Node coloring

By default, the nodes in the Mapper output are colored by the average filter value: for all points in a node, the average filter value is computed, and then a color map is applied to all nodes. Currently, the color map is Matplotlib’s default “jet” color map, with a range from the lowest to the highest filter value of all points. Low filter values are represented by blue, high filter values by red.

The “jet” color map
Default coloring is average filter value
Can also specify color by filter function via `point_color = f`
point_color = data[:,0]
point_color = (data[:,0]**2 + data[:,1]**2)**0.5
name = 'length';
point_color = (data[:,0]**2 + data[:,1]**2)**0.5
Alternatively, can specify node color

node_color = [1, 2, 3, 4, 5, 6, 7, 8]
In Jupyter notebook

Choose how to color vertices in TDA mapper graph

```
nodes = mapper_output.nodes
node_color = None

# node_color = [1, 2, 3, 4, 5, 6, 7, 8]  # Coloring the 8 nodes using 1D array
# point_color = None

point_color = f  # color nodes using average filter value
# point_color = (data[:,0]**2 + data[:,1]**2)**0.5  # color nodes in mapper output using average length
# point_color = data[:,0]  # color nodes in mapper output using average of first coordinates

name = 'custom scheme'
node_color = mapper_output.postprocess_node_color(node_color, point_color, point_labels)
```

Output the color code for each node in TDA mapper graph

```
node_color

array([-0.03425722, -0.01541899, -0.01547266, -0.000479 ,  0.00117447,  
0.01467286,  0.01467839,  0.03424094])
```
In Jupyter notebook

Choose how to color vertices in TDA mapper graph

```python
nodes = mapper_output.nodes
def node_color(nodes):
    # Coloring the 8 nodes using 1D array
    node_color = None
    node_color = [1, 2, 3, 4, 5, 6, 7, 8]
    point_color = None
    # point_color = f 
    # color nodes using average filter value
    # point_color = (data[:,0]**2 + data[:,1]**2)**0.5 
    # color nodes in mapper output using average length
    # point_color = data[:,0]
    # color nodes in mapper output using average of first coordinates
    name = 'custom scheme'
    node_color = mapper_output.postprocess_node_color(node_color, point_color, point_labels)

node_color = node_color(nodes)

array([ 1.,  2.,  3.,  4.,  5.,  6.,  7.,  8.])
```
In [56]:  nodes

| 912, 915, 916, 918, 921, 926, 927, 933, 934, 936, 943, 950, 951, 956, 957, 959, 961, 962, 966, 970, 971, 973, 980, 987, 988, 989, 999 | -0.03658378562302291 |


node_color

color(array([[-0.03425722, -0.01541899, -0.01547266, -0.0000479 ,  0.00117447,  0.01467286, 0.01467839, 0.03424094]])