All of the Fairness for Edge Prediction with Optimal Transport Supplementary Materials

1 BACKGROUND ON OPTIMAL TRANSPORT

Finding correspondences between two sets of points is a longstanding problem in machine learning and data mining applications that occurs whenever one wants to align these latter. One way of achieving this is to learn a mapping that assigns the points across the two sets in an optimal way, i.e., by matching them with respect to a proximity given by a certain metric. Optimal transport (OT) problem, first introduced in [Monge, 1781], provides an efficient tool for this problem by finding an optimal one-to-one transport map between the two sets calculated by considering the geometrical proximity of points in them.

In practice, OT can be formulated as a problem of aligning two empirical distributions $\hat{\mu}_0$ and $\hat{\mu}_1$ supported on two point sets $X_0 = \{x_0^{(i)} \in \mathbb{R}^d\}_{i=1}^{N_0}$ and $X_1 = \{x_1^{(j)} \in \mathbb{R}^d\}_{i=1}^{N_1}$. As the original Monge problem mentioned above may not have a solution in some cases, [Kantorovich, 1942] proposed to relax it by searching instead for a coupling γ defined as a joint probability distribution over $X_0 \times X_1$ with marginals $\hat{\mu}_0$ and $\hat{\mu}_1$. The intuition behind such a replacement is to allow one-to-many soft assignments between the points of the two sets rather than stricter one-to-one assignments only. The goal of the obtained Monge-Kantorovich problem is then to minimize the cost of transport w.r.t. some metric $l : X_0 \times X_1 \to \mathbb{R}^+$:

$$\min_{\gamma \in \Pi(\hat{\mu}_0, \hat{\mu}_1)} \langle M, \gamma \rangle_F \tag{1}$$

where $\langle \cdot, \cdot \rangle_F$ is the Frobenius dot product, M is a dissimilarity matrix, i.e., $M_{ij} = l(x_0^{(i)}, x_1^{(j)})$, defining the cost of associating $x_0^{(i)}$ with $x_1^{(j)}$ and $\Pi(\hat{\mu}_0, \hat{\mu}_1) = \{\gamma \in \mathbb{R}_+^{N_0 \times N_1} | \gamma \mathbf{1} = \hat{\mu}_0, \gamma^T \mathbf{1} = \hat{\mu}_1\}$ is a set of doubly stochastic matrices. This problem admits a unique solution γ^* and defines a metric on the space of probability measures (called the Wasserstein distance) as follows:

$$W_M(\hat{\mu}_0, \hat{\mu}_1) = \min_{\gamma \in \Pi(\hat{\mu}_0, \hat{\mu}_1)} \langle M, \gamma \rangle_F.$$

Wasserstein distance has been efficiently used in many of machine learning fields including domain adaptation [Courty et al., 2017, Redko et al., 2019], clustering [Laclau et al., 2017] and multi-class classification [Frogner et al., 2015] to name a few. One application that is of a particular interest is that of fairness classification where the Wasserstein distance and the notion of Wasserstein barycenter given by a weighted mean of Wasserstein distances have been recently exploited in [Gordaliza et al., 2019, Jiang et al., 2019, Zehlike et al., 2020]. In this paper, we aim at adapting a similar approach for fair edge prediction in graphs.

2 PROOFS

Proof of Theorem 1

Proof. We start by upper-bounding $\mathbb{P}(h(V, V') = 1 | S = 1)$ and then proceed by giving a lower bound for $\mathbb{P}(h(V, V') = 1 | S = 0)$.

$$\mathbb{P}(h(V, V') = 1 | S = 1) = \frac{\mathbb{P}(S = 1 | h(V, V') = 1) \mathbb{P}(h(V, V') = 1)}{\mathbb{P}(S = 1)} \tag{2}$$

$$= \frac{\mathbb{P}(S = 1, S' = 1 \vee S = 1, S' = 0 | h(V, V') = 1) \mathbb{P}(h(V, V') = 1)}{\mathbb{P}(S = 1)} \tag{3}$$

$$\leq \frac{\mathbb{P}(S = 1, S' = 1 \vee S = 0, S' = 0 | h(V, V') = 1) \mathbb{P}(h(V, V') = 1)}{\mathbb{P}(S = 1)} \tag{4}$$

$$= \frac{\mathbb{P}(S = S' | h(V, V') = 1) \mathbb{P}(h(V, V') = 1)}{\mathbb{P}(S = S')} \tag{5}$$

$$= \mathbb{P}(h(V, V') = 1 | S = S').$$

where (2) is obtained from Bayes' theorem; (3) follows from the fact that $P(A) = P(A|B) + P(A|\bar{B})$; (4) and (5) follow from assumptions A1 and A2, respectively.

