Network Streams, Embeddings, and Topology Learning

Gonzalo Mateos

Dept. of ECE and Goergen Institute for Data Science
University of Rochester
gmateosb@ece.rochester.edu
http://www.ece.rochester.edu/~gmateosb/

Ack.: NSF Awards CCF-1750428, ECCS-1809356, CCF-1934962

December 15, 2021

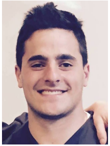
Credits

PhD students

Yang Li
EE, UR

Bernardo Marenco
Math, UdelaR

Saman Saboksayr
EE, UR

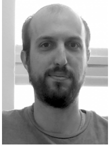


Max Wasserman
CS, UR

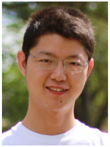
Collaborators

Paola Bermolen
Math, UdelaR

Marcelo Fiori
Math, UdelaR



Federico Larroca
EE, UdelaR



Zhengwu Zhang
Stats, UNC C. Hill

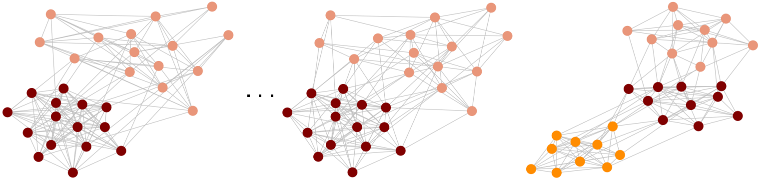
Online network change point detection

Online change point detection for random dot product graphs

Accelerated topology identification from smooth signals

Problem statement

- **Given:** Stream of undirected graph observations $\{\mathbf{A}[t]\}$



- **Model:** Random Dot Product Graph (RDPG) [Athreya et al'17]
- **Goal:** Detect in an online fashion when the underlying model changed

- Contributions and impact
  - $\Rightarrow$ Marry sequential chage-point detection with graph representation learning
  - $\Rightarrow$ Explainable algorithm for (pseudo) real-time network monitoring
  - $\Rightarrow$ Guaranteed error-rate control, insights on detection delay

Random dot product graphs

- Consider a latent space $\mathcal{X}_d \subset \mathbb{R}^d$ such that for all

$$\mathbf{x}, \mathbf{y} \in \mathcal{X}_d \quad \Rightarrow \quad \mathbf{x}^\top \mathbf{y} \in [0, 1]$$

$\Rightarrow$ Inner-product distribution $F : \mathcal{X}_d \mapsto [0, 1]$

- Random dot product graphs (RDPGs) are defined as follows:

$$\mathbf{x}_1, \ldots, \mathbf{x}_N \overset{\text{i.i.d.}}{\sim} F,$$
$$A_{ij} \,\big|\, \mathbf{x}_i, \mathbf{x}_j \sim \text{Bernoulli}(\mathbf{x}_i^\top \mathbf{x}_j)$$

for $1 \leq i, j \leq N$, where $A_{ij} = A_{ji}$ and $A_{ii} \equiv 0$

- A particularly tractable latent position random graph model
  - $\Rightarrow$ Vertex positions $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_N]^\top \in \mathbb{R}^{N \times d}$
  - $\Rightarrow$ Connection probabilities $\mathbf{P} = \mathbf{X}\mathbf{X}^\top$

S. J. Young and E. R. Scheinerman, "Random dot product graph models for social networks," *WAW*, 2007

Connections to other models

- RDPGs ecompass several other classic models for network graphs

  Ex: Erdös-Renyi $G_{N,p}$ graphs with $d = 1$ and $\mathcal{X}_d = \{\sqrt{p}\}$

  Ex: SBM random graphs by constructing F with pmf

$$\mathrm{P}\left(\mathbf{X} = \mathbf{x}_q\right) = \alpha_q, \quad q = 1, \ldots, Q$$

after selecting d and $\mathbf{x}_1, \ldots, \mathbf{x}_Q$ such that $\pi_{qr} = \mathbf{x}_q^\top \mathbf{x}_r$

- Approximation results for SBMs justify the expressiveness of RDPGs
- RDPGs are special cases of latent position models [Hoff et al'02]

