SZEREGI CZASOWE

Przekształcenia szeregu czasowego
Zagadnienie stacjonarności
Modele ARMA

Marzena Nowakowska

Wydział Zarządzania i Modelowania Komputerowego
Politechnika Świętokrzyska

Budynek C, p. 3.21

spimn@tu.kielce.pl

Przekształcenia wstępne

Cel

Mają więc na celu taką transformację szeregu wejściowego, aby wynik był szeregiem stacjonarnym.

Do podstawowych przekształceń zalicza się:

- transformacje potęgowe, w szczególności Boxa-Coxa, w celu eliminowania niejednorodności wariancji,
- eliminację trendu i sezonowości poprzez różnicowanie, jednokrotne lub wielokrotne z opóźnieniem 1 lub opóźnieniem sezonowym,
- eliminację trendu i sezonowości poprzez dekompozycję w celu uzyskania składnika losowego (*to już było*)

Transformacja Boxa-Coxa

$$f_{\lambda(x)} = \begin{cases} \dfrac{x^{\lambda} - 1}{\lambda} & \text{dla } \lambda \neq 0 \\ ln(x) & \text{dla } \lambda = 0 \end{cases}$$

Transformację można stosować tylko w przypadku szeregu, którego wszystkie wartości są większe od zera.

Przesłanki stosowania transformacji:

- występowanie wzrostu lub spadku amplitudy wahań sezonowych (rosnąca lub malejąca amplituda wraz z poziomem szeregu)
- niejednorodna zmienność danych w kolejnych wartościach szeregu,
- duży wpływ obserwacji odstających
- dążenie do uzyskania rozkładów zbliżonych do normalnego, aby działały metody wymagające normalności

Jeżeli zastosuje się transformację Boxa-Coxa, to prognozy i odpowiadające im przedziały predykcyjne są tworzone dla danych przekształconych. Aby uzyskać dla danych pierwotnych, należy zastosować transformację odwrotną:

$$\tilde{f}_{\lambda(x)} = \begin{cases} (x \cdot \lambda + 1)^{1/\lambda} & \text{dla } \lambda \neq 0 \\ exp(x) & \text{dla } \lambda = 0 \end{cases}$$

Różnicowanie

<u>Różnicowanie z opóźnieniem 1</u> szeregu X_t definiuje się jako różnicę:

$$\nabla X_t = X_t - X_{t-1} = (1-B)X_t$$

gdzie B jest operatorem przesunięcia wstecz zdefiniowanym jako: $BX_t = X_{t-1}$

<u>Różnicowanie d-krotne</u> szeregu X_t z opóźnieniem 1 definiuje się jako różnicę (różnicowanie rzędu d):

$$\nabla^d X_t = \nabla\left(\nabla^{d-1} X_t\right) = \nabla^{d-1} X_t - \nabla^{d-1} X_{t-1} \quad \text{dla } d \geq 1 \quad \text{oraz} \quad \nabla^0 X_t = X_t$$

Operator d-krotnego różnicowania z opóźnieniem 1 zapisuje jako:

$$\nabla^d = (1-B)^d$$

Jeżeli po d-krotnym różnicowaniu z opóźnieniem 1 szeregu niestacjonarnego otrzymuje się szereg stacjonarny, to szereg pierwotny jest <u>szeregiem zintegrowanym w stopniu *d*</u>, co zapisuje się jako: $X_t \sim I(d)$, przy czym d jest najmniejszą liczbą całkowitą, dla której jest stacjonarnie.

<u>Różnicowanie z opóźnieniem (sezonowym) *s*</u> szeregu X_t definiuje się jako różnicę:

$$\nabla_s X_t = X_t - X_{t-s} = (1-B^s)X_t$$

gdzie $B^s X_t = B(B^{s-1}X_t) = X_{t-s}$ jest operatorem różnicowania z opóźnieniem s (operator różnicowania sezonowego).

Własności różnicowania

- Różnicowanie z opóźnieniem 1 usuwa z szeregu trend liniowy, jeżeli taki występuje.
- Wykonanie *d*-krotnego różnicowania z opóźnieniem 1 powoduje, że z danych zostaje wyeliminowany wielomian stopnia *d*, jeżeli taki występuje.
- Różnicowanie z opóźnieniem *s* usuwa z szeregu sezonowość o okresie *s*, jeżeli taki występował. Dodatkowo usunięty jest też liniowy, jeżeli taki istniał.
- Niekiedy należy zastosować więcej niż jedną operację różnicowania aby uzyskać postać stacjonarną szeregu.
- Operację różnicowania z opóźnieniem *s* można stosować kilkakrotnie (*d*-krotnie). Można tez stosować kolejno operacje różnicowania dla różnych opóźnień.
- Operatory różnicowania są przemienne. To oznacza, że można najpierw wykonać operację różnicowania sezonowego a następnie różnicować z opóźnieniem 1 lub odwrotnie.

Funkcja BoxCox(x, lambda) # forecast

Zwraca transformację zmiennej wejściowej przy użyciu transformacji Boxa-Coxa. *InvBoxCox*() odwraca transformację. Wartości ujemne funkcja przekształca zgodnie z zależnością:

$$f_{\lambda(x)} = \begin{cases} \dfrac{sign(x)|x|^{\lambda} - 1}{\lambda} & \text{dla } \lambda \neq 0 \\ ln(x) & \text{dla } \lambda = 0 \end{cases}$$

x	Wektor numeryczny lub szereg czasowy klasy *ts*.
lambda	Parametr transformacji.

Funkcja BoxCox.lambda(x, method = c("guerrero", "loglik"), lower = -1, upper = 2) zwraca liczbę wskazującą parametr transformacji Boxa-Coxa.

Funkcja diff(x, ...)

Zwraca zróżnicowany szereg wielokrotnie z podanym opóźnieniem.

x	Wektor numeryczny lub szereg czasowy klasy *ts*.
lag	Wartość całkowita określająca wartość opóźnienia, domyślnie: 1.
differences	Krotność (rząd) różnicowania; domyślnie: 1.

