

## Okruh 4 - Ekonometria časových radov

Jakub Chalmovianský,  
Tatiana Keseliová,  
Vlastimil Reichel

4. januára 2021

## Kapitola 4

# Ekonometria časových radov v Matlabe

Pod pojmom časový rad (dynamický rad, vývojový rad) rozumieme postupnosť pozorovaní  $y_t$  kvantitatívnej charakteristiky usporiadaný v čase od minulosti do prítomnosti pre  $t = 1, 2, \dots, N$ . Pri klasickej analýze časových radov predpokladáme, že každý časový rad môže obsahovať štyri zložky: trend  $T_t$ , sezónnu zložku  $S_t$ , cyklickú zložku  $C_t$  a náhodnú zložku  $\epsilon_t$  v tvare

$$y_t = T_t + S_t + C_t + \epsilon_t.$$

Trend je všeobecná tendencia vývoja skúmaného javu za dlhé obdobie. Je výsledkom dlhodobých a stálych procesov. Môže byť rastúci, klesajúci, prípadne môže časový rad existovať bez trendu. Sezónna zložka je pravidelne sa opakujúca odchýlka od trendovej zložky. Perióda tejto zložky je menšia ako celková veľkosť sledovaného obdobia. Cyklická zložka udáva kolísanie okolo trendu v dôsledku dlhodobého cyklického vývoja. Náhodná (stochastická) zložka sa nedá popísať žiadnou funkciou času. Chápeme ju teda ako nekontrolovaný, na sebe nezávislý vplyv alebo inováciu v časovom rade.

Autoregresný model časového radu je založený na poznatku, že každá hodnota v časovom rade, závisí na predošlých hodnotách tohto radu. Autoregresný model stupňa  $p$ ,  $AR(p)$ , je definovaný ako:

$$y_t = a_0 + a_1 y_{t-1} + a_2 y_{t-2} + \dots + a_p y_{t-p} + \epsilon_t,$$

kde  $a_1, a_2, \dots, a_p$  sú autoregresné koeficienty a  $\epsilon_t$  je proces bieleho šumu s nulovou strednou hodnotou a konštantným rozptylom  $\sigma^2$ .

Model kľazavých priemerov stupňa  $q$ ,  $MA(q)$ , je jednoduchý proces využívaný na modelovanie dynamiky stacionárnych časových radov, ktorý vieme zapísať ako:

$$y_t = \epsilon_t + \beta_1\epsilon_{t-1} + \beta_2\epsilon_{t-2} + \cdots + \beta_q\epsilon_{t-q},$$

kde  $\beta_1, \beta_2, \dots, \beta_p$  sú parametre modelu a  $\epsilon_t$  je proces bieleho šumu. Kombináciou  $AR(p)$  procesu a  $MA(q)$  procesu vzniká  $ARMA(p,q)$  model.

Konkrétny  $ARMA(p,q)$  model vytvoríme v Matlabe pomocou funkcie **arima**, zápis:  $m = \text{arima}(p,0,q)$ . Pre odhad tohto  $ARMA(p,q)$  na dátach časovej rady je potrebné využiť funkciu **estimate**, zápis:  $[\text{ARMA}(p,q), \text{Cov}, \log L, \text{info}] = \text{estimate}(m, Y)$ , kde vstupnými parametrami sú model  $m$  získaný z funkcie **arima** a príslušné dáta  $Y$ . Výstupnými parametrami sú odhad  $ARMA(p,q)$  modelu, variančne-kovariančná matica  $\text{Cov}$ , logaritmus vierohodnosti  $\log L$  a sumarizácia výsledkov  $\text{info}$ . Získanie rezíduí (odchýliek skutočných a modelových hodnôt) využívame príkaz **infer**, zápis:  $e = \text{infer}(\text{ARMA}(p,q), Y)$ .

Aurokorelačná (ACF) a parciálna autokorelačná (PACF) funkcia slúžia k identifikácii a odhadu modelu časových radov. Na vykreslenie ACF v Matlabe využívame funkciu **autocorr**, zápis:  $\text{autocorr}(Y)$ , kde  $Y$  je vyšetřovaný časový rad. PACF vykresľuje funkcia **parcorr**, zápis je rovnaký ako v prípade ACF.

Pre výpis informačných kritérií využívame v Matlabe funkciu **aicbic**, zápis:  $[\text{aic}, \text{bic}] = \text{aicbic}(\log L, \text{numParam}, \text{numObs})$ . Vstupnými parametrami sú logaritmus vierohodnosti príslušného modelu  $\log L$ , počet odhadovaných parametrov  $\text{numParam}$  v súvislosti s logaritmom vierohodnosti a dĺžka dátového súboru  $\text{numObs}$ . Výstupnými parametrami sú hodnoty Akaikeho,  $\text{aic}$ , a Schwarzovho,  $\text{bic}$ , informačného kritéria.

Na vytvorenie predpovedí slúži v prostredí Matlabu funkcia **forecast**, zápis:  $[\text{f}, \text{MSE}] = \text{forecast}(\text{Mdl}, \text{numperiods}, Y_0)$ . Vstupné parametre sú model  $\text{Mdl}$  (plne špecifikovaný arima model ako výstup funkcie **arima** alebo **estimate**), kladné celé číslo určujúce veľkosť predpovedného horizontu  $\text{numperiods}$  a počiatočné hodnoty vložené do modelu pre získanie predpovede  $Y_0$ .

Rozšítený Dickey-Fullerov test jednotkového koreňa počíta funkcia **adftest**, zápis:  $[\text{h}, \text{pValue}, \text{stat}, \text{cValue}, \text{reg}] = \text{adftest}(YY(:,i), \text{'lags'}, 12)$ . Vstupným parametrom je časový rad, na ktorom testujeme nulovú hypotézu, že časový rad obsahuje jednotkový koreň. Nepovinný parameter *'lags'* špecifikuje množstvo oneskorení, ktoré sa majú zahrnúť do modelu, defaultne je hodnota nastavená

## 4 KAPITOLA 4. EKONOMETRIA ČASOVÝCH RADOV V MATLABE

na 0. Výstupnými parametrami sú:  $h$  (rozhodnutie o (ne)zamietnutí nulovej hypotézy),  $pValue$  (p-hodnota testu),  $stat$  (hodnota testovej štatistiky),  $cValue$  (kritické hodnoty) a  $reg$  (štruktúra obsahujúca regresné štatistiky).

Alternatívnym testom stacionarity časového radu je KPSS (Kwiatkowski, Phillips, Schmidt a Shin) test, ktorý sa v Matlabe vykonáva pomocou funkcie **kpsstest**, zápis: `[h,pValue,stat,cValue,reg] = kpsstest(YY(:,i),'lags',1)`. Nulová hypotéza testu znie: Časový rad je trendovo stacionárny. Vstupné a výstupné parametre sú analogické ako v prípade **adftest**.