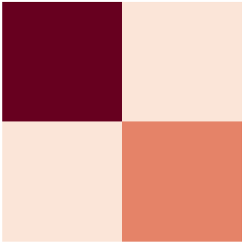
$$A_{ij} \,\big|\, \mathbf{x}_i, \mathbf{x}_j \sim \mathrm{Bernoulli}(\kappa(\mathbf{x}_i, \mathbf{x}_j))$$

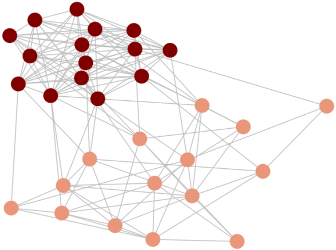
  $\Rightarrow$ Approximate these accurately for large enough d [Tang et al'13]

- **Q:** Given a graph $\mathbf{A}$, how do we estimate the latent positions $\mathbf{X}$?

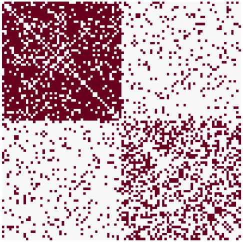
Adjacency spectral embedding

$\mathbf{P} = \mathbf{X}\mathbf{X}^\top$

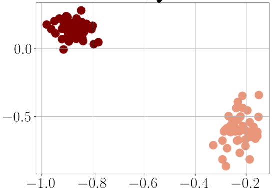




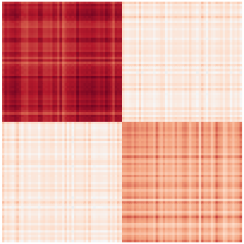
$\mathbf{A} = \mathbf{Q}\Lambda\mathbf{Q}^\top$



$\hat{\mathbf{X}} = \hat{\mathbf{Q}}\hat{\Lambda}^{1/2}$

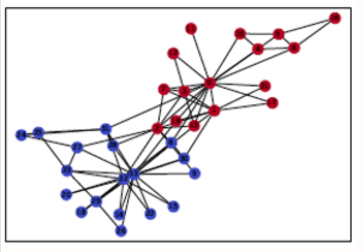


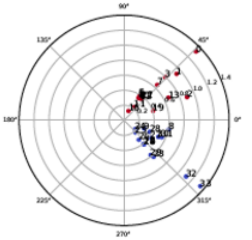
$\hat{\mathbf{P}} = \hat{\mathbf{X}}\hat{\mathbf{X}}^\top$



Interpretability of the embeddings

- Ex: Zachary's karate club graph with $N = 34$ (left)





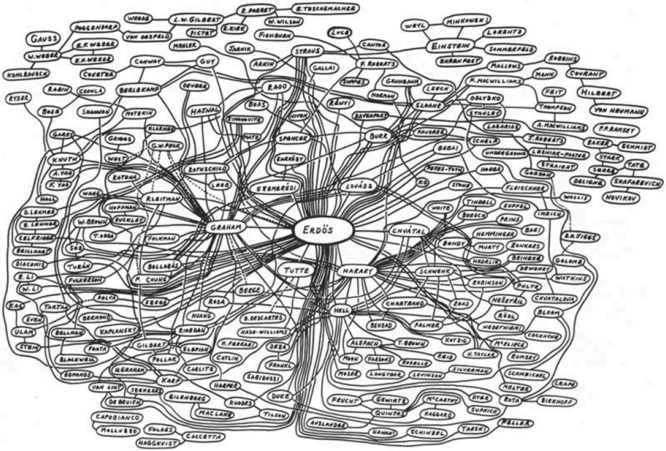
- Node embeddings (rows of $\hat{\mathbf{X}}$) for $d = 2$ (right)
  - Club's administrator ($i = 0$) and instructor ($j = 33$) are orthogonal
- Interpretability of embeddings a valuable asset for RDPGs

  $\Rightarrow$ Magnitudes indicate how well connected nodes are

  $\Rightarrow$ Angles indicate positions in latent space (affinity to link)

