Introduction to Network Analysis using Pajek

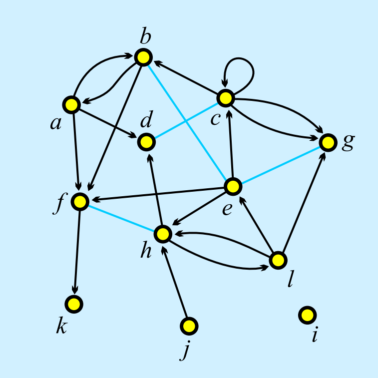
Description of networks

Vladimir Batagelj

IMFM Ljubljana, IAM UP Koper and University of Ljubljana

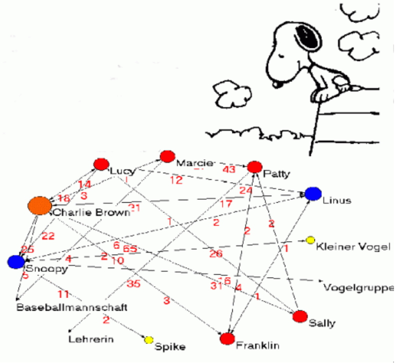
7th Internatonal Summer School
THEORY AND METHODS OF NETWORK ANALYSIS
HSE, Moscow, Russia, 19-23 June, 2017

Outline



e-mail: vladimir.batagelj@fmf.uni-lj.si
slides: 7ISS-nets.pdf
version: June 20, 2017

Networks



Alexandra Schuler/ Marion Laging-Glaser:
Analyse von Snoopy Comics

A network is based on two sets – set of nodes (vertices), that represent the selected units, and set of links (lines), that represent ties between units. They determine a graph. A link can be directed – an arc, or undirected – an edge.

Additional data about nodes or links can be known – their properties (attributes). For example: name/label, type, value, . . .

Network = Graph + Data

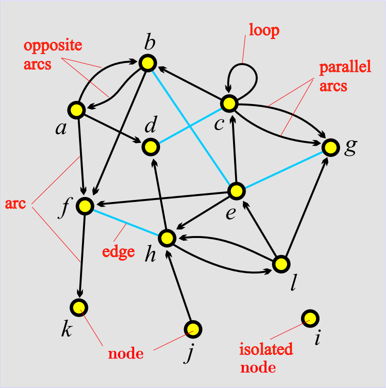
The data can be measured or computed.

Networks / Formally

A network $\mathcal{N} = (\mathcal{V}, \mathcal{L}, \mathcal{P}, \mathcal{W})$ consists of:

- a graph $\mathcal{G} = (\mathcal{V}, \mathcal{L})$, where $\mathcal{V}$ is the set of nodes, $\mathcal{A}$ is the set of arcs, $\mathcal{E}$ is the set of edges, and $\mathcal{L} = \mathcal{E} \cup \mathcal{A}$ is the set of links.
  $n = |\mathcal{V}|$, $m = |\mathcal{L}|$
- $\mathcal{P}$ node value functions / properties: $p : \mathcal{V} \rightarrow A$
- $\mathcal{W}$ link value functions / weights: $w : \mathcal{L} \rightarrow B$

Graph

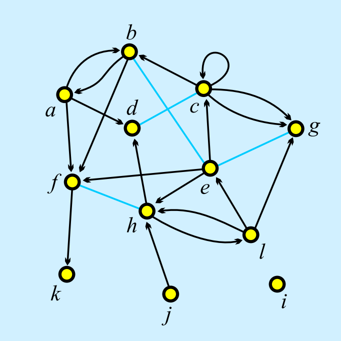


unit, actor – node, vertex
tie, line – link, edge, arc

arc = directed link, (a, d)
a is the initial node,
d is the terminal node.

edge = undirected link, $(c : d)$
c and d are end nodes.

