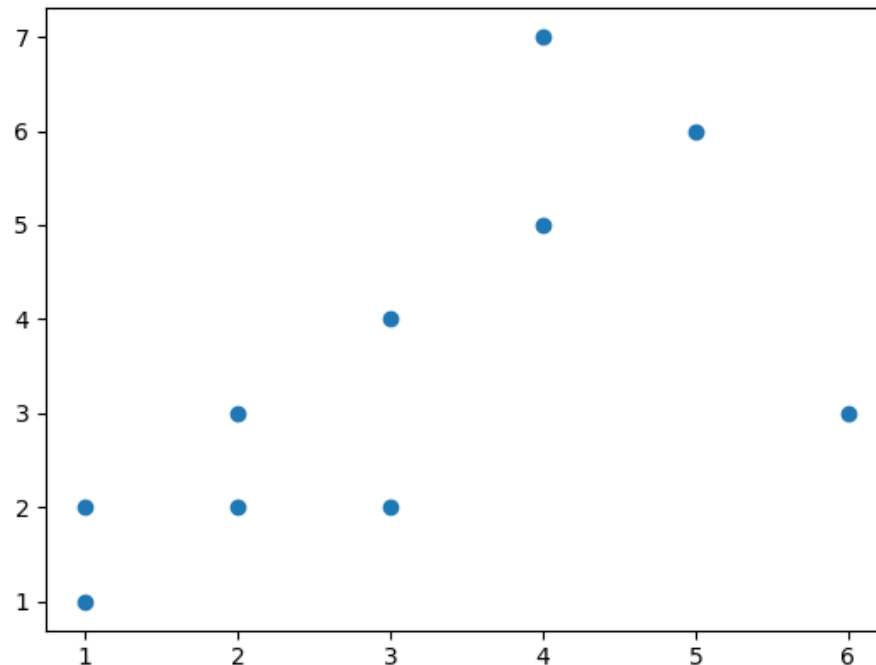


On Quality Metric Ensembles in Context of Hierarchical Clustering

Bachelor Thesis
Rene Nespithal
18.11.2019

Hierarchical Clustering

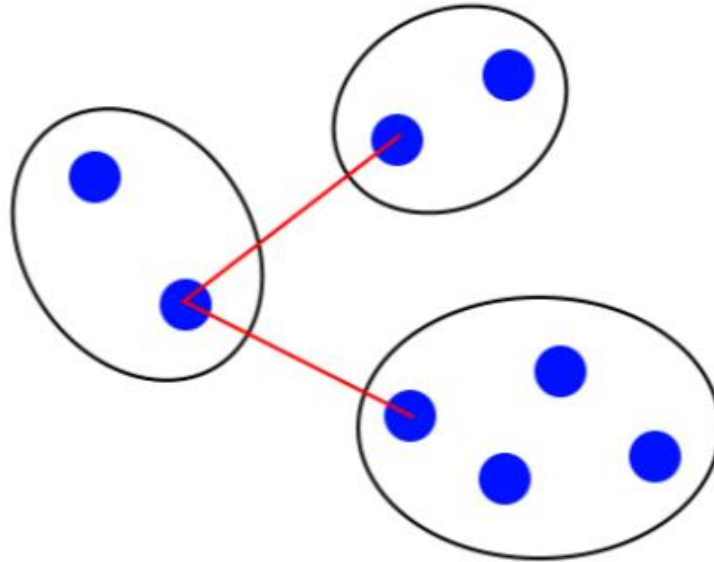
- Agglomerative ("bottom-up")
- Divisive ("top-down")



Linkage Methods

- Single Linkage

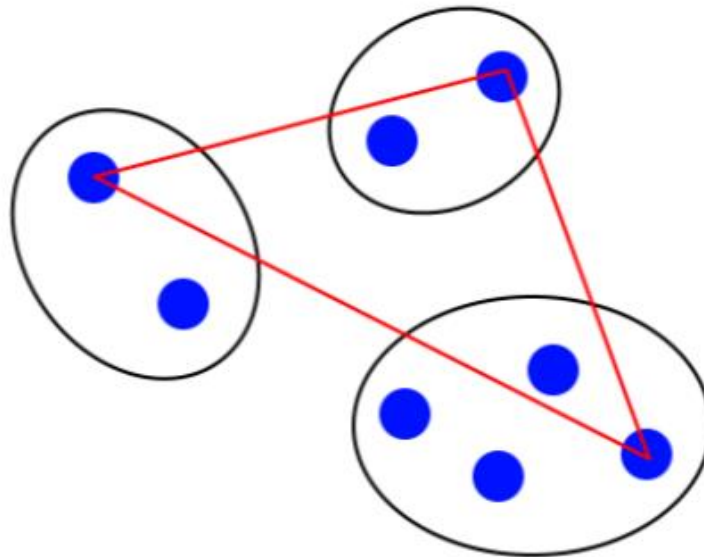
$$d(u, v) = \min(\text{dist}(u[i], v[j]))$$



Linkage Methods

- Complete Linkage

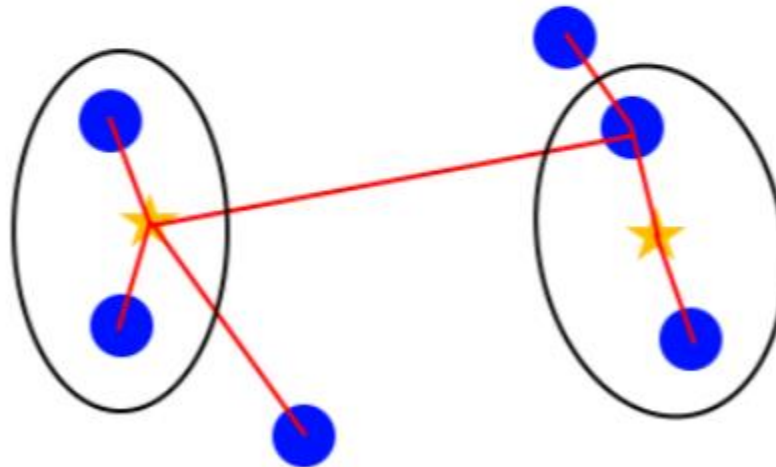
$$d(u, v) = \max(\text{dist}(u[i], v[j]))$$



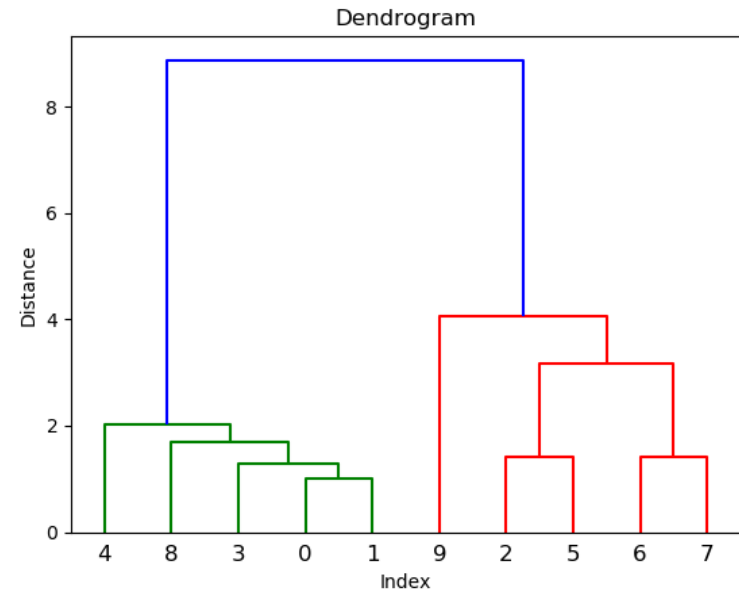
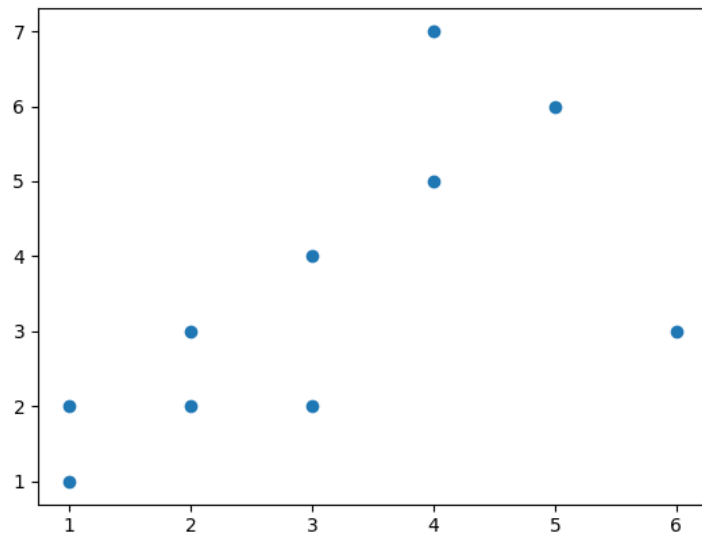
Linkage Methods