Mathematicians collaboration graph

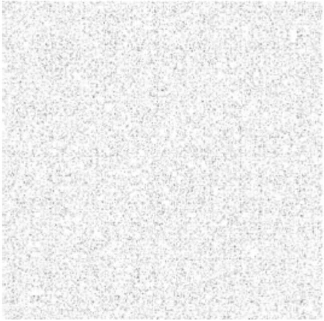
- Ex: Mathematics collaboration network centered at Paul Erdös

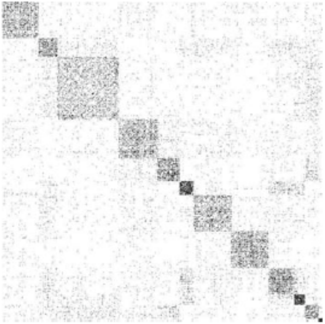


- Most mathematicians have an Erdös number of at most 4 or 5
  - ⇒ Drawing created by R. Graham in 1979

Mathematicians collaboration graph

- Coauthorship graph $G(\mathcal{V}, \mathcal{E})$, $N = 4301$ nodes with Erdös number ≤ 2
  $\Rightarrow$ No discernible structure from the adjacency matrix $\mathbf{A}$ (left)

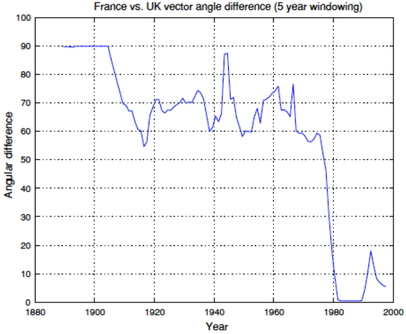




- Community structure revealed after row-colum permutation (right)
  (i) Obtained the ASE $\hat{\mathbf{X}}$ for the mathematicians
  (ii) Performed angular k-means on $\hat{\mathbf{X}}$'s rows [Scheinerman-Tucker'10]

International relations

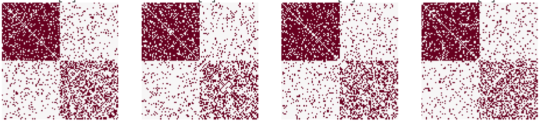
- Ex: Dynamic network G_t of international relations among nations
  - $\Rightarrow$ Nations $(i, j) \in \mathcal{E}_t$ if they have an alliance treaty during year t



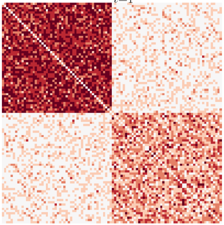
- Track the angle between UK and France's ASE from 1890-1995
  - Orthogonal during the late 19th century
  - Came closer during the wars, retreat during Nazi invasion in WWII
  - Strong alignment starts in the 1970s in the run up to the EU

- **Idea:** Estimating function approach [Kirsch-Tadjuidje'15]
  - $\Rightarrow$ Training set of m "clean" graphs with no change point

$\mathbf{A}[1]$ $\mathbf{A}[2]$ $\mathbf{A}[3]$ $\mathbf{A}[4]$

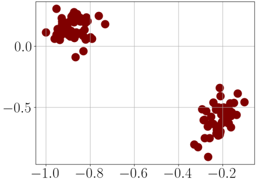


$$\bar{\mathbf{A}} = \frac{1}{m}\sum_{t=1}^{m} \mathbf{A}[t]$$



$\longrightarrow$

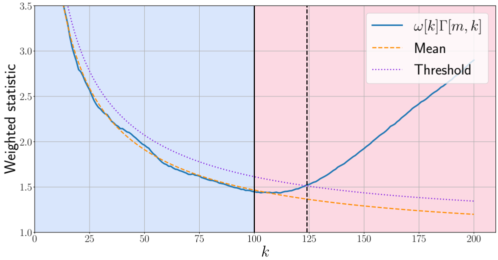
$\hat{\mathbf{X}}$



Online change point detection: Monitoring

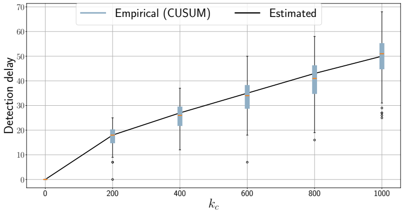
- Sequentially observe matrices $\mathbf{A}[m+1], \mathbf{A}[m+2], \ldots$
  - Monitor the cumulative sum $\mathbf{S}[m,k] = \sum_{t=m+1}^{m+k} \left(\hat{\mathbf{X}}\hat{\mathbf{X}}^\top - \mathbf{A}[t] \right)$

