DECSAI
Departamento de Ciencias de la Computación e I.A.
Universidad de Granada

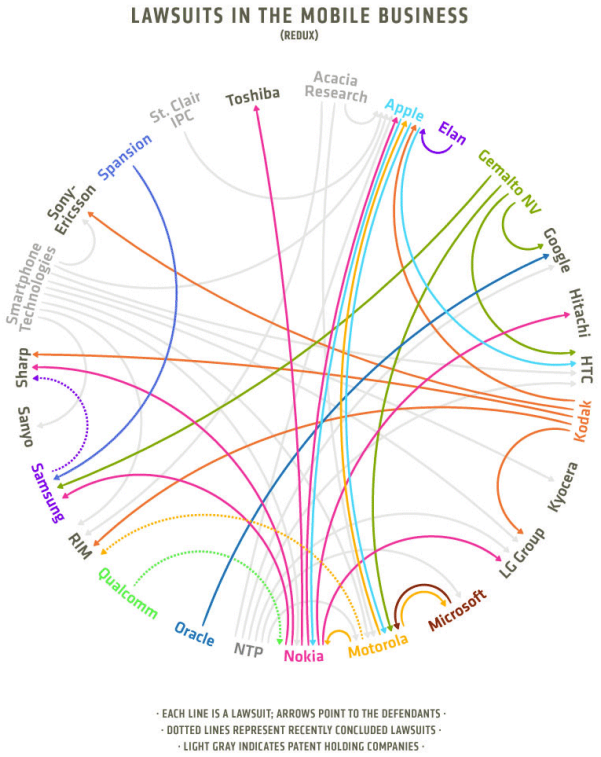
Predicción de enlaces

© Fernando Berzal, berzal@acm.org

Predicción de enlaces

- El problema de la predicción de enlaces
- Evaluación de resultados
- Métodos basados en similitud
  - Métodos locales
  - Métodos globales
  - Métodos cuasi-locales
- Métodos probabilísticos
- Métodos algorítmicos
  - Métodos basados en clasificadores
  - Métodos basados en metaheurísticas
  - Métodos basados en factorizaciones
- Técnicas de preprocesamiento
- Apéndice: Clasificación en redes

Predicción de enlaces



Predicción de enlaces

El problema

Dada una instantánea de una red
en el instante de tiempo t, G(t) = (V(t), E(t)),
¿cuál será el conjunto de enlaces
que se formará en el instante t+Δ?

$E(t) \rightarrow E(t+\Delta)$?

Predicción de enlaces

Aplicaciones

- Sistemas de recomendación
  - "Collaborative filtering" (vs. content-based filtering)
  - Redes sociales
- Integración de datos
  - Resolución de entidades (a.k.a. record linkage)
- Bioinformática
  - Predicción de interacciones entre proteínas

Evaluación

Evaluación de resultados

Como en cualquier problema de clasificación…

- **Métricas**
  Cómo evaluar la "calidad" de un modelo de clasificación.
- **Métodos**
  Cómo estimar, de forma fiable, la calidad de un modelo.
- **Comparación**
  Cómo comparar el rendimiento relativo
  de dos modelos de clasificación alternativos.

Evaluación: Métricas

Matriz de confusión
(confusion matrix)

		Predicción	
		C_P	C_N
Clase real	C_P	TP: True positive	FN: False negative
	C_N	FP: False positive	TN: True negative

Precisión del clasificador

accuracy = (TP+TN)/(TP+TN+FP+FN)

Evaluación: Métricas

Limitaciones de la precisión ("accuracy") :

Supongamos un problema con 2 clases no equilibradas:

- 9990 ejemplos de la clase N (ausencia de enlaces)
- 10 ejemplos de la clase P (presencia de enlaces)

Si el modelo de clasificación siempre dice que los ejemplos son de la clase N, su precisión es

9990/10000 = **99.9%**

Totalmente engañosa, ya que nunca detectaremos ningún ejemplo de la clase P.

Evaluación: Métricas

Alternativa: Matriz de costes

C(i|j)		Predicción	
		C_P	C_N
Clase real	C_P	C(P|P)	C(N|P)
	C_P	C(P|N)	C(N|N)

El coste de clasificación será proporcional a la precisión del clasificador sólo si

$$\forall i,j: i \neq j \quad C(i|j) = C(j|i)$$
$$C(i|i) = C(j|j)$$

Evaluación: Métricas

Medidas "cost-sensitive"

		Predicción	
		C_P	C_N
Clase real	C_P	TP: True positive	FN: False negative
	C_N	FP: False positive	TN: True negative

precision = TP/(TP+FP)

True positive recognition rate

recall = sensitivity = hit-rate = TP/P = TP/(TP+FN)

True negative recognition rate

specificity = TN/N = TN/(TN+FP)

Evaluación: Métricas

Medidas "cost-sensitive"

		Predicción	
		C_P	C_N
Clase real	C_P	TP: True positive	FN: False negative
	C_N	FP: False positive	TN: True negative

F-score

Media armónica de precision y recall:

F = 2*precision*recall / (precision+recall)

F = 2TP / (2TP+FP+FN)

Evaluación: Métricas

Medidas "cost-sensitive"

		Predicción	
		C_P	C_N
Clase real	C_P	TP: True positive	FN: False negative
	C_N	FP: False positive	TN: True negative

F-score (β)

Media armónica ponderada entre precision y recall:

$$F = (1+\beta^2)*precision*recall / (\beta^2*precision+recall)$$

Evaluación: Métricas

Medidas "cost-sensitive"

		Predicción	
		C_P	C_N
Real	C_P	TP	FN
	C_N	FP	TN

Accuracy

		Predicción	
		C_P	C_N
Real	C_P	TP	FN
	C_N	FP	TN

Recall

		Predicción	
		C_P	C_N
Real	C_P	TP	FN
	C_N	FP	TN

Precision

		Predicción	
		C_P	C_N
Real	C_P	TP	FN
	C_N	FP	TN

F-measure

Evaluación: Métricas

En el caso de la predicción de enlaces…

Normalmente, sólo nos interesarán aquellos enlaces candidatos que es más probable que se formen [top k]:

- La precisión [precision] nos indica el porcentaje de acierto dentro de los k enlaces más probables:

$$\textbf{precision(k) = TP(k) / k}$$

- Accuracy, recall (sensitivity), specificity y F-score **no** aportan información adicional en este contexto.

Evaluación: Métodos

Para evaluar la precisión de un modelo de clasificación nunca debemos utilizar el conjunto de entrenamiento (lo que nos daría el "**error de resustitución**" del clasificador), sino un conjunto de prueba independiente:

Por ejemplo, podríamos reservar 2/3 de los ejemplos disponibles para construir el clasificador y el 1/3 restante lo utilizaríamos de **conjunto de prueba** para estimar la precisión del clasificador.

