```
function [prior_cluster prior_unique_vector] = permutation_based_optimize(input_row, data_matrix, N)

  input_vector = data_matrix(input_row,:); %stores the input vector
  data_matrix(input_row,:) = []; %removes the input vector from the dataset

  %iterates until the cluster consists of a single vector
  [current_cluster current_unique_vector] = permutation_based_generate(input_vector, data_matrix, N);
  prior_cluster = current_cluster;
  prior_unique_vector = current_unique_vector;
  current_cluster = current_cluster;

  while(size(current_cluster,1) > 1)

    prior_cluster = current_cluster;
    prior_unique_vector = current_unique_vector;

    [current_cluster current_unique_vector] = permutation_based_generate(input_vector, current_cluster, N);

  endwhile

endfunction
```