Graph / Sets – NET



$$\mathcal{V} = \{a, b, c, d, e, f, g, h, i, j, k, l\}$$

$$\mathcal{A} = \{(a, b), (a, d), (a, f), (b, a), (b, f), (c, b), (c, c), (c, g)_1, (c, g)_2, (e, c), (e, f), (e, h), (f, k), (h, d), (h, l), (j, h), (l, e), (l, g), (l, h)\}$$

$$\mathcal{E} = \{(b\colon e), (c\colon d), (e\colon g), (f\colon h)\}$$

$$\mathcal{G} = (\mathcal{V}, \mathcal{A}, \mathcal{E})$$

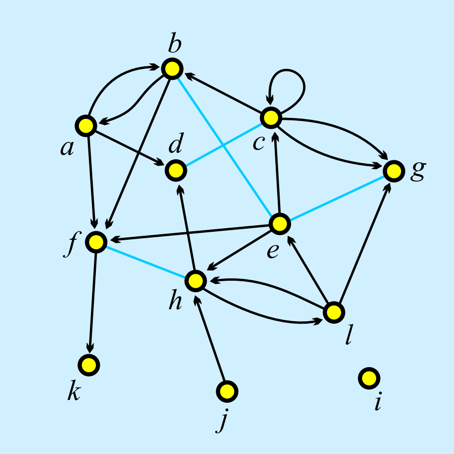
$$\mathcal{L} = \mathcal{A} \cup \mathcal{E}$$

$\mathcal{A} = \emptyset$ – undirected graph; $\mathcal{E} = \emptyset$ – directed graph.

Pajek: local: `GraphSet`; `TinaSet`;

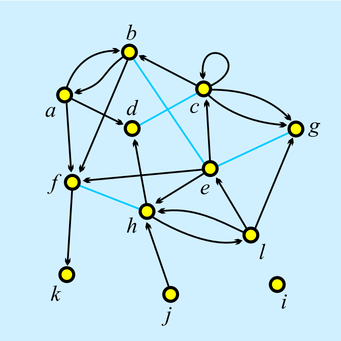
WWW: `GraphSet` / `net`; `TinaSet` / `net`, picture `picture`.

Graph / Sets – NET




```
*Vertices  12
  1 "a"     0.1020  0.3226
  2 "b"     0.2860  0.0876
  3 "c"     0.5322  0.2304
  4 "d"     0.3259  0.3917
  5 "e"     0.5543  0.4770
  6 "f"     0.1552  0.6406
  7 "g"     0.8293  0.3249
  8 "h"     0.4479  0.6866
  9 "i"     0.8204  0.8203
 10 "j"     0.4789  0.9055
 11 "k"     0.1175  0.9032
 12 "l"     0.7095  0.6475
*Arcs
  1  2
  2  1
  1  4
  1  6
  2  6
  3  2
  3  3
  3  7
  3  7
  5  3
  5  6
  5  8
  6 11
  8  4
 10  8
 12  5
 12  7
  8 12
 12  8
*Edges
  2  5
  3  4
  5  7
  6  8
```

Graph / Neighbors – NET



$N_A(a) = \{b, d, f\}$
$N_A(b) = \{a, f\}$
$N_A(c) = \{b, c, g, g\}$
$N_A(e) = \{c, f, h\}$
$N_A(f) = \{k\}$
$N_A(h) = \{d, l\}$
$N_A(j) = \{h\}$
$N_A(l) = \{e, g, h\}$

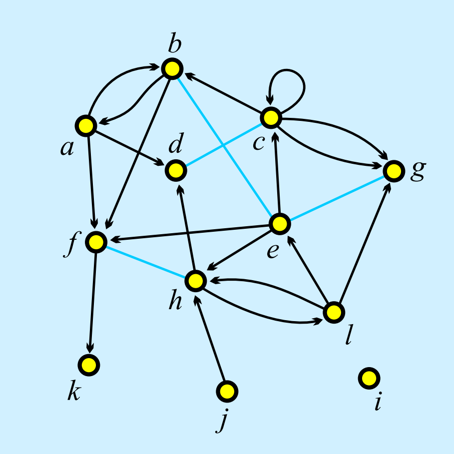
$N_E(e) = \{b, g\}$
$N_E(c) = \{d\}$
$N_E(f) = \{h\}$

Pajek: local: `GraphList`; `TinaList`;
WWW: `GraphList` / `net`; `TinaList` / `net`.

$$N(v) = N_A(v) \cup N_E(v), \qquad \text{also} \quad N_{out}(v),\ N_{in}(v)$$

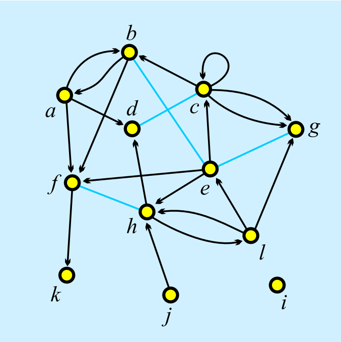
Star in v, $S(v)$ is the set of all links with v as their initial node.