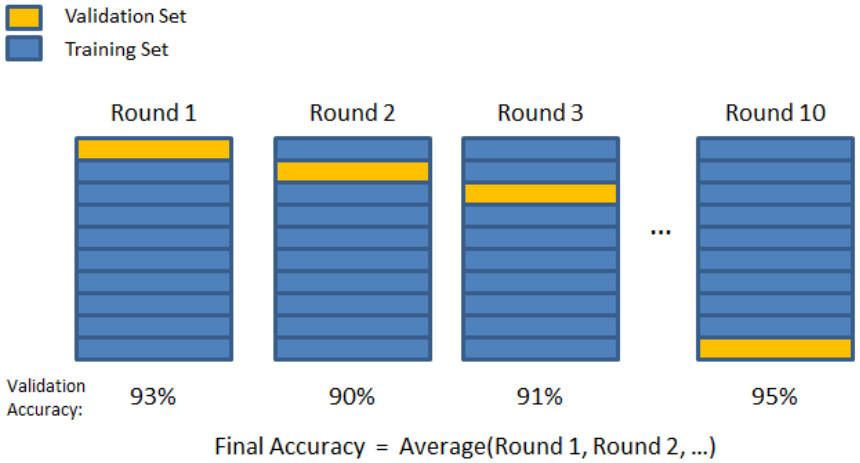
Evaluación: Métodos

Validación cruzada
[k-CV: k-fold Cross-Validation]

- Se divide aleatoriamente el conjunto de datos en k subconjuntos de intersección vacía (más o menos del mismo tamaño). Típicamente, k=10.
- En la iteración i, se usa el subconjunto i como conjunto de prueba y los k-1 restantes como conjunto de entrenamiento.
- Como medida de evaluación del método de clasificación se toma la media aritmética de las k iteraciones realizadas.

Evaluación: Métodos

Validación cruzada
[k-CV: k-fold Cross-Validation]



https://chrisjmccormick.wordpress.com/2013/07/31/k-fold-cross-validation-with-matlab-code/

Evaluación: Métodos

Validación cruzada

Variantes de la validación cruzada

- **"Leave one out":**
  Se realiza una validación cruzada con k particiones del conjunto de datos, donde k coincide con el número de ejemplos disponibles.

- **Validación cruzada estratificada:**
  Las particiones se realizan intentando mantener en todas ellas la misma proporción de clases que aparece en el conjunto de datos completo.

Evaluación: Métodos

Bootstraping

Muestreo uniforme con reemplazo de los ejemplos disponibles (esto es, una vez que se escoge un ejemplo, se vuelve a dejar en el conjunto de entrenamiento y puede que se vuelva a escoger).

NOTA: Método utilizado en "ensembles".

Evaluación: Métodos

Bootstraping

0.632 bootstrap

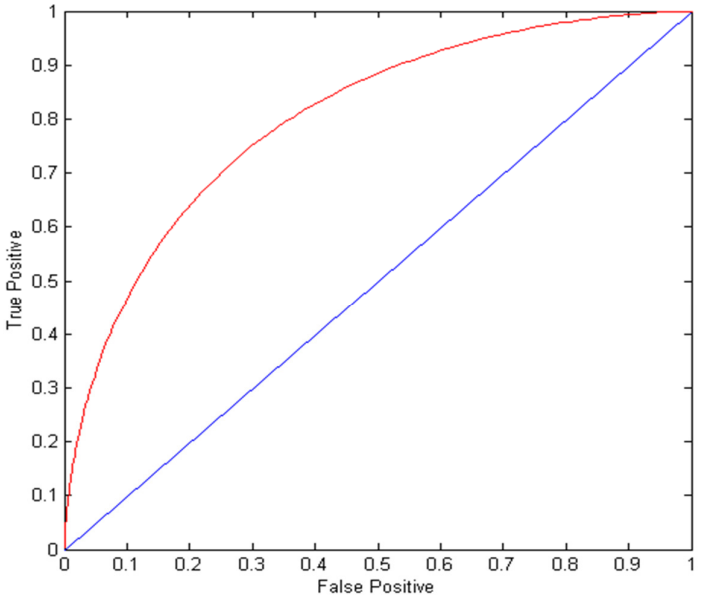
- Dado un conjunto de d datos, se toman d muestras. Los datos que no se escojan formarán parte del conjunto de prueba.
- En torno al 63.2% de las muestras estarán en el "bootstrap" (el conjunto de entrenamiento) y el 36.8% caerá en el conjunto de prueba, ya que $(1\text{-}1/d)^d \approx e^{-1} = 0.368$
- Si repetimos el proceso k veces, tendremos:

$$acc(M) = \sum_{i=1}^{k} (0.632 \times acc(M_i)_{test\_set} + 0.368 \times acc(M_i)_{train\_set})$$

Evaluación: Comparación

Curvas ROC

Receiver Operating Characteristics



TPR = TP/(TP+FN) Eje vertical: "true positive rate"

FPR = FP/(FP+TN) Eje horizontal: "false positive rate"

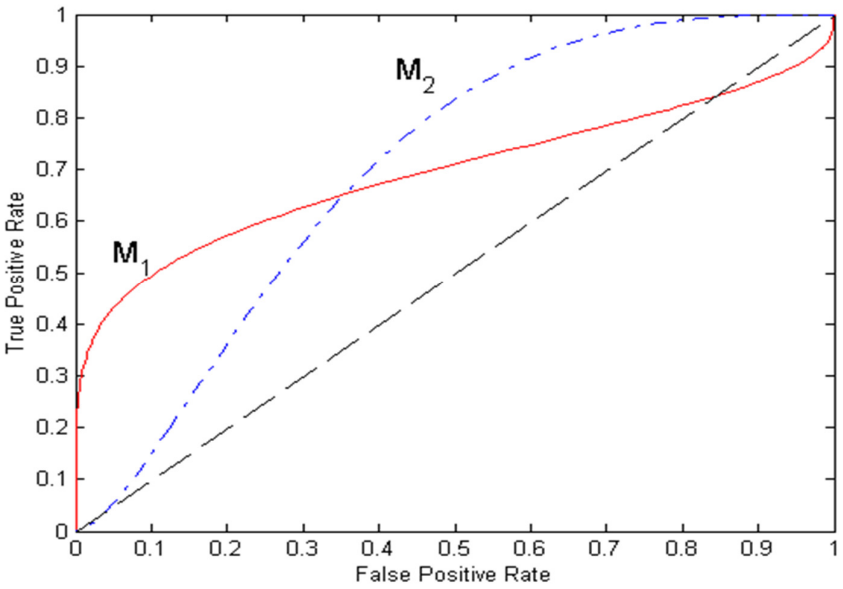
Evaluación: Comparación

Curvas ROC

- Desarrolladas en los años 50 para analizar señales con ruido: caracterizar el compromiso entre aciertos y falsas alarmas.
- Permiten comparar visualmente distintos modelos de clasificación.
- **AUC**: El área que queda bajo la curva es una medida de la precisión [accuracy] del clasificador:
  - Cuanto más cerca estemos de la diagonal (área cercana a 0.5), menos preciso será el modelo.
  - Un modelo "perfecto" tendrá área 1.

Evaluación: Comparación

Curvas ROC



Ningún modelo es consistentemente mejor que el otro: M_1 es mejor para FPR bajos, M_2 para FPR altos.

