Measures of centrality, PageRank algorithm

You are given <u>iMDB actors collaboration network</u> in Pajek format (edge list and LNA formats are also available). Your task is to find the most important actors according to different measures of centrality.

	1 IA																	18 VIIIA	
1	1 14 DC																	2 8 EC	
	Degree Centrality	2 IIA											13 IIIA	14 IVA	15 VA	16 VIA	17 VIIA	Eigenvector Centrality	
2	3 12	4 13											5 9	6 4	7 1	8 1	9 3	10 2	
	BC	cc											SC	CCOEF	C _{COEF} -1	MNC	ECCOEF	PR	
	Betweenness Centrality	Closeness Centrality											Subgraph Centrality	Clustering Coefficient	inverse CCOEF	max. neighb. comp.	edge clustering coefficient	PageRank	
3	11 2	12 8											91 1	14 2	15 3	16 3 DAANG	17 2	18 1	
	RL	IC											CCxxx	SC₀	LAC	DMNC	SECCOEF	LR	
	RangeLimited Betweenness	Information Centrality	3 IIIA	4 IVB	5 VB	6 VIB	7 VIIB	8 VIIIB	9 VIIIB	10 VIIIB	11 IB	12 IIB		odd Subgraph Centrality	loc. avg. Connectivity	dens. max. neighb. comp.	sum of ECCOEF	LeaderRank	
4	19 4	20 2	21 1	57 1	58 1		61 1	62 1	63 1	64 1	65 1	89 1	90 1	32 2	33 1	34 1	35 2	36 2	
	BN	RC	IG	DCxx	BCxx	CCxx	ECCxx	KSxx	PRxx	1Gxx	RCxx	DCxxx	BCxxx	SC₅	KL	COCCOEF	PECcoeff	KS	
	BottleNeck Centrality	Radiality Centrality	Integration											even Subgraph Centrality	Clique Level	coexpr. weight CCOEF	PCC×ECCOEF	KatzStatus	
5	37 1			93 1					96 1		47 1		49 1	50 1	51 1	52 1		54 1	
	RWBC	RWCC	CC _{2,3,4}	ECCxxx	PRxxx	KSxxx	CCCcoef	RCxxx	IGxxx	DCBC	BCCC	CCKS	KSPR	DCPR	β	SC₂	NC	EC ₂	
	RandomWalk Betweenness	RandomWalk Closeness	2,3,4- localized-CC												Bipartivity	2-localized-SC	Neighborhood Centrality	2-localized-EC	
6	55 2	56 2	57-71 1	72 1			75 1	76 1	77 1		79 1	80 1	81 1	82 1	50 1	84 1	85 1	86 1	
	σ	ECC	WDC	DCECC	CCECC	BCECC	KSECC	PRECC	IGECC	DCCC	BCKS	CCPR	KSIG	DCIG	DCCcoef	SC₃	LI	EC ₃	
	stress Centrality	Eccentricity	Weighted Degree													3-localized-SC	Lobby Index	3-localized-EC	
7	87 1	88 1	89-103 1	104 1	105 1	106 1	107 1	108 1	109 1		111 1			114 1	115 1	50 1	117 1	118 1	
	BC _{2,3,4}	ECC ⁻¹	SDC	DCRC	CCRC	BCRC	KSRC	PRRC	IGRC	DCKS	BCPR	CCIG	DCPR	BCIG	ECCRC	BCCcoef	SC₄	EC₄	
	2,3,4- localized-BC	inverse Eccentricity	Sphere Degree Centrality														4-localized-SC	4-localized-EC	
															Betwee	nness-ba	ased		
							22 2 23 2 24 2 25 1 26 1 27 1 117 1							Distance-based					
	z mass Hybrid					FC	FD	US	DIS	ASS	DAM	DAM UC		Linear Combinations					
	Name					Functional Centrality	Functional Diversity	UniScore	Pairwise Dis- connectivity	Assortative Mixing	Damage	United compl. Centrality		Subgraph-based					
	Name												Clustering Coefficient-based						
							28 1 29 1 30 1 31 1 13 1 116 0 118 1							Edge Clustering Coefficient-based					
					EI	СМ	NαC	MC	HGI	HYP	HC	Spectral-based							
School	Schoch (University of Konstanz)					Essentiality Index	Complexity Measure	Normalized α Centrality	Moduland Centrality	Harary Graph Information	Hyperbolic Index	Harmonic Centrality	Miscellaneous						
	index wedaute centuring centuring information index centuring																		

