Supplementary Material: Learning from Graph Structure under Differential Privacy

Rafael Pinot[1,2], Florian Yger[1], Cédric Gouy-Pailler[2], and Jamal Atif[1]

[1] Université Paris-Dauphine, PSL Research University, LAMSADE, Paris, France
[2] Institut LIST, CEA, Université Paris-Saclay, LADIS, Palaiseau, France
rafael.pinot@dauphine.fr

1 Paper Proofs

Lemma 1. *Let $G = (\mathcal{G}, w)$, $w \in \mathcal{W}_E$, $\gamma \in (0,1)$, and $\epsilon > 0$. Let $\{\mathcal{R}_1, ..., \mathcal{R}_{|V|-1}\}$ denotes the ranges used in the successive calls of the Exponential mechanism in* $\text{PAMST}(\mathcal{G}, u_{\mathcal{G}}, w, \epsilon)$*, $\mathcal{T}$ the spanning tree topology returned by this algorithm and $w(\mathcal{T})$ its associated total weight. Then, if one denotes $w^* = \sum_{i=1}^{|V|-1} \min_{r \in \mathcal{R}_i} w(r)$, one gets with probability at least $1-\gamma$*

$$w\left(\mathcal{T}\right) - w^* \leq \frac{4(|V|-1)}{\epsilon}\left((|V|-1)\ln\left(\frac{|V|-1}{\gamma}\right) + \sum_{i=1}^{|V|-1}\ln|\mathcal{R}_i|\right).$$

Proof. For clarity of the proof, we denote $\epsilon' := \frac{\epsilon}{|V|-1}$, $OPT_i := OPT_{u_{\mathcal{G}}}\left(w_{|\mathcal{R}_i}\right)$, $u_i := u_{\mathcal{G}}\left(w, \text{Ex}\left(\mathcal{G}, w, u_{\mathcal{G}}, \mathcal{R}_i, \epsilon'\right)\right)$, and finally $r_i \in \mathcal{R}_i$ the edge selected at the i^{th} step of the algorithm. According to Theorem 1 for all $i \in [|V|-1]$, and $t \in \mathbb{R}$

$$\mathbb{P}\left[u_i \leq OPT_i - \frac{2\mathcal{S}_{u_{\mathcal{G}}}}{\epsilon'}\left(t + \ln|\mathcal{R}_i|\right)\right] \leq \exp(-t). \tag{1}$$

Then, by union bound, one gets

$$\mathbb{P}\left[\exists i \ s.t \ \left\{u_i \leq OPT_i - \frac{2\mathcal{S}_{u_{\mathcal{G}}}}{\epsilon'}\left(t + \ln|\mathcal{R}_i|\right)\right\}\right] \leq (|V|-1)\exp(-t). \tag{2}$$

This probability is also straightforwardly an upper bound of

$$\mathbb{P}\left[\sum_{i=1}^{|V|-1} u_i \leq \sum_{i=1}^{|V|-1} OPT_i - \frac{2\mathcal{S}_{u_{\mathcal{G}}}}{\epsilon'}\left(t + \ln|\mathcal{R}_i|\right)\right]. \tag{3}$$

Given the form of the chosen utility function, one gets for all $i \in [|V|-1]$ $OPT_i = 0$, and since in our setting $\mathcal{S}_{u_{\mathcal{G}}} = 2$

$$= \mathbb{P}\left[\sum_{i=1}^{|V|-1} u_i \leq -\frac{4}{\epsilon'}\left((|V|-1)\,t + \sum_{i=1}^{|V|-1}\ln|\mathcal{R}_i|\right)\right]. \tag{4}$$

Since $u_i = \min_{r \in \mathcal{R}_i} w(r) - w(r_i)$, one has $\sum_{i=1}^{|V|-1} u_i = w^* - \sum_{i=1}^{|V|-1} w(r_i)$, therefore

$$= \mathbb{P}\left[w^* - \sum_{i=1}^{|V|-1} w(r_i) \leq -\frac{4}{\epsilon'} \left((|V|-1)\, t + \sum_{i=1}^{|V|-1} \ln |\mathcal{R}_i| \right) \right]. \tag{5}$$

