FACULTAD DE INGENIERÍA

UNIVERSIDAD DE LA REPÚBLICA URUGUAY

Aprendizaje Automático para Datos en Grafos

Aprendiendo Grafos desde Datos

Marcelo Fiori
Muy basado en transparencias de **Gonzalo Mateos**

mfiori@fing.edu.uy
http://www.fing.edu.uy/~mfiori/

20 de octubre de 2022

Aprendiendo grafos a partir de señales suaves

1. Métodos estadísticos para inferencia de topología del grafo

2. Aprendiendo grafos a partir de observaciones de señales suaves

Formulación del problema

Motivación

- Buscamos grafos sobre el cual las señales admitan ciertas regularidades
  - Predicción por vecino más cercano (a.k.a. graph smoothing)
  - Aprendizaje semi-supervisado
- Muchos datos de redes del mundo real son suaves
  - Grafos basados en similitudes entre atributos de vértices
  - A menudo, formación de redes se basa en homofilia, proximidad en espacio latente

Formulación del problema

Motivación

- Buscamos grafos sobre el cual las señales admitan ciertas regularidades
  - Predicción por vecino más cercano (a.k.a. graph smoothing)
  - Aprendizaje semi-supervisado
- Muchos datos de redes del mundo real son suaves
  - Grafos basados en similitudes entre atributos de vértices
  - A menudo, formación de redes se basa en homofilia, proximidad en espacio latente

Planteo de problema

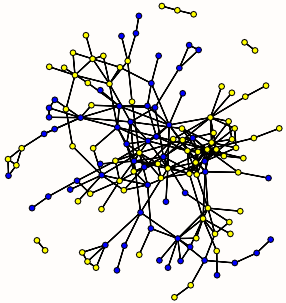
Dadas observaciones $\mathcal{X} := \{\mathbf{x}_p\}_{p=1}^{P}$, identificar un grafo G tal que las señales $\mathcal{X}$ sean suaves en G.

- Criterio: Energía de Dirichlet en el grafo G con Laplaciana $\mathbf{L}$

$$\mathrm{TV}(\mathbf{x}) = \mathbf{x}^\top \mathbf{L} \mathbf{x}$$

Ejemplo: Predecir funciones de proteínas

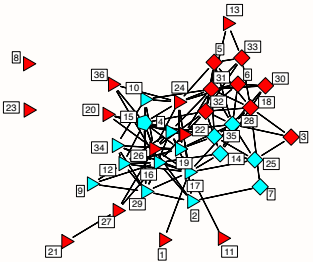
- Datos de levadura, formalmente *Saccharomyces cerevisiae*
  - Grafo: 134 vértices (proteínas) y 241 aristas (interacciones entre proteínas)



- Señal: anotación funcional cascadas de señalización intracelular(ICSC)
  - $x_i = 1$ si proteína i anotada ICSC (amarillo), $x_i = 0$ si no(azul)

Ejemplo: Prácticas de abogados

- Relaciones laborales entre abotados [Lazega'01]
  - Grafo: 36 abogados, aristas indican colaboraciones



- Señal: varios atributos a nivel de nodo $\mathbf{x} = \{x_i\}_{i \in \mathcal{V}}$ incluyendo

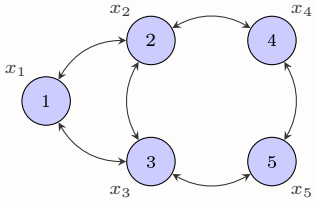
  $\Rightarrow$ Tipo de práctica, i.e., litigante (red) y corporativo (cyan)

- Intuimos que abogados colaboran más con pares de su misma clase de práctica legal

  $\Rightarrow$ El conocimiento de colaboraciones es útil para predecir el tipo de práctica

Digresión II: Graph signal processing (GSP)

- Grafo G con matriz de adyacencia $\mathbf{A} \in \mathbb{R}^{N \times N}$
  $\Rightarrow A_{ij} =$ proximidad entre i y j
- Tenemos una señal $\mathbf{x} \in \mathbb{R}^N$ en el grafo
  $\Rightarrow x_i =$ valor de la señal en el nodo i

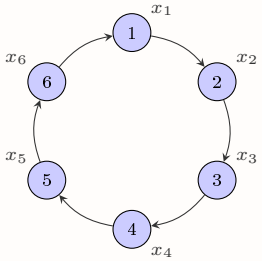


- Graph Signal Processing $\rightarrow$ Explotar la estructura codificada en $\mathbf{A}$ para procesar $\mathbf{x}$

Digresión II: Importancia de la estructura temporal de una señal

- El procesamento de señales se basa en explotar la estructura de la señal

- Tiempo discreto descripto a través de un grafo cíclico
  - $\Rightarrow$ El instante de tiempo n le sigue al $n-1$
  - $\Rightarrow$ El valor x_n está relacionado a x_{n-1}
- Formalizado bajo la noción de frecuencia



