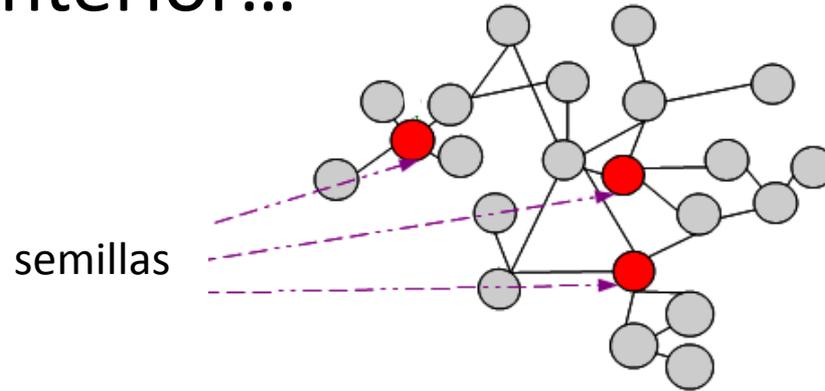
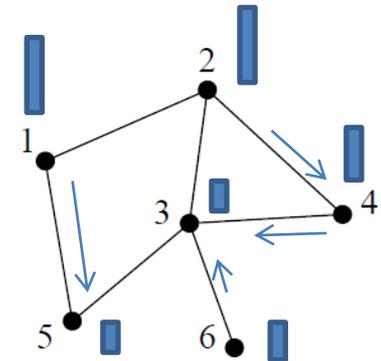


Laplacianos y aprendizaje semisupervisado

En el capítulo anterior...



Functional Flow



$$p_i^{(t+1)} = \sum_{j=1}^N \frac{1}{k_j} a_{ij} p_j^{(t)}$$

$$\mathbf{p}_s^{(t=0)} = (0, \dots, 0, 1, 0, \dots, 0)^T$$

1, ..., s-1, s, s+1, ... N

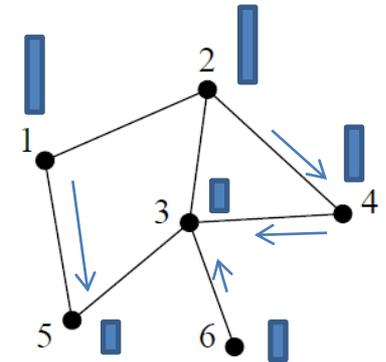
$$\mathbf{p}_s^{(t+1)} = A D^{-1} \mathbf{p}_s^{(t)} \longrightarrow \mathbf{p}_s^{(\tau)}, \text{ con } 0 < \tau \ll \infty$$

$$\frac{dx_i}{dt} = C \sum_{j=1}^N w_{ij} (x_j - x_i) \theta((x_j - x_i))$$

Difusión en redes

Pensamos en un **proceso difusivo** en el que el flujo desde el *nodo-j* al *nodo-i* es **proporcional** a la diferencia de material: $flujo_{j \rightarrow i} \sim C * (x_j - x_i)$

cte de difusión



$$\begin{aligned} \frac{dx_i}{dt} &= C \sum_{j=1}^N A_{ij} (x_j - x_i) \\ &= C \sum_{j=1}^N A_{ij} x_j - C \underbrace{\sum_{j=1}^N A_{ij} x_i}_{k_i} \\ &= C \sum_{j=1}^N A_{ij} x_j - C k_i x_i \\ &= C \sum_{j=1}^N A_{ij} x_j - C \delta_{ij} k_j x_j \end{aligned}$$

$$\frac{dx_i}{dt} = -C \sum_{j=1}^N \underbrace{(\delta_{ij} k_j - A_{ij})}_{[L]_{ij}} x_j$$

$$\frac{dx}{dt} = -C \cdot L \mathbf{x}$$

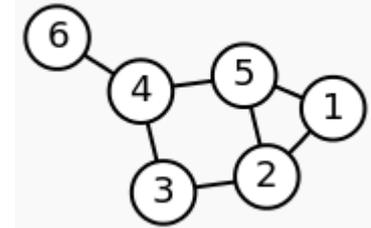
$$\frac{dx}{dt} = C \nabla^2 x$$

Ec de difusion de calor

Difusión en redes

Pensamos en un proceso difusivo en el que el flujo desde el *nodo-j* al *nodo-i* es **proporcional** a la diferencia de material: $flujo_{j \rightarrow i} \sim C * (x_j - x_i)$

cte de difusión



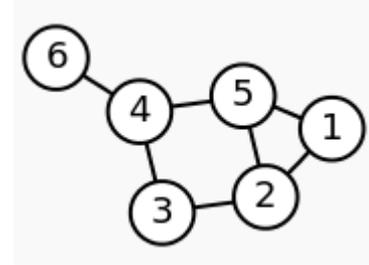
$$\frac{dx_i}{dt} = -C \sum_{j=1}^N \underbrace{(\delta_{ij}k_j - A_{ij})}_{[L]_{ij}} x_j \quad L = D - A$$

