Clustering Coefficient and Transitivity

A triangle is a complete subgraph of G with 3 vertices.
$\lambda(G)$ = number of triangles in G; $\lambda(v)$ is defined accordingly; $\lambda(G) = \frac{1}{3}\sum_v \lambda(v)$
A triple is a subgraph of G with 3 nodes and 2 edges; A triple is a *triple at* v if v incident with both edges.

$$\tau(v) = \binom{d(v)}{2} = \frac{d^2(v) - d(v)}{2}, \ \tau(G) = \sum_v \tau(v)$$

We define *clustering coefficient* as $c(v) = \lambda(v)/\tau(v)$.

Given $V' = \{v \in V | d(v) \geq 2\}$ we define cc of G as

$$C(G) = \frac{1}{V'} \sum_{v \in V'} c(v)$$

.

Transitivity of G is defined as

$$T(G) = \frac{3\lambda(G)}{\tau(G)}$$

Clustering Coefficient and Transitivity

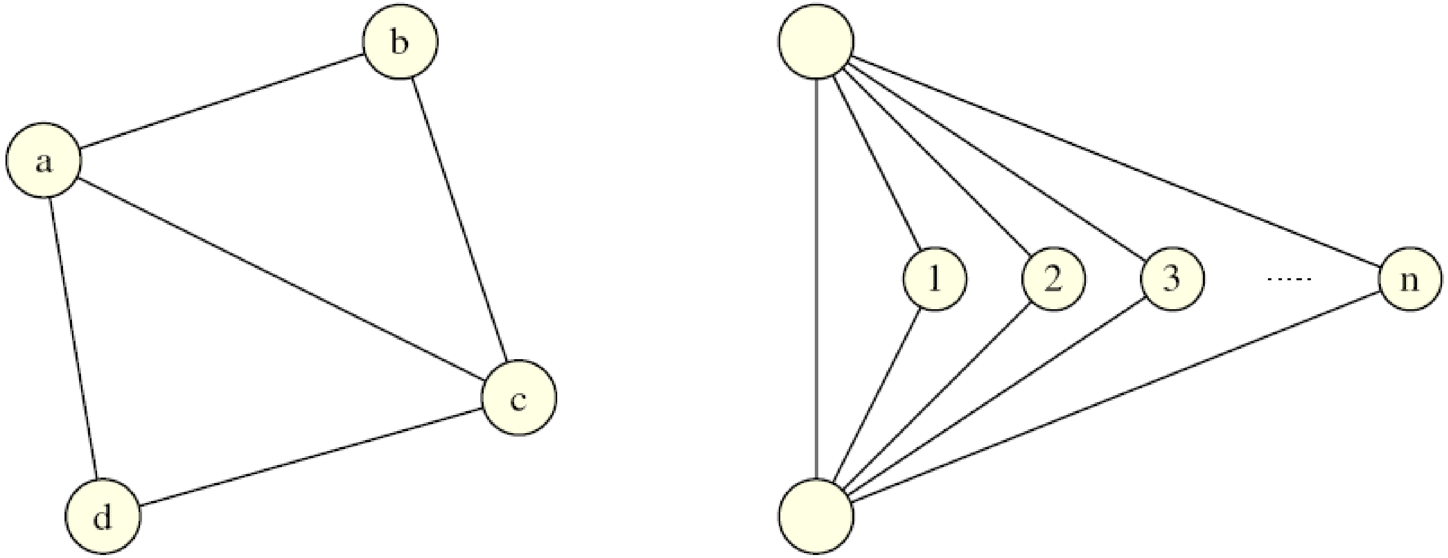

Fig. 11.2. On the left: Graph with clustering coefficients: $c(a) = c(c) = 2/3$, $c(b) = c(d) = 1$, $C(G) = \frac{1}{4}(2 + 4/3) \approx 0.83$ and transitivity $T(G) = 3 \cdot 2/8 = 0.75$. On the right: family of graphs where $T(G) \to 0$, $C(G) \to 1$ for $n \to \infty$.

[BE] "Network Analysis"

Clustering Coefficient and Transitivity

Transitivity by Bollobas and Riordan

$$T(G) = \frac{\sum_{v \in V'} \tau(v) c(v)}{\sum_{v \in V'} \tau(v)}$$

- If all nodes have the same degree then $C(G) = T(G)$
- If all clustering coefficients are equal then $C(G) = T(G)$

Computing Clustering Coefficient

Computing cc = computing triples (trivial, how?) + computing triangles
Computing triangles = $O(nd_{max}^2)$ – trivial, $O(n^{2..376})$ – mat-mat multiplication

Approximation for very large networks

$X_i \in [0, M]$ is a random var; k is number of samples; ϵ is error bound

Hoeffding inequality

$$\Pr\left(\left|\frac{1}{k}\sum_{i=1}^{k} X_i - \mathbb{E}\left[\frac{1}{k}\sum_{i=1}^{k} X_i\right]\right| \geq \epsilon\right) \leq e^{\frac{-2k\epsilon^2}{M^2}}$$

Lemma: If we consider the constant error bound then there exist algorithms that approximate the clustering coefficients for each node $c(v)$ and the transitivity $T(G)$ in time $O(n)$. The clustering coefficient $C(G)$ can be approximated in time in O(1).