Graph / Neighbors – NET




```
*Vertices  12
  1 "a"    0.1020   0.3226
  2 "b"    0.2860   0.0876
  3 "c"    0.5322   0.2304
  4 "d"    0.3259   0.3917
  5 "e"    0.5543   0.4770
  6 "f"    0.1552   0.6406
  7 "g"    0.8293   0.3249
  8 "h"    0.4479   0.6866
  9 "i"    0.8204   0.8203
 10 "j"    0.4789   0.9055
 11 "k"    0.1175   0.9032
 12 "l"    0.7095   0.6475
*Arcslist
  1  2  4  6
  2  1  6
  3  2  3  7  7
  5  3  6  8
  6 11
  8  4 12
 10  8
 12  5  7  8
*Edgeslist
  2  5
  3  4
  5  7
  6  8
```

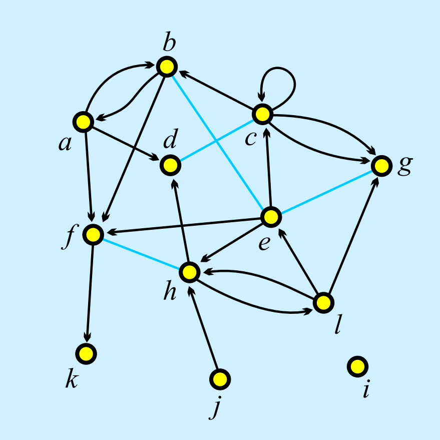
Graph / Matrix – MAT



	a	b	c	d	e	f	g	h	i	j	k	l
a	0	1	0	1	0	1	0	0	0	0	0	0
b	1	0	0	0	1	1	0	0	0	0	0	0
c	0	1	1	1	0	0	2	0	0	0	0	0
d	0	0	1	0	0	0	0	0	0	0	0	0
e	0	1	1	0	0	1	1	1	0	0	0	0
f	0	0	0	0	0	0	0	1	0	0	1	0
g	0	0	0	0	1	0	0	0	0	0	0	0
h	0	0	0	1	0	1	0	0	0	0	0	1
i	0	0	0	0	0	0	0	0	0	0	0	0
j	0	0	0	0	0	0	0	1	0	0	0	0
k	0	0	0	0	0	0	0	0	0	0	0	0
l	0	0	0	0	1	0	1	1	0	0	0	0

Pajek: local: `GraphMat`; `TinaMat`, picture `picture`;
WWW: `GraphMat` / `net`; `TinaMat` / `net`, `paj`.
Graph G is simple if in the corresponding matrix all entries are 0 or 1.

Graph / Matrix – MAT




```
*Vertices   12
  1 "a"     0.1020   0.3226
  2 "b"     0.2860   0.0876
  3 "c"     0.5322   0.2304
  4 "d"     0.3259   0.3917
  5 "e"     0.5543   0.4770
  6 "f"     0.1552   0.6406
  7 "g"     0.8293   0.3249
  8 "h"     0.4479   0.6866
  9 "i"     0.8204   0.8203
 10 "j"     0.4789   0.9055
 11 "k"     0.1175   0.9032
 12 "l"     0.7095   0.6475
*Matrix
 0 1 0 1 0 1 0 0 0 0 0 0
 1 0 0 0 1 1 0 0 0 0 0 0
 0 1 1 1 0 0 2 0 0 0 0 0
 0 0 1 0 0 0 0 0 0 0 0 0
 0 1 1 0 0 1 1 1 0 0 0 0
 0 0 0 0 0 0 0 1 0 0 1 0
 0 0 0 0 1 0 0 0 0 0 0 0
 0 0 0 1 0 1 0 0 0 0 0 1
 0 0 0 0 0 0 0 0 0 0 0 0
 0 0 0 0 0 0 0 1 0 0 0 0
 0 0 0 0 0 0 0 0 0 0 0 0
 0 0 0 0 1 0 1 1 0 0 0 0
```

Node Properties / CLU, VEC, PER

All three types of files have the same structure:

`*vertices` n	n is the number of nodes
v_1	node 1 has value v_1
...	
v_n	

CLUstering – partition of nodes – nominal or ordinal data about nodes
$v_i \in \mathbb{N}$: node i belongs to the cluster/group v_i;
VECtor – numeric data about nodes
$v_i \in \mathbb{R}$: the property has value v_i on node i;
PERmutation – ordering of nodes
$v_i \in \mathbb{N}$: node i is at the v_i-th position.

