Solutions to Problem Set 1

1. By the algebra of OLS, this is equivalent to a t-test testing the hypothesis of a slope coefficient of 1 in a regression of $\tilde{y}_t$ on $\tilde{y}_{t-1}$ where $\tilde{y}_t$ is the residual in a regression of y_t on a constant and the time trend, i.e.

$$\tilde{y}_t = y_t - \begin{pmatrix}1 & t\end{pmatrix}\begin{pmatrix}T & \Sigma t \\ \Sigma t & \Sigma t^2\end{pmatrix}^{-1}\begin{pmatrix}\Sigma y_t \\ \Sigma t y_t\end{pmatrix}$$

$$\therefore \tilde{y}_t = y_t - \begin{pmatrix}1 & t\end{pmatrix}\begin{pmatrix}T & T(T+1)/2 \\ T(T+1)/2 & T(T+1)(2T+1)/6\end{pmatrix}^{-1}\begin{pmatrix}\Sigma y_t \\ \Sigma t y_t\end{pmatrix}$$

$$\therefore \tilde{y}_t = y_t - \begin{pmatrix}1 & t/T\end{pmatrix}\begin{pmatrix}T & T(T+1)/2T \\ T(T+1)/2T & T(T+1)(2T+1)/6T^2\end{pmatrix}^{-1}\begin{pmatrix}\Sigma y_t \\ \Sigma \frac{t}{T} y_t\end{pmatrix}$$

$$\therefore T^{-1/2}\tilde{y}_t = T^{-1/2} y_t - \begin{pmatrix}1 & t/T\end{pmatrix}\begin{pmatrix}1 & (T+1)/2T \\ (T+1)/2T & (T+1)(2T+1)/6T^2\end{pmatrix}^{-1}\begin{pmatrix}T^{-3/2}\Sigma y_t \\ T^{-3/2}\Sigma \frac{t}{T} y_t\end{pmatrix}$$

$$\therefore T^{-1/2}\tilde{y}_{[Tr]} \to \sigma B(r) - \begin{pmatrix}1 & r\end{pmatrix}\begin{pmatrix}1 & 1/2 \\ 1/2 & 1/3\end{pmatrix}^{-1}\begin{pmatrix}\sigma\int_0^1 B(r)dr \\ \sigma\int_0^1 rB(r)dr\end{pmatrix}$$

$$\therefore T^{-1/2}\tilde{y}_{[Tr]} \to \sigma B(r) - \begin{pmatrix}1 & r\end{pmatrix}\begin{pmatrix}4 & -6 \\ -6 & 12\end{pmatrix}\begin{pmatrix}\sigma\int_0^1 B(s)ds \\ \sigma\int_0^1 sB(s)ds\end{pmatrix}$$

$$\therefore T^{-1/2}\tilde{y}_{[Tr]} \to \sigma B(r) - \sigma\{4\int_0^1 B(s)ds - 6\int_0^1 sB(s)ds - 6r\int_0^1 sB(s)ds + 12r\int_0^1 sB(s)ds\}$$

$$\therefore T^{-1/2}\tilde{y}_{[Tr]} \to \sigma B^\tau(r)$$

where

$$B^\tau(r) = B(r) - \{4\int_0^1 B(s)ds - 6\int_0^1 sB(s)ds - 6r\int_0^1 B(s)ds + 12r\int_0^1 sB(s)ds\}$$

And hence, $t \to \dfrac{\frac{1}{2}(B^\tau(1)^2 - 1)}{\sqrt{\int_0^1 B^\tau(r)^2 dr}}$.

Here's the program for simulating this:.

```
randn('seed',123);
n=1000;
for imc=1:1000;
b=cumsum(randn(n,1)/sqrt(n));
s=[1/n:1/n:1]';
bstar=b-((4*mean(b))-(6*mean(s.*b))-(6*mean(b)*s)+(12*mean(s.*b)*s));
t(imc)=0.5*((bstar(end)^2)-1)/sqrt(mean(bstar.^2));
end;
t=sort(t);
[t(10) t(50) t(100)]
```

Running this code, the answer is:
-3.6480 -3.1278 -2.8331

2. $S_T(r) = T^{-1/2}\Sigma_{t=1}^{[Tr]}(\varepsilon_t - \bar{\varepsilon}) = T^{-1/2}\Sigma_{t=1}^{[Tr]}\varepsilon_t - rT^{1/2}\bar{\varepsilon} \to B(r) - rB(1)$
also known as a "Brownian bridge".