Transformacje w R - przykład

```
# Szereg pbk to produkt krajowy brutto Polski

# Mozna rĂłwniez wczytac plik tekstowy z menu Import Dataset
# pod tytulem zakladki Environment
# uwaga przy czytanie za pomoca instrukcji read.csv trzeba daĂ‡ znaĂ‡,
# czy pierwszy wiersz zawiera nazwy kolumn (domyslnie jest przyjmowane, ĹĽe zawiera)

pkb <- ts(read.csv("c:/R/SzeregiCzasowe/Dane/pkb.txt", header=FALSE), start=1995, frequency=4)

# Transformacja Boxa-Coxa
pkb.sqrt<- BoxCox(pkb, lambda = 0.5)
pkb.log<- BoxCox(pkb, lambda = 0)
par(mfrow=c(3,1))
plot(pkb, main="Dane oryginalne")
plot(pkb.sqrt, main="Box-Cox, lambda=0.5")
plot(pkb.log, main="Box-Cox, lambda=0")
library(forecast)
lambda <- BoxCox.lambda(pkb, mthod = "loglike")      #  wyznaczenie wartoĹ›ci lambda, por. opis funkcji
pkb.est<- BoxCox(pkb, lambda = BoxCox.lambda(pkb, method = "loglik"))
plot(pkb.log, main="Box-Cox, lambda=est")

# RĂłĹĽnicowanie z opĂłĹşnieniem prostym (domyslan wartoĹ›Ă‡ d=1)
library(expsmooth)  # dane usgdp
library(forecast)
data(usgdp)
tsdisplay(usgdp)
usgdp.diff <- diff(usgdp)
tsdisplay(usgdp.diff)

# RĂłĹĽnicowanie z opĂłĹĽnieniem sezonowym, d=12 (s=12)
library(forecast)
AirPass.diff12 <- diff(AirPassengers, lag=12, differences=1)
tsdisplay(AirPassengers)
tsdisplay(AirPass.diff12)
par(mfcol=c(1,2))
plot(AirPassengers); plot(AirPass.diff12)
AirPass.diff12.diff_x <- diff(AirPass.diff12, lag=1, differences=3)  # demo rĂłĹĽnic dla rzedĂłw: 1-3
tsdisplay(AirPass.diff12.diff_x)
```

Stacjonarność szeregu

W szeregu niestacjonarnym mogą wystąpić zjawiska (łącznie lub pojedynczo):

- średnia zamiast utrzymywać się na stałym poziomie, rośnie wraz z czasem,
- z roku na rok wzrasta różnica między rocznymi minimami i maksimami
- proces wykazuje silną sezonowość.

Stosowanie testów sprawdzających, czy dany proces jest stacjonarny nazywa się weryfikacją hipotezy stacjonarności szeregu.

Test pierwiastka jednorodnego (*unit root test*):
Jeżeli pierwiastkiem równania (wielomianu) charakterystycznego opisującego proces stochastyczny, zapisanego w postaci autoregresyjnej, jest jedynka, to proces nie jest to stacjonarny (i zintegrowany w stopniu pierwszym). W przypadku, gdy pierwiastki są do modułu różne od 1, to proces jest stacjonarny.

<u>Hipoteza zerowa</u> zakłada obecność pierwiastka jednostkowego - niestacjonarność procesu:

H0: Szereg czasowy jest niestacjonarny

Najpopularniejszym narzędziem do badania stacjonarności szeregu (na podstawie testu pierwiastka jednorodnego) jest test (raczej rzadko) DF lub **rozszerzony test Dickeya-Fullera (ADF) (najczęściej).**

Uwagi nt. testu ADF

Istnieją trzy typy szeregów czasowych, według których można obliczyć statystyki testowe testu Dickey-Fullera.

1) zerowa średnia — brak przecięcia. Szereg czasowy jest błądzeniem losowym bez trendu i bez dryfu
2) pojedyncza średnia — obejmuje wyraz wolny. Szereg czasowy jest błądzeniem losowy bez trendu i z dryfem
3) punkt przecięcia i trend. Szereg czasowy jest błądzeniem losowy z trendem i z dryfem

Testy stacjonarności nie rozwiązują definitywnie zagadnienia stacjonarności:

- występują problemy z odróżnieniem pierwiastka jednostkowego od pierwiastków niewiele różniących się od jedynki,
- w przypadku małej próby, wnioski mogą być fałszywe,
- większość testów nie sprawdza wszystkich czynników mogących mieć wpływ na niestacjonarność.

Test stacjonarności DF/ADF szeregu czasowego sprowadza się do sprawdzenia jego stopnia zintegrowania.

Funkcja adf.test(x, ...) biblioteka aTSA

Funkcja dla zwraca wyniki testu ADF trzech typów szeregów (jak podano wcześniej).

x	Wektor numeryczny lub szereg czasowy klasy ts.
nlag	Wielkość maksymalnego opóźnienia, do którego jest wykonane testowanie.
output	Flaga informująca, czy wyniki mają być wyprowadzone do do konsoli.

Przykład

```
library(aTSA)    # test Dickeya-Fullera
library(forecast)  # funkcje autokorelacji

x <- as.ts(rnorm(200))  # biaĹ‚y szum, stacjonarny; H0 odrzucone
par(mfrow=c(3,1))
plot(x)
Acf(x, lag.max=30)
Pacf(x, lag.max=30)
adf.test(x, nlag = 4)

y <- as.ts(cumsum(x))   # bĹ‚adzenie losowe, niestacjonarny; H0 nieodrzucone
par(mfrow=c(3,1))
plot(y)
Acf(y, lag.max=30)
Pacf(y, lag.max=30)
adf.test(y, nlag = 4)
```

Model AR(p)

Model autoregresyjny rzędu p jest modelem szeregu czasowego X_t danym zależnością:

$$X_t = \phi_0 + \phi_1 \cdot X_{t-1} + \phi_2 \cdot X_{t-2} + \ldots + \phi_p \cdot X_{t-p} + Z_t$$

gdzie:

ϕ_0 – stała, dla uproszczenia często pomijana,

$\phi_1, \ldots, \phi_p$ – parametry modelu,

Z_t – szereg typu biały szum WN(0, σ^2).

