OBJECTIVES

Similarity Measures

- SimRank

AMRITA VISHWA VIDYAPEETHAM | AHEAD Online

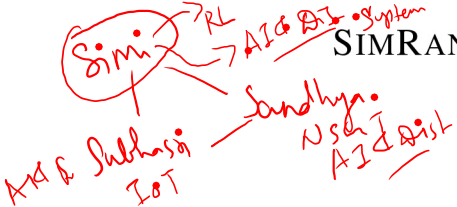
SIMRANK

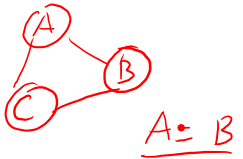
BASIC NOTION

- SimRank is a general algorithm that determines only the similarity of structural context
- Two objects are similar if they are referenced by similar objects
- Could be applied to any domain with relevant relations

SimRank Algorithm

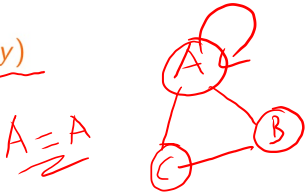
SimRank





- SimRank is a link analysis algorithm that works on a graph **G** to measure the similarity between two vertices **u** and **v** in the graph.
- The score $s(u,v) \in [0,1]$, which is equal to 1 if $u = v$.
- The SimRank iterates on the similarity index of the neighbors of u and v itself.

$$s(u,v) = \frac{C}{|\Gamma(u)||\Gamma(v)|} \sum_{x \in \Gamma(u)} \sum_{y \in \Gamma(v)} s(x,y)$$



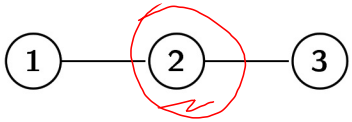
SIMARANK ..

RECURSIVE FORMULA

$$s(u,v) = \frac{C}{|\Gamma(u)||\Gamma(v)|} \sum_{i=1}^{|\Gamma(u)|} \sum_{j=1}^{|\Gamma(v)|} s(\Gamma_i(u), \Gamma_j(v))$$

- $\Gamma(u)$ is neighbors of u
- C is important factor typically in the range 0.6 - 0.9

EXAMPLE: PATH GRAPH



$|\Gamma(1)| = |\{2\}| = 1$, $|\Gamma(2)| = |\{1, 3\}| = 2$, $|\Gamma(3)| = |\{2\}| = 1$

Take $C = 0.6$

ITERATION 0

$S(1,1) = 1,\ S(2,2) = 1,\ S(3,3) = 1$

$S(1,2) = 0,\ S(1,3) = 0,\ S(2,3) = 0$

ITERATION 1

$$S(1,2) = \frac{0.6}{1\times 2}(S(2,1)^{0} + S(2,3)^{0}) = 0$$

$$S(1,3) = \frac{0.6}{1\times 1}(S(2,2)^{1}) = 0.6$$

$$S(2,3) = \frac{0.6}{2\times 1}(S(1,2) + S(1,3)^{0.6}) = 0.18$$

ITERATION 2

$$S(1,2) = \frac{0.6}{1\times 2}(S(2,1)^{0} + S(2,3)^{0}) = 0$$

$$S(1,3) = \frac{0.6}{1\times 1}(S(2,2)^{1}) = 0.6$$

$$S(2,3) = \frac{0.6}{2\times 1}(S(1,2) + S(1,3)^{0.6}) = 0.18$$

ITERATION 3

$$S(1,2) = \frac{0.6}{1\times 2}(S(2,1)^{0} + S(2,3)^{0.18}) = 0.054$$

$$S(1,3) = \frac{0.6}{1\times 1}(S(2,2)^{1}) = 0.6$$

$$S(2,3) = \frac{0.6}{2\times 1}(S(1,2) + S(1,3)^{0.6}) = 0.18$$

AMRITA VISHWA VIDYAPEETHAM AHEAD Online

SUMMARY

SIMRANK

- Recursive Algorithm
- Path Graph Example

AMRITA VIDYAPEETHAM | AHEAD Online

References

- Simrank: Similarity analysis explanation and python implementation from scratch https://towardsdatascience.com/simrank-similarity-analysis-1d8d5a18766a.
- Jeh, G., and Widom, J.
  Simrank: a measure of structural-context similarity.
  In *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining* (2002), pp. 538–543.
- Zinoviev, D.
  Complex network analysis in Python:
  Recognize-construct-visualize-analyze-interpret.
  Pragmatic Bookshelf, 2018.