When collecting the network data consider to provide as much properties as possible.

Example: Wolfe Monkey Data

inter.net	inter.net	sex.clu	age.vec	rank.per
*Vertices 20	1 6 5	*vertices 20	*vertices 20	*vertices 20
1 "m01"	1 7 9	1	15	1
2 "m02"	1 8 7	1	10	2
3 "m03"	1 9 4	1	10	3
4 "m04"	1 10 3	1	8	4
5 "m05"	1 11 3	1	7	5
6 "f06"	1 12 7	2	15	10
7 "f07"	1 13 3	2	5	11
8 "f08"	1 14 2	2	11	6
9 "f09"	1 15 5	2	8	12
10 "f10"	1 16 1	2	9	9
11 "f11"	1 17 4	2	16	7
12 "f12"	1 18 1	2	10	8
13 "f13"	2 3 5	2	14	18
14 "f14"	2 4 1	2	5	19
15 "f15"	2 5 3	2	7	20
16 "f16"	2 6 1	2	11	13
17 "f17"	2 7 4	2	7	14
18 "f18"	2 8 2	2	5	15
19 "f19"	2 9 6	2	15	16
20 "f20"	2 10 2	2	4	17
*Edges	2 11 5			
1 2 2	2 12 4			
1 3 10	2 13 3			
1 4 4	2 14 2			
	...			

Important note: 0 is not allowed as node number.

Pajek's Project File / `PAJ`

All types of data can be combined into a single file – Pajek's project file *file*`.paj`.
The easiest way to do this is:

- read all data files in Pajek,
- compute some additional data,
- delete (dispose) some data,
- save all as a project file with `File/Pajek Project File/Save`.

Next time you can restore everything with a single `File/Pajek Project File/Read`.

Wolfe network as a Pajek's project file (PDF/paj).

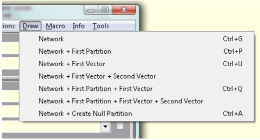
Representations of properties

Properties of nodes $\mathcal{P}$ and links $\mathcal{W}$ can be measured in different scales: numerical, ordinal and nominal. They can be input as data or computed from the network.

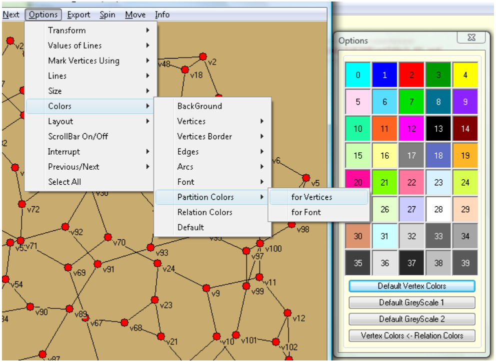
In Pajek numerical properties of nodes are represented by vectors, nominal properties by partitions or as labels of nodes. Numerical property can be displayed as size (width and height) of node (figure), as its coordinate; and a nominal property as color or shape of the figure, or as a node's label (content, size and color).

We can assign in Pajek numerical values to links. They can be displayed as value, thickness or grey level. Nominal vales can be assigned as label, color or line pattern (see Pajek manual, section **4.3**).

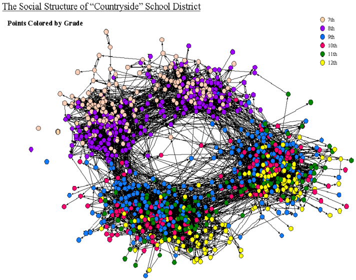
Some related operations

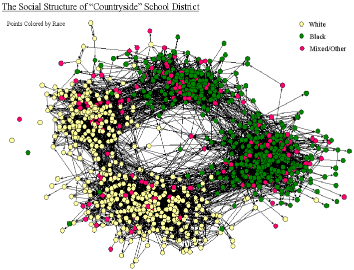



```
Operations/Network+Vector/Transform/Put Coordi
Network/Create Vector/Get Coordinate
[Draw] Options
[Draw] Layout/Energy/Kamada-Kawai/Free
[Draw] Export/2D/EPS-PS
```



Display of properties – school (Moody)





