

Appendix for “Transfer Learning on Edge Connecting Probability Estimation Under Graphon Model”

Anonymous Author(s)
Affiliation
Address
email

870 A Notation Table

Table A1: Notations used throughout the paper

Notation	Description
n_s, n_t	Number of nodes in the source and target graphs
$\mathbf{A}_s, \mathbf{A}_t$	Adjacency matrices of source and target graphs, $\mathbf{A} \in \{0, 1\}^{n \times n}$
$\mathbf{P}_s, \mathbf{P}_t$	True connection probability matrices for source and target graphs
$\hat{\mathbf{P}}_s^{\text{ini}}, \hat{\mathbf{P}}_t^{\text{ini}}$	Initial estimators via neighborhood smoothing
$\hat{\mathbf{P}}_t^{\text{trans}}, \hat{\mathbf{P}}_t^{\text{trans2}}$	Transferred estimator before and after smoothing
$\hat{\mathbf{P}}_t^{\text{res}}$	Smoothed residual graphon estimator
$\hat{\mathbf{P}}_t$	Final graphon estimator for the target domain
$\hat{\mathbf{P}}_{ij}$	Estimated connecting probability between node i and j
$\text{Ber}(p)$	Bernoulli distribution with parameter p
f_s, f_t	Latent graphon functions for source and target domains
$u_{s,i}, u_{t,i}$	Latent position of node i in $[0, 1]$
$\Pi(\mu, \nu)$	Set of couplings (transport plans) between μ and ν
$\hat{\pi} \in [0, 1]^{n_s \times n_t}$	The optimal transport plan estimated by Gromov-Wasserstein
$\tilde{\pi} \in [0, 1]^{n_s \times n_t}$	Normalized transport plan
δ	Threshold for triggering the debiasing step
ϵ	Entropic regularization parameter in Gromov-Wasserstein optimization
λ	Density shift parameter for source-target perturbation.
$\text{KL}(\pi \mid \mu \otimes \nu)$	Kullback-Leibler divergence: $\text{KL}(\pi \mid \mu \otimes \nu) = \sum_{i,j} \pi_{ij} \log \frac{\pi_{ij}}{\mu_i \nu_j}$
$\mathbf{P}_s \otimes \mathbf{P}_t$	Kronecker product of \mathbf{P}_s and \mathbf{P}_t
$\ \cdot\ _F$	Frobenius norm: $\ \mathbf{X}\ _F = \sqrt{\sum_{i,j} X_{ij}^2}$
$\ \cdot\ _2$	L_2 norm: $\ f - g\ _2 = \left(\int_0^1 \int_0^1 (f(u, v) - g(u, v))^2 du dv \right)^{1/2}$
$\ \cdot\ _\infty$	Infinity norm: $\ \mathbf{X}\ _\infty = \max_{i,j} X_{ij} $
$\ \cdot\ _{\text{op}}$	Operator norm: $\ \mathbf{X}\ _{\text{op}} = \sigma_{\max}(\mathbf{X})$

871 B Proof of Theorem 4.1

872 In this proof, we will use the notation $B(x; \tau)$ to denote a ball of radius τ around x , with respect to
873 some appropriate distance, which will be clear from the context. Before delving into the proof, let us
874 first recall the definition of strong convexity:

Definition B.1. A function f is said to be strongly convex in a neighborhood around x_* (namely $B(x_*; \tau)$), if f satisfies:

$$f(y) \geq f(x) + \langle y - x, \nabla f(x) \rangle + \frac{\mu}{2} \|y - x\|_2^2$$

875 Our proof relies on an application of the following lemma:

Lemma B.2. Suppose f is a convex function which is minimized at x_* and furthermore it is μ -strongly convex on $B(x_*; \tau) = \{x : \|x - x_*\|_2 \leq \tau\}$. Consider a perturbation g of f such that i) $\|f - g\|_\infty \leq \delta$ and ii) $g - f$ is Lipschitz with Lipschitz-constant κ . If $\delta \leq \mu\tau^2/4$, we have:

$$\|x_* - y_*\|_2 \leq \frac{2\kappa}{\mu} \equiv \frac{2\|f - g\|_{\text{Lip}}}{\mu}.$$

876 *Proof.* We divide the proof of Lemma B.2 into two smaller lemmas. The first lemma is as follows:

Lemma B.3. Consider two functions f and g , such that $f - g$ is κ -Lipschitz. Suppose x_* and y_* are minimizers of f and g respectively. If f is strongly convex on $B(x_*; \tau)$, and $y_* \in B(x_*; \tau)$, then

$$\|x_* - y_*\|_2 \leq \frac{2\kappa}{\mu}.$$

877 *Proof.* The proof essentially follows from Proposition 4.32 of [5]. Here, we sketch the proof for the
878 ease of the readers. As y_* is the minimizer of g , we have:

$$\begin{aligned} f(y_*) - f(x_*) &= (f - g)(y_*) - (f - g)(x_*) + \underbrace{g(y_*) - g(x_*)}_{\leq 0} \\ &\leq (f - g)(y_*) - (f - g)(x_*) \leq \kappa \|x_* - y_*\|_2. \end{aligned}$$

On the other, from the strong convexity of f on $B(x_*; \tau)$ and the assumption that $y_* \in B(x_*; \tau)$, we have:

$$f(y_*) \geq f(x_*) + \frac{\mu}{2} \|x_* - y_*\|_2^2,$$

where we use the fact that $\nabla f(x_*) = 0$. Combining these two equations, we conclude:

$$\frac{\mu}{2} \|x_* - y_*\|_2^2 \leq f(y_*) - f(x_*) \leq \kappa \|x_* - y_*\|_2 \implies \|x_* - y_*\|_2 \leq \frac{2\kappa}{\mu}.$$

879 This completes the proof. \square

880 One of the requirements of Lemma B.3 is that $y_* \in B(x_*; \tau)$. The following lemma gives a sufficient
881 condition for this condition to be satisfied:

882 **Lemma B.4.** Assume f is μ -strongly convex on $B(x_*; \tau)$. If $\|f - g\|_\infty \leq \delta$, $\delta \leq \tau\mu^2/4$, then the
883 minimizer of g also lies in $B(x_*; \tau)$.

Proof. Suppose $y_* \notin B(x_*; \tau)$. Then $\|x_* - y_*\| > \tau$. Therefore, $\exists t \in (0, 1)$ such that $x(t) = tx_* + (1 - t)y_*$ lies on the boundary, i.e., $\|x_* - x(t)\|_2 = \tau$. By the strong convexity of f on $B(x_*; \tau)$, we have:

$$f(x(t)) \geq f(x_*) + \frac{\mu\tau^2}{2}.$$

884 On the other hand:

$$f(x(t)) \leq g(x(t)) + \delta \leq tg(x_*) + (1 - t)g(y_*) + \delta \leq g(x_*) + \delta.$$

Hence, we can conclude that:

$$g(x_*) \geq f(x_*) + \frac{\mu\tau^2}{2} - \delta$$

885 This immediately contradicts the fact that $|g(x_*) - f(x_*)| \leq \delta$ as $\tau^2 > 4\delta/\mu$. This completes the
886 proof. \square

887 Finally, the claim in Lemma B.2 is established by combining the arguments of B.3 and Lemma
888 B.4. \square

889 We use Lemma B.2 to complete the proof of Theorem 4.1. With $\mu = (1/n_s, \dots, 1/n_s)$ and
890 $\nu = (1/n_t, \dots, 1/n_t)$ The oracle EGW optimization problem (with respect to $\mathbf{P}_s, \mathbf{P}_t$) can be
891 written as:

$$\begin{aligned} \mathcal{L}(\pi) &= \frac{1}{2} \sum_{ijkl} (\mathbf{P}_{s,ik} - \mathbf{P}_{t,jl}) \pi_{ij} \pi_{kl} + \epsilon \sum_{ij} \pi_{ij} \log(n_s n_t \pi_{ij}) \\ &= \frac{1}{2} \sum_{ijkl} \mathbf{P}_{s,ik}^2 \pi_{ij} \pi_{kl} + \frac{1}{2} \sum_{ijkl} \mathbf{P}_{t,jl}^2 \pi_{ij} \pi_{kl} - \sum_{ijkl} \mathbf{P}_{s,ik} \mathbf{P}_{t,jl} \pi_{ij} \pi_{kl} + \epsilon \sum_{ij} \pi_{ij} \log \pi_{ij} + \log(n_s n_t) \underbrace{\epsilon \sum_{ij} \pi_{ij}}_{=1} \\ &= \frac{1}{2} \|\mathbf{P}_s\|_F^2 + \frac{1}{2} \|\mathbf{P}_t\|_F^2 - \sum_{ijkl} \mathbf{P}_{s,ik} \mathbf{P}_{t,jl} \pi_{ij} \pi_{kl} + \epsilon \sum_{ij} \pi_{ij} \log \pi_{ij} + \epsilon \log(n_s n_t). \end{aligned}$$

892 It is immediate that the first, second, and fourth terms do not contribute to the estimation of π .
893 Ignoring them, we redefine the oracle loss function as:

$$f(\pi) = - \sum_{ijkl} \mathbf{P}_{s,ik} \mathbf{P}_{t,jl} \pi_{ij} \pi_{kl} + \epsilon \sum_{ij} \pi_{ij} \log \pi_{ij} = -2\text{tr}(\pi^\top \mathbf{P}_s \pi \mathbf{P}_t) + \epsilon \sum_{ij} \pi_{ij} \log \pi_{ij}$$