Degree matrix	Adjacency matrix	Laplacian matrix
$\begin{pmatrix} 2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 3 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 3 & 0 & 0 \\ 0 & 0 & 0 & 0 & 3 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}$	$\begin{pmatrix} 0 & 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \end{pmatrix}$	$\begin{pmatrix} 2 & -1 & 0 & 0 & -1 & 0 \\ -1 & 3 & -1 & 0 & -1 & 0 \\ 0 & -1 & 2 & -1 & 0 & 0 \\ 0 & 0 & -1 & 3 & -1 & -1 \\ -1 & -1 & 0 & -1 & 3 & 0 \\ 0 & 0 & 0 & -1 & 0 & 1 \end{pmatrix}$

Propiedades

$$L = D - A$$

- L es simétrica
- L es semidefinida-positiva (i.e. autovalores $\lambda_i \geq 0, i = 1 \dots N$)
- Suma por columnas y por filas de L es cero
- Siempre $\lambda = 0$ es autovalor de L con autovector $\mathbf{v}_0 = (1, 1, \dots, 1)$, ya que $L \mathbf{v}_0 = 0 \mathbf{v}_0 = 0$
- L es una matriz singular (i.e. no inversible)



Laplacian matrix					
2	-1	0	0	-1	0
-1	3	-1	0	-1	0
0	-1	2	-1	0	0
0	0	-1	3	-1	-1
-1	-1	0	-1	3	0
0	0	0	-1	0	1

Propiedades

$$L = D - A$$

Laplaciano **combinatorio**

- L es simétrica
- L es semidefinida-positiva (i.e. autovalores $\lambda_i \geq 0, i = 1 \dots N$)
- Suma por columnas y por filas de L es cero
- Siempre $\lambda = 0$ es autovalor de L con autovector $\mathbf{v}_0 = (1, 1, \dots, 1)$, ya que $L \mathbf{v}_0 = 0 \mathbf{v}_0 = 0$
- L es una matriz singular (i.e. no inversible)
- Si el grafo tiene p componentes, de tamaños n_1, n_2, \dots, n_p

$$L = \begin{pmatrix} \boxed{L_1} & 0 & \dots \\ 0 & \boxed{L_2} & \dots \\ \vdots & \vdots & \ddots \end{pmatrix}.$$

Notar que ahora habrá p autovectores de L asociados a $\lambda = 0$ de la forma

$$(\underbrace{1, \dots, 1}_{n_1}, 0, 0, \dots, 0)$$

$$(0, \dots, 0, \underbrace{1, \dots, 1}_{n_2}, 0, 0, \dots, 0)$$

$$(0, \dots, 0, 0, \dots, 0, \underbrace{1, \dots, 1}_{n_3}, 0, 0, \dots, 0)$$

Propiedades

$$L = D - A$$

Laplaciano **combinatorio**

- L es simétrica
- L es semidefinida-positiva (i.e. autovalores $\lambda_i \geq 0, i = 1 \dots N$)
- Suma por columnas y por filas de L es cero
- Siempre $\lambda = 0$ es autovalor de L con autovector $\mathbf{v}_0 = (1, 1, \dots, 1)$, ya que $L \mathbf{v}_0 = 0 \mathbf{v}_0 = 0$
- L es una matriz singular (i.e. no inversible)
- Si el grafo tiene p componentes, de tamaños n_1, n_2, \dots, n_p , existen p autovectores de L asociados a $\lambda = 0$

Corolario:

- el segundo autovalor de L es $\lambda_2 \neq 0$ si el grafo posee una única componente
- λ_2 se denomina **conectividad algebraica** de la red

Autovectores de L



(a) a linear unweighted graph with two segments

$$L\phi_i = \lambda_i \phi_i$$



autovector i-esimo

ϕ_i define un campo
escalar sobre el grafo

Nota1: cualquier campo puede
ser escrito como CL de
autovectores de L

Nota2: autovectores de bajo
índice están asociados a **campos
más suaves** sobre el grafo