Setting $(|V|-1)\exp(-t) = \gamma$, replacing ϵ', since $w(\mathcal{T}) = \sum_{i=1}^{|V|-1} w(r_i)$, and using Equations (2) and (5) one gets

$$w(\mathcal{T}) - w^* \geq \frac{4(|V|-1)}{\epsilon} \left((|V|-1) \ln \left(\frac{|V|-1}{\gamma} \right) + \sum_{i=1}^{|V|-1} \ln |\mathcal{R}_i| \right) \tag{6}$$

with probability lower than γ. One finally obtains

$$w(\mathcal{T}) - w^* < \frac{4(|V|-1)}{\epsilon} \left((|V|-1) \ln \left(\frac{|V|-1}{\gamma} \right) + \sum_{i=1}^{|V|-1} \ln |\mathcal{R}_i| \right) \tag{7}$$

with probability at least $1-\gamma$.

For proving Lemma 2 from the main paper one need the following intermediary result

Proposition 1. *Let $G = (V, E, w)$ the studied graph, $\mathcal{T}^*$ the minimum spanning tree of G, $(S_i)_{i \in [|V|-1]}$ the sets of nodes at every step of* PAMST*, $(\mathcal{R}_i)_{i \in [|V|-1]}$ the corresponding sets of edges connecting S_i and $V \backslash S_i$. Finally, for all $i \in [|V|-1]$, the sets of edges incident to one node in S_i are denoted $I_i = \bigcup_{j \in [i]} \mathcal{R}_j$. Then, for all i in $[|V|-1]$, $\mathcal{R}_i$ contains at least one edge from $\mathcal{T}^*$, and I_i contains at least i edges from $\mathcal{T}^*$.*

Proof. For readability of the proofs, we denote by $|.|_{\mathcal{T}^*}$ the number of edges from $\mathcal{T}^*$ that a set of edges contains. Let us first suppose that $\exists i \in [|V|-1]$ such that $\mathcal{R}_i$ contains no edge from $\mathcal{T}^*$, then we managed to produce a cut of the graph $(S_i, V \backslash S_i)$ such that no edge of $\mathcal{T}^*$ crosses it. By definition of a spanning tree, this is impossible. Therefore for all i in $[|V|-1]$, $|\mathcal{R}_i|_{\mathcal{T}^*} \geq 1$.
Second, let us show by recurrence that for all $n \in [|V|-1]$, $P_n :=$ "$|I_n|_{\mathcal{T}^*} \geq n$" is true.
P_1 is true according to the first point of this proof and since $\mathcal{R}_1 = I_1$. Given $k \in [|V|-2]$, let us suppose that P_k is true. If $I_{k+1} \backslash I_k = \mathcal{R}_{k+1} \backslash I_k \neq \emptyset$ then P_{k+1} is automatically satisfied. Otherwise let us suppose that $|I_k|_{\mathcal{T}^*} \leq k$, then $|I_{k+1}|_{\mathcal{T}^*} \leq k$. According to the first part of the proposition, $\exists q \geq 1$ such that $|\mathcal{R}_{k+1}|_{\mathcal{T}^*} = q$, therefore $|I_{k+1} \backslash \mathcal{R}_{k+1}|_{\mathcal{T}^*} \leq k - q$. Then the graph edge-induced by $I_{k+1} \backslash \mathcal{R}_{k+1}$ contains at least $q+1$ subtrees of $\mathcal{T}^*$. Since $|\mathcal{R}_{k+1}|_{\mathcal{T}^*} = q$, at least one subtree, denoted S^*, does not have an incident edge in $\mathcal{T}^* \cap \mathcal{R}_{k+1}$. We thus

produced a cut $(S^*, V \backslash S^*)$ such that no edge of $\mathcal{T}^*$ crosses it. This contradiction implies that our assumption is false, then $|I_k|_{\mathcal{T}^*} = |I_{k+1}|_{\mathcal{T}^*} \geq k+1$. We just proved that P_k implies P_{k+1}, thus, Proposition 1 is proved by recurrence.