Similarly, we define $g(\cdot)$ as the sample version (with respect to $(\hat{\mathbf{P}}_s^{\text{init}}, \hat{\mathbf{P}}_t^{\text{init}})$):

$$g(\pi) = - \sum_{ijkl} \hat{\mathbf{P}}_{s,ik} \hat{\mathbf{P}}_{t,jl} \pi_{ij} \pi_{kl} + \epsilon \sum_{ij} \pi_{ij} \log \pi_{ij} = -2\text{tr}(\pi^\top \hat{\mathbf{P}}_s \pi \hat{\mathbf{P}}_t) + \epsilon \sum_{ij} \pi_{ij} \log \pi_{ij}.$$

894 For notation simplicity define $\delta_1 = \|\mathbf{P}_s - \hat{\mathbf{P}}_s\|_{\infty, \infty}$ and $\delta_2 = \|\mathbf{P}_t - \hat{\mathbf{P}}_t\|_{\infty, \infty}$. We have:

$$\begin{aligned} \frac{1}{2} |f(\pi) - g(\pi)| &= \left| \text{tr}(\pi^\top \mathbf{P}_s \pi \mathbf{P}_t^\top) - \text{tr}(\pi^\top \hat{\mathbf{P}}_s \pi \hat{\mathbf{P}}_t^\top) \right| \\ &= \left| \sum_{i,j} \pi_{ij} \left(\mathbf{P}_s \pi \mathbf{P}_t^\top - \hat{\mathbf{P}}_s \pi \hat{\mathbf{P}}_t^\top \right)_{ij} \right| \\ &\leq \max_{i,j} \left| \mathbf{P}_s \pi \mathbf{P}_t^\top - \hat{\mathbf{P}}_s \pi \hat{\mathbf{P}}_t^\top \right|_{ij} \underbrace{\sum_{i,j} \pi_{ij}}_{=1} \\ &= \max_{i,j} \left| \mathbf{P}_s \pi \mathbf{P}_t^\top - \hat{\mathbf{P}}_s \pi \hat{\mathbf{P}}_t^\top \right|_{ij} \\ &\leq \max_{i,j} \left| \mathbf{P}_s \pi \mathbf{P}_t^\top - \mathbf{P}_s \pi \hat{\mathbf{P}}_t^\top \right|_{ij} + \max_{i,j} \left| \mathbf{P}_s \pi \hat{\mathbf{P}}_t^\top - \hat{\mathbf{P}}_s \pi \hat{\mathbf{P}}_t^\top \right|_{ij} \\ &= \max_{i,j} \left| \sum_k (\mathbf{P}_s \pi)_{ik} (\mathbf{P}_t - \hat{\mathbf{P}}_t)_{jk} \right| + \max_{i,j} \left| \sum_k (\mathbf{P}_s - \hat{\mathbf{P}}_s)_{ik} (\pi \hat{\mathbf{P}}_t^\top)_{kj} \right| \\ &\leq \|\mathbf{P}_t - \hat{\mathbf{P}}_t\|_{\infty, \infty} \max_i \sum_k |(\mathbf{P}_s \pi)_{ik}| + \|\hat{\mathbf{P}}_s - \mathbf{P}_s\|_{\infty, \infty} \max_j \sum_k |(\pi \hat{\mathbf{P}}_t^\top)_{kj}| \\ &\leq \delta_2 \max_i \sum_k |(\mathbf{P}_s \pi)_{ik}| + \delta_1 \max_j \sum_k |(\pi \hat{\mathbf{P}}_t^\top)_{kj}| \\ &\leq \delta_2 \max_i \sum_k \left| \sum_l \mathbf{P}_{s,il} \pi_{lk} \right| + \delta_1 \max_j \sum_k \left| \sum_l \pi_{kl} \mathbf{P}_{t,jl} \right| \\ &\leq \delta_2 \|\mathbf{P}_s\|_\infty \sum_{kl} \pi_{kl} + \delta_1 \|\hat{\mathbf{P}}_t\|_\infty \sum_{kl} \pi_{kl} := \delta_2 \|\mathbf{P}_s\|_\infty + \delta_1 \|\tilde{\mathbf{P}}_t\|_\infty = \delta_1 + \delta_2. \end{aligned}$$

Therefore, we conclude that

$$\|f - g\|_\infty \leq \|\mathbf{P}_s - \hat{\mathbf{P}}_s\|_{\infty, \infty} + \|\mathbf{P}_t - \hat{\mathbf{P}}_t\|_{\infty, \infty} := \Delta_{\text{pert}}.$$

895 Now, from the definition of f and g , we also have:

$$\begin{aligned}\nabla_{\pi} f(\pi) &= -(\mathbf{P}_s \otimes \mathbf{P}_t + \mathbf{P}_s^{\top} \otimes \mathbf{P}_t^{\top}) \text{vec}(\pi) + \epsilon((1 + \log \pi_{ij}))_{i,j} \\ \nabla_{\pi} g(\pi) &= -(\hat{\mathbf{P}}_s \otimes \hat{\mathbf{P}}_t + \hat{\mathbf{P}}_s^{\top} \otimes \hat{\mathbf{P}}_t^{\top}) \text{vec}(\pi) + \epsilon((1 + \log \pi_{ij}))_{i,j} \\ \nabla_{\pi}^2 f(\pi) &= -(\mathbf{P}_s \otimes \mathbf{P}_t + \mathbf{P}_s^{\top} \otimes \mathbf{P}_t^{\top}) + \epsilon \text{diag} \left(\left\{ \frac{1}{\pi_{ij}} \right\} \right) \\ \nabla_{\pi}^2 g(\pi) &= -(\hat{\mathbf{P}}_s \otimes \hat{\mathbf{P}}_t + \hat{\mathbf{P}}_s^{\top} \otimes \hat{\mathbf{P}}_t^{\top}) + \epsilon \text{diag} \left(\left\{ \frac{1}{\pi_{ij}} \right\} \right)\end{aligned}$$

As a consequence, we have:

$$\|\nabla_{\pi}(g - f)(\pi)\|_F \leq 2 \left\| (\mathbf{P}_s \otimes \mathbf{P}_t) - (\hat{\mathbf{P}}_s \otimes \hat{\mathbf{P}}_t) \right\|_{\text{op}} \|\pi\|_F.$$

Hence, we have:

$$\kappa = 2 \left\| (\mathbf{P}_s \otimes \mathbf{P}_t) - (\hat{\mathbf{P}}_s \otimes \hat{\mathbf{P}}_t) \right\|_{\text{op}}.$$

Now, from the Hessian, we know $\nabla_{\pi}^2 f(\pi^*) \succ 0$ if $\max_{ij} \pi_{ij}^* < \epsilon/(2\|\mathbf{P}_s \otimes \mathbf{P}_t\|_{\text{op}})$. This condition is met as per our assumption that

$$\|\pi^*\|_{\infty} \leq \frac{\epsilon}{C_1 \|\mathbf{P}_s \otimes \mathbf{P}_t\|_{\text{op}}}.$$

with $C_1 > 2$. Fix C_2 such that $(1/C_2) + (1/C_1) < (1/2)$, and set $\tau = \epsilon/(C_2 \|\mathbf{P}_s \otimes \mathbf{P}_t\|_{\text{op}})$. Then, for any $\|\pi - \pi^*\|_F \leq \tau$,

$$\max_{ij} \pi_{ij} \leq \tau + \max_{ij} \pi_{ij}^* \leq \frac{\epsilon}{C_2 \|\mathbf{P}_s \otimes \mathbf{P}_t\|_{\text{op}}} + \frac{\epsilon}{C_1 \|\mathbf{P}_s \otimes \mathbf{P}_t\|_{\text{op}}} < \frac{\epsilon}{2\|\mathbf{P}_s \otimes \mathbf{P}_t\|_{\text{op}}}.$$

In this neighborhood, we have:

$$\inf_{\pi \in B(\pi^*; \tau)} \inf_{\|v\|_2=1} v^{\top} \nabla_{\pi}^2 f(\pi) v \geq \left\{ \left(\frac{1}{C_1} + \frac{1}{C_2} \right)^{-1} - 2 \right\} \|\mathbf{P}_s \otimes \mathbf{P}_t\|_{\text{op}} := \frac{1}{C_3} \|\mathbf{P}_s \otimes \mathbf{P}_t\|_{\text{op}} := \frac{\mu}{2}.$$

Therefore, using Lemma B.2 we can conclude:

$$\|\hat{\pi} - \pi^*\|_F \leq \frac{20 \left\| (\mathbf{P}_s \otimes \mathbf{P}_t) - (\hat{\mathbf{P}}_s \otimes \hat{\mathbf{P}}_t) \right\|_{\text{op}}}{\|\mathbf{P}_s \otimes \mathbf{P}_t\|_{\text{op}}} \quad \text{as soon as} \quad \Delta_{\text{pert}} \leq \frac{\tau^2 \mu}{4}.$$

896 By our definition of δ and μ , we have:

$$\frac{\tau^2 \mu}{4} = \frac{\left(\frac{\epsilon}{C_2 \|\mathbf{P}_s \otimes \mathbf{P}_t\|_{\text{op}}} \right)^2}{4} \cdot \frac{2\|\mathbf{P}_s \otimes \mathbf{P}_t\|_{\text{op}}}{C_3}.$$

897 Recall that our assumption is:

$$\Delta_{\text{pert}} = \|\mathbf{P}_s - \hat{\mathbf{P}}_s\|_{\infty, \infty} + \|\mathbf{P}_t - \hat{\mathbf{P}}_t\|_{\infty, \infty} \leq \frac{\tau^2 \mu}{4}.$$

898 Substituting the expression for $\tau^2 \mu$:

$$\Delta_{\text{pert}} \leq \frac{\epsilon^2}{2C_2^2 C_3 (\|\mathbf{P}_s \otimes \mathbf{P}_t\|_{\text{op}})}.$$

899 This guarantees that the perturbed solution $\hat{\pi}$ remains in the neighborhood $B(\pi^*, \tau)$ of the true
900 solution π^* . Thus, the stability of the optimal transport plan is ensured under perturbations of the
901 graphon estimators, completing the proof.