Caminatas al azar...y Laplacianos

$$p_i(t + 1) = \sum_{j=1}^N \frac{1}{k_j} a_{ij} p_j(t)$$

$p_i(t)$ probabilidad de encontrar a Juan en el nodo- i , en el paso temporal t

- Esta misma ecuación aplica a procesos de tipo **difusivos** en una red, donde en cada paso temporal toda la cantidad de material que se encuentra en el nodo- i es repartida y enviada a sus vecinos

$$x_i(t + 1) = \sum_{j=1}^N \frac{1}{k_j} a_{ij} x_j(t)$$

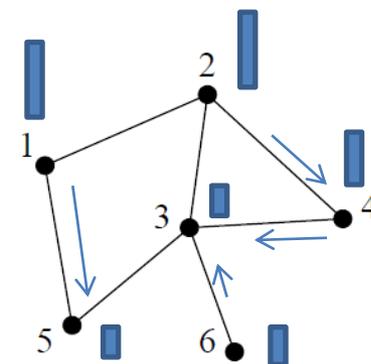
$x_j(t)$ cantidad de material que a tiempo t se encuentra en el nodo- j

el recurso x_j se reparte *extensivamente*

Difusión *random-walk* en redes

A cada paso, la cantidad x_i de cada nodo se altera

$$x_i(t + 1) = \sum_{j=1}^N \frac{1}{k_j} a_{ij} x_j(t)$$



en cada nodo hay una cantidad x_i de material

$$\Delta x_i = x_i(t + 1) - x_i(t) = \sum_{j=1}^N \frac{1}{k_j} a_{ij} x_j(t) - x_i(t)$$

$$= \sum_{j=1}^N \frac{1}{k_j} a_{ij} x_j(t) - \delta_{ij} x_j(t) \quad \delta_{ij} = 1 \text{ si } i = j$$

$$= - \sum_{j=1}^N \underbrace{\left(\delta_{ij} - \frac{1}{k_j} a_{ij} \right)}_{[L^{rw}]_{ij}} x_j(t)$$

← Laplaciano *random-walk*

$$\Delta \mathbf{x} = -L^{rw} \mathbf{x}$$

$$\frac{d\mathbf{x}}{dt} = -L^{rw} \mathbf{x}$$

$$\frac{d\mathbf{x}}{dt} = \nabla^2 \mathbf{x}$$

Laplacianos

Entonces, vimos dos tipos de procesos difusivos:

$$\frac{dx_i}{dt} = -C \sum_{j=1}^N \overbrace{(\delta_{ij}k_j - a_{ij})}^{[L]_{ij}} x_j \quad L = D - A$$

$$\frac{dx_i}{dt} = - \sum_{j=1}^N \overbrace{\left(\delta_{ij} - \frac{1}{k_j} a_{ij} \right)}^{[L^{rw}]_{ij}} x_j(t) \quad L^{rw} = I - D^{-1}A = D^{-1}L$$

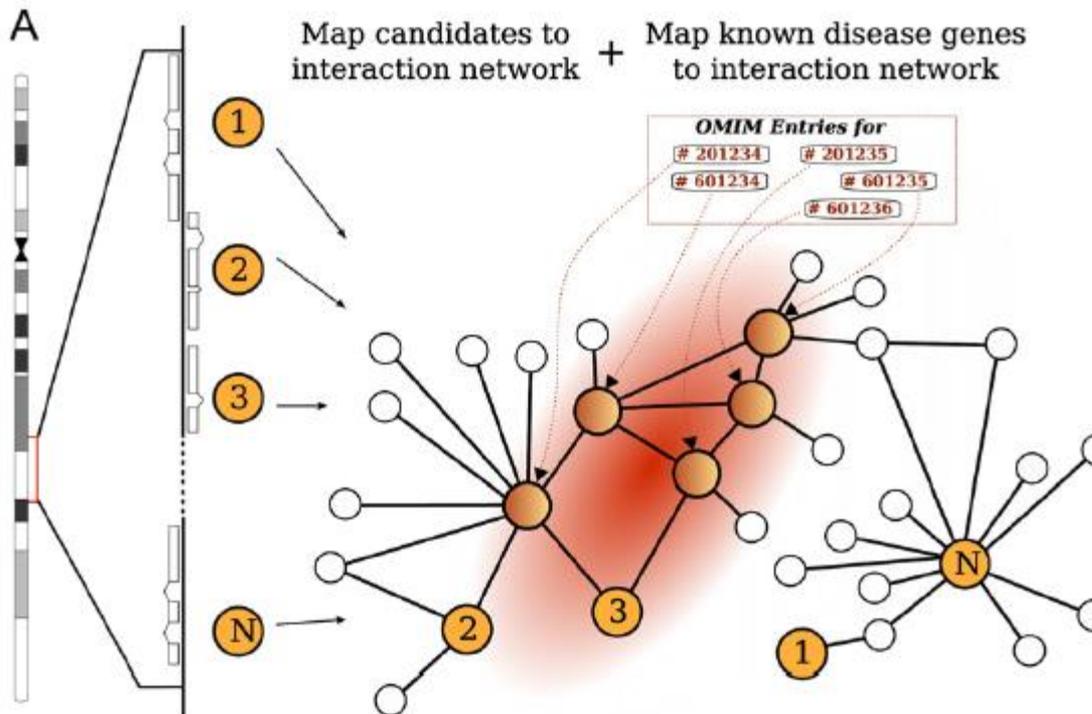
$$L^{sym} \equiv D^{-1/2}LD^{-1/2} = I - D^{-1/2}LD^{-1/2}$$