---

### Cvičenie 4.1 - Základná práca s časovými radmi

#### Zadanie:

- a) Vygenerujte nasledujúce časové rady s dĺžkou  $T = 150$ : ARMA(2,1) model, ARMA(2,1) model s deterministickým trendom, ARMA(2,1) model spolu so stochastickým trendom. Prvých 50 pozorovaní následne vyraďte. Uvažujte pri tom nasledovné hodnoty:  $a_0 = 0.1$ ,  $a_1 = 0.8$ ,  $a_2 = -0.2$ ,  $b_1 = 0.5$  a modely formulujte pomocou rovníc:

- ARMA(2,1) bez trendu:

$$y_t = a_0 + a_1 y_{t-1} + a_2 y_{t-2} + \epsilon_t + \beta_1 \epsilon_{t-1},$$

- ARMA(2,1) + deterministický trend:

$$y_t = a_0 + a_1 y_{t-1} + a_2 y_{t-2} + \epsilon_t + \beta_1 \epsilon_{t-1} + a_0 t,$$

- ARMA(2,1) + stochastický trend:

$$y_t = a_0 + a_1 y_{t-1} + a_2 y_{t-2} + \epsilon_t + \beta_1 \epsilon_{t-1} + \sum_{t=1}^T \epsilon_t.$$

- b) Vykreslite korelogramy (autokorelačné a parciálne autokorelačné funkcie) vygenerovaných časových radov.

#### Riešenie:

- a) Pre získanie rovnakých výsledkov využijeme voľbu generátora náhodných čísel pomocou príkazu **rng** a definujeme jednotlivé hodnoty potrebné pre vytvorenie zadaných časových radov.

```
% nastaveni generatoru nahodnych cisel
>> rng(123456)
% nastaveni parametru
>> T = 150; a0 = 0.1; a1 = 0.8; a2 = -0.2; b1 = 0.5;
```

Na vytvorenie časového radu budeme potrebovať vektor náhodných čísiel z normálneho rozdelenia, ktorý vygenerujeme pomocou funkcie **randn**. Následne pomocou určených rovníc pre jednotlivé ARMA(2,1) procesy vytvoríme požadované časové rady. Prvých 50 pozorovaní vyradíme.

```
Y = zeros(T,3); % tvorba matice nul o velikosti Tx3
eps = randn(T,1); % generovani vektoru cisel z normalniho
    rozdeleni o velikosti Tx1
eps2 = randn(T,1); % generovani vektoru cisel
    pro stochasticky trend
% simulace ARMA(2,1) procesu
for t = 3:T
    Y(t,1) = a0 + a1*Y(t-1,1) + a2*Y(t-2,1) + eps(t)
        + b1*eps(t-1);
    Y(t,2) = a0 + a1*Y(t-1,2) + a2*Y(t-2,2) + eps(t)
        + b1*eps(t-1)+t*a0;
    Y(t,3) = a0 + a1*Y(t-1,3) + a2*Y(t-2,3) + eps(t)
        + b1*eps(t-1)+sum(eps2(1:t));
end
YY = Y(51:T,:); % ulozeni vybrane casti vektoru Y
    do noveho vektoru YY
```

- b) Časové rady vykreslíme potom príkazov **subplot** a **plot**. Na vykreslenie korelogramov využijeme funkcie **autocorr** a **parcorr**.

```
%% Vykresleni casove rady
figure_1 = figure;
subplot(3,1,1)
plot(YY(:,1)) % vykresleni casove rady
title('ARMA(2,1) model')
subplot(3,1,2)
plot(YY(:,2)) % vykresleni casove rady
title('ARMA(2,1) model + deterministický trend')
subplot(3,1,3)
plot(YY(:,3)) % vykresleni casove rady
title('ARMA(2,1) model + stochastický trend')
%% Vykresleni korelogramu casovych rad
figure_2 = figure;
for i=1:3
    subplot(3,2,(2*i-1))
```

```

autocorr(YY(:,i)) % ACF
subplot(3,2,2*i)
parcorr(YY(:,i)) % PACF
end

```

Výstupom vyššie zmienených kódov sú obrázky 4.1 a 4.2. Obrázok 4.1 zachytáva priebeh nami vygenerovaných radov. Prvý graf obrázku 4.1 zachytáva časový rad, u ktorého nie je možné jednoznačne konštatovať, že obsahuje trendové chovanie, druhý graf naopak vykazuje rastúci trend a u tretieho grafu opäť nie je možné bez nejakého dodatočného testovania jednoznačne rozhodnúť o prítomnosti trendu. My samozrejme vďaka tomu, že sme si dáta vygenerovali vieme, aké zložky dané časové rady obsahujú.

Základnú identifikáciu uskutočníme pomocou autokorelačnej a parciálnej autokorelačnej funkcie. Z obrázku 4.2 prvého stĺpca je zrejmé, že dochádza k štatisticky významnej korelácii v jednotlivých časoch. Zaujímavé sú z tohto pohľadu grafy na druhom a treťom riadku prvého stĺpca, kde vidíme pomerne silnú a perzistentnú koreláciu v čase. Zodpovedá to našej informácii o tom, že tieto časové rady obsahujú deterministický či stochastický trend. Pomocou druhého stĺpca grafu na obrázku 4.2 je možné odhadnúť konkrétny ARMA(p,q) proces, ktorým sa daný časový rad riadi. Vzhľadom k priebehu korelácií v korelograme (korelácie sú štatisticky významné po niekoľkých prvých obdobiach a striedajú znamienka) by bolo vhodné k modelovaniu voliť model, ktorý obsahuje ako AR tak aj MA zložku.

---

## Cvičenie 4.2 - Základná práca s časovými radmi

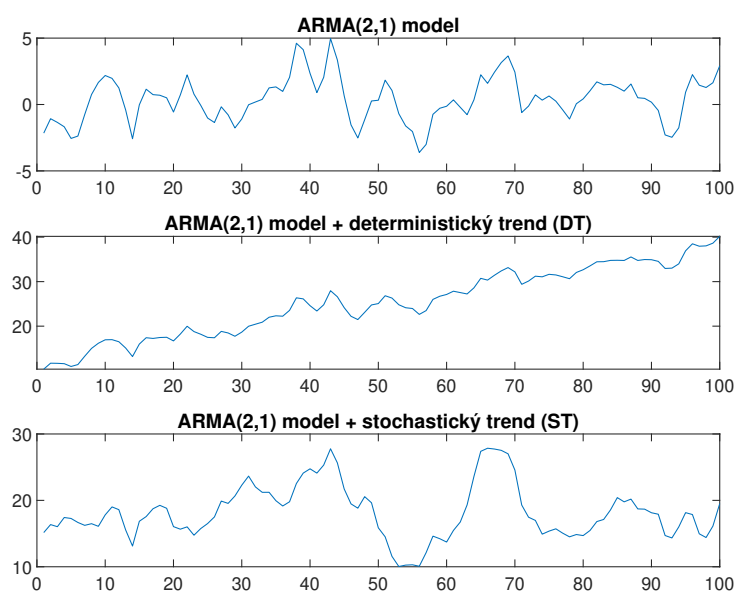
**Zadanie:** Využite dáta vygenerované v Cvičení 2.1 pre ARMA(2,1) model (bez trendov). Zvoľte nasledujúce tri špecifikácie modelov: AR(1), ARMA(2,1) a MA(1). Overte, že generované dáta skutočne zodpovedajú ARMA(2,1) procesu. K overeniu použite autokorelačnú funkciu (ACF) a výpočet informačných kritérií (AIC, BIC).