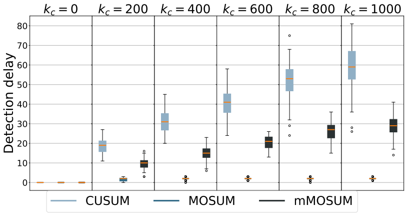
Proposition: For large k and under the null hypothesis, $\Gamma[m,k] := \|\mathbf{S}[m,k]\|^2$ has a generalized χ^2 distribution.



Delay and change detectability

- **Q:** Can we get insights on the incurred detection delay?
  Solution $k^* \geq k_c$ of $\omega[k^*]\mathbb{E}_{ac}[\Gamma[m,k^*]] \geq \mathrm{th}[k^*]$



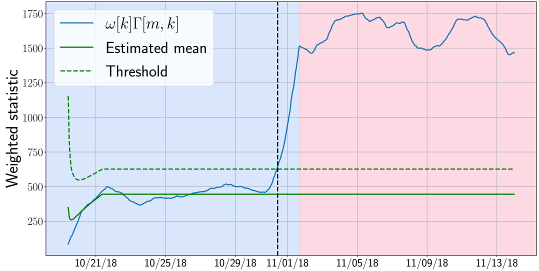


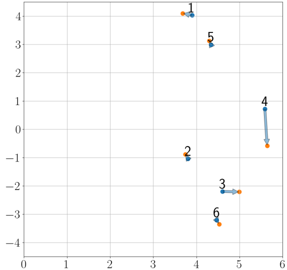
- **Q:** Under what conditions will we miss a change?
  Need to have a small model "perturbation-to-imperfection" ratio

Wireless network monitoring

- Extended RDPG to handle weighted, directed networks
- Real network of Wi-Fi APs. Hourly RSSI measurements for $N = 6$ nodes

  $\Rightarrow$ Ground-truth from network admin: *AP 4 was moved on 10/30*

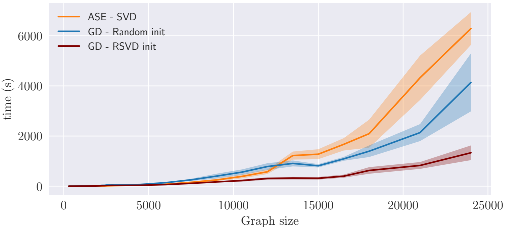


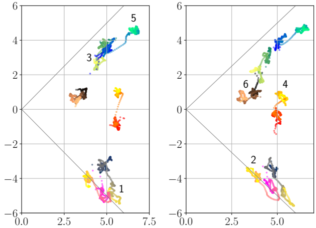


- Explainability via interpretable ASE $\Rightarrow$ Identify source of change
- Reproducibility $\Rightarrow$ Try it @ `https://github.com/git-artes/cpd_rdpg`

Outlook

- Non-convex gradient-based ASE for scalability and model tracking
  - Handle missing data, aligned embeddings via warm restarts





- Embeddings and online change-point detection from graph signals
- Statistical properties of non-parametric weighted RDPG

$$\mathbb{E}\left[e^{tA_{ij}}|\mathbf{X}\right] = \sum_{m=0}^{\infty} \frac{t^m \mathbb{E}\left[A_{ij}^m\right]}{m!} = 1 + \sum_{m=1}^{\infty} \frac{t^m \mathbf{x}_i^\top[m]\mathbf{x}_j[m]}{m!}$$

Network topology inference

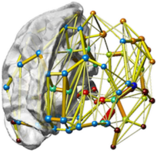
Online change point detection for random dot product graphs

