

EÖTVÖS LORÁND TUDOMÁNYEGYETEM
TERMÉSZETTUDOMÁNYI KAR

VARMA FOLYAMATOK IDENTIFIKÁCIÓJA

MSc Szakdolgozat

Gilányi Gergely Tamás

Alkalmazott Matematika MSc

Sztochasztika szakirány

Témavezető:

Pröhle Tamás

Egyetemi Tanársegéd

ELTE Valószínűségszámítás Tanszék



Budapest, 2014

Tartalomjegyzék

| | |
|---|-----------|
| Bevezető | 5 |
| 1. VARMA folyamatokról általánosan | 6 |
| 1.1. Arma folyamatok | 6 |
| 1.2. VARMA folyamatok | 7 |
| 1.3. VARMA folyamat $VAR(1)$ reprezentációja | 11 |
| 1.4. VARMA autokovarianciája és autokorrelációja | 13 |
| 1.5. VARMA lineáris transzformációi | 16 |
| 2. VARMA identifikációja | 18 |
| 2.1. Final forma | 18 |
| 2.2. Echelon forma | 20 |
| 2.3. Kronecker index | 23 |
| 2.4. Identifikációs algoritmus stacionárius esetben | 27 |
| 2.4.1. Skalár $ARMAX$ reprezentáció | 28 |
| 2.4.2. Az algoritmus lépései stacionárius esetben | 32 |
| 2.4.3. Algoritmussal kapcsolatos tételek | 36 |
| 2.5. Identifikációs algoritmus kointegrált esetben | 37 |
| 2.5.1. Algoritmusunk adaptálása kointegrált folyamatok esetén | 37 |
| 2.5.2. A kointegrált algoritmus lépései | 38 |
| 2.5.3. Algoritmussal kapcsolatos tételek | 42 |
| 3. Program | 44 |
| 3.1. Tesztelés | 45 |
| 3.2. A programok forráskódjai | 47 |
| 3.2.1. Főprogram | 48 |
| 3.2.2. Null modell értékelő függvény | 49 |
| 3.2.3. Aktuális modell értékelő függvény | 50 |
| 3.2.4. Meghívó függvény | 51 |

Köszönetnyilvánítás

Szeretnék köszönetet mondani a témavezetőmnek Pröhle Tamásnak azt a sok időt és figyelmet, amit nekem és a munkámnak szánt. Folyamatos segítségével, érdekes ötleteivel, és szakmai tanácsaival jelentősen segítette ennek a dolgozatnak a létrejöttét!

Szeretnék még köszönetet mondani családomnak, és barátaimnak a sok támogatásért és bátorításért, amit tanulmányaim során nyújtottak!

Bevezető

Az első fejezetben általánosan fogok beszélni a VARMA folyamatokról és azok általános tulajdonságáról. Többek között a konstrukciójáról, a VAR és MA reprezentációjáról, és azok létezéséről, a $\Gamma(r)$ kovariancia függvényének kiszámolásáról, és a VARMA folyamatok lineáris transzformációjáról, és azok fokának felső becsléséről.

A második fejezetben ismertetem az VARMA identifikációjához használt népszerűbb alakokat, a Final formát, az Echelon formát, és az Inverz Echelon formát. Ezek közül az Inverz Echelon formával foglalkozom mélyrehatóbban, és ilyen alakban szeretném a folyamatunkat az y_t -t T különböző időpontú megfigyelésünk alapján megkonstruálni. Itt ismertetem a Kronecker indexek, és Invariánsainak a fogalmát, és ezeknek a jelentőségét. A későbbi fejezetekben ezeket a Kronecker invariánsokat szeretném meghatározni aszimptotikusan stacionárius, és kointegrált esetben.

Aszimptotikusan stacionárius esetben ismertetem Poskitt [4] új és gyorsan számolható algoritmusát az ARMAX reprezentáció segítségével, aminek segítségével az előbb említett Kronecker invariánsokat, és a koordináták azon permutációját meghatározhatjuk, amik szerint ezek a Kronecker invariánsok a megfelelő sorrendben vannak. Szóval meghatározzuk a koordinátáink azon sorrendjét, amik szerint a Kronecker indexek monoton csökkenő sorrendben lesznek rendezve, majd ez után alkalmazva Poskitt [2] cikkében használt Akaike-szerű, és Bayes-szerű információs kritérium függvényének a segítségével meghatározzuk a Kronecker invariánsainkat. Az algoritmus működését és konvergenciáját biztosítja az algoritmus utáni tétel.

Kointegrált esetben elsőnek kiszámoljuk a kointegrációs rangot Poskitt [3] cikkében taglalt függvény segítségével, majd hasonlóan használjuk az algoritmusunkat, mint aszimptotikusan stacioner esetben. Itt Engle-Granger [8] féle EC felírást, és a Beveridge-Nelson felírást használjuk, majd ennek segítségével megkonstruáljuk az ECARMAX reprezentációt. Az algoritmusunk során ezt fogjuk felhasználni. Innentől kezdve az algoritmusunkat hasonlóan az előző részhez futtatjuk a Kronecker invariánsok, és a koordináták megfelelő sorrendjének megtalálásához.

1. fejezet

VARMA folyamatokról általánosan

1.1. Arma folyamatok

Ebben a fejezetben bevezetem egyszerűbb (egy dimenziós) folyamatok segítségével az *ARMA* folyamatot, majd ezután *VAR* és *MA* folyamatok segítségével a *VARMA* folyamatot. Először definiáljuk a mozgóátlag, és az autoregresszív folyamatot.

1.1.1. Definíció. Legyen $\dots, \varepsilon_0, \varepsilon_1, \dots$ egy egyszerű fehér zaj folyamat úgy, hogy $E(\varepsilon_k) = \mu = 0$ várható értékkel, és $D(\varepsilon_k) = \sigma^2$ szórásnégyzettel. Emellett legyenek $m_0, \dots, m_q \in \mathbb{R}$.

$$u_n = m_0\varepsilon_n + m_1\varepsilon_{n-1} + \dots + m_q\varepsilon_{n-q} = \sum_{i=0}^q m_i\varepsilon_{n-i}$$

Ekkor azt mondjuk, hogy az alábbi $(u_n)_{n \in \mathbb{N}}$ egy q -ad rendű (véges) mozgóátlag folyamat. Jelölése $MA(q)$.

Tekintsük a következő polinomot: $m(z) = m_0 + m_1z + \dots + m_qz^q = \sum_{i=0}^q m_i z^i$. Ezt a polinomot az u_n folyamat karakterisztikus polinomjának nevezzük. Ha erre a polinomra igaz, hogy $m(z) \neq 0$, ha $|z| \leq 1$, akkor a folyamatunk invertálható.

1.1.2. Definíció. Legyen ξ_n egy stacionárius folyamat. Legyen $a_n \in \mathbb{R}$ és ε_n fehér zaj folyamat, ekkor

$$y_n = \mu + a_1y_{n-1} + a_2y_{n-2} + \dots + a_p y_{n-p+1} + \varepsilon_n = \sum_{i=1}^p a_i y_{n-i} + \varepsilon_n$$

Ekkor azt mondjuk, hogy az alábbi $(y_n)_{n \in \mathbb{Z}}$ egy p -ed rendű (véges) autoregresszív folyamat μ várható értékkel. Jelölése $AR(p)$.

Tekintsük a következő polinomot: $a(z) = a_0 + a_1z + \dots + a_pz^p = \sum_{i=0}^p a_i z^i$. Ezt a polinomot az y_n folyamat karakterisztikus polinomjának nevezzük. Ha erre a polinomra igaz, hogy

$a(z) \neq 0$, ha $|z| \leq 1$, akkor a folyamatunk stabilnak nevezzük. Ezeknek az alapfolyamatoknak a segítségével szeretnénk definiálni a számunkra lényeges *ARMA* folyamatot. Az *ARMA* folyamat ahogy a nevéből is adódik egy autoregresszív és egy mozgóátlag részből áll.

1.1.3. Definíció. Legyen y_n egy p -ed rendű autoregresszív folyamat a_1, \dots, a_p együtthatókkal, és legyen u_n egy q -ad rendű mozgóátlag folyamat m_0, \dots, m_q együtthatókkal. Ekkor az alábbi y_n folyamatot egy μ várható értékű (p, q) rendű autoregresszív mozgóátlag folyamatnak hívjuk (*ARMA*), ha y_n az alábbi módon áll elő:

$$y_n = \mu + a_1 y_{n-1} + a_2 y_{n-2} + \dots + a_p y_{n-p} + m_0 \varepsilon_n + m_1 \varepsilon_{n-1} + \dots + m_q \varepsilon_{n-q}$$

1.1.1. Megjegyzés. A fentebbi *ARMA*(p, q) folyamat felírható úgy is, hogy

$$y_n = \sum_{i=1}^p a_i y_{n-i} + \sum_{i=0}^q m_i \varepsilon_{n-i}$$

1.2. VARMA folyamatok

Ebben a fejezetben az előző fejezetben taglalt folyamatok több dimenzióra történő kiterjesztéséről lesz szó. Sajnos nem olyan egyszerű a munkánk, hogy le tudjuk annyival ezt a kérdést, hogy n darab független *ARMA* folyamatot egymás alá rakjunk, mivel ezen folyamatok között megengedett az összefüggőség. Definiálni fogjuk a többváltozós *AR* folyamatot, a *VAR* folyamatot, majd utána a többdimenziós *MA* folyamatot. Ezekből konstruáljuk meg a *VARMA* folyamatot (amivel a későbbiekben foglalkozni fogunk).

Elsőnek definiáljuk a többváltozós autoregresszív folyamatot:

1.2.1. Definíció. Legyen $y_t \in \mathbb{R}^n$ vektor. Legyen $A_1, \dots, A_p \in \mathbb{R}^{n \times n}$ konstans mátrixok, és legyen $\varepsilon_t \in \mathbb{R}^n$ vektor fehér zaj. Továbbá legyen $\mu = (\mu_1, \dots, \mu_n)$ a várható érték vektorunk. Ekkor

$$y_t = \mu + A_1 y_{t-1} + A_2 y_{t-2} \dots + A_p y_{t-p} + \varepsilon_t$$

Ekkor y_t -t egy μ várhatóértékű n dimenziós p -ed rendű *VAR* folyamatnak nevezzük.

Most definiáljuk a többváltozós mozgóátlag folyamatot:

1.2.2. Definíció. Legyen $u_t \in \mathbb{R}^n$ vektor. Legyen $M_1, \dots, M_q \in \mathbb{R}^{n \times n}$ konstans mátrixok. Legyen $\varepsilon_t \in \mathbb{R}^n$ nulla várható értékű fehér zaj folyamat Σ_ε kovariancia mátrixszal. Ekkor

$$u_t = \mu + \varepsilon_t + M_1 \varepsilon_{t-1} + \dots + M_q \varepsilon_{t-q} = \mu + \sum_{i=0}^q M_i \varepsilon_{t-i},$$

ahol $M_0 = I$. Ekkor azt mondjuk, hogy u_t egy n dimenziós q -ad rendű mozgóátlag folyamat.

1.2.1. Megjegyzés. Általában a μ vektorunkat nullának tekintjük az egyszerűség kedvéért.

A későbbiekben majd szükségünk lesz egy fontosabb tételre, amit most fogok kimondani és bizonyítani. Történetesen a Wold felbontásnak többdimenziós változatára.

1.2.1. Tétel. *Legyen u_t többváltozós q -ad rendű MA folyamat. Ekkor u_t -nek akkor létezik $VAR(\infty)$ reprezentációja, ha $\det(I_k + M_1z + \dots + M_qz^q) \neq 0$ minden $z \in \mathbb{R}$ és $|z| \leq 1$*

1.2.2. Megjegyzés. *Ha ez a tulajdonság teljesül u_t -re, akkor a mozgóátlag folyamatunkat invertálhatónak hívjuk.*

Bizonyítás: Elsőnek definiáljuk az L eltolás operátort. Ez az operátor azt tudja, hogy a folyamatunkat az időben balra eltolja eggyel, azaz formálisan $L\varepsilon_t = \varepsilon_{t-1}$. Ennek a felhasználásával próbáljuk felírni a mozgóátlag folyamatunkat:

$$u_t = \varepsilon_t + M_1\varepsilon_{t-1} + \dots + M_q\varepsilon_{t-q} = (I_n + M_1L + \dots + M_qL^q)\varepsilon_t = M(L)\varepsilon_t$$

Itt $M(L)$ egy mátrix, aminek az elemei maximum q -ad rendű polinomok (más néven polinom mátrix). Ha $M(L)$ teljesíti az invertálhatóság kritériumát, akkor $M(L)^{-1}$ -el balról megszorozhatjuk mindkettő egyenletünket.

$$M(L)^{-1}u_t = \varepsilon_t$$

Most $M(L)^{-1}$ -et szeretnénk felírni úgy, hogy VAR alakban álljon elő az egyenletünk. Tehát legyen

$$M(L)^{-1} = \Pi(L) = I_n - \sum_{i=1}^{\infty} \Pi_i L^i$$

Itt Π_i felírható $\Pi_1 = M_1$ induló mátrixértékkel, és a többi tag meg rekurzívan az alábbi egyenlet alapján (mátrixszorzással ellenőrizhető):

$$\Pi_i = M_i - \sum_{k=1}^{i-1} \Pi_{i-k} M_k$$

Ebből következik, hogy u_t -nek létezik VAR reprezentációja, ha a feltételeink teljesülnek, és az alábbi alakban áll elő:

$$\varepsilon_t = \Pi(L)u_t$$

□

A bizonyításunk után szemléltetés gyanánt tekintsünk egy egyszerű példát:

1.2.1. Példa. Legyen y_t egy n dimenziós $MA(1)$ folyamat. Formálisan:

$$y_t = \varepsilon_t + M_1 \varepsilon_{t-1}$$

ahol M_1 egy $n \times n$ konstans mátrix, $y_t = (y_{1,t}, \dots, y_{n,t})'$, ε_t egy nulla várható értékű fehér zaj folyamat egy nem szinguláris Σ_ε kovariancia mátrixszal. Innen egyszerű átalakítások után a következő alak adódik:

$$\varepsilon_t = y_t - M_1 \varepsilon_{t-1}$$

Egyszerű behelyettesítéssel könnyen adódik, hogy:

$$\varepsilon_t = y_t - M_1(y_{t-1} - M_1(\varepsilon_{t-2})) = y_t - M_1 y_{t-1} - M_1^2 \varepsilon_{t-2} \quad (1.1)$$

$$= y_t - M_1 y_{t-1} + \dots + (-M_1)^n y_{t-n} + (-M_1)^{n+1} \varepsilon_{t-n-1} \quad (1.2)$$

$$\approx y_t + \sum_{i=1}^{\infty} (-M_1)^i y_{t-i} \quad (1.3)$$

Innen következik az alábbi $VAR(\infty)$ alak:

$$y_t = \varepsilon_t - \sum_{i=1}^{\infty} (-M_1)^i y_{t-i}$$

Ez egy $VAR(\infty)$ stabilis folyamat, ha $M_1^i \xrightarrow{i \rightarrow \infty} 0$. Ez pedig pontosan akkor teljesül, ha M_1 sajátértékei kisebbek, mint 1. Azaz

$$\det(I_n - M_1 z) \neq 0, \forall z \in \mathbb{C}, |z| \leq 1$$

Többszörös változós folyamatok esetén természetesen az is igaz, hogy a kovariancia struktúrája a folyamatunknak nem feltétlenül szimmetrikus. Nézzük meg például a $\text{Cov}(u_t, u_{t-1})$ -et. Az ő kovariancia mátrixuk legyen Σ_1 . Ekkor $\Sigma_1(1, 2) = \text{Cov}(u_{t,1}, u_{(t-1),2})$. Viszont $\Sigma_1(2, 1) = \text{Cov}(u_{t,2}, u_{(t-1),1})$. Ennek a kettőnek pedig nem feltétlenül kell megegyeznie.

1.2.3. Megjegyzés. Ez az előbbi szerencsére nem jelent nekünk nagy problémát, mert az viszont igaz, hogy $\text{Cov}(u_t, u_{t+k}) = \text{Cov}(u_{t+k}, u_t)'$ -vel, azaz csak egy transzponálás a különbség a két kovariancia mátrix között.

Ezek után építsük fel ezekből a várva várt VARMA folyamatunkat, aminek az identifikációjával foglalkozni fogunk!