I. Degree centrality and clustering coefficients

- 1. **(code)** Find the most important actors according to degree centrality $d_i = \frac{k_i}{n-1}$, where n is the number of network nodes and k_i is the degree of node i. What kind of actors have the highest d_i (e.g. Hollywood, Bollywood, international, unknown)?
- 2. **(code)** Find the most important actors according to clustering coefficient $C_i = \frac{2t_i}{k_i(k_i-1)}$, where k_i is the degree of node i and t_i is the number of triads including node i. You should use the link triad counting algorithm from previous labs. What kind of actors have the highest C_i (e.g. Hollywood, Bollywood, international, unknown)?
- 3. **(answer)** Find the most important actors according to μ -corrected clustering coefficient $C_i^{\mu} = \frac{2t_i}{k_i \mu}$, where k_i is the degree of node i, t_i is the number of triads including node i and μ is the maximum number of triads over a link. You should use the link triad counting algorithm from previous labs. What

II. Closeness and betweenness centrality

- 1. **(code)** Find the most important actors according to closeness centrality $\ell_i^{-1} = \frac{1}{n-1} \sum_{j \neq i} \frac{1}{d_{ij}}$, where n is the number of network nodes and d_{ij} is the distance between nodes i and j. You should use the breadth-first search algorithm from previous labs. What kind of actors have the highest ℓ_i^{-1} (e.g. Hollywood, Bollywood, international, unknown)?
- 2. **(answer)** Find the most important actors according to betweenness centrality $\sigma_i = \frac{1}{n^2} \sum_{st} \frac{g_{st}^i}{g_{st}}$, where n is the number of network nodes, g_{st} is the number of geodesic paths between nodes s and t, and g_{st}^i is the number of such paths through node i. You should ask the course instructor to do these computations for you. What kind of actors have the highest σ_i (e.g. Hollywood, Bollywood, international, unknown)?

III. Eigenvector centrality and PageRank algorithm

- 1. **(code)** Find the most important actors according to eigenvector centrality $e_i = \lambda_1^{-1} \sum_j A_{ij} e_j$, where A is the network adjacency matrix and λ_1 is a normalizing constant. You should use the power iteration algorithm shown below. What kind of actors have the highest e_i (e.g. Hollywood, Bollywood, international, unknown)?
- 2. **(code)** Find the most important actors according to PageRank algorithm $p_i = \alpha \sum_j A_{ij} \frac{p_j}{k_j} + \frac{1-\alpha}{n}$, where A is the network adjacency matrix, n is the number of network nodes, k_i is the degree of node i and α is the damping factor set to 0.85. You should use the PageRank algorithm shown below. What kind of actors have the highest p_i (e.g. Hollywood, Bollywood, international, unknown)?

```
input graph G, precision \epsilon
output eigenvector centrality E
    1: E \leftarrow \text{array of ones}
   2: do
              U \leftarrow \text{array of zeros}
    3:
              for nodes i \in N do
    4:
   5:
                  for neighbors j \in \Gamma_i do
                       U[i] \leftarrow U[i] + E[j]
    6:
    7:
              u \leftarrow ||U||
             for nodes i \in N do
    8:
                   U[i] \leftarrow U[i] \cdot n/u
   9:
             \Delta \leftarrow \|E - U\|
  10:
              E \leftarrow \ddot{U}
  11:
  12: while \Delta > \epsilon
  13: return E
```

```
input graph G, damping \alpha, precision \epsilon
output PageRank ranks P
    1: P \leftarrow \text{array of } n^{-1} - \text{s}
              U \leftarrow \text{array of zeros}
    3:
              for nodes i \in N do
    4:
                   for predecessors j \in \Gamma_i^{in} do
                        U[i] \leftarrow U[i] + P[j] \cdot \alpha / k_i^{out}
    6:
              u \leftarrow ||U||
    7:
              for nodes i \in N do
    8:
                   U[i] \leftarrow U[i] + (1-u)/n
    9:
              \Delta \leftarrow \|P - U\|
  10:
  11:
              P \leftarrow U
  12: while \Delta > \epsilon
  13: return P
```