- Ward Linkage

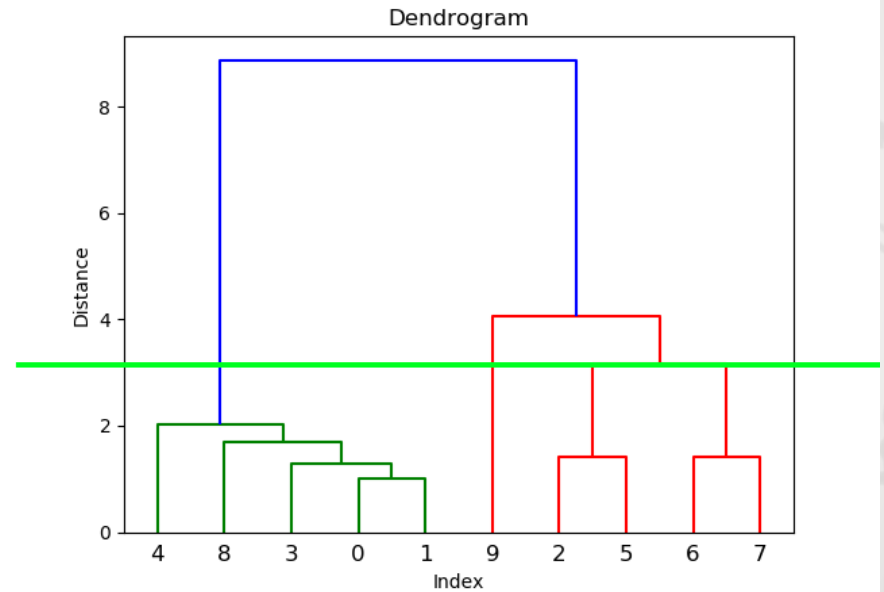
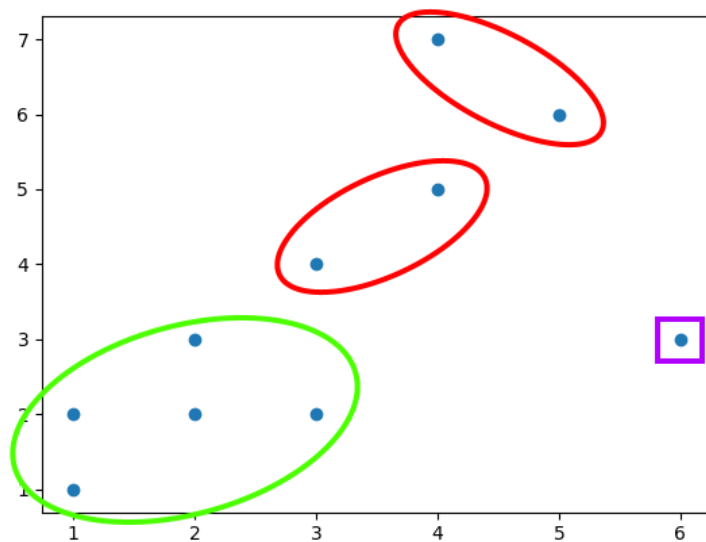
$$d(u, v) = \sqrt{\frac{|v| + |s|}{T}d(v, s)^2 + \frac{|v| + |t|}{T}d(v, t)^2 - \frac{|v|}{T}d(s, t)^2}$$



