

```

function [final_cluster_matrix final_delta] = iterative_delta_clustering_unsup(dataset, mutually_exclusive, geometric,
N)

num_rows = size(dataset,1);

%if true, then we apply geometric averaging (larger deltas)
if(geometric)

    s = std(dataset(:,1:N)); %calculates the standard deviation of the dataset in each dimension
    s = norm(s)/e

%otherwise, we apply arithmetic averaging (smaller deltas)
else

    s = std(dataset(:,1:N)); %calculates the standard deviation of the dataset in each dimension
    s = mean(s); %takes the average standard deviation
    alpha = 1.5; %this is a constant used to adjust the standard deviation
    s = s*alpha

endif
%-----

num_iterations = 25;
delta_vector = [s/num_iterations : s/num_iterations : s];
delta_vector = delta_vector.^2; %we square to save one operation

max_change_count = 0;

%we have to run one iteration outside the loop-----
delta = delta_vector(1);
is_available = ones(1,num_rows);
current_cluster_matrix = zeros(num_rows, num_rows);
final_cluster_matrix = current_cluster_matrix;

final_delta = 0;

for i = 1 : num_rows

    current_vector = dataset(i,1:N);

    for j = 1 : num_rows

        test_vector = dataset(j,1:N);
        diff_vector = (current_vector .- test_vector).^2;
        diff = sum(diff_vector);

        %if true, then we add row j to cluster i
        if(diff <= delta && is_available(j))

            current_cluster_matrix(i,j) = 1;

            if(mutually_exclusive == 1)

                is_available(j) = 0;

            endif

        endif

    endfor

endfor

```

```

endfor

prior_cluster_matrix = current_cluster_matrix;

%main loop-----
for k = 2 : num_iterations

    delta = delta_vector(k);
    is_available = ones(1,num_rows);
    current_cluster_matrix = zeros(num_rows, num_rows);

    for i = 1 : num_rows

        current_vector = dataset(i,1:N);

        for j = 1 : num_rows

            test_vector = dataset(j,1:N);
            diff_vector = (current_vector .- test_vector).^2;
            diff = sum(diff_vector);

            %if true, then we add row j to cluster i
            if(diff <= delta && is_available(j))

                current_cluster_matrix(i,j) = 1;
                is_available(j) = 0;

            endif

        endfor

    endfor

    %counts the number of changes to the cluster_matrix
    diff_matrix = (current_cluster_matrix .- prior_cluster_matrix).^2;
    change_count = sum(diff_matrix(:))

    if(change_count > max_change_count)

        max_change_count = change_count;
        final_cluster_matrix = current_cluster_matrix;
        final_delta = delta;

    endif

    prior_cluster_matrix = current_cluster_matrix;

endfor

final_delta = sqrt(final_delta);

endfunction

```