Model AR(p) jest zapisywany z wykorzystaniem operatora autoregresji $\phi(B)$ rzędu p jako:

$$\phi(B)X_t = \phi_0 + Z_t$$

gdzie:

$\phi(B) = 1 - \phi_1 B - \phi_2 B^2 - \ldots - \phi_p B^p$ jest operatorem autoregresji rzędu p,

B jest operatorem przesunięcia wstecz.

Model autoregresji jest słabostacjonarny, jeżeli wszystkie pierwiastki równania charakterystycznego (wielomianu autoregresyjnego stopnia p):

$$1 - \phi_1 \cdot y^1 - \phi_2 \cdot y^2 - \ldots - \phi_p \cdot y^p = 0$$

leżą poza okręgiem jednostkowym, tzn. spełniają nierówność $|y| > 1$. To oznacza, że jeżeli parametry ww. równania zapewniają wspomniane rozwiązanie, to proces AR(p) o tych parametrach jest stacjonarny.

Model jest wykorzystywany do identyfikacji procesów stacjonarnych

Identyfikacja modelu AR(p)

Reguła identyfikacji modelu autoregresji AR(p) jest oparta na postaci teoretycznej funkcji autokorelacji cząstkowej $\alpha(k)$ dla przesunięcia (opóźnienia) k. Jeżeli szereg $\{X_t\}$ jest procesem AR(p), to funkcja autokorelacji cząstkowej tego szeregu znika dla opóźnień większych niż rząd p procesu, tzn.

$$\alpha(k) \neq 0 \ \text{jeżeli } k \leq p$$
$$\alpha(k) = 0 \ \text{jeżeli } k > p$$

W praktyce stosuje się ww. właściwość do cząstkowej autokorelacji próbkowej.
Jeżeli cząstkowa autokorelacja próbkowa PACF(k) znajduje się pomiędzy przedziałami ufności (- 1,95/√n, 1,95/√n) dla $k > p$, to można oczekiwać, że dane realizują proces AR(p).

Identyfikacja modelu AR(p) - przykład

```
library(forecast)
plot(AirPassengers)

# Transformacja Boxa-Coxa; poszukiwanie stałej lambda
lamb <- BoxCox.lambda(AirPassengers, method = "loglik")
par(mfrow=c(2,1))
for (lamb in seq(from=-1,to=0.9,by=0.1))
{
  AirPass.B_C<- BoxCox(AirPassengers, lambda = lamb)
  plot(AirPass.B_C, main=paste("Box-Cox, lambda=", as.character(lamb)))
  Pacf(AirPass.B_C, lag.max=52)
}

# grid(nx=12)
par(mfrow=c(2,1))
# Transformacja: różnicowanie sezonowe z opóźnieniem 12 i proste rzedu: od 1 do 5
AirPass.diff12 <- diff(AirPassengers, lag=12, differences=1)
plot(AirPass.diff12)
for (d in seq(from=1,to=5,by=1))
{
  AirPass.diff12.diff_x <- diff(AirPass.diff12, lag=1, differences=d)
  plot(AirPass.diff12.diff_x, main=paste("Roznicowanie; 1 x 12 sezowowe, proste ",
                                          paste(as.character(d), "x 1")))
  Pacf(AirPass.diff12.diff_x, lag.max=52)
}

# to samo mozna przeprowadzić dla szeregu usgdp
```

Model MA(q)

Model średniej ruchomej rzędu q jest modelem szeregu czasowego X_t danym zależnością:

$$X_t = \mu + Z_t + \theta_1 \cdot Z_{t-1} + \theta_2 \cdot Z_{t-2} + \ldots + \theta_q \cdot Z_{t-q}$$

gdzie:

μ – wyraz wolny jest wartością oczekiwaną szeregu czasowego opisanego modelem MA(q), często zakładana jako równa zero i dlatego pomijana,

$\theta_1, \ldots, \theta_q$ – parametry modelu,

Z_t – szereg typu biały szum WN(0, σ^2).

Model MA(q) jest zapisywany z wykorzystaniem operatora średniej ruchomej $\theta(B)$ rzędu q jako:

$$X_t = \mu + \theta(B) Z_t$$

gdzie:

$\theta(B) = 1 + \theta_1 B + \theta_2 B^2 + \ldots + \theta_p B^p$ jest operatorem średniej ruchomej rzędu q,

B jest operatorem przesunięcia wstecz.

Założeniem modelu MA(p) było uznanie, że wartość bieżąca szeregu zależy od wartości białego szumu lub nieprzewidywalnych szoków, co można przedstawić w postaci liniowej. Model MA obrazuje procesy będące skończoną sumą ważoną białego szumu z bieżącego oraz q poprzednich okresów

<u>Każdy</u> proces średniej ruchomej <u>MA jest stacjonarny</u>.

Model jest wykorzystywany do identyfikacji procesów stacjonarnych

Identyfikacja modelu MA(q)

Reguła identyfikacji modelu autoregresji MA(q) jest oparta na postaci teoretycznej funkcji autokorelacji $\rho(k)$ dla przesunięcia (opóźnienia) k. Jeżeli szereg $\{X_t\}$ jest procesem MA(q), to funkcja autokorelacji tego szeregu znika dla opóźnień większych niż rząd q procesu, tzn.

$$\rho(k) \neq 0 \;\text{ jeżeli } \mathrm{k} \leq \mathrm{q}$$

$$\rho(k) = 0 \;\text{ jeżeli } \mathrm{k} > q$$

Każdy model stacjonarny o tej własności ma reprezentację w postaci odpowiedniego modelu MA(q).
W praktyce stosuje się ww. właściwość do autokorelacji próbkowej sprawdzając, czy dla opóźnień k większych niż q można uznać ją za statystycznie nieistotną.
Jeżeli autokorelacja próbkowa ACF(k) znajduje się pomiędzy przedziałami ufności (- 1,95/√n, 1,95/√n) dla k > q, to można oczekiwać, że dane realizują proces MA(q).

