arma 項の利用方法

前回の ar 項の復習から始めましょう。EViews のサンプルファイル hs.wf1[1]を使って次に示す自己回帰モデルを推定してみましょう。

$$HS_t = c_1 + u_t \tag{1}$$
$$u_t = \rho_1 u_{t-1} + \rho_2 u_{t-2} + \epsilon_t \tag{2}$$

この場合、Estimatoin ダイアログには次のよう入力するという事を前回ご説明しました。

```
hs c ar(1) ar(2)
```

u_t が次のように記述できる場合、EViews マニュアルではこれを AR (p) モデルと呼びます。

$$u_t = \rho_1 u_{t-1} + \rho_2 u_{t-2} + \cdots + \rho_p u_{t-p} + \epsilon_t \tag{3}$$

これに対し、移動平均モデル MA (q) というものもあります。

$$u_t = \epsilon_t + \theta_1 \epsilon_{t-1} + \theta_2 \epsilon_{t-2} + \cdots + \theta_q \epsilon_{t-q} \tag{4}$$

書籍やソフトウェアによっては θ の符号をマイナスにしている事もあります。EViews の場合はここに示すようにプラスとします。次のようなモデルを考えてみましょう。

$$\begin{aligned} HS_t &= c_1 + u_t \\ u_t &= \epsilon_t + \theta_1 \epsilon_{t-1} \end{aligned} \tag{5}$$

これは移動平均過程 MA (1) と呼ばれるものです。EViews での書き方は次のようになります。

```
hs c ma(1)
```

それでは AR (p) と MA (q) を組み合わせた ARMA (p, q) モデルを次に示します。

$$\begin{aligned} u_t = {} & \rho_1 u_{t-1} + \rho_2 u_{t-2} + \cdots + \rho_p u_{t-p} + \epsilon_t + \theta_1 \epsilon_{t-1} \\ & + \theta_2 \epsilon_{t-2} + \cdots + \theta_q \epsilon_{t-q} \end{aligned} \tag{6}$$

具体例として次のような ARMA $(2, 1)$ モデルを想定してみましょう。

$$\begin{aligned} HS_t &= c_1 + u_t \\ u_t &= \rho_1 u_{t-1} + \rho_2 u_{t-2} + \epsilon_t + \theta_1 \epsilon_{t-1} \end{aligned} \tag{7}$$

EViews では次のようにします。

```
hs c ar(1)ar(2) ma(1)
```

ちなみに式-7 は次のように書くこともできます。

$$HS_t = c_1 + \rho u_{t-1} + \rho u_{t-2} + \epsilon_t + \theta_1 \epsilon_{t-1}$$

[1] Program Files\EViews7\Example Files\EV7 Manual Data\Chapter21 にあります。

ここまでは被説明変数 HS_t を定数項に回帰させて、その残差が自己回帰型になっている、という文脈で話を進めてきました。しかし、次のようなモデルの場合も ARMA (p, q) と呼びます。HS_t が自己回帰型になっています。

$$HS_t = \beta_1 HS_{t-1} + \beta_2 HS_{t-2} + \epsilon_t + \theta_1 \epsilon_{t-1} \tag{8}$$

もちろん、この場合のコマンドは次のようになります。

```
hs hs(-1) hs(-2) ma(1)
```

一方、

$$HS_t = \beta_1 HS_{t-1} + \beta_2 HS_{t-2} + \rho_1 u_{t-1} + \epsilon_t \tag{9}$$

であれば、

```
hs hs(-1) hs(-2) ar(1)
```

となります。ARMA (p, q) モデルを利用する場合、具体的にどのような式であるのか明確にしてから推定を行うようにしましょう。ちなみに EViews のマニュアルでも触れているように、右辺に定数項だけを置き、誤差項について ARMA (p, q) モデルを利用する、式-7 のようなモデルを標準的な Box-Jenkins の ARMA モデルと呼びます。しかし、実際に式-8 や式-9 と明確に区別して呼ばれることは少ないようです。このことが、初学者にとって ar や ma を使う際に混乱を招いているのかもしれません。

記述方法について

例えば AR(1) モデルとして次のようなモデルを使っている場合と、

$$y_t = c_1 + \phi_1 y_{t-1} + \epsilon_t \tag{10}$$

EViews のマニュアルのように

$$y_t = c + u_t \tag{11}$$

$$u_t = \rho_1 u_{t-1} + \epsilon_t \tag{12}$$

つまり、

$$y_t = c + \rho_1 u_{t-1} + \epsilon_t \tag{13}$$

としている場合があります。何か違うことをやっているようにも見えますが、式-12 に 0 というと定数項があると考えると、式-10 と同じことになります。式-11 で y_t は定数項と u_t の和ですから、y_t の特徴は u_t の特徴そのものです。