Accelerated topology identification from smooth signals

Learning graphs from data

- Learning graphs from nodal observations
- Ex: Central to network neuroscience

  $\Rightarrow$ Functional network from fMRI signals



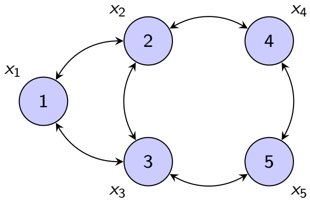
- Most GSP works: how known graph $G(\mathcal{V}, \mathcal{E})$ affects signals and filters
  - Feasible for e.g., physical or infrastructure networks
  - Links are tangible and directly observable
- Still, acquisition of updated topology information is challenging

  $\Rightarrow$ Sheer size, reconfiguration, privacy and security
- Here, reverse path: how to use GSP to infer the graph topology?
- **Goal:** fast, scalable algorithm with convergence rate guarantees

See also arXiv:2110.09677 [cs.LG]

Graph signal processing (GSP)

- Graph G with adjacency matrix $\mathbf{W} \in \mathbb{R}^{N \times N}$
  - $\Rightarrow W_{ij}$ = proximity between i and j
- Define a signal $\mathbf{x} \in \mathbb{R}^{N}$ on top of the graph
  - $\Rightarrow x_i$ = signal value at node $i \in \mathcal{V}$



- Total variation of signal $\mathbf{x}$ with respect to Laplacian $\mathbf{L} = \mathbf{D} - \mathbf{W}$

$$\text{TV}(\mathbf{x}) = \mathbf{x}^\top \mathbf{L} \mathbf{x} = \frac{1}{2} \sum_{i \neq j} W_{ij} (x_i - x_j)^2$$

- Graph Signal Processing $\rightarrow$ Exploit structure encoded in $\mathbf{L}$ to process $\mathbf{x}$
  - $\Rightarrow$ Use GSP to learn the underlying G or a meaningful network model

Problem formulation

Rationale

- Seek graphs on which data admit certain regularities
  - Nearest-neighbor prediction (a.k.a. graph smoothing)
  - Semi-supervised learning
  - Efficient information-processing transforms

- Many real-world graph signals are smooth (i.e., $\text{TV}(\mathbf{x})$ is small)
  - Graphs based on similarities among vertex attributes
  - Network formation driven by homophily, proximity in latent space

Problem statement

Given observations $\mathcal{X} := \{\mathbf{x}_p\}_{p=1}^P$, identify a graph G such that signals in $\mathcal{X}$ are smooth on G.

Signal smoothness meets edge sparsity

- Form $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_P] \in \mathbb{R}^{N \times P}$, let $\bar{\mathbf{x}}_i^\top \in \mathbb{R}^{1 \times P}$ denote its i-th row

  $\Rightarrow$ Euclidean distance matrix $\mathbf{E} \in \mathbb{R}_+^{N \times N}$, where $E_{ij} := \|\bar{\mathbf{x}}_i - \bar{\mathbf{x}}_j\|^2$

- Neat trick: link between smoothness and sparsity [Kalofolias'16]

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

  $\Rightarrow$ Sparse $\mathcal{E}$ when data come from a smooth manifold

  $\Rightarrow$ Favor candidate edges (i, j) associated with small E_{ij}

- Shows that edge sparsity on top of smoothness is redundant

- Parameterize graph learning problems in terms of $\mathbf{W}$ (instead of $\mathbf{L}$)

  $\Rightarrow$ Advantageous since constraints on $\mathbf{W}$ are decoupled

Scalable topology identification framework

- General purpose graph-learning framework

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

$\Rightarrow$ Logarithmic barrier forces positive degrees $\mathbf{d} = \mathbf{W}\mathbf{1}$

$\Rightarrow$ Penalize large edge-weights to control sparsity

- Efficient algorithms incurring $O(N^2)$ cost

$\Rightarrow$ Primal-dual (PD) [Kalofolias'16] and ADMM [Wang et al'21]