Evaluación: Comparación

Curvas ROC

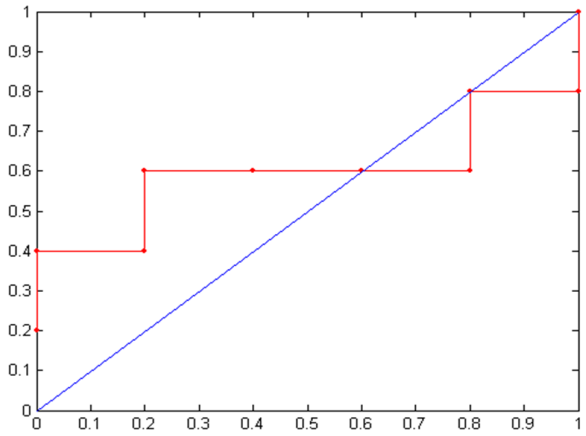
¿Cómo se construye la curva ROC?

- Se usa un clasificador que prediga la probabilidad de que un ejemplo E pertenezca a la clase positiva P(+|E)
- Se ordenan los ejemplos en orden decreciente del valor estimado P(+|E)
- Se aplica un umbral para cada valor distinto de P(+|E), para el que se cuenta el número de TP, FP, TN y FN.

TPR = TP/(TP+FN)
FPR = FP/(FP+TN)

Evaluación: Comparación

Curvas ROC



Ejemplo	P(+|E)	Clase
1	0.95	+
2	0.93	+
3	0.87	-
4	0.85	-
5	0.85	-
6	0.85	+
7	0.76	-
8	0.53	+
9	0.43	-
10	0.25	+

Clase	+	-	+	-	-	-	+	-	+	+	
	0.25	0.43	0.53	0.76	0.85	0.85	0.85	0.87	0.93	0.95	1.00
TP	5	4	4	3	3	3	3	2	2	1	0
FP	5	5	4	4	3	2	1	1	0	0	0
TN	0	0	1	1	2	3	4	4	5	5	5
FN	0	1	1	2	2	2	2	3	3	4	5
TPR	1	0.8	0.8	0.6	0.6	0.6	0.6	0.4	0.4	0.2	0
FPR	1	1	0.8	0.8	0.6	0.4	0.2	0.2	0	0	0

Evaluación: Comparación

Curvas ROC

Cálculo del área bajo la curva AUC

Se puede aproximar muestreando pares de enlaces del conjunto de validación y enlaces no existentes:

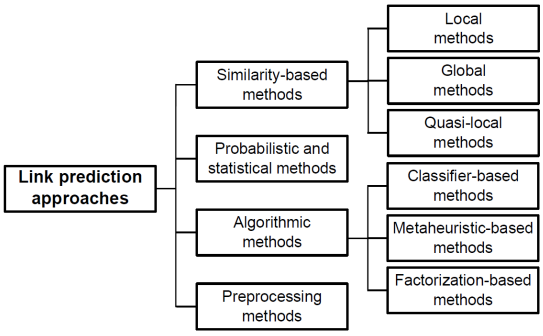
$$AUC = (n' + 0.5n'') / n$$

donde **n** es el número pares muestreados, **n'** es el número de pares en los que enlace del conjunto de validación recibió una probabilidad de existencia mayor que el enlace no existente y **n''** es el número de enlaces en los que hubo un empate.

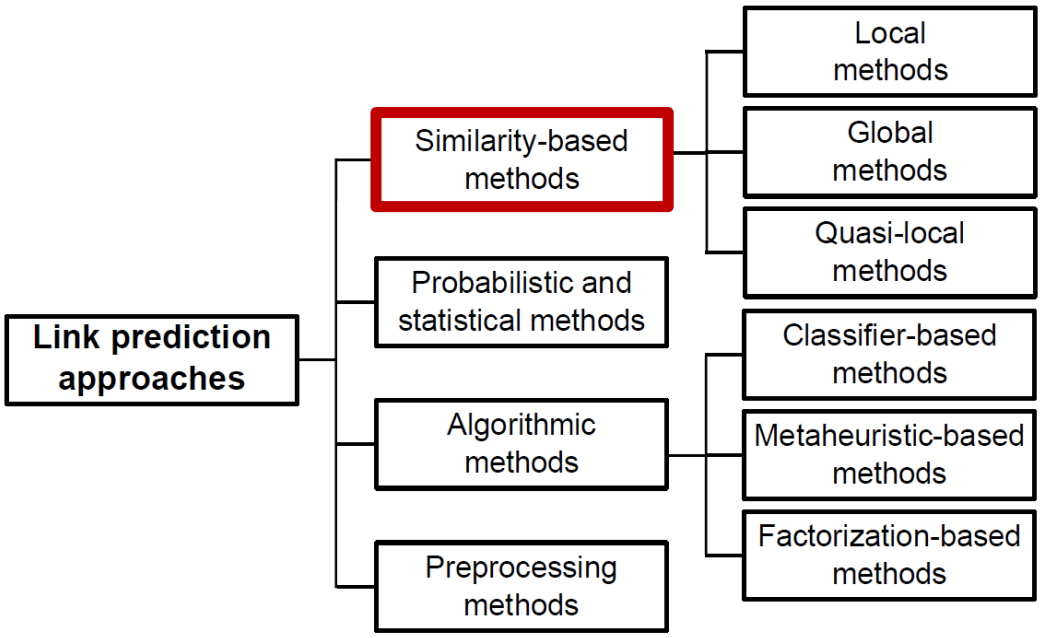
Predicción de enlaces

Métodos de predicción de enlaces

- Métodos basados en similitud
  - Métodos locales
  - Métodos globales
  - Métodos cuasi-locales
- Métodos probabilísticos
- Métodos algorítmicos
  - Métodos basados en clasificadores
  - Métodos basados en metaheurísticas
  - Métodos basados en factorizaciones



Métodos basados en similitud



Métodos basados en similitud

Hipótesis

Los nodos de una red tienden a formar enlaces con otros nodos similares

Idea básica

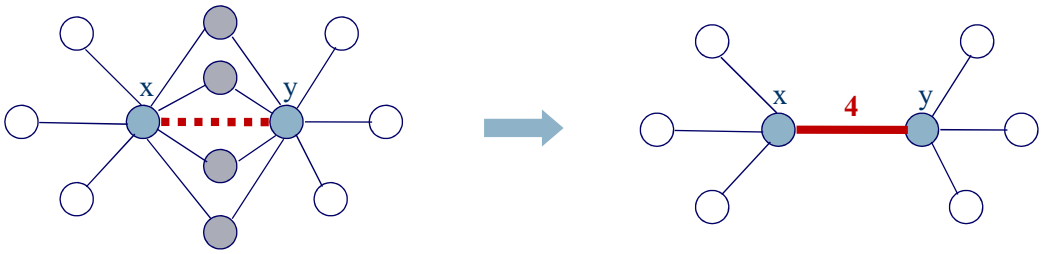
Si definimos una función de similitud s(x,y) entre parejas de nodos, podemos utilizar dicha función para establecer un ranking que nos indique qué enlaces es más probable que se formen en el futuro.