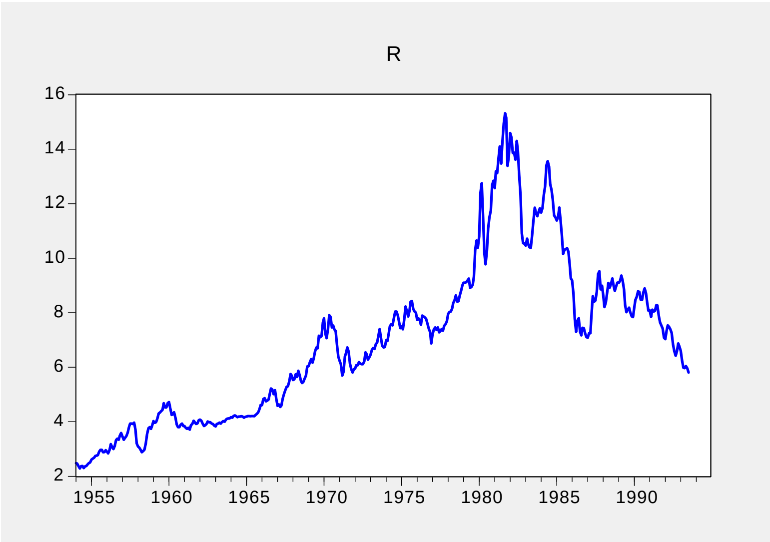
AR 過程

ar 項や ma 項を推定式で利用すると、EViews は通常の OLS に比べ、2 つの追加情報として「Inverted AR Roots」と「ARMA Structure」を提供しますので、その意味を確認しておきます。

Inverted AR Roots

推定結果の画面の下に「Inverted AR Roots」という項目を表示します。これは特性方程式の根 z の逆数です。$\frac{1}{|z|} \leq 1$ の場合に時系列データは定常となります。EViews のサンプルデータを利用して「Inverted AR Roots」の情報をどうやって利用すると価値があるのか、実際に EViews を操作して体験してみましょう。

EViews のサンプルファイル arma1.wf1(前出の hs.wf1 と同じフォルダにあります) にある変数 r を使います。



r の短期予測を行うという目的のために r について AR(1) モデル推定します。Estimation ダイアログに次のように入力します。

```
r c ar(1)
```

Equation オブジェクトの名前を eq01 とします。推定結果は次のようになります。

$$r_t = 7.897293 + u_t$$
$$u_t = 0.992715 u_{t-1} + \epsilon_t$$

したがって AR 特性方程式は、

$$0 = 1 - 0.992715z$$

ですから特性方程式の解 z は、

$$z = 1.007338$$

Inverted AR Root とは $\frac{1}{Z}$ ですから、この場合は元に戻って、

$$\frac{1}{Z} = \frac{0.992755}{1} = 0.992715$$

になります。AR 特性方程式の解 z については次の関係が知られています。

$$
\begin{aligned}
|z| \geq 1, \frac{1}{|z|} \leq 1 \quad &\Leftrightarrow \quad \text{定常} \\
|z| < 1, \frac{1}{|z|} > 1 \quad &\Leftrightarrow \quad \text{非定常}
\end{aligned}
$$

いま、$\frac{1}{Z} = 0.992755 < 1$ ですから、誤差項 u_t は定常過程であることが分かります。それでは AR(2) についても考えてみましょう。つまり、

```
r c ar(1) ar(2)
```

として eq02 を推定します。

$$
\begin{aligned}
r_t &= 7.506918 + u_t \\
u_t &= 1.319871 u_{t-1} - 0.328103 u_{t-2} + \epsilon_t
\end{aligned}
$$

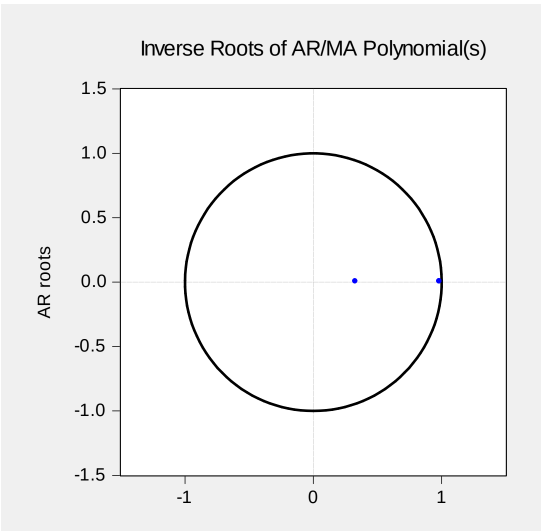
AR 特性方程式は、

$$0 = 1 - 1.319871z + 0.328103z^2$$

この 2 次方程式の解は、3.0103 と 1.0125 です。したがって、Inverted AR Root はそれぞれ 0.332193, 0.987654 になります。

