

# *Time Series Models: A Guide for Practitioners*

Yan Zeng

Version 1.0.2, last revised on 2015-02-17.

## **Abstract**

A user's guide on time series models, based on Ruppert [1].

## **Contents**

<b>1</b>	<b>Test of Stationarity</b>	<b>2</b>
<b>2</b>	<b>AR(1) model</b>	<b>2</b>
<b>3</b>	<b>AR(<math>p</math>) model</b>	<b>3</b>
<b>4</b>	<b>MA Processes</b>	<b>3</b>
<b>5</b>	<b>ARMA Model</b>	<b>4</b>
<b>6</b>	<b>ARIMA Model</b>	<b>4</b>
<b>7</b>	<b>Unit Root Tests</b>	<b>5</b>
<b>8</b>	<b>Forecasting</b>	<b>5</b>
<b>9</b>	<b>Partial Autocorrelation Coefficients</b>	<b>6</b>
<b>10</b>	<b>Cointegration</b>	<b>7</b>
<b>11</b>	<b>Long-Memory Processes</b>	<b>7</b>
<b>12</b>	<b>AR(1)/ARCH(1) Model</b>	<b>8</b>
<b>13</b>	<b>ARIMA(<math>p_A, d, q_A</math>)/GARCH(<math>p_G, q_G</math>) Models</b>	<b>9</b>
<b>14</b>	<b>Regression with ARMA/GARCH Errors</b>	<b>10</b>

# 1 Test of Stationarity

- Visual inspection via time series plot (stationarity implies *mean-reversion* phenomenon).
- Plot of autocorrelation functions (ACF).

Most statistical software will plot a sample ACF with *test bounds*. These bounds are used to test the null hypothesis that an autocorrelation coefficient is 0. The null hypothesis is rejected if the sample autocorrelation is outside the bounds. The usual level of the test is 0.05.

- Hypothesis testing via a simultaneous test (Ljung-Box test).

A *simultaneous test* is one that tests whether a group of null hypotheses are all true versus the alternative that at least one of them is false. The null hypothesis of the Ljung-Box test is  $H_0$ :

$$\rho(1) = \rho(2) = \dots = \rho(K) = 0 \text{ for some } K, \text{ say } K = 5 \text{ or } 10.$$

The Ljung-Box test was implemented using R's `Box.test` function.  $K$  is called `lag` when `Box.test` is called and `df` in the output.

- Hypothesis testing via the Durbin-Watson test.

The null hypothesis of the Durbin-Watson test is that the first  $p$  autocorrelation coefficients are all 0, where  $p$  can be selected by the user. In the R function `durbin.watson` in the `car` package,  $p$  is called `max.lag` and has a default value of 1. The  $p$ -value is computed by `durbin.watson` using bootstrapping (The calculated  $p$ -value could have substantial random variation due to bootstrapping with the default of  $B = 1000$  resamples. Using a larger number of resamples will compute the  $p$ -value with more accuracy.). The `lmtest` package of R has another function, `dwttest`, that computes the Durbin-Watson test, but only with  $p = 1$ . `dwttest` uses either a normal approximation (default) or an exact algorithm to calculate the  $p$ -value.

## 2 AR(1) model

**Properties of an AR(1) process.** An AR(1) process  $Y_t - \mu = \phi(Y_{t-1} - \mu) + \epsilon_t$  ( $t = 1, 2, \dots$ ) is weakly stationary if and only if  $|\phi| < 1$  and  $Y_0$  satisfies

$$E[Y_0] = \mu, \text{ Var}(Y_0) = \frac{\sigma_\epsilon^2}{1 - \phi^2}. \tag{1}$$

In this case, we have for any  $t$  and  $h$ ,

$$\begin{aligned} E(Y_t) &= \mu \\ \gamma(0) = \text{Var}(Y_t) &= \frac{\sigma_\epsilon^2}{1 - \phi^2} \\ \gamma(h) = \text{Cov}(Y_t, Y_{t+h}) &= \frac{\sigma_\epsilon^2 \phi^{|h|}}{1 - \phi^2} \\ \rho(h) = \text{Corr}(Y_t, Y_{t+h}) &= \phi^{|h|}. \end{aligned}$$

Moreover, the process has the *infinite moving average* [MA( $\infty$ )] representation:

$$Y_t = \mu + \epsilon_t + \phi\epsilon_{t-1} + \phi^2\epsilon_{t-2} + \dots = \mu + \sum_{h=0}^{\infty} \phi^h \epsilon_{t-h}.$$

If  $Y_0$  does not satisfy condition (1), but we still have  $|\phi| < 1$ , then  $(Y_t)_{t=1}^{\infty}$  is not stationary but converges to the stationary distribution as  $t \rightarrow \infty$ . If  $|\phi| \geq 1$ , then  $(Y_t)_{t=1}^{\infty}$  is either explosive or is a random walk and cannot stationary.<sup>1</sup>

---

<sup>1</sup>It is assumed the observations start at  $Y_1$  so that  $Y_0$  is not available.