Dendrogram



Dendrogram



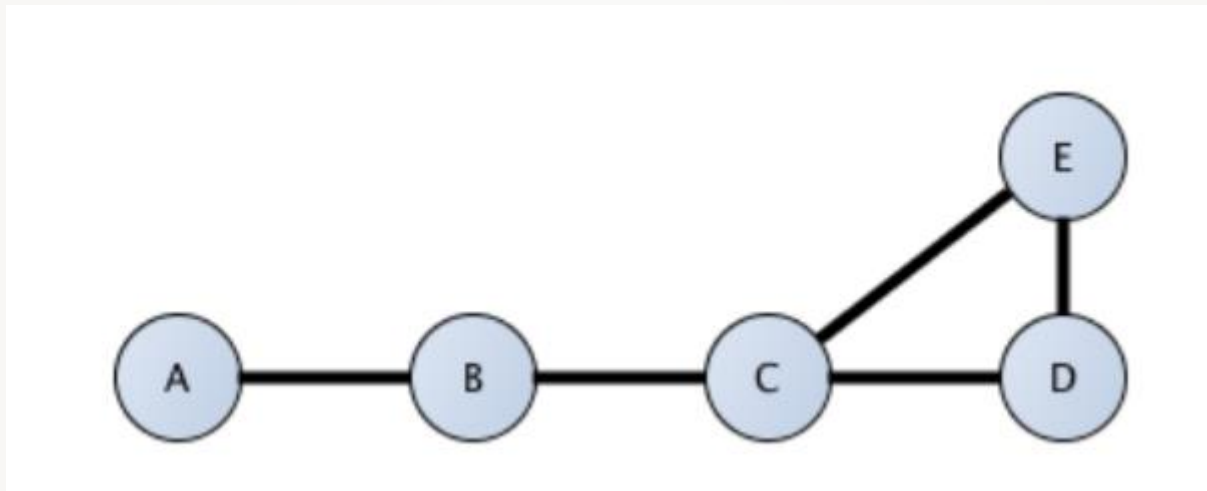
Quality Metrics

- Modularity
- Conductance
- Edge Betweenness Centrality



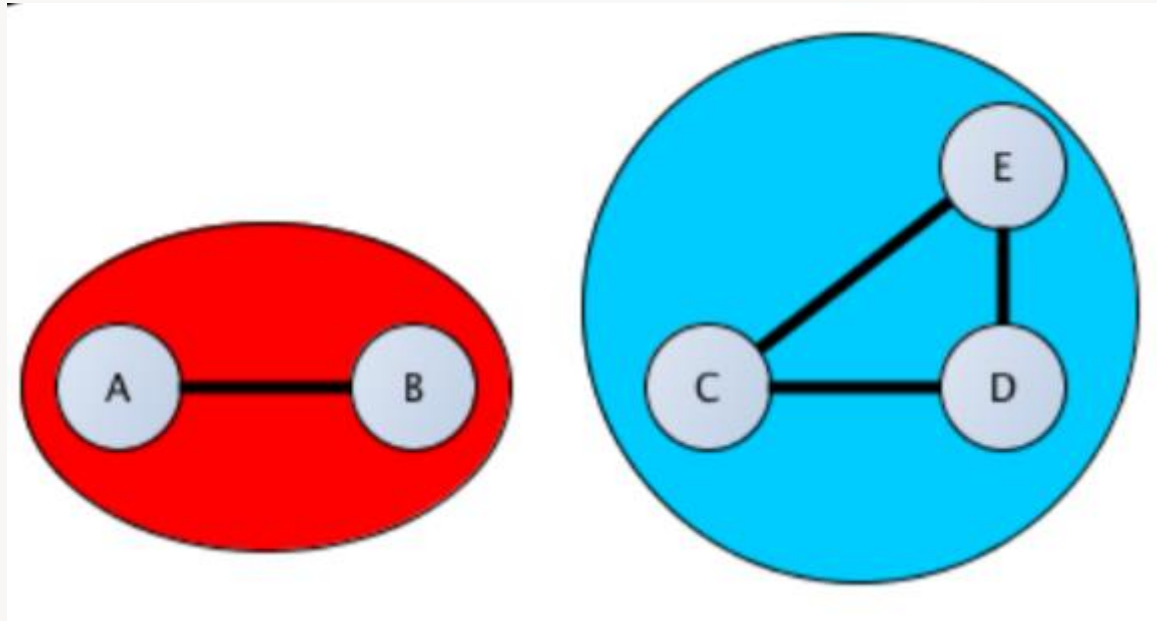
Quality Metrics

- Modularity



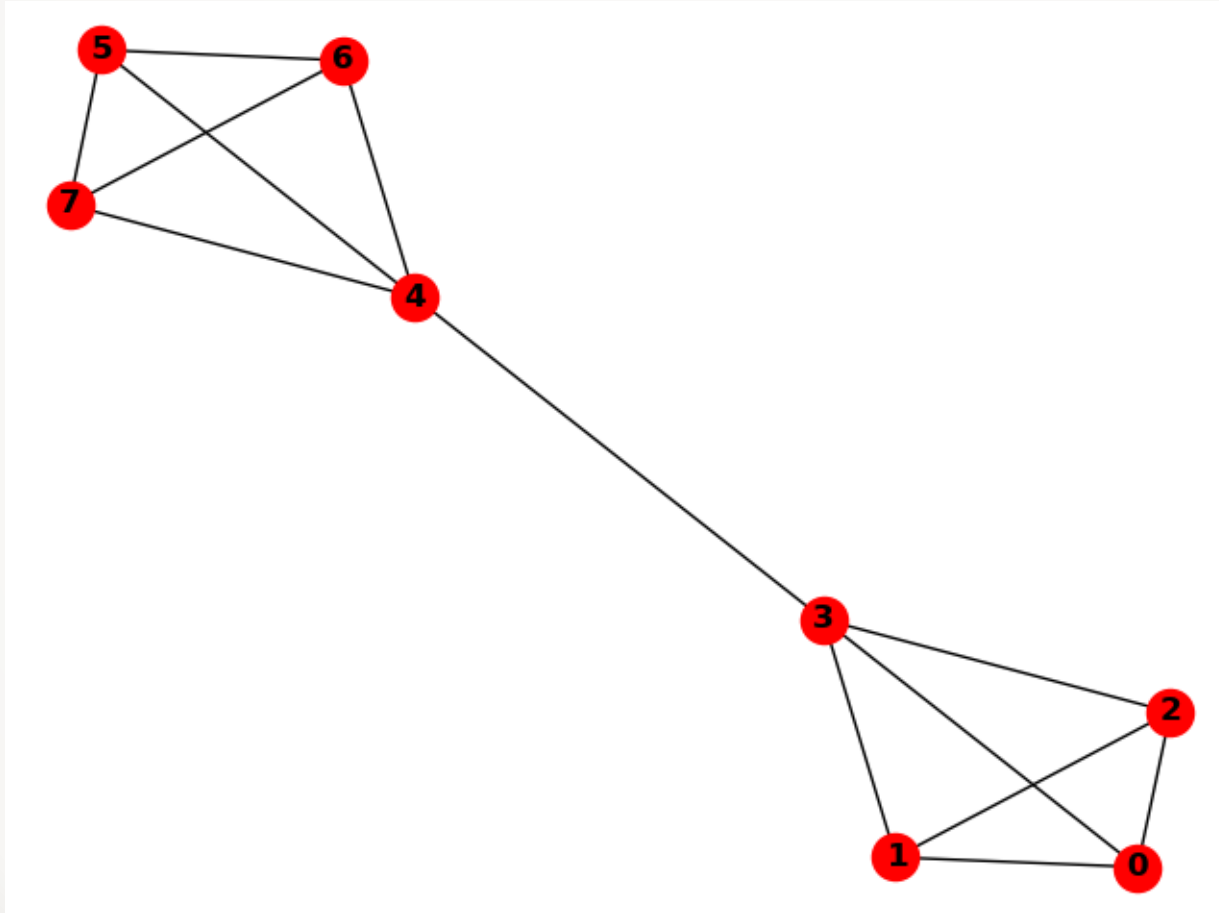
Quality Metrics

- Modularity



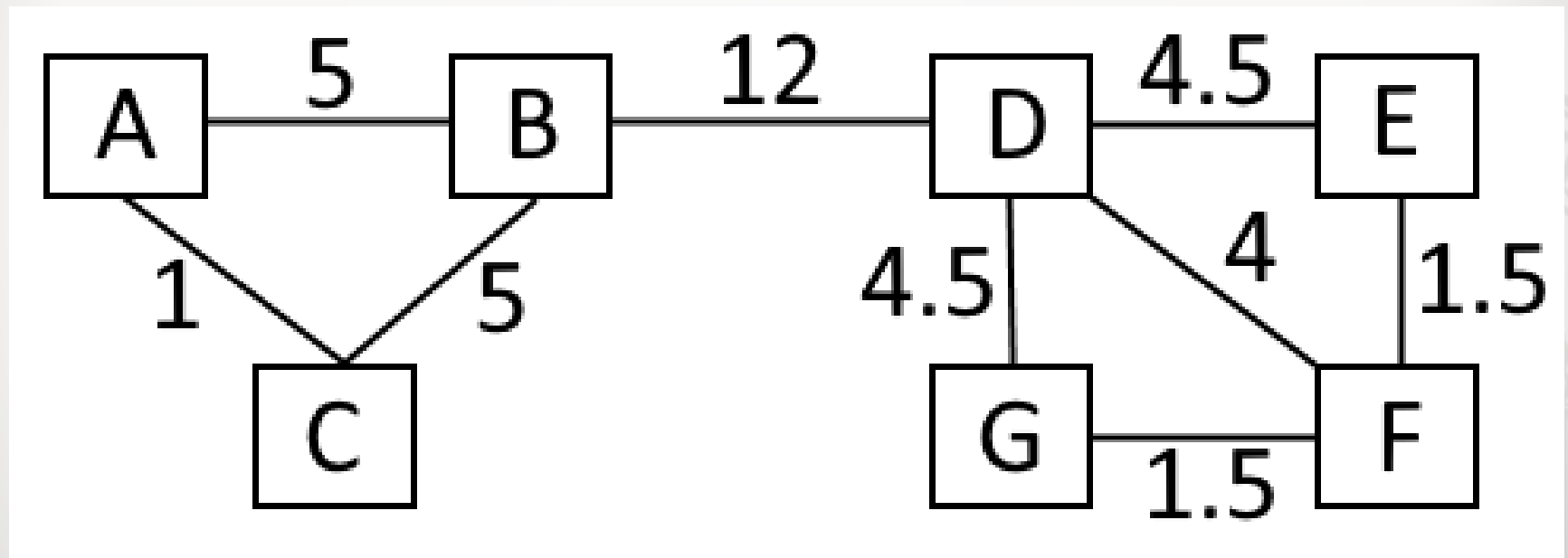
Quality Metrics

- Conductance

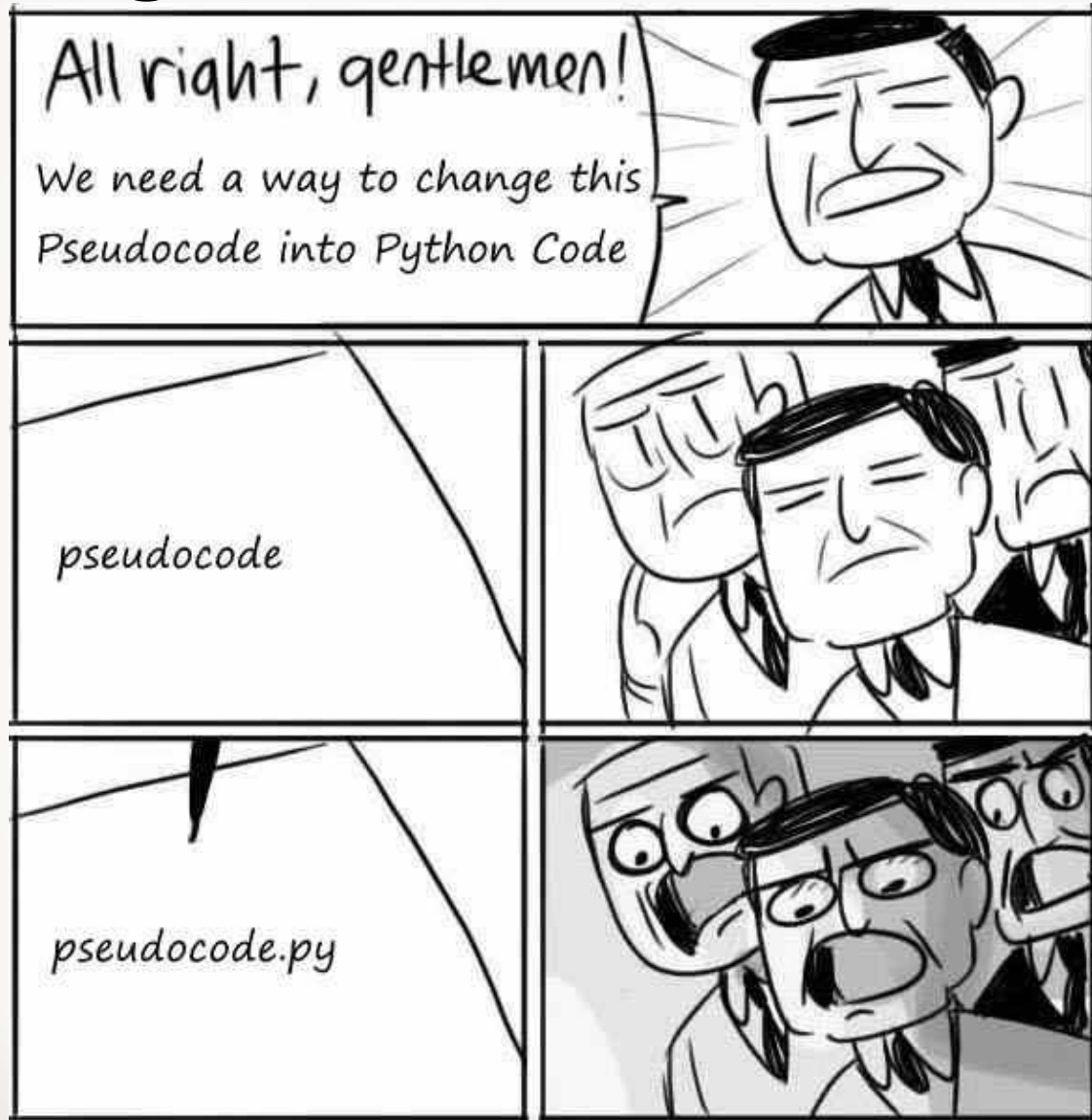


Quality Metrics

- Edge Betweenness Centrality



Dendrogram Transformation



Dendrogram Transformation

Algorithm to transform a tree-like structure into a network

```