C Additional Simulation Results

C.1 Additional results for Impact of Density Shift

To evaluate the robustness of GTRANS under density discrepancies between the source and target domains, we introduce a density shift parameter $\lambda \in [-0.5, 0.5]$ to simulate structured perturbations, defined as $f_s(u, v) = f_t(u, v) + \xi$, where $\xi \sim U(0, \lambda)$ if $\lambda > 0$ and $\xi \sim U(\lambda, 0)$ if $\lambda < 0$. We investigate the performance of GTRANS-GW and GTRANS-EGW across 10 different graphon functions (IDs: 1 to 10) with a fixed target sample size of 50 and a source sample size of 200. Each experiment is repeated 50 times to evaluate the adaptability of the models under varying levels of density shift. The results are summarized in Figure C1.

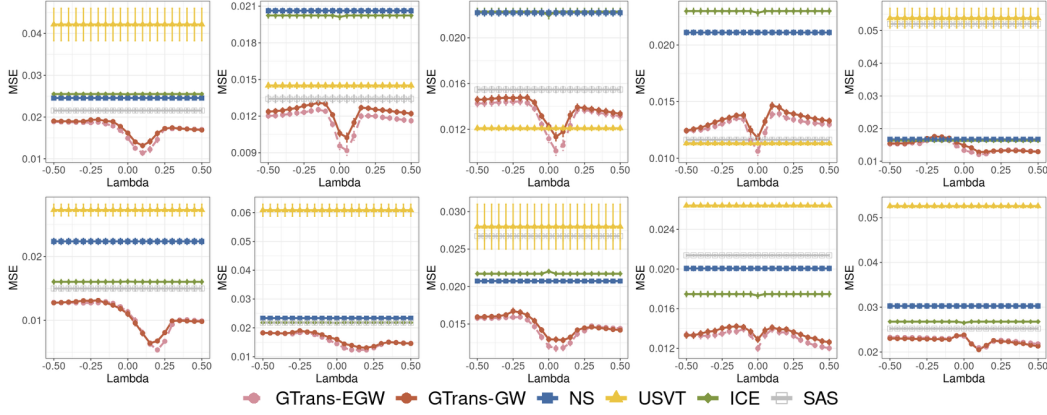


Figure C1: MSE performance of five methods as the density shift parameter λ varies from -0.5 to 0.5 , with error bars representing ± 0.1 standard deviations. GTRANS-GW (red circles with solid line), GTRANS-EGW (pink circles with dashed line), NS (blue squares with solid line), USVT (yellow triangles with solid line), ICE (green diamonds with solid line), SAS (gray hollow squares with solid line). Top row: graphons 1-5; bottom row: graphons 5-10.

Figure C1 shows the average MSE of different methods as the density shift parameter λ varies from -0.5 to 0.5 . Both **GTRANS-GW (red solid line)** and **GTRANS-EGW (pink dashed line)** demonstrate substantial improvements over the baseline methods (NS, USVT, ICE, and SAS), which remain constant across all λ values. Notably, as λ shifts from negative to positive, GTRANS methods effectively adjust their alignment, capturing structural changes and minimizing MSE. This result underscores the capability of GTRANS-GW and GTRANS-EGW to dynamically adapt to source-target discrepancies, achieving consistently lower MSE across all scenarios. Also, we notice that for dense graphons like Graphon 2 (density = 0.5) and Graphon 9 (density = 0.5), GTRANS-GW and GTRANS-EGW exhibit symmetric U-shaped MSE curves around zero, indicating balanced performance regardless of source density. In contrast, for sparse graphons like Graphon 6 (density = 0.13) and Graphon 10 (density = 0.19), the lowest MSE is achieved with a slightly denser source, highlighting the need for stronger initial signals when the target is sparse. Unique structural patterns in Graphon 5 (anti-diagonal) and Graphon 7 (oscillatory) cause non-monotonic fluctuations under density shift, reflecting sensitivity to structural misalignments.

C.2 Additional Simulation for Transfer Between Different Graphons

We conduct simulations across ten distinct graphon structures, indexed from 1 to 10. For each graphon, we generate a target domain with a fixed sample size of $n_t = 50$ and a source domain with $n_s = 500$, simulating realistic structural perturbations by adding uniform noise sampled from $[-0.01, 0.01]$. The evaluation includes USVT, ICE, and SAS, and our proposed transfer learning frameworks: GTRANS-GW and GTRANS-EGW ($\epsilon = 0.01$). For comprehensive analysis, we compute the Mean Squared Error (MSE) for each method and aggregate the results.

Table C2: Comparison of different methods for various transfer scenarios. Best-performing methods are **bolded**.