ARMA Structure

特性方程式の解の情報や、誤差項の過程について考察するコマンドが View/ARMA Structure に用意されています。このコマンドは推定式に ar や ma 項が含まれている場合のみ選択可能になります。さっそく、特性方程式の解の大きさをグラフで確認してみましょう。Equation オブジェクトの View/ARMA Structure/Roots として Graph を選択します。



Inverted AR Root が単位円の中にあって、定常であることが一目で分かります。
コマンド ARMA Structure には上述の Roots 以外にも面白いコマンドが用意されていますが、それらについてはまた別の機会にご紹介します。それでは EQ01 と EQ02 のどちらを選択すべきでしょうか。その場合は Estiamtion の画面に表示された AIC 同士または SIC 同士の小さいな方を選択しましょう。

推定結果の注意点

AR 過程を含むモデルを推定すると、Inverted AR Roots 以外は通常の OLS 推定と同じ情報を出力します。しかし、解釈には注意が必要です。残差を使った統計量は u_t ではなく、ϵ_t を使って計算を行っています。EViews のマニュアルでは $\hat{u}_t$ を unconditional residuals, $\hat{\epsilon}_t$ を one-period ahead forecast errors と呼びます。$\hat{u}_t$ は次式のように求めます。

$$\hat{u}_t = y_t - x_t' b \tag{14}$$

例えば、式-11 に外生変数 x の入った AR(1) モデルを考えます。

$$\begin{aligned} y_t &= c + \beta x_t + u_t \\ u_t &= \rho_1 u_{t-1} + \epsilon_t \end{aligned} \tag{15}$$

u_t は一行目の式だけを使って、$\hat{u}_t = y_t - \hat{c} - \hat{\beta} x_t$ として計算します。一方、一期先予測の誤差 $\hat{\epsilon}_t$ は次のように求めます。

$$\begin{aligned} \hat{\epsilon}_t &= y_t - \hat{y}_t \\ &= y_t - \left(\hat{c} + \hat{\beta} x_t + \hat{\rho}_1 u_{t-1}\right) \end{aligned} \tag{16}$$

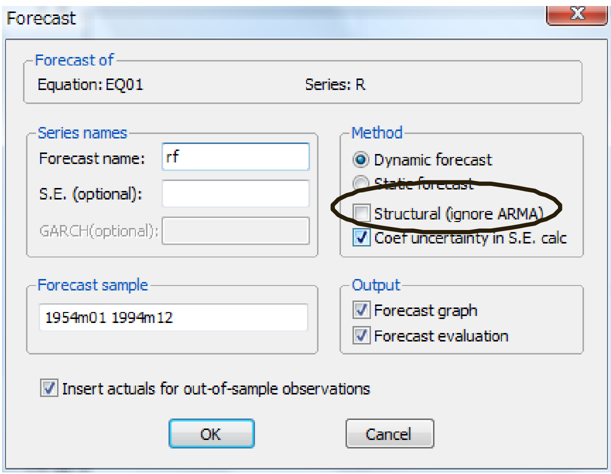
残差の計算に $\hat{\rho}_1 \hat{u}_{t-1}$ の情報を追加しています。具体的に統計量の例を挙げて言うと、次に示す R^2 や DW 統計量の計算にはこの $\hat{\epsilon}_t$ を用います。

$$R^2 = 1 - \frac{\hat{\epsilon}_t' \hat{\epsilon}_t}{(y_t - \bar{y}_t)' (y_t - \bar{y}_t)}$$

$$\mathrm{DW} = \frac{\sum_{t=2}^{T} (\hat{\epsilon}_t - \hat{\epsilon}_{t-1})^2}{\sum_{t=1}^{T} \hat{\epsilon}_t^2}$$

この他にも推定結果の一覧には残差 $\hat{\epsilon}_t$ を使った統計量があります。

$\hat{u}_t$ と $\hat{\epsilon}_t$ の違いをご説明したので、最後に予測を行う場合のオプションについても確認しておくことにしましょう。



arma 項を含む推定式で予測を行う場合も考慮して、Forecast ボックスのオプション "Structural(Ignore ARMA)" のオプションは常に外された状態になっています。AR 過程や MA 過程が存在する場合は、その推定情報も使って予測することになりますので、予測力は向上すると考えられます。

前回に引き続き、今回は ARMA についてご説明を行いました。AR 特性方程式に関する基礎知識は時系列分析について解説のなされている書籍をご参考ください。このように ar 項や ma 項を利用することは、時系列データを自分なりにモデル化していることに他なりません。次回は EViews の簡単なプログラミング機能を使って仮想の時系列データを作ってみることにします。

以上

株式会社ライトストーン
2012 年 1 月