
```

function s = applyColors(s,colors, indices, sets)
%APPLYCOLORS Applies colors to a scatter plot s.
% colors can take four forms:
% 1. (1,3) RGB triple. This applies one color to all markers of the scatter
%    plot.
% 2. (m,3) RGB matrix. This applies a unique color to all markers of the
%    scatter plot as specified by the matrix.
% 3. (m,1) integer vector. This assigns a color to each marker by indexing
%    from this vector to the current colormap.
    arguments
        s
        colors
        indices = []
        sets = []
    end
    numMarkers = size(s.XData,2);
    if all(size(colors) == [numMarkers, 1]) || all(size(colors) == [1 3]) ||
all(size(colors) == [numMarkers 3])
        s.CData = colors;
    else
        applyGroupedColors(s, indices, sets, colors);
    end

end

function s = applyGroupedColors(s, indices, node_sets, colors)
%This function applies a row-wise color scheme to a
%scatter plot on the assumption that that scatter plot comes from an
%incidence matrix.
% TODO: replace this with s.CData vector input and colormap setting
%
% This function gives user control over which color to assign to each node.
% It is possible to color sets of nodes uniquely using CData's vector arg
% form, but this restricts the possible colors to MATLAB's colormaps.
%
% s: scatter object. Each point is a nz entry in the incidence
%    matrix.
% indices: (m,1) double. This is the object returned by running find()
%    on the incidence matrix that the scatter object represents. Passing in
%    the y-indices will color the scatter object by rows, and passing in
%    the
%    x-indices will color the scatter object by columns.
% node_sets: (k,1) cell array. Each cell is a 1D array of integers on 1
%    to m representing nodes. Nodes in a single cell are to be filled in
%    the
%    same color. If y-indices are passed in for indices, then these cells
%    should have integers representing hyperedges.
% colors:(k,3) RGB matrix. The ith row of this matrix is the color of the
%    ith cell of node_sets.

    ColorData = zeros(size(indices,1),3);

```

```
for i = 1:size(node_sets,2)
    color = colors(i, :);
    markers_to_color = node_sets{i};

    idx = ismember(indices, markers_to_color);
    idx = cast(repmat(idx, 1, 3), "double");

    ColorData = ColorData + idx * diag(color);
end
s.CData = ColorData;
end
```

Published with MATLAB® R2021b