Slide 2

Ci sono diversi criteri per fare una visualizzazione dei grafi.

Si parte dal un pressupposto che i nodi non hanno coordinate x,y; ultimamente a volte ce li hanno e si chiamano *attributed nodes*, ossia nodi che hanno degi attributi.

In questo caso si può decidere di posizionare i nodi scegliendo due attributi tra quelli con cui si rappresentano i nodi e mettere un attributo sull’asse delle x, un altro attributo sull’asse delle y e rappresentare l’altro attributo mediante un colore.

Il problema generale è visualizzare nodi che non hanno attributi, quindi non si può selezionare una coordinata x ed una coordinata y.

Quindi come posizionare questi nodi privi di attributi? Si posizionano a random nell’immagine?

Slide 3

CI sono dei criteri che derivano da principi psicologici:

• Crossing number

Il numero degli edge che si incrociano.

Se si fanno incrociare tanti nodi, diventa un mischione.

L’ideale è che tutti gli edge non si intersecassero: è possibile, ma in modo limitato, perché quando il numero di nodi si alza diventa impossibile non avere intersezioni.

• Simmetria

La psicologia dice che la simmetria è bella perché noi umani funzioniamo per simmetria (basti pensare che abbiamo due occhi…).

• Slope number

Quanto gli archi sono orientati o quanto seguono un’unica orientazione.

(Nella figura si ha elevato crossing, poca simmetria, gli archi hanno tutte le possibili orientazioni, quanto gli archi sono curvi e poi la risoluzione angolare.)

I criteri principali sono: crossing, simmetria e slope.

Slide 4

**Le convenzioni per la visualizzazione dei grafi**

• Polyline drawaing

Si può imporre che degi edge siano realizzati tramite linee poligonali (linee spezzate per congiungere i nodi) per evitare il maggior numero di crossing.

• Straight-line drawing

Gli edge sono rappresentati da linee dritte.

• Orthogonal drawing

• Grid drawing

• Upward (downward) drawing

• Planar drawing

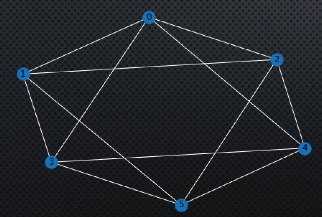
A seconda della libreria utilizzata per la visualizzazione di grafi, si possono implementare diversi criteri/convezioni.

Slide 5

Criterio più importante per la visualizzazione e qualità dei grafi: **CROSSING NUMBER**.

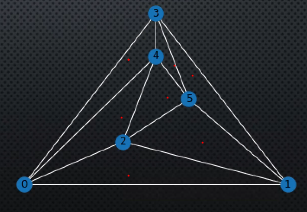
Si può richiedere che gli edge che si incrocino siano il meno possibile.

Risultato: si ha un grafo planare perhé non si hanno intersezioni.

I grafi possono essere resi planari se si vedono in 3D.

Questo è un grafo non planare, però, ad esempio,si può pensare che l’edge che collega 3 e 4 sia sotto, mentre gli edge che collegano 1 a 5 e 2 a 5 siano sopra.

Slide 6

**Eulero** formalizza quante sono le facce di un grafo planare, ossia gli spazi tra edge non intersecanti, in base al numero di edge e di vertici, qui sono 7: 

Slide 7

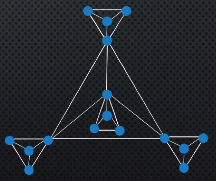
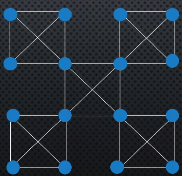
Solitamente ciò che si richiede è la **SIMMETRIA** ed esistono diversi tipi di simmetrie.

Un **isomorfismo** è una trasformazione che fa sì che la posizione dei vertici, quindi dei nodi, non venga inalterata dopo la trasformazione.

La similarità tra i vertici è indicata dalla connessione tra i vertici.

La trasformazione in simmetria di un grafo fa in modo che i vertici che sono vicini prima rimangano vicini anche dopo.

Se il grafo è pesato vale la simmetria in termini di peso dell’edge, se il grafo non è pesato vuol dire che la trAsformazione in simmetria ompone che vertici che erano connessi prima, lo siano anche dopo (altrimenti si andrebbero a ridurre il numero di edge del grafo).

