Contents


```
function [r_norm] = cg_standard(A, b, x_0, show)
```

CG_STANDARD Introduction

```
         Prints summary of conjugant gradient algorithm for solving
         standard linear Ax = b equation.

A:       The left-hand-side matrix in Ax = b


b:       The right-hand-side of the equation to be solved


x_0:     The vector from where to start the CG method search.


Vectors stored:
      a_vec:  vector of alpha values, of the same length of x_0
      x_vec:  values for iterative solution to CG method
      r_vec:  vector of residuals, of the same length of x_0
      B_vec:  scalar to make p_{k} and p_{k-1} A-conjugate
      p_vec:  conjugate vector, of the same length as x_0
```

Establishing solution placeholders

```
    size = length(x_0);
    nonz = nnz(A);

    a_vec = zeros(size,1);
    x_vec = zeros(size,1);
    r_vec = zeros(size,1);
    B_vec = zeros(1,1);
    p_vec = zeros(size,1);

    x_vec(:,1) = x_0;
    r_vec(:,1) = A * x_0 - b;
    p_vec(:,1) = -r_vec(:,1);

    r_norm = zeros(1);

    k = 1;

    tolerance = 1e-6;
```

The CG method

```
    tic
    while norm(r_vec(:,k)) > tolerance

        a_vec(k)     = r_vec(:,k)' * r_vec(:,k) / ...
                       (p_vec(:,k)' * A * p_vec(:,k));

        x_vec(:,k+1) = x_vec(:,k) + a_vec(k) * p_vec(:,k);

        r_vec(:,k+1) = r_vec(:,k) + a_vec(k) * A * p_vec(:,k);
        r_norm(k) = norm(r_vec(:,k+1));

        B_vec(:,k+1) = r_vec(:,k+1)' * r_vec(:,k+1) / ...
                       (r_vec(:,k)' * r_vec(:,k));

        p_vec(:,k+1) = -r_vec(:,k+1) + B_vec(:,k+1) * p_vec(:,k);

        k = k+1;

    end
    total_time = toc;
```

Summarizing Results

```
    if show == "yes"

        dis = ["Solving Ax=b for " + size + " x " + size + " matrix A," ...
               + " which has " + size*size + " elements and " ...
               + nonz + " nonzero elements. There were " ...
               + k + " iterations needed to reach zero residual." ...
               + " A total time of " + total_time + " seconds elapsed." ...
               + " The 1D solution vector for x is:"];

        disp(dis);
        disp(x_vec(:,end))

    end


end
```