L : Laplaciano combinatorio o no-normalizado

L^{rw}: Laplaciano random-walk

L^{sym}: Laplaciano normalizado o simetrico

Priorización de nuevas asociaciones gen/enfermedad



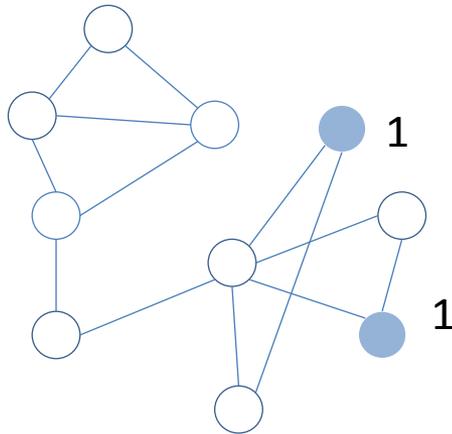
Algoritmos para **propagar sentido de pertenencia** al conjunto de interés:

- Random Walk with Restart** (Kholer 2009)
- Net-Rank** (Chen 2009)
- Net-Propagation** (Vanunu 2010)
- Functional Flow** (Navieba 2005)

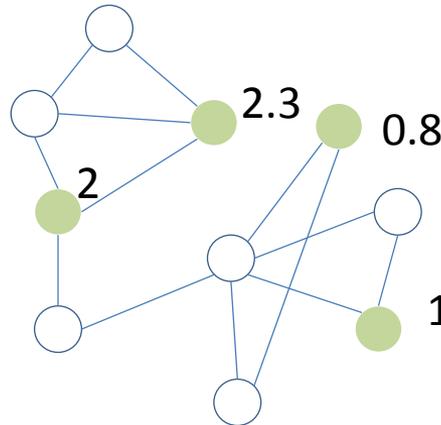
Lo podemos pensar también como un proceso de **difusión con fuentes**

Difusión en redes y aprendizaje semi-supervisado

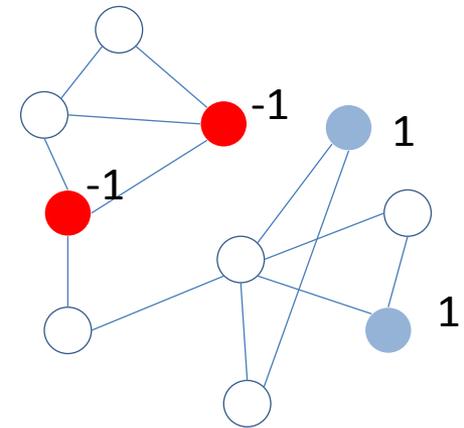
Tengo información parcial asociada a un subconjunto de nodos (etiquetas, o valores reales) y quiero utilizarla para inferir propiedades de los no-etiquetados. Cada nodo va a propagar su etiqueta de manera iterativa hasta converger.



Problema de priorización:
/ nodos etiquetados con valor 1 y $N-l$ nodos con valor 0.



Problema de regresión:
/ nodos etiquetados con valores reales y $N-l$ nodos con valor 0.