The ACF  $\rho(h)$  of an AR(1) process depends only on one parameter ( $\phi$ ) and has only a limited range of shapes (the magnitude of its ACF decays geometrically to zero). If the sample ACF of the data does not behave in one of these ways, then an AR(1) model is unsuitable. The remedy is to use more AR parameters, to switch to another class of models such as the moving average (MA) or autoregressive moving average (ARMA) models.

**Residuals and Model Checking.** To verify that an AR(1) process is a suitable model, we can check on its residuals; any autocorrelation in the residuals is evidence against the assumption of an AR(1) process. Tests include ACF plot and the Ljung-Box test of the residuals (see Section 1). Any residual ACF value outside the test bounds is significantly different from 0 at the 0.05 level. The danger here is that some sample ACF values will be significant merely by chance, and to guard against this danger, one can use the Ljung-Box test that *simultaneously* tests that all autocorrelations up to a specified lag are zero.<sup>2</sup>

**Estimation of AR(1) processes.** Assuming the residuals are i.i.d. normal and  $(Y_t)_{t=1}^\infty$  is Markov, we can use the *maximum likelihood estimator* (MLE) or the *conditional least-square estimator* to estimate  $\mu$ ,  $\phi$ ,  $\sigma_\epsilon$ . The default method for the function `arima` in R is to use the conditional least-squares estimates as starting values for maximum likelihood. The MLE is returned. See Ruppert [1, page 218] for details.

### 3 AR( $p$ ) model

An AR( $p$ ) process has the form

$$Y_t - \mu = \phi_1(Y_{t-1} - \mu) + \cdots + \phi_p(Y_{t-p} - \mu) + \epsilon_t,$$

where  $\epsilon_t$ 's are weak white noise process  $WN(0, \sigma_\epsilon^2)$ . It can be shown that  $\{1 - (\phi_1 + \cdots + \phi_p)\} > 0$  for a stationary process. The ACFs of AR( $p$ ) processes with  $p > 1$ , although having complicated formulas, can be computed and plotted using R's `ARMAacf` function.

Most of the concepts for AR(1) models generalize easily to AR( $p$ ) models. The conditional least squares or maximum likelihood estimators can be calculated using software such as R's `arima` function.

Residual autocorrelation can be detected by examining the sample ACF of the residuals and using the Ljung-Box test. Many statistical software packages have functions to automate the search for the AR model that optimizes AIC or other criteria, e.g. the `auto.arima` function in R's `forecast` package.

One problem with AR models is that they often need a rather large value of  $p$  to fit a data set. When AR models are not parsimonious, the remedy is to use an MA or ARMA model.

### 4 MA Processes

The idea behind AR processes is to feed past data back into the current value of the process. The effect is to have at least some correlation at *all* lags. Sometimes data show correlation at only short lags, while AR processes do not behave this way and do not provide a parsimonious fit. A useful alternative to an AR model is a moving average (MA) model. E.g., the MA(1) process is

$$Y_t - \mu = \epsilon_t + \theta\epsilon_{t-1},$$

---

<sup>2</sup>When the Ljung-Box test is applied to residuals, a correction is needed to account for the use of  $\hat{\phi}$  in place of the unknown  $\phi$ . Some software makes this correction automatically. In R the correction is not automatic but is done by setting the `fitdf` parameter in `Box.test` to the number of parameters that were estimated, so for an AR(1) model `fitdf` should be 1.

where  $\epsilon_t$ 's are  $\text{WN}(0, \sigma_\epsilon^2)$ . One can show that

$$\begin{aligned} E(Y_t) &= \mu, \\ \text{Var}(Y_t) &= \sigma_\epsilon^2(1 + \theta^2), \\ \gamma(1) &= \theta\sigma_\epsilon^2, \\ \gamma(h) &= 0 \text{ if } |h| > 1, \\ \rho(1) &= \frac{\theta}{1 + \theta^2}, \\ \rho(h) &= 0 \text{ if } |h| > 1. \end{aligned}$$

The MA( $q$ ) process is

$$Y_t = \mu + \epsilon_t + \theta_1\epsilon_{t-1} + \cdots + \theta_q\epsilon_{t-q}.$$

One can show that  $\gamma(h) = 0$  and  $\rho(h) = 0$  if  $|h| > q$ .  $\gamma(h)$  and  $\rho(h)$  when  $|h| \leq q$  can be computed in R by the function `ARMAacf`.

MA( $q$ ) models can be fit easily using, for example, the `arima` function in R. The Ljung-Box test is applied to the residuals to check the fitted MA( $q$ ) model is adequate.

