```
% Load the exercise 2, part 1
PC2_exe2_part1_gaps;
close all;
clc;

%==========================================================================
% ------------------------  ADL model ----------------------------------
%==========================================================================
%% Part 2a) Autocorrelation test for the residuals
%  -------------------- Estimation ------------------------
% Model 2:
%      GDPgr=a+\sum^M_{i=0} b_i Tax_{t-i} + \sum^N_{j=1} c_j GDPgr_{t-j} + u_t
% In matrix form
%                          Y = X B + u
% X=[1 Xlags Ylags];
% where:
% Xlags = [Tax_t Tax_{t-1} Tax_{t-2} ... Tax_{t-M}], with M=12;
% Ylags = [GDPgr_{t-1} GDPgr_{t-2} ... GDPgr_{t-N}], with N=11;

% ===== Construct the Xlags and Ylags: use the function lagmatrix
M = 12;                              % lags number of Tax shock
N = 11;                              % lags number of GDP growth

Xlmat = lagmatrix(Tax,0:M);                        % lag matrix of tax shock
Ylmat = lagmatrix(GDPgr,1:N);                        % lag matrix of GDP growth

% ===== construct X matrix in model 2.
Xlags = Xlmat(M+1:end,:);
Ylags = Ylmat(M+1:end,:);
XM2   = [Xlags, Ylags];
% ===== use function fitlm to estimate model 2
ADLmdl = fitlm(XM2,Y);
disp(ADLmdl)
```

```
Linear regression model:
    y ~ 1 + x1 + x2 + x3 + x4 + x5 + x6 + x7 + x8 + x9 + x10 + x11 + x12 + x13 + x14 + x15 + x16 + x17 + x18 + x19 +

Estimated Coefficients:
```

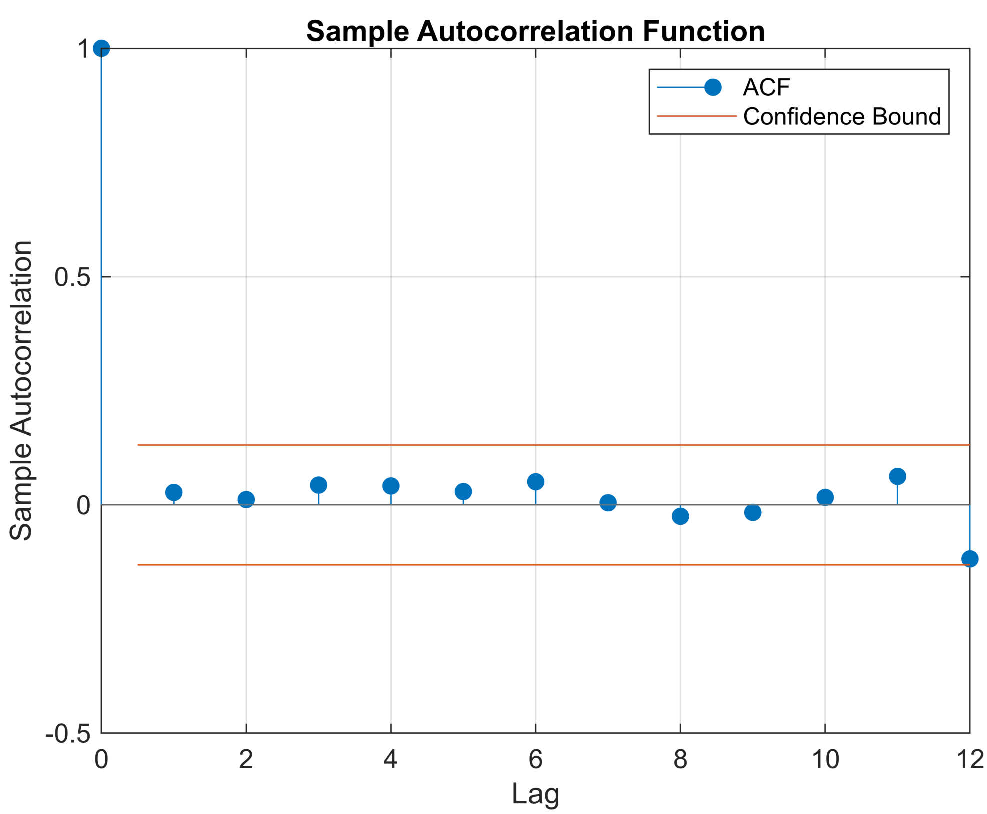
	Estimate	SE	tStat	pValue
(Intercept)	0.65642	0.1555	4.2213	3.6361e-05
x1	-0.043802	0.28077	-0.156	0.87618
x2	-0.28499	0.27848	-1.0234	0.30732
x3	0.27523	0.27852	0.98821	0.3242
x4	-0.56167	0.27921	-2.0117	0.045551
x5	-0.30514	0.28263	-1.0796	0.28156
x6	-0.10314	0.28313	-0.36428	0.71602
x7	-0.25971	0.28302	-0.91764	0.35987
x8	-0.75364	0.24244	-3.1086	0.0021438
x9	-0.12748	0.24826	-0.51347	0.60817
x10	-0.10326	0.2477	-0.41687	0.6772
x11	-0.071477	0.24771	-0.28854	0.77322
x12	0.33431	0.24808	1.3476	0.17926