# length of Z == amount of datapoints
# first cluster index starts at n + 1
n = len(Z)
a = []

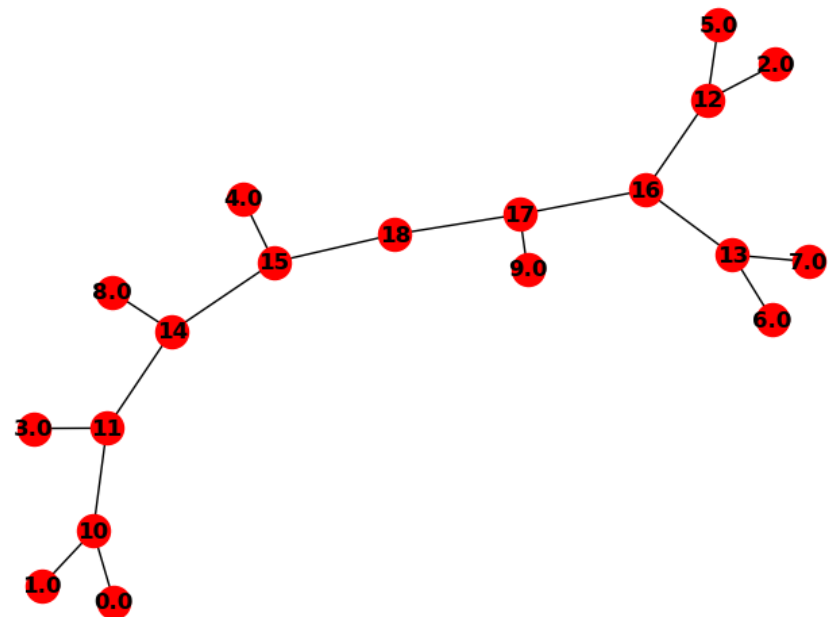
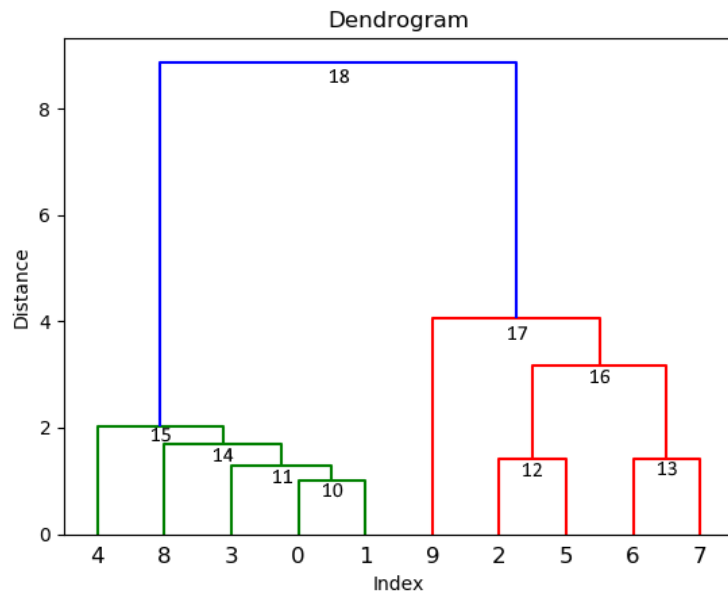
for item in Z:
    # keep track of current cluster index
    n = n + 1

    # euclidean distance between x and y
    if item[2] == 0:
        continue

    # add tuples to the nodelist
    x,y = (tuple(item[: -2]))
    a.append(tuple([x,n]))
    a.append(tuple([y,n]))

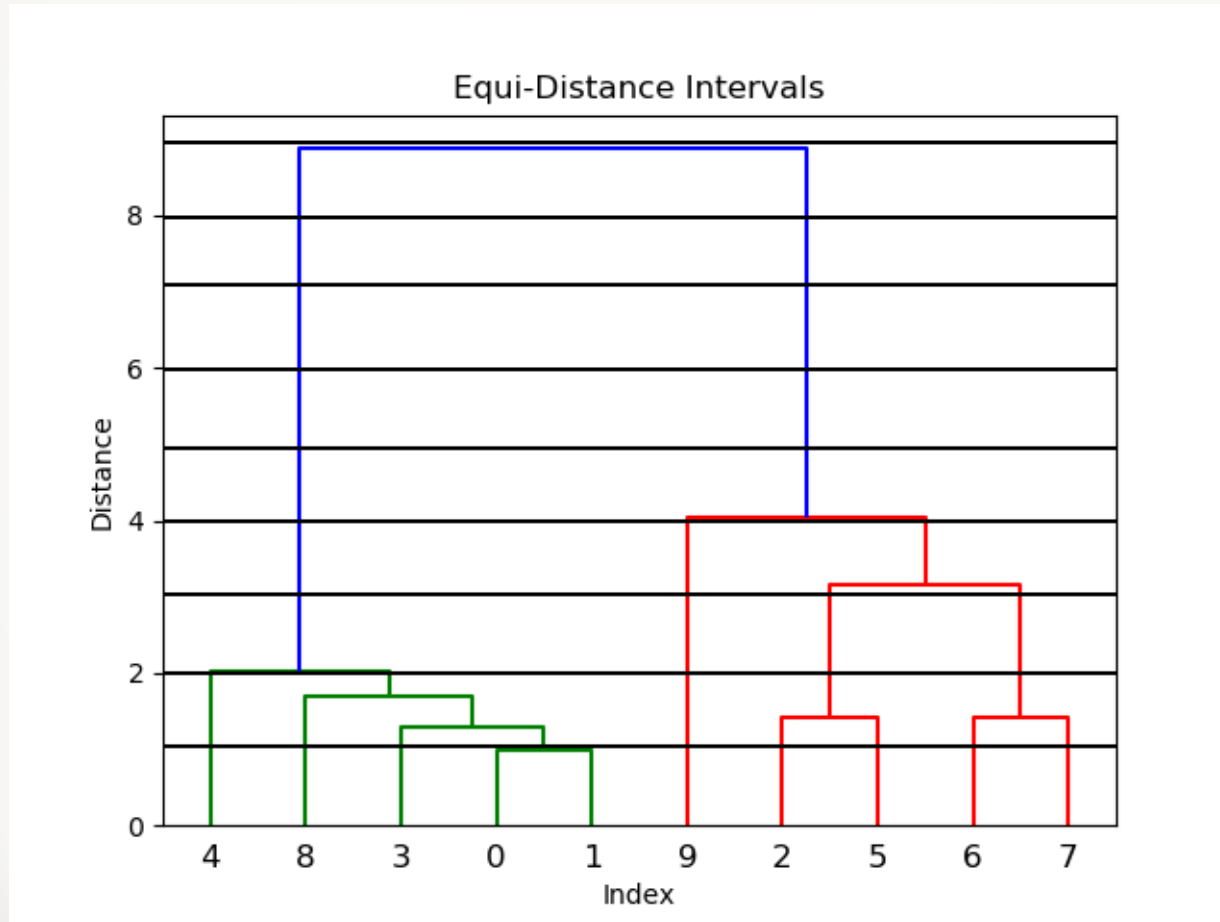
# apply nodelist to the networkX graph
networkX_graph = nx.Graph()
networkX_graph.add_edges_from(a)
```