Objectives

Similarity Measures

- Simrank Example
- Using NetworkX

AMRITA VISHWA VIDYAPEETHAM | AHEAD Online

SIMARANK ..

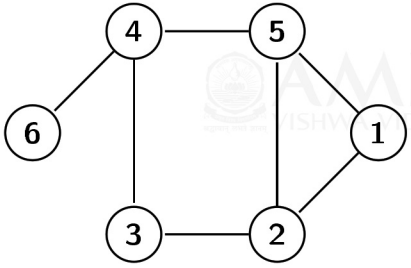
RECURSIVE FORMULA

$$s(u,v) = \frac{C}{|\Gamma(u)||\Gamma(v)|} \sum_{i=1}^{|\Gamma(u)|} \sum_{j=1}^{|\Gamma(v)|} s(\Gamma_i(u), \Gamma_j(v))$$

- $\Gamma(u)$ is neighbors of u
- C is a constant typically in the range 0.6 - 0.8

EXAMPLE II (1/3)

$|\Gamma(1)| = |\{2, 5\}| = 2,\ |\Gamma(2)| = |\{1, 3, 5\}| = 3$
$|\Gamma(3)| = |\{2, 4\}| = 2,\ |\Gamma(4)| = |\{3, 5, 6\}| = 3$
$|\Gamma(5)| = |\{1, 2, 4\}| = 3,\ |\Gamma(6)| = |\{4\}| = 1$



ITERATION 0

$S(1,1) = 1,\ S(2,2) = 1,\ S(3,3) = 1,\ S(4,4) = 1,\ S(5,5) = 1,$
$S(6,6) = 1$

$S(1,2) = 0,\ S(1,4) = 0, \ldots,\ S(5,6) = 0$

ITERATION 1

$$S(1,2) = \frac{0.6}{2\times3}(S(2,1)^{0} + S(2,3)^{0} + S(2,5)^{0} + S(5,1)^{0} + S(5,3)^{0} + S(5,5)^{1}) = 0.01$$

$$S(1,3) = \frac{0.6}{2\times2}(S(2,2)^{1} + S(2,4) + S(5,2) + S(5,4)) = 0.15$$

$$S(1,4) = \frac{0.6}{2\times3}(S(2,3) + S(2,5) + S(2,6) + S(5,3) + S(5,5)^{1} + S(5,6)) = 0.01$$

$$S(1,5) = \frac{0.6}{2\times3}(S(2,1) + S(2,2)^{1} + S(2,4) + S(5,1) + S(5,2) + S(5,4)) = 0.01$$

$$S(1,6) = \frac{0.6}{2\times2}(S(2,4) + S(5,4)) = 0$$

EXAMPLE II (2/3)

ITERATION 1 ..

$$S(2,3) = \frac{0.6}{3\times2}(S(1,2)+S(1,4)+S(3,2)+S(3,4)+S(5,2)+S(5,4)) = 0$$

$$S(2,4) = \frac{0.6}{3\times3}(S(1,3)+S(1,5)+S(1,6)+S(\overset{1}{3},3)+S(3,5)+S(3,6)+S(5,3)+S(\overset{1}{5},5)+S(5,6)) = 0.133$$

$$S(2,5) = \frac{0.6}{3\times3}(S(\overset{1}{1},1)+S(1,2)+S(1,4)+(S(3,1)+S(3,2)+S(3,4)+(S(5,1)+S(5,2)+S(5,4)) = 0.066$$

$$S(2,6) = \frac{0.6}{3\times3}(S(1,4)+S(3,4)+S(5,4) = 0$$

$$S(3,4) = \frac{0.6}{2\times3}(S(2,3)+S(2,5)+S(2,6)+S(4,3)+S(4,5)+S(4,6)) = 0$$

$$S(3,5) = \frac{0.6}{2\times3}(S(2,1)+S(\overset{1}{2},2)+S(2,4)+(S(4,1)+S(4,2)+S(\overset{1}{4},4)) = 0.2$$