1.2.3. Definíció. Legyen egy p -ed rendű vektor autoregresszív (VAR) folyamatunk, és egy q -ad rendű mozgótag folyamatunk (MA). Ekkor a kettőjük által alkotott VARMA folyamat a következőképpen néz ki:

$$y_t = \mu + A_1 y_{t-1} + A_2 y_{t-2} + \dots + A_p y_{t-p} + \varepsilon_t + M_1 \varepsilon_{t-1} + M_2 \varepsilon_{t-2} + \dots + M_q \varepsilon_{t-q}.$$

Ahol ε_t nulla várhatóértékű fehér zaj folyamat egy nem szinguláris Σ_ε kovariancia mátrixszal.

Ezt a folyamatot úgy nevezzük, hogy VARMA(p, q).

1.2.4. Megjegyzés. A μ értékét általában nullának tekintjük az egyszerűség kedvéért.

A folyamatunk felírható az eltolás operátor segítségével is egy általunk könnyebben használható rövidebb alakban is:

$$A(L)y_t = M(L)\varepsilon_t$$

Ahol $A(L) = I_n - A_1L - A_2L^2 - \dots - A_pL^p$, és $M(L) = I_n + M_1L + M_2L^2 + \dots + M_qL^q$.

1.2.4. Definíció. Az y_t VARMA(p, q) folyamatunkról azt mondjuk, hogy stabil, ha $A(z) = (A_0 + A_1z + \dots + A_pz^p) \neq 0$, ha $|z| \leq 1$

Az y_t VARMA stabil folyamatunknak létezik MA reprezentációja, amit a $A(L)^{-1}$ balról való szorzásával érhetünk el. Formálisan

$$y_t = A(L)^{-1}M(L)\varepsilon_t = \Phi(L)\varepsilon_t = \sum_{i=0}^{\infty} \Phi_i \varepsilon_{t-i}$$

Ahol a $\Phi(L) = A(L)^{-1} * M(L)$. Itt $\Phi(L)$ megadható rekurzívan az alábbi módon:

$$A(L)\Phi(L) = (I_n - A_1L - A_2L^2 - \dots - A_pL^p) \left(\sum_{i=0}^{\infty} \Phi_i L^i \right) \quad (1.4)$$

$$= I_n + \sum_{i=1}^{\infty} \left(\Phi_i - \sum_{j=1}^i A_j \Phi_{i-j} \right) L^i \quad (1.5)$$

$$= I_n + M_1L + \dots + M_qL^q = M(L) \quad (1.6)$$

Innen mivel a polinomok együtthatói megegyeznek, ezért azonnal adódik az alábbi rekurzív képlet:

$$\Phi_0 = M_0 = I_n M_i = \Phi_i - \sum_{j=1}^i A_j \Phi_{i-j} \quad (1.7)$$

$$\Phi_i = M_i + \sum_{j=1}^i A_j \Phi_{i-j} \quad (1.8)$$

Ha az MA operátorunk $M(L)$ teljesíti az invertálhatósági feltételt (azaz $M(z) \neq 0$ minden $z \in \mathbb{C}, |z| \leq 1$), akkor a VARMA folyamatunkat invertálhatónak hívjuk. Ekkor létezik neki tiszta VAR reprezentációja az alábbi módon:

$$\varepsilon_t = M(L)^{-1}A(L)y_t = y_t - \sum_{i=1}^{\infty} \Pi_i y_{t-i} = \Pi(L)y_t$$

Kiszámolható a Π_i úgy, hogy felírjuk mátrixösszeg alakban $A(L)$ -t is, és $M(L)$ -t is, és erre

felírjuk az alábbi egyenletünket:

$$M(L)\Pi(L) = (I_n + M_1L + M_2L^2 + \dots + M_qL^q)(I_n - \sum_{i=1}^{\infty} \Pi_i L^i) \quad (1.9)$$

$$= I_n + \sum_{i=1}^{\infty} \left(M_i - \sum_{j=1}^i M_{i-j} \Pi_j \right) L^i \quad (1.10)$$

$$= I_n - A_1L - \dots - A_pL^p = A(L) \quad (1.11)$$

Ahol $A_0 = M_0 = I_n$ és $M_j = 0$, ha $j > q$, $A_j = 0$, ha $j > p$. Innen azonnal adódik, hogy

$$-A_i = M_i - \sum_{j=1}^{i-1} M_{i-j} \Pi_j - \Pi_i$$

Innen adódik, hogy

$$\Pi_i = A_i + M_i - \sum_{j=1}^{i-1} M_{i-j} \Pi_j$$

1.3. VARMA folyamat VAR(1) reprezentációja

Ebben a fejezetben a célunk, hogy a VARMA folyamatunkat átírjuk egyszerűen egy VAR(1) folyamattá. Ebben segítségünkre lesz egy Y_t és U_t folyamat, és egy A hipermátrix. Legyen y_t egy n dimenziós VARMA(p,q) folyamat. Definiáljuk $Y_t \in \mathbb{R}^{n(p+q) \times 1}$, és $U_t \in \mathbb{R}^{n(p+q) \times 1}$ vektort. Legyen

$$Y_t = (y_t, y_{t-1}, \dots, y_{t-p+1}, \varepsilon_t, \varepsilon_{t-1}, \dots, \varepsilon_{t-q+1})'$$

és

$$U_t = (\varepsilon_t, 0, \dots, 0, \varepsilon_t, 0, \dots, 0)'$$

Ezek után definiáljuk $A \in \mathbb{R}^{n(p+q) \times n(p+q)}$ hipermátrixot az alábbi módon:

$$A := \begin{pmatrix} A_{1,1} & A_{1,2} \\ A_{2,1} & A_{2,2} \end{pmatrix}$$

Ahol $A_{1,1} \in \mathbb{R}^{np \times np}$, $A_{1,2} \in \mathbb{R}^{np \times nq}$, $A_{2,1} \in \mathbb{R}^{nq \times np}$, és $A_{2,2} \in \mathbb{R}^{nq \times nq}$ dimenziós hipermátrix. $A_{i,j}$ $i, j = 1, 2$ -re a következő:

$$A_{1,1} = \begin{pmatrix} A_1 & A_2 & \dots & A_p \\ I_n & 0 & \dots & 0 \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \dots & I_n & 0 \end{pmatrix}$$

$$A_{1,2} = \begin{pmatrix} M_1 & M_2 & \cdots & M_q \\ 0 & 0 & \cdots & 0 \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & 0 \end{pmatrix}$$

$$A_{2,1} = 0$$

Azaz $A_{2,1}$ nullmátrix.

$$A_{2,2} = \begin{pmatrix} 0 & 0 & \cdots & 0 \\ I_n & 0 & \cdots & 0 \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \cdots & I_n & 0 \end{pmatrix}$$

Innen adódik nekünk a folyamatunk $n(p+q)$ dimenziós VAR(1) reprezentációja az alábbi módon:

$$Y_t = AY_{t-1} + U_t$$

1.3.1. Megjegyzés. Egyszerű mátrixszorzással ellenőrizhető, hogy az átalakításunk helyes, és valóban megegyezik a bal és a jobb oldalunk.

1.3.1. Állítás. A fentebbi y_t segítségével megkonstruált Y_t VAR(1) folyamatunk akkor és csak akkor stabil, ha az eredeti y_t VARMA(p, q) folyamatunk stabil volt.

Bizonyítás: Legyen Y_t $n(p+q) \times 1$ dimenziós VAR(1) folyamat. Legyen y_t n dimenziós VARMA(p, q) folyamat. Ekkor ha Y_t stabil az azt jelenti, hogy $\det(I_{n(pq)} - Az) \neq 0 \forall z \in \mathbb{C}, |z| \leq 1$. Ez a determináns átírható A definíciója szerint az alábbi módon:

$$\det(I_{n(p+q)} - Az) = \det(I_{np} - A_{1,1}z) \det(I_{nq} - A_{2,2}z) \quad (1.12)$$

$$= \det(I_n - A_1z - \dots - A_pz^p) \quad (1.13)$$

Itt 1.13 azért áll fenn, mert ha $A_{1,1}$ nem szinguláris mátrix, akkor a determináns elő áll az alábbi alakban:

$$\det A = \det(A_{1,1}) \det(A_{2,2} - A_{2,1}A_{1,1}^{-1}A_{1,2})$$

1.14 azért igaz, mert $(I_{nq} - A_{2,2}z)$ egy alsó háromszög mátrix, tehát a determinánsa pontosan a főátlóbeli elemek szorzata, azaz 1, és a másik tag $\det(I_{np} - A_{1,1}z)$ pedig azért egyenlő $A(z)$ -vel, mert ha kiszámoljuk a determinánsát, akkor azonnal egyszerűen adódik (sorcserek, és a determináns kifejtési tétel segítségével).

Itt a jobb oldalon pontosan az y_t stabilitásának az egyenletét kaptuk. Tehát pontosan akkor stabil Y_t , ha y_t stabil. \square

1.4. VARMA autokovarianciája és autokorrelációja

Legyen y_t n dimenziós, nulla várható értékű stabil VARMA(p, q) folyamat az alábbi együtthatókkal:

$$y_t = A_1 y_{t-1} + \dots + A_p y_{t-p} + \varepsilon_t + M_1 \varepsilon_{t-1} + \dots + M_q \varepsilon_{t-q}$$

Az autokovariancia mátrixa y_t -nek kiszámolható többek között például az MA reprezentáció segítségével. Legyen y_t MA reprezentációja az alábbi:

$$y_t = \sum_{i=0}^{\infty} \Phi_i \varepsilon_{t-i}$$

Ekkor az autokovariancia mátrixa y_t -nek az alábbi módon számolható:

$$\Gamma_y(h) = E(y_t y_{t-h}') = \sum_{i=0}^{\infty} \Phi_{h+i} \Sigma_\varepsilon \Phi_i'$$

Ennél kényelmesebb módszer kiszámolni a $\Gamma_y(h)$ -t úgy, hogy az y_t re vonatkozó alapegyenletünket beszorozzuk y_{t-h} -val és veszünk egy várható értéket. Ekkor

$$\begin{aligned} E(y_t y_{t-h}') &= A_1 E(y_{t-1} y_{t-h}') + \dots + A_p E(y_{t-p} y_{t-h}') + E(\varepsilon_t y_{t-h}') \\ &+ M_1 E(\varepsilon_{t-1} y_{t-h}') + \dots + M_q E(\varepsilon_{t-q} y_{t-h}') \end{aligned}$$

A folyamatunk felírásából (mind a VARMA, mind az MA -ból) következik, hogy ha $s < t$, akkor $E(\varepsilon_t y_s') = 0$. Tegyük fel, hogy $h > q$, ekkor a kovarianciára a következő formula igaz:

$$\Gamma_y(h) = A_1 \Gamma_y(h-1) + A_2 \Gamma_y(h-2) + \dots + A_p \Gamma_y(h-p)$$

Ha $p > q$ és $\Gamma_y(0), \dots, \Gamma_y(p-1)$ adott, akkor az autokovarianciáink egyszerűen kiszámíthatók $\Gamma_y(h)$ -ra minden $h = p, p+1, \dots$ esetén.

A kezdeti kovarianciák kiszámolásához meg alkalmazzuk az előző fejezetben taglalt $VAR(1)$ reprezentációját a folyamatunknak. Ekkor Y_t kovarianciájára a következő igaz:

$$\Gamma_Y(0) = \text{Cov}(Y_t, Y_t) = E(Y_t, Y_t') = E(AY_{t-1}, (AY_{t-1})') + \Sigma_U = A\Gamma_Y(0)A' + \Sigma_U$$

Rendezzük át az egyenletünket:

$$\Gamma_Y(0) - A\Gamma_Y(0)A' = \Sigma_U$$

Itt alkalmazzuk a vec operátort, azaz vegyük a bal oldal és a jobb oldalnak is a vektORIZÁLTJÁT!

$$vec(\Gamma_Y(0) - A\Gamma_Y(0)A') = vec(\Sigma_U)$$

1.4.1. Állítás. *A vec operátor lineáris.*

Bizonyítás: Ellenőrizzük le a linearitás tulajdonságait. Legyen $A, B \in \mathbb{R}^{m \times n}$, és $\lambda \in \mathbb{R}$.

$$vec(A + B) = vec(A) + vec(B) \quad (1.14)$$

$$vec(\lambda A) = \lambda vec(A) \quad (1.15)$$

Az első azért következik, mert legyen

$$A = \begin{pmatrix} a_{1,1} & a_{1,2} & \cdots & a_{1,n} \\ a_{2,1} & a_{2,2} & \cdots & a_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m,1} & a_{m,2} & \cdots & a_{m,n} \end{pmatrix}$$

és

$$B = \begin{pmatrix} b_{1,1} & b_{1,2} & \cdots & b_{1,n} \\ b_{2,1} & b_{2,2} & \cdots & b_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ b_{m,1} & b_{m,2} & \cdots & b_{m,n} \end{pmatrix}$$

Ekkor az alábbi igaz:

$$A + B = \begin{pmatrix} a_{1,1} + b_{1,1} & a_{1,2} + b_{1,2} & \cdots & a_{1,n} + b_{1,n} \\ a_{2,1} + b_{2,1} & a_{2,2} + b_{2,2} & \cdots & a_{2,n} + b_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m,1} + b_{m,1} & a_{m,2} + b_{m,2} & \cdots & a_{m,n} + b_{m,n} \end{pmatrix}$$

Ennek a vektoriálisa megegyezik a következő oszlopvektorral:

$$vec(A + B) = (a_{1,1} + b_{1,1}, a_{1,2} + b_{1,2}, \dots, a_{1,n} + b_{1,n}, a_{2,1} + b_{2,1}, \dots, a_{m,n} + b_{m,n})'$$

Ami pontosan megegyezik $vec(A) + vec(b)$ -vel! A második pont triviálisan adódik, mert konstanssal való szorzásnál mindegy, hogy először a mátrix elemeit szorozzuk meg, és utána alkalmazzuk a vec műveletet, vagy már alaphól a vektorizáltat szorozzuk meg a λ -val. \square

Ekkor az állításunkat felhasználva következik, hogy

$$vec(\Gamma_Y(0)) - vec(A\Gamma_Y(0)A') = vec(\Sigma_U)$$

A vec operátor tulajdonságai alapján tudjuk, hogy:

$$vec(AXB) = (B^T \otimes A)vec(X)$$

Ahol \otimes alatt a Kronecker szorzatot értjük. Ezt a tulajdonságot felhasználva az egyenletünk a következőképpen alakul:

$$Ivec(\Gamma_Y(0)) - (A \otimes A)vec(\Gamma_Y(0)) = vec(\Sigma_U)$$

Innen kiemelve a $vec(\Gamma_Y(0))$ -t és $(I - (A \otimes A))$ -al leosztva (inverzzel szorozva) a következő egyenletet kapjuk a keresett $vec(\Gamma_Y(0))$ -ra:

$$vec(\Gamma_Y(0)) = (I - A \otimes A)^{-1}vec(\Sigma_U)$$

Itt y_t stabilitásából következik Y_t stabilitása, amiből következik, hogy létezik az inverze a mátrixunknak. Így tehát az invertálhatóság kérdésével szerencsére nem szükséges foglalkoznunk. A vektoriálisból, ha visszaalakítjuk a mátrixunkat (azaz a mátrixunk első sora a vektor első n eleme lesz, második sora a második n eleme lesz etc...), akkor kapunk egy könnyen számolható képletet $\Gamma_Y(0)$ -ra.

$\Gamma_Y(0)$ -nk ekkor adott lesz, aminek kiszámolásával egyszerűen megadhatjuk $\Gamma_y(0), \dots, \Gamma_y(p-1)$ -et, mivel

$$\Gamma_Y(0) = \begin{pmatrix} \Gamma_{1,1}(0) & \Gamma_{1,2}(0) \\ \Gamma_{2,1}(0) & \Gamma_{2,2}(0) \end{pmatrix}$$

Ahol a kovariancia blokkok a következők lesznek kis számolás után:

$$\Gamma_{1,1}(0) = \begin{pmatrix} \Gamma_y(0) & \Gamma_y(1) & \cdots & \Gamma_y(p-1) \\ \Gamma_y(-1) & \Gamma_y(0) & \cdots & \Gamma_y(p-2) \\ \vdots & \vdots & \ddots & \vdots \\ \Gamma_y(-p+1) & \Gamma_y(-p+2) & \cdots & \Gamma_y(0) \end{pmatrix}$$

$$\Gamma_{1,2}(0) = \begin{pmatrix} E(y_t \epsilon'_t) & E(y_t \epsilon'_{t-1}) & \cdots & E(y_t \epsilon'_{t-q+1}) \\ 0 & E(y_{t-1} \epsilon'_{t-1}) & \cdots & E(y_{t-1} \epsilon'_{t-q+1}) \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & E(y_{t-p+1} \epsilon'_{t-q+1}) \end{pmatrix}$$

$$\Gamma_{2,2}(0) = \begin{pmatrix} \Sigma_U & 0 & \cdots & 0 \\ 0 & \Sigma_U & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \Sigma_U \end{pmatrix}$$

Innen az első blokk elemei megadják nekünk a keresett kovarianciáinkat. Ehhez szükséges volt nekünk, hogy $p > q$ legyen. Itt ha a feltételünk nem teljesül, akkor az autoregresszív részt kiegészíthetjük úgy, hogy hozzáadunk nulla koordinátákat addig, hogy a $p > q$ feltételünk teljesülhessen. Így ki tudjuk számítani a módszerünk szerint tetszőleges VARMA folyamatra a kívánt autokovarianciát!