- Estructura cíclica $\Rightarrow$ Transformada de Fourier $\Rightarrow \tilde{\mathbf{x}} = \mathbf{F}^H \mathbf{x}$ $\left(F_{kn} = \frac{e^{j2\pi kn/N}}{\sqrt{N}} \right)$
- **Transformada de Fourier** $\Rightarrow$ Proyección en el *eigenvector space* del ciclo

Digresión II: Graph Fourier Transform

- Adyacencia $\mathbf{A}$, Laplaciana $\mathbf{L}$, o genéricamente graph shift $\mathbf{S} = \mathbf{V}\mathbf{\Lambda}\mathbf{V}^{-1}$

  $\Rightarrow S_{ij} = 0$ para $i \neq j$ y $(i,j) \notin \mathcal{E}$ (captura la estructura local en G)

- La Graph Fourier Transform (GFT) de $\mathbf{x}$ se define como

$$\tilde{\mathbf{x}} = \mathbf{V}^{-1}\mathbf{x}$$

- Mientras que la GFT inversa (iGFT) de $\tilde{\mathbf{x}}$ se define como

$$\mathbf{x} = \mathbf{V}\tilde{\mathbf{x}}$$

  $\Rightarrow$ Los vectores propios $\mathbf{V} = [\mathbf{v}_1, ..., \mathbf{v}_N]$ son la base frecuencial (átomos)

Digresión II: Graph Fourier Transform

- Adyacencia $\mathbf{A}$, Laplaciana $\mathbf{L}$, o genéricamente graph shift $\mathbf{S} = \mathbf{V}\mathbf{\Lambda}\mathbf{V}^{-1}$
  - $\Rightarrow S_{ij} = 0$ para $i \neq j$ y $(i,j) \notin \mathcal{E}$ (captura la estructura local en G)
- La Graph Fourier Transform (GFT) de $\mathbf{x}$ se define como

$$\tilde{\mathbf{x}} = \mathbf{V}^{-1}\mathbf{x}$$

- Mientras que la GFT inversa (iGFT) de $\tilde{\mathbf{x}}$ se define como

$$\mathbf{x} = \mathbf{V}\tilde{\mathbf{x}}$$

  - $\Rightarrow$ Los vectores propios $\mathbf{V} = [\mathbf{v}_1, ..., \mathbf{v}_N]$ son la base frecuencial (átomos)
- Estructura adicional
  - $\Rightarrow$ Si $\mathbf{S}$ es normal, entonces $\mathbf{V}^{-1} = \mathbf{V}^H$ y $\tilde{x}_k = \mathbf{v}_k^H \mathbf{x} = \langle \mathbf{v}_k, \mathbf{x} \rangle$
  - $\Rightarrow$ Vale Parseval: $\|\mathbf{x}\|^2 = \|\tilde{\mathbf{x}}\|^2$
- **GFT** $\Rightarrow$ Proyección sobre el espacio de vectores propios de la matriz de shift $\mathbf{S}$

Digresión II: Modos frecuenciales de la Laplaciana

- Total variation de la señal $\mathbf{x}$ con respecto a $\mathbf{L}$

$$\mathrm{TV}(\mathbf{x}) = \mathbf{x}^T\mathbf{L}\mathbf{x} = \sum_{i,j=1,j>i}^{N} A_{ij}(x_i - x_j)^2$$

  $\Rightarrow$ Medida de suavidad de la señal sobre el grafo G (energía de Dirichlet)

- Para los valores propios de la Laplaciana $\mathbf{V} = [\mathbf{v}_1, \cdots, \mathbf{v}_N]$ $\Rightarrow$ $\mathrm{TV}(\mathbf{v}_k) = \lambda_k$

  $\Rightarrow$ Podemos ver a $0 = \lambda_1 < \cdots \leq \lambda_N$ como frecuencias

Digresión II: Modos frecuenciales de la Laplaciana

- Total variation de la señal $\mathbf{x}$ con respecto a $\mathbf{L}$

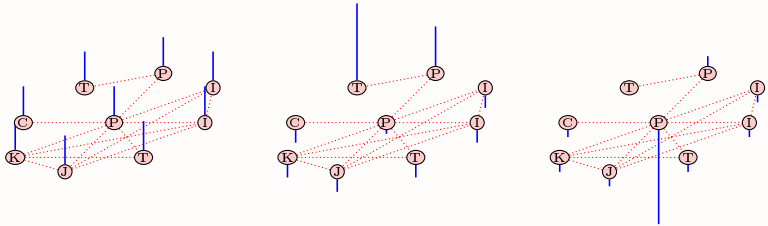
$$\mathrm{TV}(\mathbf{x}) = \mathbf{x}^T \mathbf{L} \mathbf{x} = \sum_{i,j=1, j>i}^{N} A_{ij}(x_i - x_j)^2$$

  $\Rightarrow$ Medida de suavidad de la señal sobre el grafo G (energía de Dirichlet)

- Para los valores propios de la Laplaciana $\mathbf{V} = [\mathbf{v}_1, \cdots, \mathbf{v}_N]$ $\Rightarrow$ $\mathrm{TV}(\mathbf{v}_k) = \lambda_k$