$$S(3,6) = \frac{0.6}{2\times1}(S(2,4)+S(\overset{1}{4},4)) = 0.3$$

$$S(4,5) = \frac{0.6}{2\times3}(S(3,1)+S(3,2)+S(3,4)+(S(5,1)+S(5,2)+S(5,4)) = 0$$

$$S(4,6) = \frac{0.6}{3\times1}(S(3,4)+S(5,4)+S(6,4) = 0$$

$$S(5,6) = \frac{0.6}{3\times1}(S(1,4)+S(2,4)+S(\overset{1}{4},4) = 0.2$$

AMRITA VISHWA VIDYAPEETHAM | AHEAD Online

EXAMPLE II (3/3)

ITERATION 2

$S(1,2) = \frac{0.6}{2\times3}(\overset{0.01}{S(2,1)} + \overset{0}{S(2,3)} + \overset{0.066}{S(2,5)} + \overset{0.01}{S(5,1)} + \overset{0.2}{S(5,3)} + \overset{1}{S(5,5)}) =$

0.128

$S(1,3) = \frac{0.6}{2\times2}(\overset{1}{S(2,2)} + \overset{0.133}{S(2,4)} + \overset{0.066}{S(5,2)} + \overset{0}{S(5,4)}) = 0.179$

$S(1,4) = \frac{0.6}{2\times3}(\overset{0}{S(2,3)} + \overset{0.066}{S(2,5)} + \overset{0}{S(2,6)} + \overset{0.2}{S(5,3)} + \overset{1}{S(5,5)} + \overset{0.2}{S(5,6)}) =$

0.1466

$S(1,5) = \frac{0.6}{2\times1}(\overset{0.01}{S(2,1)} + \overset{1}{S(2,2)} + \overset{0.13}{S(2,4)} + \overset{0.01}{S(5,1)} + \overset{0.066}{S(5,2)} + \overset{0}{S(5,4)}) =$

0.1216

$S(1,6) = \frac{0.6}{2\times2}(\overset{0.133}{S(2,4)} + \overset{0}{S(5,4)}) = 0.039$

$S(2,3) = \frac{0.6}{3\times2}(\overset{0.01}{S(1,2)} + \overset{0.01}{S(1,4)} + \overset{0}{S(3,2)} + \overset{0}{S(3,4)} + \overset{0.066}{S(5,2)} + \overset{0}{S(5,4)}) =$

0.086

$S(2,4) = \frac{0.6}{3\times3}(\overset{0.15}{S(1,3)} + \overset{0.01}{S(1,5)} + \overset{0}{S(1,6)} + \overset{1}{S(3,3)} + \overset{0.2}{S(3,5)} + \overset{0.3}{S(3,6)} +$
$\overset{0.2}{S(5,3)} + \overset{1}{S(5,5)} + \overset{0.2}{S(5,6)}) = 0.204$

$S(2,5) = \frac{0.6}{3\times3}(\overset{1}{S(1,1)} + \overset{0.01}{S(1,2)} + \overset{0.01}{S(1,4)} + (\overset{0.15}{S(3,1)} + \overset{0}{S(3,2)} +$
$\overset{0}{S(3,4)} + (\overset{0.01}{S(5,1)} + \overset{0.066}{S(5,2)} + \overset{0}{S(5,4)}) = 0.083$

$S(2,6) = \frac{0.6}{3\times3}(\overset{0.01}{S(1,4)} + \overset{0}{S(3,4)} + \overset{0}{S(5,4)} = 0.0006$

$S(3,4) =$

$\frac{0.6}{2\times3}(\overset{0}{S(2,3)} + \overset{0.066}{S(2,5)} + \overset{0}{S(2,6)} + \overset{0}{S(4,3)} + \overset{0}{S(4,5)} + \overset{0}{S(4,6)}) = 0.004$

$S(3,5) = \frac{0.6}{2\times3}(\overset{0.01}{S(2,1)} + \overset{1}{S(2,2)} + \overset{0.133}{S(2,4)} + (\overset{0.01}{S(4,1)} + \overset{0.133}{S(4,2)} +$
$\overset{1}{S(4,4)}) = 0.2286$

