

```

1  import numpy as np
2  import cvxpy as cp
3  import matplotlib.pyplot as plt
4  from scipy import sparse
5  from pygsp import graphs, filters
6  from numpy.linalg import inv
7  import itertools as it
8
9  # Count the edges given a threshold
10 def count_edges(Lap, threshold):
11     count = 0
12     size = Lap.shape[0]
13     for i in range(0,size):
14         for j in range(0,i):
15             if abs(Lap[i][j]) > threshold:
16                 count = count + 1
17     if count > 0:
18         return(count)
19     else:
20         return(-1)
21
22 def detect_edges(L1, L2, threshold):
23     count = 0
24     size = L1.shape[0]
25     for i in range(0,size):
26         for j in range(0,i):
27             if (abs(L1[i][j]) > threshold) and L2[i][j] > 0:
28                 count = count + 1
29     return(count)
30
31 # Count the number of edges two laplacians have in common.
32 # Computes the difference and counts the number of components
33 # sufficiently close to 0.
34 def compare_edges_strict(L1, L2):
35     Dif = abs(L1 - L2)
36     size = Dif.shape[0]
37     count = 0
38     for i in range(0,size):
39         for j in range(0,i):
40             if Dif[i][j] < .005:
41                 count = count + 1
42     return(count)
43
44 # Computes the proportion of correct edges in the Learned Laplacian
45 def Precision(EstL, GTL, threshold):
46     return(detect_edges(EstL, GTL, threshold)/count_edges(EstL, threshold))
47
48 # Computes the proportion edges in Ground truth Laplacian
49 # that appear in learned Laplacian
50 def Recall(EstL, GTL, threshold):
51     return(detect_edges(EstL, GTL, threshold)/count_edges(GTL, 0))
52
53 def F_Measure(EstL, GTL, threshold):
54     P = Precision(EstL, GTL, threshold)
55     R = Recall(EstL, GTL, threshold)
56     if P+R > 0:
57         return(2*((P*R)/(P+R)))
58     else:
59         return(-1)
60
61 def Relative_Error(L1, L2):
62     return(np.linalg.norm(L1 - L2)/np.linalg.norm(L1))
63
64 def SSE(L1, L2):
65     return(Relative_Error(L1,L2)**2)
66
67 def ComputeMetrics(EstL, GTL, threshold):

```

```
68     return({"Precision":Precision(EstL, GTL, threshold),
69           "Recall":Recall(EstL, GTL, threshold),
70           "F-Measure":F_Measure(EstL, GTL, threshold),
71           "Relative Error":Relative_Error(EstL, GTL),
72           "SSE":SSE(EstL, GTL)})
```