  $\Rightarrow$ Podemos ver a $0 = \lambda_1 < \cdots \leq \lambda_N$ como frecuencias

- Ejemplo: grafo con $N = 10$, $k = 1$, $k = 2$, $k = 9$



Modelo de análisis factorial basado en la Laplaciana

- Consideremos un grafo desconocido G con Laplaciana $\mathbf{L} = \mathbf{V}\mathbf{\Lambda}\mathbf{V}^\top$
  - $\Rightarrow$ Adoptamos la base $\mathbf{V}$ GFT como as signal representation matrix
- Factor-analysis model para la señal observada en el grafo [Dong et al'16]

$$\mathbf{x} = \mathbf{V}\boldsymbol{\chi} + \boldsymbol{\epsilon}$$

  - $\Rightarrow$ Variables latentes $\boldsymbol{\chi} \sim \mathcal{N}(\mathbf{0}, \mathbf{\Lambda}^\dagger)$ ($\approx$ coeficientes GFT)
  - $\Rightarrow$ Término de error isotrópico $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \sigma^2\mathbf{I})$
  - $\Rightarrow$ ¿Cuáles serán los coeficientes más grandes en $\boldsymbol{\chi}$?
  - $\Rightarrow$ ¿asociados a qué valores propios? ¿cómo repercute eso en $\mathbf{x}$?

Modelo de análisis factorial basado en la Laplaciana

- Consideremos un grafo desconocido G con Laplaciana $\mathbf{L} = \mathbf{V}\mathbf{\Lambda}\mathbf{V}^\top$
  - $\Rightarrow$ Adoptamos la base $\mathbf{V}$ GFT como as signal representation matrix
- Factor-analysis model para la señal observada en el grafo [Dong et al'16]

$$\mathbf{x} = \mathbf{V}\boldsymbol{\chi} + \boldsymbol{\epsilon}$$

  - $\Rightarrow$ Variables latentes $\boldsymbol{\chi} \sim \mathcal{N}(\mathbf{0}, \mathbf{\Lambda}^\dagger)$ ($\approx$ coeficientes GFT)
  - $\Rightarrow$ Término de error isotrópico $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \sigma^2\mathbf{I})$
  - $\Rightarrow$ ¿Cuáles serán los coeficientes más grandes en $\boldsymbol{\chi}$?
  - $\Rightarrow$ ¿asociados a qué valores propios? ¿cómo repercute eso en $\mathbf{x}$?
- Suavidad: esto favorece señales "pasa bajo" $\mathbf{x}$
  - $\Rightarrow$ Valores propios chicos de $\mathbf{L}$ (baja frecuencia) $\rightarrow$ Peso alto en los factores

Inferencia como denoising via graph kernel regression

- Estimador *Maximum a posteriori* (MAP) de las variables $\boldsymbol{\chi}$

$$\hat{\boldsymbol{\chi}}_{\text{MAP}} = \arg\min_{\boldsymbol{\chi}} \left\{ \|\mathbf{x} - \mathbf{V}\boldsymbol{\chi}\|^2 + \alpha \boldsymbol{\chi}^\top \boldsymbol{\Lambda} \boldsymbol{\chi} \right\}$$

  $\Rightarrow$ Parameterizado por $\mathbf{V}$ y $\boldsymbol{\Lambda}$, que no conocemos

- Definamos el predictor $\mathbf{y} := \mathbf{V}\boldsymbol{\chi}$, entonces el regularizador queda:

$$\boldsymbol{\chi}^\top \boldsymbol{\Lambda} \boldsymbol{\chi} = \mathbf{y}^\top \mathbf{V} \boldsymbol{\Lambda} \mathbf{V}^\top \mathbf{y} = \mathbf{y}^\top \mathbf{L} \mathbf{y} = \text{TV}(\mathbf{y})$$

  $\Rightarrow$ Denoiser (basado en TV Laplaciano) de $\mathbf{x}$, prior de suavidad en $\mathbf{y}$

Inferencia como denoising via graph kernel regression

- Estimador *Maximum a posteriori* (MAP) de las variables $\boldsymbol{\chi}$

$$\hat{\boldsymbol{\chi}}_{\text{MAP}} = \arg\min_{\boldsymbol{\chi}} \left\{ \|\mathbf{x} - \mathbf{V}\boldsymbol{\chi}\|^2 + \alpha \boldsymbol{\chi}^\top \boldsymbol{\Lambda} \boldsymbol{\chi} \right\}$$

  $\Rightarrow$ Parameterizado por $\mathbf{V}$ y $\boldsymbol{\Lambda}$, que no conocemos

- Definamos el predictor $\mathbf{y} := \mathbf{V}\boldsymbol{\chi}$, entonces el regularizador queda:

$$\boldsymbol{\chi}^\top \boldsymbol{\Lambda} \boldsymbol{\chi} = \mathbf{y}^\top \mathbf{V} \boldsymbol{\Lambda} \mathbf{V}^\top \mathbf{y} = \mathbf{y}^\top \mathbf{L} \mathbf{y} = \text{TV}(\mathbf{y})$$

  $\Rightarrow$ Denoiser (basado en TV Laplaciano) de $\mathbf{x}$, prior de suavidad en $\mathbf{y}$

- **Idea:** buscar simultáneamente $\mathbf{L}$ y la representación sin ruido (denoised) $\mathbf{y} = \mathbf{V}\boldsymbol{\chi}$

$$\min_{\mathbf{L}, \mathbf{y}} \left\{ \|\mathbf{x} - \mathbf{y}\|^2 + \alpha \mathbf{y}^\top \mathbf{L} \mathbf{y} \right\}$$

Formulación y algoritmo

- Dadas señales $\mathcal{X} := \{\mathbf{x}_p\}_{p=1}^{P}$ en $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_P] \in \mathbb{R}^{N \times P}$, resolver

$$\min_{\mathbf{L},\mathbf{Y}} \left\{ \|\mathbf{X} - \mathbf{Y}\|_F^2 + \alpha \text{trace}\left(\mathbf{Y}^\top \mathbf{L} \mathbf{Y}\right) + \frac{\beta}{2} \|\mathbf{L}\|_F^2 \right\}$$
$$\text{s. to} \quad \text{trace}(\mathbf{L}) = N,\ \mathbf{L}\mathbf{1} = \mathbf{0},\ L_{ij} = L_{ji} \leq 0,\ i \neq j$$

  - $\Rightarrow$ Hay que fijar "la escala" ($trace(\mathbf{L}) = N$). Si no $\mathbf{L} = 0$ y $\mathbf{X} = \mathbf{Y}$ es solución
  - $\Rightarrow$ Es fácil ver que $trace(L) = N$ fija la norma ℓ_1 de $\mathbf{L}$
  - $\Rightarrow$ Función objetivo: Ajuste a datos + suavidad + sparsity en aristas
  - $\Rightarrow$ No conjuntamente convexa en $\mathbf{L}$ y $\mathbf{Y}$, pero bi-convexa

Formulación y algoritmo

- Dadas señales $\mathcal{X} := \{\mathbf{x}_p\}_{p=1}^P$ en $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_P] \in \mathbb{R}^{N \times P}$, resolver

$$\min_{\mathbf{L},\mathbf{Y}} \left\{ \|\mathbf{X} - \mathbf{Y}\|_F^2 + \alpha \text{trace}\left(\mathbf{Y}^\top \mathbf{L} \mathbf{Y}\right) + \frac{\beta}{2} \|\mathbf{L}\|_F^2 \right\}$$
$$\text{s. to} \quad \text{trace}(\mathbf{L}) = N,\ \mathbf{L}\mathbf{1} = \mathbf{0},\ L_{ij} = L_{ji} \leq 0,\ i \neq j$$

  - $\Rightarrow$ Hay que fijar "la escala" ($trace(\mathbf{L}) = N$). Si no $\mathbf{L} = 0$ y $\mathbf{X} = \mathbf{Y}$ es solución
  - $\Rightarrow$ Es fácil ver que $trace(L) = N$ fija la norma ℓ_1 de $\mathbf{L}$
  - $\Rightarrow$ Función objetivo: Ajuste a datos + suavidad + sparsity en aristas
  - $\Rightarrow$ No conjuntamente convexa en $\mathbf{L}$ y $\mathbf{Y}$, pero bi-convexa

- Enfoque algorítmico: minimización alternada (AM)

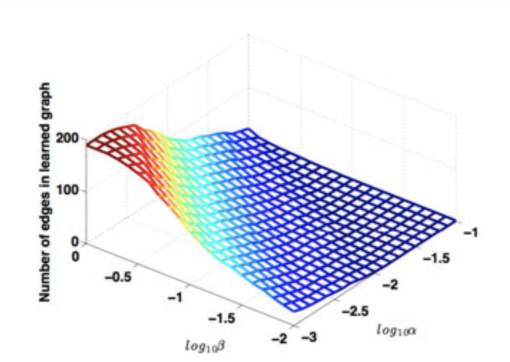
  (S1) Fijado $\mathbf{Y}$: resolver para $\mathbf{L}$ via interior-point methods, ADMM

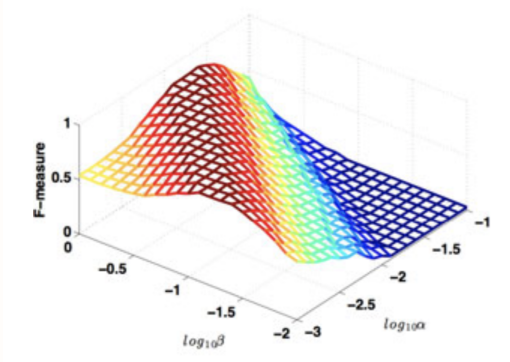
  (S2) Fijado $\mathbf{L}$, solución cerrada: filtro pasa-bajos, versión suavizada de $\mathbf{X}$

$$\mathbf{Y} = (\mathbf{I} + \alpha \mathbf{L})^{-1} \mathbf{X}$$

Impacto de los parámetros en esparsidad y accuracy

- Se generan muchas señales en un grafo sintético Erdős-Rényi
  $\Rightarrow$ Se infiere el grafo para distintos valores de α and β