Métodos basados en similitud

Métodos locales

Dos nodos se consideran similares

si tienen vecinos compartidos…



Métodos basados en similitud

Métodos locales (1/5)

- **CN** [Common Neighbors]

$$s(x,y) = |\Gamma_x \cap \Gamma_y|$$

- **AA** [Adamic-Adar index]

$$s(x,y) = \sum_{z \in \Gamma_x \cap \Gamma_y} \frac{1}{\log |\Gamma_z|}$$

- **RA** [Resource Allocation index]

$$s(x,y) = \sum_{z \in \Gamma_x \cap \Gamma_y} \frac{1}{|\Gamma_z|}$$

Métodos basados en similitud

Métodos locales (2/5)

- **RA-CNI** [Resource Allocation index based on Common Neighbor Interaction]

$$s(x,y) = \sum_{z \in \Gamma_x \cap \Gamma_y} \frac{1}{|\Gamma_z|} + \sum_{e_{i,j} \in E, |\Gamma_i| < |\Gamma_j|, i \in \Gamma_x, j \in \Gamma_y} \left(\frac{1}{|\Gamma_i|} - \frac{1}{|\Gamma_j|} \right)$$

- **PA** [Preferential Attachment index]: Modelo de Barabasi- Albert

$$s(x,y) = |\Gamma_x||\Gamma_y|$$

- **J** [Jaccard index]

$$s(x,y) = \frac{|\Gamma_x \cap \Gamma_y|}{|\Gamma_x \cup \Gamma_y|}$$

Métodos basados en similitud

Métodos locales (3/5)

- **SA** [Salton index] = Cosine similarity

$$s(x,y) = \frac{|\Gamma_x \cap \Gamma_y|}{\sqrt{|\Gamma_x||\Gamma_y|}}$$

- **SO** [Sorensen index]

$$s(x,y) = \frac{2|\Gamma_x \cap \Gamma_y|}{|\Gamma_x| + |\Gamma_y|}$$

- **LLHN** [Local Leicht-Holme-Newman index]

$$s(x,y) = \frac{|\Gamma_x \cap \Gamma_y|}{|\Gamma_x||\Gamma_y|}$$

Métodos basados en similitud

Métodos locales (4/5)

- **HPI** [Hub-Promoted Index] $$s(x,y) = \frac{|\Gamma_x \cap \Gamma_y|}{\min(|\Gamma_x|, |\Gamma_y|)}$$
- **HDI** [Hub-Depressed Index] $$s(x,y) = \frac{|\Gamma_x \cap \Gamma_y|}{\max(|\Gamma_x|, |\Gamma_y|)}$$
- **IA** [Individual Attraction index]

$$s(x,y) = \sum_{z \in \Gamma_x \cap \Gamma_y} \frac{|e_{z,\Gamma_x \cap \Gamma_y}| + 2}{|\Gamma_z|}$$

- **SIA** [Simple IA]

$$s(x,y) = \sum_{z \in \Gamma_x \cap \Gamma_y} \frac{|e_{\Gamma_x \cap \Gamma_y}| + 2}{|\Gamma_z||\Gamma_x \cap \Gamma_y|}$$

Métodos basados en similitud

Métodos locales (5/5)

- **MI** [Mutual Information]

$$s(x,y) = -I(e_{x,y}|\Gamma_x \cap \Gamma_y) = -I(e_{x,y}) + \sum_{z \in \Gamma_x \cap \Gamma_y} I(e_{x,y}; z)$$

- **LNB** [Local Naive Bayes]

$$s(x,y) = \sum_{z \in \Gamma_x \cap \Gamma_y} f(z) \log(oR_z)$$

- **CAR** [CAR-based indices]: Local communities

$$s(x,y) = \sum_{z \in \Gamma_x \cap \Gamma_y} 1 + \frac{|\Gamma_x \cap \Gamma_y \cap \Gamma_z|}{2}$$

Métodos basados en similitud

Métodos locales

VENTAJAS

- Eficientes
- Paralelizables

DESVENTAJAS

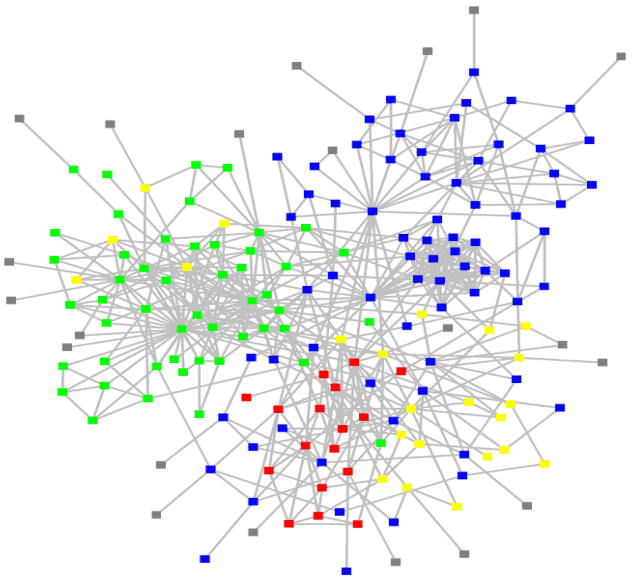
- Sólo consideran información local
  (de hecho, sólo se calcula la similitud entre pares de nodos con vecinos compartidos, i.e. a distancia 2)

  p.ej. Redes de mundo pequeño
  [small-world networks]

Métodos basados en similitud

Métodos globales

Se utiliza toda la información de la topología de la red



Métodos basados en similitud

Métodos globales: Caminos en la red

- **NSP** [Negated Shortest Path]

$$s(x,y) = -|shortest\ path_{x,y}|$$

- **KI** [Katz Index]

$$s(x,y) = \sum_{l=1}^{\infty} \beta^l |paths^l_{x,y}| = \sum_{l=1}^{\infty} \beta^l (A^l)_{x,y}$$

- **GLHN** [Global Leicht-Holme-Newman index]

$$S = I + \sum_{l=1}^{\infty} \phi^l A^l$$

Métodos basados en similitud

Métodos globales: Caminos aleatorios

- **RW** [Random Walks]

$$\overrightarrow{p^x}(t) = M^T \overrightarrow{p^x}(t-1)$$

- **RWR** [Random Walks with Restart]

$$\overrightarrow{p^x}(t) = \alpha M^T \overrightarrow{p^x}(t-1) + (1-\alpha)s^x$$

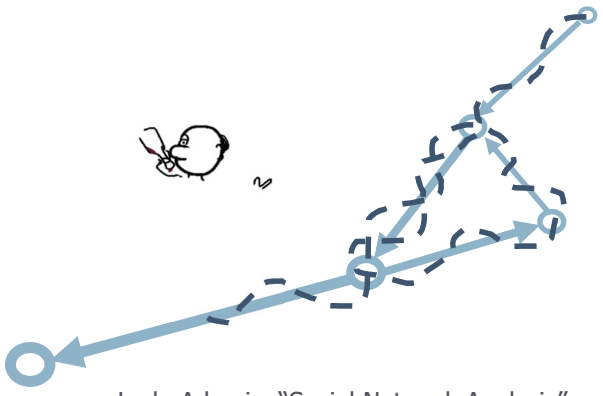
- **FP** [Flow Propagation]: Usando la matriz Laplaciana en vez de la matriz de adyacencia (L=D-A).
- **MERW** [Maximal Entropy Random Walk], teniendo en cuenta la tendencia a conectarse con nodos centrales.

