SZEREGI CZASOWE

Estymacja i prognozowanie szeregów czasowych SARIMA
Analiza własności reszt. Ocena prognoz

Marzena Nowakowska

Wydział Zarządzania i Modelowania Komputerowego
Politechnika Świętokrzyska

Budynek C, p. 3.21

spimn@tu.kielce.pl

Estymacja modelu

Estymacja (dopasowanie, oszacowanie modelu) to wyznaczenie wartości parametrów modelu na podstawie danych zawartych w realizacji szeregu czasowego. Takie wartości są estymatorami prawdziwych wartości parametrów modelu.

W przypadku najszerszego modelu SARIMA liczba wartości do wyznaczenia jest równa $p + q + P + Q + 1$:

- p i P współczynników autoregresji zwykłej i sezonowej,
- $q + Q$ współczynników średniej ruchomej zwykłej i sezonowej,
- wariancji białego szumu.

Jeżeli w modelu uwzględni się stałą, to liczba parametrów może się zwiększyć o 1.

Najczęstsze metody estymacji: momentów (MM), najmniejszych kwadratów (LS), największej wiarygodności (ML).

Pakiet	Funkcje
stats	ar(), arima()
forecast	Arima(), auto.arima()
astsa	sarima()
FitAR	FitAR()
FitARMA	FitARMA()
itsmr	arma(), burg(), hannan(), ia(), yw()
TSA	arimax()
tseries	arma()

Funkcja Arima(x, ...) biblioteka forecast

Estymuje (dopasowuje) model ARIMA do jednowymiarowych szeregów czasowych. Jest rozszerzeniem funkcji *arima* z pakietu *stats*. Główna różnica polega na tym, że ta funkcja pozwala na uwzględnienie w modelu dryfu.

x	Wektor numeryczny lub szereg czasowy klasy *ts*.
order	Specyfikacja zwykłej (niesezonowej) części modelu SARIMA – rzędy (p, d, q); order = c(0, 0, 0).
seasonal	Specyfikacja sezonowej części modelu SARIMA, z domyślną wartością częstotliwości (przejętą z szeregu czasowego) – rzędy (P, D, Q); seasonal = list(order=c(0, 0, 0), period=NA).
include.mean	Flaga informująca o tym, czy model ma zawierać średnią (stałą). Wartością domyślną jest PRAWDA dla szeregów niezróżnicowanych, FAŁSZ dla szeregów zróżnicowanych.
include.drift	Flaga informująca o tym, czy w modelu ma być uwzględniony dryf liniowy (regresja liniowa); include.drift = FALSE.
include.constant	Flaga informująca, czy uwzględnić stałą w modelu. Jeśli ma wartość TRUE, to *include.mean* jest ustawiony na wartość TRUE dla szeregów niezróżnicowanych, a parametr *include.drift* ma wartość TRUE dla szeregów zróżnicowanych. Jeżeli w szeregu jest wystepuje więcej niż jedno różnicowanie, żadna stała nie jest uwzględniana niezależnie od wartości tego argumentu.
lambda	Parametr transformacji Boxa-Coxa.
	Flaga informująca o sposobie odwrócenia przekształcenia Boa-Coxa w przypadku prognoz; domyślnie biasadj = FALSE – transformacja wsteczna dostarczająca prognozy medianowe. Dla biasadj = PRAWDA – korekta w celu uzyskania średnich prognoz i dopasowanych wartości).

Funkcja Arima(x, ...) cd

transform.pars	Flaga logiczna informująca o tym, czy parametry części AR modelu mają być przekształcone tak, aby był zachowany warunek stacjonarności szeregu. Nie stosowany dla *method* = "CSS".
fixed	Opcjonalny wektor liczbowy o tej samej długości, co całkowita liczba współczynników do oszacowania w postaci: (ϕ_1, ..., ϕ_p θ_1, ..., θ_q, Φ_1, ..., Φ_P, Θ_1, ..., Θ_Q, μ), μ jest składnikiem przesunięcia. Gdy *include.mean*=TRUE, składnik musi być obecny a nieobecny, jeżeli w modelu jest różnicowanie. Wektor zawiera wartości parametrów, które mają mieć w modelu ustalone wartości (w szczególności, jeżeli dany parametr ma być odrzucony należy ma nadać wartość 0), wartość NA oznacza, że odpowiadający tej wartości parametr ma być estymowany.
init	Opcjonalny wektor liczbowy o tej samej długości, co całkowita liczba współczynników do oszacowania zawierający początkowe wartości parametrów.
method	Metoda estymacji parametrów modelu. Wykaz wartości: "CSS-ML" (domyśla, estymacja dwuetapowa, najpierw wartości początkowe, wykorzystane w drugim etapie z zastosowaniem ML), "ML", "CSS" (*conditional sum-of-squares*).

...

Wynikiem jest lista klasy *arima* zawierająca komponenty, m.in.:

- *x* szereg czasowy (dane),
- *sigma2* (estymator wariancji reszt innowacyjnych / rezyduów innowacyjnych,
- *coef* (wyestymowane współczynniki modelu),
- *var.coef* (macierz kowariancji współczynników modelu; na przekątnej są estymowaneparam.blStd wartości wariacji współczynników),
- *arma* (postać modelu SARMIA, zgodnie ze wzorcem: (ϕ_1, ..., ϕ_p θ_1, ..., θ_q, Φ_1, ..., Φ_P, Θ_1, ..., Θ_Q, μ)),
- miary oceny modelu: *aic* (obowiązujące dla metody ML), *aicc*, *bic*,
- *fitted* (wartości prognozowane przez model dla indeksów czasowych szeregu *x*).