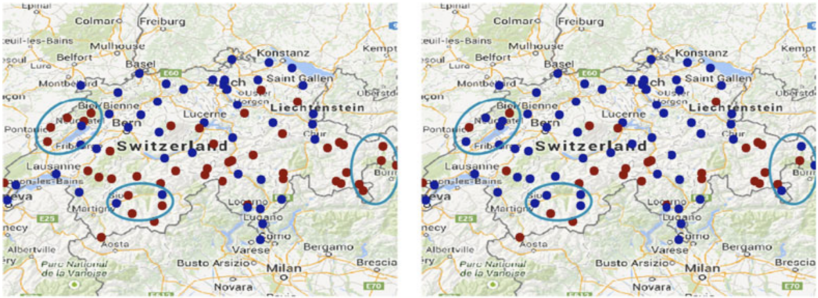




- Hay más aristas si crece β y si decrece α
- Si hay poco ruido, el cociente β/α determina la performance

Ejemplo: Temperatura en Suiza

- $N = 89$ estaciones meteorológicas con mediciones mensuales de temperatura (1981-2010)
  $\Rightarrow$ Aprender un grafo G donde las temperaturas varíen suavemente
- La distancia geográfica puede no ser una buena idea $\Rightarrow$ diferentes alturas



- Partición en dos usando spectral clustering en G
  $\Rightarrow$ Clusters: rojo (estaciones en altura) y azul (estaciones bajas)
- Kmeans aplicado directo a las temperaturas (derecha) falla

Signal smoothness meets edge sparsity

- Recordemos $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_P] \in \mathbb{R}^{N \times P}$, sea $\bar{\mathbf{x}}_i^\top \in \mathbb{R}^{1 \times P}$ la fila número i
  - $\Rightarrow$ Matriz de distancia Euclidea $\mathbf{Z} \in \mathbb{R}_+^{N \times N}$, donde $Z_{ij} := \|\bar{\mathbf{x}}_i - \bar{\mathbf{x}}_j\|^2$
- Truquito: relación entre suavidad y esparsidad [Kalofolias'16]

$$\sum_{p=1}^{P} \mathrm{TV}(\mathbf{x}_p) = \mathrm{trace}(\mathbf{X}^\top \mathbf{L} \mathbf{X}) = \frac{1}{2} \|\mathbf{A} \circ \mathbf{Z}\|_1$$

  - $\Rightarrow$ Aristas $\mathcal{E}$ esparsas cuando los datos son suaves
  - $\Rightarrow$ Favorece aristas candidatas (i, j) asociadas a valores chicos de Z_{ij}

Signal smoothness meets edge sparsity

- Recordemos $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_P] \in \mathbb{R}^{N \times P}$, sea $\bar{\mathbf{x}}_i^\top \in \mathbb{R}^{1 \times P}$ la fila número i
  - $\Rightarrow$ Matriz de distancia Euclidea $\mathbf{Z} \in \mathbb{R}_+^{N \times N}$, donde $Z_{ij} := \|\bar{\mathbf{x}}_i - \bar{\mathbf{x}}_j\|^2$
- Truquito: relación entre suavidad y esparsidad [Kalofolias'16]

$$\sum_{p=1}^{P} \mathrm{TV}(\mathbf{x}_p) = \mathrm{trace}(\mathbf{X}^\top \mathbf{L} \mathbf{X}) = \frac{1}{2} \|\mathbf{A} \circ \mathbf{Z}\|_1$$

  - $\Rightarrow$ Aristas $\mathcal{E}$ esparsas cuando los datos son suaves
  - $\Rightarrow$ Favorece aristas candidatas (i, j) asociadas a valores chicos de Z_{ij}
- Parameteriza el problema de aprender el grafo en términos de $\mathbf{A}$ (en vez de $\mathbf{L}$)
  - $\Rightarrow$ Ventajas porque las restricciones en $\mathbf{A}$ quedan desacopladas

Formulación del método

- Modelo de aprendizaje de grafos propuesto [Kalofolias'16]

$$\min_{\mathbf{A}} \left\{ \|\mathbf{A} \circ \mathbf{Z}\|_1 - \alpha \mathbf{1}^\top \log(\mathbf{A}\mathbf{1}) + \frac{\beta}{2}\|\mathbf{A}\|_F^2 \right\}$$
$$\text{s. to} \quad \operatorname{diag}(\mathbf{A}) = \mathbf{0},\ A_{ij} = A_{ji} \geq 0,\ i \neq j$$