Métodos basados en similitud

Métodos globales: Caminos aleatorios

Relación con PageRank

Google



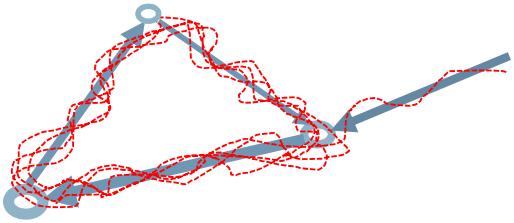
Lada Adamic, "Social Network Analysis"
https://www.coursera.org/course/sna

El PageRank de Google mide la importancia de un nodo en la red en proporción a la fracción de tiempo que un caminante aleatorio pasaría en él.

Métodos basados en similitud

Métodos globales: Caminos aleatorios

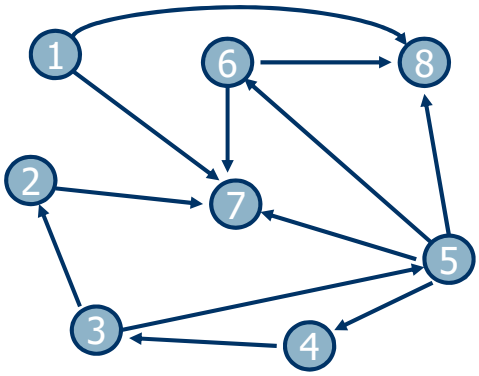
Relación con PageRank

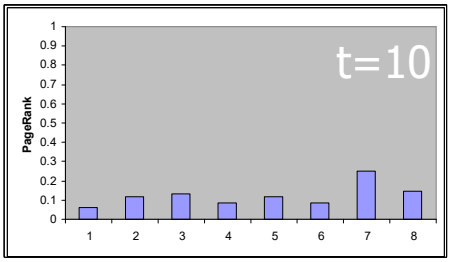


Problema: Atrapado en la red

Solución: Teletransporte

Salto aleatorio con una probabilidad dada.





Métodos basados en similitud

Métodos globales

- **SimRank** (cómo de pronto se encontrarán dos caminantes que empiezan en nodos diferentes y siguen un camino aleatorio).

- **PLM** [Pseudoinverse of the Laplacian Matrix]

$$s(x,y) = \frac{L^+_{x,y}}{\sqrt{L^+_{x,x} L^+_{y,y}}}$$

Métodos basados en similitud

Métodos globales

- **ACT** [Average Commute Time]: Número medio de pasos que hay que dar para llegar de x a y.

$$n(x,y) = |E|(L^+_{x,x} + L^+_{y,y} - 2L^+_{x,y})$$

$$s(x,y) = \frac{1}{L^+_{x,x} + L^+_{y,y} - 2L^+_{x,y}}$$

- **RFK** [Random Forest Kernel]

$$S = (I + L)^{-1}$$

- **BI** [Blondel Index]

$$S(t) = \frac{AS(t-1)A^T + A^T S(t-1)A}{\|AS(t-1)A^T + A^T S(t-1)A\|_F}$$

Métodos basados en similitud

Métodos globales

VENTAJAS

- Utilizan toda la información topológica de la red
- No están limitados a medir similitudes entre nodos que tengan vecinos compartidos (i.e. a distancia 2)

DESVENTAJAS

- Complejidad computacional.
- Paralelización compleja.

Métodos basados en similitud

Métodos cuasi-locales

Balance entre medidas locales y globales

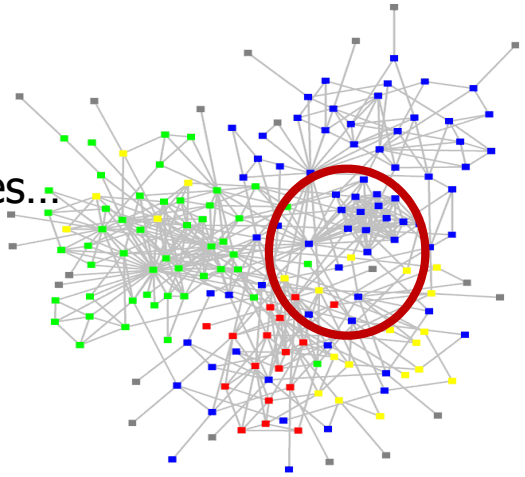
- Casi tan eficientes como los métodos locales.
- Consideran más información topológica que los métodos locales…

Métodos basados en similitud

Métodos cuasi-locales

Equilibro intermedio entre métodos locales y globales

- Casi tan eficientes como los métodos locales.
- Consideran más información topológica que los métodos locales…



Métodos basados en similitud

Métodos cuasi-locales

- **LPI** [Local Path Index]: Extensión del índice de Katz

$$S = \sum_{i=2}^{l} \beta^{i-2} A^i$$

- **LRW** [Local Random Walks]

$$s^{x,y}(t) = \frac{|\Gamma_x|}{2|E|} \overrightarrow{p_y^x}(t) + \frac{|\Gamma_y|}{2|E|} \overrightarrow{p_x^y}(t)$$

- **SRW** [Superposed Random Walks]

$$s^{x,y}(t) = \sum_{i=1}^{t} \left(\frac{|\Gamma_x|}{2|E|} \overrightarrow{p_y^x}(i) + \frac{|\Gamma_y|}{2|E|} \overrightarrow{p_x^y}(i) \right)$$

Métodos basados en similitud

Métodos cuasi-locales

- **ORA-CNI** [3rd Order Resource Allocation based on Common Neighbor Interactions]

$$s(x,y) = \sum_{z \in \Gamma_x \cap \Gamma_y} \frac{1}{|\Gamma_z|} + \sum_{e_{i,j} \in E, |\Gamma_i| < |\Gamma_j|, i \in \Gamma_x, j \in \Gamma_y} \left(\frac{1}{|\Gamma_i|} - \frac{1}{|\Gamma_j|} \right) + \beta \sum_{[x,p,q,y] \in paths^3_{x,y}} \frac{1}{|\Gamma_p||\Gamma_q|}$$

- **FL** [Friend Link], similar a LPI

$$s(x,y) = \sum_{i=2}^{l} \frac{1}{i-1} \frac{(A^i)_{x,y}}{\prod_{j=2}^{i} (|V| - j)}$$

- **PFP** [PropFlow Predictor], similar a RWR

Métodos basados en similitud

Métodos cuasi-locales

PFP [PropFlow Predictor]

Input: Network $G = (V, E)$, node x and max path length l.
Output: Score $S_{x,y}$ for all $n \leq l$-degree neighbors of y from x.