Scenario	GTRANS-GW	GTRANS-EGW	NS	USVT	ICE	SAS
1 \rightarrow 2	1.4 \pm 0.3	1.3 \pm 0.3	2.1 \pm 0.2	1.5 \pm 0.3	2.1 \pm 0.2	1.4 \pm 0.4
1 \rightarrow 3	1.6 \pm 0.3	1.5 \pm 0.3	2.3 \pm 0.3	1.2 \pm 0.2	2.2 \pm 0.3	1.6 \pm 0.3
1 \rightarrow 4	1.3 \pm 0.1	1.3 \pm 0.2	2.1 \pm 0.2	1.0 \pm 0.1	2.3 \pm 0.2	1.1 \pm 0.2
1 \rightarrow 5	1.6 \pm 0.2	1.5 \pm 0.2	1.7 \pm 0.1	3.5 \pm 1.5	1.7 \pm 0.2	5.5 \pm 1.0
1 \rightarrow 6	1.0 \pm 0.2	1.1 \pm 0.2	2.1 \pm 0.3	3.0 \pm 0.9	1.7 \pm 0.3	1.6 \pm 0.4
1 \rightarrow 7	1.1 \pm 0.4	0.9 \pm 0.4	2.3 \pm 0.3	5.9 \pm 2.7	2.2 \pm 0.3	2.4 \pm 0.6
1 \rightarrow 8	1.5 \pm 0.3	1.4 \pm 0.3	2.0 \pm 0.2	1.5 \pm 0.2	2.1 \pm 0.2	2.8 \pm 0.7
1 \rightarrow 9	1.8 \pm 0.4	1.7 \pm 0.3	2.0 \pm 0.2	2.6 \pm 0.1	1.7 \pm 0.2	2.1 \pm 0.1
1 \rightarrow 10	2.2 \pm 0.3	2.2 \pm 0.3	3.1 \pm 0.6	5.3 \pm 0.6	2.7 \pm 0.3	2.6 \pm 0.3
2 \rightarrow 1	1.9 \pm 0.2	1.8 \pm 0.2	2.5 \pm 0.3	5.1 \pm 4.6	2.7 \pm 0.3	2.1 \pm 0.3
2 \rightarrow 3	1.6 \pm 0.3	1.6 \pm 0.3	2.3 \pm 0.3	1.2 \pm 0.2	2.2 \pm 0.3	1.6 \pm 0.3
2 \rightarrow 4	1.3 \pm 0.2	1.3 \pm 0.2	2.1 \pm 0.2	1.0 \pm 0.1	2.3 \pm 0.2	1.1 \pm 0.2
2 \rightarrow 5	1.6 \pm 0.2	1.5 \pm 0.2	1.7 \pm 0.1	3.5 \pm 1.5	1.7 \pm 0.2	5.5 \pm 1.0
2 \rightarrow 6	1.0 \pm 0.2	1.0 \pm 0.2	2.1 \pm 0.3	3.0 \pm 0.9	1.7 \pm 0.3	1.6 \pm 0.4
2 \rightarrow 7	1.6 \pm 0.2	1.6 \pm 0.2	2.3 \pm 0.3	5.9 \pm 2.7	2.2 \pm 0.3	2.4 \pm 0.6
2 \rightarrow 8	1.5 \pm 0.2	1.5 \pm 0.2	2.0 \pm 0.2	1.5 \pm 0.2	2.1 \pm 0.2	2.8 \pm 0.7
2 \rightarrow 9	1.8 \pm 0.5	1.6 \pm 0.4	2.0 \pm 0.2	2.6 \pm 0.1	1.7 \pm 0.2	2.1 \pm 0.1
2 \rightarrow 10	2.2 \pm 0.3	2.2 \pm 0.3	3.1 \pm 0.6	5.3 \pm 0.6	2.7 \pm 0.3	2.6 \pm 0.3
3 \rightarrow 1	1.6 \pm 0.5	1.7 \pm 0.2	2.5 \pm 0.3	5.1 \pm 4.6	2.7 \pm 0.3	2.1 \pm 0.3
3 \rightarrow 2	1.5 \pm 0.3	1.4 \pm 0.3	2.1 \pm 0.2	1.5 \pm 0.3	2.1 \pm 0.2	1.4 \pm 0.4
3 \rightarrow 4	1.4 \pm 0.2	1.3 \pm 0.2	2.1 \pm 0.2	1.0 \pm 0.1	2.3 \pm 0.2	1.1 \pm 0.2
3 \rightarrow 5	1.4 \pm 0.2	1.4 \pm 0.2	1.7 \pm 0.1	3.5 \pm 1.5	1.7 \pm 0.2	5.5 \pm 1.0
3 \rightarrow 6	0.9 \pm 0.2	1.0 \pm 0.2	2.1 \pm 0.3	3.0 \pm 0.9	1.7 \pm 0.3	1.6 \pm 0.4
3 \rightarrow 7	1.5 \pm 0.2	1.5 \pm 0.2	2.3 \pm 0.3	5.9 \pm 2.7	2.2 \pm 0.3	2.4 \pm 0.6
3 \rightarrow 8	1.5 \pm 0.3	1.4 \pm 0.3	2.0 \pm 0.2	1.5 \pm 0.2	2.1 \pm 0.2	2.8 \pm 0.7
3 \rightarrow 9	1.9 \pm 0.3	1.8 \pm 0.3	2.0 \pm 0.2	2.6 \pm 0.1	1.7 \pm 0.2	2.1 \pm 0.1
3 \rightarrow 10	2.2 \pm 0.3	2.1 \pm 0.3	3.1 \pm 0.6	5.3 \pm 0.6	2.7 \pm 0.3	2.6 \pm 0.3
4 \rightarrow 1	1.7 \pm 0.2	1.7 \pm 0.2	2.5 \pm 0.3	5.1 \pm 4.6	2.7 \pm 0.3	2.1 \pm 0.3
4 \rightarrow 2	1.3 \pm 0.3	1.3 \pm 0.3	2.1 \pm 0.2	1.5 \pm 0.3	2.1 \pm 0.2	1.4 \pm 0.4
4 \rightarrow 3	1.4 \pm 0.3	1.3 \pm 0.3	2.3 \pm 0.3	1.2 \pm 0.2	2.2 \pm 0.3	1.6 \pm 0.3
4 \rightarrow 5	1.3 \pm 0.2	1.3 \pm 0.2	1.7 \pm 0.1	3.5 \pm 1.5	1.7 \pm 0.2	5.5 \pm 1.0
4 \rightarrow 6	1.0 \pm 0.2	1.1 \pm 0.2	2.1 \pm 0.3	3.0 \pm 0.9	1.7 \pm 0.3	1.6 \pm 0.4
4 \rightarrow 7	1.5 \pm 0.2	1.5 \pm 0.2	2.3 \pm 0.3	5.9 \pm 2.7	2.2 \pm 0.3	2.4 \pm 0.6
4 \rightarrow 8	1.4 \pm 0.2	1.4 \pm 0.2	2.0 \pm 0.2	1.5 \pm 0.2	2.1 \pm 0.2	2.8 \pm 0.7
4 \rightarrow 9	1.5 \pm 0.3	1.4 \pm 0.3	2.0 \pm 0.2	2.6 \pm 0.1	1.7 \pm 0.2	2.1 \pm 0.1
4 \rightarrow 10	2.2 \pm 0.3	2.1 \pm 0.3	3.1 \pm 0.6	5.3 \pm 0.6	2.7 \pm 0.3	2.6 \pm 0.3
5 \rightarrow 1	2.0 \pm 0.3	2.1 \pm 0.3	2.5 \pm 0.3	5.1 \pm 4.6	2.6 \pm 0.2	2.1 \pm 0.3
5 \rightarrow 2	1.7 \pm 0.3	1.6 \pm 0.3	2.1 \pm 0.2	1.5 \pm 0.3	2.1 \pm 0.2	1.4 \pm 0.4
5 \rightarrow 3	1.7 \pm 0.3	1.7 \pm 0.3	2.3 \pm 0.3	1.2 \pm 0.2	2.2 \pm 0.3	1.6 \pm 0.3
5 \rightarrow 4	1.4 \pm 0.1	1.5 \pm 0.2	2.1 \pm 0.2	1.0 \pm 0.1	2.3 \pm 0.2	1.1 \pm 0.2
5 \rightarrow 6	1.2 \pm 0.2	1.2 \pm 0.2	2.1 \pm 0.3	3.0 \pm 0.9	1.7 \pm 0.3	1.6 \pm 0.4
5 \rightarrow 7	1.8 \pm 0.3	1.8 \pm 0.3	2.3 \pm 0.3	5.9 \pm 2.7	2.2 \pm 0.2	2.4 \pm 0.6
5 \rightarrow 8	1.5 \pm 0.2	1.7 \pm 0.2	2.0 \pm 0.2	1.5 \pm 0.2	2.1 \pm 0.2	2.8 \pm 0.7
5 \rightarrow 9	2.1 \pm 0.3	2.0 \pm 0.3	2.0 \pm 0.2	2.6 \pm 0.1	1.8 \pm 0.3	2.1 \pm 0.1
5 \rightarrow 10	2.6 \pm 0.3	2.5 \pm 0.3	3.1 \pm 0.6	5.3 \pm 0.6	2.8 \pm 0.3	2.6 \pm 0.3
6 \rightarrow 1	2.0 \pm 0.3	2.0 \pm 0.3	2.5 \pm 0.3	5.1 \pm 4.6	2.7 \pm 0.3	2.1 \pm 0.3
6 \rightarrow 2	1.3 \pm 0.3	1.3 \pm 0.3	2.1 \pm 0.2	1.5 \pm 0.3	2.1 \pm 0.2	1.4 \pm 0.4
6 \rightarrow 3	1.5 \pm 0.3	1.5 \pm 0.3	2.3 \pm 0.3	1.2 \pm 0.2	2.2 \pm 0.3	1.6 \pm 0.3
6 \rightarrow 4	1.2 \pm 0.2	1.2 \pm 0.2	2.1 \pm 0.2	1.0 \pm 0.1	2.3 \pm 0.2	1.1 \pm 0.2
6 \rightarrow 5	1.6 \pm 0.2	1.5 \pm 0.2	1.7 \pm 0.1	3.5 \pm 1.5	1.7 \pm 0.2	5.5 \pm 1.0