Dendrogram Transformation



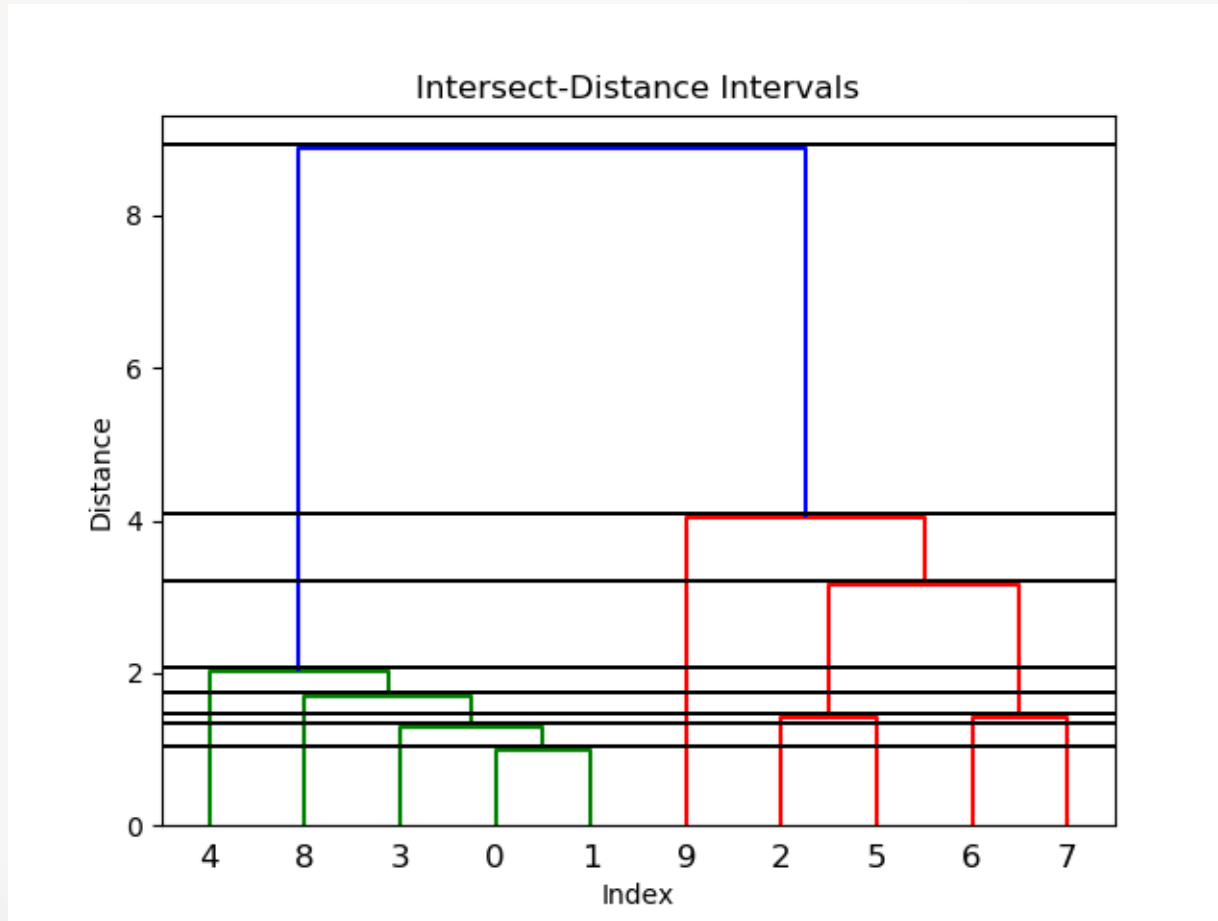
Cut Intervals

- Equi-Distant Intervals



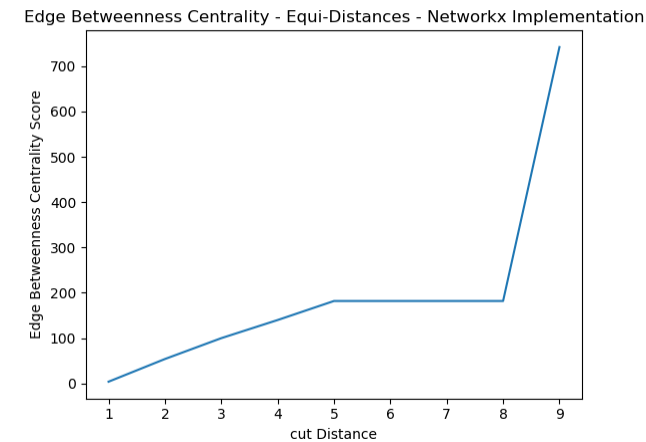
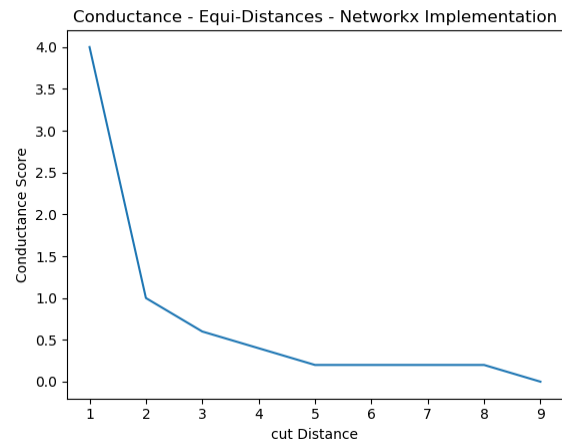
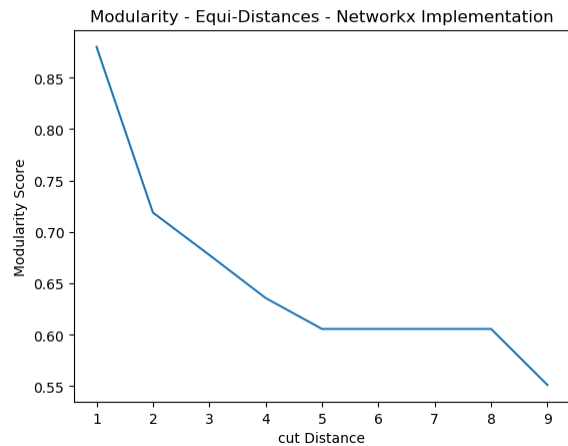
Cut Intervals

- Intersect-Distant Intervals



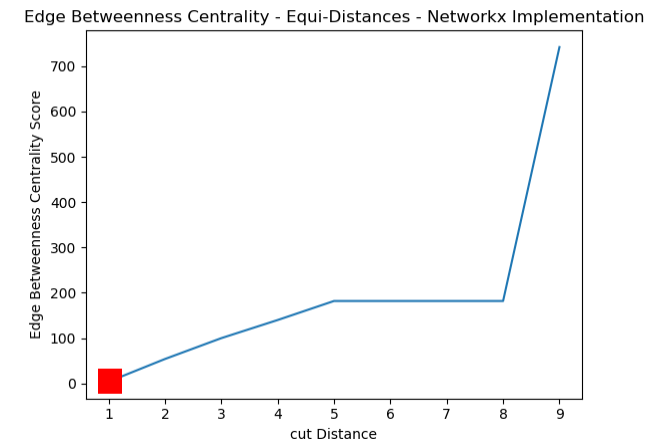
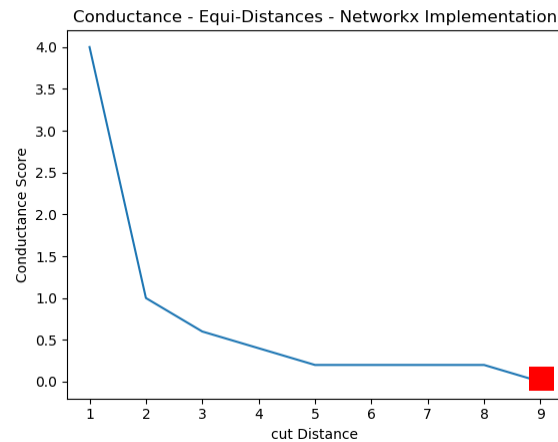
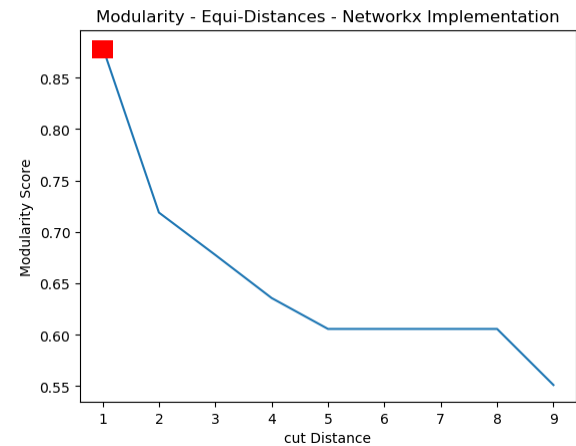
Results

- Equi-Distant Intervals



Results

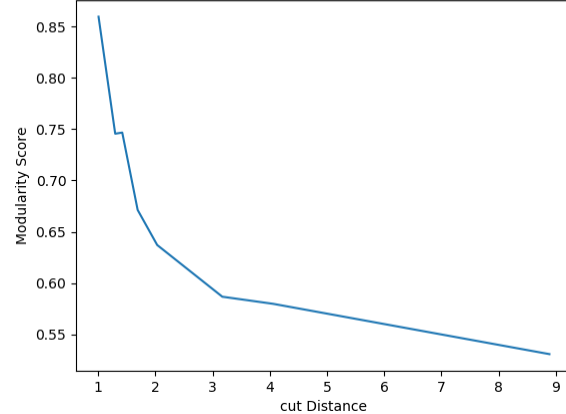
- Equi-Distant Intervals



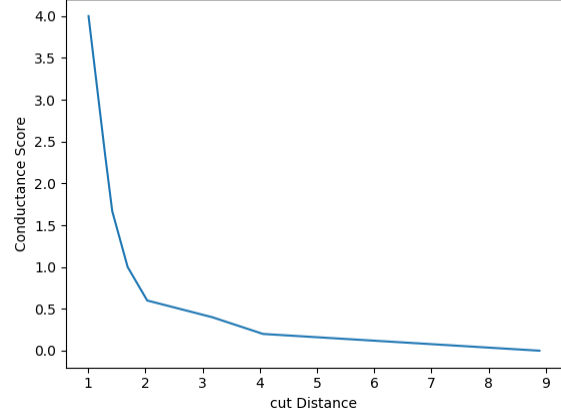
Results

- Intersect-Distant Intervals

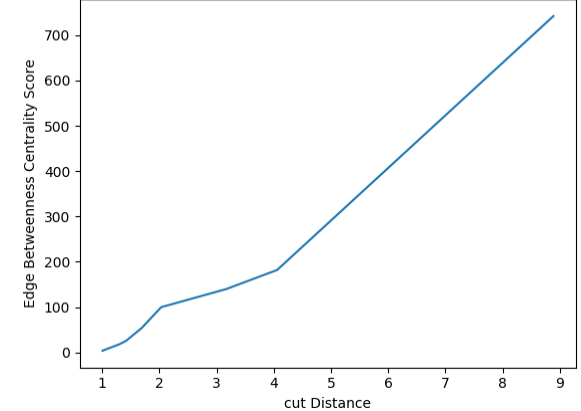
Modularity - Intersect-Distances - Networkx Implementation