Lemma 2. *Denoting $\mathcal{T}^*$ the MST of G, and with the above notations for w^*, one has $w^* \leq w(\mathcal{T}^*)$.*

Proof. $\mathcal{T}^*$ can always be seen as a set of weight-ordered edges $(e_i)_{i\in[|V|-1]}$ for which $w(e_1) < ... < w\left(e_{|V|-1}\right)$. We only consider the strict inequalities for the sake of readability, however a generalization of this proof to classical inequalities is straightforward since treating the equality cases is immediate.

Let $k > 1$, for all $j > k$, thanks to Proposition 1, one has $|I_j|_{\mathcal{T}^*} \geq j$. Therefore, at least $j - k + 1$ edges in $I_j \cap \mathcal{T}^*$ have a weight lower or equal to $w(e_{|V|-k})$, thus at least $j - k$ steps of the algorithm have an optimal solution with a weight lower or equal to $w(e_{|V|-k})$. Therefore, for all $k \geq 2$, there is at most $k-1$ sets $\{\mathcal{R}_{i_1}, ..., \mathcal{R}_{i_{k-1}}\}$ such that $\min\limits_{r\in\mathcal{R}_{i_\bullet}} w(r) > w(e_{|V|-k})$.

Then, in the worst case scenario, for all $k \in \{2, ..., |V|-1\}$, there exist exactly $k-1$ steps such that $\min\limits_{r\in\mathcal{R}_{i_\bullet}} w(r) > w(e_{|V|-k})$. The only way to have such sets is if one has $\{i_1, ..., i_{|V|-1}\}$ such that

$$\min_{r\in\mathcal{R}_{i_1}} w(r) > w(e_{|V|-2}) \geq ... > w(e_1) \geq \min_{r\in\mathcal{R}_{i_{|V|-1}}} w(r). \tag{8}$$

Moreover, since for all $i \in [V-1]$, $|\mathcal{R}_i|_{\mathcal{T}^*} \geq 1$, one gets

$$w(e_{|V|-1}) \geq \min_{r\in\mathcal{R}_i} w(r), \text{ for all } i \in [V-1]. \tag{9}$$

Finally, one has

$$w(e_{|V|-1}) \geq \min_{r\in\mathcal{R}_{i_1}} w(r) > w(e_{|V|-2}) \geq \cdots > w(e_1) \geq \min_{r\in\mathcal{R}_{i_{|V|-1}}} w(r). \tag{10}$$

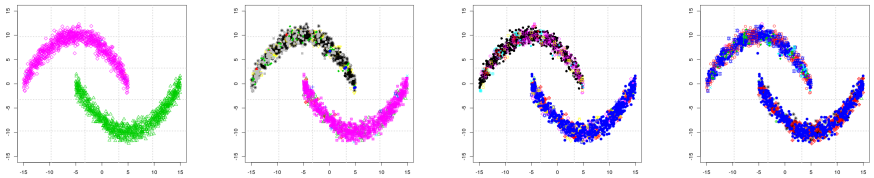
Therefore

$$w^* = \sum_{i\in[|V|-1]} \min_{r\in\mathcal{R}_i} w(r) \leq \sum_{i\in[|V|-1]} w(e_{|V|-i}) = w(\mathcal{T}^*). \tag{11}$$

