

# Class 2: Stochastic volatility models and heteroskedasticity

Andrew Parnell  
`andrew.parnell@ucd.ie`



# Learning outcomes

- ▶ Learn how to model changing variance in a time series
- ▶ Understand how to fit ARCH, GARCH and SVM models in JAGS
- ▶ Know how to compare and plot the output from these models

# General principles of models for changing variance

- ▶ So far we have looked at models where the mean changes but the variance is constant:

$$y_t \sim N(\mu_t, \sigma^2)$$

- ▶ In this module we look at methods where instead:

$$y_t \sim N(\alpha, \sigma_t^2)$$

- ▶ These are:
  - ▶ Autoregressive Conditional Heteroskedasticity (ARCH)
  - ▶ Generalised Autoregressive Conditional Heteroskedasticity (GARCH)
  - ▶ Stochastic Volatility Models (SVM)
- ▶ They follow the same principles as ARIMA, but work on the standard deviations or variances instead of the mean
- ▶ forecast doesn't include any of these models so we'll use JAGS. There are other R packages to fit these models

## Extension 1: ARCH

- ▶ An ARCH(1) Model has the form:

$$\sigma_t^2 = \gamma_1 + \gamma_2 \epsilon_{t-1}^2$$

where  $\epsilon_t$  is the residual, just like an MA model

- ▶ Note that  $\epsilon_t = y_t - \alpha$  so the above can be re-written as:

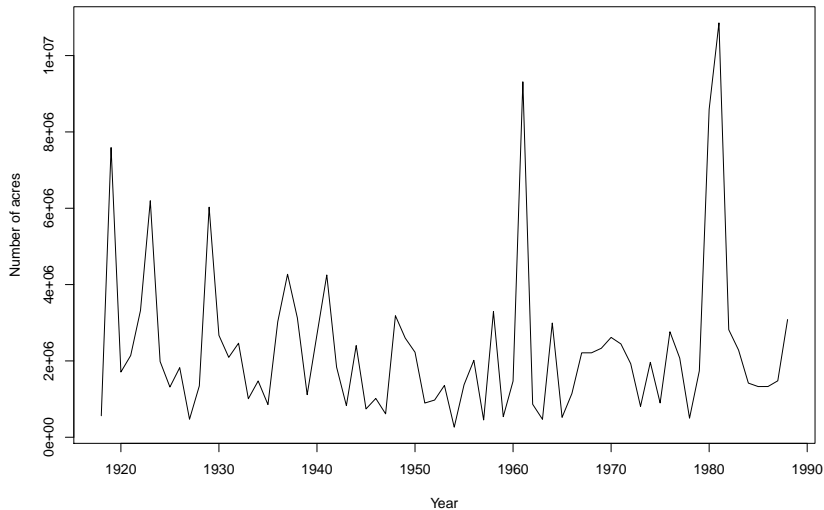
$$\sigma_t^2 = \gamma_1 + \gamma_2 (y_{t-1} - \alpha)^2$$

- ▶ The variance at time  $t$  thus depends on the previous value of the series (hence the autoregressive in the name)
- ▶ The residual needs to be squared to keep the variance positive.
- ▶ The parameters  $\gamma_1$  and  $\gamma_2$  also need to be positive, and usually  $\gamma_1 \sim U(0, 1)$

# JAGS code for ARCH models

```
model_code = '  
model  
{  
  # Likelihood  
  for (t in 1:T) {  
    y[t] ~ dnorm(alpha, sigma[t]^-2)  
  }  
  sigma[1] ~ dunif(0, 3)  
  for(t in 2:T) {  
    sigma[t] <- sqrt(gamma_1 + gamma_2 * pow(y[t-1] - alpha, 2))  
  }  
  
  # Priors  
  alpha ~ dnorm(0.0, 100^-2)  
  gamma_1 ~ dunif(0, 100)  
  gamma_2 ~ dunif(0, 100)  
}
```

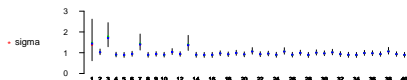
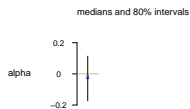
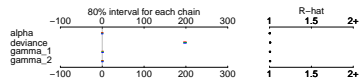
## Reminder: forest fires data



# ARCH(1) applied to forest fires data

```
plot(ff_run)
```

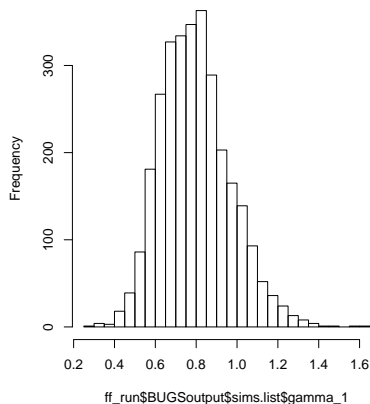
Bugs model at "5", fit using jags, 3 chains, each with 2000 iterations (first 1000 discarded)