Conductance - Intersect-Distances - Networkx Implementation



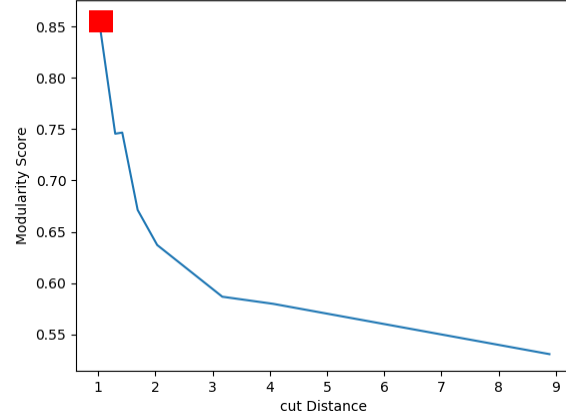
Edge Betweenness Centrality - Intersect-Distances - Networkx Implementation



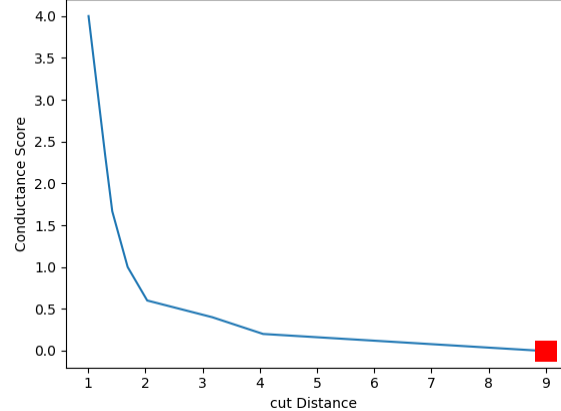
Results

- Intersect-Distant Intervals

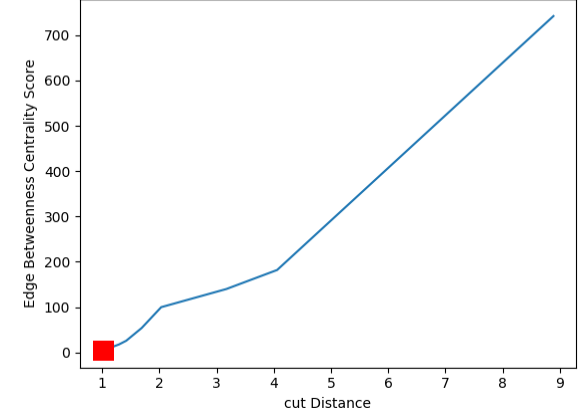
Modularity - Intersect-Distances - Networkx Implementation



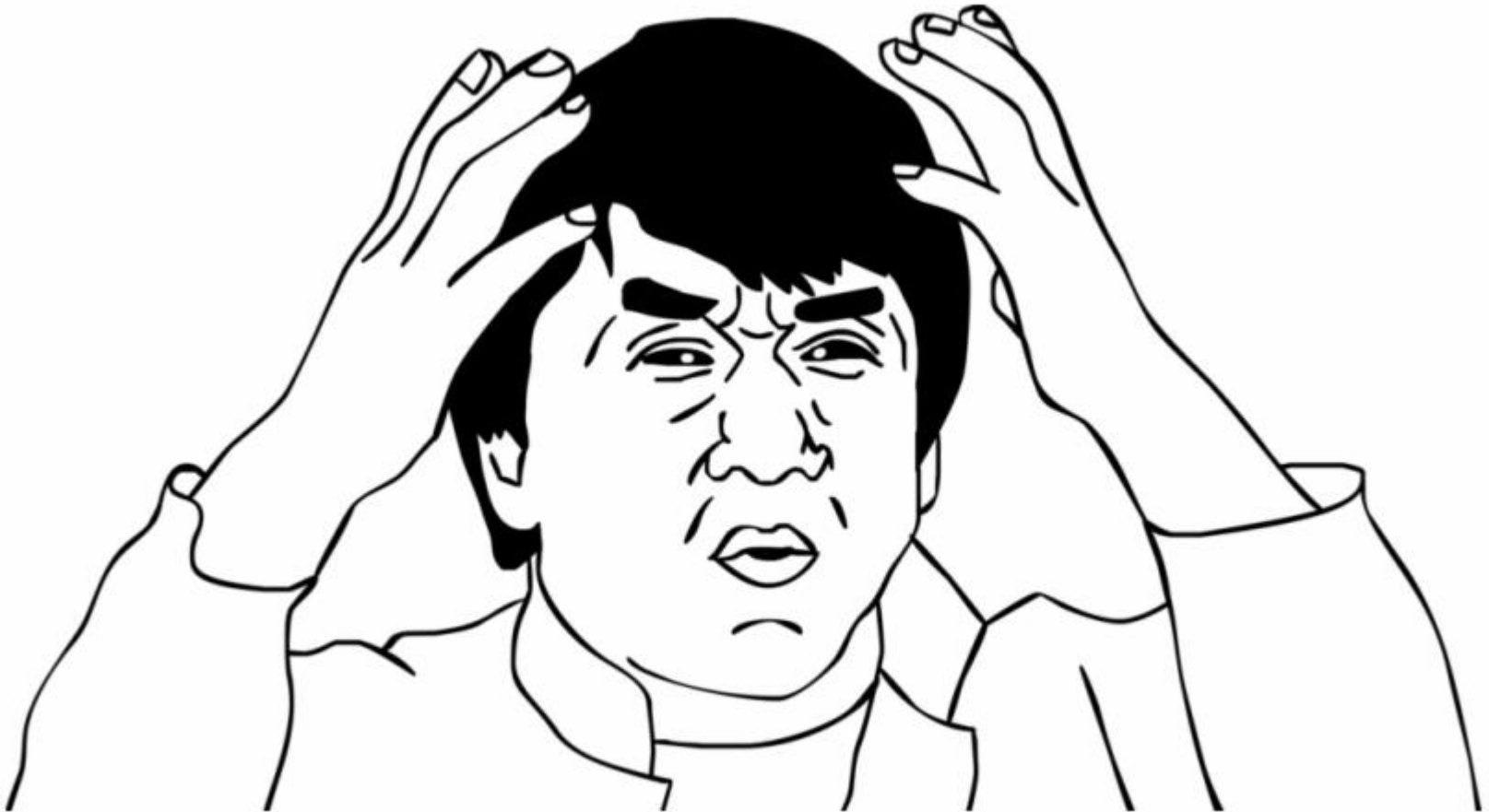
Conductance - Intersect-Distances - Networkx Implementation



Edge Betweenness Centrality - Intersect-Distances - Networkx Implementation

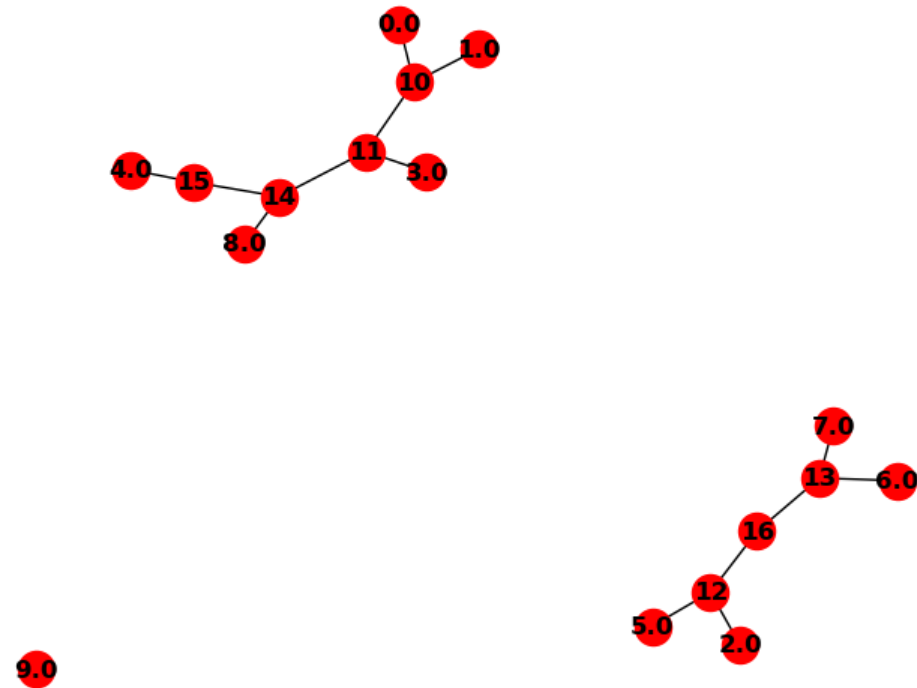
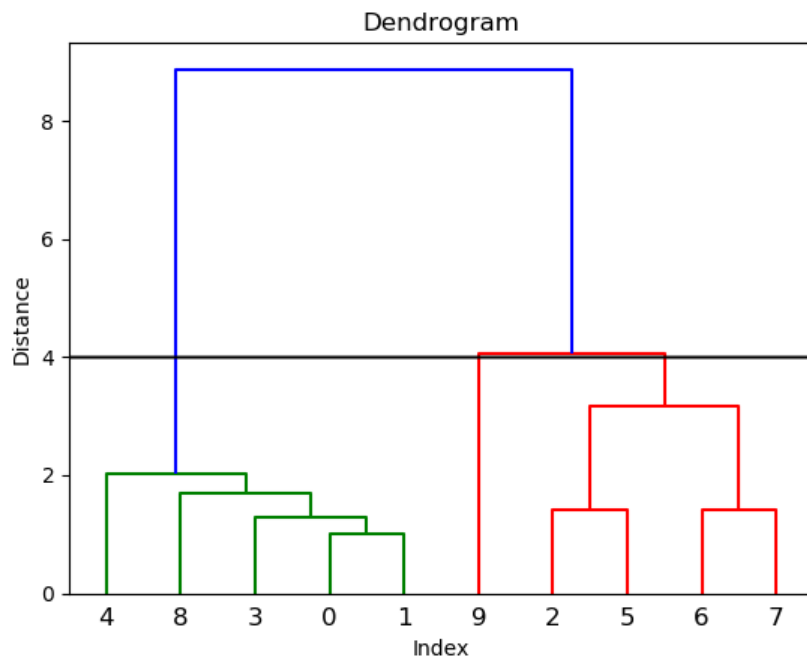