Types of networks

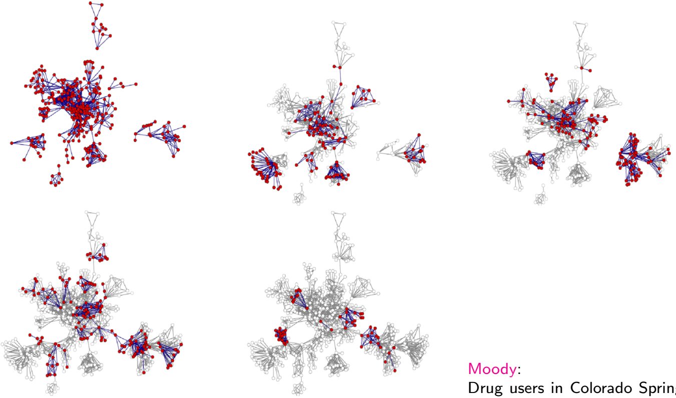
Besides ordinary (directed, undirected, mixed) networks some extended types of networks are also used:

- 2-mode networks, bipartite (valued) graphs – networks between two disjoint sets of nodes.
- multi-relational networks.
- temporal networks, dynamic graphs – networks changing over time.
- specialized networks: representation of genealogies as *p-graphs*; Petri's nets, ...

The network (input) file formats should provide means to express all these types of networks. All interesting data should be recorded (respecting privacy).

Temporal networks

In a temporal network the presence/activity of node/link can change through time. Pajek supports two types of descriptions of temporal networks based on presence and on events.



Moody:
Drug users in Colorado Springs, 5 years

Temporal network

Temporal network

$$\mathcal{N}_T = (\mathcal{V}, \mathcal{L}, \mathcal{P}, \mathcal{W}, T)$$

is obtained if the time T is attached to an ordinary network. T is a set of time points $t \in T$.

In temporal network nodes $v \in \mathcal{V}$ and links $l \in \mathcal{L}$ are not necessarily present or active in all time points. If a link $l(u, v)$ is active in time point t then also its endnodes u and v should be active in time t.

We will denote the network consisting of links and nodes active in time $t \in T$ by $\mathcal{N}(t)$ and call it a time slice in time point t. To get time slices in `Pajek` use

```
Network/Temporal Network/Generate in time
```

Temporal networks – presence


```
*Vertices 3
1 "a" [5-10,12-14]
2 "b" [1-3,7]
3 "e" [4-*]
*Edges
1 2 1 [7]
1 3 1 [6-8]
```

Time.net.

Node *a* is present in time points 5, 6, 7, 8, 9, 10 and 12, 13, 14.

Edge (1 : 3) is present in time points 6, 7, 8.

$*$ means 'infinity'.

A link is present, if both its endnodes are present.

Temporal networks – events

Event	Explanation
TI t	initial events – following events happen when time point t starts
TE t	end events – following events happen when time point t is finished
AV v n s	add vertex v with label n and properties s
HV v	hide node v
SV v	show node v
DV v	delete node v
AA u v s	add arc (u, v) with properties s
HA u v	hide arc (u, v)
SA u v	show arc (u, v)
DA u v	delete arc (u, v)
AE u v s	add edge $(u : v)$ with properties s
HE u v	hide edge $(u : v)$
SE u v	show edge $(u : v)$
DE u v	delete edge $(u : v)$
CV v s	change property of node v to s
CA u v s	change property of arc (u, v) to s
CE u v s	change property of edge $(u : v)$ to s
CT u v	change (un)directedness of link (u, v)
CD u v	change direction of arc (u, v)
PE u v s	replace pair of arcs (u, v) and (v, u) by single edge $(u : v)$ with properties s
AP u v s	add pair of arcs (u, v) and (v, u) with properties s
DP u v	delete pair of arcs (u, v) and (v, u)
EP u v s	replace edge $(u : v)$ by pair of arcs (u, v) and (v, u) with properties s