Identyfikacja modelu MA(q) - przykład

```
#
library(forecast)
plot(AirPassengers)

# Transformacja Boxa-Coxa; poszukiwanie stałej lambda
lamb <- BoxCox.lambda(AirPassengers, method = "loglik")
par(mfrow=c(2,1))
for (lamb in seq(from=-5,to=5,by=0.2))
{
  AirPass.B_C<- BoxCox(AirPassengers, lambda = lamb)
  plot(AirPass.B_C, main=paste("Box-Cox, lambda=", as.character(lamb)))
  Acf(AirPass.B_C, lag.max=52)
}

# grid(nx=12)

# Transformacja: różnicowanie sezonowe z opóźnieniem 12 i proste rzedu: od 1 do 5
AirPass.diff12 <- diff(AirPassengers, lag=12, differences=1)
plot(AirPass.diff12)
par(mfrow=c(2,1))
for (d in seq(from=1,to=5,by=1))
{
  AirPass.diff12.diff_x <- diff(AirPass.diff12, lag=1, differences=d)
  plot(AirPass.diff12.diff_x, main=paste("Różnicowanie; 1 x 12 sezonowe, proste ",
                                         paste(as.character(d), "x 1")))
  Acf(AirPass.diff12.diff_x, lag.max=52)
}
```

Model ARMA(p, q)

Modele autoregresji i średniej ruchomej (ARMA) dostarczają oszczędny opis (słabo) stacjonarnego procesu stochastycznego za pomocą dwóch wielomianów, jednego dla autoregresji (AR), a drugiego dla średniej ruchomej (MAMA).

Model ARMA(p, q) jest modelem szeregu czasowego X_t danym zależnością:

$$X_t = \phi_1 \cdot X_{t-1} + \phi_2 \cdot X_{t-2} + \ldots + \phi_p \cdot X_{t-p} + Z_t + \theta_1 \cdot Z_{t-1} + \theta_2 \cdot Z_{t-2} + \ldots + \theta_q \cdot Z_{t-q}$$

lub w postaci skróconej:

$$X_t = \sum_{i=1}^{p} \phi_i X_{t-1} + Z_t + \sum_{i=1}^{q} \theta_i Z_{t-i}$$

gdzie

Z_t – szereg typu biały szum WN(0, σ^2).

$\phi_1, \ldots, \phi_p$ – parametry części modelu przypisane do AR,

$\theta_1, \ldots, \theta_q$ – parametry części modelu przypisane do MA.

Aby zapewnić jednoznaczność reprezentacji w ww. zależności zakłada się, że wielomiany odpowiednio stopnia p i q:

$$\phi(y) = 1 - \phi_1 \cdot y^1 - \phi_2 \cdot y^2 - \ldots - \phi_p \cdot y^p$$

$$\theta(y) = 1 + \theta_1 y^1 + \theta_2 y^2 + \ldots + \theta_q y^q$$

nie mają wspólnych czynników, tzn. nie mogą być zredukowane do wielomianów mniejszych stopni.

Model jest wykorzystywany do identyfikacji procesów stacjonarnych

Model ARMA(p, q) - cd

Model ARMA(p, q) jest zapisywany z wykorzystaniem operatora autoregresji $\phi(B)$ rzędu p i operatora średniej ruchomej rzędu q $\theta(B)$ jako:

$$\phi(B)X_t = \theta(B)Z_t$$ gdzie B jest operatorem przesunięcia wstecz.

Przykład postaci modelu ARMA(1, 1): $X_t = \phi_1 X_{t-1} + Z_t + \theta_1 Z_{t-1}$

Modele AR(p) i MA(Q) są szczególnymi przypadkami modelu ARMA(p, q):

- AR(p) = ARMA(p, 0)
- MA(q) = ARMA(0, q)

Aby model ARMA(p, q) był <u>stacjonarny</u> wystarczy, aby jego część odpowiadająca modelowi AR(p) była stacjonarna, ponieważ modele MA(q) są zawsze stacjonarne. To oznacza, że model <u>jest słabostacjonarny, jeżeli</u> wszystkie pierwiastki równania charakterystycznego definiowanego dla modelu AR(p) (wielomianu autoregresyjnego stopnia p):

$$1 - \phi_1 \cdot y^1 - \phi_2 \cdot y^2 - \dots - \phi_p \cdot y^p = 0$$

leżą poza okręgiem jednostkowym, tzn. spełniają nierówność $|y| > 1$.

Własności modelu ARMA(p, q)

Chwilowo pominięte

Odwracalność (*inversability*) oznacza możliwość przedstawienia wartości zakłócenia szeregu w chwili *t* jako kombinacji liniowej bieżącej i wcześniejszych wartości szeregu. ARMA(p, q) jest procesem odwracalnym, gdy istnieje ciąg stałych $\{\pi_j\}$, takich że zachodzi możliwość przedstawienia modelu ARMA(p, q) w postaci szeregu AR(∞):

$$Z_t = \sum_{i=0}^{\infty} \pi_i Z_{t-i} \quad \text{oraz} \quad \sum_{i=1}^{\infty} |\pi_i| < \infty$$

Odwracalność procesu ARMA jest równoważna warunkowi nałożonemu na współczynniki $\{\theta_i\}$:

$$\theta(y) = 1 + \theta_1 y^1 + \theta_2 y^2 + \ldots + \theta_q y^q \neq 0 \text{ dla wszystkich } |y| \leq 1$$

co oznacza, że odpowiadajmy ww. równaniu mu proces AR(q) jest stacjonarny.
Współczynniki θ_i są określone przez rozwinięcie szeregu potęgowego, określonego zależnością od θ i ϕ.