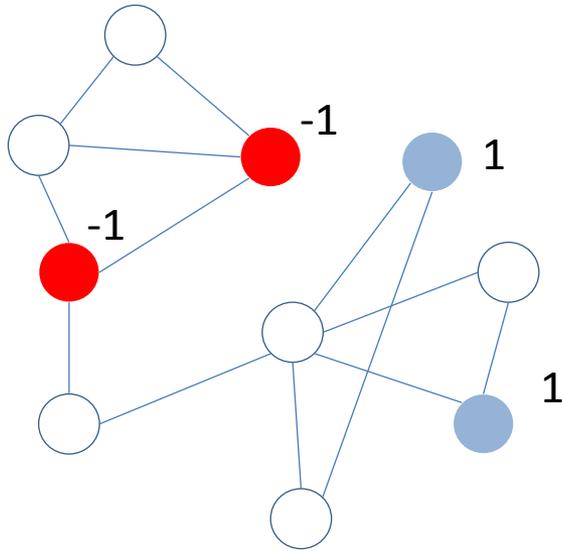


Problema de clasificación:
/ nodos etiquetados con valor 1 (azul) o -1 (rojo) y $N-l$ nodos con valor 0. Nota: para multiclase se define campo vectorial sobre el grafo.

El problema de aprendizaje **semi-supervisado** consiste en encontrar un etiquetado de los nodos del grafo consistente con

- I. El etiquetado inicial (incompleto)
- II. La geometría inducida por la estructura de la red (smoothness or manifold assumption)

Difusión en redes y aprendizaje semi-supervisado



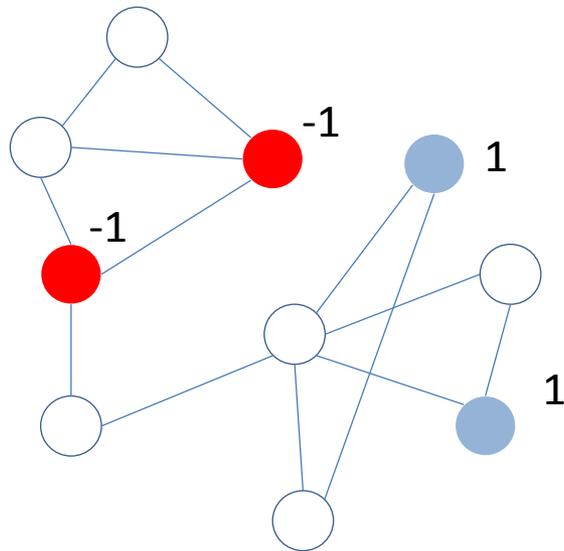
Algoritmo 1 Label Propagation (Zhu 2002)

- Computo de matriz de adyacencia W
- Computo de matriz diagonal D : $D_{ii} \leftarrow \sum_j w_{ij}$
- Inicializo $\hat{Y}^{(t=0)} \leftarrow (y_1, \dots, y_l, 0, 0, \dots, 0)$
- Itero hasta convergencia
 1. $\hat{Y}^{(t+1)} \leftarrow D^{-1}W\hat{Y}^{(t)}$
 2. $\hat{Y}_l^{(t+1)} \leftarrow Y_l$
- Etiqueta del nodo- i resulta $sign(\hat{y}_i^{(\infty)})$

Sea el grafo G ,

- nodos $1, 2, \dots, l$ etiquetados no trivialmente según $Y_l = (y_1, \dots, y_l)$
- nodos $l+1, \dots, N$ etiquetados con valor 0
- Queremos propagar la información por la red y estimar el vector de etiquetas asintótico: $\hat{Y} = (\hat{Y}_l, \hat{Y}_u)$

Difusión en redes y aprendizaje semi-supervisado



Algoritmo 3 Label Propagation (Zhou 2004)

- Computo de matriz de adyacencia W , se fija $w_{ii}=0$
- Computo de matriz diagonal D : $D_{ii} \leftarrow \sum_j w_{ij}$
- Computo Laplaciano simetrico

$$\mathcal{L} \leftarrow D^{-1/2} W D^{-1/2}$$

- Inicializo $\hat{Y}^{(t=0)} \leftarrow (y_1, \dots, y_l, 0, 0, \dots, 0)$
- Elijo $\alpha \in [0, 1)$
- Itero hasta convergencia

Difusión con fuentes $\longrightarrow \hat{Y}^{(t+1)} \leftarrow \alpha \mathcal{L} \hat{Y}^{(t)} + (1 - \alpha) \hat{Y}^{(0)}$

- Etiqueta del nodo- i resulta $sign(\hat{y}_i^{(\infty)})$

Buscando buenas soluciones minimizando cosas

El problema de aprendizaje supervisado consiste en encontrar un etiquetado de los nodos del grafo consistente con