Diagnostyka modelu

Aby można było stosować dopasowany model do dalszej analizy (np. konstrukcji prognoz), należy zweryfikować poprawność jego dopasowania, czyli wykonać diagnostykę modelu. Umożliwi to również porównanie konkurencyjnych modeli i wybór najlepszego (najlepiej dopasowanego do danych i spełniającego wymagane założenia).

Diagnostyka modelu obejmuje:

- analizy reszt
  - losowość (brak autokorelacji),
  - normalność rozkładu,
  - stała wariancja - homoskedastyczność wariancji,
- ocena istotności i jakości dopasowania modelu do danych na podstawie prognoz
  - kryteria informacyjne,
  - test istotności parametrów,
  - miary oceny dobroci dopasowania modelu do danych

Reszty i reszty innowacyjne

Reszty w modelu szeregu czasowego są tym, co pozostaje po dopasowaniu modelu. Są równe różnicy między obserwacjami a odpowiadającymi im dopasowanymi wartościami: $e_t = x_t - \hat{x}_t$, gdzie $\hat{x}_t$ jest wartością prognozowaną przez model.

Jeśli w modelu zastosowano transformację, to często przydatne jest przyjrzenie się resztom na przekształconej skali. Nazywa się je "resztami innowacyjnymi" (*innovation residuals*) lub "innowacjami" (*innovations*). Na przykład, jeżeli dla celów modelowania, dane zostały przekształcone logarytmicznie: $y_t = \log(x_t)$, reszty innowacyjne są dane przez $y_t - \hat{y}_t$, podczas gdy zwykłe reszty są dane przez ww. e_t. Jeśli nie zastosowano żadnego przekształcenia, wówczas reszty innowacyjne są identyczne jak zwykłe reszty i w takich przypadkach nazywa się je po prostu "resztami".

Reszty są przydatne w sprawdzaniu, czy model odpowiednio uchwycił informacje zawarte w danych.

Jeśli w resztach innowacyjnych można zaobserwować wzorce, model nie jest za dobry i prawdopodobnie powinien być ulepszony (zmieniony).

Diagnostyka modelu; losowość reszt

Brak autokorelacji reszt; sprawdza się, czy reszty można uznać za ciąg losowy (obecność autokorelacji sygnalizuje zależności(skorelowanie) bieżących wartości składnika losowego od wartości przeszłych):

- ✓ wykres reszt; nie powinno być widocznych regularnych wzorców (trendy, sezonowość) ani niejednorodnej wariancji,
- ✓ wykresy ACF i PACF nie powinny mieć widocznych istotnych korelacji; jest to tzw. test białoszumowości (biały szum = ciąg zmiennych losowych nieskorelowanych o jednakowym rozkładzie o średniej 0 (zero) i wariancji σ^2).
- ✓ testy autokorelacji, testy losowości reszt (ogólna postać H0: reszty są losowe):
  - test Ljung-Boxa

    H0: Dane są rozłożone niezależnie (reszty mają rozkłady niezależne, tj. korelacje w populacji, z której pobrano próbę, wynoszą 0, więc wszelkie obserwowane korelacje w danych wynikają z losowości procesu losowania).

    H1: Dane nie mają niezależnego rozkładu; wykazują korelację szeregową.

    Statystyka testowa zawiera opóźnienia w szeregu czasowym od opóźnienia 1 do podanej maksymalnej liczby opóźnień *k* (badane autokorelacje są uwzględniane, począwszy od autokorelacji z opóźnieniem 1, skończywszy na *k*-tym opóźnieniu). Statystyka ma rozkład chi-2 z liczbą stopni swobody równą *k*.

    Podczas testowania reszt oszacowanego modelu ARIMA stopnie swobody należy dostosować tak, aby odzwierciedlić oszacowanie parametrów. Na przykład dla modelu ARIMA(p,0,q) stopnie swobody powinny być ustawione na wartość: $k - (p + q)$, przy czym $k > p + q$.

Diagnostyka modelu; normalność rozkładu reszt

- ✓ wykres kwantylowy

  QQ plot, wykres rozrzutu kwantyli znormalizowanych reszt i kwantyli rozkładu teoretycznego N(0, 1)

  Jeżeli dane mają rozkład normalny, punkty na wykresie Q-Q będą leżeć na prostej przekątnej.

  Im bardziej punkty na wykresie znacznie odbiegają od prostej ukośnej linii, tym mniejsze prawdopodobieństwo, że zestaw danych ma rozkład normalny.
- ✓ histogram (zbliżony do rozkładu normalnego)
- ✓ testy zgodności z rozkładem normalnym; Shapiro-Wilka, Jarque-Bera;
  - H0: reszty mają rozkład normalny
  - H0: reszty mają rozkład różny od normalnego

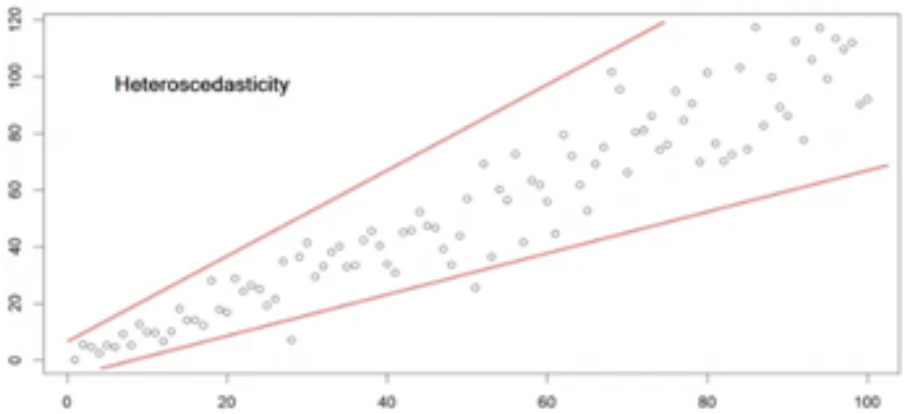
Diagnostyka modelu; homoskedastyczność reszt