A folyamatunk autokorrelációja úgy számolható, hogy

$$R_y(h) = \text{diag}(\Gamma_y(0)^{-\frac{1}{2}})\Gamma_y(h)\text{diag}(\Gamma_y(0)^{-\frac{1}{2}})$$

Itt $\text{diag}(\Gamma_y(0)^{-\frac{1}{2}})$ -el kell leosztanunk, ami a $\Gamma_y(0)$ főátlójának az inverzének a gyöke.

1.5. VARMA lineáris transzformációi

Ebben a fejezetben megismerkedhetünk a VARMA folyamatainknak a lineáris transzformációival, a transzformációk hatására létrejött folyamatok fontosabb tulajdonságaival mint például rendjével, és dimenziójával.

Elsőnek nézzük meg mozgóátlag folyamatokra!

1.5.1. Állítás. *Legyen y_t egy n dimenziós q -ad rendű mozgóátlag folyamat nem szinguláris Σ_ε kovarianciamátrixszal. Ekkor ennek az y_t folyamatnak egy lineáris transzformációja egy véges rendű mozgóátlag folyamatot ad, és nem növeli a lineáris transzformáció a folyamatunk rangját.*

Bizonyítás: Legyen y_t a fentebbi feltételeknek megfelelő folyamat. Így legyen

$$y_t = M(L)\varepsilon_t = \varepsilon_t + M_1\varepsilon_{t-1} + \dots + M_q\varepsilon_{t-q}$$

Továbbá legyen F egy $M \times n$ -es M rangú transzformáció mátrix. Ekkor definiáljuk z_t folyamatunkat úgy, hogy $z_t = Fy_t$ legyen. Ekkor z_t egy M dimenziós mozgóátlag lesz az alábbi együtthatókkal:

$$\begin{aligned} N_1 &= FM_1 \\ N_2 &= FM_2 \\ &\vdots \\ N_q &= FM_q \end{aligned} \tag{1.16}$$

Ezek az N_i együtthatók segítségével felírható z_t invertálható $MA(\hat{q})$ alakban:

$$z_t = F\varepsilon_t + N_1\varepsilon_{t-1} + \dots + N_q\varepsilon_{t-q}$$

Innen következik, hogy $\hat{q} \leq q$, mivel \hat{q} rangja maximum annyi lehet, a z_t reprezentációja szerint, mint y_t rangja. \square

1.5.1. Megjegyzés. Erre a z_t folyamatra a kovariancia függvény egyszerűen kiszámolható az y_t kovariancia függvényének segítségével az alábbi módon:

$$\Gamma_z(h) = E((Fy_t)(Fy_{t-h})') = F\Gamma_y(h)F' = \quad (1.17)$$

$$= \begin{cases} \sum_{i=0}^{q-h} FM_{i+h}\Sigma_\varepsilon M_i'F' & \text{ha } h = 0, 1, \dots, q \\ 0 & \text{ha } h > q \end{cases} \quad (1.18)$$

A következőkben nézzük meg, hogy a VARMA folyamatokra alkalmazott lineáris transzformációk milyen folyamatok is lesznek.

1.5.2. Állítás. Legyen y_t egy n dimenziós, stabil, invertálható VARMA(p, q) folyamat. Legyen F egy $(M \times n)$ -es M rangú mátrix. Ekkor $z_t = Fy_t$ folyamatnak van VARMA(\hat{p}, \hat{q}) reprezentációja úgy, hogy $\hat{p} \leq Kp$ és $\hat{q} < (K - 1)p + q$.

Bizonyítás: Írjuk fel a folyamatunkat eltolás operátoros alakban:

$$A(L)y_t = M(L)\varepsilon_t$$

Osszunk le $adj(A)$ -val mindkét oldalt. Ez megtehető, mert y_t stabil.

$$|A|y_t = adj(A(L))M(L)\varepsilon_t$$

Következő lépésben szorozzuk be mind a két oldalt az F transzformációkkal.

$$|A|z_t = Fadj(A(L))M(L)\varepsilon_t$$

Innen látszik, hogy $adj(A)A(L)$ foka maximum $p(n-1)$ lehet (adjungálás + mátrixszorzás miatt), és mivel lineáris transzformáltja egy $MA(q)$ folyamatnak $MA(\hat{q})$ folyamat, ezért $\hat{q} \leq p(n-1) + q$.

Az AR operátor $|A(L)|$ fokszáma maximum np lehet, mivel $A(L)$ foka maximum p és a determináns a főátlóbeli elemek szorzata. Így mivel n sorunk van, ezért np fokú lehet maximum az új AR operátorunk. \square

2. fejezet

VARMA identifikációja

2.1. Final forma

Tegyük fel, hogy y_t stacionárius és nulla várható értékű. Tegyük fel továbbá, hogy van stabil, invertálható VARMA(p, q) reprezentációja. Tekintsük a folyamatunkat a következő eltolásoperátoros alakban:

$$A(L)y_t = M(L)\varepsilon_t \quad (2.1)$$

Ahol $A(L) = \sum_{i=0}^p A_i L^i$, és $M(L) = \sum_{i=0}^q M_i L^i$. Emellett tegyük fel, hogy $[A(z) : M(z)]$ balról relatív prímek, és a fehér zaj kovariancia mátrix Σ_ε nem szinguláris.

2.1.1. Definíció. Ekkor azt mondjuk, hogy 2.1-ről, hogy Final formában van, ha $A(L) = \alpha(L)$, ahol $\alpha(L) = \sum_{i=0}^p \alpha_i L^i$ egy p -ed fokú polinom úgy, hogy $\alpha_0 = 1$ és $\alpha_p \neq 0$.

Ha egy kezdeti $A(L)y_t = M(L)\varepsilon_t$ VARMA(p, q) folyamatból indulunk ki, akkor az egyik legkézenfekvőbb módszer a Final formára hozásban az, hogy szorozzuk meg mind a két oldalt az $A(z)$ adjungáltjával. Ez megtehető, mivel a folyamatunk invertálható volt. Köztudott, hogy $(adj A(z))A(z) = \det A(z)$, ezért ha az adjungálttal szorzunk balról, akkor bal oldalon pont egy Final formának megfelelő $\alpha(L)$ polinomot kapunk. Formálisan:

$$\det A(z)y_t = (adj A(L))M(L)\varepsilon_t$$

Nézzünk erre egy példát:

2.1.1. Példa. Legyen y_t egy egyszerű 2 dimenziós VARMA(2, 0) folyamat az alábbi operátorokkal:

- $A(z) = \begin{pmatrix} 1 & -\alpha \\ 0 & 1 \end{pmatrix}$
- $M(z) = I_2$

Ekkor a folyamatunk reprezentációja úgy néz ki, hogy:

$$\begin{pmatrix} 1 & \alpha \\ 0 & 1 \end{pmatrix} y_t = I_2 \varepsilon_t$$

Könnyen kiszámolható, hogy $\det(A(z)) = \det\left(\begin{pmatrix} 1 & \alpha \\ 0 & 1 \end{pmatrix}\right) = 1$ és $\text{adj}A(z) = \begin{pmatrix} 1 & 0 \\ \alpha L & 1 \end{pmatrix}$

Ezeket behelyettesítve megkapjuk a folyamatunk Final forma előállítását:

$$y_t = \begin{pmatrix} 1 & 0 \\ \alpha L & 1 \end{pmatrix} \varepsilon_t$$

Nézzük meg egy második 3 dimenziós példát a Final formára alakításról:

2.1.2. Példa. Legyen

$$A(L) = \begin{pmatrix} 1 - \theta_1 L & -\theta_2 L & 0 \\ 0 & 1 - \theta_3 L - \theta_4 L^2 & -\theta_5 L \\ 0 & 0 & 1 \end{pmatrix}$$

és

$$M(L) = \begin{pmatrix} 1 - \eta_1 L & 0 & 0 \\ 0 & 1 - \eta_2 & 0 \\ 0 & 0 & 1 - \eta_3 L \end{pmatrix}$$

Az AR operátorunk determinánsa : $(1 - \theta_1 L)(1 - \theta_3 L - \theta_4 L^2)$ lesz, míg az adjungáltja a következő:

$$\text{adj}(A(L)) = \begin{pmatrix} 1 - \theta_3 L - \theta_4 L^2 & \theta_2 L & \theta_2 \theta_5 L^2 \\ 0 & 1 - \theta_1 L & \theta_5 L - \theta_1 \theta_5 L^2 \\ 0 & 0 & (1 - \theta_1 L)(1 - \theta_3 L - \theta_4 L^2) \end{pmatrix}$$

Így tehát a folyamatunk Final formában a következő lesz:

$$(1 - \theta_1 L)(1 - \theta_3 L - \theta_4 L^2) y_t = \begin{pmatrix} 1 - \theta_3 L - \theta_4 L^2 & \theta_2 L & \theta_2 \theta_5 L^2 \\ 0 & 1 - \theta_1 L & \theta_5 L - \theta_1 \theta_5 L^2 \\ 0 & 0 & (1 - \theta_1 L)(1 - \theta_3 L - \theta_4 L^2) \end{pmatrix} \times \begin{pmatrix} 1 - \eta_1 L & 0 & 0 \\ 0 & 1 - \eta_2 & 0 \\ 0 & 0 & 1 - \eta_3 L \end{pmatrix} \varepsilon_t$$

2.2. Echelon forma

Ebben a részben az Echelon (létra) formával fogunk megismerkedni közelebbről. Ez egy olyan alakja a VARMA folyamatoknak, amit az identifikációs algoritmusunkhoz használni szeretnénk a továbbiakban. Mivel az y_t sztochasztikus viselkedése nagyban függ $[A(z) : M(z)]$ operátor páruunktól ε_t zajon keresztül, ezért tegyük fel, hogy az ε_t -nk az alábbi feltételeknek eleget tesz:

1'. Feltevés. Legyen $\varepsilon_t = (\varepsilon_{1,t}, \dots, \varepsilon_{v,t})'$ stacionárius, ergodikus, martingál differencia sorozat. Legyen \mathcal{F}_t az ε_s -ek által generált σ -algebra, azaz $\mathcal{F}_t = \sigma\{\varepsilon_s : s \leq t\}$. Ekkor $E(\varepsilon_t | \mathcal{F}_{t-1}) = 0$. Emellett $E(\varepsilon_t \varepsilon_t' | \mathcal{F}_{t-1}) = \Sigma > 0$, és $E(\varepsilon_{j,t}^k) < \infty$ $j = 1, \dots, v$ és $k \geq 2$ -re.

Legyen a v dimenziós invertálható VARMA folyamatunk:

$$A(L)y_t = M(L)\varepsilon_t$$

Legyen az $[A(z) : M(z)]$ bal oldali relatív prím polinommatrix, legyen a rendje $p = \max_{1 \leq i \leq v} p_i$, ahol $n_i = \delta_i$, azaz $[A(z) : M(z)]$ i -edik sorának a foka. Ezeket az $n_i, i = 1, \dots, v$ egész számokat hívjuk Kronecker indexeknek. Ezek a Kronecker indexek meghatározzák a zaj struktúráját a folyamatunknak. Echelon formából két ekvivalens fajta létezik, a klasszikus Echelon forma, és az Inverz Echelon forma. A kettő közötti különbség az, hogy az egyik az autoregresszív operátorunkra szabja meg a feltételeinket, míg a második a mozgóátlag operátorunkra. Definiáljuk elsőnek az Echelon formát:

2.2.1. Definíció. Legyen y_t egy n dimenziós VARMA folyamat. Legyen $p = \max_{1 \leq i \leq v} p_i$ és $n_i = \delta_i$, azaz az i -edik sor foka. Ekkor $[A(z) : M(z)]$ Echelon formában van, ha az alábbiak teljesülnek:

1. $M_0 = A_0$, azaz a polinom mátrixunk konstans együtthatói megegyeznek.

2.

$$a_{rr}(z) = 1 + a_{rr,1}z + a_{rr,2}z^2 + \dots + a_{rr,n_r}z^{n_r} = 1 + \sum_{i=1}^{n_r} a_{rr,n_i}z^{n_i} \quad (2.2)$$

$$a_{rc}(z) = a_{rc,n_r-n_{rc}+1}z + \dots + a_{rc,n_r}z^{n_r} = \sum_{i=n_r-n_{rc}+1}^{n_r} a_{rc,n_i}z^{n_i} \quad (2.3)$$

3. $m_{rc} = m_{rc,0} + m_{rc,1}z + m_{rc,2}z^2 + \dots + m_{rc,n_r}z^{n_r} = \sum_{i=1}^{n_r} m_{rc,i}z^i$

Most definiáljuk az általunk használni kívánt Inverz Echelon formát:

2.2.2. Definíció. Legyen y_t egy n dimenziós VARMA folyamat. Legyen $p = \max_{1 \leq i \leq v} p_i$ és $n_i = \delta_i$, azaz az i -edik sor foka. Ekkor $[A(z) : M(z)]$ Echelon formában van, ha az alábbiak teljesülnek:

1. $M_0 = A_0$

2.

$$m_{rr}(z) = 1 + m_{rr,1}z + m_{rr,2}z^2 + \dots + m_{rr,n_r}z^{n_r} = 1 + \sum_{i=1}^{n_r} m_{rr,n_i}z^{n_i} \quad (2.4)$$

$$m_{rc}(z) = m_{rc,n_r-n_{rc}+1}z + \dots + m_{rc,n_r}z^{n_r} = \sum_{i=n_r-n_{rc}+1}^{n_r} m_{rc,n_i}z^{n_i} \quad (2.5)$$

3. $a_{rc} = a_{rc,0} + a_{rc,1}z + a_{rc,2}z^2 + \dots + a_{rc,n_r}z^{n_r} = \sum_{i=1}^{n_r} a_{rc,i}z^i$

Ahol n_{rc} az alábbi módon számolható egyszerűen:

$$n_{rc} = \begin{cases} \min(n_r + 1, n_c) & \text{ha } r \geq c \\ \min(n_r, n_c) & \text{ha } r < c \end{cases}$$

2.2.1. Megjegyzés. Tekintsük a két indexes Kronecker indexeket. Ezek definíciójából adódik, hogy a főátló fölött $\min(n_r, n_c)$ szerepel. Ebből látszik, hogy az Echelon formában levő MA mátrixunk konstans mátrixa csak alsó háromszög mátrix lehet.

2.2.2. Megjegyzés. Az előbbi három pontból több egyszerű következtetés vonható le:

- Könnyen látható, hogy az első feltétel azt határozza meg nekünk, hogy $M(z)$ és $A(z)$ konstans mátrixai megegyezzenek. Természetesen ezek nem feltétlenül csak az identitás mátrixok lehetnek, mivel ha tekintjük a második feltételt is, akkor láthatjuk, hogyha $n_r + 1 = n_{rc}$ -vel, akkor azokban az esetekben is keletkezhet konstans tagunk.
- A második sor meghatározza az $M(z)$ mátrixunk elemeinek minimális, és maximális fokszámait.
- A harmadik feltétel pedig az $A(z)$ mátrixra vonatkozó enyhébb feltétel. Itt csak a maximális fokszámot korlátozzuk le a sor Kronecker indexére.