Przyczynowość (*casuality*) oznacza możliwość przedstawienia wartości szeregu w chwili *t* jako kombinacji liniowej bieżącego i wcześniejszych zakłóceń. ARMA(p, q) jest procesem przyczynowym, gdy istnieje ciąg stałych $\{\psi_j\}$, takich że zachodzi możliwość przedstawienia modelu ARMA(p, q) w postaci szeregu MA(∞):

$$X_t = \sum_{i=0}^{\infty} \psi_i Z_{t-i} \quad \text{oraz} \quad \sum_{i=1}^{\infty} |\psi_i| < \infty$$

Przyczynowość procesu ARMA jest równoważna (udowodnione twierdzenie) warunkowi nałożonemu na współczynniki $\{\phi_i\}$:

$$\phi(y) = 1 - \phi_1 \cdot y^1 - \phi_2 \cdot y^2 - \ldots - \phi_p \cdot y^p \neq 0 \text{ dla wszystkich } |y| \leq 1$$

Współczynniki ψ_i są określone przez rozwinięcie szeregu potęgowego, określonego zależnością od θ i ϕ.

Modele sezonowe: AR(P)$_s$, MA(Q)$_s$, ARMA(P, Q)$_s$

- Model sezonowy AR(P)$_s$

  $$\Phi(B^s)X_t = Z_t$$

- Model sezonowy MA(Q)$_s$

  $$X_t = \Theta(B^s)Z_t$$

- Model sezonowy ARMA(P, Q)$_s$

  $$\Phi(B^s)X_t = \Theta(B^s)Z_t$$

gdzie:

B^s – operator różnicowania z opóźnieniem s (operator różnicowania sezonowego),

$\Phi(B^s)$ – operator sezonowej autoregresji rzędu P: $\Phi(B^s) = 1 - \Phi_1 B^s - \Phi_2 B^{2s} - \ldots - \Phi_P B^{Ps}$

$\Theta(B^s)$ – operator sezonowej średniej ruchomej rzędu Q: $\Theta(B^s) = 1 + \Theta_1 B^s + \Theta_2 B^{2s} + \ldots + \Theta_Q B^{Qs}$

Z_t – szereg typu biały szum WN(0, σ^2).

Analogicznie do ARMA(p, q), model ARMA(P, Q) jest odwracalny gdy pierwiastki wielomianów $\Phi(y^s)$ i $\Theta(y^s)$ leżą poza kołem jednostkowym (tzn. wartości wielomianów dla argumentów leżących w obrębie koła są różne od zera.

Model mieszany: ARMA(p, q) (P, Q)$_s$

Jest połączeniem operatorów autoregresji i średniej ruchomej sezonowych i niesezonowych:

$$\Phi(B^s)\phi(B)X_t = \Theta(B^s)\theta(B)Z_t$$

Zachowanie ACF i PACF dla takich modeli jest kombinacją zachowania sezonowych i niesezonowych części modelu.

ARMA(p, q)(P, Q)$_s$ - przykłady

Przykłady (z dokładnością do stałej):

AR(1): $X_t = \phi_1 \cdot X_{t-1} + Z_t$

AR(1)$_{12}$: $X_t = \Phi_1 \cdot X_{t-12} + Z_t$

MA(1): $X_t = Z_t + \theta_1 \cdot Z_{t-1}$

MA(1)$_{12}$: $X_t = Z_t + \Theta_1 \cdot Z_{t-12}$

ARMA(1, 1): $X_t = \phi_1 X_{t-1} + Z_t + \theta_1 Z_{t-1}$

ARMA(1, 1)$_{12}$: $X_t = \Phi_1 X_{t-12} + Z_t + \Theta_1 Z_{t-12}$

Przykłady (z dokładnością do stałej):

ARMA(0,1) x (1,0)$_{12}$: $X_t = \Phi_1 \cdot X_{t-12} + Z_t + \theta_1 Z_{t-1}$

Skąd się wzięło:

ARMA(p, q) x (P, Q)$_{12}$: ARMA(0,1) x (1,0)$_{12}$ → p=0, q=1, P=1, Q=0, s=12

$\Phi(B^{12})\,\phi(B)X_t = \Theta(B^{12})\,\theta(B)Z_t$ → $\Phi(B^{12})X_t = \theta(B)Z_t$ → $(1 - \Phi_1 B^{12})X_t = (1 + \theta_1 B)Z_t$ →

→ $X_t - \Phi_1 B^{12} X_t = Z_t + \theta_1 B Z_t$ → $X_t - \Phi_1 X_{t-12} = Z_t + \theta_1 Z_{t-1}$ → $X_t = \Phi_1 X_{t-12} + Z_t + \theta_1 Z_{t-1}$

Modele ARMA(p, q)(P, Q)$_s$ są wykorzystywane do identyfikacji procesów stacjonarnych.

Cechy funkcji korelacyjnych dla modeli ARMA wykorzystywane w identyfikacji modelu ARMA

Modele zwykłe – identyfikacja rzędów zwykłych *p* i *q*			
	AR(p)	MA(q)	ARMA(p, q)
ACF	Zanika stopniowo	Znika dla k > q	Zanika stopniowo
PACF	Znika dla k > p	Zanika stopniowo	Zanika stopniowo
Modele sezonowe – identyfikacja rzędów sezonowych *P* i *Q*			
	$AR(P)_s$	$MA(Q)_s$	$ARMA(P, Q)_s$
ACF	Zanika stopniowo dla k = hs (wielokrotności okresu)	Znika dla k > Qs	Zanika stopniowo dla k = hs (wielokrotności okresu)
PACF	Znika dla k > Ps	Zanika stopniowo dla k = hs (wielokrotności okresu)	Zanika stopniowo dla k = hs (wielokrotności okresu)
h = 1, 2, 3, ...			