Scenario	GTRANS-GW	GTRANS-EGW	NS	USVT	ICE	SAS
6 → 7	1.8 ± 0.3	1.8 ± 0.3	2.3 ± 0.3	5.9 ± 2.7	2.2 ± 0.3	2.4 ± 0.6
6 → 8	1.6 ± 0.2	1.5 ± 0.2	2.0 ± 0.2	1.5 ± 0.2	2.1 ± 0.2	2.8 ± 0.7
6 → 9	1.5 ± 0.4	1.6 ± 0.4	2.0 ± 0.2	2.6 ± 0.1	1.7 ± 0.2	2.1 ± 0.1
6 → 10	2.3 ± 0.3	2.3 ± 0.3	3.1 ± 0.6	5.3 ± 0.6	2.7 ± 0.3	2.6 ± 0.3
7 → 1	2.0 ± 0.3	1.9 ± 0.3	2.5 ± 0.3	5.1 ± 4.6	2.7 ± 0.3	2.1 ± 0.3
7 → 2	1.3 ± 0.3	1.3 ± 0.3	2.1 ± 0.2	1.5 ± 0.3	2.1 ± 0.2	1.4 ± 0.4
7 → 3	1.6 ± 0.3	1.5 ± 0.3	2.3 ± 0.3	1.2 ± 0.2	2.2 ± 0.3	1.6 ± 0.3
7 → 4	1.3 ± 0.2	1.2 ± 0.2	2.1 ± 0.2	1.0 ± 0.1	2.3 ± 0.2	1.1 ± 0.2
7 → 5	1.6 ± 0.2	1.5 ± 0.2	1.7 ± 0.1	3.5 ± 1.5	1.7 ± 0.2	5.5 ± 1.0
7 → 6	0.9 ± 0.2	1.0 ± 0.3	2.1 ± 0.3	3.0 ± 0.9	1.7 ± 0.3	1.6 ± 0.4
7 → 8	1.6 ± 0.2	1.6 ± 0.2	2.0 ± 0.2	1.5 ± 0.2	2.1 ± 0.2	2.8 ± 0.7
7 → 9	1.7 ± 0.4	1.7 ± 0.4	2.0 ± 0.2	2.6 ± 0.1	1.7 ± 0.2	2.1 ± 0.1
7 → 10	2.3 ± 0.4	2.2 ± 0.3	3.1 ± 0.6	5.3 ± 0.6	2.7 ± 0.3	2.6 ± 0.3
8 → 1	1.8 ± 0.3	1.5 ± 0.3	2.5 ± 0.3	5.1 ± 4.6	2.7 ± 0.3	2.1 ± 0.3
8 → 2	1.4 ± 0.3	1.4 ± 0.3	2.1 ± 0.2	1.5 ± 0.3	2.1 ± 0.2	1.4 ± 0.4
8 → 3	1.7 ± 0.3	1.6 ± 0.3	2.3 ± 0.3	1.2 ± 0.2	2.2 ± 0.3	1.6 ± 0.3
8 → 4	1.3 ± 0.2	1.3 ± 0.2	2.1 ± 0.2	1.0 ± 0.1	2.3 ± 0.2	1.1 ± 0.2
8 → 5	1.6 ± 0.2	1.5 ± 0.2	1.7 ± 0.1	3.5 ± 1.5	1.7 ± 0.2	5.5 ± 1.0
8 → 6	1.0 ± 0.2	1.0 ± 0.2	2.1 ± 0.3	3.0 ± 0.9	1.7 ± 0.3	1.6 ± 0.4
8 → 7	1.3 ± 0.3	1.4 ± 0.3	2.3 ± 0.3	5.9 ± 2.7	2.2 ± 0.3	2.4 ± 0.6
8 → 9	1.9 ± 0.4	1.9 ± 0.3	2.0 ± 0.2	2.6 ± 0.1	1.7 ± 0.2	2.1 ± 0.1
8 → 10	2.3 ± 0.3	2.3 ± 0.3	3.1 ± 0.6	5.3 ± 0.6	2.7 ± 0.3	2.6 ± 0.3
9 → 1	2.4 ± 0.3	2.3 ± 0.2	2.5 ± 0.3	5.1 ± 4.6	2.7 ± 0.3	2.1 ± 0.3
9 → 2	1.5 ± 0.3	1.5 ± 0.3	2.1 ± 0.2	1.5 ± 0.3	2.1 ± 0.2	1.4 ± 0.4
9 → 3	1.9 ± 0.3	1.6 ± 0.3	2.3 ± 0.3	1.2 ± 0.2	2.2 ± 0.3	1.6 ± 0.3
9 → 4	1.6 ± 0.2	1.6 ± 0.2	2.1 ± 0.2	1.0 ± 0.1	2.3 ± 0.2	1.1 ± 0.2
9 → 5	1.6 ± 0.2	1.3 ± 0.2	1.7 ± 0.1	3.5 ± 1.5	1.7 ± 0.2	5.5 ± 1.0
9 → 6	1.2 ± 0.2	1.2 ± 0.2	2.1 ± 0.3	3.0 ± 0.9	1.7 ± 0.3	1.6 ± 0.4
9 → 7	1.9 ± 0.2	1.9 ± 0.3	2.3 ± 0.3	5.9 ± 2.7	2.2 ± 0.3	2.4 ± 0.6
9 → 8	2.0 ± 0.2	1.9 ± 0.2	2.0 ± 0.2	1.5 ± 0.2	2.1 ± 0.2	2.8 ± 0.7
9 → 10	2.2 ± 0.3	2.3 ± 0.3	3.1 ± 0.6	5.3 ± 0.6	2.7 ± 0.3	2.6 ± 0.3
10 → 1	2.1 ± 0.3	2.0 ± 0.3	2.5 ± 0.3	5.1 ± 4.6	2.7 ± 0.3	2.1 ± 0.3
10 → 2	1.3 ± 0.3	1.2 ± 0.3	2.1 ± 0.2	1.5 ± 0.3	2.1 ± 0.2	1.4 ± 0.4
10 → 3	1.6 ± 0.3	1.5 ± 0.3	2.3 ± 0.3	1.2 ± 0.2	2.2 ± 0.3	1.6 ± 0.3
10 → 4	1.2 ± 0.2	1.1 ± 0.2	2.1 ± 0.2	1.0 ± 0.1	2.3 ± 0.2	1.1 ± 0.2
10 → 5	1.6 ± 0.2	1.5 ± 0.2	1.7 ± 0.1	3.5 ± 1.5	1.7 ± 0.2	5.5 ± 1.0
10 → 6	1.3 ± 0.4	1.4 ± 0.3	2.1 ± 0.3	3.0 ± 0.9	1.7 ± 0.3	1.6 ± 0.4
10 → 7	1.9 ± 0.3	1.8 ± 0.3	2.3 ± 0.3	5.9 ± 2.7	2.2 ± 0.3	2.4 ± 0.6
10 → 8	1.6 ± 0.2	1.5 ± 0.2	2.0 ± 0.2	1.5 ± 0.2	2.1 ± 0.2	2.8 ± 0.7
10 → 9	1.5 ± 0.3	1.5 ± 0.4	2.0 ± 0.2	2.6 ± 0.1	1.7 ± 0.2	2.1 ± 0.1

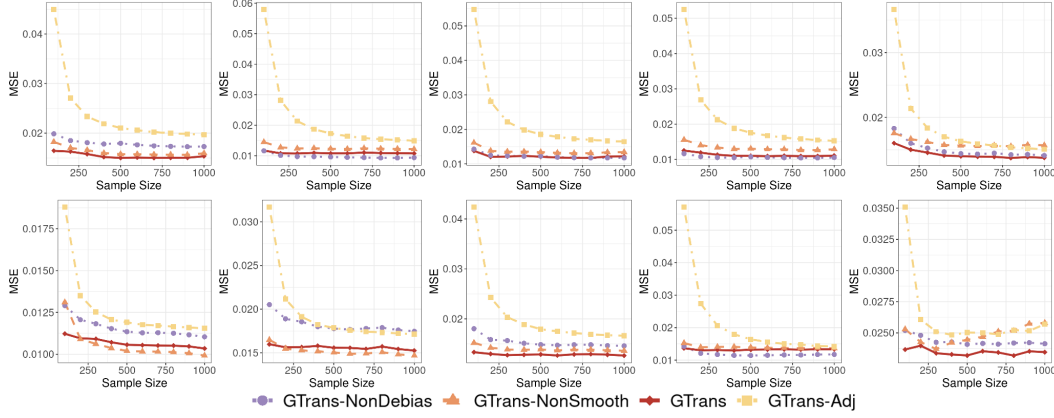
932 C.3 Ablation Study

933 Figure C2 presents the average estimation errors (MSE) of four configurations: **GTRANS** (red dia-
934 mond with solid line), **GTRANS-NonDebias** (purple circle with dotted line), **GTRANS-NonSmooth**
935 (orange triangle with dashed line), and **GTRANS-Adj** (yellow square with dash-dotted line). The
936 x-axis represents the source sample size, and the y-axis shows the average MSE over 50 runs.

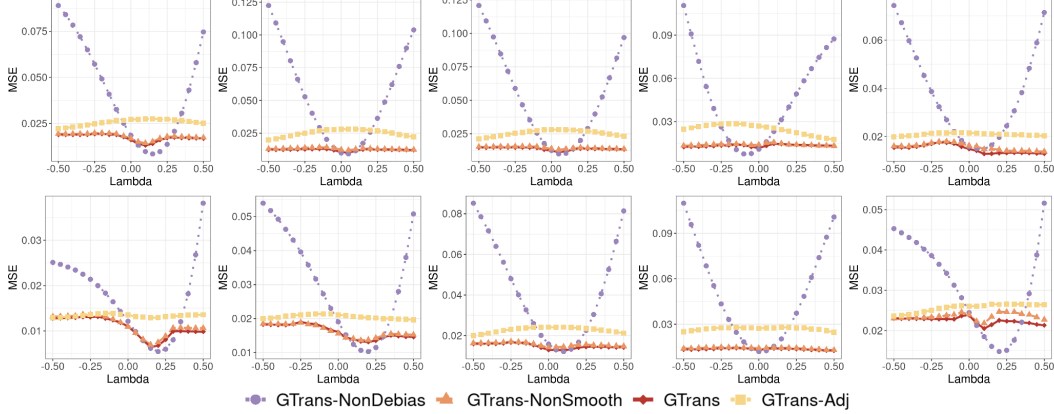
937 The four configurations are derived as follows: **GTRANS**: This is the full version, including all three
938 steps from Algorithm 1. **GTRANS-NonDebias**: This variant removes the debiasing step (Step 3),
939 directly using the transferred estimator $\hat{\mathbf{P}}_t^{trans2}$ as the final output. **GTRANS-NonSmooth**: This
940 variant omits the neighborhood smoothing steps in the initial estimation (Step 1). **GTRANS-Adj**:
941 This variant replaces the neighborhood smoothing (**Step 1**) entirely with raw adjacency matrices

942 A_s and A_t . In this setting, the optimal transport step (**Step 2**) operates directly on the unprocessed
 943 adjacency representations.

944 From the results, we observe that the complete **GTRANS** consistently outperforms its ablated variants
 945 across most graphon structures. The absence of smoothing (**GTRANS-NonSmooth**) introduces sharp
 946 oscillations in MSE, indicating the sensitivity to noise. **GTRANS-NonDebias** struggles to correct
 947 for transfer-induced discrepancies, especially when source-target alignment is imperfect. **GTRANS-Adj**
 948 shows consistently higher MSE, highlighting that raw adjacency matrices lack smoothness for
 949 effective GW-based alignment. This underscores the necessity of the smoothing step in GTRANS to
 950 achieve consistent and robust estimations. Therefore, the debiasing and smoothing mechanisms in
 951 GTRANS are not just additive; they are essential for robustness.



(a) Ablation Study with Increasing Source Sample Size.



(b) Ablation Study with Varying λ .