Az inverz Echelon forma megkonstruálható az alábbi Ψ leképezés segítségével, amit az alábbi módon definiálunk:

$$M(z)\Psi(z) = A(z)$$

Itt $\Psi(z) = \sum_{i=0}^{\infty} \Psi_i z^i$. Ψ_i könnyen számolható az VAR reprezentációra vonatkozó rekurzió alapján:

$$\sum_{j=0}^i M_j \Psi_{i-j} = A_i, \quad i = 0, 1, \dots, p \quad (2.6)$$

$$\sum_{j=0}^p M_j \Psi_{i-j} = 0, \quad i = p+1, p+2, \dots \quad (2.7)$$

2.2.3. Megjegyzés. Egyértelmű, hogy $\Psi_0 = I$, mivel $A_0 = M_0$. Emellett tudjuk, hogy $\|\Psi_i\| < \infty$, ahol $\|\Psi_i\|^2 = \text{tr} \Psi_j \Psi_j'$ az euklideszi norma. Ez a második meglátás abból következik egyszerűen, hogy mivel a folyamatunk invertálható, ezért $\det M(z) \neq 0$, ha $|z| \leq 1$, amiből következik, hogy $\|\Psi_i\| \rightarrow 0$ exponenciális gyorsasággal ahogy $i \rightarrow \infty$. Emellett még az is igaz, hogy a következő hatványsor konvergens $\forall |z| \leq 1$ -re:

$$\Psi(z) = \lim_{N \rightarrow \infty} \sum_{i=0}^N \Psi_i z^i$$

A VARMA folyamatunk invertálása után a folyamatunk átírható $\varepsilon_t = \sum_{j=0}^{t-1} \Psi_j y_{t-j}$ folyamattá. Nevezzük $ARMA_E(\nu)$ -nek az inverz Echelon formában levő VARMA folyamatok osztályát, ahol $\nu = \{n_1, n_2, \dots, n_v\}$. Ekkor az $ARMA_E(\nu)$ meghatározza a VARMA modelünk kanonikus struktúráját a McMillan index segítségével, ami a következő:

$$m = \sum_{i=1}^v n_i$$

2'. Feltevés. Tegyük fel, hogy $[A(z) : M(z)]$ bal oldali relatív prímek, és $[A(z) : M(z)] \in ARMA_E(\nu)$. Emellett tegyük még fel, hogy

- $\det A(z)$ nem azonosan nulla mátrix
- $\det M(z)$ nem azonosan nulla mátrix
- $\det A(z) \neq 0$, ha $|z| < 1$ - kvázi stabilitás
- $\det M(z) \neq 0$, ha $|z| \leq 1$ - azaz az invertálást.

Itt jegyezzük meg, hogy a Feltevés 2' megengedi $A(z)$ -nek, hogy az egységkörön legyenek gyökei neki. Stacionárius esethez mondjuk ki a Feltevés 2'-nek az erősebb változatát, mely szerint:

3'. Feltevés. Tegyük fel, hogy Feltevés 2' teljesül, és e mellé teljesül az is, hogy $\det A(z) \neq 0$, ha $|z| \leq 1$, azaz a folyamatunk stabil.

2.3. Kronecker index

Ebben a fejezetben az y_t VARMA(\mathbf{p}, \mathbf{q}) folyamat Kronecker indexeinek a meghatározása lesz a célunk. Legyen y_t a szokásos v dimenziós VARMA(\mathbf{p}, \mathbf{q}) folyamatunk:

$$A(L)y_t = M(L)\varepsilon_t$$

Legyen továbbá ε_t egy v dimenziós zajfolyamat úgy, hogy $E(\varepsilon_t) = 0$, $\text{Cov}(\varepsilon_t, \varepsilon_s) = 0$, ha $s \neq t$, továbbá legyen ε_t kovariancia mátrixa Σ pozitív szemidefinit mátrix. Legyen továbbá $A(z) = I - A_1z - A_2z^2 - \dots - A_pz^p$ legfeljebb p -ed fokú polinom feletti mátrix és $M(z) = 1 - M_1z - M_2z^2 - \dots - M_qz^q$ legfeljebb q -ad fokú polinom feletti mátrix.

Legyen a Γ_l az a $v \times v$ dimenziós mátrix, ami y_t -nek az l . autokovariancia mátrixa, azaz $\Gamma_l = \text{Cov}(y_t, y_{t-l})$. Itt elemenként $\Gamma_{l,ij} = \text{Cov}(y_{t,i}, y_{t-l,j})$.

Vegyük $A(L)y_t$ folyamatunkra a kovarianciát y_{t-l} -el. Ekkor, ha $l > q$, akkor a következő kovariancia egyenlet érvényes:

$$\Gamma_l - A_1\Gamma_{l-1} - \dots - A_p\Gamma_p = 0 \quad (2.8)$$

Ez az egyenletünk amiatt teljesül, mert ha tekintjük a jobb oldalt azaz $M(L)\varepsilon_t$, akkor $E(M(L)\varepsilon_t y_{t-l})$ lesz. Mivel $M(L)$ foka csupán q , és $E(\varepsilon_t, \varepsilon_s) = 0$, ha $t \neq s$, ezért a jobb oldalunk azonosan nulla lesz minden $l > q$ -ra.

2.3.1. Megjegyzés. Természetesen itt megjegyezném, hogy ha $l < q$ lenne, akkor a jobb oldalunk nem azonosan nulla lenne, hanem egy mátrix, ami Σ -tól és $M(L)$ -től függene.

2.3.2. Megjegyzés. a 2.8 szerint kiszámolhatóak a kovarianciáink úgy, hogy $l = q + 1, \dots, 2q + 1$ egyenletet felírunk.

Vegyük az autokovarianciák alapján a következő H_l Hankel blokkmátrixot.

$$H_l = \begin{pmatrix} \Gamma_1 & \Gamma_2 & \cdots & \Gamma_l \\ \Gamma_2 & \Gamma_3 & \cdots & \Gamma_{l+1} \\ \cdots & \cdots & \ddots & \cdots \\ \Gamma_l & \Gamma_{l+1} & \cdots & \Gamma_{2l-1} \end{pmatrix}$$

Legyen $H = H_\infty$ végtelen dimenziós hipermatrix. Nyilvánvaló, hogy ez a H mátrix felírható úgy is, hogy konstruálunk y_t vektorok segítségével egy múlt és egy jövő vektort, és azok kovarianciáját vesszük. Legyen P (Past) a múlt vektor, és F (Future) a jövő vektor. Formálisan:

$$P_{t-1} = (y'_{t-1}, y'_{t-2}, y'_{t-3}, \dots)$$

és

$$F_t = (y'_t, y'_{t+1}, y'_{t+2}, \dots)$$

Ekkor $H = H_\infty = \text{Cov}(F_t, P_{t-1})$ teljesül. (egyszerű kovariancia számítással ellenőrizhető) Legyen a H mátrixunk $(i-1)v + j$ -edik sora $h_{i,j}$, azaz legyen a hipermátrixunk i -edik mátrixának j -edik sora. Ennek segítségével szeretnénk a Kronecker indexeinket meghatározni.

A H kovariancia mátrixunk a következő fontos tulajdonságokkal bír:

- i ha a $h_{i,j}$ sor az őt megelőző $h_{i_1,j_1}, \dots, h_{i_s,j_s}$ sorok lineáris kombinációja, akkor $h_{i+1,j}$ a $h_{i_1+1,j}, \dots, h_{i_s+1,j}$ sorok lineáris kombinációja, ugyanis a Hankel mátrixunk minden későbbi hipersora a korábbiak csonkolásával azonos.
- ii ha $\text{rang}(H)$ véges, akkor a folyamatunknak létezik VARMA reprezentációja.
- iii ha $\text{rang}(H) = m < \infty$, akkor már a $\text{rang}(H_m)$ értéke is egyenlő m -el, és ezt a rangot nevezzük az y_t folyamatunk (Hankel mátrixunk) Mcmillan indexének (fokszámának).

Ezek szerint a véges rangú H_m hipermátrixbeli mátrixok minden sorindexére, azaz minden $j = 1, \dots, v$ van egy olyan d_j index, amelyre ha $d \leq d_j$, akkor $h_{d,j}$ sor még független az előző $h_{i,j}$ -től, ahol $i < d$. Azonban, ha $d > d_j$, akkor $h_{d,j}$ sor már a korábbi $h_{i,j}$ $i < d$ soraink valamilyen lineáris kombinációja. Ezeket a d_j indexeket nevezzük a sorok Kronecker indexének. A Kronecker indexek vektora (d_1, \dots, d_v) . A Kronecker indexek vektora erősen függhet a koordináták sorrendjétől, és a koordinátáink Kronecker indexe is függhet attól, hogy az adott folyamat koordinátáit hányadik folyamatkoordinátaként soroltuk fel. Általában nem igaz, hogy a többdimenziós folyamatok egyes koordinátáinak a Kronecker indexe független volna attól, hogy a koordináta hányadik. A következő állítás egy fontos és természetes tulajdonságára hívja fel a figyelmet a Kronecker indexeknek. Mégpedig arra, hogy az y_t vektorunk koordinátáinak sorrendjének bizonyos módon történő megváltoztatása nem befolyásolja a Kronecker indexek halmazát.

2.3.1. Állítás. A Kronecker indexek halmaza független a koordináták sorrendjétől

Bizonyítás: Elég belátnunk, hogy két egymás utáni koordináta felcserélésétől nem függ a Kronecker indexek halmaza, mert az elemi algebrából ismert tétel szerint egymás utáni elemek cseréjével előállíthatjuk az elemek tetszőleges permutációját.

Két szomszédos koordináta felcserélése mindig az alábbi hét eset valamelyikének felel meg. A Hankel mátrixunk sorai elé írjunk egy 0-t vagy egy x -et úgy, hogy 0-t akkor, ha az adott sor függ az előtte levő soroktól, és x -et, ha az adott sor független az előtte levő soroktól.

Ezt a címkézést az alábbi példa szemlélteti:

$$H = \begin{matrix} x \\ x \\ x \\ 0 \\ \dots \end{matrix} \begin{pmatrix} \Gamma_{1,11} & \Gamma_{1,12} & \dots & \Gamma_{1,1k} & \dots \\ \Gamma_{1,21} & \Gamma_{1,22} & \dots & \Gamma_{1,2k} & \dots \\ \Gamma_{1,31} & \Gamma_{1,32} & \dots & \Gamma_{1,3k} & \dots \\ \Gamma_{1,41} & \Gamma_{1,42} & \dots & \Gamma_{1,4k} & \dots \\ \dots & \dots & \dots & \ddots & \dots \end{pmatrix}$$

Ha tekintjük a $h_{i,j}$ $i = 1, \dots$ elemeket, akkor azt tapasztaljuk, hogy előbb benne csak x -ek vannak, majd pedig egy indextől kezdve csak nullák lesznek benne. Ez a Hankel mátrix struktúrája miatt van, azaz a kovarianciák miatt, mivel a kovarianciák kiszámolására elegendő lépés után után felírhatunk egy rekurzív képletet, ami pont egy lineáris kombinációt biztosít a következő sorra. Tehát, ha egy $h_{i,j}$ sor 0 lett, akkor az utána levő $h_{k,j}$ sorok is összefüggőek lesznek $k = i + 1, \dots$ -ra. Emellett tudjuk, hogy az i . sor Kronecker indexe az pontosan megegyezik azzal, hogy $\#(x \text{ van } h_{k,i} \text{ előtt}, k \in \mathbb{Z}^+)$, azaz pontosan megegyezik az x -ek számával a $h_{k,i}$ sorok előtt, ahol $k \in \mathbb{Z}^+$.

Legyen u és v két egymás utáni folyamat koordináta. Legyen u a k -adik, míg v a $k + 1$ -edik. A Hankel mátrixunk i . hipersorában az (u,v) ' párnak megfelelő két sorvektorunkat jelölje nekünk u_i és v_i . A továbbiakban azt vizsgáljuk meg, hogy a Hankel mátrixunk sorai elé írt 0 vagy x oszlopvektor elemei hogyan változnak az u és a v felcserélésekor. Mivel a cserélt sorok szomszédosak, ezért elég nekünk az (u,v) párnak megfelelő sorok előtti jeleket vizsgálnunk, hiszen a többi sor jele nyilván nem változik szomszédos sorok felcserélésének esetén. Legyen (u_i, v_i) az eredeti pár, és (u'_i, v'_i) az új pár.

i $\frac{x}{x}$ sor felcserélése után is $\frac{x}{x}$ marad.

Ez azért igaz, mert v_i független volt az előzményétől, és a cserével a v_i előzménye szűkül, tehát az új u'_i koordinátánk mindenképpen x marad. Ha az u_i egy sorral lejjebb kerülve függővé válna az előzményétől, akkor ez azt jelentené, hogy v_i miatt lett összefüggő. De ez az eset nem lehetséges, hiszen ekkor ez előállítást adna u_i -ra v_i segítségével, ami előállítást biztosítana v_i -re is. És mivel feltettük, hogy v_i független az előtte levő soroktól (azaz x), ezért ez ellentmondás.

ii $\frac{0}{0}$ a cserénk után is $\frac{0}{0}$ marad.

Mivel u_i előzménye bővül, ezért az v'_i az 0 lesz. Mivel v_i előzménye csökken, de u_i már az előtte levők előzményétől már függött (lineáris kombinációjuk volt), ezért v'_i is nulla lesz.

iii $\frac{0}{x}$ a csere után $\frac{x}{0}$ lesz.

Az u_i előzménye bővül és az 0 is volt, ezért v'_i főleg 0 lesz. v_i előzménye viszont csökken ezzel a cserével, ezért u'_i az x lesz.

iv $\frac{x}{0}$ mellé feltesszük azt is, hogy $(u_{i-1}, v_{i-1}) \frac{x}{x}$ volt, és v_i felírható u_i felhasználása nélkül is, akkor a csere után $\frac{0}{x}$ lesz.

A feltétel szerint u_i nem csak a korábbiaktól függ, hanem u_i nélküli korábbiaktól is, azaz létezik egy lineáris kombináció az u_i nélküliekkel felírva. Emiatt v'_i nulla lesz. Az u_i előtti vektorok halmaza viszont csak ezzel a v_i vektorral bővült. S mivel ez a v_i vektor felírható az u_i nélküli előzmények segítségével, és u_i független az u_i előtti soroktól, ezért az új v'_i -nk x lesz.

v $\frac{x}{0}$ a csere után is $\frac{x}{0}$ marad, ha az előző (u_{i-1}, v_{i-1}) párra már nem $\frac{x}{x}$ volt. A u'_i mellé 0 kerül, hiszen az u_{i-1} már függött az előző soroktól, így u_i is függni fog az előzőektől. Az v'_i mellé viszont x kerül, mert a terünk csak ezzel a v_i -vel bővült, ami már az előző soroknak lineáris kombinációja volt, és attól u_i független volt.

vi $\frac{x}{0}$ a csere után is $\frac{x}{0}$ marad, ha az összes előző $\frac{x}{x}$ volt, és a cserélt v_i nem írható fel az u_i felhasználása nélkül.

v'_i mellé 0 kerül, mert v_i segítségével egy új felírását tudjuk u_i -nek megkonstruálni u'_i mellé viszont x kerül, mert a feltevésünk szerint v_i nem írható fel u_i felhasználása nélkül, így mindenképp független lesz az előtte levő soroktól.

vii $\frac{x}{0}$ a csere után is $\frac{0}{x}$ marad, ha az előző (u_{i-1}, v_{i-1}) párra már nem $\frac{x}{x}$ volt.

u'_i mellé nulla kerül, mert u'_i -nek megfelelő sorokról feltettük, hogy már nullára váltottak, és ha egyszer nullára váltanak, akkor nullák is maradnak. Ha v'_i az u_i előzményeiből is felírható lenne, akkor nem váltana x -ről 0-ra u_i , mert cserével nem bővülne az u_i előtti által generált tér. Tehát v_i mellé x -nek kell kerülnie.

A fentebbi esetek mindegyikében változatlan marad a két felcserélt koordináta Kronecker indexeinek a halmaza. Sőt, az utolsó és utolsó előtti esetet kivéve a koordináta viszi is magával a Kronecker indexét. \square

2.3.1. Következmény. *Ha monoton csökkenő sorrendbe szeretnénk rendezni a Kronecker indexeket, akkor az egyértelmű úgy, hogy csak kisebb Kronecker indexeket cserélünk fel nagyobb Kronecker indexűre.*

2.3.3. Megjegyzés. *Ha növekvő sorrendbe szeretnénk rendezni őket, akkor minden esetben meg kellene vizsgálnunk az előző sorokkal kapcsolatos függőséget, emiatt az a módszer körülményes lenne nekünk, amellett, hogy ha az utolsó két eset egyike állna fenn, akkor lehetséges lenne olyan, hogy két koordinátát felcserélünk, de a hozzájuk tartozó Kronecker indexek nem cserélődnek fel. Emiatt a monoton csökkenő sorrendet fogjuk felhasználni a továbbiakban.*

2.3.1. Definíció. Legyen (p_1, \dots, p_v) a Kronecker indexek vektora. Legyen további r egy permutáció. Ekkor $(p_{r(1)}, \dots, p_{r(v)})$ vektort Kronecker invariánsnak nevezzük, ha $p_{r(j)} \geq p_{r(i)}$, akkor $j < i$. Tehát a monoton csökkenő sorrendbe rendezett Kronecker indexeket Kronecker invariánsoknak nevezzük.

