
Matlab Demonstration (11/01/2013)

We solve Exercise 11(c) of Section 3.10.

We create 2 functions:

```
function ret = myF(x)
ret = [x(1)^2 + x(2)^3 - 5; x(1)^3 + x(2)^2 - 2];
```

and

```
function ret = myJ(x)
ret = [2*x(1),  3*x(2)^2;  3*x(1)^2, 2*x(2)];
```

We use the modified `newton_sys.m`:

```
function y = newton_sys_mod ( F, J, x0, TOL, Nmax )

%NEWTON_SYS   solve the system of nonlinear equations F(x) = 0 using
%             Newton's method
%
%     calling sequences:
%             y = newton_sys ( F, J, x0, TOL, Nmax )
%             newton_sys ( F, J, x0, TOL, Nmax )
%
%     inputs:
%             F       vector-valued function of a vector argument which
%                     defines the system of equations to be solved
%             J       matrix-valued function which computes the Jacobian
%                     associated with the function F
%             x0      vector containing initial guess for solution of
%                     nonlinear system
%             TOL     convergence tolerance - applied to maximum norm of
%                     difference between successive approximations
%             NMax    maximum number of iterations to be performed
%
%     output:
%             y       approximate solution of nonlinear system
%
%     dependencies:
```

```
%             this routine uses both LUfactor and LUsolve
%
%     NOTE:
%             if NEWTON_SYS is called with no output arguments, each
%             approximation to the solution is displayed
%
%             if the maximum number of iterations is exceeded, a meesage
%             to this effect will be displayed and the current approximation
%             will be returned in the output value
%

old = x0;
for i = 1 : Nmax
    Fold = feval(F,old);
Jold = feval(J,old);
dx = -Jold\Fold;
    new = old + dx;

if ( nargout == 0 )
   disp ( new )
end

if ( max(abs(dx)) < TOL )
   if ( nargout == 1 )
      y = new;
   end
   return
else
   old = new;
end
end

disp('newton_sys error: Maximum number of iterations exceeded');
if ( nargout == 1 ) y = new; end;
```

Use:

```
> newton_sys('myF', 'myJ', [1;1], 1e-10, 100)
```