x13	-0.033934	0.25225	-0.13452	0.89312
x14	0.285	0.068834	4.1405	5.0455e-05
x15	0.10872	0.070949	1.5324	0.12696
x16	-0.10755	0.071079	-1.5131	0.13177
x17	-0.056701	0.070297	-0.80659	0.42083
x18	-0.080028	0.070615	-1.1333	0.2584
x19	0.028451	0.069554	0.40905	0.68293
x20	-0.057496	0.069161	-0.83134	0.40674
x21	-0.017755	0.069036	-0.25718	0.7973
x22	0.043875	0.068857	0.63719	0.52471
x23	0.040344	0.068388	0.58993	0.55588
x24	-0.018179	0.065934	-0.27572	0.78304

```
Number of observations: 232, Error degrees of freedom: 207
Root Mean Squared Error: 0.898
R-squared: 0.232,  Adjusted R-Squared: 0.143
F-statistic vs. constant model: 2.61, p-value = 0.00014
```

```
% store estimated values
BhatM2      = ADLmdl.Coefficients.Estimate;              % estimators
resM2       = ADLmdl.Residuals.Raw;              % residuals
covBhatM2   = ADLmdl.CoefficientCovariance;              % estimated variance of
estimators

% -------------------- Residual diagnostic -----------------
% ===== plot autocorrelogram: autocorr
figure
autocorr(resM2,12)
ylim([-0.5 1])
```



```
% ===== Ljung-Box Q-test for residual autocorrelation: lbqtest
[~,pval_lbq] = lbqtest(resM2,"Lags",1:12);
disp(pval_lbq)
```

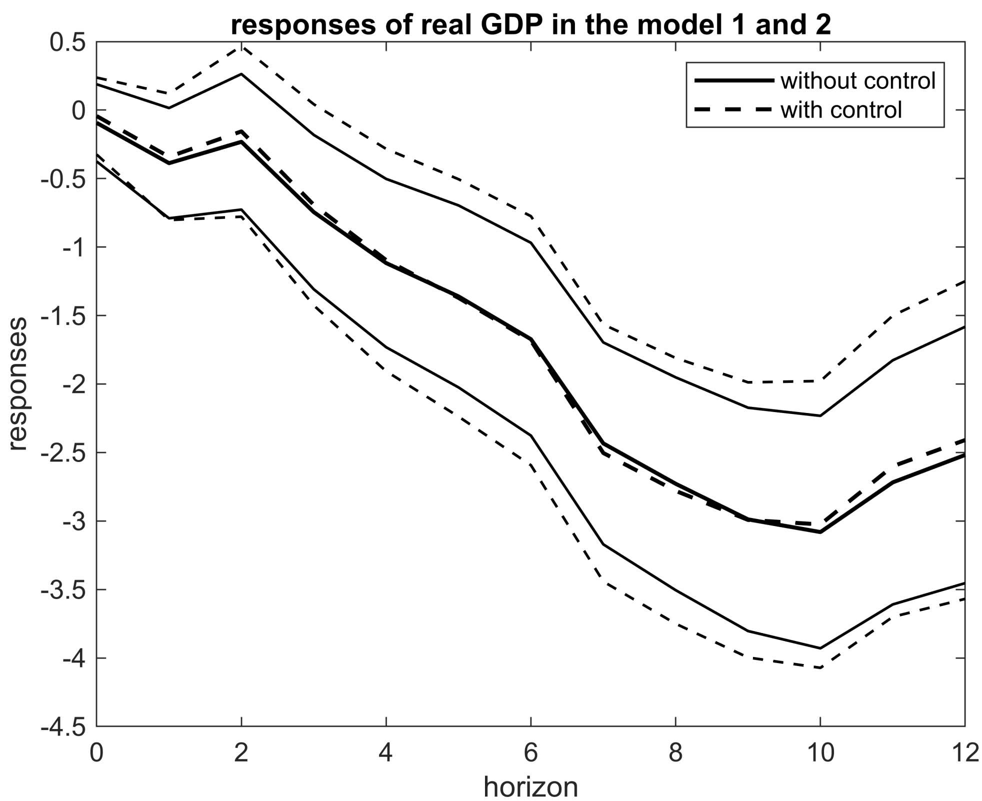
```
    0.6750    0.9009    0.8831    0.8991    0.9375    0.9291    0.9652    0.9794    0.9895    0.9947    0.9887    0
```

```
%% Part 2b) Calculate the (cumulated) impulse responses
% --------------------- (cumulated) impulse responses ----------------
% B = [a,b0,b1,b2,..b12,c1,c2,..., c11]';
bhat_M2 = BhatM2(2:M+2,1);    % [b0,b1,b2,...,b12]'
chat_M2 = [BhatM2(M+3:end);0]; % [c1,c2,...c12]', c12 = 0;
del_i = zeros(M+1,1);
del_i(1,1) = bhat_M2(1,1); % b0
Del = del_i(1,1);
for i=1:12
    del_i(i+1,1) = bhat_M2(i+1,1) + chat_M2(1:i,1)'*Del;
    Del = [del_i(i+1,1);Del];
end
respM2 = cumsum(del_i);
% ------------------- BOOTSTRAP -----------------------------------
% Using boostrap to calculate the estimated variance of the responses.