- I. El etiquetado inicial (incompleto)
- II. La geometría inducida por la estructura de la red

Buenas y suaves soluciones

El problema de aprendizaje supervisado consiste en encontrar un etiquetado de los nodos del grafo consistente con

- I. El etiquetado inicial (incompleto)
- II. La geometría inducida por la estructura de la red

Elementos etiquetados

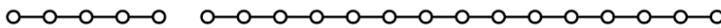
Sea $\hat{Y} = (\hat{Y}_l, \hat{Y}_u)$

no-etiquetados

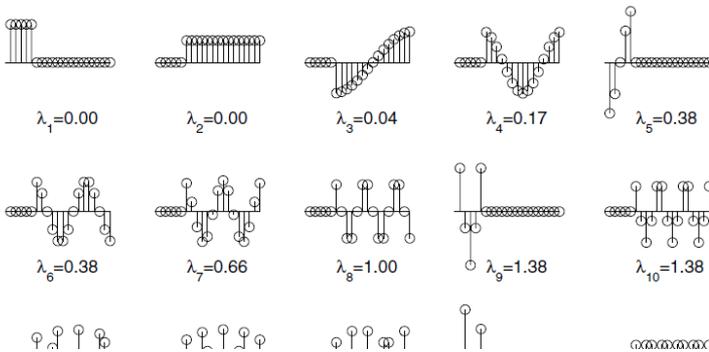
Consistencia con etiquetado inicial:

$$\sum_{i=1}^l (\hat{y}_i - y_i)^2 = \|\hat{Y}_l - Y_l\|^2$$

Laplaciano...



(a) a linear unweighted graph with two segments



funcion suave

$$\check{Y} = \sum_{i=1}^p a_i \phi_i$$

$p < n$

Cualquier campo puede ser escrito como:

$$F = \sum_{i=1}^N a_i \phi_i \quad \text{con} \quad L\phi_i = \lambda_i \phi_i$$

$(\lambda_i < \lambda_{i+1})$

La solución buscada:

$$\check{Y}^* = \operatorname{argmin}_{\{a_1 \dots a_p\}} \|\check{Y}_l - Y_l\|^2$$

Otra manera...regularización en grafos

El problema de aprendizaje supervisado consiste en encontrar un etiquetado de los nodos del grafo consistente con

- I. El etiquetado inicial (incompleto)
- II. La geometría inducida por la estructura de la red

etiquetados

Sea $\hat{Y} = (\hat{Y}_l, \hat{Y}_u)$

no-etiquetados

penalizamos apartamiento de etiquetado original

Consistencia con etiquetado inicial:

$$\sum_{i=1}^l (\hat{y}_i - y_i)^2 = \|\hat{Y}_l - Y_l\|^2$$

Consistencia con geometría de los datos:

$$\frac{1}{2} \sum_{i,j=1}^n \mathbf{W}_{ij} (\hat{y}_i - \hat{y}_j)^2$$

penalizamos cambios bruscos de etiquetado

Regularización en grafos

El problema de aprendizaje supervisado consiste en encontrar un etiquetado de los nodos del grafo consistente con

- I. El etiquetado inicial (incompleto)
- II. La geometría inducida por la estructura de la red

penalizamos apartamiento
de etiquetado original

penalizamos cambios
bruscos de etiquetado

$$\mathcal{H}(\hat{Y}) = \sum_{i=1}^l (\hat{y}_i - y_i)^2 + \frac{1}{2} \sum_{i,j=1}^N w_{ij} (\hat{y}_i - \hat{y}_j)^2$$

$$= \|\hat{Y}_l - Y_l\|^2 + \hat{Y}^T L \hat{Y}$$

Se trata de encontrar el etiquetado \hat{Y} que minimice $\mathcal{H}(\hat{Y})$

En la práctica se suele agregar un término de regularización (que representa un costo basal de etiquetado, para romper degeneración)

$$\mathcal{H}(\hat{Y}) = \|\hat{Y}_l - Y_l\|^2 + \mu \hat{Y}^T L \hat{Y} + \mu \epsilon \|\hat{Y}_l\|^2$$

Buscando buenas soluciones

$$\mathcal{H}(\hat{Y}) = \|\hat{Y}_l - Y_l\|^2 + \mu \hat{Y}^T L \hat{Y} + \mu \epsilon \|\hat{Y}_l\|^2 \quad (1)$$