## 5 ARMA Model

The ARMA( $p, q$ ) model is

$$(Y_t - \mu) = \phi_1(Y_{t-1} - \mu) + \cdots + \phi_p(Y_{t-p} - \mu) + \epsilon_t + \theta_1\epsilon_{t-1} + \cdots + \theta_q\epsilon_{t-q}.$$

Using the backwards operator  $B$ , this can be written as

$$(1 - \phi_1B - \cdots - \phi_pB^p)(Y_t - \mu) = (1 + \theta_1B + \cdots + \theta_qB^q)\epsilon_t.$$

The ARMA(1,1) model has

$$\rho(1) = \frac{(1 + \phi\theta)(\phi + \theta)}{1 + \theta^2 + 2\phi\theta}, \quad \rho(k) = \phi\rho(k-1), \quad k \geq 2. \quad (2)$$

So, after one lag the ACF of an ARMA(1,1) process decays in the same way as the ACF of an AR(1) process with the same  $\phi$ .

The parameters of ARMA models can be estimated by maximum likelihood or conditional least-squares. The estimation methods for AR( $p$ ) models are very similar to those for AR(1) models. For MA and ARMA, because the noise terms  $\epsilon_1, \dots, \epsilon_n$  are unobserved, there are complications that are best left for advanced time series texts.

## 6 ARIMA Model

Often the first or perhaps second differences of nonstationary time series are stationary. A time series  $Y_t$  is said to be an ARIMA( $p, d, q$ ) process if  $\Delta^d Y_t$  is ARMA( $p, q$ ). We will say that a process is I( $d$ ) if it is stationary after being differenced  $d$  times; such a process is said to be “integrated to order  $d$ .”

For example, the AR(1) process is stationary and varies randomly about its mean; the integral of this process behaves much like a random walk in having no fixed level to which it reverts; and the second integral has *momentum*. Once the process starts moving upward or downward, it tends to continue in that direction. If data show momentum like this, then the momentum is an indication that  $d = 2$ .

The `arima` function in R makes the assumption that a differenced process has mean zero. The R function `auto.arima` allows a differenced process to have a nonzero mean, which is called the **drift** in the output.

## 7 Unit Root Tests

For an ARMA( $p, q$ ) process  $\{Y_t\}$  satisfying

$$(Y_t - \mu) = \phi_1(Y_{t-1} - \mu) + \cdots + \phi_p(Y_{t-p} - \mu) + \epsilon_t + \theta_1\epsilon_{t-1} + \cdots + \theta_q\epsilon_{t-q}$$

to be stationary, all roots of the polynomial

$$1 - \phi_1x - \cdots - \phi_px^p$$

have absolute values greater than one. If there is a *unit root*, then the ARMA process is nonstationary and behaves much like a random walk. The *explosive case* is when a root has an absolute value less than 1. R's `auto.arima` function automatically selects an ARIMA model and `polyroot` function can be used to check if the roots of  $1 - \hat{\phi}_1x - \cdots - \hat{\phi}_px^p$  have absolute values greater than 1.

**Unit root tests.** Unit root tests are used to decide if an *AR model* has an absolute root equal to 1. Popular tests include

- The augmented Dickey-Fuller test (`adf.test` in R's `tseries` package). The null hypothesis is that there is a unit root and the usual alternative is that the process is stationary but one can instead use the alternative that the process is explosive.
- The Phillips-Perron test (`pp.test` in R's `tseries` package). It is similar to the ADF test.
- The KPSS test (`kpss.test` in R's `tseries` package). The null hypothesis is stationarity and the alternative is a unit root, just opposite of the hypotheses for the Dickey-Fuller and Phillips-Perron tests.

**How Do Unit Root Tests Work?** The Dickey-Fuller test is based on the AR(1) model

$$Y_t = \phi Y_{t-1} + \epsilon_t. \quad (3)$$

The null hypothesis ( $H_0$ ) is  $\phi = 1$  and the alternative ( $H_1$ ) is stationarity  $\phi < 1$ . Model (3) is equivalent to

$$\Delta Y_t = \pi Y_{t-1} + \epsilon_t \quad (4)$$

where  $\pi = \phi - 1$ . Stated in terms of  $\pi$ ,  $H_0$  is  $\pi = 0$  and  $H_1$  is  $\pi < 0$ . The Dickey-Fuller test regresses  $\Delta Y_t$  on  $Y_{t-1}$  and tests  $H_0$ .

The augmented Dickey-Fuller test expands model (3) by adding a time trend and lagged values of  $\Delta Y_t$ :

$$\Delta Y_t = \beta_0 + \beta_1 t + \pi Y_{t-1} + \sum_{j=1}^p \gamma_j \Delta Y_{t-j} + \epsilon_t.$$

The hypotheses are still  $H_0 : \pi = 0$  and  $H_1 : \pi < 0$ . There are several methods for selecting  $p$ . The `adf.test` function has a default value of  $p$  equal to `trunc((length(x)-1)^(1/3))`, where  $\mathbf{x}$  is the input series.