Similarly, we get for $\mathbb{P}(h(V, V') = 1 | S = 0)$:

$$\begin{aligned}
\mathbb{P}(h(V, V') = 1 | S = 0) &= \frac{\mathbb{P}(S = 0 | h(V, V') = 1)\mathbb{P}(h(V, V') = 1)}{\mathbb{P}(S = 0)} \\
&= \frac{\mathbb{P}(S = 0, S' = 1 \vee S = 0, S' = 0 | h(V, V') = 1)\mathbb{P}(h(V, V') = 1)}{\mathbb{P}(S = 0)} \\
&\geq \frac{\mathbb{P}(S = 0, S' = 1 \vee S = 1, S' = 0 | h(V, V') = 1)\mathbb{P}(h(V, V') = 1)}{\mathbb{P}(S = 0)} \\
&= \frac{\mathbb{P}(S \neq S' | h(V, V') = 1)\mathbb{P}(h(V, V') = 1)}{\mathbb{P}(S \neq S')} \\
&= \mathbb{P}(h(V, V') = 1 | S \neq S').
\end{aligned}$$

Combined together, we obtain that:

$$\frac{\mathbb{P}(h(V, V') = 1 | S \neq S')}{\mathbb{P}(h(V, V') = 1 | S = S')} \leq \frac{\mathbb{P}(h(V, V') = 1 | S = 0)}{\mathbb{P}(h(V, V') = 1 | S = 1)} \leq \tau.$$

□

Proof of Corollary 1

Proof. We use Theorem 1 to obtain

$$\begin{aligned}
&\text{DI}(h, \mathbb{V}, S \oplus S') \leq \text{DI}(h, \mathbb{V}, S) \leq \tau \\
&\implies \frac{1}{\tau}\mathbb{P}(h(V, V') = 1 | S \neq S') \leq 1 - \mathbb{P}(h(V, V') = 0 | S = S') \\
&\implies (1 + (\frac{1}{\tau} - 1))\mathbb{P}(h(V, V') = 1 | S \neq S') + \mathbb{P}(h(V, V') = 0 | S = S') \leq 1 \\
&\implies \mathbb{P}(h(V, V') = 1 | S \neq S') + \mathbb{P}(h(V, V') = 0 | S = S') \\
&\qquad\qquad \leq (\frac{1}{\tau} - 1)\mathbb{P}(h(V, V') = 1 | S \neq S') \\
&\implies \text{BER}(h, \mathbb{V}, S \oplus S') \leq \frac{1}{2} - \frac{\mathbb{P}_1(h)}{2}\left(\frac{1}{\tau} - 1\right).
\end{aligned}$$

For the second part of the statement, we use [Gordaliza et al., 2019, Theorem 2.2] to obtain

$$\min_{h \in \mathcal{H}} \text{BER}(h, \mathbb{V}, S \oplus S') = \frac{1}{2}(1 - \mathrm{d}_{\text{TV}}(\gamma_0, \gamma_1)).$$

We further use the equality between the Wasserstein distance with Hamming distance used as a cost function and the total variation metric to obtain the final result. □