**Riešenie:** Jednotlivé špecifikácie vytvoríme pomocou funkcie **arima**. Nasledne ich odhadneme na dátach z Cvičenia 2.1 pomocou funkcie **estimate**.

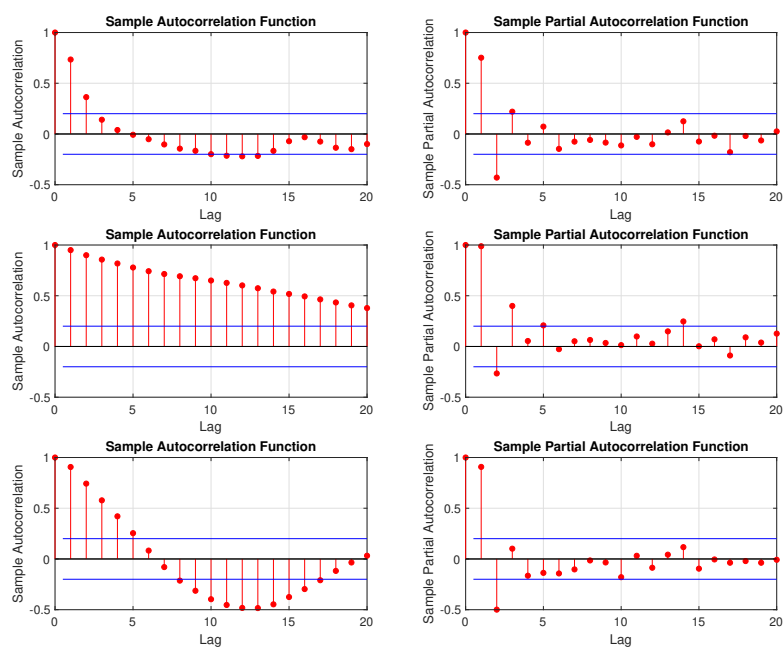
```

%identifikace ARMA procesu
>> Model_AR10 = arima(1,0,0); % vytvoreni ARMA(1,0) modelu

```



Obr. 4.1: Pribeh časových radov



Obr. 4.2: Grafy autokorelačnej a parciálnej autokorelačnej funkcie

## 8 KAPITOLA 4. EKONOMETRIA ČASOVÝCH RADOV V MATLABE

```
>> Model_AR21 = arima(2,0,1); % vytvoreni ARMA(2,1) modelu
>> Model_AR01 = arima(0,0,1); % vytvoreni ARMA(0,1) modelu

% odhad AR(1), ARMA(2,1) a MA(1) modelu na datech
>> [ARMA10_est,EPC10,logL10,info10] = estimate(Model_AR10,data);
>> [ARMA21_est,EPC21,logL21,info21] = estimate(Model_AR21,data);
>> [ARMA01_est,EPC01,logL01,info01] = estimate(Model_AR01,data);
```

Pomocou vykreslenia rezíduí z modelu a výpočtom informačných kritérií budeme schopný určiť, ktorá špecifikácia je pre jednotlivé modely najvhodnejšia. Na získanie rezíduí využijeme **infer**. Funkcia **autocorr** okrem vykreslenia ACF poskytuje aj číselný výpis, ktorý si vypíšeme do tabuľky pomocou príkazu **table**.

```
% ziskani rezidui z AR(1), ARMA(2,1) a MA(1) odhadu
ARMA_resid(:,1) = infer(ARMA10_est,data);
ARMA_resid(:,2) = infer(ARMA21_est,data);
ARMA_resid(:,3) = infer(ARMA01_est,data);

% Vykresleni korelogramu ziskanych rezidui
figure_3 = figure;
for i=1:3
    subplot(3,2,(2*i-1))
        autocorr(ARMA_resid(:,i)) % ACF
    subplot(3,2,2*i)
        parcorr(ARMA_resid(:,i)) % PACF
end
%% ciselny vypis ACF
for i=1:3
    [acf(:,i),lags,bounds]=autocorr(ARMA_resid(:,i))
end
ARMA10=acf(:,1)
ARMA21=acf(:,2)
ARMA01=acf(:,3)
ACF_funkce=table(lags,ARMA10,ARMA21,ARMA01)
```

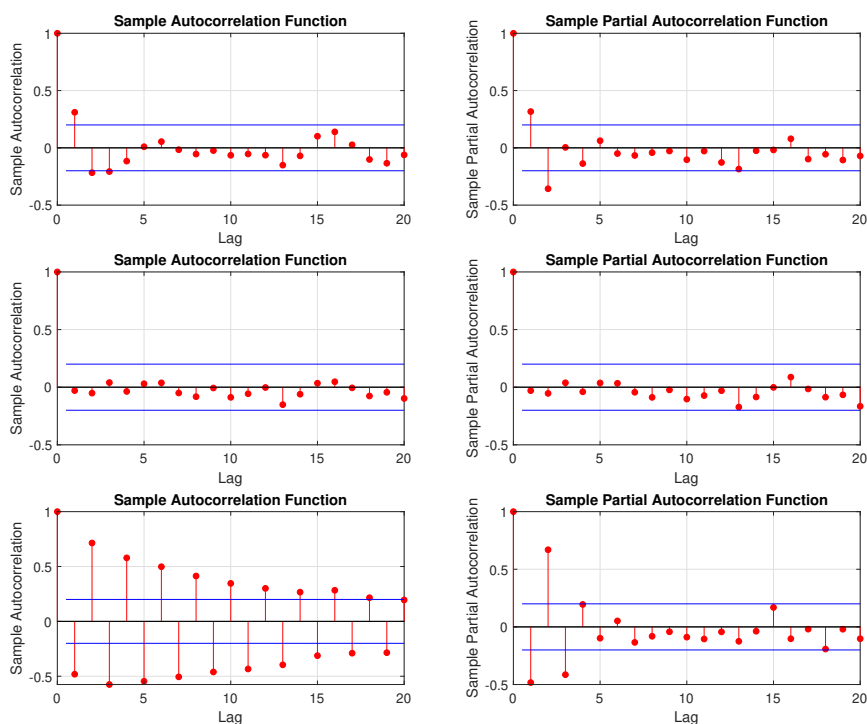
K rozhodnutiu, či sme správne identifikovali model nám môže opäť pomôcť autokorelačná a parciálna autokorelačná funkcia. V prípade, že sme správne trafili  $p$  a  $q$  stupne ARMA( $p,q$ ) modelu, by autokorelácia rezíduí z týchto modelov mala byť štatisticky nevýznamná. V zbytkovej zložke teda nie je viditeľne zachytené žiadne systematické správanie, ktoré by malo byť zohľadnené