2 Arbitrary shaped clusters

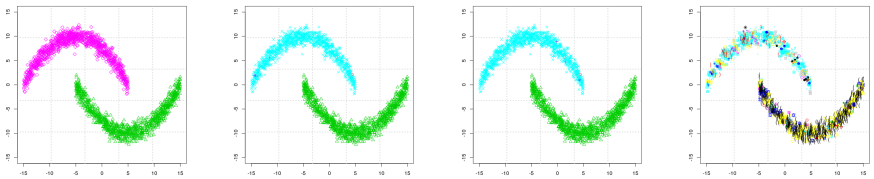
We conducted some experiments on an arbitrarily shaped data set (Two inested moon shapes), for several levels of privacy to demonstrate the accuracy of PAMST. For this purpose, the data set have been simulated based on 1000 uniform point sampling over arbitrary shapes, perturbed by some random Gaussian noise. The data set is composed of well separated clusters, and therefore, constitute a suitable benchmark for the evaluation of any clustering method. Although this data set is not initially a graph, the euclidean distance matrix

between all node pairs can be interpreted as a complete weighted graph and thus MST-based clustering can be performed. We compare private clustering based on PAMST, Private Kruskal, and Laplace methods. Since the gap between the accuracy PAMST/Kruskal and the Laplace increases accordingly with the density of the graph, we choose to sparsify the distance matrix. Also for the sake of readability of the figures, we omit to represent the edges of this data set.



(a) MSDR (b) Laplace, $\epsilon = 1.0$(c) Laplace, $\epsilon = 0.5$(d) Laplace, $\epsilon = 0.1$

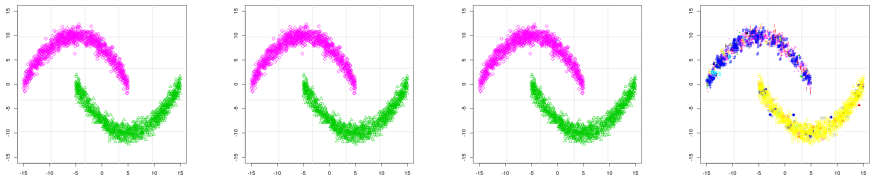
Fig. 1: Moons experiment for 1000 nodes. This experiment compares the accuracy of the non private clustering method (MSDR), with the private (Laplace non iterative), for several privacy levels



(a) MSDR (b) PAMST, $\epsilon = 1.0$ (c) PAMST, $\epsilon = 0.5$ (d) PAMST, $\epsilon = 0.1$

Fig. 2: Moons experiment for 1000 nodes. This experiment compares the accuracy of the non private clustering method (MSDR), with the private (PAMST), for several privacy levels

As expected according to the analysis conducted in the main part of the paper, when trying to ensure privacy levels lower to 1, The Laplace method is not sufficiently precise to prevent the clustering algorithm from failing. On the other hand PAMST, and Kruskal by producing a more accurate approximated minimum spanning tree allows MSDR to present a convincing partition of the data set, for this same values of ϵ. As explained in the paper, since PAMST and private Kruskal obtain similar results, these experiments prove the utility of PAMST for applications such as node clustering in a graph under differential privacy.



(a) MSDR (b) Kruskal, $\epsilon = 1.0$ (c) Kruskal, $\epsilon = 0.5$ (d) Kruskal, $\epsilon = 0.1$

Fig. 3: Moons experiment for 1000 nodes. This experiment compares the accuracy of the non private clustering method (MSDR), with private Kruskal, for several privacy levels

3 Reusable implementation

We joined to our submission a R package implementing the methods. The proposed implementation focuses on re-usability of the code. In our opinion re-usability in machine learning is much stronger than reproducibility. The interested reader should refer to Gael Varoquaux's blog post[3] for a good insight on this question. privateMST package is self-content. To install it one needs the following command from bash:

```
R CMD INSTALL privateMST.{tar.gz,zip} // depending on your operating system
```

or within R:

```
install.packages(pkgs = "pathtopackage/privateMST.{tar.gz,zip}", repos = NULL)
```

[3] http://gael-varoquaux.info/programming/of-software-and-science-reproducible-science-what-why-and-how.html