DLI Accelerated Data Science Teaching Kit

Lecture 17.3 - Centralities: Degree, Betweenness, Clustering Coefficient

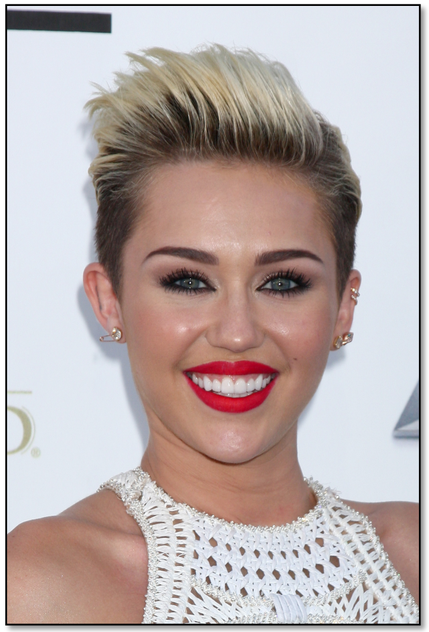
Centrality

= “Importance”

Why Node Centrality?

What can we do if we can rank all the nodes in a graph (e.g., Facebook, LinkedIn, Twitter)?

- Find celebrities or influential people in a social network (Twitter)
- Find “gatekeepers” who connect communities (headhunters love to find them on LinkedIn)



Why Node Centrality?

Helps graph analysis, visualization, understanding, e.g.,

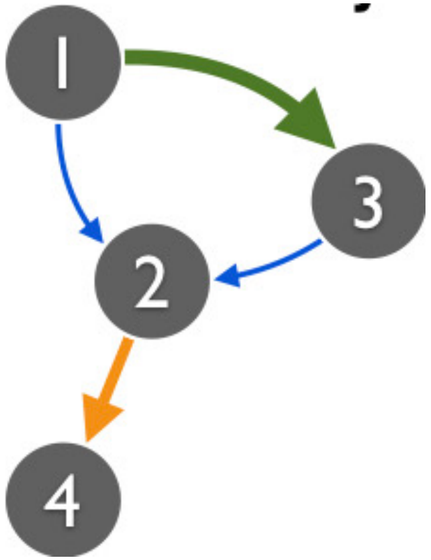
- Let us rank nodes, group or study them by centrality
- Only show subgraph formed by the top 100 nodes, out of the millions in the full graph
  - Similar to google search results (ranked, and they only show you 10 per page)
- Most graph analysis packages already have centrality algorithms implemented. Use them!

Can also compute edge centrality. Here we focus on node centrality.

Degree Centrality (Easiest)

Degree = number of neighbors

- For directed graphs
  - In degree = No. of incoming edges
  - Out degree = No. of outgoing edges
- For undirected graphs, only degree is defined.
- Algorithms?
  - Sequential scan through edge list
  - What about for a graph stored in SQLite?



1, 2
1, 3
2, 4
3, 2

Computing Degrees using SQL

Recall simplest way to store a graph in SQLite:

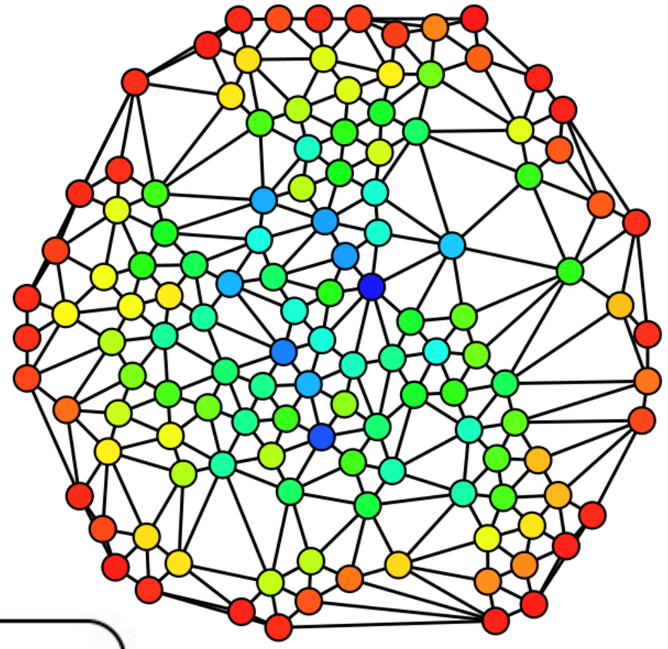
```
edges(source_id, target_id)
```

1, 2
1, 3
2, 4
3, 2

1. If slow, first create index for each column
2. Use group by statement to find out degrees

```
select count(*) from edges group by source_id;
```

Betweenness Centrality



High betweenness = "gatekeeper"

Betweenness of a node v

$$= \sum_{s \neq v \neq t \in V} \frac{\sigma_{st}(v)}{\sigma_{st}}$$

Number of shortest paths between s and t that **goes through v**

Number of shortest paths between s and t

= how often a node serves as the "bridge" that connects two other nodes.

(Local) Clustering Coefficient

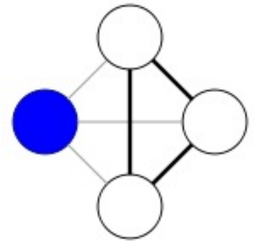
A node's clustering coefficient is a measure of how close the node's neighbors are from forming a clique.

1 = Neighbors form a clique

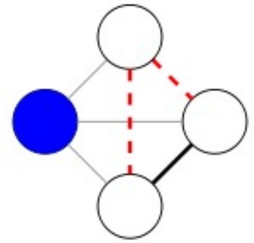
0 = No edges among neighbors

(Assuming undirected graph)

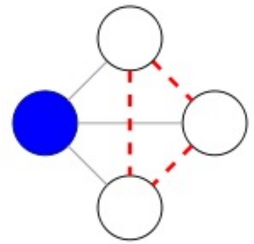
"Local" means it's for a node; can also compute a graph's "global" coefficient



c = 1



c = 1/3



c = 0

Image source: http://en.wikipedia.org/wiki/Clustering_coefficient

Computing Clustering Coefficients...

Requires triangle counting

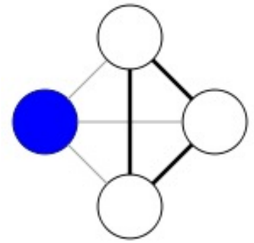
Real social networks have a lot of triangles

Friends of friends are friends

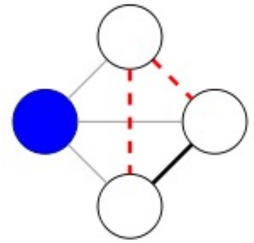
Triangles are expensive to compute

(neighborhood intersections; several approx. algos)

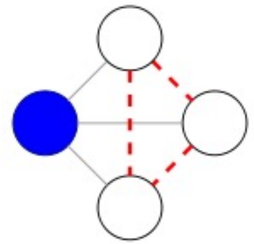
Can we do that quickly?



c = 1



c = 1/3



c = 0

Algorithm details:
Faster Clustering Coefficient Using Vertex Covers
http://www.cc.gatech.edu/~ogreen3/_docs/2013VertexCoverClusteringCoefficients.pdf

Super Fast Triangle Counting

[Tsourakakis ICDM 2008]

But: triangles are expensive to compute
(3-way join; several approx. algos)