**Automatic selection of an ARMA model.** The R function `auto.arima` can select all three parameters,  $p$ ,  $d$ , and  $q$ , for an ARIMA model. The differencing parameter  $d$  is selected using the KPSS test. If the null hypothesis of stationarity is accepted when the KPSS is applied to the original time series, then  $d = 0$ . Otherwise, the series is differenced until the KPSS accepts the null hypothesis. After that,  $p$  and  $q$  are selected using either AIC or BIC.  $AIC_c$  in `auto.arima`'s output is the value of the corrected AIC criterion:

$$AIC_c = AIC + \frac{2p(p+1)}{n-p-1}.$$

## 8 Forecasting

Most time series software packages offer functions to automate forecasting. R's `predict` function forecasts using an "object" returned by the `arima` fitting function.

To know the uncertainty of the predictions, one first computes the variance of the forecast error. Then a  $(1 - \alpha)100\%$  prediction interval is the forecast itself plus or minus the forecast error's standard deviation times  $z_{\alpha/2}$  (the normal upper quantile). The use of  $z_{\alpha/2}$  assume that  $\epsilon_1, \dots$  is Gaussian white noise.

**Forecasting using an AR(1) process**  $Y_{n+1} = \mu + \phi(Y_n - \mu) + \epsilon_{n+1}$ . The general formula for the  $k$ -step-ahead forecast is

$$\hat{Y}_{n+k} = \hat{\mu} + \hat{\phi}^k(Y_n - \hat{\mu}).$$

If  $|\hat{\phi}| < 1$ , as is true for a stationary series, then as  $k$  increases, the forecasts will converge exponentially fast to  $\hat{\mu}$ . The  $k$ -step-ahead forecasting error is

$$Y_{n+k} - \hat{Y}_{n+k} \approx \phi(Y_{n+k-1} - \hat{Y}_{n+k-1}) + \epsilon_{n+k} = \dots = \phi^{k-1}\epsilon_{n+1} + \phi^{k-2}\epsilon_{n+2} + \dots + \phi\epsilon_{n+k-1} + \epsilon_{n+k},$$

which implies  $\text{Var}(Y_{n+k} - \hat{Y}_{n+k}) = \left(\frac{1-\phi^{2k}}{1-\phi^2}\right)\sigma_\epsilon^2 \rightarrow \frac{\sigma_\epsilon^2}{1-\phi^2}$  as  $k \rightarrow \infty$ . So the variance of the forecast error does not diverge as  $k \rightarrow \infty$ , but rather the variance converges to  $\gamma(0)$ . This is an example of the general principle that for any stationary ARMA process, the variance of the forecast error converges to the marginal variance.

For a random walk where  $\phi = 1$ , the variance of the  $k$ -step-ahead forecast error is  $k\sigma_\epsilon^2$  and diverges to  $\infty$  as  $k \rightarrow \infty$ .

**Forecasting using an MA(1) process**  $Y_t - \mu = \epsilon_t - \theta\epsilon_{t-1}$ . The general formula for the  $k$ -step-ahead forecast is

$$\hat{Y}_{n+1} = \hat{\mu} - \hat{\theta}\hat{\epsilon}_n, \hat{Y}_{n+k} = \hat{\mu} \text{ for all } k > 2.$$

**Forecasting using an ARMA(1,1) process**  $Y_t - \mu = \phi(Y_{t-1} - \mu) + \epsilon_t - \theta\epsilon_{t-1}$ . The one-step-ahead forecast is

$$\hat{Y}_{n+1} = \hat{\mu} + \phi(\hat{Y}_n - \hat{\mu}) - \hat{\theta}\hat{\epsilon}_n$$

and the  $k$ -step-ahead forecast is

$$\hat{Y}_{n+k} = \hat{\mu} + \hat{\phi}(\hat{Y}_{n+k-1} - \hat{\mu}), k \geq 2.$$

For stationary ARIMA processes, the variance of the  $k$ -step-ahead forecast error variance converges to a finite value as  $k \rightarrow \infty$ , but for a nonstationary ARIMA process this variance converges to  $\infty$ .

## 9 Partial Autocorrelation Coefficients

The partial autocorrelation function (PACF) can be useful for identifying the order of an *AR process*. The  $k$ th partial autocorrelation, denoted by  $\phi_{k,k}$ , for a stationary process  $Y_t$  is the correlation between  $\bar{Y}_t$  and  $Y_{t+k}$ , conditional given  $Y_{t+1}, \dots, Y_{t+k-1}$ . Let  $\hat{\phi}_{k,k}$  denote the estimate of  $\phi_{k,k}$ .  $\hat{\phi}_{k,k}$  can be calculated by fitting the regression model

$$Y_t = \phi_{0,k} + \phi_{1,k}Y_{t-1} + \dots + \phi_{k,k}Y_{t-k} + \epsilon_{k,t}.$$

If  $Y_t$  is an  $\text{AR}(p)$  process, then  $\phi_{k,k} = 0$  for  $k > p$ . Therefore, a sign that a time series can be fit by an  $\text{AR}(p)$  model is that the sample PACF will be nonzero up to  $p$  and then will be nearly zero for larger lags. If the sample PACF decays slowly to zero, rather than dropping abruptly to zero, then it is an indication that this time series should not be fit by a pure AR process. An MA or ARMA process would be preferable.

When computing resources were expensive, the standard practice was to identify a tentative ARMA model using the sample ACF and PACF, fit this model, and then check the ACF and PACF of the residuals. If the residual ACF and PACF revealed some lack of fit, then the model could be enlarged. Now many data analyst prefer to start with a relatively large set of models and compare them with selection criteria such as AIC and BIC. This can be done automatically by `auto.arima` in R.

## 10 Cointegration

**Concepts.** Two time series,  $Y_{1,t}$  and  $Y_{2,t}$  are *cointegrated* if each is  $I(1)$  but there exists a  $\lambda$  such that  $Y_{1,t} - \lambda Y_{2,t}$  is stationary. More generally, a  $d$ -dimensional multivariate time series is *cointegrated of order  $r$*  if the component series are  $I(1)$  but  $r$  independent linear combination of the components are  $I(0)$  for some  $r$ ,  $0 < r \leq d$ .

Beside applications in finance, the notion of cointegration also helps with the understanding of regression analysis. The result of regression could be spurious when the residuals are integrated, which should make one cautious about regression with nonstationary time series. However, if  $Y_t$  is regressed on  $X_t$  and the two series are cointegrated, then the residual will be  $I(0)$  so that least-square estimator will be consistent.

The Phillips-Ouliaris cointegration test regresses one integrated series on others and applies the Phillips-Perron unit root test to the residuals. The null hypothesis is that the residuals are unit root nonstationary, which implies that the series are *not* cointegrated. Therefore, a small  $p$ -value implies that the series *are* cointegrated and therefore suitable for regression analysis. The residuals will still be correlated and so they should be modeled as such. In practice, though stationary, the residuals could have a large amount of autocorrelation and may have long-term memory. They take a long time to revert to their mean of zero. Devising a profitable trading strategy can be problematic.

The Phillips-Ouliaris test is run using the R function `po.test` in the `tseries` package.

**Vector error correction model (VECM).**

$$\Delta \mathbf{Y}_t = \alpha \beta^T \mathbf{Y}_{t-1} + \epsilon_t,$$

so that  $\beta$  is the cointegration vector and  $\alpha$  specifies the speed of mean-reversion and is called the *loading matrix* or *adjustment matrix*.

**Pair trading.** Pairs trading requires the trader to find cointegrated pairs of assets, to select from these the pairs that can be traded profitably after accounting for transaction costs, and finally to design the trading strategy which includes the buy and sell signals.

One of the risks involved is model risk: the error-correction model may be incorrect. Even if the model is correct, one must use estimates based on past data and the parameters might change, perhaps rapidly. Another risk is that one can go bankrupt before a stationary process reverts to its mean.

## 11 Long-Memory Processes

Economic and financial time series often show long memory (ACFs decay at a polynomial rate) and volatility clustering, while stationary ARIMA processes have only short memories (ACFs decay to zero exponentially fast) and constant conditional variance. We can use fractional ARIMA processes to model long memory and GARCH processes to model volatility clustering.

We extend the differencing operator  $\Delta^d$  to real number  $d > -1$  as

$$\Delta^d Y_t = (1 - B)^d Y_t = \sum_{k=0}^d \binom{d}{k} (-1)^k Y_{t-k}.$$

where  $\binom{d}{k} = \frac{d!}{k!(d-k)!}$  is defined for all real  $d$  except  $0, -1, -2, \dots$  via Gamma function  $\Gamma(t) = \int_0^\infty x^{t-1} e^{-x} dx$  with the fact  $\Gamma(t) = (t-1)\Gamma(t-1)$ .  $Y_t$  is called a *fractional ARIMA( $p, d, q$ ) process* (FARIMA( $p, d, q$ ) process), if  $\Delta^d Y_t$  is an ARIMA( $p, q$ ) process.

A more parsimonious model can sometimes be used if the differencing is fractional. Negative autocorrelation at lag-1 and little additional autocorrelation often indicate overdifferencing.

The function `fracdiff` in R's `fracdiff` package will fit a FARIMA( $p, d, q$ ) process. The values of  $p$ ,  $d$ , and  $q$  must be input; I am not aware of any R function that will choose  $p$ ,  $d$ , and  $q$  automatically in the way this can be done for an ARIMA process using `auto.arima`. Fractional differencing was done with the `diffseries` function in R's `fracdiff` package.

## 12 AR(1)/ARCH(1) Model

**Review of the AR(1) model.** The AR(1) process  $Y_t - \mu = \phi(Y_{t-1} - \mu) + \epsilon_t$  has constant mean and variance:

$$E[Y_t] = \mu, \text{Var}(Y_t) = \frac{\sigma_\epsilon^2}{1 - \phi^2},$$

nonconstant conditional mean:

$$E[Y_t | \mathcal{F}_{t-1}] = \mu + \phi(Y_{t-1} - \mu),$$

and constant conditional variance:

$$\text{Var}[Y_t | \mathcal{F}_{t-1}] = E[(Y_t - E[Y_t | \mathcal{F}_{t-1}])^2 | \mathcal{F}_{t-1}] = \sigma_\epsilon^2.$$

The ACF function of  $Y_t$  is

$$\rho(h) = \text{Corr}(Y_t, Y_{t+h}) = \phi^{|h|}.$$

**The ARCH(1) model.** The process  $a_t$  is called an *ARCH(1) process* if

$$a_t = \sqrt{\omega + \alpha_1 a_{t-1}^2} \epsilon_t$$

where  $\epsilon_t$  is an i.i.d.  $\text{WN}(0, 1)$ ,  $\omega > 0$ , and  $\alpha_1 \in [0, 1)$ .  $a_t$  has constant mean and variance:

$$E[a_t] = 0, \text{Var}(a_t^2) = \frac{\omega}{1 - \alpha_1},$$

constant conditional mean:

$$E[a_t | \mathcal{F}_{t-1}] = 0,$$

and nonconstant conditional variance:

$$\sigma_t^2 := \text{Var}[a_t | \mathcal{F}_{t-1}] = \omega + \alpha_1 a_{t-1}^2 \tag{5}$$

The ACF function of  $a_t$  and the ACF function of  $a_t^2$  are, respectively

$$\rho(h) = 0 \text{ if } h \neq 0, \rho_{a^2}(h) = \alpha_1^{|h|}. \tag{6}$$

Equation (5) is crucial to understanding how GARCH processes work. If  $a_{t-1}$  has an unusually large absolute value, then  $\sigma_t$  is larger than usual and so  $a_t$  is also expected to have an unusually large magnitude. This volatility propagates since when  $a_t$  has a large deviation that makes  $\sigma_{t+1}^2$  large so that  $a_{t+1}$  tends to be large and so on. Similarly, if  $a_{t-1}^2$  is unusually small, then  $\sigma_t^2$  is small, and  $a_t^2$  is also expected to be small, and so forth. Because of this behavior, unusual volatility in  $a_t$  tends to persist, though not forever. The conditional variance tends to revert to the unconditional variance provided that  $\alpha_1 < 1$ , so that the process is stationary with a finite variance.

Note the ARCH(1) process provides an example of uncorrelated but dependent random variables. Also note since  $a_t$  has zero mean and constant variance and  $\rho(h) = 0$  for  $h \neq 0$ ,  $(a_t)$  is weak white noise. The first step to see if a weak white noise has ARCH effect is to examine its ADF and the ADF of its square according to equation (6).

**The AR(1)/ARCH(1) model.** An AR(1) process has a nonconstant conditional mean but a constant conditional variance, while an ARCH(1) process is just the opposite. If both the conditional mean and variance of the data depend on the past, then we can combine the two models. E.g. an ARCH(1) process can be used as the noise term of an AR(1) process.

Let  $a_t$  be an ARCH(1) process so that  $a_t = \sqrt{\omega + \alpha_1 a_{t-1}^2} \epsilon_t$  where  $\epsilon_t$  is i.i.d.  $\text{N}(0,1)$ , and suppose that

$$u_t - \mu = \phi(u_{t-1} - \mu) + a_t.$$



The process  $u_t$  is an AR(1) process, except that the noise term ( $a_t$ ) is not i.i.d. white noise but rather an ARCH(1) process which is only weak white noise. We note

$$E[u_t] = \mu, \text{Var}(u_t) = \frac{\text{Var}(a_t)}{1 - \phi^2} = \frac{\omega}{1 - \alpha_1} \cdot \frac{1}{1 - \phi^2},$$

and

$$E[u_t | \mathcal{F}_{t-1}] = \mu + \phi(u_{t-1} - \mu), \text{Var}[u_t | \mathcal{F}_{t-1}] = \text{Var}[a_t | \mathcal{F}_{t-1}] = \omega + \alpha_1 a_{t-1}^2.$$

**ARCH( $p$ ) models.**  $a_t$  is an ARCH( $p$ ) process if  $a_t = \sigma_t \epsilon_t$ , where

$$\sigma_t = \sqrt{\omega + \sum_{i=1}^p \alpha_i a_{t-i}^2}$$

is the conditional standard deviation of  $a_t$  given the past values  $a_{t-1}, a_{t-2}, \dots$  of this process. Like an ARCH(1) process, an ARCH( $p$ ) process is uncorrelated with a zero mean (both conditional and unconditional) and a constant unconditional variance, but its conditional variance is nonconstant.

### 13 ARIMA( $p_A, d, q_A$ )/GARCH( $p_G, q_G$ ) Models

A deficiency of ARCH( $p$ ) models is that the conditional standard deviation process has high-frequency oscillations with high volatility coming in short bursts. GARCH models permit a wider range of behavior, in particular, more persistent volatility. The GARCH( $p, q$ ) model is

$$a_t = \sigma_t \epsilon_t$$

where

$$\sigma_t = \sqrt{\omega + \sum_{i=1}^p \alpha_i a_{t-i}^2 + \sum_{i=1}^q \beta_i \sigma_{t-i}^2}.$$

Because past values of the  $\sigma_t$  processes are fed back into the present value, the conditional standard deviation can exhibit more persistent periods of high or low volatility than seen in an ARCH process. The process  $a_t$  is uncorrelated with a stationary mean and variance and  $a_t^2$  has an ACF like an ARMA process.

**Residuals for ARIMA( $p_A, d, q_A$ )/GARCH( $p_G, q_G$ ) models.** When one fits an ARIMA( $p_A, d, q_A$ )/GARCH( $p_G, q_G$ ) model to a time series  $Y_t$ , there are two types of residuals. The *ordinary residual*, denoted  $\hat{a}_t$ , is the difference between  $Y_t$  and its conditional expectation and it estimates  $a_t$ . A *standardized residual*, denoted  $\hat{\epsilon}_t$ , is an ordinary residual divided by its conditional standard deviation  $\hat{\sigma}_t$  and it estimates  $\epsilon_t$ . The standardized residuals should be used for model checking. If the model fits well, then neither  $\hat{\epsilon}_t$  nor  $\hat{\epsilon}_t^2$  should exhibit serial correlation.

**GARCH processes have heavy tails.** Stock returns have “heavy-tailed” or “outlier-prone” probability distributions. One reason for outlier may be that the conditional variance is not constant, and the outliers occur when the variance is large. GARCH process exhibit heavy tails even if  $\{\epsilon_t\}$  is Gaussian. Therefore, when we use GARCH models, we can model both the conditional heteroskedasticity and the heavy-tailed distributions of financial market data. Nonetheless, many financial time series have tails that are heavier than implied by a GARCH process with Gaussian  $\{\epsilon_t\}$ . To handle such data, one can assume  $\{\epsilon_t\}$  is an i.i.d. white noise process with a heavy-tailed distribution.

**Fitting ARMA/GARCH models.** An ARMA/GARCH model can be fitted using R’s `garchFit` function in the `fGarch` package.

**GARCH models as ARMA models.** If  $a_t$  is a GARCH process, then  $a_t^2$  is an ARMA process but with weak white noise, not i.i.d. white noise. More precisely, assume

$$a_t = \sigma_t \epsilon_t$$

and

$$\sigma_t^2 = \omega + \sum_{i=1}^p \alpha_i a_{t-i}^2 + \sum_{i=1}^q \beta_i \sigma_{t-i}^2.$$

Define  $\eta_t = a_t^2 - \sigma_t^2$ . Then  $(\eta_t)$  is an uncorrelated process with zero mean and constant variance.<sup>3</sup> Define  $\mu = \omega/[1 - \sum_{i=1}^p (\alpha_i + \beta_i)]$ , then we can show that

$$a_t^2 - \mu = \sum_{i=1}^p (\alpha_i + \beta_i)(a_{t-i}^2 - \mu) - \sum_{i=1}^q \beta_i \eta_{t-i} + \eta_t,$$

so that  $a_t^2$  is an ARMA( $p, q$ ) process with mean  $\mu$ . Consequently, the ACF of  $a_t^2$  is the same as the ACF of an ARMA( $p, q$ ) process.

**GARCH(1,1) processes.** If  $a_t$  is GARCH(1,1), then  $a_t^2$  is ARMA(1,1). So the ACF of  $a_t^2$  can be obtained from formula (2):

$$\rho_{a^2}(1) = \frac{\alpha_1(1 - \alpha_1\beta_1 - \beta_1^2)}{1 - 2\alpha_1\beta_1 - \beta_1^2}, \quad \rho_{a^2}(k) = (\alpha_1 + \beta_1)^{k-1} \rho_{a^2}(1), \quad k \geq 2.$$

There are infinitely many values of  $(\alpha_1, \beta_1)$  with the same value of  $\rho_{a^2}(1)$  and a higher value of  $\alpha_1 + \beta_1$  means a slower decay of  $\rho_{a^2}$  after the first lag. The capability of the GARCH(1,1) model to fit the lag-1 autocorrelation and the subsequent rate of decay separately is important in practice. It appears to be the main reason that the GARCH(1,1) model fits so many financial time series.