% ===== boostrap
ndraws=10000;
P=chol(covBhatM2,'lower');
gamM2_bstr=zeros(ndraws,M+1);
gamM2_2bstr=zeros(ndraws,M+1);
for j=1:ndraws
    Bhat_bstr=BhatM2+P*mvnrnd(0,1,M+N+2); % draw beta from the normal distribution
    bhat=Bhat_bstr(2:M+2);
    ahat=[Bhat_bstr(M+3:end); 0];
    Del=bhat(1);
    del_i=zeros(M+1,1);
    del_i(1)=Del(1);
    for i=1:12
        del_i(i+1,1)=bhat(i+1)+ahat(1:i)'*Del;
        Del=[del_i(i+1); Del];
    end
    bsum=cumsum(del_i);
    gamM2_bstr(j,:)=bsum';
end
IRFmean_M2= mean(gamM2_bstr,1)';
IRFse_M2 = std(gamM2_bstr,1)';

% 68\% confidence interval (1 standard deviation)
respM2_lb=respM2+tinv(0.16,T).*IRFse_M2;
respM2_ub=respM2+tinv(0.84,T)*IRFse_M2;

%------------------- END BOOSTRAP -----------------------------
```

```
% Replicate figure 5
figure
% plot the responses from the benchmark
plot(hor,resp,'-k','LineWidth',1.5); hold on
% plot the responses from the regression 2
plot(hor,respM2,'--k','LineWidth',1.5); hold on
% plot the upper and lower bounds
plot(hor,resp_lb,'-k','LineWidth',1); hold on
plot(hor,resp_ub,'-k','LineWidth',1); hold on
plot(hor,respM2_lb,'--k','LineWidth',1); hold on
plot(hor,respM2_ub,'--k','LineWidth',1); hold on
legend('without control','with control');
xlabel('horizon');
ylabel('responses');
title('responses of real GDP in the model 1 and 2');
```




```
%% Part 2c) Information criteria (IC)
% ADL model:
% Y_t=a+b0 X_t+b1 X_{t-1}+...+bp X_{t-p}+c1 Y_{t-1}+...+cq Y_{t-q}+u_t
lagmax     = 10; % maximum lag number in the model (p=q)
Ynew       = GDPgr(2:end); % because of missing values of the first obs
Xnew       = Tax(2:end);
AIC        = zeros(lagmax,1);
AICc       = zeros(lagmax,1);
BIC        = zeros(lagmax,1);
HQ         = zeros(lagmax,1);
for p=1:lagmax
```

```
    % re-estimate model for each case of p
    Ylags        = lagmatrix(Ynew,1:p);               % lagmatrix for GDP growth
    Xlags        = lagmatrix(Xnew,0:p);               % lagmatrix for exogenous tax
changes
    Ytemp        = Ynew(p+1:end,1);               % dependent variable Y
    Xtemp        = [Xlags(p+1:end,:),Ylags(p+1:end,:)];               % X=[Xlags
Ylags]
    [nobs,npara] = size(Xtemp);       % number of observations and number of
parameters
    mdl_p        = fitlm(Xtemp,Ytemp);% function: fitlm to estimate the model
    logL         = mdl_p.LogLikelihood;               % store value of loglikelihood
    [aic,bic,ic] = aicbic(logL,npara,nobs);               % function: aicbic for
calculate information criteria
    % save the results
    AIC(p)  = aic;
    AICc(p) = ic.aicc;
    BIC(p)  = bic;
    HQ(p)   = ic.hqc;
end
[~,AIClags]    = min(AIC);
[~,AICclags]   = min(AICc);
[~,BIClags]    = min(BIC);
[~,HQlags]     = min(HQ);

% display
ICtable = array2table([(1:lagmax)',AIC,AICc,BIC,HQ],"VariableNames",
["lags","AIC","AICc","BIC","HQ"]);
disp(ICtable)
```

lags	AIC	AICc	BIC	HQ
1	652.01	652.11	662.48	656.23
2	651	651.26	668.43	658.02
3	648.16	648.64	672.52	657.98
4	646.62	647.4	677.91	659.23
5	646.54	647.7	684.73	661.93
6	647.08	648.71	692.17	665.25
7	638.98	641.16	690.93	659.92
8	632.06	634.89	690.88	655.78
9	632.3	635.85	697.95	658.77
10	633.62	638	706.09	662.84

```
disp(['AIC selects the lag number = ', num2str(AIClags)])
```

```
AIC selects the lag number = 8
```

```
disp(['AICc selects the lag number = ', num2str(AICclags)])
```

```
AICc selects the lag number = 8
```

```
disp(['BIC selects the lag number = ', num2str(BIClags)])
```

```
BIC selects the lag number = 1
```

```
disp(['HQ selects the lag number = ', num2str(HQlags)])
```

```
HQ selects the lag number = 8
```