$\Rightarrow$ Barrera logarítmica fuerza grado de nodos no nulo

$\Rightarrow$ Penaliza pesos grandes para controlar esparsidad

Formulación del método

- Modelo de aprendizaje de grafos propuesto [Kalofolias'16]

$$\min_{\mathbf{A}} \left\{ \|\mathbf{A} \circ \mathbf{Z}\|_1 - \alpha \mathbf{1}^\top \log(\mathbf{A}\mathbf{1}) + \frac{\beta}{2}\|\mathbf{A}\|_F^2 \right\}$$
$$\text{s. to} \quad \operatorname{diag}(\mathbf{A}) = \mathbf{0},\ A_{ij} = A_{ji} \geq 0,\ i \neq j$$

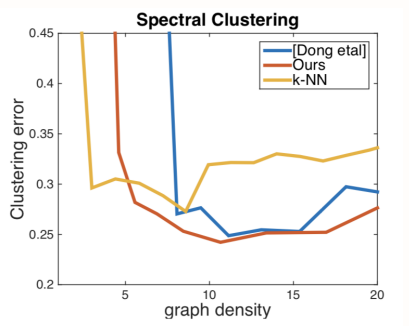
  - $\Rightarrow$ Barrera logarítmica fuerza grado de nodos no nulo
  - $\Rightarrow$ Penaliza pesos grandes para controlar esparsidad

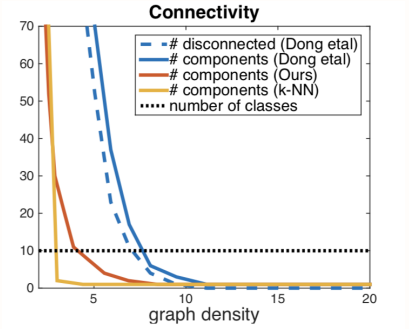
- Método primal-dual que se puede paralelizar, costo $O(N^2)$
- Laplacian-based factor analysis (Dong) revisitado. Se formula **(S1)** como

$$\min_{\mathbf{A}} \left\{ \|\mathbf{A} \circ \mathbf{Z}\|_1 - \log(\mathbb{I}\{\|\mathbf{A}\|_1 = N\}) + \frac{\beta}{2}\left(\|\mathbf{A}\mathbf{1}\|^2 + \|\mathbf{A}\|_F^2\right) \right\}$$
$$\text{s. to} \quad \operatorname{diag}(\mathbf{A}) = \mathbf{0},\ A_{ij} = A_{ji} \geq 0,\ i \neq j$$

Ejemplo: aprender el grafo de dígitos de USPS

- 1001 imágenes de los 10 dígitos, altamente desbalanceados ($2{,}6i^2$)
  - $\Rightarrow$ 10 clases via aprender el grafo + spectral clustering
- Comparamos los métodos basados en suavidad que vimos, más el grafo k-NN





- Performance más robusta a la densidad del grafo
  - $\Rightarrow$ Posiblemente debido a no tener nodos aislados

Aprendizaje de grafos seleccionando aristas

- **Idea:** parameterizar la topología a encontrar con un vector indicatriz de aristas

Aprendizaje de grafos seleccionando aristas

- **Idea:** parameterizar la topología a encontrar con un vector indicatriz de aristas
- Tomemos el grafo completo con N nodos, que tiene $M := \binom{N}{2}$ aristas
  $\Rightarrow$ La matriz de incidencia $\mathbf{B} := [\mathbf{b}_1, \ldots, \mathbf{b}_M] \in \mathbb{R}^{N \times M}$
- La Laplaciana del grafo candidato $G(\mathcal{V}, \mathcal{E})$ [Chepuri et al'17]

$$\mathbf{L}(\boldsymbol{\omega}) = \sum_{m=1}^{M} \omega_m \mathbf{b}_m \mathbf{b}_m^\top$$

  $\Rightarrow$ Vector binario indicatriz de aristas $\boldsymbol{\omega} := [\omega_1, \ldots, \omega_M]^\top \in \{0, 1\}^M$
  $\Rightarrow$ Ofrece control explícito sobre el número de aristas $\|\boldsymbol{\omega}\|_0 = |\mathcal{E}|$

Aprendizaje de grafos seleccionando aristas

- **Idea:** parameterizar la topología a encontrar con un vector indicatriz de aristas
- Tomemos el grafo completo con N nodos, que tiene $M := \binom{N}{2}$ aristas
  $\Rightarrow$ La matriz de incidencia $\mathbf{B} := [\mathbf{b}_1, \ldots, \mathbf{b}_M] \in \mathbb{R}^{N \times M}$
- La Laplaciana del grafo candidato $G(\mathcal{V}, \mathcal{E})$ [Chepuri et al'17]

$$\mathbf{L}(\boldsymbol{\omega}) = \sum_{m=1}^{M} \omega_m \mathbf{b}_m \mathbf{b}_m^\top$$

  $\Rightarrow$ Vector binario indicatriz de aristas $\boldsymbol{\omega} := [\omega_1, \ldots, \omega_M]^\top \in \{0,1\}^M$
  $\Rightarrow$ Ofrece control explícito sobre el número de aristas $\|\boldsymbol{\omega}\|_0 = |\mathcal{E}|$

Problema: Dadas observaciones $\mathcal{X} := \{\mathbf{x}_p\}_{p=1}^P$, aprender un grafo sin pesos $G(\mathcal{V}, \mathcal{E})$ tal que señales de $\mathcal{X}$ sean suaves en G y $|\mathcal{E}| = K$.