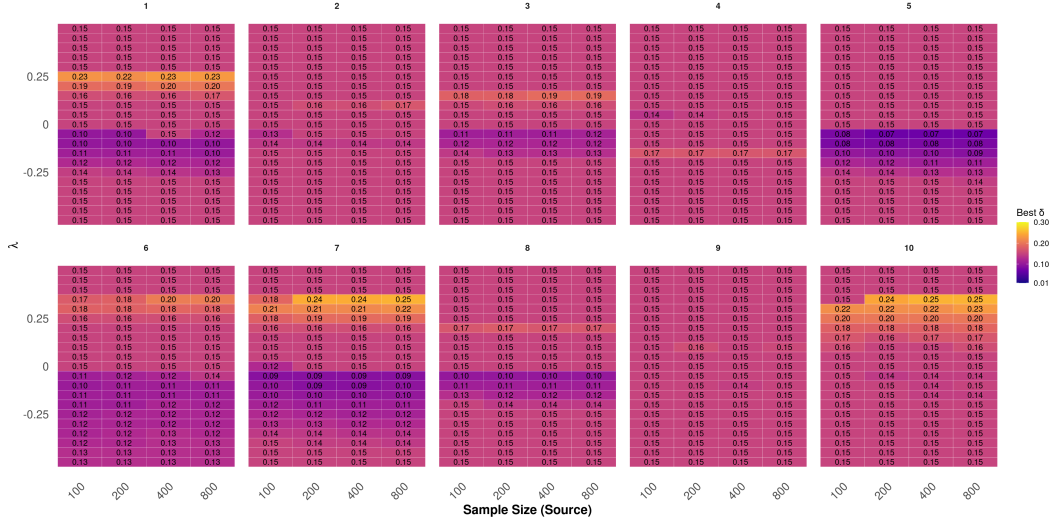
Figure C2: Ablation study of different estimation methods: (a) illustrates the impact of increasing source sample size on estimation accuracy, and (b) demonstrates the effect of varying density shift (λ) on the performance of GTRANS, GTRANS-NonDebias, GTRANS-NonSmooth, and GTRANS-Adj.

952 C.4 Hyperparameter Selection

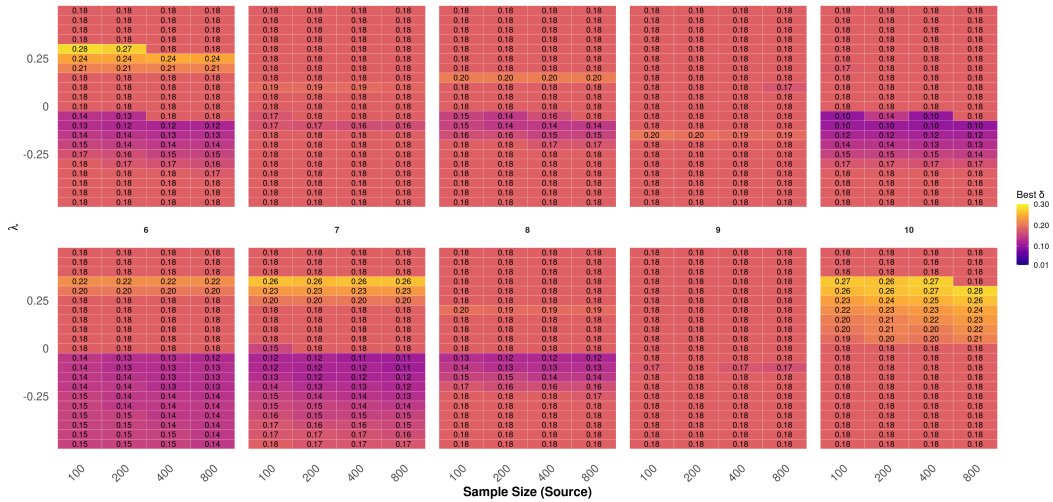
953 C.4.1 Debiasing Threshold δ Selection

954 **Threshold δ .** We evaluate the stability of the selected threshold δ^* under matched source and target
 955 distributions. Fixing the target size at $n_t = 50$, we vary the source size in $\{100, 200, 400, 800\}$ and
 956 generate both graphs from the same graphon (`graphon_id` $\in \{1, \dots, 10\}$). Source perturbations are
 957 introduced via a structural noise parameter $\lambda \in [-0.5, 0.5]$. Thresholds $\delta \in \{0.01, 0.02, \dots, 0.50\}$
 958 are evaluated using the mean squared error (MSE) between the estimated and true target graphons.

We evaluate a fixed threshold $\delta = 0.15$ across various source perturbations. If it aligns with the side of the optimal jump point determined by the GW distance d , it is selected; otherwise, the best threshold on that side is chosen. Empirically, $\delta = 0.15$ remains robust, while under EGW with $\epsilon = 0.01$, the optimal choice converges to $\delta = 0.18$.



(a) Optimal threshold δ across Graphon IDs for **GTRANS-GW**.



(b) Optimal threshold δ across Graphon IDs for **GTRANS-EGW**.

Figure C3: Heatmaps of the optimal threshold δ across different Graphon IDs. **GTRANS-GW** (top) shows that $\delta = 0.15$ is consistently optimal or near-optimal, while **GTRANS-EGW** ($\epsilon = 0.01$) (bottom) favors $\delta = 0.18$ in most scenarios, with slight increases under high perturbation.

Figure C3 shows that the optimal threshold δ depends on the graphon's inherent density and the perturbation level λ . For dense graphons, δ remains stable, reflecting robustness to structural shifts. In contrast, sparse graphons exhibit a sharp increase in δ under positive λ , as perturbations enhance connectivity and reduce noise sensitivity. The asymmetry arises from a bias–variance trade-off. Denser sources ($\lambda > 0$) yield high-quality but mismatched transfer, favoring delayed debiasing with larger δ^* ; sparser sources ($\lambda < 0$) require earlier correction, hence smaller δ^* . Besides, GW distance increases with denser sources due to structural mismatch, and decreases with sparser ones despite degraded transfer—supporting a larger δ for dense, and smaller for sparse sources.

971 C.4.2 Regularization Parameter ϵ Selection

972 **Threshold ϵ .** To determine the optimal regularization parameter ϵ in EGW, we perform K -fold
 973 cross-validation over a candidate set $\epsilon_{\text{list}} = \{0.001, 0.005, 0.01, 0.05, 0.1\}$. This process evaluates
 974 the MSE between the estimated and true target graphon. We observe that $\epsilon = 0.01$ is consistently
 975 selected as the best-performing choice across different settings. To further illustrate this, figure C4
 976 presents boxplots of the average MSE for each ϵ under varying source sizes 100, 200, 300, \dots , 1000
 977 with a fixed target size of $n_t = 50$. Each boxplot shows the MSE distribution across ten graphon
 978 types. While some graphons like Graphon 9 benefit from larger ϵ (e.g., 0.1) due to oscillatory patterns,
 979 $\epsilon = 0.01$ consistently achieves the lowest error, demonstrating its robustness across diverse structures.

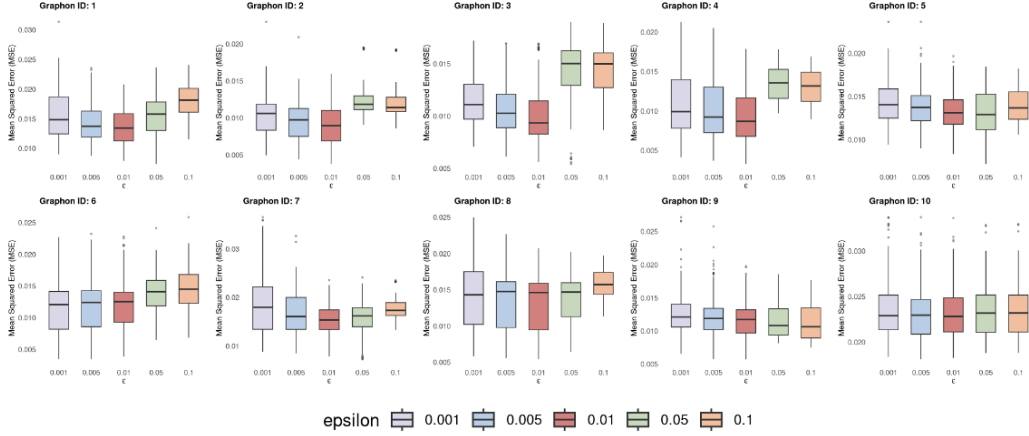


Figure C4: Boxplot comparison of different regularization parameters ϵ for GTRANS-EGW across varying source sample sizes. The results show that $\epsilon = 0.01$ (red) consistently maintains lower variance and median MSE in most graphons, indicating its robustness.

980 D Additional Real Data Results

981 D.1 Additional results for graph augmentation task

982 In Transfer Learning, selecting the optimal source dataset and identifying the best class correspon-
 983 dence are crucial for effective knowledge transfer, as the structural characteristics of source labels
 984 significantly impact performance. To determine optimal class-to-class transfer pairings, we first
 985 estimate graphons separately for each source and target class using neighborhood smoothing [61].
 986 Subsequently, we compute the pairwise Gromov–Wasserstein (GW) distances as well as the Entropic
 987 Gromov–Wasserstein (EGW) distances with a commonly chosen regularization parameter $\epsilon = 0.01$
 988 for each target-source graphon pair, identifying the best-matched class from the source domain for
 989 each target class based on structural similarity. We summarize the key structural statistics of all
 990 datasets used in our experiments in Table D3. Table D4 summarizes the GW distances calculated
 991 between each pair of source and target labels across the considered datasets, while Table D5 presents
 992 the corresponding EGW distances, reflecting the regularized alignment between graphon pairs.