$S(3,6) = \frac{0.6}{2\times1}(\overset{0.133}{S(2,4)} + \overset{1}{S(4,4)}) = 0.34$

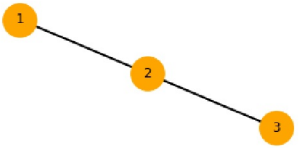
$S(4,5) = \frac{0.6}{2\times3}(\overset{0.15}{S(3,1)} + \overset{0}{S(3,2)} + \overset{0}{S(3,4)} + (\overset{0.01}{S(5,1)} + \overset{0.066}{S(5,2)} +$
$\overset{0}{S(5,4)}) = 0.0266$

$S(4,6) = \frac{0.6}{3\times1}(\overset{0}{S(3,4)} + \overset{0}{S(5,4)} + \overset{0}{S(6,4)} = 0$

$S(5,6) = \frac{0.6}{3\times1}(\overset{0.01}{S(1,4)} + \overset{0.133}{S(2,4)} + \overset{1}{S(4,4)} = 0.2286$

AMRITA VISHWA VIDYAPEETHAM India

USING NETWORKX (1/2)



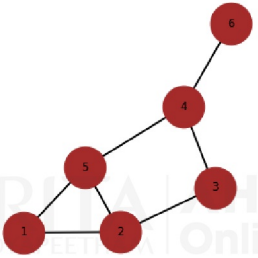

```
import networkx as nx
G = nx.path_graph(3)
G1=nx.convert_node_labels_to_integers(G, first_label=1)
sim=nx.simrank_similarity(G1)
sim
{1: {1: 1, 2: 0.0, 3: 0.9},
 2: {1: 0.0, 2: 1, 3: 0.0},
 3: {1: 0.9, 2: 0.0, 3: 1}}
```

USING NETWORKX (2/2)

SIMARANK IN MATRIX FORM

```
import numpy as np
np.array([[sim[u][v] for v in G1] for u in G1])
array([[1.  , 0.  , 0.9],
       [0.  , 1.  , 0. ],
       [0.9, 0.  , 1. ]])
```

EXAMPLE 2 (1/2)




```
import networkx as nx
import matplotlib.pyplot as plt
G2 = nx.Graph([(1, 5),(1, 2),(2,5),(2,3),(3,4),(4,5),(4,6)])
```

AMRITA AHEAD Online
VISHWA VIDYAPEETHAM

EXAMPLE 2 (2/2)

```
sim=nx.simrank_similarity(G2, importance_factor=0.6)
import numpy as np
x=np.array([[sim[u][v] for v in G2] for u in G2])
np.around(x, decimals=2, out=None)
array([[1.  , 0.18, 0.19, 0.22, 0.18, 0.09],
       [0.18, 1.  , 0.14, 0.29, 0.06, 0.29],
       [0.19, 0.14, 1.  , 0.07, 0.25, 0.06],
       [0.22, 0.29, 0.07, 1.  , 0.04, 0.37],
       [0.18, 0.06, 0.25, 0.04, 1.  , 0.02],
       [0.09, 0.29, 0.06, 0.37, 0.02, 1.  ]])
```

Summary

Simarank Examples

- Using Recursive Formula
- NetworkX

AMRITA VISHWA VIDYAPEETHAM | AHEAD Online

Objectives

Similarity Measures

- Assortative Mixing

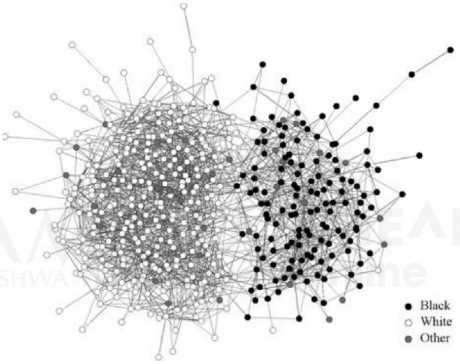
AMRITA VISHWA VIDYAPEETHAM | AHEAD Online

Assortative Mixing

Basics

- Assortative Mixing is a measure of the tendency to associate with others whom they perceive as being similar in some way
- A network is considered as assortative if a significant fraction of the edges run between vertices are of the same type.
- Commonly applied in social networks and undirected multigraphs

Assortative Mixing



- Friendship network at a US high school. The vertices in this network represent 470 students at a US high school (ages 14 to 18 years). The vertices are color coded by race as indicated in the key. Data from the National Longitudinal Study of Adolescent Health [1].