- Cost has no Lipschitz gradient $\rightarrow$ No convergence rates

V. Kalofolias, "How to learn a graph from smooth signals," *AISTATS*, 2016

Equivalent reformulation

- Handle constraints on entries of $\mathbf{W}$
  - Hollow and symmetric $\rightarrow$ Retain $\mathbf{w} := \mathrm{vec}[\mathrm{triu}[\mathbf{W}]] \in \mathbb{R}_+^{N(N-1)/2}$
  - Non-negative $\rightarrow \mathbb{I}\{\mathbf{w} \succeq \mathbf{0}\} = 0$ if $\mathbf{w} \succeq \mathbf{0}$, else $\mathbb{I}\{\mathbf{w} \succeq \mathbf{0}\} = \infty$
- Equivalent unconstrained, non-differentiable reformulation

$$\min_{\mathbf{w}} \left\{ \underbrace{\mathbb{I}\{\mathbf{w} \succeq \mathbf{0}\} + 2\mathbf{w}^\top \mathbf{e} + \beta\|\mathbf{w}\|_2^2}_{:=f(\mathbf{w})} - \underbrace{\alpha \mathbf{1}^\top \log(\mathbf{S}\mathbf{w})}_{:=-g(\mathbf{S}\mathbf{w})} \right\}$$

$\Rightarrow$ $\mathbf{S}$ maps edge weights to nodal degrees, i.e., $\mathbf{d} = \mathbf{S}\mathbf{w}$

- Non-differentiable $f(\mathbf{w})$ is strongly convex, $g(\mathbf{d})$ is strictly convex
  - Problem $\min_{\mathbf{w}}\{f(\mathbf{w}) + g(\mathbf{S}\mathbf{w})\}$ has a unique optimal solution $\mathbf{w}^\star$
  - Amenable to fast dual-based proximal gradient (FDPG) solver

A. Beck and M. Teboulle, "A fast dual proximal gradient algorithm for convex minimization and applications," *Oper. Res. Lett.*, 2014

Dual problem and its properties

- Variable splitting: $\min_{\mathbf{w},\mathbf{d}} \{f(\mathbf{w}) + g(\mathbf{d})\}$, s. to $\mathbf{d} = \mathbf{S}\mathbf{w}$
  - Attach Lagrange multipliers $\boldsymbol{\lambda} \in \mathbb{R}^N$ to equality constraints
  - Lagrangian $\mathcal{L}(\mathbf{w}, \mathbf{d}, \boldsymbol{\lambda}) = f(\mathbf{w}) + g(\mathbf{d}) - \langle \boldsymbol{\lambda}, \mathbf{S}\mathbf{w} - \mathbf{d} \rangle$

- (Minimization form) dual problem is $\min_{\boldsymbol{\lambda}} \{F(\boldsymbol{\lambda}) + G(\boldsymbol{\lambda})\}$, where

$$F(\boldsymbol{\lambda}) := \max_{\mathbf{w}} \left\{ \langle \mathbf{S}^\top \boldsymbol{\lambda}, \mathbf{w} \rangle - f(\mathbf{w}) \right\},$$

$$G(\boldsymbol{\lambda}) := \max_{\mathbf{d}} \left\{ \langle -\boldsymbol{\lambda}, \mathbf{d} \rangle - g(\mathbf{d}) \right\}$$

- Strong convexity of f implies a Lipschitz gradient property for F

Lemma. Function $F(\boldsymbol{\lambda})$ is smooth, and the gradient $\nabla F(\boldsymbol{\lambda})$ is Lipschitz continuous with constant $L := \frac{N-1}{\beta}$.

Fast dual-based proximal gradient method

- **Key:** apply accelerated proximal gradient method to the dual

$$\boldsymbol{\lambda}_k = \mathbf{prox}_{L^{-1}G}\left(\boldsymbol{\omega}_k - \frac{1}{L}\nabla F(\boldsymbol{\omega}_k)\right),$$
$$t_{k+1} = \frac{1+\sqrt{1+4t_k^2}}{2},$$
$$\boldsymbol{\omega}_{k+1} = \boldsymbol{\lambda}_k + \left(\frac{t_k - 1}{t_{k+1}}\right)[\boldsymbol{\lambda}_k - \boldsymbol{\lambda}_{k-1}]$$

- Rewrite in terms of problem parameters L, α, β, $\mathbf{S}$, signals in $\mathbf{e}$