modelom. Toto spĺňa iba model ARMA(2,1), viď obrázok 4.3 druhý riadok grafu. Samotné veľkosti autokorelácií sú potom vypísané na obrazovku do tabuľky ACF\_funkce. Výstup je pre prvé štyri oneskorenia zachytený v tabuľke 4.1.

lags	ARMA(1,0)	ARMA(2,1)	ARMA(0,1)
0	1	1	1
1	0,31023	-0,029434	-0,48151
2	-0,21704	-0,051567	0,71428
3	-0,20683	0,040115	-0,57544
4	-0,11594	-0,036621	0,57878
⋮	⋮	⋮	⋮

Tabuľka 4.1: Parciálna autokorelačná funkcia pre rezíduá



Obr. 4.3: Grafy autokorelačnej a parciálnej autokorelačnej funkcie pre rezíduá

Pre výpis Akaikeovho a Bayesovho informačného kritéria slúži funkcia **aicbic**. Výstup je následne zachytený v tabuľke 4.2. Model, ktorý popisuje správanie

dát najlepšie, bude mať najnižšiu hodnotu informačných kritérií.

```
%% Vypocet informacnich kriterii
ic=zeros(3,2);
[ic(1,1),ic(1,2)] = aicbic(logL10,Model_AR10.Q+...
Model_AR10.P,length(data));
[ic(2,1),ic(2,2)] = aicbic(logL21,Model_AR21.Q+...
Model_AR21.P,length(data));
[ic(3,1),ic(3,2)] = aicbic(logL01,Model_AR01.Q+...
Model_AR01.P,length(data));
AIC=ic(:,1);
BIC=ic(:,2);
model={'ARMA10','ARMA21','ARMA01'}';
table(model,AIC,BIC)
```

Model	AIC	BIC
ARMA(1,0)	304,53	307,13
ARMA(2,1)	284,04	291,86
ARMA(0,1)	385,67	388,28

Tabuľka 4.2: Informačné kritériá

### Cvičenie 4.3 - Predikcie

**Zadanie:** Využite vzorku dát zo stacionárneho časového radu z Cvičenia 2.1 k predikovaníu.

1. Využite prvých 48 pozorovaní k odhadu ARMA(1,0), ARMA(2,1) a ARMA(0,1) modelu. Získajte rekurzívne jednokrokové predikcie a jednokrokové chyby predikcie pre každý z týchto modelov. Postupne teda zväčšujte obdobie pre odhad. Označte si jednokrokové predikcie z modelu ARMA(1,0) ako  $\{f_{1t}\}$ , z modelu ARMA(2,1) ako  $\{f_{2t}\}$  a z modelu ARMA(0,1) ako  $\{f_{3t}\}$ . Výsledky porovnajte (pomocou vykreslenia) a taktiež zahrňte naivné predpovede, kedy ako predikovaná hodnota bude uvažovaná hodnota z predchádzajúceho obdobia.
2. Porovnajte kvalitu predpovedí na základe ukazovateľa strednej štvorcovej chyby predikcie, MSPE, prípadne odmocniny zo strednej štvorcovej chyby predikcie, RMSPE.

**Riešenie:**

```

% predvytvoreni vektoru pro predikovane hodnoty
N=length(data); %delka casove rady
L=48; %zakladni sirka rozsirujiciho se okna
H=N-L; %počet rozsireni zakladniho okna o 1 obdobi
f11 = zeros(H,1); f21 = zeros(H,1); f31 = zeros(H,1);
% rucne vytvoreny ukazatel prubehu cyklu
fprintf('\nPrubeh odhadu\n')
fprintf('0%%      50%%      100%%\n')
fprintf('|')
% pomocna promenna pro graficke znazorneni prubehu
pom_step = 0.05; %0.05 = posun po 5%
pom_graph = pom_step;

for T = 1:H
    [f_Model_AR10,~] = estimate(Model_AR10,data(3:L-1+T),...
    'Y0', data(2),'Display','off');
    [f11(T),~] = forecast(f_Model_AR10,1,'Y0',data(3:L-1+T));
    [f_Model_AR21,~] = estimate(Model_AR21,data(3:L-1+T),...
    'Y0', data(1:2),'Display','off');
    [f21(T),~] = forecast(f_Model_AR21,1,'Y0',data(3:L-1+T));
    [f_Model_AR01,~] = estimate(Model_AR01,data(3:L-1+T),...
    'Y0', data(2),'Display','off');
    [f31(T),~] = forecast(f_Model_AR01,1,'Y0',data(3:L-1+T));
    % graficke zobrazeni prubehu po 5 %
    if T/H >= pom_graph
        fprintf('|')
        pom_graph = pom_graph + pom_step;
    end
end

%% vykresleni
fN1 = zeros(N,1); %vytvoreni vektoru pro naivni predikce
fN1(2:end) = data(1:end-1); %naivni predikce y_pred=y(-1)
figure3 = figure;
plot(1:N,data,'color','k','LineWidth',1.15)
hold on
plot(L+1:N,f11,'color','b','LineWidth',1.15)
plot(L+1:N,f21,'color','r','LineWidth',1.15)
plot(L+1:N,f31,'color','c','LineWidth',1.15)

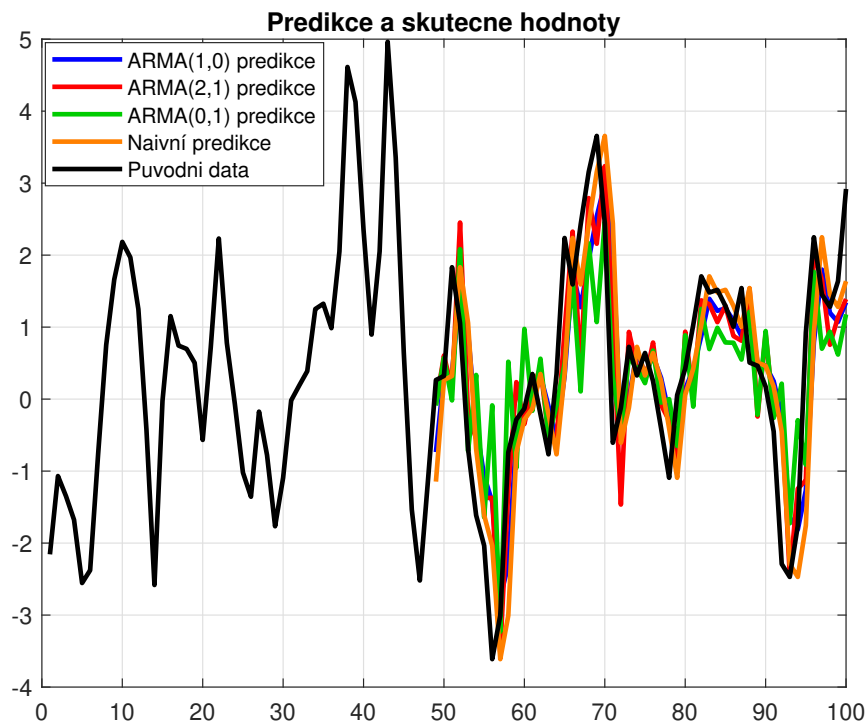
```

```

plot(L+1:N,fN1(L+1:N),'color','m','LineWidth',1.15)
grid on
% prikaz, který přida legendu do grafu
legend('Puvodni data','ARMA(1,0) predikce',...
'ARMA(2,1) predikce', 'ARMA(0,1) predikce','Naivní predikce')
title('Predikce a skutečne hodnoty')