Assortative Mixing..

- The measure Assortativity is the number of edges that run between vertices of the same type minus the number of such edges we would expect to find if the configuration model is assumed.
- Model assumption: edges were positioned at random while preserving the vertex degrees

Edges between vertices

- The number of edges that run between vertices of the same type is:

$$\sum_{(i,j)\in E} \delta(c_i, c_j) = \frac{1}{2} \sum_{i,j} a_{i,j} \delta(c_i, c_j)$$

- c_i the type of vertex i, $a_{i,j}$ is the actual number of edges between i and j
- $\delta(c_i, c_j)$ =1 if $c_i = c_j$ and $\delta(c_i, c_j) = 0$ otherwise.

Assortative Mixing ..

Expected Number of Edges

- The expected number of edges that run between vertices of the same type is:

$$\frac{1}{2}\sum_{i,j}\frac{k_i k_j}{2m}\delta(c_i, c_j)$$

- k_i and k_j are the degrees of i and j
- m is the number of edges of the graph
- $\frac{k_i k_j}{2m}$ is the expected number of edges between vertices i and j in the configuration model assumption

AMRITA VISHWA VIDYAPEETHAM | AHEAD Online

Assortative Mixing ..

Modularity

- The probability that this edge goes to node j is $\frac{k_i k_j}{2m}$, since the number of edges attached to j is k_j and the total number of edge ends in the network is $2m$ (the sum of all node degrees).
- Since node i has k_i edges attached to it, the expected number of edges between i and j is $\frac{k_i k_j}{2m}$.
- Hence the difference between the actual and expected number of edges connecting nodes of the same type, expressed as a fraction with respect to the total number of edges m, is called modularity, and given by:

$$Q = \frac{1}{2m} \sum_{i,j} \left(a_{i,j} - \frac{k_i k_j}{2m} \right) \delta(c_i, c_j)$$

Summary

Similarity Measures

- Assortative Mixing

AMRITA VISHWA VIDYAPEETHAM | AHEAD Online

References

Newman, M.
Networks.
Oxford university press, 2018.

Zinoviev, D.
Complex network analysis in Python: Recognize-construct-visualize-analyze-interpret.
Pragmatic Bookshelf, 2018.

AMRITA VISHWA VIDYAPEETHAM | AHEAD Online

Objectives

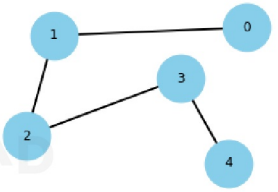
Similarity Measures

- Assortative Mixing: Examples

AMRITA VISHWA VIDYAPEETHAM | AHEAD Online

EXAMPLE 1: PATH GRAPH

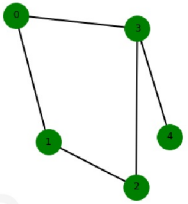
```
import networkx as nx
import matplotlib.pyplot as plt
G = nx.path_graph(5)
pos=nx.random_layout(G)
fig, ax = plt.subplots(figsize=(5,5))
nx.draw (G, ax=ax, pos=pos)
```



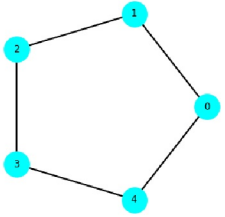

```
am=nx.degree_assortativity_coefficient(G)
am
-0.3333333333333333
```

EXAMPLES: 2 & 3

```
G = nx.path_graph(5)
G.add_edges_from([(0,3)])
am=nx.degree_assortativity_coefficient(G)
-0.6666666666666691
```



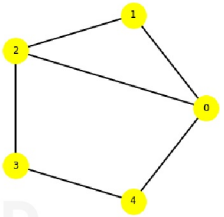

```
G = nx.path_graph(5)
G.add_edges_from([(0,4)])
am=nx.degree_assortativity_coefficient(G)
```