Results



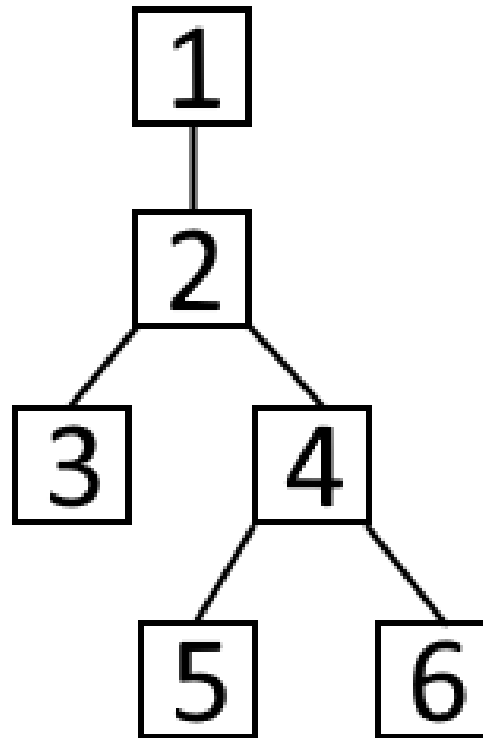
Results

- Eyeballing



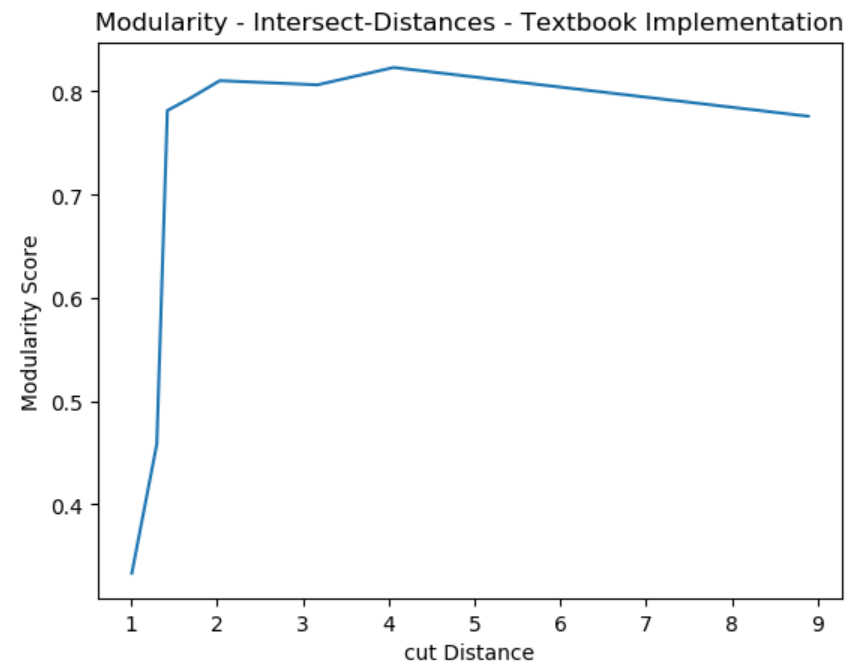
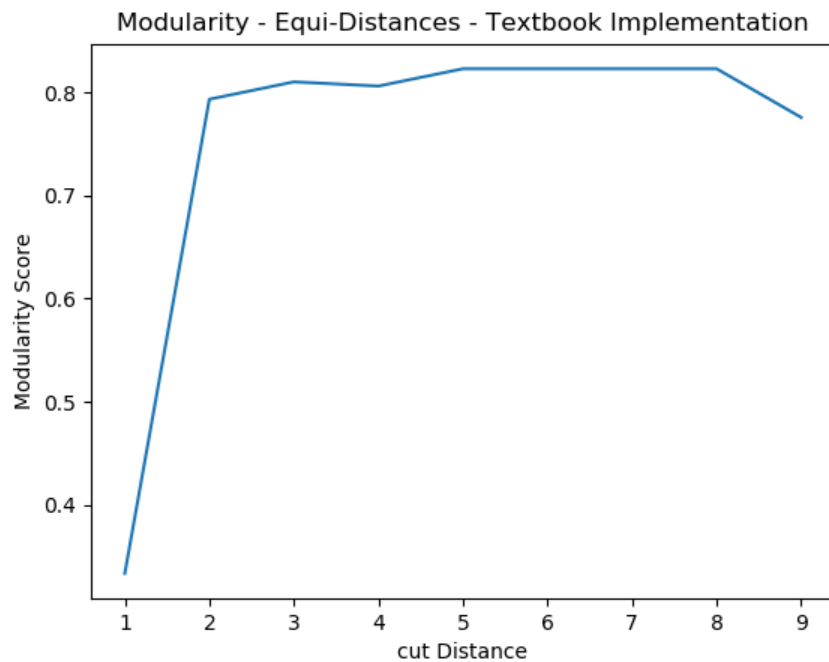
Results

- Modularity



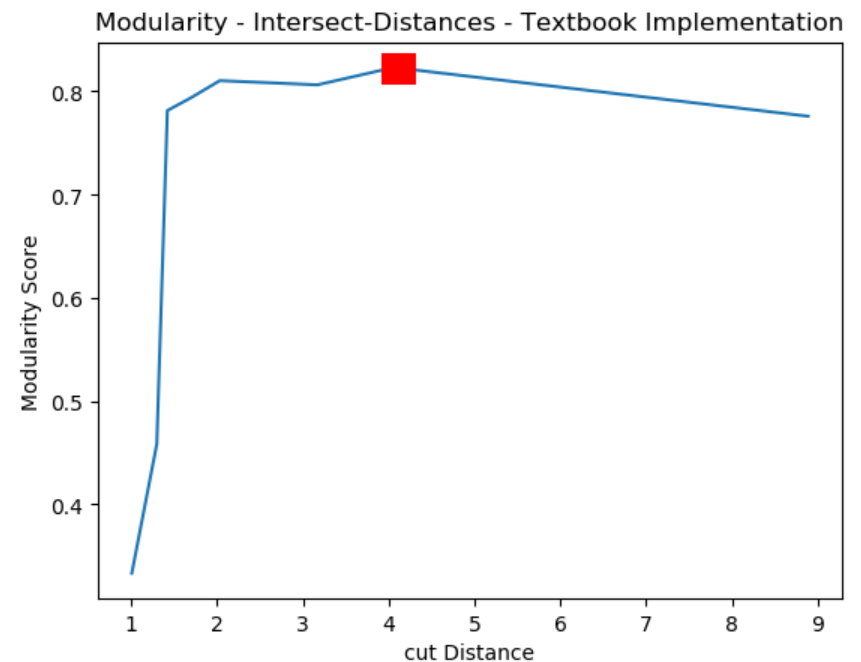
Results

- Modularity



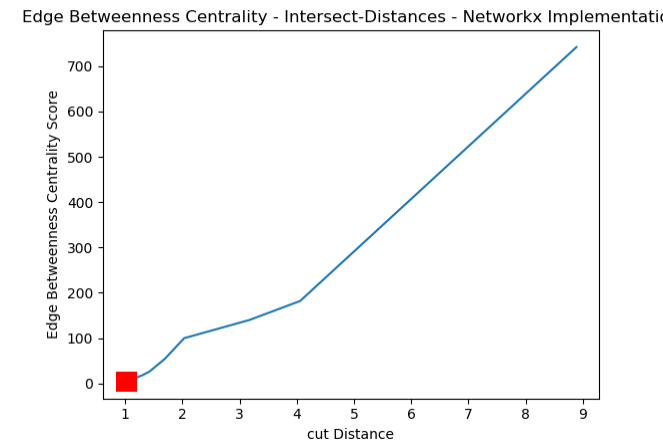
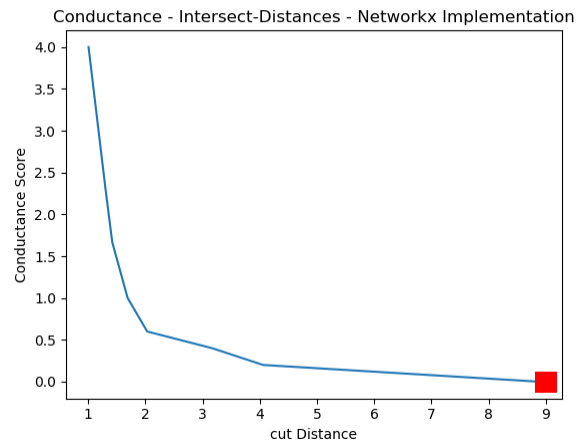
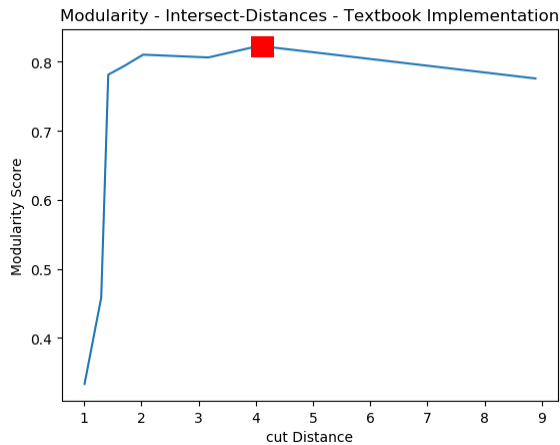
Results