993 For the IMDB-Binary dataset, when transferring from Reddit-Binary, the best-matching source label
 994 for target label 1 is label 0 (GW = 0.3276, EGW = 0.3152), and for target label 2, it is also source label
 995 0 (GW = 0.4103, EGW = 0.3987). When COLLAB is used as the source, target labels 1 and 2 best
 996 match source label 1 (GW = 0.1910, EGW = 0.1784, and GW = 0.2012, EGW = 0.1895, respectively).
 997 For IMDB-Multi, transferring from Reddit-Binary, all three target labels (1, 2, and 3) have their
 998 lowest GW distances with source label 0 (GW = 0.4437, 0.4260, and 0.5172, respectively), while
 999 the EGW distances are slightly improved (EGW = 0.4321, 0.4155, and 0.5053). When transferring
 1000 from COLLAB, the best-matching source labels are more varied: target labels 1, 2, and 3 correspond
 1001 best with source label 1 (GW = 0.1751, 0.2084, and 0.1552, respectively; EGW = 0.1690, 0.1968,
 1002 and 0.1443). Finally, in the bioinformatics setting of PROTEINS-Full transferring from D&D, target

labels 1 and 2 both best align structurally with source label 1 (GW = 0.0616, EGW = 0.0592, and GW = 0.0652, EGW = 0.0624, respectively).

These results suggest clear structural correspondences between target and source labels, enabling effective and informed transfer learning across domains.

Table D3: Statistics of the datasets used in our experiments.

Property	REDDIT-B	IMDB-B	IMDB-M	COLLAB	PROTEINS-FULL	D&D
#Graphs	2,000	1,000	1,500	5,000	1,113	1,178
#Classes	2	2	3	3	2	2
Avg. #Nodes	429.63	19.77	13.00	74.49	25.22	284.32
Avg. #Edges	497.75	96.53	65.94	2457.78	226.41	715.66

Table D4: Gromov-Wasserstein distance between each source and target label pair across datasets. Bold values denote the best-matching source label for each target label.

Source → Target	IMDB-B from REDDIT-B		IMDB-B from COLLAB		IMDB-M from REDDIT-B			IMDB-M from COLLAB			PROTEINS-FULL from D&D	
Target label	1	2	1	2	1	2	3	1	2	3	1	2
Source label 0	0.3276	0.4103	0.4710	0.3996	0.4437	0.4260	0.5172	0.3521	0.3844	0.3014	0.0631	0.0622
Source label 1	0.3300	0.4143	0.1910	0.2012	0.4542	0.4604	0.5326	0.1751	0.2084	0.1552	0.0616	0.0652
Source label 2	—	—	0.5475	0.4702	—	—	—	0.4092	0.4155	0.3573	—	—

Table D5: Entropic- Gromov-Wasserstein distance with $\epsilon = 0.01$ between each source and target label pair across datasets. Bold values denote the best-matching source label for each target label.

Source → Target	IMDB-B from REDDIT-B		IMDB-B from COLLAB		IMDB-M from REDDIT-B			IMDB-M from COLLAB			PROTEINS-FULL from D&D	
Target label	0	1	0	1	0	1	2	0	1	2	0	1
Source label 0	0.3276	0.4104	0.4711	0.3996	0.4438	0.4260	0.5172	0.3523	0.3845	0.3014	0.0631	0.0584
Source label 1	0.3300	0.4142	0.1910	0.1306	0.4549	0.4360	0.5280	0.1750	0.2085	0.1552	0.0652	0.0497
Source label 2	—	—	0.5479	0.4773	—	—	—	0.4092	0.4155	0.3573	—	—

D.2 Application to Link Prediction

Existing literature applied graphon estimation to the link prediction task [61, 43], where the goal is to predict missing or future connections between nodes in a network.

Datasets. We evaluate our method on six real-world social networks that span diverse domains: dolphins, karate, football, firm, hamster, and wiki-vote. These datasets reflect a range of interaction types, including animal social behavior (e.g., dolphins), human relationships and corporate affiliations (e.g., karate, firm), and communication or membership networks (e.g., football, wiki-vote). Table D6 summarizes key statistics of the six real-world social networks. The datasets vary considerably in size and structural properties. The number of nodes ranges from 33 (firm) to 2,426 (hamster), and the number of edges spans from 78 (karate) to 16,630 (hamster). The average degree reflects network connectivity, with hamster showing the highest average degree (13.71) and karate the lowest (4.59). Network density, defined as the ratio of observed edges to the total number of possible edges, also varies widely—from sparse graphs like hamster (0.0057) and wiki-vote (0.0074) to denser graphs such as firm (0.1723). This diversity makes the collection a representative testbed for evaluating the robustness and adaptability of link prediction algorithms under different graph topologies. We select wiki-vote as the source network due to its relatively large size, which provides a more informative and stable initial graphon estimate for transfer. Link prediction is then performed on each of the remaining datasets treated as target graphs. The experiments are repeated over 50 random seeds.

Table D6: Basic statistics of the real-world datasets used in graphon estimation.

Dataset	#Nodes	#Edges	Avg. Deg.	Density
dolphins	62	159	5.13	0.0841
karate	34	78	4.59	0.1390
football	115	613	10.66	0.0935
firm	33	91	5.52	0.1723
hamster	2426	16630	13.71	0.0057
wiki-vote	889	2914	6.56	0.0074

Experimental Setup. We simulate a realistic link prediction task using a masking-based evaluation strategy following [61]. Specifically, we randomly mask a subset of edges in the upper triangular portion of the target adjacency matrix to form a test set. Let $M \in \{0, 1\}^{n \times n}$ be a masking matrix with $M_{ij} \sim \text{Bernoulli}(1 - p)$, where p is the test ratio. The observed matrix is $\mathbf{A}_{ij}^{\text{mask}} = M_{ij} \mathbf{A}_{t,ij}$, meaning each edge is observed with probability $1 - p$. We set $p = 0.1$ unless otherwise specified. Both **GTRANS-GW** and **GTRANS-EGW** are applied to \mathbf{A}^{mask} to estimate $\hat{\mathbf{P}}_t$, and each experiment is repeated 50 times with different seeds to report averaged performance.

Evaluation. To evaluate the performance of link prediction, we computed the area under the receiver operating characteristic curve (AUC). Evaluation is based on the masked (unobserved) entries where $M_{ij} = 0$, using the original adjacency \mathbf{A}_t as ground truth. For a threshold $t > 0$, the false positive rate (FPR) and true positive rate (TPR) are defined as:

$$r_{\text{FP}}(t) = \frac{\sum_{ij} \mathbf{1}(\hat{\mathbf{P}}_{t,ij} > t, \mathbf{A}_{ij}^{\text{true}} = 0, M_{ij} = 0)}{\sum_{ij} \mathbf{1}(\mathbf{A}_{t,ij} = 0, M_{ij} = 0)}$$

$$r_{\text{TP}}(t) = \frac{\sum_{ij} \mathbf{1}(\hat{\mathbf{P}}_{t,ij} > t, \mathbf{A}_{t,ij} = 1, M_{ij} = 0)}{\sum_{ij} \mathbf{1}(\mathbf{A}_{t,ij} = 1, M_{ij} = 0)}$$

These quantities are used to construct the ROC curve by varying the threshold t , and the AUC is computed as the area under this curve.

Results. Figure D5 shows the ground-truth optimal thresholds ϵ^* across all source-target pairs, selected via grid search. Based on these results, we observe that a fixed choice of $\delta^* = 0.15$ for GTRANS-GW and $\delta^* = 0.18$ for GTRANS-EGW provides near-optimal performance across most settings. We therefore adopt these values as default thresholds in our final evaluation.

Table D7 reports the AUC scores (mean \pm standard deviation) for link prediction across four baseline graphon methods: USVD, NS, ICE, and SAS. The results show that our method consistently outperforms baselines. Notably, our transfer approach achieves the highest AUC on four out of five datasets: dolphins (0.7635), firm (0.7109), football (0.8676), karate (0.8210). On the hamster dataset, neighborhood smoothing (NS) achieves the best performance (0.9513), slightly outperforming our method (0.9395), though both methods perform substantially better than the others. This can be attributed to the fact that hamster is a large-scale network, and transferring from a smaller source network (wiki-vote) may not fully capture the complexity of the target structure.

Interestingly, we observe that GTRANS-GW with $\delta = 0.15$ and GTRANS-EGW with $\epsilon = 0.01$, $\delta = 0.18$ yield *identical performance* in link prediction across all evaluated datasets. The reason lies in the nature of link prediction: it only evaluates predicted probabilities on a sparse subset of held-out

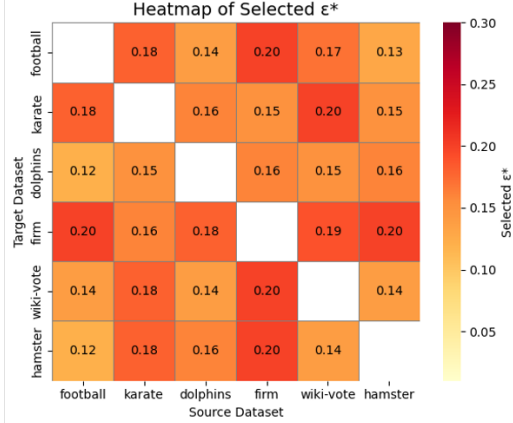


Figure D5: Optimal thresholds δ^* across dataset transfer pairs. Diagonal entries (self-pairs) are omitted.

edges, typically concentrated in high-probability regions. As a result, global structural differences in the estimated matrices are largely masked, and both methods produce similar rankings over the test set, leading to indistinguishable AUC and MSE scores.

