IMPROVED LARGE-SCALE GRAPH LEARNING THROUGH RIDGE SPECTRAL SPARSIFICATION

DANIELE CALANDRIELLO1,2, IOANNIS KOUTIS3, ALESSANDRO LAZARIC4, MICHAL VALKO1
1SEQUEL TEAM, INRIA LILLE - NORD EUROPE. 2 LCSL-IIIT/MIT. 3 NEW JERSEY INSTITUTE OF TECHNOLOGY. 4 FACEBOOK AI RESEARCH PARIS.

**RIDGE SPECTRAL SPARSIFIERS**

**Definition 1.** A $(\gamma, \epsilon)$-spectral sparsifier of $G$ is a re-weighted subgraph $H \subseteq G$ whose Laplacian $L_H$ satisfies

$$(1 - \epsilon)L_G - \gamma I \preceq L_H \preceq (1 + \epsilon)L_G + \gamma I,$$

Spectrum is preserved with mixed multiplicative/additive error $1 - \epsilon |\lambda_i(L_G) - \gamma| \leq |\lambda_i(L_H) - (1 + \epsilon)| \lambda_i(L_G) + \epsilon |\lambda_i(L_G)|$.

Preserves all directions larger than $\gamma$.

An $(\epsilon, \gamma)$-spectral sparsifier is a traditional $\epsilon$-sparsifier.

**Proposition 1.** [Informal]. Starting from the empty graph, construct $H$ by adding each edge in $G$ to $H$ independently with probability $p = (3/2 + \epsilon)\lambda + 2\epsilon |\lambda(L_G)|$. Then, with $p = 1 - \epsilon$, $H$ is an $(\epsilon, \gamma)$-sparsifier with $\lambda(L_H) \leq \kappa L_G$ edges.

Computing $\lambda(\gamma)$ requires $O(n)$ time/space and multiple passes over the graph. Can we do better?

**DISTRIBUTED SEQUENTIAL SPARSIFICATION**

**Algorithm 1** DiSpRe algorithm.

1. Input: $G = (V, E)$. Output: $(\gamma, \epsilon)$-sparsifier $H_G$.
2. Partition $G$ into $k$ sub-graphs $H_1, \ldots, H_k$.
3. Initialize set $S_i = \emptyset$ for all $i \in [k].$
4. For $h = 1, \ldots, k$ do
5. Pick two sparsifiers $H,h_r$.
6. Add $h_r$ to $S_i$ for all $i \in [k].$
7. End.
9. The last sparsifier in $S_k$.

**Theorem 1.** If $c(\cdot)$ is an $(\cdot, \gamma)$-sparsifier of $G$, then with probability $1 - \delta$:

- Each sub-graph $H_i$ has an $(\cdot, \gamma)$-sparsifier $G_i$.
- If $|\lambda(\gamma)|$ is the sparsifying parameter $\lambda$.

$$\text{SSL with DiSpRe}$$

**Setting.** The labels are bounded $|y_i(x)| \leq c$ and $x$ is the set of centered functions such that $f(x) = (x) \leq 2c$.

**Theorem 2.** If the labels $y_i$ are centered then, w.p. $1 - \delta$, $\hat{y}_i$ computed on a $(\cdot, \gamma)$-sparsifier $H$ satisfies

$$b(\hat{y}_i, y_i) \leq b(\hat{y}_i, y_i) + \beta \leq \beta(\epsilon, \gamma) = \sum_{i=1}^{n} \left( \frac{2}{\epsilon^2} \left( \frac{1 + \epsilon}{1 - \epsilon} \right) \left( 1 + \frac{\epsilon}{1 - \epsilon} \right) \right)^\gamma.$$

**LapSpMo with DiSpRe**

**Theorem 3.** Let $\hat{y}$ be the LatStap solution computed using $L_G$ and $\hat{y}$ the solution computed using $L_H$. Then,

$$b(\hat{y}, y) \leq b(\hat{y}, y) + \beta \leq \beta(\epsilon, \gamma) = \sum_{i=1}^{n} \left( \frac{2}{\epsilon^2} \left( \frac{1 + \epsilon}{1 - \epsilon} \right) \left( 1 + \frac{\epsilon}{1 - \epsilon} \right) \right)^\gamma.$$

**EXPERIMENTS**

**Dataset:** Amazon co-purchase graph from https://snap.stanford.edu/data/com-Amazon.html (Yang and Leskovec, 2012).

$\mathbb{L}_1 = 0.7$, 2, 163 nodes, natural, artificially sparse (true graph known only to Amazon).

**Target:** For $\mathbb{L}_0$ sparsification and smallest edge weight $L_0$, for SSL sign(y).

$\mathbb{L}_0 = 0.7$, 2, 163 nodes, natural, artificially sparse (true graph known only to Amazon).

**CONCLUSION**

- New sparsification techniques that achieve better approximation guarantees.

**REFERENCES**