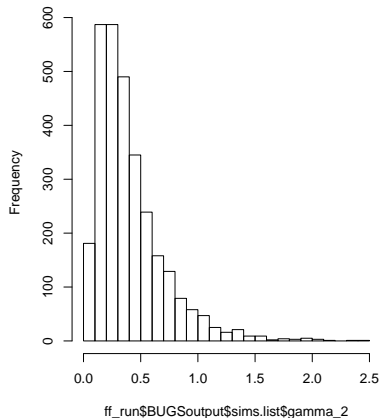
# Plot the ARCH parameters

```
par(mfrow=c(1,2))  
hist(ff_run$BUGSoutput$sims.list$gamma_1, breaks=30)  
hist(ff_run$BUGSoutput$sims.list$gamma_2, breaks=30)
```

Histogram of ff\_run\$BUGSoutput\$sims.list\$gamma\_1



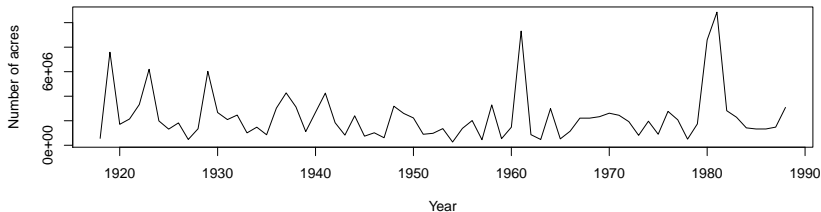
Histogram of ff\_run\$BUGSoutput\$sims.list\$gamma\_2



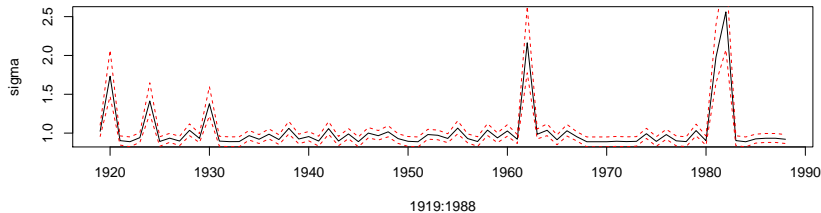


# Plot the posterior standard deviations

**Time Series of forest fires**



**Posterior standard deviation**



## From ARCH to GARCH

- ▶ The Generalised ARCH model works by simply adding the previous value of the variance, as well as the previous value of the observation
- ▶ The GARCH(1,1) model thus has:

$$\sigma_t^2 = \gamma_1 + \gamma_2(y_{t-1} - \alpha)^2 + \gamma_3\sigma_{t-1}^2$$

- ▶ Strictly speaking  $\gamma_1 + \gamma_2 < 1$  though like the stationarity conditions in ARIMA models we can relax this assumption and see if the data support it
- ▶ It's conceptually easy to extend to general GARCH(p,q) models which add in extra previous lags

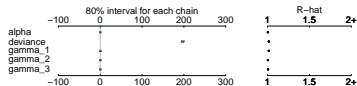
## Example of using the GARCH(1,1) model

```
model_code = '  
model  
{  
  # Likelihood  
  for (t in 1:T) {  
    y[t] ~ dnorm(alpha, sigma[t]^2)  
  }  
  sigma[1] ~ dunif(0,10)  
  for(t in 2:T) {  
    sigma[t] <- sqrt(gamma_1 + gamma_2 * pow(y[t-1] - alpha, 2)  
                    + gamma_3 * pow(sigma[t-1], 2))  
  }  
  # Priors  
  alpha ~ dnorm(0, 10^-2)  
  gamma_1 ~ dunif(0, 10)  
  gamma_2 ~ dunif(0, 10)  
  gamma_3 ~ dunif(0, 10)  
}  
'
```

# Using the forest fire data again

```
plot(ff_run_2)
```

Bugs model at "6", fit using jags, 3 chains, each with 2000 iterations (first 1000 discarded)