W przypadku modeli sezonowych analizuje się zachowanie funkcji ACF i PAFC dla opóźnień będących wielokrotnościami okresu s, tzn. $k = s, 2s, 3s$, ... Należy pamiętać, że dla modeli sezonowych wartości funkcji ACF i PACF dla opóźnień $k \neq s, 2s, 3s$, ... są równe zero.

Zaleca się, aby w pierwszej kolejności wykonać identyfikację *P* i *Q*, a dopiero później *p* i *q*.

Ogólne uwagi nt. dopasowania modeli ARMA

1.
Dla danych rzeczywistych analizuje się funkcje ACF i PACF próbkowe, co oznacza, że charakterystyczne własności modelu mogą być spełnione w przybliżeniu.
2.
Dla analizowanych danych różne modele (AR, MA, ARMA) będą równie dobrze lub równie niedobrze opisywały zjawisko. Zazwyczaj buduje się kilak modeli i z nich wybiera najlepszy stosując określone narzędzia oceny.
3.
W przypadku modeli ARMA typowe zachowania funkcji ACF i PACF to kombinacja zaników wykładniczych i sinusoid tłumionych. Nie ma prostej reguły na określenie rzędów.
4.
Reguły identyfikacji modelu ARMA można stosować, gdy analizowane dane uzna się za realizacje procesu stacjonarnego. Ponieważ w praktyce występują szeregi niestacjonarne, dopasowanie modeli stacjonarnych wymaga odpowiedniego przekształcenia danych oryginalnych, np. poprzez różnicowanie, co w ogólnym przypadku prowadzi do dopasowania modelu SARIMA(p, d, q) x $(P, D, Q)_s$.
5.
W porównaniu do modeli AR(p) i MA(q), model ARMA(p, q) zwykle wymaga dużo mniejszej liczby parametrów (czyli mniejszej liczby rzędów *p* i *q*) aby uzyskać dobre dopasowanie do danych.

Szeregi niestacjonarne – modele ARIMA i SARIMA

Model ARIMA(p, d, q) (model zintegrowany autoregresji i średniej ruchomej) jest modelem szeregu czasowego X_t, gdy szereg czasowy $(1\text{-}B)^d X_t$,gdzie parametr $d \geq 0$ określa poziom zintegrowania, jest modelem ARMA(p, q). Model ARIMA(p, d, q) dla szeregu X_t spełnia równanie różnicowe postaci:

$$\phi(B)(1-\mathrm{B})^{\mathrm{d}} X_t = \theta(B) Z_t$$

gdzie:

$\phi(B)$ – operator autoregresji rzędu p,

$\theta(B)$ – operatora średniej ruchomej rzędu q,

d – stopie} integracji szeregu X_t.

Model ARIMA jest składową trzech modeli szeregów:

- AR(p), gdzie p określa rząd autoregresji (poszczególne wartości szeregu czasowego można opisać za pomocą modelu jego na podstawie wcześniejszych obserwacji),
- I(d), gdzie d jest rzędem różnicowania (liczba krotności różnicowania danych, aby stały się stacjonarne),
- MA(q), gdzie q określa rząd średniej ruchomej (liczba opóźnień dla błędów prognozy w równaniu predykcji). Przy szacowaniu wartości następnego szeregu czasowego uwzględniane są błędy oszacowań lub prognoz z przeszłości. Różnica między oszacowaniem X_t a faktycznie zaobserwowaną wartością X_t jest oznaczona jako Z_t.

Parametry $\{\phi_i\}$ modelu są takie, że $\phi(z) \neq 0$ dla $|z| \leq 1$.

Dla $d = 0$ model ARIMA redukuje się do modelu ARMA.

Rozbudowanie modelu ARIMA o sezonowość z okresem s dostarcza model SARIMA(p,d,q)(P,D,Q)$_s$ w którym trójka (p, d, q) reprezentuje niesezonową część modelu a (P, D, Q) część sezonową.

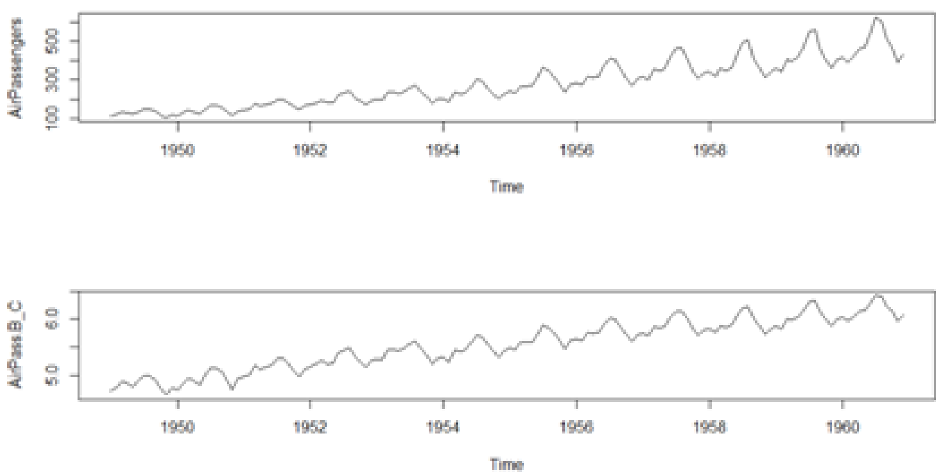
$$\phi(B)\, \Phi(B^s)\, (1-\mathrm{B})^{\mathrm{d}}\, (1-\mathrm{B})^{\mathrm{D}}\, X_t = \theta(B)\, \Theta(B^s)\, Z_t$$

gdzie operatorom rzedu prostego p i q oraz integracji stopnia d odpowidając operatory sezonowe oznaczone wielkimi literami alfabetu.