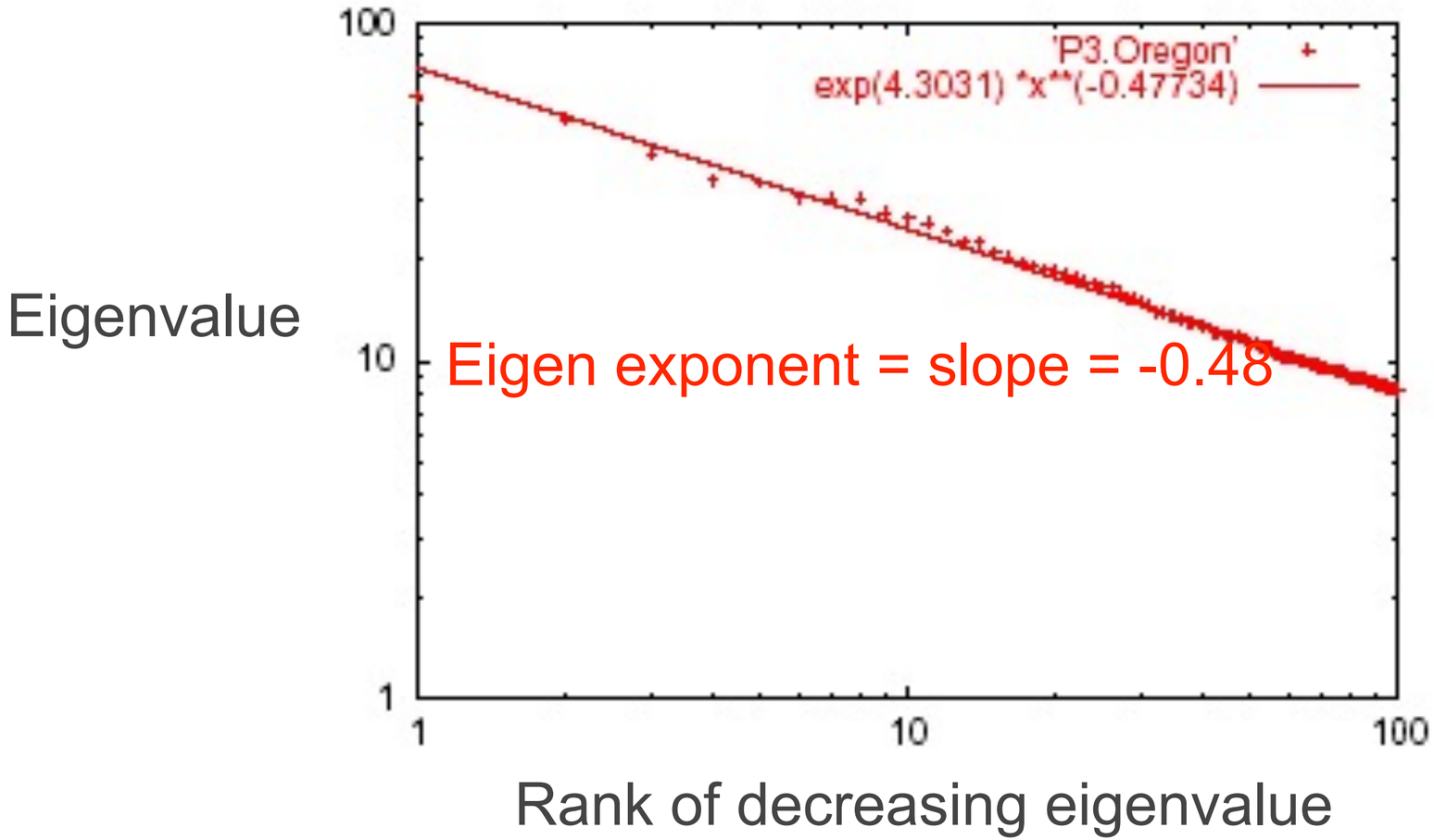
Q: Can we do that quickly?