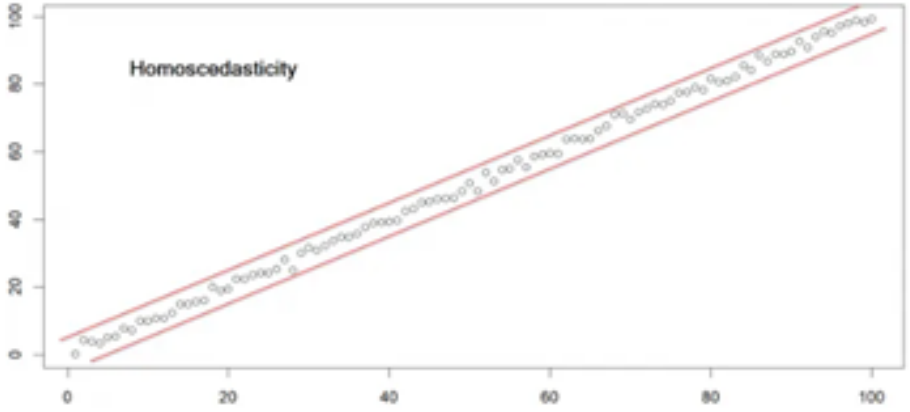
Heteroskedastyczność to sytuacja, w której wariancja reszt modelu regresji nie jest taka sama dla wszystkich przewidywanych (prognozowanych) wartości zmiennej celu. Innymi słowy, zmienność reszt (tj. składnika błędu) rośnie lub maleje w zakresie przewidywań. Pojęciem przeciwnym jest **homoskedastyczność**.

- ✓ ilustracja na wykresie rozrzutu *wartości prognozowane*(oś x) vs. *reszty* (oś y),
- ✓ test homoskedastyczności wariancji (jednorodnej wariancji) reszt; test portmanteau-Q, test Breusch-Pagana, test White'a.

  H0: Rozkład reszt ma stałą wariancję (reszty są homoskedastyczne).

  H1: Wariancja rozkładu reszt nie jest stała (reszty są heteroskedastyczne).





Źródło: https://www.codingprof.com/3-easy-ways-to-test-for-heteroscedasticity-in-r-examples/

Są funkcje w R realizujące ww. testy ale dla przypadku klasycznej regresji liniowej (w tym wielokrotnej). **Jeżeli student znajdzie rozwiązanie dla modelu ARIMA uzyska 10 punktów**. Diagnostyka wizualna jest dla tego przypadku polecana przez różnych autorów.

Funkcja Box.test(x, ...) biblioteka stats

Oblicza statystykę testową Boxa-Pierce'a lub Ljunga-Boxa do zbadania hipotezy zerowej o tym, czy jakakolwiek grupa autokorelacji szeregu czasowego jest różna od zera. Zamiast testować losowość przy każdym odrębnym opóźnieniu, testuje „ogólną" losowość na podstawie liczby opóźnień – jest to test portmanteau.

x	Szereg czasowy.
lag	Parametr informujący o tym, jaka ma być maksymalna wartości opóźnienia dla autokorelacji w celu wyliczenia statystyki testowej; wartość *k*
type	Wersja testu; dopuszczalne wartości: "Box-Pierce" (domyślna), "Ljung-Box" .
fitdf	Wartość, która musi być odjęta od liczby stopni swobody. mająca znaczenia, jeżeli szereg jest szeregiem reszt. Wartość domyślna: 0. W przypadku estymacji modelu ARIMA należy podać sumę wartości rzędów składowych modelu: $p + P + q + Q$; $k > p + P + q + Q$

Funkcja tsdiag(x, ...) biblioteka stats

Oblicza statystykę testową Boxa-Pierce'a lub Ljunga-Boxa do zbadania hipotezy zerowej o tym, czy jakakolwiek grupa autokorelacji szeregu czasowego jest różna od zera. Zamiast testować losowość przy każdym odrębnym opóźnieniu, testuje „ogólną" losowość na podstawie liczby opóźnień – jest to test portmanteau.

object	Dopasowany model szeregu czasowego.
gof.lag	Maksymalna liczba opóźnień dla testu portmanteau losowości dla opóźnień od 1 do opóźnienia *gof.lag*.

…

Wyznaczone w funkcji *tsdiag* p-wartości dla testu losowości nie uwzględniają korekty liczby stopni swobody, którą należy zastosować analizując reszty z dopasowanego modelu, co może doprowadzić do błędnych wniosków.

Funkcja qqnorm(x, ...) biblioteka stats

Tworzy wykres QQ kwantyli wartości *x* (empirycznych) i kwantyli rozkładu N(0,1) (teoretycznych).

x	Wektor numeryczny (w szczególności szereg czasowy klasy *ts*).
plot.it	Flaga informująca o tym, czy wykres ma być wyświetlany; *plot.it* = TRUE
ylim, main, xlab, ylab	Parametry graficzne; main="Normal Q-Q plot", xlab = "Theoretical Quantiles", ylab = "Sample Quantiles".