2.4. Identifikációs algoritmus stacionárius esetben

Ebben a fejezetben D.S. Poskitt [4] cikke alapján egy olyan algoritmust fogok ismertetni, amely meghatározza a megfigyelt folyamat koordinátáinak Kronecker indexét, és a koordináták azon permutációját amelyek mellett a Kronecker indexek monoton csökkenő sorrendbe rendeződnek.

Az algoritmusunk lényege vázlatosan a következő:

Elsőnek vezessük be a Poskitt[2] féle információs kritériumot, ami a következő:

$$IC_T(n) = T \log \sigma(n) + p_T(d_j(n))$$

Itt $\sigma(n)$ a maximul likelihood, p_t egy Penalty Term függvény. Poskitt[4] szerint ahol ez a függvény felveszi a lokális minimumát, az az n lesz a Kronecker indexünk. Célunk ennek az $IC_T(n)$ függvénynek a kiszámolása lesz lépésenként, majd az összehasonlítása az előző $IC_T(n-1)$ -el. Amíg jobb modellt kapunk az n növelésével addig számoljuk ezt, majd amikor már a bővebb modell nem lesz jobb, mint a szűkebb, akkor leállítjuk a bővítést és elfogadjuk a talált értéket a Kronecker indexünknek.

Ennek az $IC_T(n)$ -nek a kiszámolásához szükségünk van lépésenként egy *ARMAX* modell illesztésére, mivel az *ML* becslésünk számolása lényegében eszerint történik. Ennek az *ARMAX* felírásnak bemutatását, létezését, és fontosabb tulajdonságait a következő alfejezetben fogom taglalni.

Az algoritmusunk kezdetekor vegyünk minden koordinátánknak a nullad fokú *ARMAX* modelljét. Ez megegyezik lényegében az adott koordinátának az összes többire vett regressziójával.

2.4.1. Megjegyzés. Ekkor még nem tudhatjuk, hogy a koordinátánk Kronecker invariánsa 0 vagy sem. Ezt csak akkor tudjuk megállapítani, ha illesztünk egy elsőrendű *ARMAX* modellt, és az lényegesen rosszabbnak bizonyul, mint a nullad rendű. (azaz az információs kritériumunk nőni kezdett)

Következő lépésben minden sorra illesztünk egy 1. rendű *ARMAX* modellt úgy, hogy mintha az az utolsó koordináta volna. Ezt megtehetjük, hiszen az invariáns permutációban 1 Kronecker indexű koordináták után csak a nulla Kronecker indexűek lehetnek, és persze nincs még olyan koordinátánk, amire rábizonyítottuk volna, hogy a Kronecker invariánsa 0. Ha találunk olyan koordinátát, aminek az 1-ed fokú *ARMAX* modellje

rosszabb, mint a 0-ad rendű, akkor a Kronecker invariánsa nulla, és ezt (vagy ezeket) a koordinátákat bemértnek tekintjük és a csökkenő sorrendben levő Kronecker index sorozat (Kronecker invariáns vektor) végére tesszük.

Következő lépésben 2-od fokú *ARMAX* modelleket illesztünk, de úgy, hogy a többi koordináta jelenidejű értékei közül csak azokat vesszük figyelembe, amiknek a Kronecker invariánsát még nem határoztuk meg, azaz nem nulla. Ebben a lépésben azokat a koordinátákat fogjuk megtalálni, amiknek a Kronecker invariánsa 1.

Ezt az algoritmust addig folytatjuk, amíg az összes koordináta esetén megtaláljuk a fokszám szerinti szukcesszív legjobb modellt.

2.4.2. Megjegyzés. *Természetesen az algoritmusunk elején megadunk egy alkalmas h értéket, amit a ciklusunk végértékének tekintünk. Azaz más szóval, ha n eléri az előre megadott h értéket, azaz $n = h$ lesz, akkor a beméretlen koordináták Kronecker invariánsait h -nak vesszük.*

Az algoritmusunk precíz leírását a következő alfejezetekben szemlélhetjük meg. Az algoritmussal kapcsolatos elméleti tételeket az algoritmusok után bemutatom, de nem bizonyítom. A bizonyításukat megtalálhatják D.S. Poskitt [4] cikkében.

2.4.1. Skalár *ARMAX* reprezentáció

Ebben az alfejezetben a később általunk használandó skalár *ARMAX* reprezentációjáról és azokkal kapcsolatos tételről lesz szó. Az identifikációs algoritmusunknak egyik legfontosabb része ez az *ARMAX* reprezentáció. Elsőnek nézzünk meg egy lemmát, ami majd segíthet nekünk az *ARMAX* reprezentációs tételünk kimondásában.

2.4.1. Lemma. *Legyen y_t egy VARMA folyamat, amely teljesíti a Feltevés 1'-et és teljesíti a Feltevés 2'-t. Legyen továbbá az y_t folyamatunk az alábbi alakú:*

$$u_t = \begin{pmatrix} u_{1,t} \\ u_{2,t} \\ \dots \\ u_{v,t} \end{pmatrix} = A(L)y_t$$

Legyen minden $j = 1, \dots, v$ -re $v_t = u_{j,t}$, azaz lépésenként legyen v_t a_j -edik koordinátánk. Ekkor létezik egy nulla várható értékű, η_t fehér zaj folyamat úgy, hogy a szórása σ_η , és v_t felírható az alábbi módon:

$$v_t = \eta_t + \mu_1 \eta_{t-1} + \dots + \mu_n \eta_{t-n}$$

ahol $n = n_j$ és μ_1, \dots, μ_n koefficiensei $\mu(z) - 1 = \sum_{s=1}^n \mu_s z^s$ meghatározzák az autokovariancia függvényét v_t -nek úgy, hogy az megegyezik $\sigma_\eta \mu(z) \mu(z^{-1})$, és $\mu(z) \neq 0$, ha $|z| \leq 1$.

Bizonyítás: Legyen $e_j = (0, \dots, 1, \dots, 0)$ a j -edik v dimenziós bázisvektor. Tudjuk, hogy a spektrál sűrűségfüggvénye $\nu_t = e_j' u_t$ -nek a következő:

$$S_\nu(\omega) = \frac{1}{2\pi} e_j' M(\omega) \Sigma M(\omega)^* e_j.$$

Az autokovariancia generáló függvény hasonlóan:

$$\rho(z) = \sum_{s=-n}^n \rho_s z^s \quad (2.9)$$

$$= e_j' M(z) \Sigma M(z^{-1})' e_j \quad (2.10)$$

ahol $n = n_j$. Az előző felírásból könnyen látszik, hogy $z^n \rho(z)$ -nek $2n$ gyöke van, és $\rho_s = \rho_{-s}$ -el minden $s = 1, \dots, n$ -re a szimmetria miatt. Itt ρ_s -ek valósak, és a gyökei $\rho(z)$ -nek valósak, vagy konjugált párok. Mivel $\rho(\omega) = 2\pi S_\nu(\omega) > 0$, ha $-\pi \leq \omega \leq \pi$, ezért $\rho(z)$ -nek nem lehetnek gyökei az egységkörön. Osszuk a gyököket két halmazra: $\{\zeta_1, \dots, \zeta_n\}$ -ra, és $\{\bar{\zeta}_1^{-1}, \dots, \bar{\zeta}_n^{-1}\}$ -ra. Most konstruáljuk meg a következő $m(z)$ operátort:

$$m(z) = m_0 \prod_{s=1}^n (1 - \zeta_s z)$$

ahol $m_0 = (\prod_s (-\zeta_s) \rho_n)^{\frac{1}{2}}$. Most $\rho(z)$ -t ha felírjuk, akkor a következő eredményeket kapjuk:

$$\rho(z) = \rho_n z^{-n} \prod_{s=1}^n (1 - \zeta_s z) (1 - \bar{\zeta}_s^{-1} z) \quad (2.11)$$

$$= m(z) m(z^{-1}) \quad (2.12)$$

Ekkor kiválaszthatunk n gyököt $\rho(z)$ -ből, hogy megkonstruáljuk $\mu(z)$ -t a következő képen:

$$\mu(z) = 1 + \mu_1 z + \dots + \mu_n z^n$$

úgy, hogy $\rho(z) = \sigma_\eta^2 \mu(z) \mu(z^{-1})$, ahol $\mu(z) = \frac{m(z)}{m(0)}$, és $\sigma_\eta^2 = m_0^2$ és $\mu(z)$ -re igaz, hogy $\mu(z) \neq 0$, ha $|z| \leq 1$. A létezése az η_t -nek következik a ν_t mozgóátlag reprezentációjából. \square

2.4.1. Következmény. *Az előző lemma következménye, hogy minden koordinátájára y_t -nek felírható soronként egy η_t -s reprezentáció egymástól függetlenül. A soronkénti η_t -k viszont nem feltétlenül lesznek egymástól függetlenek ebben a felírásunkban.*

2.4.1. Tétel. *Legyen y_t VARMA_E folyamat, ami teljesíti a Feltétel 1'-et és Feltétel 2-t, és tegyük fel, hogy a koordináták sorrendbe vannak rendezve a Kronecker invariánsok szerint, azaz $y_t = (y_{1,t}, \dots, y_{v,t})' = (y_{r(1),t}, \dots, y_{r(v),t})'$, ahol $y_{r(i)} \leq y_{r(j)}$, ha $j \leq i$, és $n_j = n_{r(j)}$ $j = 1, \dots, v$ esetén. Ekkor minden $j = 1, \dots, v$ -re a j . egyenlete*

a VARMA folyamatunknak ekvivalens a következő skalár ARMAX reprezentációval, ahol $z_t = y_j, t$

$$z_t + \sum_{s=1}^n \alpha_s z_{t-s} + \sum_{i=1}^{j-1} \beta_i y_{i,t} + \sum_{\substack{i=1 \\ i \neq j}}^v \sum_{s=1}^n \beta_{i,s} y_{i,t-s} = \eta_t + \sum_{s=1}^n \mu_s \eta_{t-s} \quad (2.13)$$

ahol $n = n_j$. Legyen továbbá

$$\alpha(z) = 1 + \sum_{s=1}^n \alpha_s z^s$$

és

$$\beta(z) = (\beta_1 + \sum_{s=1}^n \beta_{1,s} z^s, \dots, \beta_{j-1} + \sum_{s=1}^n \beta_{j-1,s} z^s, 0, \sum_{s=1}^n \beta_{j+1,s} z^s, \dots, \sum_{s=1}^n \beta_{v,s} z^s)$$

továbbá $\mu(z) = 1 + \sum_{s=1}^n \mu_s z^s$ relatív prímelek, és $y_{i,t}$ $i = 1, \dots, j-1$, és $y_{i,t-s}$ $i = 1, \dots, v$, $i \neq j$, $s = 1, \dots, n$ koordináták csak a z múltjától függenek, azaz z_t -től nem.

2.4.3. Megjegyzés. Felírhatjuk az adott $\beta(z)$, $\alpha(z)$, $\mu(z)$ segítségével is a reprezentációs egyenletünket az alábbi módon:

$$\alpha(z)z_t + \beta(z)y_t = \mu(z)\eta_t$$

Bizonyítás: Mivel y_t inverz Echelon formában van, ezért $M_0 = A_0$ egy alsó háromszög mátrix. Ezért rekurzívan megadhatjuk az első sortól kezdve a reprezentációkat az utolsóig. Legyen $a(z)$ az $A(z)$ -nek a j . sora. Azaz formálisan legyen $a(z) = e_j' A(z)$. Ekkor a j . egyenlet a következő:

$$a(L)y_t = y_{j,t} + \sum_{i=1}^{j-1} a_{ji,0} y_{i,t} + \sum_{i=1}^v \sum_{s=1}^{n_j} a_{ji,s} y_{i,t-s} = u_{j,t}.$$

Az egyenletünk végén álló $u_{j,t}$ felírható nekünk a Lemma 2.4.1 alapján az alábbi módon:

$$u_{j,t} = \nu_t = \eta_t + \mu_1 \eta_{t-1} + \dots + \mu_n \eta_{t-n} = 1 + \sum_{i=1}^n \mu_i \eta_{t-i}$$

Legyen $z_t = y_{j,t}$, és rendezzük át a fentebbi egyenletünket úgy, hogy $\alpha_s = a_{jj,s}$, $s = 1, \dots, n = n_j$, $\beta_i = a_{ji,0}$, $i = 1, \dots, v$, $i \neq j$, $s = 1, \dots, n$. Ekkor megkapjuk a tételünkben szereplő 2.13 egyenletet.

Ahhoz, hogy belássuk, hogy $a(z)$, $\mu(z)$ relatív prímelek és ezáltal a belőlük konstruált $\alpha(z)$, $\beta(z)$ is tegyük fel indirekt, hogy nem relatív prímelek. Ekkor a közös osztókat eltüntethetjük $a(L)y_t = \mu(L)\eta_t$ -ből, ami után egy $\bar{a}(L)y_t = \bar{\mu}(L)\eta_t$ -t kapunk, ahol $[\bar{a}(z) : \bar{\mu}(z)]$ foka

kisebb lesz, mint n_j . Most használjuk fel az alap mátrixaink $1, \dots, j-1, j+1, \dots, v$ sorait és $\bar{a}(L)y_t = \bar{\mu}(L)\eta_t$ -t, hogy megkonstruáljuk $[\bar{A}(z) : \bar{M}(z)] \in ARMA_E(\nu)$ mátrixunkat, ahol $\nu = (\bar{n}_1, \dots, \bar{n}_v)$ úgy, hogy $\bar{n}_j \leq n_j$, emellett $[A(z) : M(z)] = \bar{U}(z) [\bar{A}(z) : \bar{M}(z)]$, ahol $\bar{U} \neq I$ unimoduláris mátrix. Ez viszont ellent mond a Feltevés 3'-nak! Tehát az állításunk igaz.

Végül nyilvánvaló, hogy $y_{i,t-s}$ $i = 1, \dots, v, i \neq j, s = 1, \dots, n$ értékei nem függenek z_t -től. Hasonlóan igaz $y_{i,t}$ -re is, ha $i = 1, \dots, j-1$. Ez amiatt igaz, mert a struktúránk rekurzív és ortogonalizálhatjuk az innováció folyamatunkat úgy, hogy közben megtartjuk a rekurzív struktúrát, és a mozgó átlagunk sor rangját. Ahhoz, hogy ezt igazoljuk tekintsük Σ Cholesky felbontását. A Cholesky felbontás alkalmazható, mivel szimmetrikus, és pozitív szemidefinit mátrixunk van.

$$\Sigma = CDC'$$

ahol C egy alsó háromszög mátrix egység diagonális elemekkel, és $D = \text{diag}(d_1^2, \dots, d_v^2)$. Ekkor $w_t = C^{-1}\varepsilon_t$ martingál differencia fejlődési folyamat, D kovariancia mátrixszal. Ennek segítségével újra írhatjuk az $ARMA$ folyamatunkat az alábbi módon:

$$A_0 y_t + \sum_{s=1}^p A_s y_{t-s} = L_0 w_t + \sum_{s=1}^p L_s w_{t-s} \quad (2.14)$$

itt $L_0 = M_0 C$, és ez természetesen megint egy alsó háromszög mátrix lesz 1-esekkel a diagonálisban, mivel alsó háromszög mátrixok szorzata alsó háromszög mátrix lesz, és a diagonális a diagonálisok szorzata lesz. Emellett $L_s = M_s C$. Tekintsük a j . egyenletünket 2.14-ből:

$$u_{j,t} = w_{j,t} + \sum_{i=1}^{j-1} l_{ji,0} w_{i,t} + \sum_{i=1}^v \sum_{s=1}^{n_j} l_{ji,s} w_{i,t-s}.$$

Ezért az előző egyenletünkből látszik, hogy $y_{i,t}$ azaz z_t legfeljebb $w_{1,t}, \dots, w_{i-1,t}$ -től függhet. A konstrukcióból pedig adódik, hogy w_t elemei korrelálatlanok, és $E(y_{i,t} w_{j,t} | \mathcal{F}) = 0$.

□

2.4.4. Megjegyzés. Az előző Tétel szerint két érdekes megfigyelést tehetünk. Első, hogy a zaj operátor fokszáma csak az adott sor Kronecker invariánsán múlik.

Az egyidejű komponenseink csak az előtte lévő sorok értékétől függenek. Ami azt jelenti, hogy csak a nagyobb Kronecker invariánsú tagoktól függenek. Ez azt jelenti, hogy a Kronecker invariánsát $y_{r(j),t}$ -nek, ha tudjuk, akkor az elárulja nekünk, hogy milyenek lesznek a zajhosszúságaink az ARMAX reprezentációjában $y_{r(j),t}$ -nek, és mivel tudjuk, hogy a Kronecker invariáns függ a többi indexünktől, így $n_{r(j)} \leq n_{r(i)}$, $i = j+1, \dots, v$ plusz információkat ad nekünk a rekurziós struktúránkról. A pontos értéke a rákövetkező Kronecker invariánsoknak persze nem szükséges explicit nekünk, elég lesz nekünk a j . sor ARMAX

reprezentációjához a következő $d_j(n) = (j-1) + (v+1)n$ függvény. Ezt a függvényt *Penalty Term* függvénynek nevezzük. Ezt fogjuk felhasználni a következő fejezet részben a Kronecker invariánsaink megtalálására.