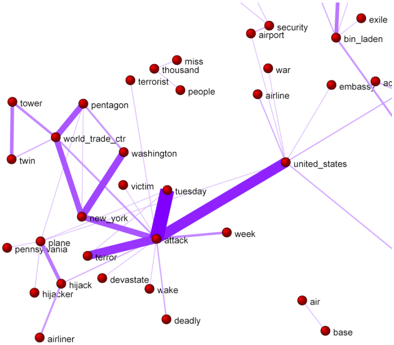
s can be empty.
In case of parallel links : k denotes the k-th link – `HE:3 14 37` hides the third edge linking nodes 14 and 37.

```
*Vertices 3
*Events
TI 1
AV 2 "b"
TE 3
HV 2
TI 4
AV 3 "e"
TI 5
AV 1 "a"
TI 6
AE 1 3 1
TI 7
SV 2
AE 1 2 1
TE 7
DE 1 2
DV 2
TE 8
DE 1 3
TE 10
HV 1
TI 12
SV 1
TE 14
DV 1
```

Time.tim. Friends.tim.

File/Network/Read Time Events

Temporal networks / September 11



Pictures in SVG: **66 days**.

Steve Corman with collaborators from Arizona State University transformed, using his Centering Resonance Analysis (**CRA**), daily Reuters news (66 days) about September 11th into a temporal network of words coappearance.

Multi-relational networks

A multi-relational network is denoted by

$$\mathcal{N} = (\mathcal{V}, (\mathcal{L}_1, \mathcal{L}_2, \ldots, \mathcal{L}_k), \mathcal{P}, \mathcal{W})$$

and contains different relations $\mathcal{L}_i$ (sets of links) over the same set of nodes. Also the weights from $\mathcal{W}$ are defined on different relations or their union.

Examples of such networks are: Transportation system in a city (stations, lines); WordNet (words, semantic relations: synonymy, antonymy, hyponymy, meronymy,...), KEDS networks (states, relations between states: Visit, Ask information, Warn, Expel person, ...), ...

. . . Multi-relational networks

The relation can be assigned to a link as follows:

- add to a keyword for description of links (*arcs, *edges, *arcslist, *edgeslist, *matrix) the number of relation followed by its name:

  ```
  *arcslist :3 "sent a letter to"
  ```

  All links controlled by this keyword belong to the specified relation. (Sampson, SampsonL)

- Any link controlled by *arcs or *edges can be assigned to selected relation by starting its description by the number of this relation.

  ```
  3:  47 14 5
  ```

  Link with endnodes 47 and 14 and weight 5 belongs to relation 3.

Computer-assisted text analysis

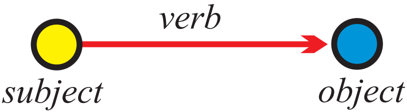
An often used way to obtain networks is the computer-assisted text analysis (CaTA).

Terms considered in TA are collected in a dictionary (it can be fixed in advance, or built dynamically). The main two problems with terms are equivalence (different words representing the same term) and ambiguity (same word representing different terms). Because of these the coding – transformation of raw text data into formal description – is done often manually or semiautomaticly. As units of TA we usually consider clauses, statements, paragraphs, news, messages, ...
Till now the thematic and semantic TA mainly used statistical methods for analysis of the coded data.
In thematic TA the units are coded as rectangular matrix *Text units* $\times$ *Concepts* which can be considered as a two-mode network.

Examples: M.M. Miller: VBPro, H. Klein: Text Analysis/ TextQuest.

. . . approaches to CaTA

In semantic TA the units (often clauses) are encoded according to the S-V-O (*Subject-Verb-Object*) model or its improvements.



Examples: Roberto Franzosi; *KEDS*, *Tabari*, KEDS / Gulf.
This coding can be directly considered as network with *Subjects* ∪ *Objects* as nodes and links (arcs) labeled with *Verbs*.

See also RDF triples in semantic web, SPARQL.