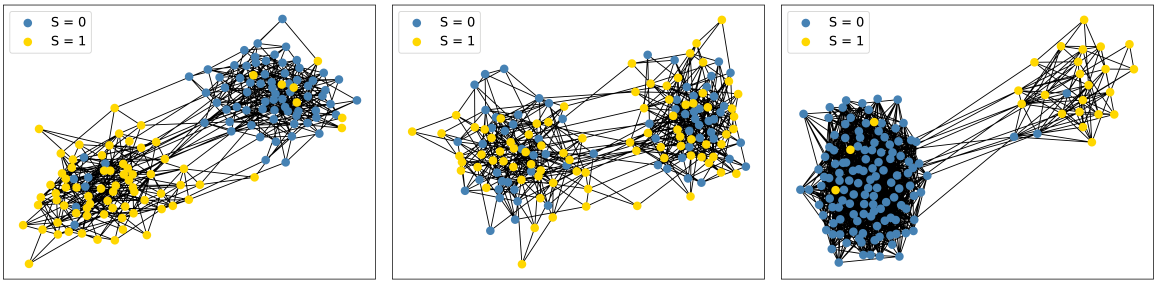
3 ADDITIONAL EXPERIMENTS AND RESULTS

3.1 Experiments on Synthetic Networks

We evaluate our approach on several synthetic graphs of controlled complexity. This allows us to cover different possible scenarios for the presence of bias in graphs in order to highlight the different features of our algorithm.

Graph generation We generate five synthetic graphs (G1–G5) composed of 150 nodes each on the basis of the stochastic block model with different block parameters. This latter is done using the networkx library [1]. For the sake of reproducibility, we provide the parameters used to generate the graphs in Table 1. We further assume that each node is associated with one value of sensitive attribute $S \in \{0, \cdots, K\}$. For the first four graphs we assume a binary sensitive attribute, while for G5 we assume S to be multiclass (K=3).

In terms of the structure, G1 corresponds to a graph with two communities and a strong dependency between the community structure and the sensitive attribute S; G2 has the same community structure as G1, but with S being independent of the structure; G3 corresponds to a graph with two imbalanced communities dependent on S and a stronger intra-connection in the smaller community; G4 is a graph with three communities, two of them being dependent on S, and the third one being independent. Finally, G5 is similar to G4 and has three communities, each of them being dependent on one of the class of the protected attribute. In the following, each graph is generated 50 times and the average metrics are reported. Figure 1 shows the different graphs.



(a) G1 (b) G2 (c) G3

(d) G4 (e) G5

Figure 1: Visualisation of the five synthetic graphs G1–G5. Node color indicates the value of the sensitive attribute associated with each node .

Obtained results We now focus on the impact of the repairing and the Laplacian regularization on the graph topology and on the node embeddings generated with N2VEC. Table 2 reports the results for the EMD repairing on all graphs, which corresponds to the case without the individual fairness term. The assortativity coefficient is calculated conditionally on the sensitive attribute, and we recall that a value close to 1 (resp. -1) implies that the graph contains more edges between the nodes with the same (resp. different) value of protected attribute, while a value close to 0 implies that the distribution of edges is balanced between nodes belonging to the same and different groups. For the representation bias (RB), it corresponds to the AUC obtained by a classifier where the input is the node embedding and the output is the protected attribute. In our context, it should ideally be close to 0.5, i.e., the performance of random guessing. We also visualise the obtained embeddings in Figure 2.

From Table 2, one can see that the coefficient of assortativity and RB are initially high for graphs G1, G3 and G4, implying that these graphs are strongly biased. In all cases, we observe that the assortativity values are much lower on the repaired graphs than on the original ones, indicating that S is no longer correlated with their latent community structure. As for the RB, we also observe an important drop of the values after the repairing, indicating that the sensitive attribute can no longer

[1] https://networkx.github.io/

Table 1: Parameters used for graph generation. *Cluster* implies that the value of S is almost equal to the community identifier, while *random* implies that S is randomly generated and is independent from the community structure of the graph.

Graphs	S	$\lvert S\rvert$	Size	Probs
G1	cluster	2	$\begin{pmatrix}75 & 75\end{pmatrix}$	$\begin{pmatrix}.10 & .005\\ .005 & .10\end{pmatrix}$
G2	random	2	$\begin{pmatrix}75 & 75\end{pmatrix}$	$\begin{pmatrix}.10 & .005\\ .005 & .10\end{pmatrix}$
G3	cluster	2	$\begin{pmatrix}125 & 25\end{pmatrix}$	$\begin{pmatrix}.15 & .005\\ .005 & .35\end{pmatrix}$
G4	random & cluster	2	$\begin{pmatrix}50 & 50 & 50\end{pmatrix}$	$\begin{pmatrix}.20 & .002 & .003\\ .002 & .15 & .003\\ .003 & .003 & .10\end{pmatrix}$
G5	cluster	3	$\begin{pmatrix}50 & 50 & 50\end{pmatrix}$	$\begin{pmatrix}.20 & .002 & .003\\ .002 & .15 & .003\\ .003 & .003 & .10\end{pmatrix}$

Table 2: Coefficients of assortativity conditionally on the protected attribute (Ass.) and representation bias (RB) for the original (O) and the repaired (R) graphs when using the EMD repairing (i.e. regularization is set to 0).

