Not All White Noise Are Created Equal

Two different types of white noise:
- **strict white noise (SWN)** — sequence of iid random variables
- **uncorrelated white noise (UWN)** — sequence of uncorrelated, but not necessarily independent, white noise

Certainly **SWN** ⇒ **UWN**.
- However **SWN** is uninformative whereas **UWN** can be informative.
Comparison of IID $N(0, 1)$ with a stationary GARCH(1,1)

Modeling Volatility

Properties of ARCH/GARCH models:
- Primary interest is in modeling changes in variance
- Provides improved estimations of the local variance (volatility)
- Not necessarily concerned with better forecasts
- Can be integrated into ARMA models
- Useful in modeling financial time series

Returns

Let $y_t = \log(x_t)$, then

\[ u_t = \Delta y_t = y_t - y_{t-1} = \log \left( \frac{x_t}{x_{t-1}} \right) = \log \left( 1 + \frac{x_t - x_{t-1}}{x_{t-1}} \right) \approx x_t - x_{t-1} \]

Returns in financial time series tend to have highly volatile periods clustered together.

S&P 500 Returns
Consider the following ARCH(1) model:

\[ y_t = \sigma_t \varepsilon_t \]
\[ \sigma_t^2 = \alpha_0 + \alpha_1 y_{t-1}^2 \]

where \( \varepsilon_t \overset{iid}{\sim} N(0, 1) \) and \( 0 < \alpha_1 < 1 \).

- The process \( y_t \) is stationary, but the conditional variance of \( y_t \) varies in time.
- Note that \( y_t \) given \( y_{t-1} \) has the distribution

\[ y_t | y_{t-1} \sim N \left( 0, \alpha_0 + \alpha_1 y_{t-1}^2 \right) \]

\( y_t^2 \) resembles an AR(1) process.

The defining equations of \( y_t \) are rewritten as

\[ y_t^2 = \sigma_t^2 \varepsilon_t^2 \]
\[ \alpha_0 + \alpha_1 y_{t-1}^2 = \sigma_t^2 \]

Subtracting gives

\[ y_t^2 - (\alpha_0 + \alpha_1 y_{t-1}^2) = \sigma_t^2 \varepsilon_t^2 - \sigma_t^2 \]

which is equivalent to

\[ y_t^2 = \alpha_0 + \alpha_1 y_{t-1}^2 + v_t \]

where \( v_t = \sigma_t^2 (\varepsilon_t^2 - 1) \).
Analysis of US GNP

Fit an AR(1) model to the US GNP and test

```r
library(tseries)

gnp96 = read.table("mydata/gnp96.dat")

gnpr = diff(log(gnp96[,2]))  # recall phi1 = .347

gnpr.ar = ar.mle(gnpr, order.max=1)

y = gnpr.ar$resid[2:length(gnpr)]  # first resid is NA

arch.y = garch(y, order=c(0,1))

summary(arch.y)
```

ARCH(m), GARCH(m,r)

**ARCH(m):**

\[ y_t = \sigma_t \varepsilon_t \]

\[ \sigma_t^2 = \alpha_0 + \sum_{j=1}^{m} \alpha_j y_{t-j}^2 \]

**GARCH(m,r):**

\[ y_t = \sigma_t \varepsilon_t \]

\[ \sigma_t^2 = \alpha_0 + \sum_{j=1}^{m} \alpha_j y_{t-j}^2 + \sum_{j=1}^{r} \beta_j \sigma_{t-j}^2 \]

Analysis of S&P 500 returns

Fit GARCH(1,1) to the data.

```r
library(tseries)

nyse = scan("mydata/nyse.dat")

nyse.g = garch(nyse, order=c(1,1))

summary(nyse.g)

u = predict(nyse.g)

plot(nyse, type="l", xlab="Time", ylab="NYSE Returns")

lines(u[,1], col="blue", lty="dashed")

lines(u[,2], col="blue", lty="dashed")
```
Model:
GARCH(1,1)

Residuals:
Min 1Q Median 3Q Max
-8.66460 -0.45228 0.08785 0.59918 4.07508

Coefficient(s):

| Estimate | Std. Error | t value | Pr(>|t|) |
|----------|------------|---------|----------|
| a0       | 6.552e-06  | 6.761e-07 | 9.691    | <2e-16 *** |
| a1       | 1.118e-01  | 4.056e-03 | 27.554   | <2e-16 *** |
| b1       | 8.086e-01  | 1.292e-02 | 62.566   | <2e-16 *** |

---

Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Jarque Bera Test
data: Residuals
X-squared = 3983.873, df = 2, p-value < 2.2e-16

Box-Ljung test
data: Squared.Residuals
X-squared = 1.5874, df = 1, p-value = 0.2077