2.4.2. Az algoritmus lépései stacionárius esetben

Az algoritmusunk precíz leírását ebben a fejezetben taglalom.

Legyenek y_t -k $t = 1, \dots, T$ a megfigyeléseink, ahol y_t v -dimenziós VARMA folyamat, ami az alábbi módon áll elő:

$$A(L)y_t = M(L)\varepsilon_t$$

Ez az y_t teljesíti a Feltétel 1'-t és a Feltétel 3'-mat. Legyen $P(1, \dots, v)' = (r(1), \dots, r(v))'$ permutáció leképezés. Ekkor $P(y_t) = (y_{r(1),t}, \dots, y_{r(v),t})$. Ezzel a permutáció leképezéssel szeretnénk monoton csökkenő sorrendbe rendezni a Kronecker indexeinket. A következő algoritmusunk meghatározza a Kronecker invariánsokat, míg a P permutáció leképezést lépésenként változtatjuk. Az algoritmusban felhasználjuk a Tétel 2.13 *ARMAX* reprezentációját is.

A stacionárius algoritmus lépései

i Inicializáció:

Legyenek a kezdeti értékek:

- $n = 1$ - az aktuális Kronecker indexünk.
- $j = v$ - az aktuálisan vizsgált koordináta (a végétől kezdjük).
- $P = I$ - kezdeti permutáció mátrixunk, ahol I az identitásmátrix.
- $\mathcal{N} = \{1, \dots, v\}$ indexhalmaz

Számoljuk ki a tapasztalati átlaggal korrigált y_t -ket, azaz $\bar{y}_t = y_t - \bar{y}$, $t = 1, \dots, T$, ahol $\bar{y} = \frac{\sum_{j=1}^T y_t}{T}$. Ezt az \bar{y} -t szeretnénk a továbbiakban használni. Ekkor minden $i \in \mathcal{N}$ -re számítsuk ki $\bar{\sigma}_{\eta,i}^2(0)$ -t, amit $\bar{y}_{k,t}$ $\bar{y}_{i,t}$ -re való regressziójának a segítségével kaphatunk meg, ahol $k = 1, \dots, v, k \neq i$. Ez lesz a kezdeti célfüggvényünk.

ii **WHILE** $j \geq 1$ esetben az alábbi lépést végezzük el:

(a) **FOR** $i(k) \in \mathcal{N}$ $k = 1, \dots, v$

- i. $\tilde{y}_t = E_{i(k),j}(Py_t)$, ahol az $E_{a,b}$ operátor felcseréli az a és b sorokat. Ezután számítsuk ki a j -edik sor *ARMAX* reprezentációját.

- A. Legyen $\tilde{A}(z) = E_{i(k),j} P A(z) P' E'_{i(k),j}$ és $\tilde{a}(z)$ legyen az $\tilde{A}(z)$ j -edik sora, azaz $\tilde{a}(z) = e'_j \tilde{A}(z)$. Tekintsük a tapasztalati autokovarianciafüggvényünket, ahol $r = 1, \dots, T - 1$:

$$\tilde{C}_y(r) = \tilde{C}_y(-r)' = \frac{\sum_{t=1}^{T-|r|} \tilde{y}_t \tilde{y}'_{t-r}}{T}$$

Célunk az alábbi egyenletekkel kiszámolni az $\hat{\tilde{a}}_s$ együtthatókat:

$$\sum_{s=0}^n \hat{\tilde{a}}_s \tilde{C}_y(r+s) = 0, r = n+1, \dots, 2n$$

Ezt azért tehetjük meg, mert tudjuk, hogy

$$\sum_{s=0}^n \hat{\tilde{a}}_s \tilde{\Gamma}_y(r+s) = 0, r = n+1, \dots, 2n$$

Ezeket hívjuk Yule-Walker egyenleteknek. Ezeknek az empirikus változata a fentebb szereplő \tilde{C}_y tapasztalati kovarianciafüggvényes egyenlet.

Ezek v -dimenziós vektorok, melyek az alábbi módon állnak elő:

$\hat{\tilde{a}}_0 = (\hat{\tilde{a}}_{1,0}, \dots, \hat{\tilde{a}}_{j-1,0}, 1, 0, \dots, 0)$ és $\hat{\tilde{a}}_s = (\hat{\tilde{a}}_{1,s}, \dots, \hat{\tilde{a}}_{v,s})$, ahol $s = 1, \dots, n$.

Ezután határozzuk meg az *ARMAX* reprezentációbeli α és β együtthatókat a következőképpen ($s = 1, \dots, n$):

$$\hat{\alpha}_s = \hat{\tilde{a}}_{j,s} \quad (2.15)$$

$$\hat{\beta}_i = \hat{\tilde{a}}_{i,0}, i = 1, \dots, j-1 \quad (2.16)$$

$$\hat{\beta}_{i,s} = \hat{\tilde{a}}_{i,s}, i = 1, \dots, v, i \neq j \quad (2.17)$$

- B. A következőekben $r = 1, \dots, n$ becsüljük meg a ν folyamatunk autokovarianciafüggvényét az alábbi módon:

$$\tilde{C}_\nu(r) = \tilde{C}_{-\nu}(r) = \sum_{s=0}^n \sum_{u=0}^n \hat{\tilde{a}}_s \tilde{C}_y(r+s-u) \hat{\tilde{a}}_u' \quad (2.18)$$

$$= \int_{-\pi}^{\pi} \hat{\tilde{a}}(\omega) \tilde{I}_y(\omega) \hat{\tilde{a}}(\omega)' \exp(-i\omega r) d\omega \quad (2.19)$$

Ahol az $\hat{\tilde{a}}(\omega) = \hat{\tilde{a}}_0 + \hat{\tilde{a}}_1 \exp(i\omega) + \dots + \hat{\tilde{a}}_n \exp(in\omega)$ és

$$\tilde{I}_y(\omega) = \frac{1}{2\pi} \sum_{s=-T+1}^{T-1} \tilde{C}_y(s) \exp(i\omega s)$$

Ezután definiáljuk a ν spektrumát:

$$\hat{\tilde{S}}_\nu(\omega) = \frac{1}{2\pi} \sum_{s=-n}^n \tilde{C}_\nu(s) \exp(i\omega s)$$

Most számítsuk ki $\hat{\mu}_s, s = 1, \dots, n$ mozgóátlag együtthatóinkat:

$$\sum_{s=0}^n \hat{\mu}_s \tilde{C}_{i\nu}(l+s) = 0, \text{ ahol } l = 1, \dots, n$$

Itt:

$$\tilde{C}_{i\nu}(r) = \int_{-\pi}^{\pi} \frac{\hat{a}(\omega) \tilde{I}_y(\omega) \hat{a}(\omega)^*}{\hat{S}_\nu(\omega)^2} \exp(-ir\omega) d\omega$$

ii. Számítsuk ki a zaj szórásának a Pszeudo Maximum Likelihood becslését.

A. Legyen $\hat{\eta}_t$ az *ARMAX* reprezentáció jobb oldala. A következő módon adható meg $\hat{\eta}_t$:

$$\hat{\eta}_t = \sum_{s=0}^n \hat{a}_{j,s} \tilde{y}(t-s) - \sum_{s=1}^n \hat{\mu}_s \hat{\eta}_{t-s} \quad (2.20)$$

$$= \sum_{s=0}^n \hat{\alpha}_s z_{t-s} + \sum_{i=1}^{j-1} \hat{\beta}_i \tilde{y}_{i,t} + \sum_{\substack{i=1 \\ i \neq j}}^v \sum_{s=1}^n \hat{\beta}_{i,s} \tilde{y}_{i,t-s} - \sum_{s=1}^n \hat{\mu}_s \hat{\eta}_{t-s} \quad (2.21)$$

Konstruáljuk meg a következő $v+1$ dimenziós $(\hat{\xi}_t, \hat{\phi}_t)'$ folyamatunkat az alábbi módon:

$$\begin{bmatrix} \hat{\xi}_t \\ \hat{\phi}_t \end{bmatrix} + \sum_{s=1}^n \hat{\mu}_s \begin{bmatrix} \hat{\xi}_{t-s} \\ \hat{\phi}_{t-s} \end{bmatrix} = \begin{bmatrix} \tilde{y}_t \\ \hat{\eta}_t \end{bmatrix}$$

Itt $\hat{\xi}_t$, és $\hat{\phi}_t$ -t a következő módon kaphatjuk meg koordinátánként:

$$\frac{\partial \hat{\eta}_t}{\partial \hat{\alpha}_s} = \hat{\xi}_{j,t-s} \quad (2.22)$$

$$\frac{\partial \hat{\eta}_t}{\partial \hat{\beta}_{i,s}} = \hat{\xi}_{i,t-s} \quad (2.23)$$

$$\frac{\partial \hat{\eta}_t}{\partial \hat{\mu}_s} = -\hat{\phi}_{t-s} \quad (2.24)$$

És a rekurziónk $\hat{\eta}_t = 0$ és $(\hat{\xi}_t, \hat{\phi}_t)' = 0$ kezdeti értékekből indul, azaz $t \leq 0$ -ra azonosan 0 értékeket vesz fel.

2.4.5. Megjegyzés. Az előbbi $\hat{\xi}$ és $\hat{\phi}$ felírásából egyszerűen látszódik, hogy

$$\frac{\partial \sum_t^T \hat{\eta}_t^2}{\partial \hat{a}_j} = \sum_{t=1}^T \hat{\xi}_{t-j} \hat{\eta}_t \quad (2.25)$$

$$\frac{\partial \sum_t^T \hat{\eta}_t^2}{\partial \hat{\mu}_j} = \sum_{t=1}^T \hat{\phi}_{t-j} \hat{\eta}_t \quad (2.26)$$

- B. Ebben a lépésben az előző lépésben megemlített $\frac{\sum_t \hat{\eta}_t^2}{T}$ -t szeretnénk minimalizálni Hannan Rissanen [5] cikke alapján Gauss-Newton eljárással.
 Legyen $\tilde{c}_{\hat{\eta}}(0) = \frac{\sum_{t=1}^T \hat{\eta}_t^2}{T}$.

$$\tilde{c}_{\hat{\eta}\hat{\xi}}(s) = \frac{\sum_{t=s+1}^T \hat{\eta}_t \hat{\xi}_{t-s}}{T}$$

$$\tilde{c}_{\hat{\eta}\hat{\phi}}(s) = \frac{\sum_{t=s+1}^T \hat{\eta}_t \hat{\phi}_{t-s}}{T}$$

A következő lépésben számítsuk ki a négyzetes hibánkat:

$$\hat{\sigma}_\eta^2(n) = \tilde{c}_{\hat{\eta}}(0) + \sum_{s=0}^n \Delta \hat{a}_s \tilde{c}_{\hat{\eta}\hat{\xi}}(s) - \sum_{s=1}^n \Delta \hat{\mu}_s \tilde{c}_{\hat{\eta}\hat{\phi}}(s)$$

Itt $\Delta \hat{a}_s$ Toeplitz regressziója $\hat{\eta}$ -nek $-\hat{\xi}_{i,t}$ -re, és $\hat{\xi}_{t-s}$ -re, ahol $i = 1, \dots, j-1$, $s = 1, \dots, n$. Az iterációnkat (Gauss-Newton) addig folytatjuk, amíg nem konvergálunk, azaz nem kerülünk igényeinknek megfelelően közel. Mivel a $\frac{\sum_t \hat{\eta}_t^2}{T}$ kvázi kvadratikus, ezért Poskitt [4] alapján elég nagy T értékekre a megfelelő pontossághoz nem szükséges két-három iterációnál több nekünk.

- iii. Most számítsuk ki az információs kritérium függvényünket:

$$IC_T(n) = T \log \hat{\sigma}_\eta^2(n) + p_T(d_j(n))$$

Itt a $p_T(d_j(n))$ a Penalty Term függvényünk, ami pozitív valós értékű, monoton növekvő $d_j(n)$ -ben és nem csökkenő T -ben.

- (b) **IF** $IC_T(n) > IC_T(n-1)$

Akkor legyen:

- $r(j) = i(k)$
- $n_{r(j)} = n - 1$
- $P = E_{i(k),j} P$
- $\mathcal{N} = \mathcal{N} \setminus i(k)$
- $j = j - 1$
- **end**

- (c) **end**

- iii **FOR** $i(k) \in \mathcal{N}$

- **IF** $n < h_t$ (előre megadott Kronecker felső korlát)
 $n = n + 1$
- **else**
 - **FOR** $i(k) \in \mathcal{N}, k = 1, \dots, v$
 - $r(j) = i(k)$
 - $n_{r(j)} = n$
 - $P = E_{i(k),j}P$
 - $\mathcal{N} \setminus i(k)$
 - $j = j - 1$
 - **end FOR** $i(k) \in \mathcal{N}$
- **end**

iv **end** when $j = 0$

2.4.3. Algoritmussal kapcsolatos tételek

Ebben a fejezetben Poskitt [4] algoritmussal kapcsolatos két tételét fogom ismertetni, amely biztosítja az algoritmusunk működését, és a Kronecker invariánsaink konvergenciáját. Az $r(i)_T$ az algoritmusunk által talált permutáció, $r(i)^0$ a valódi, azaz az elméleti permutáció a tételekben.

2.4.2. Tétel. *Tegyük fel, hogy y_t VARMA folyamat, ami teljesíti a Feltétel1'-t, és Feltétel3'-mat. Emellett legyenek a Kronecker invariáns párok, amiket az algoritmusunkból kaptunk: $(r(j)_T, n_{r(j)_T})$ -k. Legyen emellett $p_T(d_j(n))$ n és T sztochasztikus függvénye. Ekkor:*

- *ha $(r(i)_T, n_{r(i)_T}) = (r(i)^0, n_{r(i)^0})$, $i = q + 1, \dots, v$, és $p_T(d_q(n))/T \rightarrow 0$ majdnem mindig, ha $T \rightarrow \infty$, akkor $n_{r(q)T} \geq n_{r(q)}^0$.*
- *ha $(r(i)_T, n_{r(i)_T}) = (r(i)^0, n_{r(i)^0})$, $i = q + 1, \dots, v$, és $\liminf_{T \rightarrow \infty} p_T(d_q(n))/L(T) > 0$ majdnem mindig, ahol $L(T)$ egy valós értékű $\log \log T/L(T) \rightarrow 0$ nagyságrendben növekvő függvénye T -nek, ekkor $P(\lim_{T \rightarrow \infty} n_{r(q)T} \leq n_{r(q)}^0) = 1$.*

2.4.3. Tétel. *Tegyük fel, hogy y_t VARMA folyamat, ami teljesíti a Feltétel1'-t, és Feltétel3'-mat. Emellett legyenek a Kronecker invariáns párok, amiket az algoritmusunkból kaptunk: $(r(j)_T, n_{r(j)_T})$ -k. Ha $p_T(d_q(n))/T \rightarrow 0$ és $\log \log T/p_T(d_q(n)) \rightarrow 0$, ha $T \rightarrow \infty$, ekkor $r(j)_T = r(j)^0$ m.m. elég nagy T -re, és $P(\lim_{t \rightarrow \infty} n_{r(j)T} = n_{r(j)}^0) = 1$, $j = 1, \dots, v$ -re.*

2.5. Identifikációs algoritmus kointegrált esetben

2.5.1. Algoritmusunk adaptálása kointegrált folyamatok esetén

Ebben a fejezetben a kointegrált y_t folyamatokra próbáljuk meg alkalmazni az algoritmusunkat. Ehhez szükségünk lesz Engle-Granger [8] féle Error Correction reprezentációra, és a kointegrált *ARMAX* reprezentációra, és a Poskitt féle [3] kanonikus korrelációs felírására, amivel meghatározzuk a kointegrációs térünk bázisát, és a kanonikus korrelációs egyenletünk sajátértékeinek, sajátvektorainak a segítségével meghatározzuk majd a kointegrációs rangunkat.