…

Funkcja qqline(x, ...) biblioteka stats

Dodaje linię do „teoretycznego”, domyślnie normalnego, wykresu kwantyl-kwantyl, który przechodzi przez określone kwantyle; domyślnie pierwszy i trzeci kwartyl.

x	Wektor numeryczny (w szczególności szereg czasowy klasy *ts*).
distribution	Funkcja kwantylowa dla teoretycznego rozkładu odniesienia; *distribution* = *qnorm*.
…	

Funkcja shapiro.test(x) biblioteka stats

Dostarcza wyniki testu Shapiro-Wilka na normalność rozkładu zmiennej reprezentowanej przez szereg *x*. wynikiem jest obiekt klasy *htest* zawierający składowe m.in.: statystyka testowa, p-wartośc.

Funkcja hist(x, plot = TRUE, ...)

Wyznacza histogram podanych wartości danych. Zwraca wynik klasy *histogram*. Jeśli *plot* = TRUE, jest wykreślany histogram dla wynikowego obiektu.

x	Wektor numeryczny (w szczególności szereg czasowy klasy *ts*), dla którego ma być utworzony histogram.
plot	Flaga informująca o tym, czy histogram ma być kreślony; *plot* = *TRUE*.
freq	Flaga informująca o tym, czy kreślony ma być histogram gęstości (FALSE) czy częstości (TRUE, wartość domyślna).
ylim, xlim, main, xlab, ylab	Parametry graficzne; *main*= *paste*("Histogram of" , nazwa_wektora_x).
col	Kolor, który ma być użyty do wypełnienia słupków; *col* = NULL.
border	Kolor obramowania wokół słupków. Domyślnie używany jest standardowy kolor pierwszego planu.
…	

Przykład. Diagnozowanie reszt

```
# Program ARIMAResztyDiagnoza.R

library(ggplot2)
data(WWWusage)
plot(WWWusage)
library(forecast)
arima.model <- arima(WWWusage , order=c(3,1,0))
ARIMA.model <- Arima(WWWusage , order=c(3,1,0))
wyniki <-data.frame(WWWusage, ARIMA.model$x, ARIMA.model$fitted,
                    ARIMA.model$residuals, ARIMA.model$x-ARIMA.model$fitted)
View(wyniki)


# 1. Losowość reszt
reszty <- ARIMA.model$residuals
plot(reszty)  # wykres reszt
par(mfrow=c(2,1))
library(forecast)
Acf(reszty, lag.max=40); Pacf(reszty, lag.max=40)  # wykresy autokorelacji dla reszt
for (d in seq(from=4,to=20,by=1))
  { wyn_LB <- Box.test(reszty, type="Ljung-Box", lag=d, fitdf=3)
    print(d); print(wyn_LB)
  }
# wykresy diagnostyczne - funkcja tsdiag
tsdiag(ARIMA.model, gof.lag=20)
```

Przykład. Diagnozowanie reszt, cd

```
# 2. Normalność rozkładu reeszt
# wykresy kwantylowe
qqnorm(reszty, main="Wykres QQ dla reszt"); qqline(reszty)

# Wykres histogramu i funkcji gestosci rozkladu normalnego
 hist(reszty, main="Histogram gęstości reszt modelu \nARIMA dla zbioru WWWusage",
        col="darkmagenta",border="darkblue", freq=FALSE)
 pkty1 <- seq(min(reszty), max(reszty), length = 40)
 fun1 <- dnorm(pkty1, mean = mean(reszty), sd = sd(reszty))
 lines(pkty1, fun1, col = "blue", lwd = 2)

 print(shapiro.test(reszty))

# library("tseries")    # do testu Jarque-Bera
# print(jarque.bera.test(reszty))

# 3. Homoskedastyczność reszt
library("lattice")
xyplot(ARIMA.model$residuals ~ ARIMA.model$fitted, xlab="Prognozy", ylab="Reszty",
       pch=16)
```

Ocena jakości modelu; kryteria informacyjne

Kryteria informacyjne oceniają jakość dopasowania modelu na podstawie danych historycznych (wykorzystanych do budowy modelu), uwzględniając jednoczenie stopień złożoności modelu.

Dążenie do jak najlepszej zgodności z danymi uczącymi (np. poprzez zwiększenie liczby parametrów) może doprowadzać do zjawiska nadmiernego dopasowania (*overfitting).*

W związku z powyższym w kryteriach informacyjnych oceny modelu występuje składnik kary, definiując ogólną postać miary następująco:

$$Miara\_oceny\_modelu = -2\ln(L) + składnik\_kary$$

gdzie L jest funkcją wiarygodności.

- AIC - Aikaike Information Criterion:
  $$AIC(p, q) = -2\ln(L) + 2(p + q + k + 1 + P + Q)$$
- AICC - Corrected Aikaike Information Criterion:
  $$AIC(p, q) = -2\ln(L) + 2(p + q + k + 1 + P + Q)\cdot n/(n - p - q - k - 2 - P - Q)$$
- BIC - Bayesian Information Criterion:
  $$AIC(p, q) = -2\ln(L) + 2(p + q + k + 1 + P + Q)\cdot \ln(n)$$

W ww. wzorach $k = 0$, jeżeli w modelu nie jest uwzględniona stała, $k = 1$, jeżeli występuje niezerowa stała.

Kryteria informacyjne - uwagi

- Na podstawie kryterium informacyjnego, spośród kilku alternatywnych modeli wybiera się ten, dla którego wartość kryterium jest mniejsza.
- Stosowanie kryteriów informacyjnych do wyboru modelu nie zastępuje weryfikacji jego poprawności; model musi pozytywnie przejść testy diagnostyczne dotyczące reszt.
- Nie można porównywać kryteriów informacyjnych dla modeli, które wyestymowano dla danych oryginalnych oraz tych zbudowanych na podstawie danych przekształconych (np. po zastosowaniu transformacji Boxa-Coxa).