```

Skutočné hodnoty a jednokrokové predikcie jednotlivých modelov sú graficky zachytené na obrázku 4.4. Z grafu je len veľmi ťažko možné identifikovať, či je nejaký významný rozdiel medzi jednotlivými jednokrokovými predikciami.



Obr. 4.4: Predikcie a skutočné hodnoty

```

%% chyby predikce a kvalita predpovedi
e11 = f11 - data(L+1:end);
e21 = f21 - data(L+1:end);
e31 = f31 - data(L+1:end);
eN1 = fN1(L+1:N) - data(L+1:end);

```

```

MSPE11 = mean(e11.^2);
MSPE21 = mean(e21.^2);
MSPE31 = mean(e31.^2);
MSPEN1 = mean(eN1.^2);
MSPE=round([MSPE11;MSPE21;MSPE31;MSPEN1],4);

RMSPE11 = sqrt(mean(e11.^2));
RMSPE21 = sqrt(mean(e21.^2));
RMSPE31 = sqrt(mean(e31.^2));
RMSPEN1 = sqrt(mean(eN1.^2));
RMSPE=round([RMSPE11;RMSPE21;RMSPE31;RMSPEN1],4);

model_predikce={'ARMA10','ARMA21','ARMA01','NAIVNI'}';
table(model_predikce,MSPE,RMSPE)

```

Model predikcie	MSPE	RMSPE
ARMA(1,0)	1,0162	1,0081
ARMA(2,1)	0,8883	0,9425
ARMA(0,1)	1,2992	1,1398
Naivná	1,1034	1,0504

Tabuľka 4.3: Hodnoty MSPE a RMSPE

Tabuľka 4.3 zachytáva hodnoty kritérií MSPE a RMSPE, ktoré nám vracia Matlab. Najnižšia hodnota týchto kritérií poukazuje na najlepšie predikujúci model. V tomto prípade nám ako najlepšie predikujúci model vychádza model ARMA(2,1), a čo je podstatnejšie, predikuje lepšie ako naivná predikcia. V prípade, že by najlepšou možnou predikciou bola naivná predikcia, potom sme buď netrafili skutočný datagenerujúci proces, alebo majú dáta taký charakter vývoja, ktorý nie je možné modelovo zachytiť.

## Cvičenie 4.4 - Test jednotkového koreňa

**Zadanie:** Využite uložené dátové súbory ARMA(2,1) modelov z Cvičenia 2.1. Rady otestujte na prítomnosť jednotkového koreňa pomocou príslušných funkcií (**adftest**, **kpsstest**). Výsledky zapíšte do tabuľky.

**Riešenie:** Jednotlivé testy uskutočníme pomocou funkcií **adftest** a **kpsstest**. Na vytvorenie prehľadnej tabuľky v Matlabe využijeme jednotlivé rozhodnutia o (ne)zamietnutí testovanej hypotézy a p-hodnoty testu, pričom do prvého stĺpca tabuľky vypíšeme názvy jednotlivých procesov.

```

Proces={'ARMA(2,1)'; 'ARMA(2,1)+ DT'; 'ARMA(2,1)+ ST'};

for i=1:3
    [h,pValue,stat,cValue,reg] = adfctest(YY(:,i),'lags',12);
    ADF_NotUnitRoot(i,1)=h;
    ADF_pValue(i,1)=pValue;

    [h,pValue,stat,cValue,reg] = kpsstest(YY(:,i),'lags',1);
    KPSS_UnitRoot(i,1)=h;
    KPSS_pValue(i,1) = pValue;
end
table(Proces,ADP_NotUnitRoot,ADP_pValue,KPSS_UnitRoot,
      KPSS_pValue)

```

Proces	ADF	ADF pValue	KPSS	KPSS pValue
ARMA(2,1)	true	0,012949	false	0,1
ARMA(2,1) + DT	false	0,9972	false	0,1
ARMA(2,1) + ST	false	0,63794	true	0,026991

ADF: true = stacionárny; false = nestacionárny

KPSS: true = nestacionárny; false = trendovo stacionárny

Tabuľka 4.4: Výsledky testov jednotkového koreňa

V tabuľke 4.4 vidíme jednotlivé rozhodnutia testov jednotkového koreňa. Rozšírený ADF test vyhodnotil, že jedine ARMA(2,1) neobsahuje jednotkový koreň. KPSS test narozdiel od ADF testu posudzuje aj trendovú stacionaritu časového radu a preto zamietá prítomnosť jednotkového koreňa v prvých dvoch ARMA procesoch.

## Cvičenie 4.5 - Detrendovanie

**Zadanie:** Pracujte opäť s uloženými časovými radmi z Cvičenia 2.1.

- Časový rad ARMA(2,1) s deterministickým trendom sa pokúste stacionarizovať priamkou (polynómom prvého stupňa) a parabolou (polynómom druhého stupňa). Výsledné cyklické zložky vykreslite spolu s pôvodným časovým radom a trendovými zložkami.
- Časový rad ARMA(2,1) so stochastickým trendom sa pokúste stacionarizovať pomocou diferencií. Výsledky opäť vykreslite.

- c) Do jedného obrázku vykreslite skutočnú cyklickú zložku a vami získané cyklické zložky pomocou stacionarizácie priamkou, parabolou a diferenciálami.