Elsőnek tegyük fel, hogy $A(z)$ -nek $v - \rho$ gyöke van az egységkörön, és az összes többi gyökünk az egységkörön kívül helyezkedik el. Emiatt $\det A(z) = (1 - z)^{v-\rho} a_s(z)$, ahol $a_s(z) \neq 0$, $|z| \leq 1$, és $\rho < v$. Alkalmazzuk a Beveridge-Nelson dekompozíciót, így $A(z) = B(z)(1 - z) + A(1)z$, ahol $B(z) = -B_0 + B_1z + \dots + B_{p-1}z^{p-1}$. Itt $B_0 = A_0$, és $B_i = -(A_{i+1} + \dots + A_p)$, $i = 1, \dots, p - 1$. Ez pont a Engle-Granger féle *EC* reprezentációhoz vezet, azaz:

$$B(L)\Delta y_t + CEy_{t-1} = M(L)\varepsilon_t.$$

Itt $A(1) = \sum_{s=0}^p A_s = CE$, ahol C és E' $v \times \rho$ dimenziós teljes rangú mátrixok.

Tekintsük most az *EC* reprezentációt, ahol $[B(z) : M(z)]$ teljesíti az alábbi feltételeket:

- i $b_{rc,0} = m_{rc,0}$,
- ii $m_{rr}(z) = 1 + m_{rr,1}z + \dots + m_{rr,n_r}z^{n_r}$,
- iii $m_{rc}(z) = m_{rc,n_r-n_{rc}+1}z^{r_c,n_r-n_{rc}+1} + \dots + m_{rc,n_r}z^{n_r}$,
- iv $b_{rc}(z) = b_{rc,0} + b_{rc,1}z + \dots + b_{rc,n_r}z^{n_r-1}$,
- v E row reduced Echelon formában van.

Amikor ezek a feltételek teljesülnek a fentebbi y_t folyamatra, akkor ezt *ECARMA_E(ν, ρ)* formájúnak nevezzük, ahol ρ a kointegrációs rangja a struktúránknak.

A row reduced Echelon formában egy mátrix akkor van, ha Echelon formában van, és minden sor első nem nulla eleme 1. Emellett minden oszlopban, ahol a sor első nem nulla eleme van (azaz egyes), abban az összes többi elem (felette is és alatta is) azonosan nulla. Most definiáljunk Poskitt [4] cikke szerint egy T $v \times v$ dimenziós nem szinguláris transzformáció mátrixot, ahol $T = [T'_c : T'_u]'$ úgy, hogy $A(z) = A_s(z)T^{-1}\Delta(z)T$, ahol $\det A_s(z) = a_s(z)$, $\Delta(z) = [I_\rho : I_{v-\rho}(1 - z)]$ polinom blokkmátrix. Ha $x_t = [x'_{ct} : x'_{ut}]' = Ty_t$, akkor előző egyenletünk kis átrendezése után azt kapjuk, hogy $\Delta(L)x_t = TA_s(L)^{-1}M(L)\varepsilon_t$, és az első ρ sorában x_t -nek, az x_{ct} változók aszimptotikusan stacionárius folyamatok, és a megmaradt $v - \rho$ sorunk első $(1 - L)$ differenciálás után stacionárius.

2.5.1. Megjegyzés. $A(1)$ -re igaz a fentiek alapján, hogy:

$$A(1) = A_s(1)T^{-1}\Delta(1)T = FT_c$$

ahol $A(1)$ megegyezik az együttható mátrixok összegével.

2.5.1. Tétel. Legyen y_t $ECARMA_E(\nu, \rho)$ folyamat, ami teljesíti Feltétel1'-t, és Feltétel2'-t. Emellett tegyük fel, hogy a változók a Kronecker invariáns sorrendbe vannak rendezve. Ekkor minden $\Delta z_t = \Delta y_{jt}$, $j = 1, \dots, v$ -re létezik relatív prím $\alpha(z)$, $\beta(z)$, $\mu(z)$ $n = n_j$ rendű operátor, és c együtthatóvektor, hogy Δz_t -re igaz az alábbi reprezentáció:

$$\alpha(L)\Delta z_t + \beta(L)\Delta y_t + c'x_{c,t-1} + \mu(L)\eta_t \quad (2.27)$$

ahol $x_t = T_c y_t$. Emellett az is igaz, hogy $\Delta y_{i,t}$ $i = 1, \dots, j-1$, és $\Delta y_{i,t-s}$ $i = 1, \dots, v$, $i \neq j$, $s = 1, \dots, n$, $x_{c,t-1}$ koordináták csak a Δz múltjától függenek, azaz Δz_t -től nem.

2.5.2. A kointegrált algoritmus lépései

Az algoritmusunk precíz leírását kointegrált esetben ebben a fejezetben fogom taglalni. Legyenek y_t -k $t = 1, \dots, T$ a megfigyeléseink, ahol y_t v -dimenziós VARMA folyamat, ami az alábbi módon áll elő:

$$A(L)y_t = M(L)\varepsilon_t$$

Ez az y_t teljesíti a Feltétel 1'-t és a Feltétel 2'-t. Legyen $P(1, \dots, v)' = (r(1), \dots, r(v))'$ permutáció leképezés. Ekkor $P(y_t) = (y_{r(1),t}, \dots, y_{r(v),t})$. A következő algoritmusunk meghatározza a Kronecker invariánsokat, míg a P permutáció leképezést is változtatjuk. Az algoritmusban felhasználjuk a Tétel 2.13 $ARMAX$ reprezentációját is.

Első fölépés:

Határozzuk meg a kointegrációs rangunkat, és a kointegrációs terünk bázisát. Ezeket a következő sajátérték és sajátvektorfeladat megoldásával kaphatjuk meg:

$$[\lambda I_v - C_y(0)^{-1}C_y(1)C_y(0)^{-1}C_y(1)']v_t = 0$$

Rendezzük sorba ezeket a $[\lambda_{i,T}, v_{i,T}]$, $i = 1, \dots, v$ párokat úgy, hogy $\lambda_{1,T} \leq \lambda_{2,T} \leq \dots \leq \lambda_{v,T}$. Határozzuk meg $\rho = 0, \dots, v-1$ -re az alábbi függvény értékét:

$$\nabla_T(\rho) = T(v - \rho) \log \left(\sum_{j=\rho+1}^v \frac{\lambda_{j,T}}{v - \rho} \right) - T \sum_{j=\rho+1}^v \log \lambda_{j,T} + \quad (2.28)$$

$$= \rho(2v - \rho + 1) \log \log \frac{T}{2} \quad (2.29)$$

Ezután határozzuk meg ennek a függvénynek a minimumát ρ szerint.

$$\rho_y = \min_{0 \leq \rho < v} \nabla_T(\rho)$$

Ezzel a ρ_y -al fogjuk megbecsülni a kointegrációs rangunkat Poskitt [3] cikke alapján.

Legyen a bázisunk $B_T = [v_{1,T} : \dots : v_{\rho_y,T}]'$. Evvel a B_T -vel fogunk dolgozni a továbbiakban.

Második főlépés:

i Inicializáció:

Legyenek a kezdeti értékek:

- $n = 1$ - a teszt Kronecker indexünk.
- $j = v$ - a jelenleg vizsgált koordináta (a végéről kezdjük).
- $P = I$ - kezdeti permutáció mátrixunk, ahol I az identitásmátrix.
- $\mathcal{N} = \{1, \dots, v\}$ indexhalmaz.

Számoljuk ki az első differenciált értékét y_t -nek, azaz $\Delta y_t = y_t - y_{t-1}$ -et. Emellett legyen $\zeta_t = B_T y_t$, minden $t = 1, \dots, T$. Ekkor minden $i \in \mathcal{N}$ -re számítsuk ki $\tilde{\sigma}_{\eta,i}^2(0)$ -t, amit $\bar{y}_{k,t}$ $\bar{y}_{i,t}$ -re való regressziójának a segítségével kaphatunk meg, ahol $k = 1, \dots, v, k \neq i$. Ez lesz a kezdeti célfüggvényünk.

ii **WHILE** $j \geq 1$ esetben az alábbi lépést végezzük el:

(a) **FOR** $i(k) \in \mathcal{N}$ $k = 1, \dots, v$

i. $\Delta \tilde{y}_t = E_{i(k),j}[P\Delta y_t]$, ahol az $E_{a,b}$ operátor felcseréli az a és b sorokat. Ezután számítsuk ki a j -edik sor *ECARMAX* reprezentációját.

A. Legyen $\tilde{D}(z) = [E_{i(k),j}PB(z)P'E'_{i(k),j}, E_{i(k),j}PC]$ és $\tilde{d}(z)$ legyen az $\tilde{D}(z)$ j -edik sora, azaz $\tilde{d}(z) = e'_j\tilde{D}(z)$. Tekintsük a \tilde{w}_t tapasztalati autokovarianciafüggvényünket, ahol $\tilde{w}_t = [\Delta y'_t, \zeta'_{t-1}]'$, és $r = 1, \dots, T-1$:

$$\tilde{C}_w(r) = \tilde{C}_w(-r)' = \frac{\sum_{t=1}^{T-|r|} \tilde{w}_t \tilde{w}'_{t-r}}{T}$$

Számoljuk ki ezután az alábbi függvényt:

$$\tilde{I}_y(\omega) = \frac{1}{2\pi} \sum_{s=-T+1}^{T-1} \tilde{C}_y(s) \exp(i\omega s)$$

Ennek segítségével megoldhatjuk a következő egyenletet:

$$\int_{-\pi}^{\pi} \hat{d}(\omega) \tilde{I}_w(\omega) \exp(-i\omega r) d\omega = 0, r = n+1, \dots, 2n$$

Ezután határozzuk meg az *ECARMAX* reprezentációbeli α és β együtthatókat a következőképpen ($s = 1, \dots, n$):

$$\hat{\alpha}_s = \hat{d}_{j,s} \quad (2.30)$$

$$\hat{\beta}_i = \hat{d}_{i,0}, i = 1, \dots, j-1 \quad (2.31)$$

$$\hat{\beta}_{i,s} = \hat{d}_{i,s}, i = 1, \dots, v, i \neq j, r = \hat{d}_{v+r,1}, r = 1, \dots, \rho_y \quad (2.32)$$

B. A következőekben $r = 1, \dots, n$ becsljük meg a ν folyamatunk autokovarianciafüggvényét az alábbi módon:

$$\tilde{C}_\nu(r) = \tilde{C}_{-\nu}(r) = \sum_{s=0}^n \sum_{u=0}^n \hat{d}_s \tilde{C}_w(r+s-u) \hat{d}'_u \quad (2.33)$$

$$= \int_{-\pi}^{\pi} \hat{d}(\omega) \tilde{I}_w(\omega) \hat{d}'(\omega) \exp(-i\omega r) d\omega \quad (2.34)$$

Ezután definiáljuk a ν spektrumát:

$$\hat{S}_\nu(\omega) = \frac{1}{2\pi} \sum_{s=-n}^n \tilde{C}_\nu(s) \exp(i\omega s)$$

Most számítsuk ki $\hat{\mu}_s, s = 1, \dots, n$ mozgóátlag együtthatóinkat:

$$\sum_{s=0}^n \hat{\mu}_s \tilde{C}_{i_\nu}(l+s) = 0, \text{ ahol } l = 1, \dots, n$$

Itt:

$$\tilde{C}_{i_\nu}(r) = \int_{-\pi}^{\pi} \frac{\hat{d}(\omega) \tilde{I}_w(\omega) \hat{d}'(\omega)^*}{\hat{S}_\nu(\omega)^2} \exp(-ir\omega) d\omega$$

ii. Számítsuk ki a zaj szórásának a Pseudo Maximum Likelihood becslését.

A. Legyen $\hat{\eta}_t$ az *ECARMAX* reprezentáció jobb oldala. A következő módon adható meg $\hat{\eta}_t$:

$$\hat{\eta}_t = \hat{d}(L) \tilde{w}_t + (1 - \hat{\mu}(L)) \hat{\eta}_t \quad (2.35)$$

Konstruáljuk meg a következő $v + \rho_y + 1$ dimenziós $(\hat{\xi}_t, \hat{\phi}_t)'$ folyamatunkat az alábbi módon:

$$\begin{bmatrix} \hat{\xi}_t \\ \hat{\phi}_t \end{bmatrix} + \sum_{s=1}^n \hat{\mu}_s \begin{bmatrix} \hat{\xi}_{t-s} \\ \hat{\phi}_{t-s} \end{bmatrix} = \begin{bmatrix} \tilde{w}_t \\ \hat{\eta}_t \end{bmatrix}$$

Itt $\hat{\xi}_t$, és $\hat{\phi}_t$ -t a következő módon kaphatjuk meg koordinátánként:

$$\frac{\partial \hat{\eta}_t}{\partial \hat{w}_{j,s}} = \hat{\xi}_{j,t-s} \quad (2.36)$$

$$\frac{\partial \hat{\eta}_t}{\partial \hat{\mu}_s} = -\hat{\phi}_{t-s} \quad (2.37)$$

És a rekurzióink $\hat{\eta}_t = 0$ és $(\hat{\xi}_t, \hat{\phi}_t)' = 0$ kezdeti értékekből indul, azaz $t \leq 0$ -ra azonosan 0 értékeket vesz fel.

2.5.2. Megjegyzés. Az előbbi $\hat{\xi}$ és $\hat{\phi}$ felírásából egyszerűen látszódik, hogy

$$\frac{\partial \sum_t^T \hat{\eta}_t^2}{\partial \hat{a}_j} = \sum_{t=1}^T \hat{\xi}_{t-j} \hat{\eta}_t \quad (2.38)$$

$$\frac{\partial \sum_t^T \hat{\eta}_t^2}{\partial \hat{\mu}_j} = \sum_{t=1}^T \hat{\phi}_{t-j} \hat{\eta}_t \quad (2.39)$$

B. Ebben a lépésben az előző lépésben megemlített $\frac{\sum_t^T \hat{\eta}_t^2}{T}$ -t szeretnénk minimalizálni Hannan Rissanen [5] cikke alapján Gauss-Newton eljárással.

Legyen $\tilde{c}_{\hat{\eta}}(0) = \frac{\sum_{t=1}^T \hat{\eta}_t^2}{T}$.

$$\tilde{c}_{\hat{\eta}\hat{\xi}}(s) = \frac{\sum_{t=s+1}^T \hat{\eta}_t \hat{\xi}_{t-s}}{T}$$

$$\tilde{c}_{\hat{\eta}\hat{\phi}}(s) = \frac{\sum_{t=s+1}^T \hat{\eta}_t \hat{\phi}_{t-s}}{T}$$

A következő lépésben számítsuk ki a négyzetes hibánkat:

$$\hat{\sigma}_\eta^2(n) = \tilde{c}_{\hat{\eta}}(0) + \sum_{s=0}^n \Delta \hat{d}_s \tilde{c}_{\hat{\eta}\hat{\xi}}(s) - \sum_{s=1}^n \Delta \hat{\mu}_s \tilde{c}_{\hat{\eta}\hat{\phi}}(s)$$

Itt $\Delta \hat{d}_s$ Toeplitz regressziója $\hat{\eta}$ -nek $-\hat{\xi}_{i,t}$ -re, és $-\hat{\xi}_{v+r,t}$ -re, ahol $i = 1, \dots, j-1$, $r = 1, \dots, \rho_y$. $\Delta \hat{\mu}_s$ pedig Toeplitz regressziója $\hat{\eta}$ -nek $-\hat{\xi}_{t-s}$ -re és $\hat{\phi}_{t-s}$ -re nézve, ha $s = 1, \dots, n$.

iii. Most számítsuk ki az információs kritérium függvényünket:

$$IC_T(n) = T \log \hat{\sigma}_\eta^2(n) + p_T(d_j(n))$$

Itt a $p_T(d_j(n))$ a Penalty Term függvényünk, ami pozitív valós értékű, monoton növekvő $d_j(n)$ -ben és nem csökkenő T -ben.

(b) **IF** $IC_T(n) > IC_T(n-1)$ - ha a kritérium függvény megtalálta a lokális minimumát

Akkor legyen:

- $r(j) = i(k) - r(j)$ indexet $i(k)$ -ra állítjuk
- $n_{r(j)} = n - 1$ - a Kronecker indexünk az előző lesz
- $P = E_{i(k),j}P$ - a P permutáció mátrixunkat változtatjuk
- $\mathcal{N} = \mathcal{N} \setminus i(k)$ - az indexhalmazunkból kivesszük $i(k)$ -t
- $j = j - 1$ - a koordinátáink szerint eggyel feljebb lépünk.
- **end**

(c) **end**

iii **FOR** $i(k) \in \mathcal{N}$

- **IF** $n < h_t$ (előre megadott Kronecker felső korlát)

$n = n + 1$

- **else**

– **FOR** $i(k) \in \mathcal{N}, k = 1, \dots, v$

– $r(j) = i(k) - r(j)$ -t $i(k)$ -ra állítjuk.

