```matlab
% Check the gradient and Hessian of the objective function for the
 problem
% in Boyd and Vandenberghe using finite differences.
%
% Author: Terrence Alsup
% Date: March 10, 2020
% File: FDcheck.m

% Load the data and get the dimensions.
data = open('Adata.mat');
A = data.A;
[n, m] = size(A);

% Set up the objective function.
fun = @(x) objective_fun(x, A);


% Take x = 0 as the test point since we know it is in the domain.
x = zeros(n, 1);
% Compute the true function value, gradient, and Hessian.
[f, g, H] = fun(x);

% Compute the vector of finite differences of f (i.e. approximate
 gradient)
numchecks = 9; % Compute differences for dx = 10^-1,...,10^-9.

% Save the results.
grad_fderror = zeros(numchecks, 1);
hess_fderror = zeros(numchecks, 1);

fd_grad = zeros(n, 1);
fd_hess = zeros(n, n);

dx = 0.1;
for j=1:numchecks
    % Check the gradient.
    for i=1:n
        % e is the standard basis vector.
        e = zeros(n,1);
        e(i) = 1;
        fd_grad(i) = (fun(x + dx*e) - f)/dx;
    end
    grad_fderror(j) = norm(fd_grad - g);

    % Check the Hessian.
    for i=1:n
        % Sstandard basis vector.
        e = zeros(n, 1);
        e(i) = 1;
        [~, gdx, ~] = fun(x + e*dx);
        fd_hess(:,i) = (gdx - g)/dx;
    end
end
```

```
    hess_fderror(j) = norm(H - fd_hess);

    dx = dx/10;
end

% Plot the results.
figure(1); % Gradient FD plot.
loglog(0.1.^(1:numchecks), grad_fderror, '-+');
xlabel('Difference: dx');
ylabel('Error')
title('Error of gradient finite difference approximation');

figure(2); % Hessian FD plot.
loglog(0.1.^(1:numchecks), hess_fderror, '-+');
xlabel('Difference: dx');
ylabel('Error')
title('Error of Hessian finite difference approximation');
```

Published with MATLAB® R2019b