Optimización booleana con restricción de cardinalidad

- La formulación natural es resolver el problema (no convexo)

$$\min_{\boldsymbol{\omega}\in\{0,1\}^M} \operatorname{trace}(\mathbf{X}^\top \mathbf{L}(\boldsymbol{\omega})\mathbf{X}), \quad \text{s. to } \|\boldsymbol{\omega}\|_0 = K$$

- La solución se obtiene via un simple procedimiento de orden de rankings
  - Calcular scores para cada arista $c_m := \operatorname{trace}(\mathbf{X}^\top(\mathbf{b}_m\mathbf{b}_m^\top)\mathbf{X})$
  - Poner $\omega_m = 1$ para las K aristas con menor score

Optimización booleana con restricción de cardinalidad

- La formulación natural es resolver el problema (no convexo)

$$\min_{\boldsymbol{\omega}\in\{0,1\}^M} \operatorname{trace}(\mathbf{X}^\top \mathbf{L}(\boldsymbol{\omega})\mathbf{X}), \quad \text{s. to } \|\boldsymbol{\omega}\|_0 = K$$

- La solución se obtiene via un simple procedimiento de orden de rankings
  - Calcular scores para cada arista $c_m := \operatorname{trace}(\mathbf{X}^\top(\mathbf{b}_m\mathbf{b}_m^\top)\mathbf{X})$
  - Poner $\omega_m = 1$ para las K aristas con menor score
- Versión más realista con ruido aditivo (AWGN) $\mathbf{x}_p = \mathbf{y}_p + \boldsymbol{\epsilon}_p$, $p = 1, \ldots, P$

$$\min_{\mathbf{Y},\boldsymbol{\omega}\in\{0,1\}^M} \left\{\|\mathbf{X}-\mathbf{Y}\|_F^2 + \alpha \operatorname{trace}(\mathbf{Y}^\top \mathbf{L}(\boldsymbol{\omega})\mathbf{Y})\right\}, \quad \text{s. to } \|\boldsymbol{\omega}\|_0 = K$$

$\Rightarrow$ Se puede resolver con AM o semidefinite relaxation (SDR)

Resumen comparativo

- Propiedades del enfoque de selección de aristas
  - ✓ Controlamos directamente la esparsidad de aristas
  - ✓ Algoritmo simple en el caso sin ruido
  - ✓ No hay que imponer restricciones para obtener una Laplaciana
  - ✗ No garantiza conectividad de G
  - ✗ No podemos incluir pesos en las aristas

Resumen comparativo

- Propiedades del enfoque de selección de aristas
  - ✓ Controlamos directamente la esparsidad de aristas
  - ✓ Algoritmo simple en el caso sin ruido
  - ✓ No hay que imponer restricciones para obtener una Laplaciana
  - ✗ No garantiza conectividad de G
  - ✗ No podemos incluir pesos en las aristas
- Volviendo al framework de [Kalofolias'16], es más flexible

$$\begin{aligned} &\min_{\mathbf{A}} \{\|\mathbf{A} \circ \mathbf{Z}\|_1 + g(\mathbf{A})\} \\ &\text{s. to} \quad \operatorname{diag}(\mathbf{A}) = \mathbf{0},\ A_{ij} = A_{ji} \geq 0,\ i \neq j \end{aligned}$$

  - $\Rightarrow$ Engloba el factor-analysis model [Dong et al'16]
  - $\Rightarrow$ Recupera pesos de kernel gaussiano $A_{ij} := \exp\left(-\frac{\|\bar{\mathbf{x}}_i - \bar{\mathbf{x}}_j\|^2}{\sigma^2}\right)$ para

$$g(\mathbf{A}) = \sigma^2 \sum_{i,j} A_{ij}(\log(A_{ij}) - 1)$$

Case study: Clasificación

- Señales etiquetadas de grafos $\mathcal{X}_c := \{\mathbf{x}_p^{(c)}\}_{p=1}^{P_c}$ de C clases diferentes
  - $\Rightarrow$ Las señales en cada clase tienen una estructura que las distingue
- **Asumimos:** Señales de clase c son suaves respecto a un grafo (desconocido) $G_c(\mathcal{V}, \mathcal{E}_c)$
- Modelo de subespacios lineales múltiples
  - $\Rightarrow$ Señales generadas por unos pocos modos Laplacianos (componentes GFT)
  - $\Rightarrow$ Como *susbpace clustering* [Vidal'11], pero supervisado

Planteo del problema

Dadas señales de entrenamiento $\mathcal{X} = \bigcup_{c=1}^{C} \mathcal{X}_c$, aprender grafos discriminativos $\mathbf{A}_c$ bajo prior de suavidad, para luego clasificar señales de test via proyecciones de GFT.