3. Here is the program. For most series, the null of a unit root is not rejected.

```
clear all
clc
fprintf('Running...\n\n');

[year cpi emp gnpd ip gnpc r m ngnp rgnp rw sp u v
w]=textread('nelson_plosser.txt', '%f %f %f %f %f %f %f %f %f %f %f %f %f %f
%f', 'headerlines', 1);
DATA=[year cpi emp gnpd ip gnpc r m ngnp rgnp rw sp u v w];
labels=char('year','cpi','emp','gnpd','ip','gnpc','r','m','ngnp','rgnp','rw',
'sp','u','v','w');
lcpi=cpi;
lcpi(end,:)=[];
dcpi=cpi;
dcpi(1,:)=[];
dcpi=dcpi-lcpi;

LAG=zeros(1,14);
Ts=zeros(9,14);
H=zeros(14,1);
for i=2:15;
    BIC=zeros(9,1);
    tstats=zeros(9,1);
    for j=0:8;
        Y=DATA(:,i);
        lY=Y;
        lY((end),:)=[];
        if j>0;
            lY((1:(j)),:)=[];
        end
        Y((1:1+j),:)=[];
        t=[1:size(Y,1)]';
        X=ones(size(Y,1),3+j);
        L=zeros(size(Y,1),(j+1));
        dY=Y-lY;
        X(:,2)=t;
        X(:,3)=lY;
        L(:,1)=lY;
        for l=0:j;
            if l>0;
            newlY=DATA(:,i);
            if (j-l)>0;
                newlY((1:(j-l)),:)=[];
            end
            newlY((end-l:end),:)=[];
            L(:,l+1)=newlY;
```

```
            X(:,3+l)=L(:,l+1)-L(:,l);
            end
        end
        b=(X'*X)^(-1)*X'*dY;
        e=dY-X*b;
        SSR=e'*e;
        BIC(j+1)=size(Y,1)*log(SSR/size(Y,1))+(3+j)*log(size(Y,1));
        s2=SSR/(size(Y,1)-2);
        var_b=s2*(X'*X)^(-1);
        se_b=var_b(3,3)^(1/2);
        tstats(j+1)=b(3)/se_b;
        Ts(j+1,i-1)=tstats(j+1);
    end
[minBIC, optlag]=min(BIC);
optlag=optlag-1;
LAG(i-1)=optlag;
if tstats(optlag+1)<-2.89;
    H(i)=1;
end
end

lookup=zeros(14,1);
for i=1:14
    lookup(i)=Ts(LAG(i)+1,i);
end

c0=[-8.38 -35.87 -6.42 -23.04 -35.87 -6.42 -21.64 -13.79 -34.13 1.26 -16.26 -
40 -13.79 -17.58];
c1=[4.85 1.8 4.96 3.57 1.8 4.96 3.73 4.49 2.07 5.30 4.27 -5.57 4.49 4.15];

for i=1:14;
    disp([num2str(deblank(labels(i+1,:))) ':']);
    if H(i)==1;
        fprintf('Reject.\n');
    elseif H(i)==0;
        fprintf('Fail to Reject.\n');
    end
    cilo=1+(c0(i)/(80-(LAG(i)+1)));
    cihi=1+(c1(i)/(80-(LAG(i)+1)));
    disp(['Alpha in [' num2str(cilo) ',' num2str(cihi) ']']);
    fprintf('\n');
end


fprintf('Done.\n');
```

4.
```
clear all
clc
randn('seed',1);
fprintf('Running...\n\n');

alpha=[.9; .95; .99; 1];
```

```
Prob=zeros(size(alpha,1),1);

T=100;

cval=tinv(.975,T-3);

replics=1000;

x=zeros(T,1);
y=zeros(T,1);

for a=1:size(alpha,1);
    H=zeros(replics,1);
    for j=1:replics;
        for i=2:T;
            u=randn;
            v=randn;
            x(i)=alpha(a)*x(i-1)+u;
            y(i)=alpha(a)*y(i-1)+v;
        end
        X=[ones(T,1) x];
        b=regress(y,X);
        e=y-X*b;
        s2=(e'*e)/(T-2);
        var_b=s2*(X'*X)^(-1);
        se_b=var_b(2,2)^(1/2);
        t=b(2)/se_b;
        if abs(t)>cval;
            H(j)=1;
        end
    end
Prob(a)=sum(H)/replics;
end
Prob
fprintf('These are the probabilites of rejecting the hypothesis \nthat the
slope coefficient is equal to 0 \nfor when alpha is equal to .9, .95, .99,
and 1, respectively.\n');
```

Answers are
0.5210
0.6060
0.7120
0.7620

5. X_M is the monthly data and Y_Q is the quarterly GDP data.

```
T_Q=size(Y_Q,1);
T_M=size(X_M,1);
    % generate matrix to convert monthly to quarterly data
    for q=1:T_Q;
```

```
    C(q,(q-1)*3+1:q*3)=ones(1,3);
    end
% run OLS
X_Q=cat(2,C*X_M);
beta_hat_Q=inv(X_Q'*X_Q)*(X_Q'*Y_Q);
e_hat=Y_Q-X_Q*beta_hat_Q;
ac0=corrcoef(e_hat(1:end-1),e_hat(2:end));
ac=ac0(1,2);
% match 1st order autocorrelations
a_hat=fsolve(@(a)(a^5+2*a^4+3*a^3+2*a^2+a)/(3+2*a^2+4*a)-ac,0);
        for t=1:3*T_Q;
        v1(t)=a_hat^(t-1);
        end
V_hat=toeplitz(v1);
% run FGLS
beta_hat_FGLS=inv(X_Q'*inv(C*V_hat*C')*X_Q)*X_Q'*inv(C*V_hat*C')*Y_Q;
e_hat_FGLS=Y_Q-X_Q*beta_hat_FGLS;
% estimate monthly gdp data
Y_M_hat=X_M*beta_hat_FGLS+V_hat*C'*(inv(C*V_hat*C'))*e_hat_FGLS
plot(4*Y_M_hat);
```

Here is the output