Proposition. The dual variable update iteration can be equivalently rewritten as $\boldsymbol{\lambda}_k = \boldsymbol{\omega}_k - L^{-1}(\mathbf{S}\bar{\mathbf{w}}_k - \mathbf{u}_k)$, with

$$\bar{\mathbf{w}}_k = \max\left(\mathbf{0}, \frac{\mathbf{S}^\top\boldsymbol{\omega}_k - 2\mathbf{e}}{2\beta}\right),$$
$$\mathbf{u}_k = \frac{\mathbf{S}\bar{\mathbf{w}}_k - L\boldsymbol{\omega}_k + \sqrt{(\mathbf{S}\bar{\mathbf{w}}_k - L\boldsymbol{\omega}_k)^2 + 4\alpha L\mathbf{1}}}{2}$$

Algorithm summary

Algorithm 1: Topology inference via fast dual PG (FDPG)

Input parameters α, β, data $\mathbf{e}$, set $L = \frac{N-1}{\beta}$.
Initialize $t_1 = 1$ and $\boldsymbol{\omega}_1 = \boldsymbol{\lambda}_0$ at random.
for $k = 1, 2, \ldots,$ **do**

$$\bar{\mathbf{w}}_k = \max\left(\mathbf{0}, \frac{\mathbf{S}^\top \boldsymbol{\omega}_k - 2\mathbf{e}}{2\beta}\right)$$

$$\mathbf{u}_k = \frac{\mathbf{S}\bar{\mathbf{w}}_k - L\boldsymbol{\omega}_k + \sqrt{(\mathbf{S}\bar{\mathbf{w}}_k - L\boldsymbol{\omega}_k)^2 + 4\alpha L \mathbf{1}}}{2}$$

$$\boldsymbol{\lambda}_k = \boldsymbol{\omega}_k - L^{-1}(\mathbf{S}\bar{\mathbf{w}}_k - \mathbf{u}_k)$$

$$t_{k+1} = \frac{1 + \sqrt{1 + 4t_k^2}}{2}$$

$$\boldsymbol{\omega}_{k+1} = \boldsymbol{\lambda}_k + \left(\frac{t_k - 1}{t_{k+1}}\right)[\boldsymbol{\lambda}_k - \boldsymbol{\lambda}_{k-1}]$$

end

Output graph estimate $\hat{\mathbf{w}}_k = \max\left(\mathbf{0}, \frac{\mathbf{S}^\top \boldsymbol{\lambda}_k - 2\mathbf{e}}{2\beta}\right)$

- Complexity of $O(N^2)$ on par with state-of-the-art algorithms
- Non-accelerated dual proximal gradient (DPG) method for $t_k \equiv 1$, $k \geq 1$

Convergence rate analysis

- Let $\boldsymbol{\lambda}^\star$ be a minimizer of the dual cost $\varphi(\boldsymbol{\lambda}) := F(\boldsymbol{\lambda}) + G(\boldsymbol{\lambda})$. Then

$$\varphi(\boldsymbol{\lambda}_k) - \varphi(\boldsymbol{\lambda}^\star) \leq \frac{2(N-1)\|\boldsymbol{\lambda}_0 - \boldsymbol{\lambda}^\star\|_2^2}{\beta k^2}$$

$\Rightarrow$ Celebrated $O(1/k^2)$ rate for FISTA [Beck-Teboulle'09]

- Construct a primal sequence $\hat{\mathbf{w}}_k = \text{argmin}_{\mathbf{w}} \mathcal{L}(\mathbf{w}, \mathbf{d}, \boldsymbol{\lambda}_k)$