$Found = \{x\}$;
$NewSearch = \{x\}$;
$S_{x,x}$ = 1;
for ***each*** z ***in*** $V - \{x\}$ **do**
    $S_{x,z}$ = 0;
end
for $CurrentDegree$ ***from*** 0 ***to*** l **do**
    $OldSearch$ = $NewSearch$;
    $NewSearch = \emptyset$;
    for ***each*** i ***in*** $OldSearch$ **do**
        for ***each*** j ***in*** Γ_i **do**
            $S_{x,j} \leftarrow S_{x,j} + \frac{S_{x,i}}{|\Gamma_i|}$;
            if j ***is not in*** $Found$ **then**
                $Found = Found \cup \{j\}$;
                $NewSearch = NewSearch \cup \{j\}$;
            end
        end
    end
end

Métodos basados en similitud

Tabla resumen

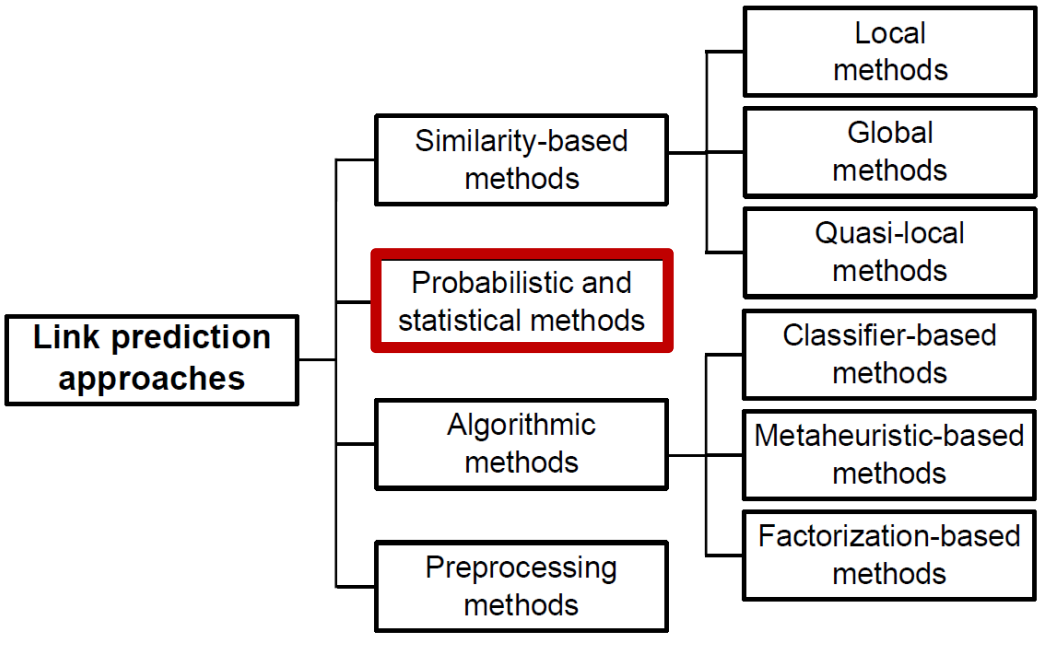
Local	CN	$O(vk^3)$	[Liben-Nowell and Kleinberg 2007]
	AA	$O(vk^3)$	[Adamic and Adar 2003]
	RA	$O(vk^3)$	[Zhou et al. 2009]
	RA-CNI	$O(vk^4)$	[Zhang et al. 2014]
	PA	$O(vk^2)$	[Barabási and Albert 1999]
	JA	$O(vk^3)$	[Jaccard 1901]
	SA	$O(vk^3)$	[Salton and McGill 1983]
	SO	$O(vk^3)$	[Sørensen 1948]
	HPI	$O(vk^3)$	[Ravasz et al. 2002]
	HDI	$O(vk^3)$	[Ravasz et al. 2002]
	LLHN	$O(vk^3)$	[Leicht et al. 2006]
	IA1	$O(vk^4)$	[Dong et al. 2011]
	IA2	$O(vk^3)$	[Dong et al. 2011]
	MI	$O(nk^6)$	[Tan et al. 2014]
	LNB	$O(O(f(z)) + vk^3)$	[Liu et al. 2011]
	CAR	$O(vk^4)$	[Cannistraci et al. 2013]

Métodos basados en similitud

Tabla resumen

Global	NSP	$O(ev \log v)$	[Liben-Nowell 2005]
	KI	$O(v^3)$	[Katz 1953]
	GLHN	$O(cv^2k)$	[Leicht et al. 2006]
	RW	$O(cv^2k)$	[Pearson 1905]
	RWR	$O(cv^2k)$	[Tong et al. 2006]
	FP	$O(cv^2k)$	[Vanunu and Sharan 2008]
	MERW	$O(cv^2k)$	[Li et al. 2011]
	SR	$O(v^2k^{2l+2})$	[Jeh and Widom 2002]
	PLM	$O(v^3)$	[Fouss et al. 2007]
	ACT	$O(v^3)$	[Fouss et al. 2007]
	RFK	$O(v^3)$	[Chebotarev and Shamis 2006]
	BI	$O(cv^2k)$	[Blondel et al. 2004]
Quasi local	LPI	$O(lv^2k)$	[Lü et al. 2009]
	LRW	$O(lv^2k)$	[Liu and Lü 2010]
	SRW	$O(lv^2k)$	[Liu and Lü 2010]
	ORA-CNI	$O(vk^6)$	[Zhang et al. 2014]
	FL	$O(lv^2k)$	[Papadimitriou et al. 2012]
	PFP	$O(vlk^l)$	[Lichtenwalter et al. 2010]

Métodos probabilísticos



Métodos probabilísticos

Hipótesis

La formación de la red se produce de acuerdo a algún modelo formal (de tipo estadístico).

Idea básica

Asumiendo que la red se ajusta a un modelo concreto, se estiman los parámetros de dicho modelo y se calcula la probabilidad de formación de cada posible enlace…

Métodos probabilísticos

Hierarchical structure model

Red organizada jerárquicamente

$$\mathcal{L}(D, \{p_n\}) = \prod p_n^{e_n} (1 - p_n)^{l_n r_n - e_n}$$

Input: Network $G = (V, E)$, number n of dendrograms to sample.
Output: Probability $P_{x,y}$ for all unconnected pairs of nodes.
$Samples$ = $\emptyset$;
for i ***from*** 1 ***to*** n **do**
    Initialize the Markov chain with a random dendrogram;
    Run Monte Carlo algorithm until equilibrium is reached;
    Insert resulting dendrogram D into $Samples$;
end
for ***each*** $e_{x,y}$ ***in*** $U_G - E$ **do**
    avg_prob = 0;
    for ***each*** $sample$ ***in*** $Samples$ **do**
        $n \leftarrow$ lower common ancestor of x and y in $sample$;
        $avg\_prob \leftarrow$ avg_prob $+ \frac{\overline{p}_n}{|Samples|}$;
    end
    $P_{x,y}$ = avg_prob;
end

Métodos probabilísticos

Stochastic block model

Red organizada en torno a comunidades…

$$\mathcal{L}(G|\mathcal{M}) = \prod_{a \leq b; a,b \in \mathcal{M}} p_{a,b}^{l_{a,b}} (1 - p_{a,b})^{r_{a,b} - l_{a,b}}$$

$$P_{x,y} = \frac{\sum_{\mathcal{M} \in \omega} \mathcal{L}(e_{x,y} \in E|\mathcal{M}) \mathcal{L}(G|\mathcal{M}) p(\mathcal{M})}{\sum_{\mathcal{M}' \in \omega} \mathcal{L}(G|\mathcal{M}') p(\mathcal{M}')}$$

Métodos probabilísticos

Cycle formation model

Red con tendencia a cerrar ciclos…

"Los amigos de mis amigos son mis amigos"

$$p_{x,y}(c_1, ..., c_k) = \frac{c_1 \prod_{i=2}^{k} c_i^{|paths_{x,y}^i|}}{c_1 \prod_{i=2}^{k} c_i^{|paths_{x,y}^i|} + (1 - c_1) \prod_{i=2}^{k} (1 - c_i)^{|paths_{x,y}^i|}}$$

Input: Network $G = (V, E)$, model degree k.
Output: Probability $P_{x,y}$ for all unconnected pairs of nodes.
Compute Generalized Clustering Coefficients $C(2), ..., C(k)$;
c_1 = Connecting probability in random graph with same degree distribution that G;
$c_2 = \frac{(1-c_1)C(2)}{c_1 - 2c_1 C(2) + C(2)}$;
for i ***from*** 3 ***to*** k **do**
  $c_i \leftarrow 0.5$;
end
for i ***from*** 3 ***to*** k **do**
  $c_i \leftarrow \arg\min_{c_i} |C(i) - f(c_1, ..., c_k)|$;
end
for ***each*** $e_{x,y}$ ***in*** $U_G - E$ **do**
  $P_{x,y} \leftarrow p_{x,y}(c_1, ..., c_k)$;
end

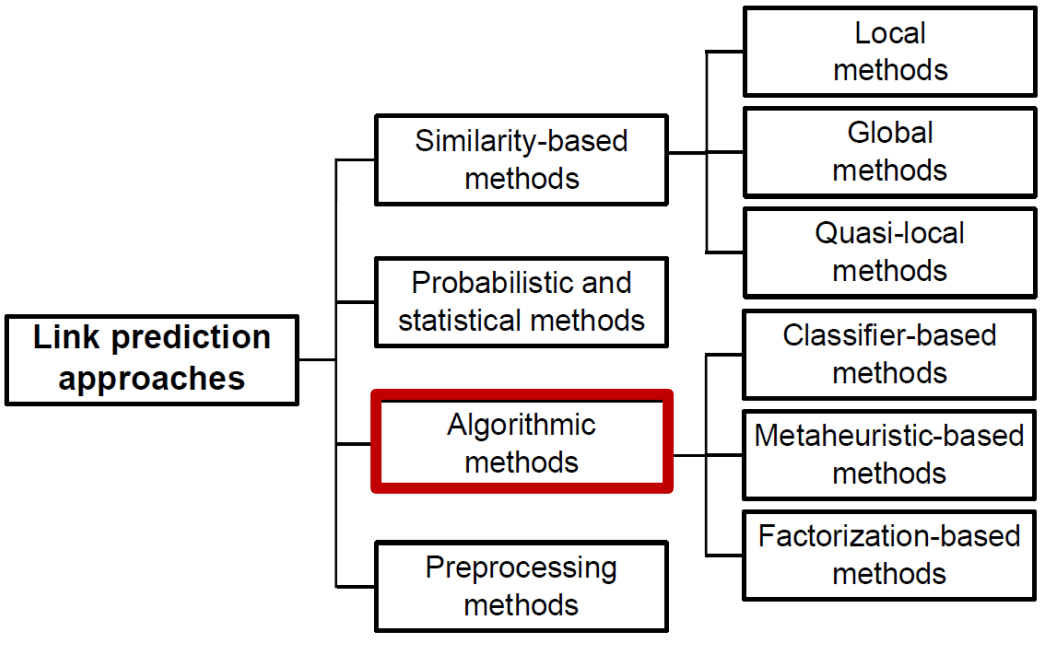
Métodos probabilísticos

Local co-occurrence model

Basado en propiedades topológicas locales ("escalable")

Input: Network $G = (V, E)$, central neighborhood set max size t, max path length k.
Output: Probability $P_{x,y}$ for all unconnected pairs of nodes.
for ***each*** $e_{x,y}$ ***in*** $U - E$ **do**
  $C_{x,y} = \emptyset$;
  for l ***from*** 2 ***to*** k **do**
    $p_i \leftarrow$ Compute and sort by length and frequency $paths^{l}_{x,y}$;
    for ***each*** p ***in*** p_i **do**
      if $|C_{x,y}| < t$ **then**
        Insert all nodes in p into $C_{x,y}$;
      end
    end
  end
  NDI = Compute non-derivable itemsets from $C_{x,y}$;
  $R_{x,y} = \emptyset$;
  for ***each*** ndi ***in*** NDI **do**
    if ndi ***in*** $C_{x,y}$ **then**
      Insert ndi into $R_{x,y}$;
    end
  end
  M = Initialize Markov Random Fields using $C_{x,y}$ and $R_{x,y}$;
  while ***not*** M *satisfies all constrains in* $R_{x,y}$ **do**
    for ***each*** r ***in*** $R_{x,y}$ **do**
      Update M to force satisfying r;
    end
  end
  $P_{x,y}$ = Infer probability of $e_{x,y}$ from M;
end

Métodos algorítmicos



Métodos algorítmicos

Métodos basados en clasificadores

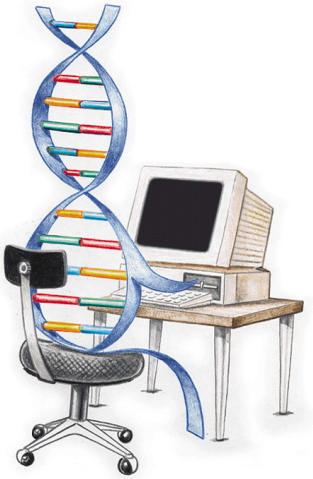
Consideran la predicción de enlaces como un problema clásico de aprendizaje supervisado (no balanceado).

- Árboles de decisión
- k-NN (vecinos más cercanos)
- SVMs [Support Vector Machines]
- Redes neuronales: perceptrones multicapa, RBFs…
- Naive Bayes
- Ensembles, p.ej. random forests

Métodos algorítmicos

Métodos basados en metaheurísticas

Algoritmos evolutivos
(permiten modelar la coexistencia de varios mecanismos de formación de enlaces).



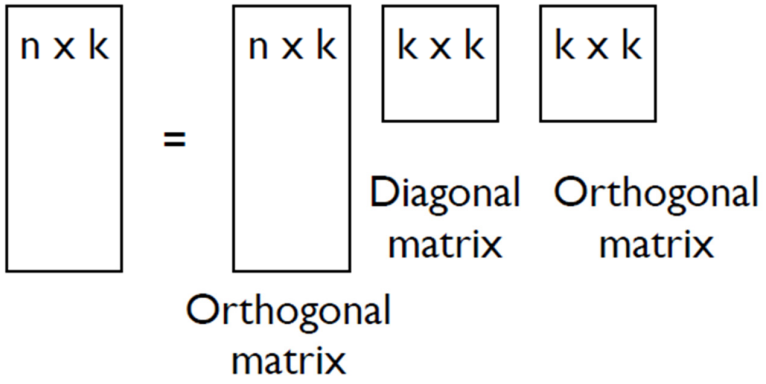
p.ej.
CMA-ES [Covariance Matrix Adaptation Evolution Strategy]