	G1		G2		G3		G4		G5	
	O	R	O	R	O	R	O	R	O	R
Ass.	$.74 \pm .07$	$.01 \pm .08$	$-.01 \pm .04$	$-.02 \pm .04$	$.71 \pm .11$	$-.03 \pm .02$	$.60 \pm .07$	$.06 \pm .11$	$.74 \pm .07$	$.32 \pm .06$
RB	$.92 \pm .04$	$.48 \pm .06$	$.52 \pm .06$	$.44 \pm .06$	$.96 \pm .04$	$.43 \pm .08$	$.85 \pm .04$	$.55 \pm .11$	$.95 \pm .02$	$.91 \pm .03$

be inferred from the embeddings learnt on the graph. For G2, however, both quantities of interest remain the same before and after the repair. This is explained by the fact that this graph is not biased and thus need no repair. Finally, in case of G5, the assortativity coefficient decreases after the repair, while RB remains high. We explain this by the fact that the used embedding technique manages to pick up the signal in the case of multi-class repair more easily and reintroduces the bias back to the task at hand. All these observations are confirmed by the plots of the repaired embeddings given in Figure 2.

Role of the Laplacian regularization parameter As our algorithm depends on the regularization parameter λ controlling the individual fairness of the repair, we proceed to its study on the synthetic graphs below. To this end, in Figure 3 we illustrates the effect of this parameter on the performance of our algorithm for the graphs considered above. Two major observations are in order here. First, for graphs G1 and G3 the strength of the Laplacian regularization leads to the increasing assortativity coefficient and RB scores below the original values, while for unbiased graph G2 it has no particular effect on them. Second, the Laplacian regularization strength is less correlated with the quantities of interest in the imbalanced (G3) and multi-class settings G5 which can be explained by the inherent drawbacks of optimal transport in this case. Dealing with these drawbacks presents an important open avenue for future research.

3.2 Details on real-world networks

Hereafter, we provide details for the real-world networks experiments reported in the main paper.

- For N2VEC-based approaches, we use the same set of default hyper-parameters: dimension of the embeddings is set to 64, the length of the walk to 15 and the window size is set to 10. For the link prediction, we train a logistic regression classifier on edge-wise feature vectors, where given a pair of users (u, v) and their respective embeddings $z(u)$, $z(v)$, the goal is to predict the existence of an edge between u and v. In what follows, edge-wise feature vectors correspond to the Hadamard product defined as element-wise multiplication between $z(u)$ and $z(v)$. We train the binary classifier by sampling non-existing edges as negative examples.