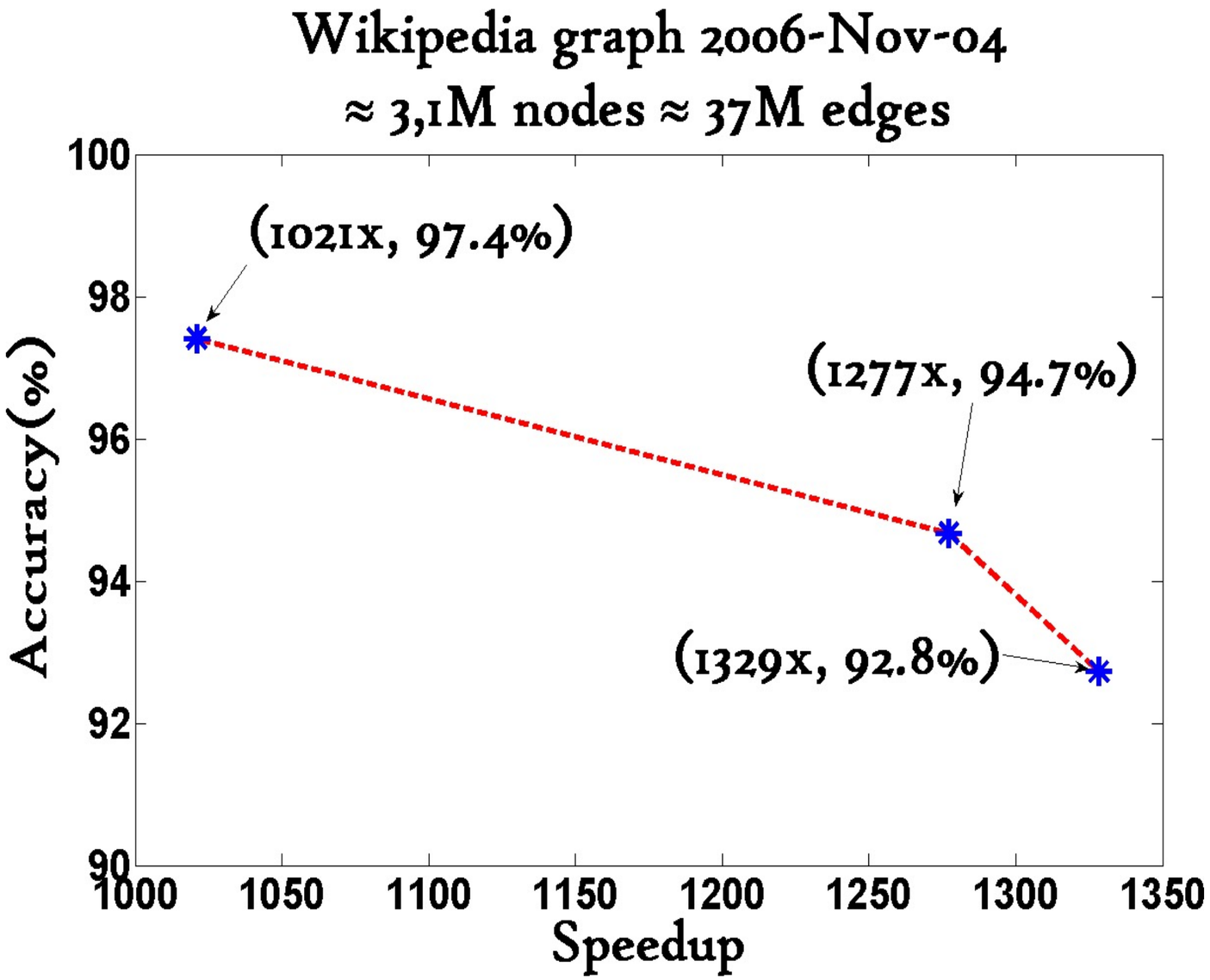
A: Yes!

#triangles = 1/6 Sum (λ_i^3)

Because of skewness, we only need top few eigenvalues!

Power Law in Eigenvalues





1000x+ speed-up, >90% accuracy

More Centrality Measures…

- Degree
- Betweenness
- Closeness, by computing
  - Shortest paths
  - “Proximity” (e.g., via random walks)
- Eigenvector
- …

DEEP
LEARNING
INSTITUTE

PRAIRIE VIEW
A&M UNIVERSITY

DLI Accelerated Data Science Teaching Kit

Thank You