The GARCH Model

Full Name:

The **G**eneralized **A**utoregressive **C**onditional **H**eteroskedasticity Model

Mathematical Notation:

$$Var(y_t \mid y_{t-1}) = \Omega + \alpha_1 \varepsilon_{t-1}^2 + \boxed{\beta_1} \sigma_{t-1}^2$$

$Var(y_t \mid y_{t-1})$	The variance today is conditional on the values of the variable yesterday
Ω	Constant
α_1	Numeric coefficient for the squared residual for the past period
ε_{t-1}^2	Squared residual for the past period
β_1	Numeric coefficient for the conditional variance from last period
σ_{t-1}^2	Conditional variance from last period

Description:

As the name suggests, the GARCH is just the generalized version of the ARCH model.

This generalization is expressed in including past variances as well as past squared residuals to estimate current (and subsequent) variances.

The generalization comes from the fact that including a single past variance would (in theory) contain in itself the explanatory power of all other previous squared error terms.

It serves as a sort of ARMA equivalent to the ARCH, where we're including both past values and past errors (albeit squared).

The GARCH Model

Implementation of the Simple Model in Python:

The library the *arch_model* method comes from

The method we are importing

The type of volatility model we are assuming for the volatility equation

```
from arch import arch_model
```

```
model_garch_1_1 = arch_model(df.returns[1:], mean = "Constant", vol = "GARCH", p = 1, q = 1)
```

The variable storing the model characteristics that we will fit later

The time series, whose volatility we wish to analyse

The type of model we are assuming for the mean equation

The orders of the model

*For an GARCH(p,q) model, simply change p from 1 to q and the q from 1 to p.