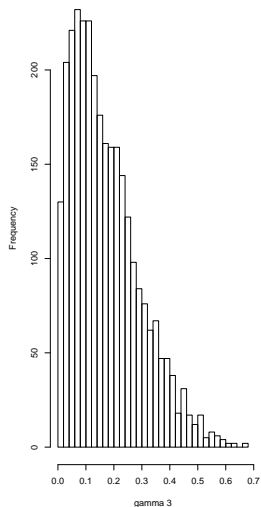
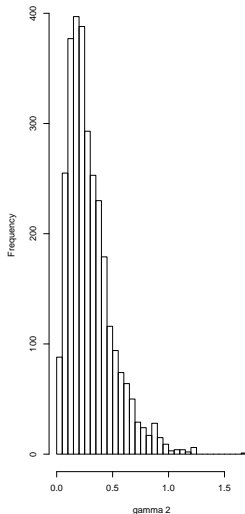
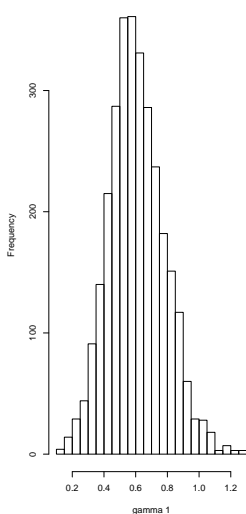


medians and 80% intervals



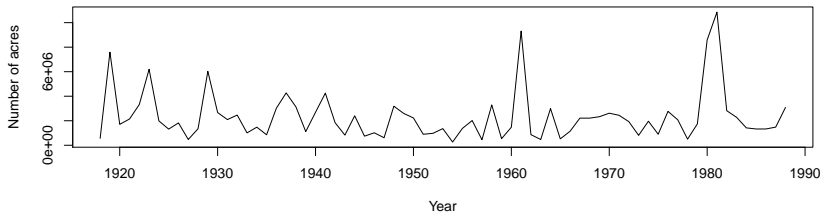
# Looking at the GARCH parameters

histogram of ff\_run\_2\$BUGSoutput\$sims.list\$gam1 histogram of ff\_run\_2\$BUGSoutput\$sims.list\$gam2 histogram of ff\_run\_2\$BUGSoutput\$sims.list\$gam3

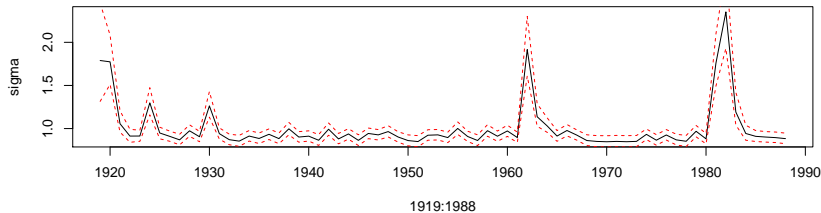


# Posterior standard deviations over time

**Time Series of forest fires**



**Posterior standard deviation**



## Compare with DIC

```
with(r_1, print(c(DIC, pD)))
```

```
## [1] 201.069002    3.731416
```

```
with(r_2, print(c(DIC, pD)))
```

```
## [1] 201.332652    4.526875
```

- ▶ Suggests not much difference between the models

# Stochastic Volatility Modelling

- ▶ Both ARCH and GARCH propose a deterministic relationship for the current variance parameter
- ▶ By contrast a Stochastic Volatility Model (SVM) models the variance as its own *stochastic process*
- ▶ SVMs, ARCH and GARCH are all closely linked if you go into the bowels of the theory
- ▶ The general model structure is often written as:

$$y_t \sim N(\alpha, \exp(h_t))$$

$$h_t \sim N(\mu + \phi(h_{t-1} - \mu), \sigma^2)$$

- ▶ You can think of an SVM being like a GLM but with a log link on the variance parameter