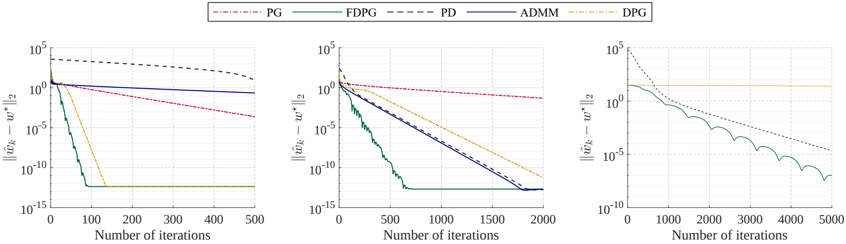
$$\hat{\mathbf{w}}_k = \underset{\mathbf{w}}{\text{argmax}} \left\{ \langle \mathbf{S}^\top \boldsymbol{\lambda}_k, \mathbf{w} \rangle - f(\mathbf{w}) \right\} = \max\left(\mathbf{0}, \frac{\mathbf{S}^\top \boldsymbol{\lambda}_k - 2\mathbf{e}}{2\beta}\right)$$

Theorem. For all $k \geq 1$, the primal sequence $\hat{\mathbf{w}}_k$ defined in terms of dual iterates $\boldsymbol{\lambda}_k$ generated by Algorithm 1 satistifies

$$\|\hat{\mathbf{w}}_k - \mathbf{w}^\star\|_2 \leq \frac{\sqrt{2(N-1)}\|\boldsymbol{\lambda}_0 - \boldsymbol{\lambda}^\star\|_2}{\beta k}.$$

Convergence performance

- Recovery of random and real-world graphs from simulated signals
  - Networks: (a) SBM, $N = 400$; (b) brain, $N = 66$; (c) MN road, $N = 2642$
  - Signals: $P = 1000$ i.i.d. smooth signals $\mathbf{x}_p \sim \mathcal{N}(\mathbf{0}, \mathbf{L}^{\dagger} + 10^{-2}\mathbf{I}_N)$
  - Examine evolution of primal variable error $\|\hat{\mathbf{w}}_k - \mathbf{w}^{\star}\|_2$



(a) (b) (c)

- FDPG converges markedly faster, uniformly across graph classes

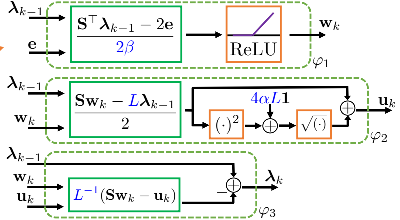
Try it out! `http://www.ece.rochester.edu/~gmateosb/code/FDPG.zip`

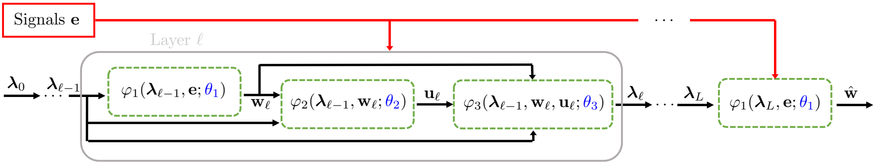
Outlook

- Learning graph topologies via algorithm unrolling

Algorithm 1: Dual PG (DPG)

Input parameters α, β, data $\mathbf{e}$, set $L = \frac{N-1}{\beta}$.
Initialize $\boldsymbol{\lambda}_0$ at random.
for $k = 1, 2, \ldots,$ **do**
$\quad \mathbf{w}_k = \max\left(\mathbf{0}, \frac{\mathbf{S}^\top \boldsymbol{\lambda}_{k-1} - 2\mathbf{e}}{2\beta}\right)$
$\quad \mathbf{u}_k = \frac{\mathbf{S}\mathbf{w}_k - L\boldsymbol{\lambda}_{k-1} + \sqrt{(\mathbf{S}\mathbf{w}_k - L\boldsymbol{\lambda}_{k-1})^2 + 4\alpha L \mathbf{1}}}{2}$
$\quad \boldsymbol{\lambda}_k = \boldsymbol{\lambda}_{k-1} - L^{-1}(\mathbf{S}\mathbf{w}_k - \mathbf{u}_k)$
end
Output graph estimate $\hat{\mathbf{w}}_k = \max\left(\mathbf{0}, \frac{\mathbf{S}^\top \boldsymbol{\lambda}_k - 2\mathbf{e}}{2\beta}\right)$





- Online dynamic graph learning from streaming signals
  - Challenge: dual problem is not strongly convex

See also arXiv:2103.03762 [cs.LG]