- Estrategia 1:
 - ✓ Forzar las etiquetas de los datos conocidos: $\hat{Y}_l = Y_l$
 - ✓ Etiquetar **el resto** de manera de minimizar: $\hat{Y}^T L \hat{Y}$
- Estrategia 1B:
 - ✓ No forzar las etiquetas de los datos conocidos
 - ✓ Buscar el etiquetado que minimice $\mathcal{H}(\hat{Y})$

La estrategia 1B es preferible para lidiar con ruido en los datos de entrada

Regularización en grafos

Se trata de encontrar el etiquetado \hat{Y} que minimice $\mathcal{H}(\hat{Y})$

$$\mathcal{H}(\hat{Y}) = \|\hat{Y}_l - Y_l\|^2 + \mu \hat{Y}^T L \hat{Y} + \mu \epsilon \|\hat{Y}_l\|^2$$

↑
 $\|S\hat{Y} - SY\|^2$

$S = I_{[l]}$ matriz diagonal con 1's en los primeros l elementos y 0 el resto

$$\frac{1}{2} \frac{\partial \mathcal{H}(\hat{Y})}{\partial \hat{Y}} = S(\hat{Y} - Y) + \mu L \hat{Y} + \mu \epsilon \hat{Y} = (S + \mu L + \mu \epsilon I) \hat{Y} - SY = 0$$

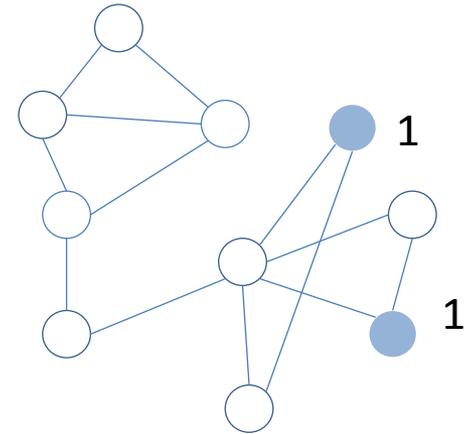
$$\Rightarrow \hat{Y} = (S + \mu L + \mu \epsilon I)^{-1} SY$$

- Las etiquetas *optimas* pueden obtenerse invirtiendo una matriz
- Esta matriz depende únicamente del Laplaciano de la red (no de las etiquetas)
- La manera en que las etiquetas se propagan depende exclusivamente de la estructura del grafo

Vínculo entre regularización y propagación de etiquetas

$$\mathcal{H}(\hat{Y}) = \|S\hat{Y} - SY\|^2 + \mu\hat{Y}^T L\hat{Y} + \mu\epsilon\|\hat{Y}_l\|^2$$

$$\Rightarrow \hat{Y}^* = (S + \mu L + \mu\epsilon I)^{-1}SY$$



Metodo iterativo de Jacobi para invertir una matriz:

$$Ax = b \quad \longrightarrow \quad x = A^{-1}b$$

$$a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n = b_1$$

$$a_{21}x_1 + a_{22}x_2 + \dots + a_{2n}x_n = b_2$$

⋮

$$a_{n1}x_1 + a_{n2}x_2 + \dots + a_{nn}x_n = b_n$$

$$x_1 = \frac{1}{a_{11}}(b_1 - a_{12}x_2 - a_{13}x_3 - \dots - a_{1n}x_n)$$

$$x_2 = \frac{1}{a_{22}}(b_2 - a_{21}x_1 - a_{23}x_3 - \dots - a_{2n}x_n)$$

⋮

$$x_n = \frac{1}{a_{nn}}(b_n - a_{n1}x_1 - a_{n2}x_2 - \dots - a_{n,n-1}x_{n-1})$$

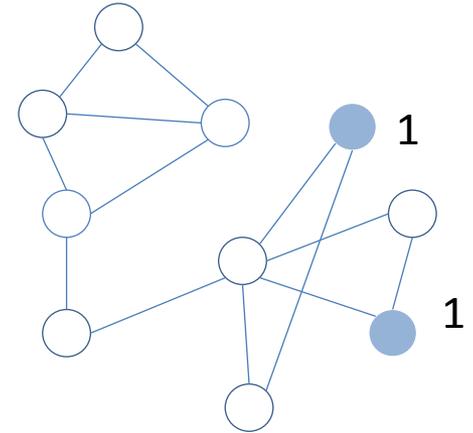
Metodo iterativo

$$x_i^{(t+1)} = \frac{1}{A_{ii}} \left(b - \sum_{j \neq i} A_{ij} x_j^{(t)} \right)$$