Ocena jakości modelu; istotność współczynników

Jakość dopasowania modelu można niekiedy poprawić eliminując parametry (współczynniki), które są statystycznie nieistotne, czyli nie odgrywają istotnej roli w wyjaśnianiu dynamiki czasowej ocenianego zjawiska. Formalnie, wykonuje się test istotności dla hipotez:

H0: *prawdziwy_parametr_modelu* = 0

H1: *prawdziwy_parametr_modelu* ≠ 0

Można poszukać testu w Internecie (np. *t_test* w bibliotece BETS opracowanej w Brazylii).

Można jednak zastosować uproszczoną procedurę oceny istotności parametrów następująco:

1. Dla każdego parametru p_i wyznaczyć wskaźnik

$$wskIst = \frac{\widehat{p_1}}{1{,}96 \cdot s.e.(\widehat{p_1})}$$

gdzie: $\widehat{p_1}$ jest estymatorem prawdziwej wartości parametru, $s.e.(\widehat{p_1})$ jest oszacowanym błędem standardowym estymatora.

2. Jeżeli $|wskIst| < 1$, to parametr p_i uznaje się za statystycznie nieistotny. W przeciwnym przypadku za istotny statystycznie, na poziomie istotności 0,05 (1,96 jest kwantylem rzędu 0,975 rozkładu N(0, 1)).

Parametry, które okażą się nieistotne statystycznie należy wyeliminować ze zbyt złożonego modelu, wykonując reestymację modelu i wskazując , które składowe modelu należy zachować.

Parametry, które okażą się nieistotne statystycznie należy wyeliminować ze zbyt złożonego modelu, wykonując reestymację modelu i wskazując , które składowe modelu należy zachować.

Przykład; kryteria informacyjne i istotność parametrów

```
# Program ARIMAOcena.R

library(forecast)
data(AirPassengers)

# Estymacja modelu ARIMA z sezonowością
Air.model1 <- Arima(AirPassengers,order=c(0,1,1),
             seasonal=list(order=c(0,1,1),period=12),lambda=0)

View(Air.model1)  # wykaz wynikow fukcji Arima

#  1. Miary informacyjne
Air.model1
AIC <- cat("AIC: ", Air.model1$aic)
AICc  <- cat("AICC: ", Air.model1$aicc)
BIC <- cat("AICC: ", Air.model1$bic)


# 2. Badanie istotności parametrów modeli ARMIA
# wyświetlone wartości s.e. to pierwiastki z przekatnej
# estymowanej macierzy kowariancji
param <- Air.model1$coef
param.bldStd <- sqrt(diag(Air.model1$var.coef))
wskIst <- param/param.bldStd
wskIstWynik <- which(abs(wskIst) >= 1) # wskazanie parametrow istotnych
```

Ocena jakości modelu; dokładność prognoz

Prognoza jest to oszacowanie wartości zmiennej w przyszłości, przy użyciu narzędzi matematycznych.

Oprócz kryteriów informacyjnych, do oceny poprawności dopasowania modelu można wykorzystać miary, które oceniają błędy prognoz na podstawie dopasowanego modelu.
Zbudowane na podstawie modelu ekonometrycznego, prognozy można podzielić na dwie zasadnicze grupy: prognozy **ex post** oraz prognozy **ex ante**.

- Prognozy *ex post* oparte są na znanych wartościach zmiennych objaśniających. W momencie, gdy zna się zrealizowaną wartość zmiennej prognozowanej – prognoza *ex post* staje się prognozą wygasłą.
- Prognozy *ex ante* są natomiast oparte na nieznanych wartościach zmiennej objaśniającej lub zmiennych objaśniających. Te nieznane w momencie prognozowania wartości dotyczące okresu prognozowanego należy – w określony sposób – ustalić.

Trafność prognozy wygasłej określa się po upływie czasu, na który prognoza była wyznaczona. Stopień trafności prognozy wygasłej ilościowej mierzy się za pomocą błędów *ex post*.
Błąd ***ex post*** to wartość odchylenia rzeczywistych realizacji zmiennej prognozowanej od obliczonych prognoz.

- ME: Mean Error
- RMSE: Root Mean Squared Error
- MAE: Mean Absolute Error
- MPE: Mean Percentage Error
- MAPE: Mean Absolute Percentage Error
- MASE: Mean Absolute Scaled Error

Ocena jakości modelu; miary dopasowania

Błąd pojedynczej prognozy PE_t (*Prediction Error*)

$$BP_t = x_t - \hat{x}_t$$

gdzie: x_t jest obserwowaną wartością szeregu w chwili *t*, $\hat{x}_t$ jest wartością szeregu w chwili *t* prognozowaną przez model.

- Błąd średni prognozy *ME*

$$ME = \frac{1}{n}\sum_{t=1}^{n} x_t - \hat{x}_t$$

Wartość ME powinna być równa zero lub bliska zeru; gdy zaobserwowane odchylenie od zera jest dodatnie, wnioskujemy, że prognozy wygasłe są niedoszacowane; gdy zaobserwowane odchylenie od zera jest ujemne, wnioskujemy, że prognozy wygasłe są przeszacowane.

- Pierwiastek błędu średniokwadratowego *RMSE*

$$RMSE = \sqrt{\frac{1}{n}\sum_{t=1}^{n} (x_t - \hat{x}_t)^2}$$

RMSE dostarcza wynik w tej samej jednostce miary, co dane. Wada miary jest stosunkowo duża podatność na obserwacje odstające.