### Riešenie:

- a) V prvej úlohe dekomponujeme časový rad na dočasnú a trvalú zložku dvoma spôsobmi, pomocou polynómov prvého a druhého stupňa. Dekompozíciu uskutočníme v cykle, pričom jednotlivé odhadnuté hodnoty budeme postupne ukladať do predpripraveného priestoru.

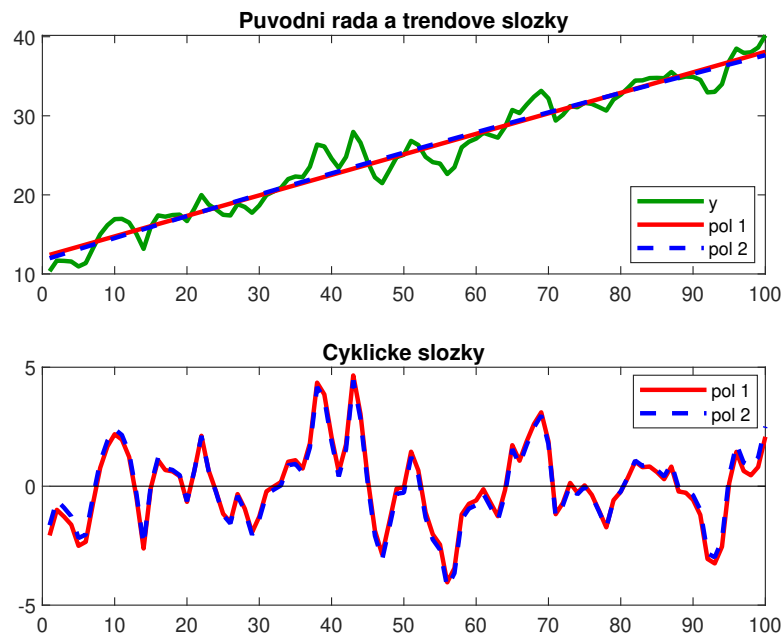
```
%A)
y=YY(:,2);
t = (1:100)'; % vektor indexujici cas
T = length(YY); % pocet pozorovani
X = ones(T,1); % matice regresoru, nyní pouze konstanta
for p = 1:2
    X(:,1+p) = t.^p;
    % time = logicky cas, umocneny podle cyklu,
    % ve kterem se nachazime (radu polynomu)
    % postupne rozsirujeme seznam regresoru
    % navysujeme rad polynomu
    model = fitlm(X,y); % odhad trendu pomoci regrese
    % reziduuum jako cyklicka slozka
    yc_pol(:,p) = model.Residuals.Raw;
    % trend = vyrovnané hodnoty
    yp_pol(:,p) = model.Fitted;
end
```

Pôvodný rad vykreslíme spolu s trendovými zložkami do jedného obrázku, do druhého vykreslíme odhadnuté cyklické zložky získané polynomiálnym detrendovaním.

```
% Vykresleni puvodni rady a detrendovanych rad
figure_2 = figure;
subplot(2,1,1)
plot(y,'g')
hold on
plot(yp_pol(:,1),'r')
plot(yp_pol(:,2),'b')
legend('y','pol 1','pol 2')
title('Puvodni rada a trendove slozky')
```

```
subplot(2,1,2)
plot(yc_pol(:,1),'r')
hold on
plot(yc_pol(:,2),'b')
plot(1:1:100,zeros(100,1),'k')
legend('pol 1','pol 2')
title('Cyklické složky')
```

Na vrchnej časti obrázku 4.5 môžeme pozorovať preloženie pôvodnej rady jej trendovou zložkou odhadnutou priamkou a parabolou. Na spodnej časti následne vidíme odhadnuté cyklické zložky.



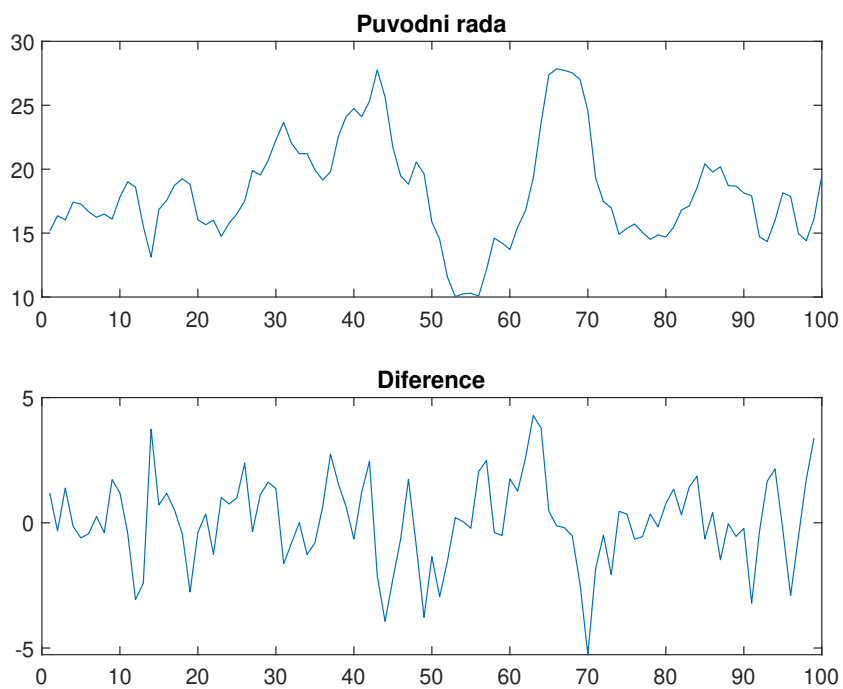
Obr. 4.5: Výsledné obrázky detrendovania pomocou polynomiálnych trendov

- b) V prípade detrendovania pomocou diferencií vytvoríme pomocou príkazu **diff** príslušné diferencie. Tie následne vykreslíme spolu s pôvodným časovým radom.

```
%B)
y2=YY(:,3);
y2_diff=diff(y2);
```



```
figure
subplot(2,1,1)
plot(y2)
subplot(2,1,2)
plot(y2_diff)
```



