Large Scale Graph Learning from Smooth Signals

Vassilis Kalofolias and Nathanaël Perraudin

Laboratoire de traitement des signaux 2, EPFL — Swiss Data Science Center, ETH Zurich and EPFL, Switzerland

SPARSITY

smoothness of data ⇒ sparsity of edges

$$\mathrm{tr}\left(X^\top L X\right) = \frac{1}{2}\|W \circ Z\|_{1,1}$$

$$Z_{ij} = \|x_i - x_j\|^2$$

OPTIMIZATION PROBLEM

$$\min_{W \in \mathcal{W}_m} \|W \circ Z\|_{1,1} - \alpha \mathbf{1}^\top \log(W\mathbf{1}) + \beta\|W\|_F^2$$

node degrees

prevent zero degrees, allow sparse W

control sparsity

ALGORITHM

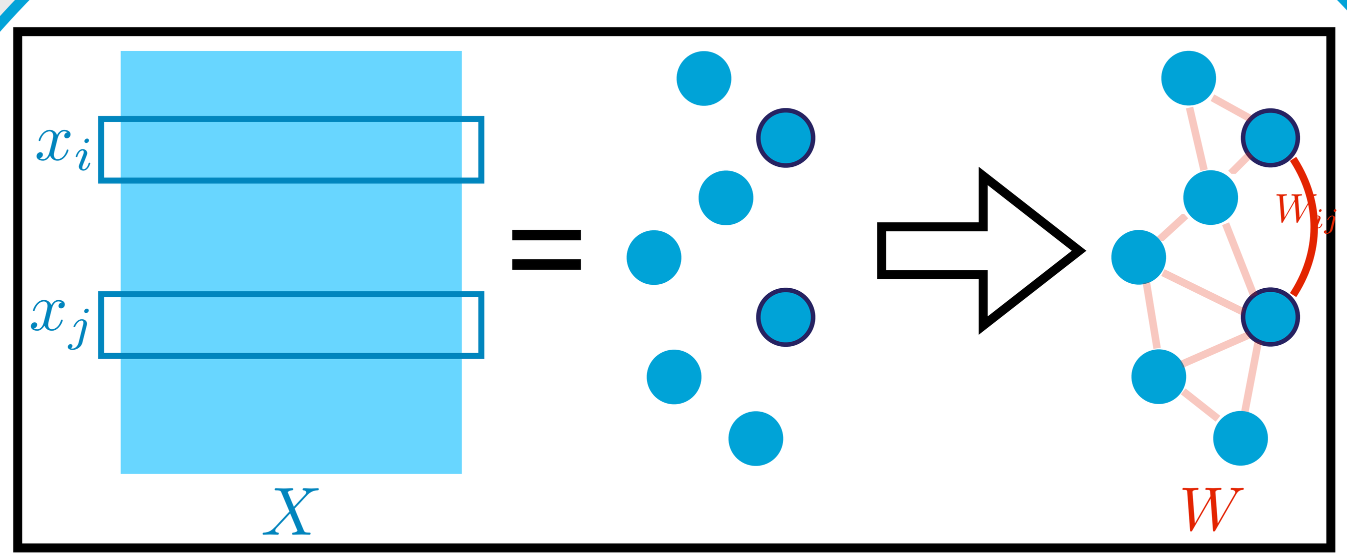
- Inputs: signals X, sparsity k
- Compute a 3k ANN graphs $\mathcal{O}(n\log(n)k)$
- Estimate θ
- Learn weights for allowed edges. Some of them are deleted $\mathcal{O}(nk)$
- Output: k NN learned graph

FIXED SUPPORT

Idea: Fix some edges to 0, learn the rest.

- Lower complexity Cost = $\mathcal{O}(|\mathcal{E}^{\text{allowed}}|)$
  Approximate Nearest Neighbours (ANN)
- Structure imposed by application
  e.g. geometric constraints

GRAPH LEARNING



Given a set of m vectors

Learn a graph (edge weights)

Assuming they are smooth:

Graph Laplacian $L = D - W$

$$\frac{1}{2}\sum_{i,j} W_{ij}\|x_i - x_j\|^2 = \mathrm{tr}\left(X^\top L X\right) \text{ is small}$$

Prefer sparse graphs!

- Better interpretability
- Smaller computation

PROBLEMS

How to scale it?
Number of edges is quadratic with the number of nodes.

How to tune the parameters?
Avoid grid search over α and β.

REGULARIZATION PARAMETERS

A) Change of parameters $\delta = \sqrt{\frac{\alpha}{\beta}}$ $\theta = \sqrt{\frac{1}{\alpha\beta}}$

Proposition 1

$$W^*(Z, \alpha, \beta) = \sqrt{\frac{\alpha}{\beta}} W^*\left(\frac{1}{\sqrt{\alpha\beta}} Z, 1, 1\right) = \delta W^*(\theta Z, 1, 1).$$