Stabilizowanie szeregu czasowego

1. Transformacja B-C ma na celu ustabilizowanie wariancji. Dane przekształcano dla stałej *lambda* = 0, co oznacza transformację logarytmiczną.

```
library(stats)
data(AirPassengers)
# 1. Transformacja B-C w celu ustabilizowania wariancji
AirPass.B_C<- BoxCox(AirPassengers, lambda = 0)
par(mfrow=c(2,1))
plot(AirPassengers)
plot(AirPass.B_C) # jest trend ale ustabilizowana wariancja
adf.test(AirPass.B_C, nlag = 4)
Acf(AirPass.B_C, lag.max=24)
```




```
Augmented Dickey-Fuller Test
alternative: stationary

Type 1: no drift no trend
     lag   ADF p.value
[1,]   0 0.913   0.902
[2,]   1 0.674   0.836
[3,]   2 0.796   0.871
[4,]   3 0.936   0.905
Type 2: with drift no trend
     lag   ADF p.value
[1,]   0 -1.82   0.400
[2,]   1 -2.02   0.321
[3,]   2 -1.65   0.465
[4,]   3 -1.61   0.481
Type 3: with drift and trend
     lag   ADF p.value
[1,]   0 -4.85    0.01
[2,]   1 -7.00    0.01
[3,]   2 -6.71    0.01
[4,]   3 -7.13    0.01
```

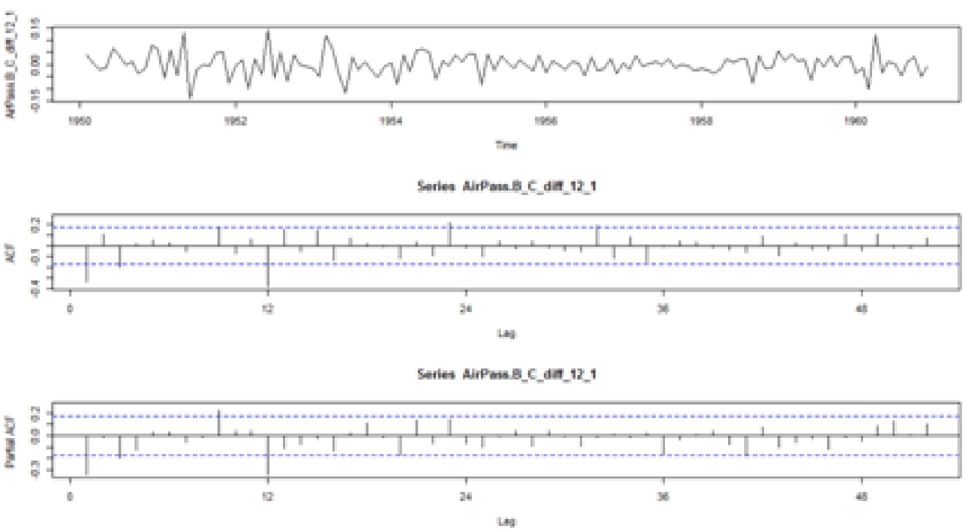
Stabilizowanie szeregu czasowego - cd

2. Proces różnicowania służy do ustabilizowania szeregu. Patrząc na autokorelację szeregu *AirPassengers*, można zadecydować o rzędzie różnicowania.

```
# 2. Transformacje różnicowe: sezonowa i prosta w celu wyeliminowania sezonowości i trendu
pom <- diff(AirPass.B_C, lag=12)    # różnicowanie sezonowe; jednokrotne z opóźnieniem  12
AirPass.B_C_diff_12_1 <- diff(pom, lag=1) # różnicowanie proste; jednokrotne z opóźnieniem 1
```