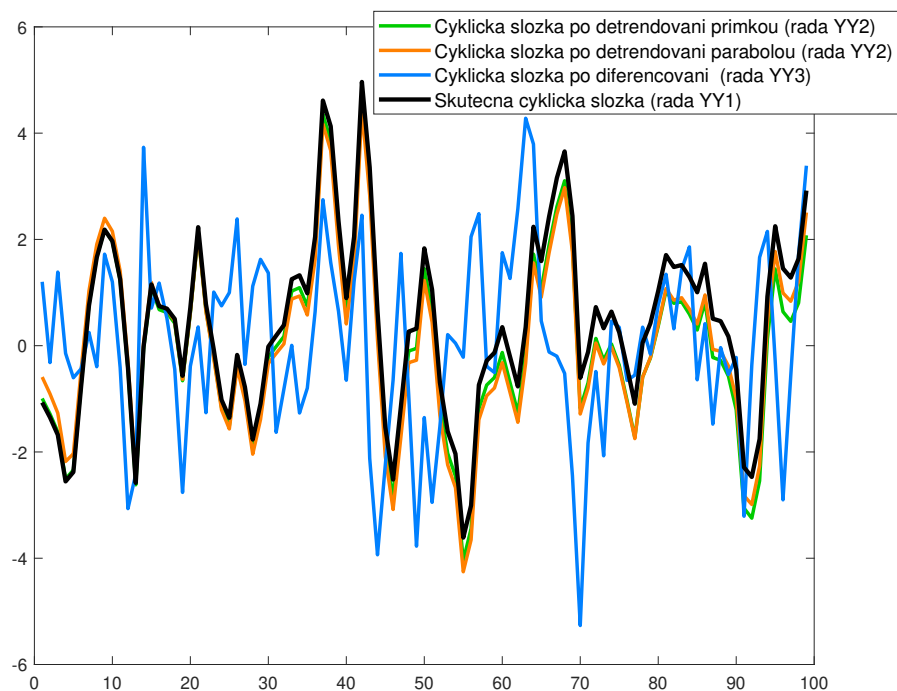
Obr. 4.6: Pôvodný časový rad a jeho diferencie

- c) Získané cyklické zložky vykreslíme do jedného obrázku využitím voľby **hold on**. Pre prehľadnosť pridáme do obrázka legendu.

```
%C)
figure
plot(YY(2:end,1),'k','LineWidth',2)
hold on
plot(yc_pol(2:end,1),'r')
plot(yc_pol(2:end,2),'b')
plot(y2_diff)
legend('Skutecna cyklicka slozka (rada YY1)',...
'Cyklicka slozka po detrendovani primkou (rada YY2)',...
'Cyklicka slozka po detrendovani parabolou (rada YY2)',...)
```

```
'Cyklicka složka po diferencování (řada YY3)'
```

Obrázok 4.7 zhŕňa všetky získané odhady pomocou polynomiálnych trendov aj pomocou diferencií, ktoré sú vykreslené spolu s pôvodným časovým radom.



Obr. 4.7: Získané cyklické složky skúmaného časového radu

## Ďalšie funkcie

Nasledujúci zoznam sumarizuje ďalšie užitočné funkcie, ktoré pri práci s časovými radmi v Matlabe možno využiť.

- **crosscorr** - príkaz počíta krížové korelácie pre časové rady  $y_1$  a  $y_2$ . Zápis: `[xcf,lags,bounds] = crosscorr(y1,y2)`, kde `xcf` je výberová krížová korelačná funkcia, `lags` je vektor čísiel oneskorenia a `bounds` je dvojprvkový vektor hornej a dolnej hranice.
- **cusumtest** - funkcia pre CUSUM test štrukturálneho zlomu, zápis: `[h,H,Stat,W,B,sumPlots] = cusumtest(X,y,'Plot','on','Test','cusumsq')`, kde  $X$  je matica prediktorov a  $y$  je odozva. Voliteľný parameter 'Plot' značí, či sa majú výsledky vykresliť alebo nie (defaultne je nastavené 'off') a parameter 'Test' značí typ testu (defaultne 'cusum', inak napríklad 'cusumsq'). Výstupnými argumentmi sú:  $h$  (rozhodnutie o (ne)zamietnutí nulovej hypotézy),  $H$  (vektor, kde hodnota 1 značí prekročenie kumulatívnych súm mimo interval spoľahlivosti, 0 inak),  $Stat$  (hodnota testovacej štatistiky),  $W$  (rekurzívne rezíduá),  $B$  (odhady rekurzívnych koeficientov), `sumPlots` (popis vykreslených grafických objektov v prípade 'Plot','on').
- **gctest** - funkcia spočíta test Grangerovskej kauzality pre VAR model, zápis: `gctest(m,'Test','f')`. Vstupným parametrom je VAR model  $m$ , parameter 'Test' nastavuje druh testovacej štatistiky, implicitne je nastavená  $\chi^2$  testová štatistika.
- **chowtest** - funkcia spočíta Chowov test štrukturálneho zlomu, zápis: `[h,pValue,stat,cValue] = chowtest(X,y,c)`. Vstupnými parametrami je matica prediktorov  $X$ ,  $y$  je odozva a  $c$  je poradie pozorovania s potenciálnym bodom zlomu. Výstupnými argumentmi sú:  $h$  (rozhodnutie o (ne)zamietnutí nulovej hypotézy),  $pValue$  (p-hodnota testu),  $stat$  (hodnota testovacej štatistiky) a  $cValue$  (vektor kritických hodnôt).
- **lbqtest** - príkaz na výkonanie Ljung-Box Q-testu autokorelácie rezíduí, zápis: `[h,pValue,stat,cValue] = lbqtest(e,'Lags',4,'Dof',2)`. Vstupnými argumentmi sú rezíduá  $e$ , počet oneskorení 'Lags' ako celé číslo, a počet stupňov voľnosti 'DoF'. Výstupnými argumentmi sú:  $h$  (rozhodnutie o (ne)zamietnutí nulovej hypotézy),  $pValue$  (p-hodnota testu),  $stat$  (hodnota testovacej štatistiky) a  $cValue$  (vektor kritických hodnôt).