Ci sono simmetrie possibili solo rispetto ad un asse, ad esempio qui c’è la simmetria rispetto all’asse verticale: , qui rispetto all’asse orizzontale, verticale e diagonale:  .

Slide 8

I primi due criteri (crossing e simmetria) sono più complessi da implementare, questo (slope) è più semplice quando ci sono meno connessioni, ovvero si richiederebbe di avere un’unica orientazione tra gli edge del grafo.

Per questa visualizzazione esistono delle visualizzazione particolari.

Slide 9

Quali sono i tre algoritmi principali che consentono di visualizzare un grafo (piccolo o grosso)?

• Quello più famoso è il *force-directed graph visualization method*: è una simulazione di sistemi fisici.

Questo metodo non fa altro che disporre all’inizio i nodi (a volte a caso nella figura, avolte in base alla somiglianza) poi si fa in modo di creare delle forze attrattive repulsive, ossia attraggono tra loro nodi vicini e poi le forze repulsive distanziano nodi lontani.

Si tratta di algoritmi iterativi.

• SPECTRAL GRAPH

Si parla essenzialmente di ***eigenvector*** e di ***eigenvalues***.

Fondalmentalmente lavorano sulla matrice di adiacenza o sulla matrice, poi su questa si calcolano gli autovettori che consentono di calcolare le coordinate che rappresnetano la proiezione dei nodi in uno spazio bidimensionale.

Poiché se si prendono due autovettori si hanno due dimensioni.

• Circular layout

Prende i vertici/nodi li partiziona in un dato modo secondo criteri dati dall’utente e li mette in un cerchio e crea connessioni tra i nodi del cerchio.

Slide 14

Soo stati presi i nodi e sono stati clusterizzati in qualche modo, ad esempio: tutti quelli che hanno votato Obama…

Si mettono in un cerchio e poi si connettono.

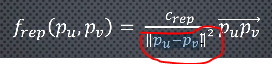
Facendo questa disposizione dei nodi si può vedere se la clusterizzazione dei nodi è anche delle relazioni.

Si possono vedere anche le relazioni tra cluster.

Slide 10

Si definiscono delle equazioni per definire le forze attrattive e repulsive, e si itera su queste forze.

Dato che si tratta di una simulazione, si raggiungerà una convergenza, ossia il momento in cui non si sposteranno più i nodi tra di loro.

La forza repulsiva viene normalizzata rispetto alla distanza tra i due nodi: 

Nodi lontanti vengono allontanati.

Forza attrattiva → i nodi vicini tendono ad essere tenuti vicini

Si calcola quindi la forza repulsiva per i nodi lontani e la forza attrattiva per i nodi vicini.

Quindi si definisce il movimento per tutti i nodi.

**Nodi vicini 1** → nodi direttamente raggiungibili da un edge

**Nodi vicini 2** → nodi direttamente raggiungibili da un cammino che passa almeno 2 archi

Come risultato finale si avrà un vettore avente lo spostamento di un particolare nodo per una data iterazione.

Slide 13

Amici nello stesso partito formano un cluster.

Slide 15

Si possono fare queste tipologie di visualizzazione e altre che supportano la *graph analysis*.

Perché si possono simulare i grafi?

È importante perché consente di andare a capire come tali grafi si formano.

Slide 16

Network sociali – grafi sociali

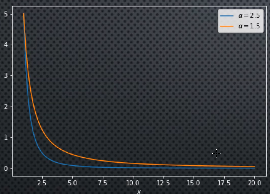
Slide 17

La *degree* di un nodo è il numero di vicini a distanza 1, cioè il numero di vicini raggiungibili da un nodo.

In un grafo sociale, ad esempio di Facebook, nodi ad alto degree rappresentano tendenzialmente persone famose poiché hanno tanti amici.

Invece nodi a distanza 2 sono nodi per cui se io sono A amica di B e B amico di C, io A sono a distanza 2 da C.

Nella *social network analysis* ci sono tanti utenti che hanno pochi amici e tanti utenti che hanno pochi amici:

Qui l’amicizia è stabilita in termini di relazione tra nodi, non di amicizia vera e propria.