

```

function [cluster unique_vector] = permutation_based_generate(input_vector, data_matrix, N)

M = size(data_matrix,1);
cell = {};
num_items = size(data_matrix,1);
max_num_unique_matches = 0;

%this generates permuted instances of the dataset
for i = 1 : M

    selected_rows = randperm(num_items,num_items);
    cell{i} = data_matrix(selected_rows,:);

endfor

prior_uniques = 0;
current_uniques = 0;
i = 1; %the number of items in each cell

%iterates until the number of unique matches decreases
while((current_uniques >= prior_uniques) && (i <= M))

    prior_uniques = current_uniques;

    for k = 1 : M %assembles the prediction matrix

        temp_matrix = cell{k};
        temp_matrix = temp_matrix(1:i,:); %limits the cell to the first i entries

        predicted_vector = find_NN(input_vector, temp_matrix, N);
        prediction_matrix(k,:) = predicted_vector;

    endfor

    u = unique(prediction_matrix, 'rows'); %returns the unique rows
    current_uniques = size(u,1); %counts the number of unique matches
    unique_vector(i) = current_uniques; %stores the number of unique matches

    i = i + 1; %increments the number of items per cell

endwhile

cluster = u;

endfunction

```