- Błąd średni prognozy *MAE*:

$$MAE = \frac{1}{n}\sum_{t=1}^{n} (x_t - \hat{x}_t)$$

gdzie: *n* jest liczbą obserwacji.

Średni błąd odnosi się do średniej wszystkich błędów w zestawie. Wartość średniego błąd zwykle nie jest przydatna, ponieważ wartości dodatnie i ujemne wzajemnie się znoszą.

Ocena jakości modelu; miary dopasowania – cd 1

- Średni błąd procentowy *MPE*

$$MPE = \frac{1}{n}\sum_{t=1}^{n}\frac{x_t - \hat{x}_t}{x_t}$$

MPE może być użytecznym kryterium do wykrywania systematycznego obciążenia w prognozach; np. duża wartość dodatnia może świadczyć o systematycznym zawyżaniu prognozy dla kolejnych okresów. Z drugiej strony duże dodatnie i ujemne wartości mogą się wzajemnie znosić, prowadząc do wartości bliskiej zero, co mylnie może świadczyć o dobrej jakości prognoz.

- Średni bezwzględny błąd procentowy *MAPE*

$$MAPE = \frac{1}{n}\sum_{t=1}^{n}\left|\frac{x_t - \hat{x}_t}{x_t}\right|$$

MAPE jest jedną z najbardziej popularnych miar dokładności prognozy opartych na względnym (procentowym) błędzie prognozy. Nie może być stosowana dla wartości mianownika równych lub bliskich zero.

Ocena jakości modelu; miary dopasowania – cd 2

- Średni bezwzględny błąd skalowany *MASE* dla liczby prognoz równej *n*

$$MASE = \frac{1}{n}\sum_{i=1}^{n} q_i$$

gdzie q_i jest błędem skalowanym prognozy dla *i*-tego okresu:

  - dla danych niesezonowych:

$$q_i = \frac{|x_t - \hat{x}_t|}{\frac{1}{T-1}\sum_{t=2}^{T}|x_t - x_{t-1}|}$$

Mianownik jest średnim błędem bezwzględnym jednoetapowej „metody prognozowania naiwnego" na zbiorze uczącym, w której jako prognoza jest przyjmowana rzeczywista wartość z poprzedniego okresu ($\hat{x}_t = x_{t-1}$).

  - dla danych sezonowych o okresie *s*:

$$q_i = \frac{|x_t - \hat{x}_t|}{\frac{1}{T-s}\sum_{t=s+1}^{T}|x_t - x_{t-s}|}$$

T jest długością zbioru uczącego.
Główna różnica w stosunku do metody dla niesezonowych szeregów czasowych polega na tym, że mianownik jest średnim błędem bezwzględnym jednoetapowej „sezonowej naiwnej metody prognozowania" na zbiorze uczącym, w której jako prognoza jest przyjmowana rzeczywista wartość z poprzedniego sezonu ($\hat{x}_t = x_{t-s}$).

Przy porównywaniu metod prognozowania preferowaną metodą jest metoda o najniższym *MASE*. Gdy MASE < 1, to oznacza, że analizowana metoda prognozowania daje średnio mniejsze błędy niż metoda naiwna.

Funkcja summary(object, ...)

Tworzy podsumowania wyników różnych funkcji dopasowania modelu.

object	Obiekt, dla którego są wyznaczone statystyki.
…	

Funkcja accuracy(object, ...)

Zwraca zestaw sumarycznych miar dokładności prognozy, w szczególności dla modelu ARIMA.

object	Obiekt klasy *forecast* lub wektor numeryczny zawierający wartości prognoz, dla którego są wyznaczone statystyki. Może to być tez wynik estymacji modelu ARIMA zwrócony przez funkcję *Arima*.
x	Opcjonalny wektor zawierający obserwowane wartości szeregu statystycznego, dla którego były wyznaczone prognozy.
test	Wskaźnik, które elementy x i f mają zostać przetestowane. Jeśli test ma wartość NULL, używane są wszystkie elementy. W przeciwnym razie test jest wektorem numerycznym zawierającym indeksy elementów do wykorzystania w teście.
…	

Zwraca obiekty klasy *matrix* zawierający statystyki: ME, RMSE, MAE, MPE, MAPE, MASE, ACF1.

Przykład; miary dopasowania, wybór modelu

```
# Program ARIMAWybor.R

# 1. estymacja zidentyfikowanego modelu 2
library(forecast)
Air.model2 <- Arima(AirPassengers,order=c(3,1,9),
                    seasonal=list(order=c(1,1,1),period=12),lambda=0)
# 2. reestymacja modelu 2 z parametrami istotnymi statystycznie
# utworzenie wektora z informacją o istotności parametrow modelu 2:
# 0 - nieistotny, NA - istotny
param1 <- Air.model2$coef
param1.bldStd <- sqrt(diag(Air.model2$var.coef))
wskIst1 <- param1/param1.bldStd
wskIst1Wynik <- which(abs(wskIst1) >= 1) # wskazanie parametrów istotnych

model2.flagi <- numeric(14)  # liczba parametrów modelu; wszystkie zera
model2.flagi[wskIst1Wynik] <- NA # (oznakowanie istotnych; czyli do estymacji)

# 3. dopasowanie modelu z pominięciem nieistotnych parametrow
Air.model2Reest <- Arima(AirPassengers,order=c(3,1,9),
                         seasonal=list(order=c(1,1,1),period=12),lambda=0,
                         fixed=model2.flagi)
# 4. porównanie miar informacyjnych modelu 2 oryginalnego i reestymowanego
AIC_m2_vs_m2rest <- cat("AIC: ", Air.model2$aic, Air.model2Reest$aic)
AICc_m2_vs_m2rest <- cat("AICC: ", Air.model2$aicc, Air.model2Reest$aicc)
BIC_m2_vs_m2rest <- cat("BIC: ", Air.model2$bic, Air.model2Reest$bic)

# 5. Ocena dokładności prognoz
summary(Air.model2)
summary(Air.model2Reest)
accuracy(Air.model2)
accuracy(Air.model2Reest)
```