– $n_{r(j)} = n$ - az $r(j)$ -edik koordináta Kronecker invariánsát beállítjuk n -re.

– $P = E_{i(k),j}P$ - a P permutáció mátrixunkat változtatjuk

– $\mathcal{N} \setminus i(k)$ - kivesszük $i(k)$ koordinátát az \mathcal{N} halmazunkból

– $j = j - 1$ - eggyel fentebbi koordinátát vizsgálunk a továbbiakban

– **end FOR** $i(k) \in \mathcal{N}$

- **end**

iv **end** when $j = 0$

2.5.3. Algoritmussal kapcsolatos tételek

Az algoritmusunk működéséhez szükséges Poskitt [4]-ből pár tételt idéznem, ami biztosítja az algoritmus működését. Ezeket nem fogom bizonyítani, ezek bizonyításai a fentebb idézett cikkben megtalálhatóak.

2.5.2. Tétel. *Legyen y_t ECARMA(ν, ρ) folyamat, ami teljesíti a Feltétel1'-t, és Feltétel2'-t. Legyen ρ_T és $(r(j)_T, n_{r(j)_T})$ a kointegrációs rang, és a Kronecker invariáns párok, amiket az algoritmusunk segítségével kapunk. Ekkor elég nagy T -re $\rho_T = \rho^0$ egy valószínűséggel, és ha $\frac{p_T(d_q(n))}{T} \rightarrow 0$, és $\log \log \frac{T}{p_T(d_q(n))} \rightarrow 0$, ha $T \rightarrow \infty$, és ekkor $r(j)_T = r(j)^0$ 1 valószínűséggel elég nagy T -re, azaz $P(\lim_{T \rightarrow \infty} n_{r(j)_T} = n_{r(j)^0}) = 1$, minden $j = 1, \dots, v$ -re.*

2.5.3. Megjegyzés. *Ebben a tételben ρ^0 a valódi kointegrációs rang, míg $n_{r(j)^0}$ a valódi Kronecker invariáns.*

3. fejezet

Program

A szakdolgozathoz készített program R programozási nyelvben készült. Célja az előbb ismertetett algoritmus alkalmazása stacionárius esetben.

A programomban Paul D. Gilbert 10 dimenziós idősorát használom fel a *dse* program-csomagból. Célunk ezekből a kivett 2, 3, és 4 dimenziós idősorok Kronecker indexeinek a meghatározása Poskitt algoritmusá szerint. A megfigyelések 1974-től 1993-ig tartanak, és 236 megfigyelésünk van minden változóra nézve. A változóink a csomag szerint a következők:

- CPI , GDP , $M1$,
- long run interest rates (RL),
- the Toronto stock exchange 300 index ($TSE300$),
- employment
- the Canada/US exchange rate (PFX),
- a commodity price index in US dollars,
- US industrial production, and
- US CPI .

A program négy fő részből áll, amelyek az alábbiak:

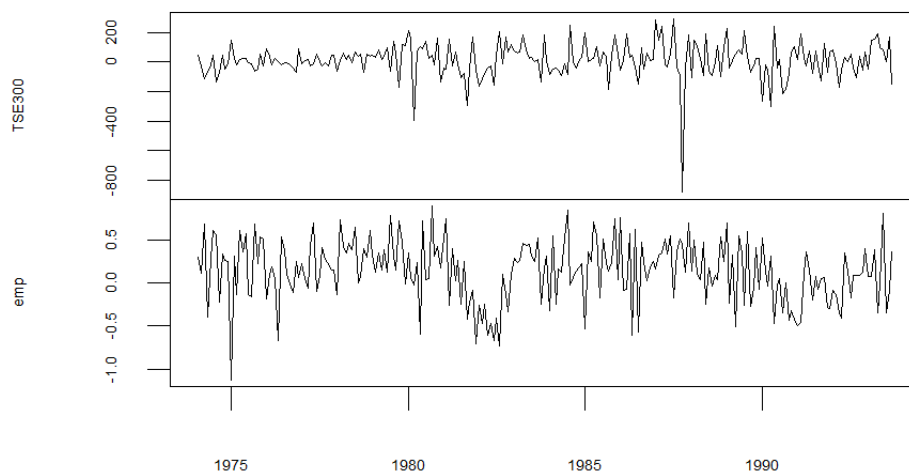
- Fő program - A főprogramban inicializálunk, és futtatjuk az alprogramokat, amik segítségével a Kronecker indexet meghatározzuk. Ebben a programban hasonlítjuk össze az alprogramoktól kapott értékeket, azaz az aktuális modellértékeket az előzőekkel.
- Null modell értékelő segédfüggvény - Ez a segédprogram kiszámítja a null modellünk értékét, amit majd össze szeretnénk hasonlítani az elsőrendű modellünkkel a főprogramunk futtatása során.
- Aktuális modell értékelő segédfüggvény - Ez a segédprogram kiszámítja az aktuális modell értékünket, amit össze szeretnénk hasonlítani az előző modellértékünkkel a főprogramunk futtatása során.
- Meghívó függvényből - Ebben a programban töltjük be az előbb ismertetett klasszikus adathalmazt, és futtatjuk a tesztünkben az algoritmusunkat különböző adathalmazunkból kivett koordinátákra.

3.1. Tesztelés

A következő inputokra teszteltem a programot:

- Elsőnek az ötödik és hatodik koordinátára,
- Másodiknak az első, a harmadik, és a tizedik koordinátára,
- Harmadiknak a harmadik, negyedik, ötödik, és hatodik koordinátára.

Az **első** teszt idősora a következő:



Ezekre a program a következő eredményeket adta:

ap

```
[1] 2 1
```

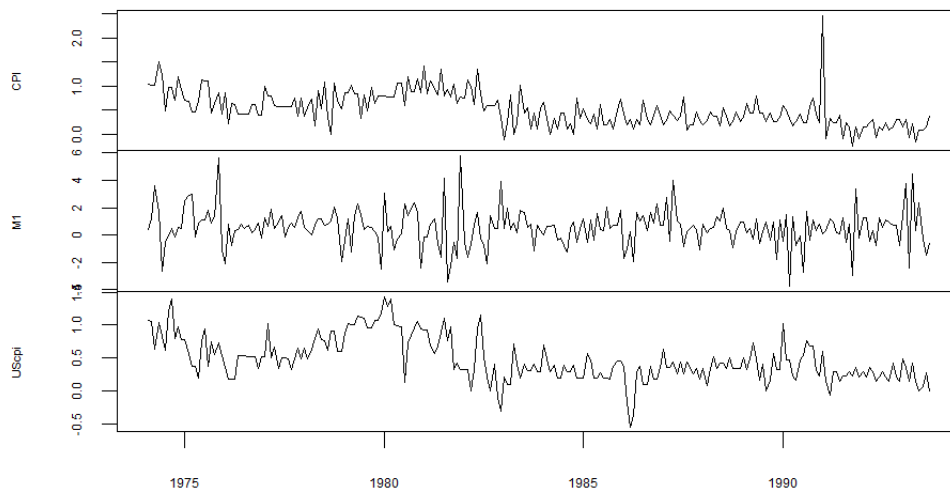
kr

```
[1] 0 1
```

ert

```
[1] 2922.733 2913.375
```

A **második** teszt idősor a következő:



Ezekre a program a következő eredményeket adta:

ap

```
[1] 3 1 2
```

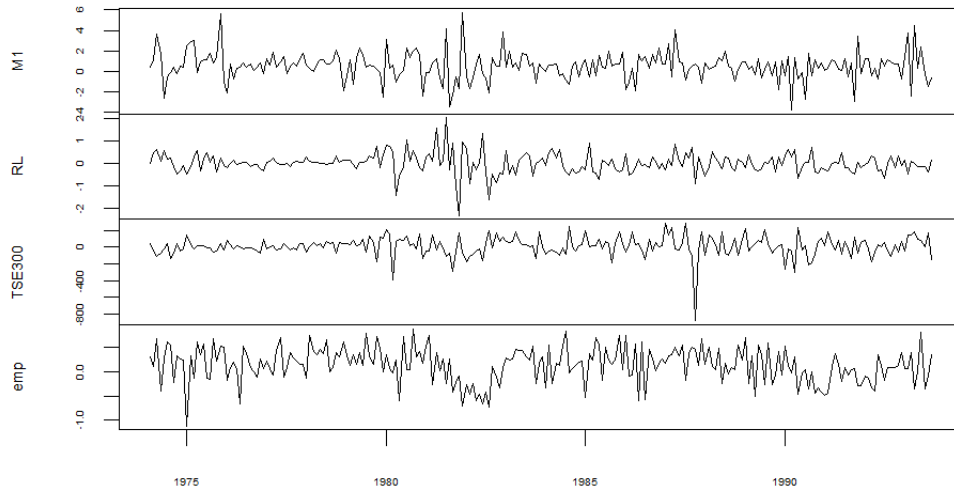
kr

```
[1] 2 2 3
```

ert

```
[1] 853.9423 845.1102 850.1850
```

A **harmadik** teszt idősor a következő:



Ezekre a program a következő eredményeket adta:

```
ap
```

```
[1] 4 1 2 3
```

```
kr
```

```
[1] 1 1 1 2
```

```
ert
```

```
[1] 2911.480 2911.032 2911.731 2915.556
```

3.2. A programok forráskódjai

Ebbe a fejezet a programjaim, és az alprogramjaim Project-R belí forráskódjait tartalmazza. A kódok elején kommentben külön feltüntettem, hogy mik a programok, vagy alprogramok inputjai, illetve outputjai. Emellett egyes lépéseit kommentekben részletezem a pontosabb megérthetőség kedvéért. A program a *dse* programcsomagot felhasználja, aminek a letöltése, és futtatása az első lényeges lépés a programunk kipróbálásához. Ezután az alprogramjainkat (ha a kódot nem szeretnénk átírni a futtatásához), akkor tároljuk el az aktuális könyvtárunk fölötti mappában, és a programkódban található néven mentjük el őket *.t* formátumban. A főprogramot *main.t* néven, a null értékelő programot *nullmod.t* néven, míg az aktuális modell értékelő függvényt *aktmod.t* néven.

3.2.1. Főprogram

```

# FŐ PROGRAM
# -----
# bemenet: y
# kimenet: ap, kr, ert
# hívott saját függvények: nullModellErt, aktModellErt
# további függvények: dse csomagból
# -----

PKron<-function(y,nm=6)
{ k<-nseriesOutput(y) # a dimenzió
  ap<-1:k # az aktuális permutáció
  akh<-1:k # az akt.koord.halmaz
  akh.k<-length(akh) # az akt.koord.halmaz számossága
  ak<-0 # az akh-ból legutóbb értékelt koordináta ap-beli indexe (még nem volt)
  kr<-rep(0,k)

  ert <- nullModellErt(y)# a null modellek értékelése
  n<-1 # az aktuális rend

  while((n<nm)&!(akh.k==0)) # míg az akh nem üres
  {
    if(ak==1) {n<-n+1;ak<-0}
    ak<-if(ak==0) akh.k else ak-1
    t.ak<-ap[ak] # az akt koordináta tényleges sorszama
    t.ek<-if(akh.k>1) ap[(1:akh.k)[-ak]] else NULL # az előző koordináták
    a.ert<-aktModellErt(t.ak,t.ek,n,y) # az aktuális modell értékelése
    if(a.ert > ert[ap[ak]])
      {ert[ap[ak]]<- a.ert} # az akt modell jobb
    else # van Kronecker
      {kr[ap[ak]]<-n-1;
ap[c(ak,akh.k)]<-c(ap[akh.k],ap[ak])# ap[ak] <-> ap[akh.k] csere
      akh.k<-akh.k-1 }
  }
  return(list(ap=ap,kr=kr,ert=ert)) }

# -----

```


3.2.2. Null modell értékelő függvény

```
# a null modellek értékelése: nullModellErt()
# ---
# bemenet: a teljes folyamat
# kimenet: ert = a k.db koordináta 0 modell értékelése
# -----

nullModellErt <- function(y)
{ k<-nseriesOutput(y)
  AR<-array(0,dim=c(1,1,1))
  AR[1,,]<-diag(1)
  MA<-array(0,dim=c(1,1,1))
  MA[1,,]<-diag(1)
  EX<-array(0,dim=c(1,1,k-1))
  EX[1,,]<-runif(k-1,-1,1)

  arma <- ARMA(A=AR, B=MA, C=EX);
  armaSS<-toARMA(arma)
  s0<-rep(NA,k)
  for(i in 1:k)
  { W<-y
    inputData(W)<-outputData(W,series=(1:k)[-i])
    outputData(W)<-outputData(W,series=i)
    M<-estMaxLik(armaSS,W)
    s0[i]<-informationTestsCalculations(M)[3]
  }
  return(s0)
}

# nullModellErt(D) # a függvény hívása
# -----
```

3.2.3. Aktuális modell értékelő függvény

```

# -----
# az aktuális modell értékelése: aktModellErt(ak,ek)
# ---
# bemenet: ak az aktuális koordináta
#          ek az előző koordináták
#          n az aktuális rend
#          y a teljes folyamat
# kimenet: ert a koordináta n-ed rendű modell értékelés
# -----

aktModellErt<-function(ak,ek,n,y)# az aktualis koord ertekelese
{ k<-nseriesOutput(y)
  # n<-1; ak<-2 ; ek<-c(1,3)
  AR<-array(0,dim=c(n+1,1,1)) # arben levők szabad változók, strukturális nullák
# vagy szabad paraméterek.
  AR[1,,]<-diag(1)
  for(i in 2:(n+1)) AR[i,,]<-runif(1,-1,1)
  MA<-array(0,dim=c(n+1,1,1))
  MA[1,,]<-diag(1)
  for(i in 2:(n+1)) MA[i,,]<-runif(1,-1,1)
  EX<-array(0,dim=c(n+1,1,k-1))
  d<-rep(0,k); if(!is.null(ek)) d[ek]<-1; d<-d[-ak]
  EX[1,,]<-d
  for(i in 2:(n+1)) EX[i,,]<-runif(k-1,-1,1)
  arma <- ARMA(A=AR, B=MA, C=EX);
  armaSS<-toARMA(arma)
  W<-y
  inputData(W)<-outputData(W,series=(1:k)[-ak])
  outputData(W)<-outputData(W,series=ak)
  M<-estMaxLik(armaSS,W)
  s1<-informationTestsCalculations(M)[3]
  return(s1)}

# -----

```

3.2.4. Meghívó függvény

```

# -----
# vesszük a dse::egJofF.1dec93.data adathalmazt

library(dse)#
rm(list=ls())
source('./main.t')# PKron() -- a főprogram
source('./nullmod.t')# nullModellErt(y) -- a null értékelés
source('./aktmod.t')# aktModellErt(ak,ek,n,y) -- az aktuális ARMAX értékelése
data(egJofF.1dec93.data);D<-egJofF.1dec93.data;rm(egJofF.1dec93.data)
D$input<-NULL
attributes(D$output)$dimnames<-list(NULL,attributes(D$output)$seriesNames)
attributes(D$output)$dimnames[[2]][c(6,8,9,10)]<-c("emp","ComPri","USip","USspi")
plot(D$output)

# -----
# első teszt
W<- D
outputData(W) <- outputData(W,series=c(5,6))
plot(W$output)
PKron(W)

# -----
# második teszt
W<- D
outputData(W) <- outputData(W,series=c(1,3,10))
plot(W$output)
PKron(W)

# -----
# harmadik teszt
W<- D
outputData(W) <- outputData(W,series=c(3,4,5,6))
plot(W$output)
PKron(W)

# -----

```

Irodalomjegyzék

- [1] D.S. Poskitt, *On the Identification and Estimation of Nonstationary and Cointegrated ARMAX Systems*
- [2] D.S. Poskitt, Helmut Lütkepohl *Specification of echelon form VARMA models*
- [3] D.S. Poskitt, *Strongly Consistent Determination of Cointegrating Rank via Canonical Correlations*
- [4] D.S. Poskitt, *Vector Autoregressive Moving Average Identification for Macroeconomic Modeling: A New Methodology*
- [5] E.J. Hannan, J.Rissanen *Recursive estimation of mixed autoregressive-moving average order*
- [6] Helmut Lütkepohl, *New Introduction to Multiple Time Series Analysis*
- [7] Pröhle Tamás, *VARMA identifikáció Echelon formában jegyzet*
- [8] Robert F. Engle, C. W. J. Granger *Co-Integration and Error Correction: Representation, Estimation, and Testing*
- [9] Ruel S. Tsay, *Parsimonious Parameterization of Vector Autoregressive Moving Average Models*
- [10] Said Nsiri, Roch Roy, *On the Identification of ARMA Echelon-Form Models*