Multi-relational temporal network – KEDS/WEIS


```
% Recoded by WEISmonths, Sun Nov 28 21:57:00 2004
% from http://www.ku.edu/~keds/data.dir/balk.html
*vertices 325
1 "AFG" [1-*]
2 "AFR" [1-*]
3 "ALB" [1-*]
4 "ALBMED" [1-*]
5 "ALG" [1-*]
 ...
318 "YUGGOV" [1-*]
319 "YUGMAC" [1-*]
320 "YUGMED" [1-*]
321 "YUGMTN" [1-*]
322 "YUGSER" [1-*]
323 "ZAI" [1-*]
324 "ZAM" [1-*]
325 "ZIM" [1-*]
*arcs :0 "*** ABANDONED"
*arcs :10 "YIELD"
*arcs :11 "SURRENDER"
*arcs :12 "RETREAT"
 ...
*arcs :223 "MIL ENGAGEMENT"
*arcs :224 "RIOT"
*arcs :225 "ASSASSINATE TORTURE"
*arcs
224: 314 153 1 [4]                890402  YUG     KSV     224  (RIOT)  RIOT-TORN
212: 314 83 1 [4]                 890404  YUG     ETHALB  212  (ARREST PERSON) ALB ETHNIC JAILED
224: 3 83 1 [4]                   890407  ALB     ETHALB  224  (RIOT)  RIOTS
123: 83 153 1 [4]                 890408  ETHALB  KSV     123  (INVESTIGATE)   PROBING
 ...
42: 105 63 1 [175]                030731  GER     CYP     042  (ENDORSE)       GAVE SUPPORT
212: 295 35 1 [175]               030731  UNWCT   BOSSER  212  (ARREST PERSON) SENTENCED TO PRIS
43: 306 87 1 [175]                030731  VAT     EUR     043  (RALLY) RALLIED
13: 295 35 1 [175]                030731  UNWCT   BOSSER  013  (RETRACT)       CLEARED
121: 295 22 1 [175]               030731  UNWCT   BAL     121  (CRITICIZE)     CHARGES
122: 246 295 1 [175]              030731  SER     UNWCT   122  (DENIGRATE)     TESTIFIED
121: 35 295 1 [175]               030731  BOSSER  UNWCT   121  (CRITICIZE)     ACCUSED
```

Kansas Event Data System *KEDS*

Two-mode networks

In a two-mode network $\mathcal{N} = ((\mathcal{U}, \mathcal{V}), \mathcal{L}, \mathcal{P}, \mathcal{W})$ the set of nodes consists of two disjoint sets of nodes $\mathcal{U}$ and $\mathcal{V}$, and all the links from $\mathcal{L}$ have one endnode in $\mathcal{U}$ and the other in $\mathcal{V}$. Often also a weight $w : \mathcal{L} \to \mathbb{R} \in \mathcal{W}$ is given; if not, we assume $w(u, v) = 1$ for all $(u, v) \in \mathcal{L}$.

A two-mode network can also be described by a rectangular matrix $\mathbf{A} = [a_{uv}]_{\mathcal{U}\times\mathcal{V}}$.

$$a_{uv} = \begin{cases} w_{uv} & (u, v) \in \mathcal{L} \\ 0 & \text{otherwise} \end{cases}$$

Examples: (persons, societies, years of membership), (buyers/consumers, goods, quantity), (parlamentarians, problems, positive vote), (persons, journals, reading).

A two-mode network is announced by `*vertices` n $n_{\mathcal{U}}$.

Authors and works.

Description

V. Batagelj

Deep South

Classical example of two-mode network are the Southern women (Davis 1941).

Davis.paj. Freeman's overview.

Names of Participants of Group I	Code Numbers and Dates of Social Events Reported in *Old City Herald*													
	(1) 6/27	(2) 3/2	(3) 4/12	(4) 9/26	(5) 2/25	(6) 5/19	(7) 3/15	(8) 9/16	(9) 4/8	(10) 6/10	(11) 2/23	(12) 4/7	(13) 11/21	(14) 8/3
1. Mrs. Evelyn Jefferson	×	×	×	×	×	×		×	×					
2. Miss Laura Mandeville	×	×	×		×	×	×	×						
3. Miss Theresa Anderson		×	×	×	×	×	×	×	×					
4. Miss Brenda Rogers	×		×	×	×	×	×	×						
5. Miss Charlotte McDowd			×	×	×		×							
6. Miss Frances Anderson			×		×	×		×						
7. Miss Eleanor Nye					×	×	×	×						
8. Miss Pearl Oglethorpe						×		×	×					
9. Miss Ruth DeSand					×		×	×	×					
10. Miss Verne Sanderson							×	×	×			×		
11. Miss Myra Liddell								×	×	×		×		
12. Miss Katherine Rogers								×	×	×		×	×	×
13. Mrs. Sylvia Avondale							×	×	×	×		×	×	×
14. Mrs. Nora Fayette						×	×		×	×	×	×	×	×
15. Mrs. Helen Lloyd							×	×		×	×	×		
16. Mrs. Dorothy Murchison								×	×					
17. Mrs. Olivia Carleton									×		×			
18. Mrs. Flora Price									×		×			