Vínculo entre regularización y propagación de etiquetas

$$\mathcal{H}(\hat{Y}) = \|S\hat{Y} - SY\|^2 + \mu\hat{Y}^T L\hat{Y} + \mu\epsilon\|\hat{Y}_l\|^2$$

$$\Rightarrow \hat{Y}^* = (S + \mu L + \mu\epsilon I)^{-1}SY$$



Metodo iterativo de Jacobi para invertir una matriz:

$$Ax = b$$

$$x_i^{(t+1)} = \frac{1}{A_{ii}} \left(b - \sum_{j \neq i} A_{ij} x_j^{(t)} \right)$$

En nuestro caso $x \equiv \hat{Y}, b \equiv SY, A = S + \mu L + \mu\epsilon I$ y la iteración de Jacobi resulta en

para nodo etiquetado

$$\hat{y}_i^{(t+1)} \leftarrow \frac{\sum_j \mathbf{W}_{ij} \hat{y}_j^{(t)} + \frac{1}{\mu} y_i}{\sum_j \mathbf{W}_{ij} + \frac{1}{\mu} + \epsilon}$$

para nodo sin etiquetar

$$\hat{y}_i^{(t+1)} \leftarrow \frac{\sum_j \mathbf{W}_{ij} \hat{y}_j^{(t)}}{\sum_j \mathbf{W}_{ij} + \epsilon}$$

Algoritmo 2 de propagación de etiquetas (!)

Regularización en grafos

Hay otras elecciones para la función a minimizar:

$$\mathcal{H}(\hat{Y}) = \|S\hat{Y} - SY\|^2 + \mu\hat{Y}^T L\hat{Y} + \mu\epsilon\|\hat{Y}_l\|^2$$

$$\mathcal{H}'(\hat{Y}) = \|\hat{Y} - SY\|^2 + \frac{\mu}{2} \sum_{i,j} w_{ij} \left(\frac{\hat{y}_i}{\sqrt{D_{ii}}} - \frac{\hat{y}_j}{\sqrt{D_{jj}}} \right) \underbrace{\|\hat{Y}_l - Y_l\|^2 + \|\hat{Y}_u\|^2}$$

Zhou 2004

Dos diferencias entre $\mathcal{H}(\hat{Y})$ y $\mathcal{H}'(\hat{Y})$:

- I. El primer término busca ajustar bien las etiquetas conocidas pero **además** favorece etiquetas 0 para nodos no-etiquetados inicialmente
- II. Las etiquetas están normalizadas por la raíz cuadrada del grado cuando se computa similaridad entre vecinos.

Se puede ver que

$$\mathcal{H}'(\hat{Y}) = \|\hat{Y} - SY\|^2 + \mu(D^{-1/2}\hat{Y})^T L(D^{-1/2}\hat{Y})$$

Regularización y propagación de etiquetas

$$\mathcal{H}'(\hat{Y}) = \|\hat{Y} - SY\|^2 + \mu(D^{-1/2}\hat{Y})^T L(D^{-1/2}\hat{Y})$$

$$\frac{1}{2} \frac{\partial \mathcal{H}'(\hat{Y})}{\partial \hat{Y}} = \hat{Y} - SY + \mu(\hat{Y} - \mathcal{L}\hat{Y})$$

$$\Rightarrow \hat{Y}^* = \left((1 + \mu)I - \mu \mathcal{L} \right)^{-1} SY$$

La iteración de Jacobi para resolver $\frac{\partial \mathcal{H}'(\hat{Y})}{\partial \hat{Y}} = 0$

$$\hat{Y}^{(t+1)} \leftarrow \alpha \mathcal{L} \hat{Y}^{(t)} + (1 - \alpha) \hat{Y}^{(0)} \quad \text{con } \mu = \frac{\alpha}{1 - \alpha}$$

O sea..

- ✓ Laplacianos: operador sobre el grafo. Asociados a flujos, random walks, etc...
- ✓ Propiedades espectrales de L
- ✓ Aprendizaje supervisado...busqueda de etiquetados (i.e. campos) **piolas**: conocimiento inicial (parcial), geometria del grafo (smoothness assumption).
- ✓ Label propagation
- ✓ Minimizacion de funcion de costo (cuadratico)
- ✓ Label prop \leftrightarrow minimizacion
- ✓ Fin de teoricas :-)

Refs

- Bengio, Delalleau, Le Roux: *Label propagation and quadratic criterion.*