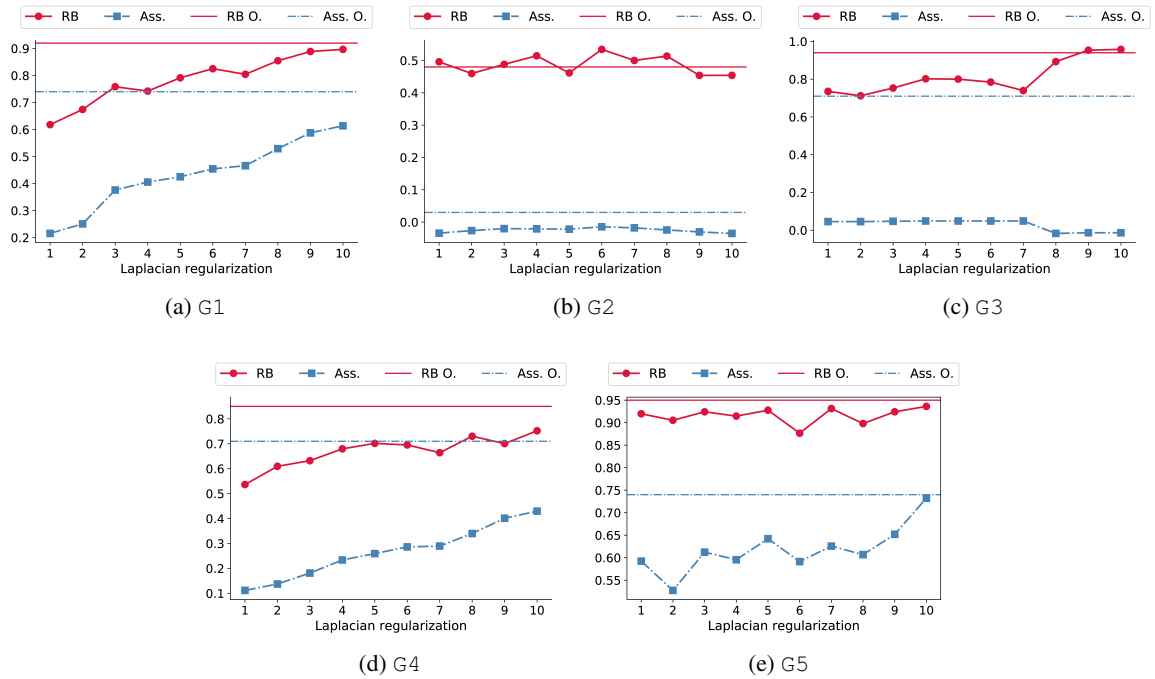
- For CNE-based approaches, we use the set of default parameters recommended by the authors of DEBAYES [Buyl and Bie, 2020]: dimension of the embeddings is set to 8 and the parameters σ_1 and σ_2 are set to 1 and 16, respectively. As for the link prediction, we follow their protocol by first computing the posterior $P(a_{ij} = 1|\text{train set})$ of the test links based on the embedding trained on the training network. Then the AUC score is computed by comparing the posterior probability of the test links and their true labels.

(a) G1 (b) G2

(c) G3 (d) G4

(e) G5

Figure 2: Visualisation with t-SNE of node embeddings computed with NODE2VEC on (left) the original graph and (right) on the repaired version.



(a) G1 (b) G2 (c) G3

(d) G4 (e) G5

Figure 3: Impact of the Laplacian regularisation on the RB and on the coefficient of assortativity (Ass.). The letter 'O' in the legend stands for original.

References

[Buyl and Bie, 2020] Buyl, M. and Bie, T. D. (2020). Debayes: a bayesian method for debiasing network embeddings. In *Proceedings ICML*, pages 2537–2546.

[Courty et al., 2017] Courty, N., Flamary, R., Habrard, A., and Rakotomamonjy, A. (2017). Joint distribution optimal transportation for domain adaptation. In *NeurIPS*, pages 3730–3739.

[Frogner et al., 2015] Frogner, C., Zhang, C., Mobahi, H., Araya-Polo, M., and Poggio, T. (2015). Learning with a wasserstein loss. In *NIPS*, page 2053–2061.

[Gordaliza et al., 2019] Gordaliza, P., del Barrio, E., Gamboa, F., and Loubes, J. (2019). Obtaining fairness using optimal transport theory. In *ICML*, pages 2357–2365.

[Jiang et al., 2019] Jiang, R., Pacchiano, A., Stepleton, T., Jiang, H., and Chiappa, S. (2019). Wasserstein fair classification. In *UAI*, page 315.

[Kantorovich, 1942] Kantorovich, L. (1942). On the translocation of masses. In *C.R. (Doklady) Acad. Sci. URSS(N.S.)*, volume 37(10), pages 199–201.

[Laclau et al., 2017] Laclau, C., Redko, I., Matei, B., Bennani, Y., and Brault, V. (2017). Co-clustering through optimal transport. In *ICML*, pages 1955–1964.

[Monge, 1781] Monge, G. (1781). Mémoire sur la théorie des déblais et des remblais. *Histoire de l'Académie Royale des Sciences*, pages 666–704.

[Redko et al., 2019] Redko, I., Courty, N., Flamary, R., and Tuia, D. (2019). Optimal transport for multi-source domain adaptation under target shift. In *AISTATS*, volume 89, pages 849–858.

[Zehlike et al., 2020] Zehlike, M., Hacker, P., and Wiedemann, E. (2020). Matching code and law: achieving algorithmic fairness with optimal transport. *Data Min. Knowl. Discov.*, 34(1):163–200.