p=3			q=9									P=1	Q=1
ar1	ar2	ar3	ma1	ma2	ma3	ma4	ma5	ma6	ma7	ma8	ma9	sar1	sma1
1	2	3	4	5	6	7	8	9	10	11	12	13	14

Ocena modelu na zbiorze testowym

Ocena dokładności prognoz na **danych treningowych (uczących)**, tzn. wykorzystywanych do budowy modelu (*in-sample accuracy*), nie jest w pełni wiarygodna; model zazwyczaj będzie dobrze dopasowany do swoich danych uczących. Aby taka ocena była bardziej wiarygodna, skuteczność różnych metod prognozowania powinno się wykonywać na danych niezależnych (*out-of-sample accuracy*). Takie dane noszą nazwę **zbioru testowego**. Zbiór treningowy i testowy tworzy się ze zbioru pierwotnego, zazwyczaj w drodze losowanie, rozdzielając je w proporcji 70%-30% lub 80%-20% długości szeregu statystycznego. W przypadku szeregu czasowego cześć testową stanowi końcówka obserwacji szeregu.

Model buduje się na danych treningowych, można oceniać również na tych danych, ale wskazane jest ocenić model na danych testowych.

Funkcja forecast(object, ...) biblioteka forecast

Funkcja wyznacza prognozy szeregów czasowych lub modeli szeregów czasowych. Funkcja wywołuje określone metody, które zależą od klasy pierwszego argumentu.

object	Szereg czasowy lub model szeregów czasowych, dla których wymagane są prognozy.
h	Liczba okresów do prognozowania.
level	Poziom ufności dla przedziałów predykcji.
lambda	Parametr transformacji Boxa-Coxa. Jeśli lambda="auto", to transformacja jest wybierana automatycznie za pomocą BoxCox.lambda. Transformacja jest ignorowana, jeśli NULL. W przeciwnym razie, dane są przekształcone zgodnie z wartością parametru.
biasadj	Flaga do wskazania transformacji Box-Coxa. Ma wartość TRUE, aby użyć skorygowanej średniej przekształconej wstecz dla transformacji Boxa-Coxa. Jeśli przekształcone dane są używane do tworzenia prognoz i dopasowanych wartości, zwykła transformacja wsteczna da w wyniku prognozy medianowe. Jeśli biasadj ma wartość TRUE, zostanie wprowadzona korekta w celu uzyskania średnich prognoz i dopasowanych wartości.

...

Zwraca obiekty klasy *matrix* zawierający statystyki: ME, RMSE, MAE, MPE, MAPE, MASE, ACF1.

Funkcja forecast(object, ...) cd

Funkcja wyznacza prognozy szeregów czasowych lub modeli szeregów czasowych. Funkcja wywołuje określone metody, które zależą od klasy pierwszego argumentu.

- *model*; lista zawierająca informacje na temat oszacowanego modelu,
- *method*; postać prognozowanego modelu,
- *mean*; prognozy punktowe dla *h* punktów czasowych,
- *lower*; dolna granica przedziałów predykcji,
- *upper*; górna granica przedziałów predykcji,
- *level*; wartości ufności skojarzone z przedziałami predykcji,
- x; oryginalne wartości szeregu,
- *residuals*; rezydua modelu dopasowanego,
- *fitted*; wartości dopasowane - prognozy jednokrokowe (dla danych treningowych)

W przypadku prognozowania za pomocą modelu ARIMA, w którym występuje część MA, prognozy mogą nie zostać obliczone generując komunikat ostrzegawczy:

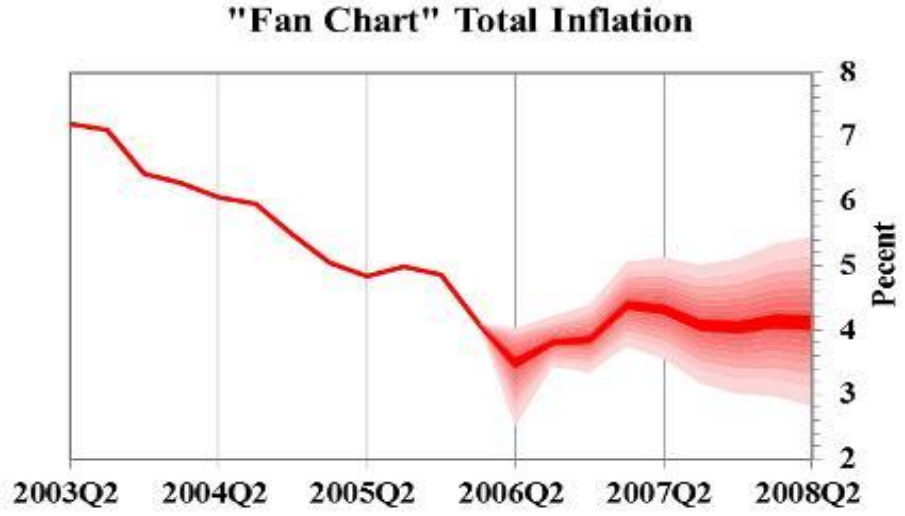
```
Warning message:
In predict.Arima(object, n.ahead = h) : MA part of model is not invertible
```

Jeżeli współczynniki MA nie są odwracalne, należy wywołać odwracalność, wybrać inną kolejność modelu lub przekazać własne parametry do wykazu wartości początkowych.