Homework: [SW] "Approximating clustering-coefficient and transitivity" (submit review by 3/6)

Spectral Methods (aka Algebraic Graph Theory, see Chapter 14 in [BE] "Network analysis")

Three main objects of interest: adjacency matrix, Laplacian, and normalized Laplacian
What their spectrum (all eigenvalues, including algebraic multiplicity) can tell about network statistics, existence of subgraphs, classification, etc.?

Let $M \in \mathbb{C}^{n \times n}$. A non-zero vector $x \in \mathbb{C}^n$ is an eigenvector of M with corresponding eigenvalue $\lambda \in \mathbb{C}$ if

$$Mx = \lambda x$$

The solution exists iff $\text{rank}(M - \lambda I) < n$ iff $\det(M - \lambda I) = 0$, i.e., the eigenvalues are roots of $\det(M - \lambda I) = 0$.
If Q is non-singular then M and $Q^{-1}MQ$ have the same eigenvalues.
• If $M \in \mathbb{R}^{n \times n}$ and $M = M^T$ then $\exists$ non-singular Q s.t. $Q^{-1} = Q^T$ and $M' = Q^{-1}MQ$ has diagonal form. Eigenvectors of M' are e_i and

diagonal entry of *M'*

$$\det(M - \lambda I) = \det(M' - \lambda I) = \prod_i (\lambda_i - \lambda)$$

One can infer $tr(M) = \sum_{i=1}^{n} \lambda_i$ (check at home).
• If $v_i = Qe_i$ then $Mv_i = \lambda_i v_i$ and $v_i^T v_j = e_i^T e_j$.

orthonormal eigenvectors of *M*

Theorem 1. Let $M \in \mathbb{R}^{n\times n}$ and $M = M^T$, then

1. M has real eigenvalues $\lambda_1 \leq ... \leq \lambda_n$ and n orthonormal eigenvectors
2. multiplicity of λ_i as an eigenvalue = multiplicity of λ_i as a root of the characteristic polynomial $\det(M - \lambda I)$ = cardinality of a maximum linearly independent set of eigenvectors corresponding to λ_i
3. $\exists\, Q$ with $Q^T = Q^{-1}$ such that $QMQ^{-1} = \text{diag}(\lambda_1, ..., \lambda_n)$
4. $\det(M) = \prod_i \lambda_i$ and $tr(M) = \sum_i \lambda_i$

Theorem 2. Let G be a graph, and A its adj matrix with ordered eigenvalues λ_i, and Δ is a max degree of G then

1. $\lambda_n \leq \Delta$
2. $G = G_1 \cup G_2 \implies spec(G) = spec(G_1) \cup spec(G_2)$
3. G is bipartite $\implies$ (if $\lambda \in spec(G)$ then $-\lambda \in spec(G)$)
4. G is simple cycle $\implies \; spec(G) = \{2\cos(\frac{2\pi k}{n}) | k \in \{1, ..., n\}\}$
5. $G = K_{n_1,n_2} \implies \lambda_1 = -\sqrt{n_1 n_2}, \lambda_2 = ... = \lambda_{n-1} = 0,$ and $\lambda_n = \sqrt{n_1 n_2}$
6. $G = K_{n_1} \implies \lambda_1 = ... = \lambda_{n-1}, \lambda_n = n - 1$

Theorem 3.

1. $\sum_{i=1}^{n} \lambda_i =$ number of loops in G
2. $\sum_{i=1}^{n} \lambda_i^2 = 2 \times$ number of edges in G
3. $\sum_{i=1}^{n} \lambda_i^3 = 6 \times$ number of triangles in G

Homework: Prove any 2 out of 3 in Theorem 3 (submit by 3/6/2014)

Laplacian matrix $L = D - A$

$$\text{Incidence matrix } B = (b_{i,e}) = \begin{cases} 1 & i \text{ is the head of } e \\ -1 & i \text{ is the tail of } e \\ 0 & \text{otherwise} \end{cases}$$

For any $x \in \mathbb{C}^n$ $x^T L x = x^T B B^T x = \sum_{ij \in E} (x_i - x_j)^2$

A graph G consists of k connected components if and only if $\lambda_1(L) = ... = \lambda_k(L) = 0$ and $\lambda_{k+1}(L) > 0$.

Trees and Laplacian: for every $i \in \{1, ..., n\}$ the number of spanning trees in G is equal to $|\det(L_i)|$, where L_i is obtained from the Laplacian L by deleting row i and column i. Moreover, the number of spanning trees is equal to $\frac{1}{n} \prod_{i \geq 2} \lambda_i(L)$.