

---

```
% this is for the 8th problem in homework2 from cpsc 546
% approximate Newton method
clear;close all;
m = 4;
n = 5;
A = rand(m,n);
x0 = zeros(n,1);

f = @(x) -sum(log(1-A*x))-sum(log(ones(n,1)-x.^2));
g = @(x) A'*(1./(1-A*x))+(2.*x./(1-x.^2));
h = @(x) A'*diag((1./(1-A*x)).^2)*A + diag(1./(1+x).^2 + 1./(1-x).^2);
```

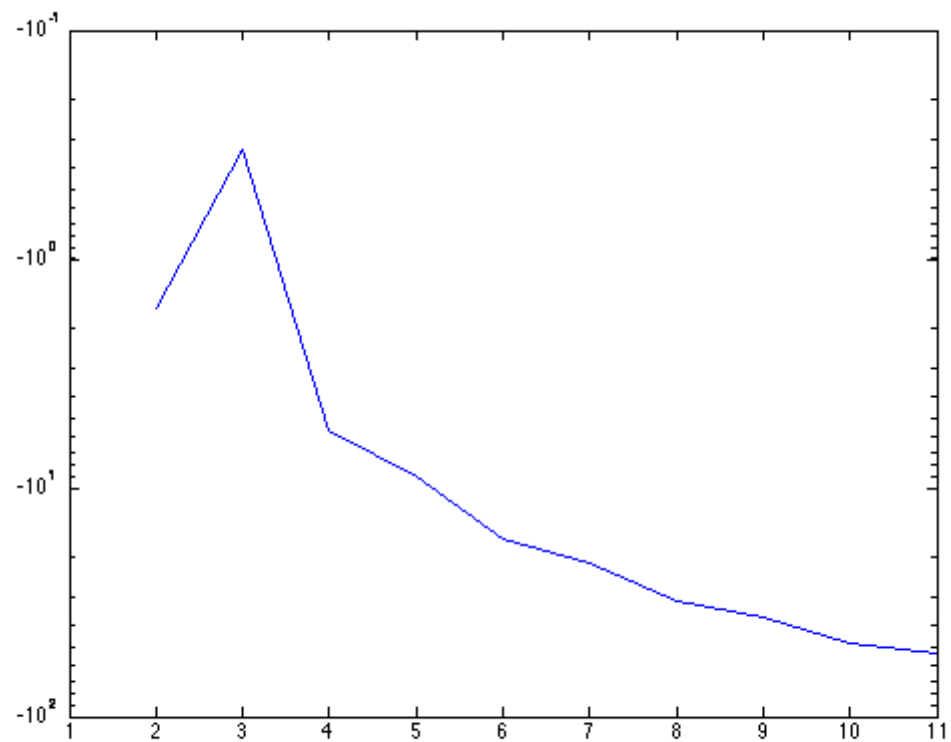
## Re-using the Hessian.

```
N = 2;
itnmax = 10;
itn = 1;
tol = 10^-10;
while 1
    if mod(itn,N) ~= 0
        p = -diag(diag(h(x0)))\g(x0);
    else
        p = p;
    end
    x0 = x0+p;
    itn = itn+1;
    plot_data(itn,:) = [f(x0)];

    if norm(p)<=tol || itn>itnmax
        break;
    end

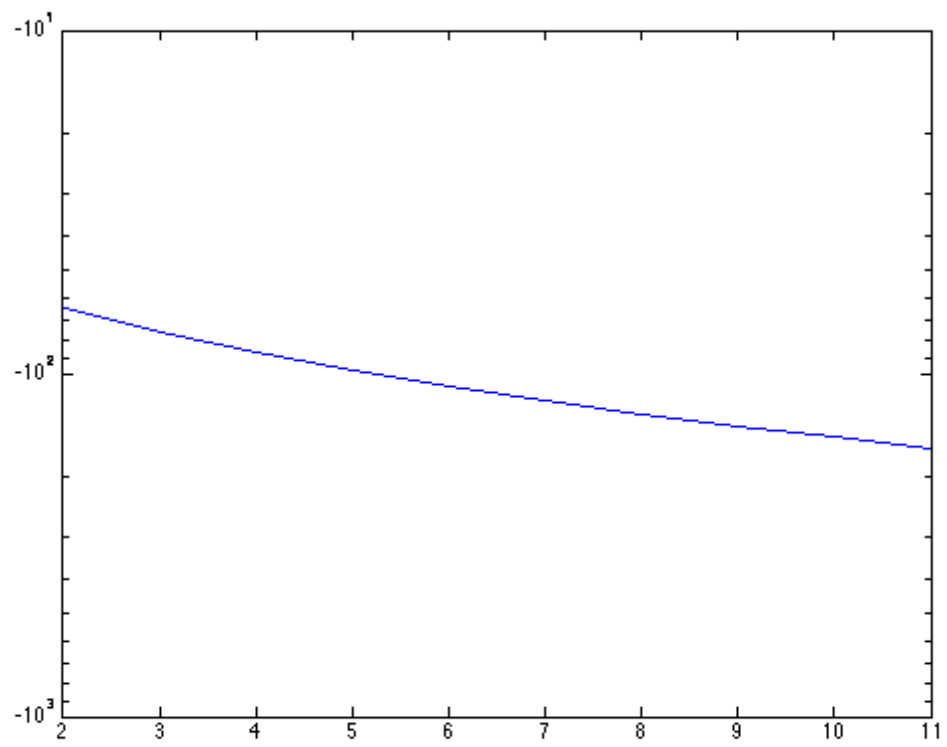
end

figure;semilogy(1:1:itn,real(plot_data(:,1)))
% figure;plot(1:1:itn,real(plot_data(:,2)))
```



## Diagonal approximation.

```
itnmax = 10;  
itn = 1;  
tol = 10^-10;  
while 1  
    p = -diag(diag(h(x0)))\g(x0);  
    x0 = x0+p;  
    itn = itn+1;  
    plot_data(itn,:) = [f(x0)];  
  
    if norm(p)<=tol || itn>itnmax  
        break;  
    end  
  
end  
  
figure;semilogy(1:1:itn,real(plot_data(:,1)))  
% figure;plot(1:1:itn,real(plot_data(:,2)))
```



*Published with MATLAB® 7.12*