Wykres wachlarzowy

W analizie szeregów czasowych **wykres wachlarzowy** to wykres, który łączy prosty wykres liniowy obserwowanych danych z przeszłości, pokazując zakresy możliwych wartości przyszłych danych wraz z linią przedstawiającą centralne oszacowanie lub najbardziej prawdopodobną wartość przyszłych wyników.

W miarę jak prognozy stają się coraz bardziej niepewne, im dalej w przyszłość, te zakresy prognoz rozprzestrzeniają się, tworząc charakterystyczne kształty klina lub „wachlarza”, stąd termin. Alternatywne formy wykresu mogą również zawierać niepewność dotyczącą danych przeszłych, takich jak dane wstępne, które podlegają rewizji.



Cechy przedziałów prognozy:

- pozwalają ocenić, jaki może być najlepszy a jaki najgorszy możliwy scenariusz,
- można je wykorzystać do porównania dokładności różnych metod (modeli) konstrukcji prognoz punktowych,
- łatwiej je interpretować niż miary takie jak: *MAE*, *MSE* czy *RMSE*.

Przykład. Prognozy i ocena na zbiorze testowym

```
# Program ARIMATrnTst

# 1. podział na zbiór treningowy i testowy
# length(AirPassengers) = 144, tst: 144*0.2 = 28,8 ~= 30
# trn: 144*0.80 = 114

library(forecast)
Air.trn<-window(AirPassengers, end=c(1958, 6))
Air.tst<-window(AirPassengers, start=c(1958, 7))

# 2. modele na zbiorze treningowym
Air.m1.trn <- Arima(Air.trn,order=c(3,1,0),
                    seasonal=list(order=c(1,1,0),period=12),lambda=0)
Air.m2.trn <- Arima(Air.trn, order=c(0,1,9),
                    seasonal=list(order=c(0,1,1),period=12),lambda=0)

# 3. wykresy szeregu i prognoz dla zbioru treningowego, miary
plot(Air.trn, lwd=2)
lines(Air.m1.trn$fitted, col="red")
lines(Air.m2.trn$fitted, col="blue")
kryteria <- c("MAE", "RMSE", "MAPE", "MASE")
m1 <- accuracy(Air.m1.trn)[,kryteria]; print(m1)
m2 <- accuracy(Air.m2.trn)[,kryteria]; print(m2)
```

Przykład. Prognozy i ocena na zbiorze testowym - cd

```
# 4. wyznaczenie prognoz dla kolejnych 30 miesięcy, miary
library(forecast)
Air.m1.prog <- forecast(Air.m1.trn, h=30, lambda=0, biasadj=TRUE)
Air.m2.prog <- forecast(Air.m2.trn, h=30, lambda=0, biasadj=TRUE)

# 5. wykresy szeregu dla zbioru testowego
x_min <- min(Air.m2.prog$mean, Air.m1.prog$mean, Air.tst)
x_max <- max(Air.m2.prog$mean, Air.m1.prog$mean, Air.tst)
y_zakres <-c(x_min, x_max)
plot(Air.tst, col="black", ylim=y_zakres, lwd=2)
lines(Air.m1.prog$mean, col="red")
lines(Air.m2.prog$mean, col="blue")

# 6. Wykresy wachlarzowe
par(mfrow=c(2,1))
plot(Air.m1.prog)
plot(Air.m2.prog)
```

Funkcja auto.arima(object, ...) biblioteka forecast

Funkcja zwraca obiekt typu arima będący oszacowana automatycznie postać modelu ARIMA.

x	Szereg czasowy klasy *ts*.
d, D	Krotność różnicowania z opóźnieniem 1 i sezonowego; brak wartości (domyślnie NA) spowoduje dopasowanie automatyczne.
max.p, max.q	Maksymalna wartość rzędów; *p* i q.
max.P, max.Q	Maksymalna wartość rzędów; *P* i *Q*.
Max.order	Maksymalny rząd modelu *p*+*q*+*P*+*Q* w przypadku, gdy nie jest stosowana metoda krokowa.
Max.d, max.D	Maksymalna krotność różnicowania z opóźnieniem 1 i sezonowego.
Start.p, start.q, start.P, start.Q	Wartości początkowe dla parametrów procesu.
stationary	Flaga, czy ograniczyć poszukiwanie do modeli stacjonarnych (TRUE); .
ic	Kryteria informacyjne stosowane do wyboru modelu; "aic", "aicc", bic".
stepwise	Flaga, czy stosować krokowy wybór optymalnego modelu (TRUE).
approximation	Czy podczas poszukiwania optymalnego modelu zastosować przybliżoną wartość kryterium informacyjnego w celu przyspieszenie obliczeń.
lambda	Opcjonalny parametr transformacji Boxa-Coxa.
…	

Prognozowanie szeregów czasowch

Prognozowanie przyszłych wartości szeregu na podstawie danych historycznych jest jednym z głównych zadań analizy szeregów czasowych.

Najbardziej popularnymi metodami (algorytmami) stosowanymi do prognozowania szeregów czasowych są:

- metody naiwne (proste)
- metody wykorzystujące modele z rodziny **ARIMA**
- algorytmy wygładzania wykładniczego
- prognozy tworzone na podstawie dekompozycji szeregu