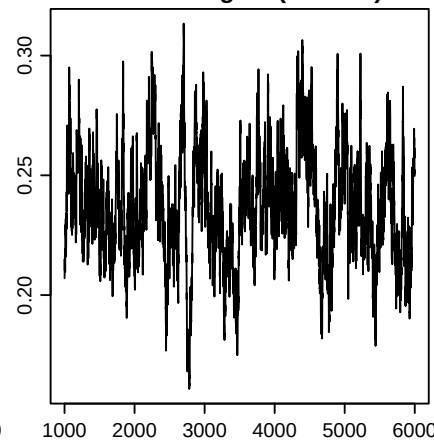
Trace of μ (thin = 1)



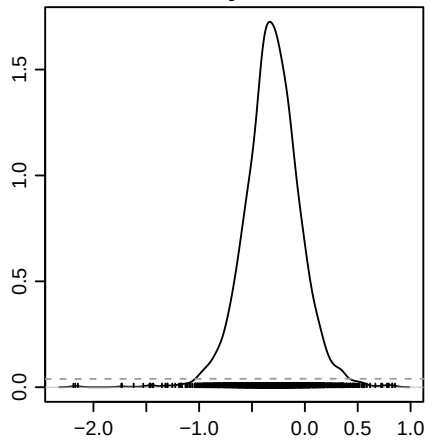
Trace of ϕ (thin = 1)



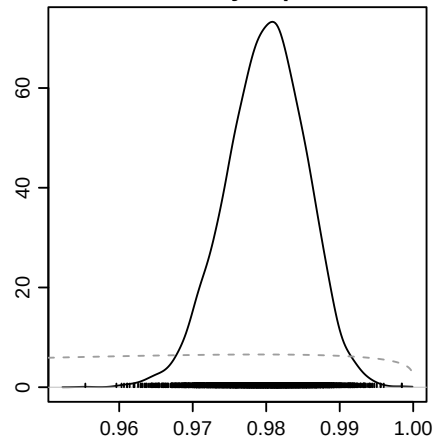
Trace of σ (thin = 1)



Density of μ



Density of ϕ



Density of σ