Métodos algorítmicos

Factorización de matrices

Muy utilizada en sistemas de recomendación

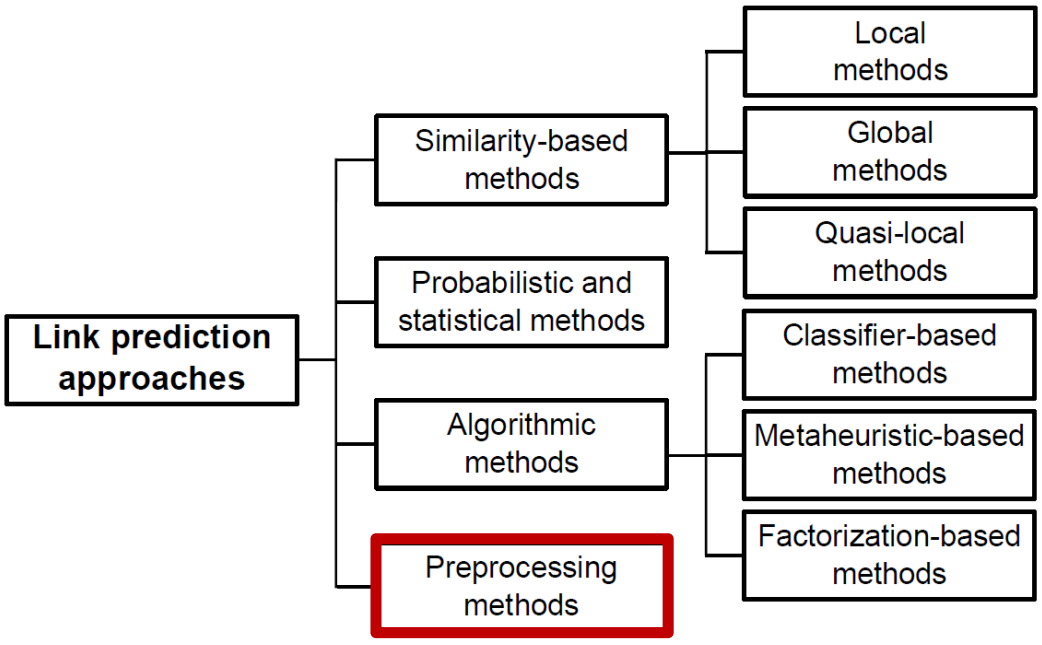
$$\mathbf{X} = \mathbf{U}\mathbf{D}\mathbf{V}^{\mathsf{T}}$$



p.ej.

SVD [Singular Value Decomposition]

Técnicas de preprocesamiento



Técnicas de preprocesamiento

Utillizadas en combinación con otros métodos, pretenden reducir el ruido presente en las redes en forma de enlaces falsos o "débiles".

- **Low-rank approximation** (SVD)
  ~ Extracción de características
- **Unseen bigrams**
  (se reemplaza un nodo por sus nodos más similares)
- **Filtering**
  (eliminación de los enlaces más débiles, determinados con la ayuda de una método de predicción de enlaces).

Apéndice: Clasificación en redes

Existen modelos de clasificación para redes basados en…

- Estructuras locales
  - Vecinos alrededor de un nodo
  - Caminos de longitud fija
- Patrones (subgrafos)
  Cada grafo se caracteriza por un vector X en el que la componente i-ésima representa la frecuencia del patrón i.
- "Decision stumps" & Boosting
- Kernels (p.ej. SVMs)
  - "Random walks" (caminos aleatorios).
  - Asignación local óptima

Agradecimientos

Víctor Martínez

Link Prediction in Networks: Methods and Applications

MSc Thesis, July 2014
Department of Computer Science and Artificial Intelligence
University of Granada (Spain)

Víctor Martínez, Fernando Berzal & Juan-Carlos Cubero:
"Adaptive degree penalization for link prediction"
Journal of Computational Science, 13:1-9, March 2016

Bibliografía

Modelos de redes

- Paul Erdös & Alfred Rényi: **On the evolution of random graphs**. Mathematical Institute of the Hungarian Academy of Sciences, 5:17-61 (1960) reprinted in Duncan, Barabasi & Watts (eds.): "The Structure and Dynamics of Networks"
- Ray Solomonoff & Anatol Rapoport: **Connectivity of random nets**. Bulletin of Mathematical Biophysics, 13:107-117 (1951) reprinted in Duncan, Barabasi & Watts (eds.): "The Structure and Dynamics of Networks"
- Duncan J. Watts & Steven H. Strogatz: **Collective dynamics of 'small-world' networks**. Nature, 393:440-442 (1998)
- Albert-László Barabási & Réka Albert: **Emergence of scaling in random networks**. Science, 286:509-512 (1999)
- Réka Albert, Hawoong Jeong & Albert-László Barabási: **Error and attack tolerance of complex networks**. Nature 406:378-382 (2000)
- M.E.J. Newman, S.H. Strogatz & D.J. Watts: **Random graphs with arbitrary degree distributions and their applications**. Physical Review E, 64:026118 (2001)
- M.E.J. Newman, S.H. Strogatz & D.J. Watts: **Random graphs models of social networks**. PNAS 99:2566-2572 (2002)
- Erzsébet Ravasz & Albert-László Barabási: **Hierarchical organization in complex networks**. Physical Review E, 67:026112 (2003)
- Mark Newman: **The structure and function of complex networks**. SIAM Review 45:167-256 (2003)

Bibliografía

Clasificación en redes

- M. Deshpande, M. Kuramochi, and G. Karypis. **Automated approaches for classifying structures**, BIOKDD'2002
- M. Deshpande, M. Kuramochi, and G. Karypis, **Frequent Sub-structure Based Approaches for Classifying Chemical Compounds**, ICDM'2003
- M. Deshpande, M. Kuramochi, N. Wale, G. Karypis, **Frequent Substructure-Based Approaches for Classifying Chemical Compounds**. IEEE TKDE 17(8): 1036-1050, 2005
- H. Fröhlich, J. Wegner, F. Sieker, and A. Zell, **Optimal Assignment Kernels For Attributed Molecular Graphs**, ICML'2005
- T. Gärtner, P. Flach, and S. Wrobel, **On Graph Kernels: Hardness Results and Efficient Alternatives**, COLT/Kernel'2003
- J. Huan, W. Wang, D. Bandyopadhyay, J. Snoeyink, J. Prins, and A. Tropsha. **Mining spatial motifs from protein structure graphs**, RECOMB'2004
- H. Kashima, K. Tsuda, and A. Inokuchi, **Marginalized Kernels Between Labeled Graphs**, ICML'2003
- T. Kudo, E. Maeda, and Y. Matsumoto, **An Application of Boosting to Graph Classification**, NIPS'2004
- P. Mahé, N. Ueda, T. Akutsu, J. Perret, and J. Vert, **Extensions of Marginalized Graph Kernels**, ICML20'04

Bibliografía – Libros de texto

- David Easley & Jon Kleinberg: **Networks, Crowds, and Markets: Reasoning About a Highly Connected World**. Cambridge University Press, 2010. ISBN 0521195330 http://www.cs.cornell.edu/home/kleinber/networks-book/
- Mark Newman: **Networks: An Introduction**. Oxford University Press, 2010. ISBN 0-19-920665-1
- Matthew O. Jackson: **Social and Economic Networks**, Princeton University Press, 2008. ISBN 0-691-13440-5