Discriminative graph learning

- Aprendizaje del grafo discriminativo c [Saboksayr et al'21]

$$\min_{\mathbf{A}_c} \left\{ \|\mathbf{A}_c \circ \mathbf{Z}_c\|_1 - \alpha \mathbf{1}^\top \log(\mathbf{A}_c \mathbf{1}) + \frac{\beta}{2} \|\mathbf{A}_c\|_F^2 - \gamma \sum_{k \neq c}^{C} \|\mathbf{A}_c \circ \mathbf{Z}_k\|_1 \right\}$$

$$\text{s. to} \quad \text{diag}(\mathbf{A}_c) = \mathbf{0},\ [\mathbf{A}_c]_{ij} = [\mathbf{A}_c]_{ji} \geq 0,\ i \neq j$$

$\Rightarrow$ Captura la topología subyacente (estructura de clase c)

$\Rightarrow$ Discriminabilidad para mejorar la performance de clasificación

Discriminative graph learning

- Aprendizaje del grafo discriminativo c [Saboksayr et al'21]

$$\min_{\mathbf{A}_c} \left\{ \|\mathbf{A}_c \circ \mathbf{Z}_c\|_1 - \alpha \mathbf{1}^\top \log(\mathbf{A}_c \mathbf{1}) + \frac{\beta}{2}\|\mathbf{A}_c\|_F^2 - \gamma \sum_{k \neq c}^{C} \|\mathbf{A}_c \circ \mathbf{Z}_k\|_1 \right\}$$

$$\text{s. to} \quad \operatorname{diag}(\mathbf{A}_c) = \mathbf{0},\ [\mathbf{A}_c]_{ij} = [\mathbf{A}_c]_{ji} \geq 0,\ i \neq j$$

  - $\Rightarrow$ Captura la topología subyacente (estructura de clase c)
  - $\Rightarrow$ Discriminabilidad para mejorar la performance de clasificación

- **Q:** Dados grafos $\{\hat{\mathbf{A}}_c\}_{c=1}^{C}$, ¿cómo clasificamos una señal de test $\mathbf{x}$?

- Pasamos $\mathbf{x}$ por un banco de C filtros pasa-bajos (LPFs)

$$\tilde{\mathbf{x}}_{F,c} = \operatorname{diag}(\tilde{\mathbf{h}})\hat{\mathbf{V}}_c^\top \mathbf{x} \quad \Rightarrow \quad \hat{c} = \operatorname*{argmax}_c \left\{ \|\tilde{\mathbf{x}}_{F,c}\|^2 \right\}$$

  - $\Rightarrow$ respuesta frecuencial LPF $\tilde{\mathbf{h}}$, base aprendida GFT para la clase c $\hat{\mathbf{V}}_c$

Reconocer emociones a partir de EEG

- Aprendizaje discriminativo de grafos para reconocer emociones a partir de señales EEG
- DEAP dataset $\Rightarrow$ 32 personas miran videos musicales (40 cada una)
  - Se les pide calificar videos: valencia, activación, like/dislike, dominancia
  - Foco en etiquetas valence: baja (1-5 rating) y alta (6-10 rating)
  - Señales adquiridas con EEG de $N = 32$ canales
- Para cada persona, se hace clasificación de valencia

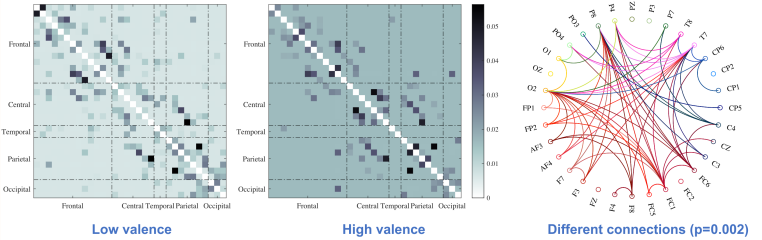
  $\Rightarrow$ Se aprenden $C = 2$ grafos y se proyectan las señales en los 8 modos más suaves

  $\Rightarrow$ Se mide con leave-one (trial)-out classification accuracy
- El accuracy medio es 92,73 %

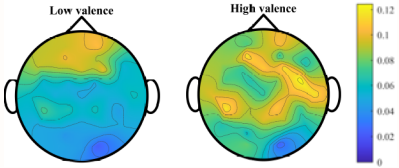
S. S. Saboksayr et al, "Online discriminative graph learning from multi-class smooth signals," *Signal Processing, 186, 108101,* 2021

Clasificación de valencia

- **Q:** ¿Qué información obtenemos de los grafos para cada clase?



- Conectividad se incrementa con la intensidad de la emoción (enlaces del lóbulo frontal)



- Promedio de los 8 modos más suaves: se puede ver diferencia en la actividad frontal