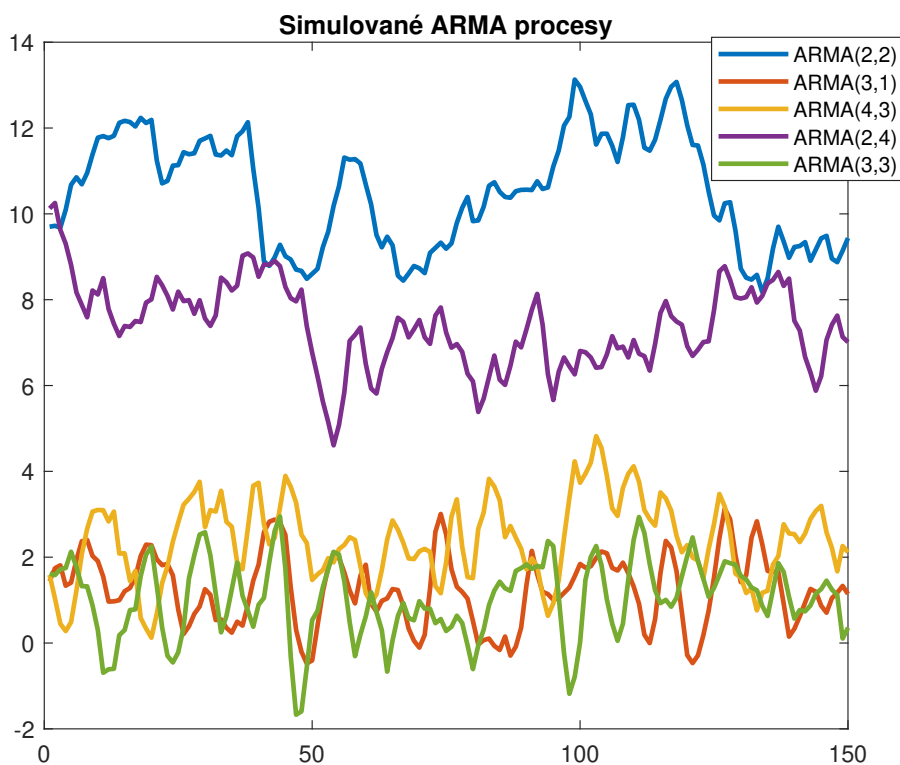
## Neriešené príklady

1. Pracujte s funkciami **arima** a **simulate**, pričom sa pokúste vytvoriť identický obrázok ako je obrázok 4.8. Voľte hodnoty:

AR koeficienty	MA koeficienty
$a_1 = 0,7$	$\beta_1 = 0,8$
$a_2 = 0,25$	$\beta_2 = 0,4$
$a_3 = -0,4$	$\beta_3 = -0,3$
$a_4 = 0,2$	$\beta_4 = 0,1$

Tabuľka 4.5: AR a MA koeficienty

Konštantu nastavte na hodnotu  $c = 0,5$  a rozptyl  $\sigma^2 = 0,1$ . Pre zhodnosť výsledkov nastavte generátor náhodných čísiel na **rng(1234)**. (*Hint*: Skúste si detailne naštudovať nápovedu funkcie **arima**.)



Obr. 4.8: Výsledný obrázok

2. Pracujte s funkciami **arima** a **simulate**. Skúste si vygenerovať časové rady nasledujúcich procesov:

- ARMA(1,0) s koeficientom  $a_1 = 0,9$
- ARMA(1,0) s koeficientom  $a_1 = 0,1$
- ARMA(1,0) s koeficientom  $a_1 = -0,5$
- ARMA(0,1) s koeficientom  $\beta_1 = 0,5$
- ARMA(0,1) s koeficientom  $\beta_1 = -0,5$
- ARMA(1,1) s koeficientami  $a_1 = 0,9$  a  $\beta_1 = -0,5$

Zamyslite sa, aký vplyv majú jednotlivé koeficienty na dynamiku časového radu.

3. Pracujte s vygenerovaným časovým radom ARMA(2,2) procesu, uloženým v súbore *arma22.mat*. Pomocou vhodnej funkcie sa pokúste odhadnúť dvojicu AR ( $a_1, a_2$ ) a dvojicu MA ( $\beta_1, \beta_2$ ) koeficientov spomenutého ARMA(2,2) procesu.
4. Vlastík si v piatok pre svoje účely vygeneroval časový rad riadiaci sa ARMA(p,q) modelom, pričom zvolil konkrétne hodnoty  $p$  a  $q$ . Bohužiaľ, cez víkend tieto konkrétne hodnoty zabudol. Jediné na čo si pamätal bolo to, že hodnoty  $p$  a  $q$  neboli menšie ako 1 a väčšie ako 4. Využite dáta z vygenerovaného časového radu (*data\_simul.mat*) a pokúste sa pomocou funkcie **aicbic** príslušné hodnoty  $p$  a  $q$  nájsť.
5. Inflácia je jedným z ukazovateľov, ktorý sa snažia rôzne úrady predikovať. Máme k dispozícii dáta o inflácii hypotetickej krajiny Jakubíkovo (súbor *inflace.mat*) a jej predpovede zo 4 úplne nezávislých inštitúcií (predikcie p1-p4). Pokúste sa určiť, ktorá z inštitúcií predikuje infláciu najlepšie.
6. Využite vzorku dát zo stacionárneho časového radu, ktorý je obsahom súboru *predikce.mat*. Využite prvých 100 pozorovaní k odhadu ARMA(4,2) modelu. Následne získajte rekurzívne jednokrokové predikcie na postupne sa rozširujúcom okne. Predikcie a skutočný časový rad vykreslite do jedného obrázku.
7. V súbore *DATA\_98Q1\_2016Q4.mat* sú uložené dáta o českej ekonomike v rozmedzí od prvého štvrťroku 1998 do posledného štvrťroku 2016. Konkrétne sa jedná o premenné: HDP ( $Y$ ), Spotreba domácností ( $C$ ) a Investície (Tvorba hrubého kapitálu, celkom) ( $I$ ).
  - (a) Časové rady vykreslite a otestujte ich na prítomnosť jednotkového koreňa pomocou ADF a KPSS testov.

- (b) V prípade, že u radov identifikujete prítomnosť jednotkového koreňa, rady transformujte pomocou logaritmických diferencií a overte, či sú tieto nové časové rady stacionárne (neobsahujú jednotkový koreň).
- 8. Využite dáta pre HDP z príkladu 7. Tieto dáta zlogaritmujte a očistite od trendu pomocou detrendovania. Trend modelujte pomocou polynómu, zväžte polynóm až štvrtého stupňa. Získané medzery vykreslite. Uložte medzeru získanú detrendovaním polynómom tretieho stupňa ako *mezera.mat*. Zhodu výsledku pre kontrolu porovnajte s priloženým dátovým setom *mezera\_pol3.mat*.
- 9. Načítajte dáta *mezera\_pol3.mat* obsahujúce medzeru HDP, kedy potenciálny produkt bol odstránený ako trend pomocou polynómu 3. stupňa.
  - (a) Overte či tento časový rad neobsahuje jednotkový koreň.
  - (b) Pre danú medzeru vykreslite korelogram. Zamyslite sa, či by se dal na tento rad aplikovať vhodný ARMA proces.
- 10. Načítajte dáta *mezera\_pol3.mat* obsahujúce medzeru HDP, kedy potenciálny produkt bol odstránený ako trend pomocou polynómu 3. stupňa.
  - (a) Časový rad modelujte pomocou ARMA(1,0) procesu. Uložte rezíduá získané z tohoto modelu.
  - (b) Vykreslite korelogram rezíduí. Zamyslite sa, či je zvolený ARMA model vhodný, prípadne či by nebolo vhodné na zvolený rad aplikovať iný ARMA proces.
  - (c) Zvážte voľbu ARMA(1,0) modelu so zachytením sezónneho chovania (voľba `Model_AR10_sezona = arima('ARLags',1, 'Seasonality',4);`. Zamyslite sa, či tento model dokázal vyriešiť problém s korelovanými rezíduami v 4., 8. a 12. oneskorení.