Overall, these results highlight the effectiveness and robustness of transfer-based graphon estimation, particularly in smaller or noisier networks where leveraging external structure is beneficial.

Table D7: AUC scores (mean \pm std) across methods. Best result per dataset is **bolded**.

Dataset	USVD	NS	ICE	SAS	Transfer
dolphins	0.7235 \pm 0.1014	0.7060 \pm 0.0901	0.7536 \pm 0.0840	0.5066 \pm 0.0635	0.7635\pm0.0857
firm	0.6564 \pm 0.1213	0.6632 \pm 0.1227	0.6426 \pm 0.1220	0.5490 \pm 0.0759	0.7109\pm0.1213
football	0.8532 \pm 0.0356	0.8675 \pm 0.0349	0.8246 \pm 0.0459	0.4456 \pm 0.0738	0.8676\pm0.0372
karate	0.7186 \pm 0.1519	0.7674 \pm 0.1201	0.8043 \pm 0.1122	0.6388 \pm 0.1115	0.8210\pm0.1035
hamster	0.8264 \pm 0.0041	0.9513\pm0.0035	0.9331 \pm 0.0034	0.5108 \pm 0.0039	0.9395 \pm 0.0047

E Additional Details

E.1 Graphon functions

We implement 10 distinct graphon structures (see Table E8) ranging from simple bilinear forms to highly structured oscillatory and piecewise functions.

Table E8: Graphon functions implemented.

ID	Graphon Function
1	$\exp(-x^{0.7} - y^{0.7})$
2	$\exp(-\max(x, y)^{0.75})$
3	$\exp(-0.5 [\min(x, y) + \sqrt{x} + \sqrt{y}])$
4	$\frac{1}{1 + \exp(-[\max(x, y)^2 + \min(x, y)^4])}$
5	$ x - y $
6	$\frac{xy}{2}$
7	$\frac{x^2 + y^2}{3} \cos\left(\frac{1}{x^2 + y^2}\right) + 0.15$
8	$\frac{x + y}{3} \cos\left(\frac{1}{x + y}\right) + 0.15$
9	$\frac{\sin(10\pi(x + y - 5))}{5} + 0.5$
10	$\frac{1}{4} \min\left(\exp\left(\sin\left(\frac{6}{(1-x)^2 + y^2}\right)\right), \exp\left(\sin\left(\frac{6}{x^2 + (1-y)^2}\right)\right)\right)$

E.2 Neighborhood Smoothing Details

For effective neighborhood smoothing, we must identify a neighborhood \mathcal{N}_i for each node i that contains only nodes with approximately homogeneous distribution. Here, the neighborhood \mathcal{N}_i in this context refers to nodes that are *close in the underlying latent space* (i.e., have similar latent positions), not nodes that are connected in the observed graph. Mathematically, node i' belongs to node i 's neighborhood if their graphon values are sufficiently close: $\|f(u_i, \cdot) - f(u_{i'}, \cdot)\|_2 \leq \eta$, where $\|\cdot\|_2$ denotes the L_2 norm with inner product $\langle f, g \rangle = \int_0^1 f(x)g(x)dx$, and η is a tolerance parameter. Empirically, this can be estimated by $\sum_{j=1}^n \|\hat{p}_{ij} - \hat{p}_{i'j}\|_2^2$. As proved in [61], under mild conditions, this quantity is bounded by: $\Delta_{ij}^2 = \max_{k \neq i, i'} \frac{1}{n} |\sum_{j=1}^n (\mathbf{A}_{ij} - \mathbf{A}_{i'j}) \mathbf{A}_{kj}|$. This bound enables efficient neighborhood definition: $\mathcal{N}_i = \{i' \neq i : \Delta_{ii'} \leq \tau\}$, where τ is a predefined threshold. Once neighborhoods \mathcal{N}_i and \mathcal{N}_j are identified, nodes within each neighborhood are treated as replicates. Thus, $\hat{\mathbf{P}}_{ij}$ can be estimated by neighbourhood smoothing, i.e., $\hat{\mathbf{P}}_{ij} = \frac{1}{2} \left(\frac{\sum_{i' \in \mathcal{N}_i} \mathbf{A}_{i'j}}{|\mathcal{N}_i|} + \frac{\sum_{j' \in \mathcal{N}_j} \mathbf{A}_{ij'}}{|\mathcal{N}_j|} \right)$, where $|\mathcal{N}_i|$ represents the number of nodes in the neighborhood \mathcal{N}_i . This averages two empirical proportions: edges between node j and nodes in \mathcal{N}_i , and edges between node i and nodes in \mathcal{N}_j .

1082 E.3 Algorithm

1083 E.3.1 Algorithm: GTRANS

Algorithm 1 GTRANS: Transfer Learning for Graphon Estimation

Require: Source adjacency matrix \mathbf{A}_s , target adjacency matrix \mathbf{A}_t , a threshold ϵ .

Ensure: Target graphon estimator $\hat{\mathbf{P}}_t$

- 1: **Step 1: Initial Graphon Estimation**
 - 2: Apply neighborhood smoothing to obtain two initial estimators for both the source and target graph: $\hat{\mathbf{P}}_s^{ini} \leftarrow NS(\mathbf{A}_s)$, $\hat{\mathbf{P}}_t^{ini} \leftarrow NS(\mathbf{A}_t)$.
 - 3: **Step 2: Transferring Step**
 - 4: Compute the optimal transport plan $\hat{\pi}$ and the optimal transportation distance d between $\hat{\mathbf{P}}_s^{ini}$ and $\hat{\mathbf{P}}_t^{ini}$.
 - 5: Apply column normalization to $\hat{\pi}$, obtaining $\tilde{\pi}$
 - 6: Transfer source graphon estimator to target domain: $\hat{\mathbf{P}}_t^{trans} \leftarrow \tilde{\pi}^T \hat{\mathbf{P}}_s^{ini} \tilde{\pi}$.
 - 7: Refine the transferred estimator by applying neighborhood smoothing to $\hat{\mathbf{P}}_t^{trans}$: $\hat{\mathbf{P}}_t^{trans2} \leftarrow NS(\hat{\mathbf{P}}_t^{trans})$.
 - 8: **if** $d > \delta$, **then**
 - 9: **Step 3: Debiasing Step:**
 - 10: Compute residual matrix: $\mathbf{R}_t \leftarrow \hat{\mathbf{P}}_t^{ini} - \hat{\mathbf{P}}_t^{trans2}$
 - 11: Apply neighborhood smoothing to residual, obtaining estimator $\hat{\mathbf{P}}_t^{res}$.
 - 12: Combine transferred estimator with smoothed residual: $\hat{\mathbf{P}}_t \leftarrow \hat{\mathbf{P}}_t^{trans2} + \hat{\mathbf{P}}_t^{res}$
 - 13: **else**
 - 14: $\hat{\mathbf{P}}_t \leftarrow \hat{\mathbf{P}}_t^{trans2}$
 - 15: **end if**
 - 16: **return** $\hat{\mathbf{P}}_t$
-

1084 E.3.2 Algorithm: Network Cross-Validation by Edge Sampling

Algorithm 2 Network Cross-Validation by Edge Sampling for Threshold Selection

Require: Adjacency matrices $\mathbf{A}_s, \mathbf{A}_t$, true probability matrix \mathbf{P}_t , threshold candidates $\{\epsilon_1, \dots, \epsilon_L\}$, number of folds K , loss function $\mathcal{L}(\cdot, \cdot)$

Ensure: Selected threshold $\hat{\epsilon}$, final loss, estimated graphon $\hat{\mathbf{P}}_t$

- 1: **Step 1: Edge Sampling**
 - Randomly partition the edges of \mathbf{A}_t into K folds, ensuring that each fold is disjoint and collectively covers the entire set of edges.
 - 2: **for** $k = 1$ to K **do**
 - 3: Mask edges in the k -th fold to form the incomplete adjacency matrix $\mathbf{A}_t^{(k)}$.
 - 4: Perform matrix completion as suggested by [32], leveraging low-rank and smoothness assumptions to recover the completed adjacency matrix.
 - 5: Apply GTRANS with each threshold ϵ_l and evaluate the loss $\mathcal{L}^{(k)}(\epsilon_l)$ on the held-out set $\Omega_c^{(k)}$
 - 6: **end for**
 - 7: **Step 2: Select Optimal Threshold**
 - 8: Average the loss across all folds: $\bar{\mathcal{L}}(\epsilon_l) = \frac{1}{K} \sum_{k=1}^K \mathcal{L}^{(k)}(\epsilon_l)$
 - 9: Select the threshold that minimizes the average loss: $\hat{\epsilon} = \arg \min_{\epsilon_l} \bar{\mathcal{L}}(\epsilon_l)$
 - 10: **Return** $\hat{\epsilon}$.
-

1085 E.4 Graph Augmentation Model Training Setup.

1086 We follow a modified version of the training configuration from [21]. Specifically, we train a GIN
 1087 model for 200 epochs using the Adam optimizer with a fixed learning rate of 0.01. The mini-batch
 1088 size is set to 128, and the hidden dimension is 64. Validation loss is monitored throughout training,
 1089 and test accuracy is reported at the epoch with the best validation performance.