**APARCH models.** Standard GARCH models cannot model the leverage effect because they model  $\sigma_t$  as a function of past values of  $a_t^2$ —whether the past values of  $a_t$  are positive or negative is not taken into account, since the square function  $x^2$  is symmetric in  $x$ . The APARCH (asymmetric power ARCH) models replace the square function with a flexible class of nonnegative functions that include asymmetric functions.

The APARCH( $p, q$ ) model for the conditional standard deviation is

$$\sigma_t^\delta = w + \sum_{i=1}^p \alpha_i (|a_{t-i}| - \gamma_i a_{t-i})^\delta + \sum_{j=1}^q \beta_j \sigma_{t-j}^\delta,$$

where  $\delta > 0$  and  $-1 < \gamma_j < 1, j = 1, \dots, p$ .

**Forecasting ARMA/GARCH processes.** Forecasting ARMA/GARCH processes is in one way similar to forecasting ARMA processes—the forecasts are the same because a GARCH process is weak white noise. What differs between forecasting ARMA/GARCH and ARMA processes is the behavior of the prediction intervals.

## 14 Regression with ARMA/GARCH Errors

When using time series regression, one often observes autocorrelated residuals. Residual correlation causes standard errors and confidence intervals (which incorrectly assume uncorrelated noise) to be incorrect. In particular, the coverage probability of confidence intervals can be much lower than the nominal value.

For this reason, linear regression with ARMA disturbances was introduced, assuming the residuals are stationary. The model was

$$Y_i = \beta_0 + \beta_1 X_{i,1} + \dots + \beta_p X_{i,p} + \epsilon_i, \quad (7)$$

where

$$(1 - \phi_1 B - \dots - \phi_p B^p)(\epsilon_t - \mu) = (1 + \theta_1 B + \dots + \theta_q B^q)u_t, \quad (8)$$

---

<sup>3</sup>How to check the variance is a constant?

and  $\{u_t\}$  is i.i.d. white noise. To accommodate volatility clustering, instead of being i.i.d. white noise, we assume  $\{u_t\}$  is a GARCH process so that

$$u_t = \sigma_t v_t \tag{9}$$

where

$$\sigma_t = \sqrt{\omega + \sum_{i=1}^p \alpha_i u_{t-1}^2 + \sum_{j=1}^q \beta_j \sigma_{t-j}^2}, \tag{10}$$

and  $\{v_t\}$  is i.i.d. white noise. The model given by equations (7)-(10) is a *linear regression model with ARMA/GARCH disturbances*. To fit this model, a three-step estimation method is followed:

1. estimate the parameters in (7) by ordinary least-squares;
2. fit model (8)-(10) to the ordinary least-squares residuals  $\epsilon_t$ ;
3. reestimate the parameters in (7) by weighted least-squares with weights equal to the reciprocals of the conditional variances from step 2.

Using linear regression model with ARMA/GARCH disturbances requires stationary residuals. However, many financial time series are nonstationary or, at least, have very high and long-term autocorrelation. When one nonstationary series is regressed upon another, it happens frequently that the residuals are nonstationary. In the extreme case where the residuals are an integrated process, the least-squares estimator is inconsistent.

The problem of correlated noise can be detected by looking at sample ACF of the residuals. The *Durbin-Watson test* can be used to test the null hypothesis of no residual autocorrelation. More precisely, the null hypothesis of the Durbin-Watson test is that the first  $p$  autocorrelation coefficients are all 0, where  $p$  can be selected by the user. In the R function `durbin.watson` in the `car` package,  $p$  is called `max.lag` and has a default value of 1. The  $p$ -value is computed by `durbin.watson` using bootstrapping.<sup>4</sup> The `lmtest` package of R has another function, `dwtest`, that computes the Durbin-Watson test, but only with  $p = 1$ . `dwtest` uses either a normal approximation (default) or an exact algorithm to calculate the  $p$ -value.

When nonstationary residuals are detected to exist, differencing may help, esp. when the residual is  $I(1)$ .

## References

- [1] David Ruppert. *Statistics and data analysis for financial engineering*. New York, Springer, 2011. 1, 3

---

<sup>4</sup>The calculated  $p$ -value could have substantial random variation due to bootstrapping with the default of  $B = 1000$  resamples. Using a larger number of resamples will compute the  $p$ -value with more accuracy.