- Modularity



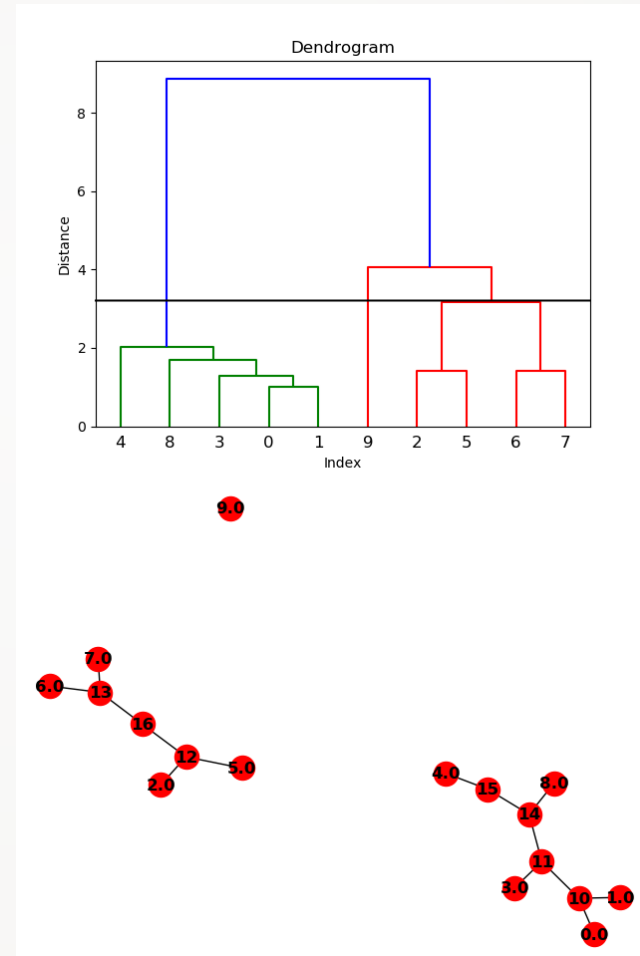
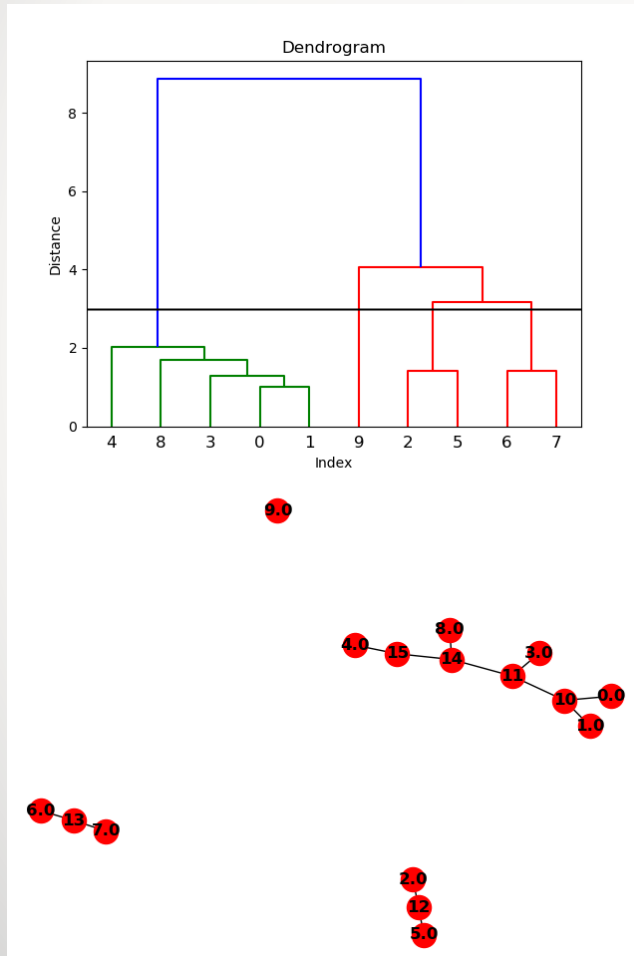
Results

- Intersect-Distant Intervals



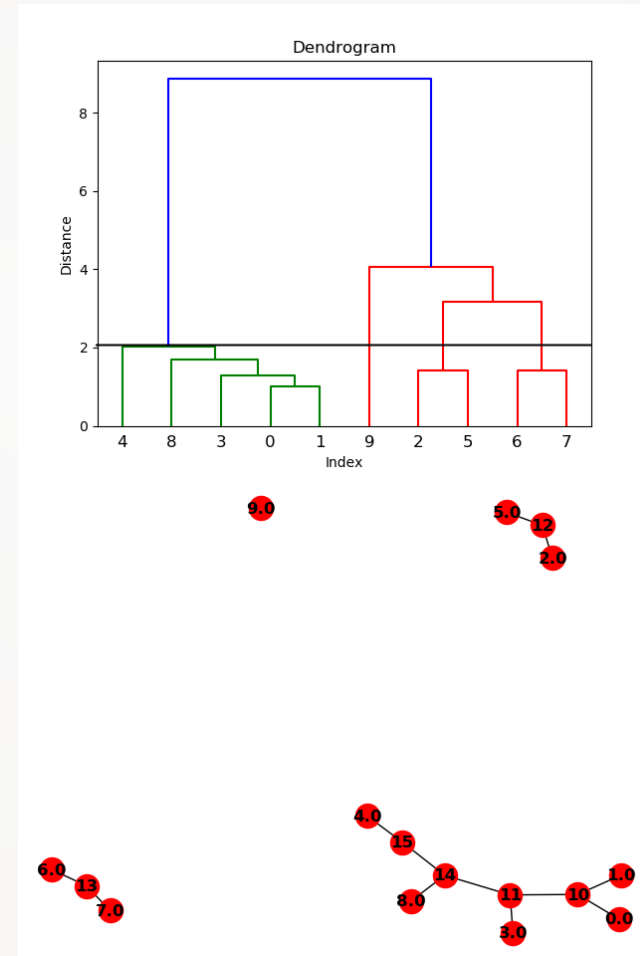
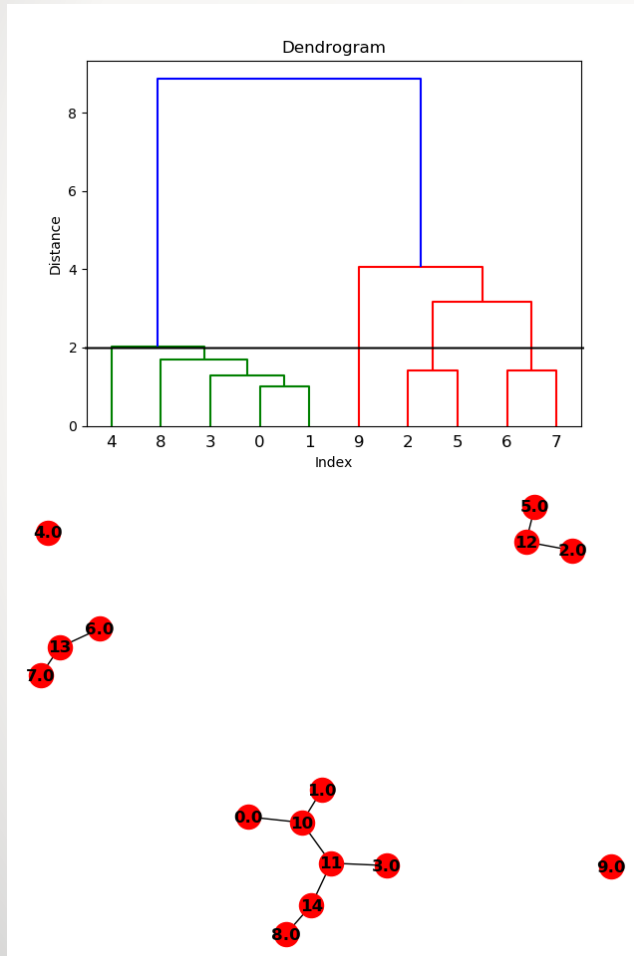
Cut Intervals

- Equi at 3 vs Intersect-Distant at 3.1 Intervals



Cut Intervals

- Equi at 2 vs Intersect-Distant at 2.03 Intervals



Vielen Dank für Eure Aufmerksamkeit