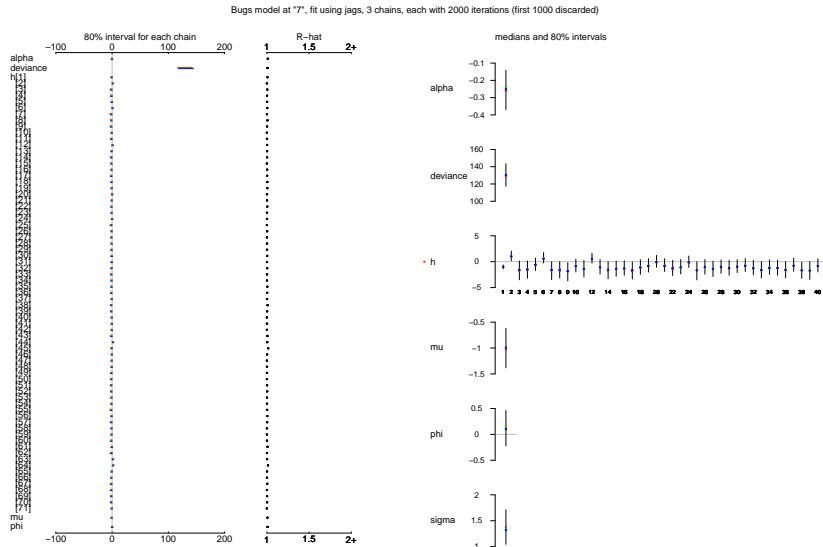


## JAGS code for the SVM model

```
model_code = '  
model  
{  
  # Likelihood  
  for (t in 1:T) {  
    y[t] ~ dnorm(alpha, sigma_h[t]^-2)  
    sigma_h[t] <- sqrt(exp(h[t]))  
  }  
  h[1] <- mu  
  for(t in 2:T) {  
    h[t] ~ dnorm(mu + phi * (h[t-1] - mu), sigma^-2)  
  }  
  
  # Priors  
  alpha ~ dnorm(0, 100^-2)  
  mu ~ dnorm(0, 100^-2)  
  phi ~ dunif(-1, 1)  
  sigma ~ dunif(0,100)  
}  
'
```

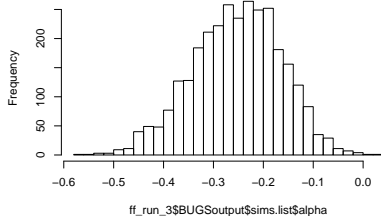
# Example of SVMs and comparison of DIC

```
plot(ff_run_3)
```

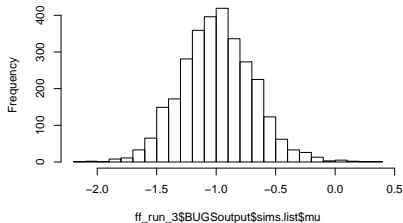


# Look at all the parameters

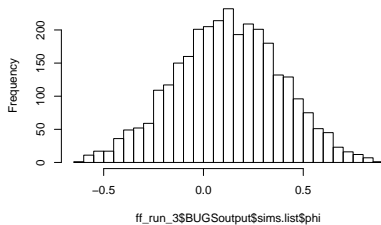
Histogram of `ff_run_3$BUGSoutput$sims.list$alpha`



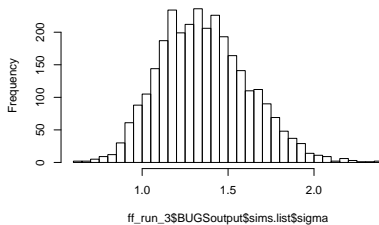
Histogram of `ff_run_3$BUGSoutput$sims.list$mu`



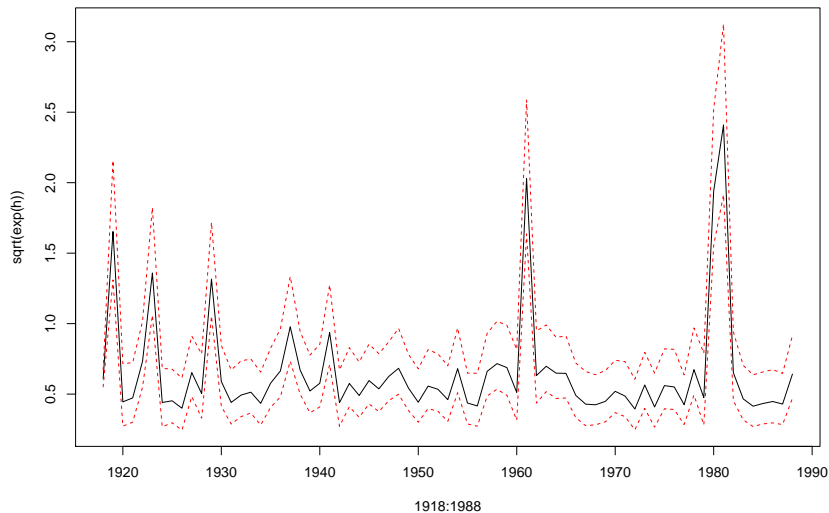
Histogram of `ff_run_3$BUGSoutput$sims.list$phi`



Histogram of `ff_run_3$BUGSoutput$sims.list$sigma`



# Plot of $\sqrt{\exp(h)}$



## Comparison with previous models

```
with(r_1, print(c(DIC, pD)))
```

```
## [1] 201.069002    3.731416
```

```
with(r_2, print(c(DIC, pD)))
```

```
## [1] 201.332652    4.526875
```

```
with(r_3, print(c(DIC, pD)))
```

```
## [1] 180.5647    50.3189
```

Much better fit, despite many extra parameters due to  $h!$

# Summary

- ▶ We know that ARCH extends the ARIMA idea into the variance using the previous values of the series
- ▶ We know that GARCH extends ARCH with previous values of the variance too
- ▶ We know that SVMs give the variance its own stochastic process
- ▶ We can combine these new models with all the techniques we have previously learnt