

Untitled

```
function [data_categories_array anchor_array] = generate_categories_N(data_array,
max_delta, N)
%generates a categorization of N-dimensional vectors
%all dimensions above N are ignored.

[x num_items] = size(data_array);

index = 1; %the position in the data_set
copy_data_array = data_array;

anchor = copy_data_array{index};%sets the first anchor data point
num_anchors = 1;
anchor_array{num_anchors} = anchor; %we store each anchor in a vector that is
returned by the function

cat_num = 0; %the position in the data_categories_array
is_taken = zeros(1,num_items + 1); %a vector we use to keep track of items that
have been used

%iterates until we run out of data points in the array
while(is_taken(index) != 1 && index <= num_items)

    cnt = 0; %the position in the queue for the current anchor
    cat_num = cat_num + 1;

    %searches available items for inclusion in the current anchor category
    for j = 1 : num_items

        temp = copy_data_array{j};

        if(is_taken(j) != 1)

            %this is to ensure we ignore any labels in the data
            calc_temp = temp(1:N);
            anchor_temp = anchor(1:N);

            delta = norm(calc_temp - anchor_temp);

            if(delta <= max_delta)

                cnt = cnt + 1;
                queue{cnt} = temp;
                is_taken(j) = 1;

            endif

        endif

    endif

endif
```

Untitled

```
endfor

    if(cnt >= 1) %if the category isn't empty we load its contents into the
data_categories_array

        data_categories_array{cat_num} = queue;
        clear queue;

    endif

    %finds the next available anchor
    break_loop = false;

    index = index + 1;

    while(break_loop == false && index <= num_items)

        temp = copy_data_array{index};

        if(is_taken(index) != 1)

            break_loop = true;
            anchor = copy_data_array{index};
            num_anchors = num_anchors + 1;
            anchor_array{num_anchors} = anchor;

        else

            index = index + 1;

        endif

    endwhile

endwhile

endfunction
```