3. Po wykonaniu ww. trzech przekształceń następuje sprawdzenie szeregu wynikowego na okoliczność stacjonarności.

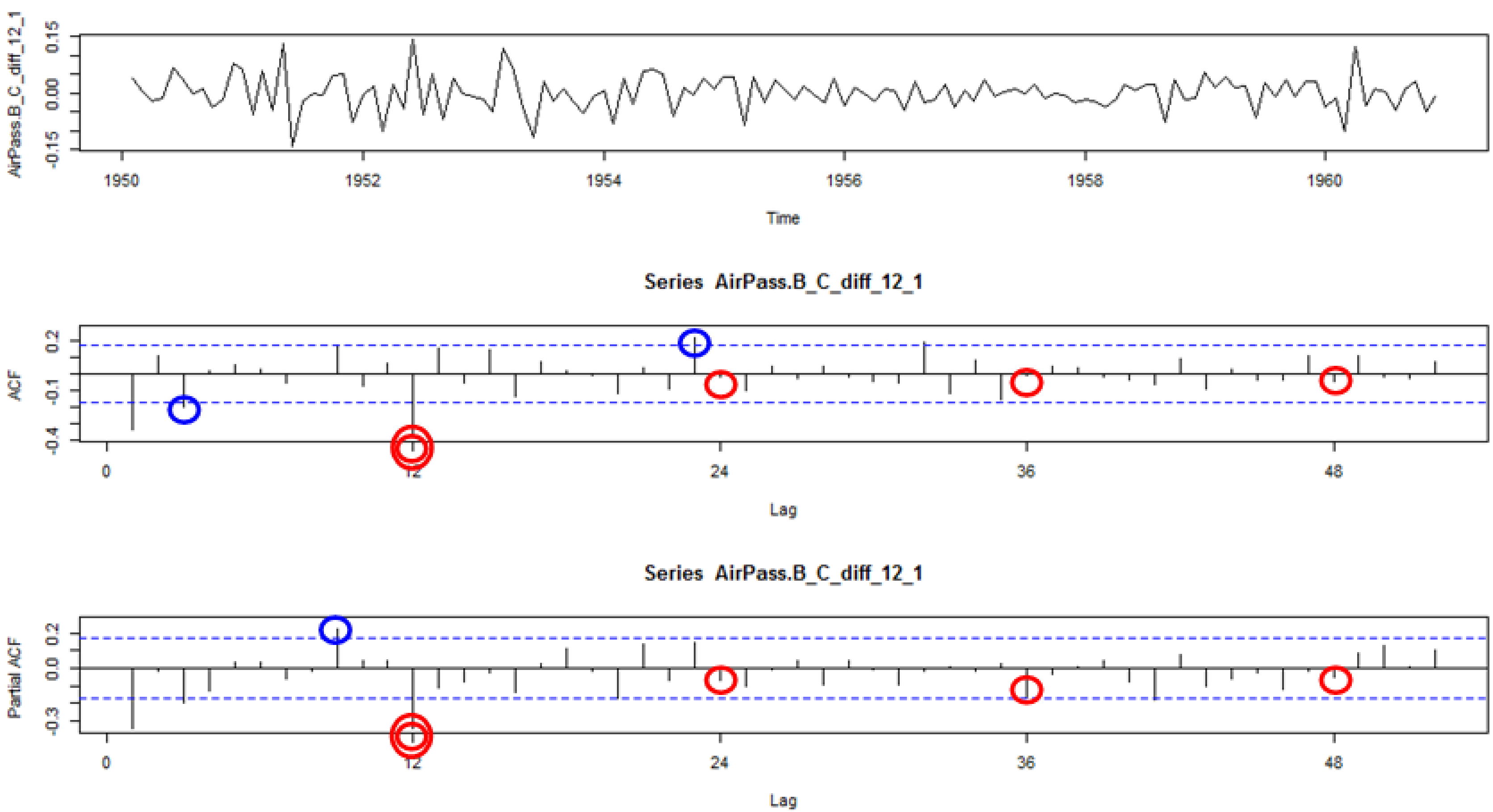
```
# 3. Sprawdzenie stacjonarności wyników różnicowania
adf.test(AirPass.B_C_diff_12_1, nlag = 4)
par(mfrow=c(3,1))
plot(AirPass.B_C_diff_12_1)
Acf(AirPass.B_C_diff_12_1, lag.max=52)
Pacf(AirPass.B_C_diff_12_1, lag.max=52
```




```
Augmented Dickey-Fuller Test
alternative: stationary

Type 1: no drift no trend
     lag    ADF p.value
[1,]   0 -16.25    0.01
[2,]   1  -9.34    0.01
[3,]   2  -8.65    0.01
[4,]   3  -7.70    0.01
Type 2: with drift no trend
     lag    ADF p.value
[1,]   0 -16.19    0.01
[2,]   1  -9.30    0.01
[3,]   2  -8.61    0.01
[4,]   3  -7.67    0.01
Type 3: with drift and trend
     lag    ADF p.value
[1,]   0 -16.16    0.01
[2,]   1  -9.29    0.01
[3,]   2  -8.62    0.01
[4,]   3  -7.70    0.01
----
Note: in fact, p.value = 0.01 means p.value <= 0.01
```

Analiza szeregu ustabilizowanego w celu identyfikacji postaci modelu



- autokorelacje dla opóźnień sezonowych
- autokorelacje dla opóźnień niesezonowych

Identyfikacja modelu SARIMA

Działania pozwalające określić parametry dla modelu szeregu czasowego po **zlogarytmowaniu** jego wartości:

- Składnik sezonowości: $s=12$
- Jednokrotne różnicowanie proste: $d=1$, jednokrotne różnicowane sezonowe: $D = 1$
- Analizuje się zachowanie dla ACF i PACF dla opóźnień będących wielokrotnościami okresu. Dla modeli sezonowych wartości tych funkcji dla opóźnień niebędących wielokrotnościami okresu są równe zero. W ww. szeregu:
    - P w składniku AR: ACF zanika stopniowo dla wielokrotności okresu i PACF znika dla $k > P \cdot s = 1 \cdot 12$
    - Q w składniku MA: ACF znika dla $k > Q \cdot s = 1 \cdot 12$ i PACF znika stopniowo dla wielokrotności okresu

  Propozycje dla parametrów sezonowych: $P = 1$, $Q=1$
- Zachowania ACF i PACF to kombinacja zaników wykładniczych i sinusoid tłumionych. Trudno wskazać prosta regułę pozwalającą na zidentyfikowanie p i q. Po zidentyfikowaniu parametrów sezonowych P i Q analizuje się przebiegi ACf i PACF dla opóźnień niebędących wielokrotnościami okresu ($h = 1, 2, 3, \ldots, s-1$). W ww. szeregu:
    - p w składniku AR: ACF zanika stopniowo i PACF znika dla $k > p = 23$ (ew. sprawdzić 3)
    - q w składniku MA: ACF znika dla $k > 9$ i PACF znika stopniowo

  Propozycje dla parametrów sezonowych: ($p = 23$, $q=9$) lub ($p = 3$, $q=9$)

SARIMA(p, d, q)(P, D, Q)$_s$

d – rząd zróżnicowania prostego
D – rząd zróżnicowania sezonowego
s – częstotliwość (sezonowość)
P – rząd sezonowy składnika AR
Q – rząd sezonowy składnika MA
p – rząd prosty składnika AR
Q – rząd prosty składnika MA

Uwzględniając identyfikację składników AR i MA:
SARIMA(23, 1, 0)(1, 1, 0)$_{12}$
SARIMA(3, 1, 0)(1, 1, 0)$_{12}$
SARIMA(0, 1,9)(0, 1, 1)$_{12}$

Rozważając modele rozbudowane:
SARIMA(23, 1, 9)(1, 1, 1)$_{12}$
SARIMA(3, 1, 9)(1, 1, 1)$_{12}$

Eksperyment z modelem pełniejszym ale mniej rozbudowanym:
SARIMA(1, 1, 1)(1, 1, 1)$_{12}$