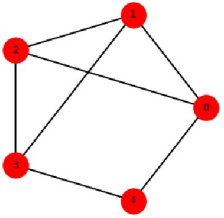
AMRITA VISHWA VIDYAPEETHAM | AHEAD Online

EXAMPLES: 3 & 4

```
G = nx.path_graph(5)
G.add_edges_from([(0,3)])
G.add_edges_from([(0,2)])
am=nx.degree_assortativity_coefficient(G)
-0.3333333333333348
```



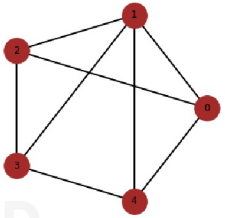

```
G = nx.path_graph(5)
G.add_edges_from([(0,4)])
G.add_edges_from([(0,2)])
G.add_edges_from([(1,3)])
am=nx.degree_assortativity_coefficient(G)
-0.16666666666667615
```



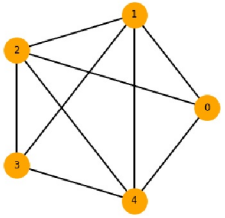
AMRITA VISHWA VIDYAPEETHAM | AHEAD Online

EXAMPLES: 5 & 6

```
G = nx.path_graph(5)
G.add_edges_from([(0,4),(0,2)])
G.add_edges_from([(1,3),(1,4)])
am=nx.degree_assortativity_coefficient(G)
-0.3333333333333333
```



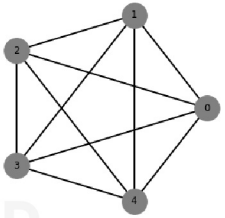

```
G = nx.path_graph(5)
G.add_edges_from([(0,4),(0,2)])
G.add_edges_from([(1,3),(1,4),(2,4)])
am=nx.degree_assortativity_coefficient(G)
-0.499999999999998
```



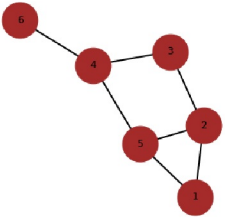
AMRITA VISHWA VIDYAPEETHAM | AHEAD Online

EXAMPLES: 7 & 8

```
G = nx.path_graph(5)
G.add_edges_from([(0,4),(0,2)])
G.add_edges_from([(1,3),(1,4)])
am=nx.degree_assortativity_coefficient(G)
NaN
```

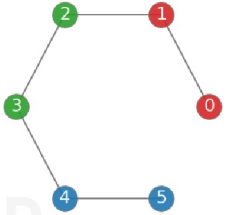



```
G= nx.Graph([(1, 5),(1, 2),(2,5),(2,3),
             (3,4),(4,5),(4,6)])
am=nx.degree_assortativity_coefficient(G)
-0.47368421052632204
```

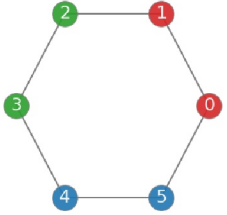


EXAMPLES: 9 & 10

```
G = nx.Graph()
G.add_nodes_from([0, 1], color="red")
G.add_nodes_from([2, 3], color="green")
G.add_nodes_from([4,5], color="blue")
print(nx.attribute_
assortativity_coefficient(G, "color"))
0.3939393939393941
```




```
G = nx.Graph()
print(nx.attribute_assortativity_
coefficient(G, "color"))
0.24999999999999992
```



Summary

Similarity Measures

- Assortative Mixing: NetworkX Examples

AMRITA VISHWA VIDYAPEETHAM | AHEAD Online

References

- Simrank: Similarity analysis explanation and python implementation from scratch https://towardsdatascience.com/simrank-similarity-analysis-1d8d5a18766a.
- Jeh, G., and Widom, J.
  Simrank: a measure of structural-context similarity.
  In *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining* (2002), pp. 538–543.
- Zinoviev, D.
  Complex network analysis in Python:
  Recognize-construct-visualize-analyze-interpret.
  Pragmatic Bookshelf, 2018.