Parameter δ only changes the scale

B) Set θ from desired sparsity level

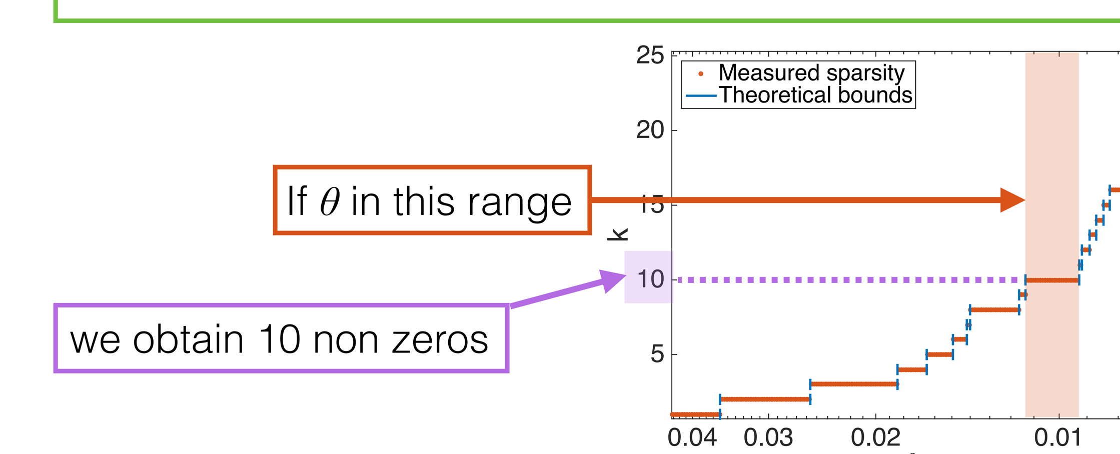
Idea: Simplify! Take **1** node:

$$\min_{w \geq 0} \theta w^\top z - \log(w^\top \mathbf{1}) + \frac{1}{2}\|w\|_2^2.$$

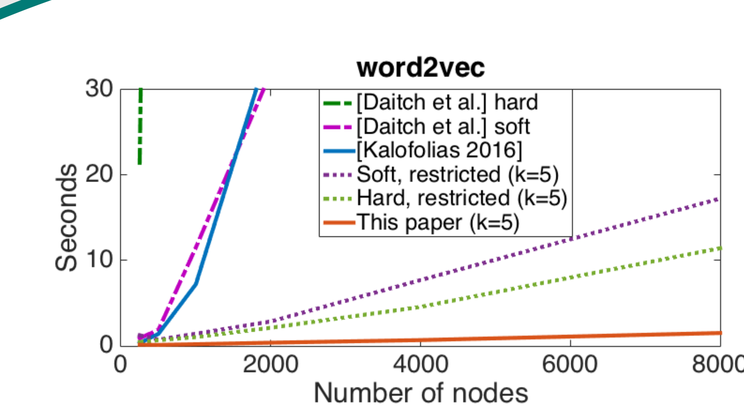
Theorem 2

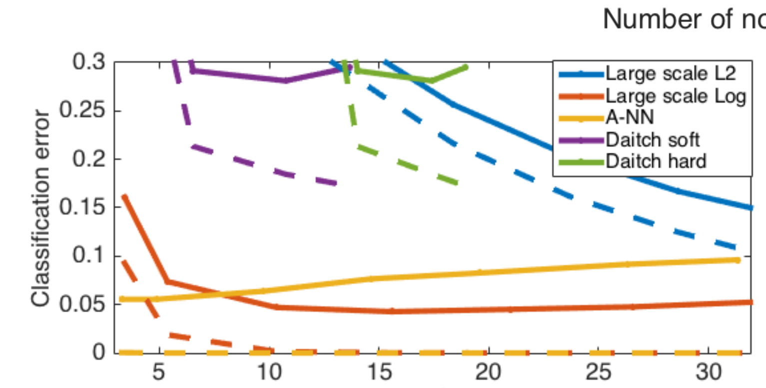
By setting θ in the $\left(\frac{1}{\sqrt{kz_{k+1}^2 - b_k z_{k+1}}}, \frac{1}{\sqrt{kz_k^2 - b_k z_k}}\right]$ range,

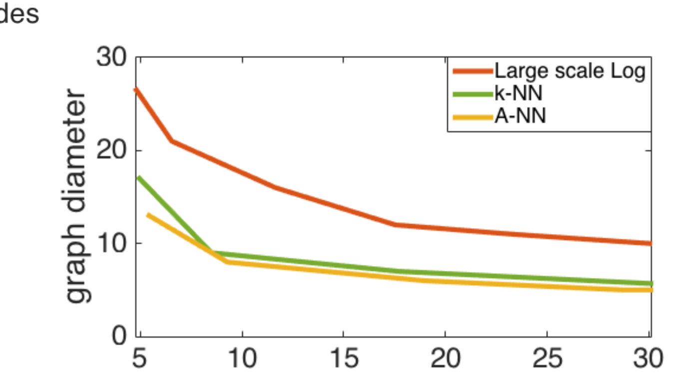
w^* has exactly k non-zero elements.



Setting θ: average of the **1** node problems







- **Scalability:** cost $\mathcal{O}(n\log(n)k)$ instead of $\mathcal{O}(n^2)$
  Faster than standard graph learning and kNN
- **Quality:** better than ANN and kNN

paper:

References

[1] V. Kalofolias. *How to learn a graph from smooth signals.* AISTATS 2016.

[2] V. Kalofolias. *From data to structures: graph learning under smoothness assumptions and applications in data science.* PhD thesis, EPFL, 2016.

Code available: GSPbox
https://epfl-lts2.github.io/gspbox-html/doc/demos/

ETH zürich — ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE