Statistica Sinica Preprint No: SS-2025-0266	
Title	Network Model Averaging Prediction for Latent Space Models by K-fold Edge Cross-Validation
Manuscript ID	SS-2025-0266
URL	http://www.stat.sinica.edu.tw/statistica/
DOI	10.5705/ss.202025.0266
Complete List of Authors	Yan Zhang, Jun Liao, Xinyan Fan, Kuangnan Fang and Yuhong Yang
Corresponding Authors	Kuangnan Fang
E-mails	xmufkn@xmu.edu.cn
Notice: Accepted author version.	

NETWORK MODEL AVERAGING PREDICTION FOR LATENT SPACE MODELS BY K-FOLD EDGE CROSS-VALIDATION

Yan Zhang[1*], Jun Liao[2*], Xinyan Fan[2], Kuangnan Fang[3†], Yuhong Yang[4]

[1]*Shanghai University of International Business and Economics*

[2]*Renmin University of China,* [3]*Xiamen University,* [4]*Tsinghua University*

Abstract: In complex systems, networks describe relationships between nodes through edges. Latent space models are widely used for network tasks such as community detection and link prediction due to their interpretability and visualization power. However, when the network size is small or the true latent dimension is large, a single latent space model may suffer from high estimation error or model misspecification. To address this, we propose Network Model Averaging (NetMA), which combines multiple latent space models with different dimensions. The weights are estimated using a K-fold edge cross-validation scheme that is specially designed for network data. Our method applies to both single-layer and multi-layer networks. We provide theoretical guarantees for NetMA. When all candidate models are misspecified, NetMA still achieves asymptotically optimal prediction. When models with large enough latent dimensions are included, NetMA assigns nearly all weights to them. We also prove that the estimated weights converge to the optimal weights. Simulation studies show that NetMA performs better than model selection and simple averaging. It even outperforms the "oracle" model when the true latent dimension is large. Applications to mutual-following and virtual event networks further highlight the strong performance of NetMA in link prediction.

Key words and phrases: Asymptotic optimality, Consistency, Edge cross-validation, Network model

*Yan Zhang and Jun Liao are co-first authors and contributed equally to this work.

†Corresponding author. E-mail address: xmufkn@xmu.edu.cn.

1. Introduction

Networks are a powerful tool for representing complex systems, where entities (nodes) are connected by relationships (edges). Unlike traditional data, network data record both node-level attributes and interactions between nodes, sometimes supplemented with pairwise features (Ma et al., 2020). In today's interconnected world, network data arise in diverse domains, such as social sciences (Serrat, 2017), international trade (Dong et al., 2021), epidemiology (Jo et al., 2021), and fraud detection (Óskarsdóttir et al., 2022). A fundamental task in network analysis is link prediction, which estimates the probability that a connection exists or will form between two nodes. Missing links are common due to factors such as unclear boundary specification, reporting errors, survey non-response (Kossinets, 2006), or resource limitations in experimental settings (Li et al., 2016). In dynamic networks, link prediction also supports forecasting future relationships (Song et al., 2022).

A widely used approach for link prediction is to fit a probabilistic model to the observed network. Existing models include random graph models (Erdös and Rényi, 1959), the p_1 model (Holland and Leinhardt, 1981), stochastic block models (SBMs) (Holland et al., 1983), latent space models (LSMs) (Hoff et al., 2002), and many variants of them (Karrer and Newman, 2011; Sewell and Chen, 2015). Among these models, LSMs are particularly appealing due to their interpretability and their ability to capture key structural properties such as transitivity, homophily, and community structure (Zhang et al., 2022). Mechanistically, these models represent each node as a vector in a low-dimensional latent space, where the probability of a connection between two nodes depends on the distance between

their latent positions. While LSMs are traditionally recognized for inferring latent node positions to facilitate visualization, their utility also extends to link prediction. The fundamental ability of these models to capture the underlying geometric structure renders them naturally robust candidates for inferring missing edges. This perspective is supported by literature demonstrating that latent factor-based approaches often outperform heuristic or neighborhood-based methods in prediction tasks. For instance, Koren et al. (2009) showed in recommender systems that matrix factorization techniques work better than standard neighbor-based methods because they effectively capture global patterns. Menon and Elkan (2011) applied this to general graphs, showing that learning latent features handles imbalanced data better than simple topological indices. Furthermore, Zhu et al. (2016) proposed a temporal latent space model for dynamic social networks. They found that modeling how node positions change over time leads to better prediction accuracy than other baseline methods. More recently, Pan et al. (2022) developed a latent space logistic regression framework that explicitly captures reciprocity and transitivity for link prediction, demonstrating superior performance in sparse social networks compared to standard baselines. Pan et al. (2026) extended LSMs to citation networks by integrating author-paper bipartite information, showing that modeling the joint latent geometry can enhance link prediction accuracy in scientific networks.

A key modeling choice in LSMs is the dimension of the latent space. In practice, two-dimensional embeddings are often used to facilitate visualization (Sosa and Buitrago, 2021; Tang and Zhu, 2025). However, this simplification may fail to capture essential structural information. Several methods have been proposed for dimension selection, including shrinkage priors (Durante and Dunson, 2014; Gwee et al., 2025), Bayesian criteria (Oh and Raftery,

2001, 2007), and cross-validation techniques (Hoff, 2007; Li et al., 2020).

Nonetheless, when the true latent dimension is relatively high and the network size is small, model selection becomes unreliable. In such cases, the resulting LSM can suffer from either large modeling bias (due to selecting a smaller latent dimension) or high estimation variance, both leading to poor predictive accuracy. Model averaging offers a principled way to address this issue by combining multiple candidate models to stabilize estimation and improve prediction. There are two main types of model averaging methods, i.e., Bayesian model averaging and frequentist model averaging. Bayesian model averaging has been studied for a long time, both in statistics and economics (Hoeting et al., 1999; Fragoso et al., 2018). In recent years, there has been a rapid development in frequentist model averaging (Yang, 2001; Hansen, 2007; Zhang and Liang, 2011; Liao and Zou, 2020; Liao et al., 2021). In terms of optimal model averaging, Hansen and Racine (2012) introduced a jackknife model averaging (JMA), which determines model averaging weights by minimizing leave-one-out cross-validation for independent data, which is extended to dependent data by Zhang et al. (2013). Gao et al. (2016) developed a model averaging method based on the leave-subject-out cross-validation for longitudinal data. Additionally, Zhang and Liu (2023) proposed an averaging prediction which determines the weights through the K-fold cross-validation.

Existing model averaging approaches are mainly designed for structured data, while network data are typically unstructured. Unstructured data have no particular format, schema or structure, such as text, images and networks (Tanwar et al., 2015). Network data, in particular, present unique challenges such as small sample size, growing parameter space and sparsity, which complicate both methodological design and theoretical analysis. A related network mixing strategy was proposed by Li and Le (2024), which determined nonnegative

weights through a single dyad split. While the Exponential Weighting method in Li and Le (2024) satisfies the unit-sum constraint, their Non-Negative Linear mixing method does not. In addition, their methods are restricted to single-layer networks. A very recent work by Qiu and Zhang (2025) considered link prediction under a transfer learning framework using model averaging. Their work focuses on privacy-preserving knowledge transfer between auxiliary and target layers. In contrast, our proposed NetMA method targets the fundamental challenge of dimension uncertainty and estimation stability in both single-layer and multi-layer networks, assuming full access to the network data. We focus on the optimal model averaging approach for link prediction in single-layer and multi-layer networks, which has not been explored before, to our knowledge. The weights are obtained through the K-fold edge cross-validation rather than a single random split. Compared to the typical K-fold cross-validation designed for model averaging (e.g., Zhang and Liu (2023)), the K-fold edge cross-validation here is more adaptive to the network data structure. On the theoretical front, we show that when the candidate models are all misspecified, NetMA exhibits an asymptotic optimality, and when the candidate set includes models with large enough latent dimensions, it asymptotically assigns all weights to the correct models. Additionally, NetMA weights converge to the optimal ones. Extensive simulations and empirical applications are conducted to show the advantages of our method on both single-layer and multi-layer networks.

2. Model Averaging Prediction for Single-Layer Networks

2.1 Model framework

Suppose that we have an undirected and unweighted single-layer network, which can be represented by a binary adjacency matrix $A \in \{0,1\}^{N\times N}$, where $A_{ij} = A_{ji} = 1$ if node i and

node j are connected, and $A_{ij} = A_{ji} = 0$ otherwise. The diagonal elements of A are set to be 0, i.e., $A_{ii} = 0$. We assume that the connectivity between each pair of nodes i and j is conditionally independent Bernoulli random variables, with $E(A) = P$ given certain latent variables and heterogeneity parameters. The primary goal of network analysis is to estimate P from A. However, unlike many types of structured data, we typically observe only a single realization of A. To address this difficulty, it is necessary to impose additional structural assumptions. In this work, we consider the following LSM proposed by Hoff et al. (2002). Specifically, for any $i, j = 1, \ldots, N$ and $i < j$, we have

$$\begin{aligned} A_{ij} &\overset{\text{ind.}}{\sim} \text{Bernoulli}\left(P_{ij}\right), \quad \text{with} \\ \text{logit}(P_{ij}) = \Theta_{ij} &= \alpha_i + \alpha_j + z_i^\top z_j, \end{aligned} \tag{2.1}$$

where $\alpha_i \in \mathbb{R}$ reflects the popularity of node i, $z_i \in \mathbb{R}^d$ denotes the latent vector of node i with $1 \leq d < N$, and $\text{logit}(x) = \log\{x/(1-x)\}$ for any $x \in (0, 1)$. In matrix form, we have $\Theta = \alpha 1_N^\top + 1_N \alpha^\top + ZZ^\top$, where $\alpha = (\alpha_1, \ldots, \alpha_N)^\top$, 1_N is the all one vector in $\mathbb{R}^N$ and $Z = (z_1, \ldots, z_N)^\top \in \mathbb{R}^{N \times d}$. To ensure the identifiability of parameters in Model (2.1), we assume the latent variables are centred, that is $JZ = Z$, where $J = I_N - 1_N 1_N^\top / N$. This constraint makes Z identifiable up to an orthogonal transformation of its rows.

Remark 1. Model (2.1) is a basic and classic form of the LSM. We use this model because we are mainly interested in the weighted average effect of latent spaces of different dimensions. Regarding the functional forms of $\ell(z_i, z_j)$, here we consider the inner product form, i.e., $\ell(z_i, z_j) = z_i^\top z_j$. Apart from this, Hoff et al. (2002) mentioned the projection form $\ell(z_i, z_j) = z_i^\top z_j / \|z_j\|$ and the distance form $\ell(z_i, z_j) = \|z_i - z_j\|$. Both forms are also suitable for our framework. Since most papers on LSMs currently use the inner product form (Zhang et al.,

2022; Tang and Zhu, 2025; Li et al., 2023), we follow this choice as well. Note that the form of ℓ is not essential, and the theoretical results hold as long as Assumption 1 below is satisfied.

Remark 2. A common feature in real-world networks is the phenomenon of homophily, which means that nodes sharing common characteristics are more likely to connect with each other (Kossinets and Watts, 2009). For example, social scientists have found that school children tend to form friendships and playgroups when they have similar demographic characteristics (McPherson et al., 2001). Therefore, we can incorporate node or edge-specific covariates which represent the similarity between nodes to reflect homophily. Specifically, we introduce the following model which is proposed by Ma et al. (2020). Assume that for any $i < j$,

$$\text{logit}(P_{ij}) = \alpha_i + \alpha_j + \beta X_{ij} + z_i^\top z_j, \tag{2.2}$$

where X_{ij} denotes the covariate of the edge between node i and node j. We further require that $X_{ij} = X_{ji}$ to ensure symmetry, and $X_{ii} = 0$ to avoid self-loops in the model. The value of X_{ij} can be either binary, indicating whether nodes i and j share a common attribute (e.g., gender, country), or continuous, representing a distance or similarity measure (e.g., differences in age or similarities in hobbies). The estimation of this model has been studied in detail in Ma et al. (2020).

Certainly, if we include covariates in our model, we would need additional assumptions, such as Assumption 4 regarding the stable rank of the covariate matrix in Ma et al. (2020). However, this is not fundamentally different from our model. The properties of the estimator obtained from the true candidate model have been extensively studied in the context of different forms of LSMs (Ma et al., 2020; Zhang et al., 2022). We conduct a simulation

study on the case with covariates, which can be found in Section 4.1.

As the network size increases, the number of parameters to be estimated also increases. Due to the large number of parameters to be estimated in Model (2.1), the estimation problem is particularly challenging. In recent years, researchers have proposed estimation methods that can be divided into two categories: one is the Bayesian approach via Markov chain Monte Carlo and the other is the maximum likelihood estimation method (Kim et al., 2018). In the former category, the parameters are treated as random effects (Hoff, 2007; Handcock et al., 2007), and specific assumptions on the distribution of parameters are required. In the latter category, the parameters are treated as fixed effects. Ma et al. (2020) was the first to adopt this idea and proposed an efficient projected gradient descent (PGD) algorithm for estimating the single-layer inner-product LSM. Due to the absence of distributional assumptions and the efficiency and scalability of PGD, many researchers have utilized the algorithm to estimate the LSM (Zhang et al., 2020, 2022; Lyu et al., 2023). In this paper, we treat the parameters as fixed effects and adopt the PGD algorithm to estimate Model (2.1).

We estimate the parameters α and Z by minimizing the following conditional negative log-likelihood:

$$\mathcal{L}(\alpha, Z) = -\sum_{i,j} \log P(A_{ij} \mid \alpha, Z) = -\sum_{i,j} \{A_{ij}\Theta_{ij} - f(\Theta_{ij})\},$$

where $f(x) = \log(1 + \exp(x))$. Then we can adopt the PGD algorithm to obtain the estimators of the parameters.

2.2 Model averaging criterion

The motivation for considering multiple models stems from the intrinsic complexity of real-world networks. In practice, the connectivity patterns between nodes are often driven by diverse latent factors (e.g., social circles, geographic locations, and shared interests) that may not be adequately captured by a single rigid dimensional space. A single low-dimensional model tends to miss fine details by focusing only on the global backbone, while a single high-dimensional model is prone to high variance. Consequently, selecting a single model is challenging, particularly when the true latent dimension is large and the network size is small. As clearly demonstrated in Case 1 of our simulation, model estimation performance remains suboptimal even when the true latent dimension is explicitly provided. Under these circumstances, a single LSM is prone to severe bias due to under fitting or high estimation variance, making it risky to rely on one selected model. Therefore, it is desirable to consider multiple candidate models, each with a different latent dimension, and combine them. By assigning optimal weights to candidate models, we can fully leverage the strengths of multiple models to achieve superior performance. This helps us better capture the structure of the network and reduces the risks of picking a wrong or unstable single model. Suppose that we have M candidate models, where the dimension of the latent vectors in the mth ($m = 1, \ldots, M$) candidate model is m. Specifically, the mth candidate model is $\Theta = \alpha 1_N^\top + 1_N \alpha^\top + Z_{(m)} Z_{(m)}^\top$, where $Z_{(m)} \in \mathbb{R}^{N \times m}$. In many practical scenarios, the network may not be fully observed. Let $\Psi = \{(i, j) : \ i, \ j = 1, \ldots, N\}$ be the set of all nodal pairs. We assume that the edge information is available only for a subset of pairs, denoted as $\Psi_1 \subset \Psi$, while the edge information for the remaining pairs, denoted as $\Psi_2 = \Psi \setminus \Psi_1$, is missing. The missing pattern is assumed to be symmetric, meaning that if $(i, j) \in \Psi_2$,

then $(j,i) \in \Psi_2$. Our goal is to predict the connection probability on the missing set Ψ_2 using the information from the observed set Ψ_1. To facilitate estimation, we assume that the missing data mechanism is missing completely at random (MCAR), following the practice of Mariadassou and Tabouy (2020). This means that Ψ_1 can be regarded as a uniform random sample from Ψ. Let $p = |\Psi_1|/|\Psi|$ be the proportion of observed edges. To estimate the parameters for partially observed networks, we adopt the zero-filling strategy established by Chatterjee (2012) and Gao and Ma (2020). Specifically, we define the zero-filled observed matrix $A(\Psi_1)$ by setting entries corresponding to missing edges in Ψ_2 to 0. We utilize the negative log-likelihood of $A(\Psi_1)$ as a surrogate objective function. This approach is justified because, under the MCAR assumption, the zero-filled matrix satisfies $E[A(\Psi_1)] = pP$ (Li and Le, 2024). Computationally, this allows us to treat the problem as a full-matrix estimation task using the efficient PGD algorithm. By applying PGD to $A(\Psi_1)$, we obtain the estimators of α and $Z_{(m)}$, denoted as $\widehat{\alpha}_{(m)} = (\widehat{\alpha}_{(m),1}, \ldots, \widehat{\alpha}_{(m),N})^\top \in \mathbb{R}^N$ and $\widehat{Z}_{(m)} = (\widehat{z}_{(m),1}, \ldots, \widehat{z}_{(m),N})^\top \in \mathbb{R}^{N\times m}$. According to Model (2.1), these yield an intermediate estimator $\widehat{P}^0_{(m)}$ for the probability matrix. Since $\widehat{P}^0_{(m)}$ targets pP and is thus biased, we explicitly correct the scaling bias by setting $\widehat{P}_{(m)} = \widehat{P}^0_{(m)}/p$. We further constrain the estimators to fall between 0 and 1.

Let $w = (w_1, \ldots, w_M)^\top$ be the weight vector with $w_m \geq 0$ $(m = 1, \ldots, M)$ and $\sum_{m=1}^M w_m = 1$. The averaging prediction for P_{ij}, $(i,j) \in \Psi_2$ is $\widehat{P}_{ij}(w) = \sum_{m=1}^M w_m \widehat{P}_{(m),ij}$, where $\widehat{P}_{(m),ij}$ is the estimated connection probability between nodes i and j for the mth candidate model.

To select model weights, our objective is to minimize the squared error, defined as $L(w) = \sum_{(i,j)\in\Psi_2}\{\widehat{P}_{ij}(w) - P_{ij}\}^2$ for the single layer network. However, the optimization of $L(w)$ depends on the true probability matrix, which is impractical to obtain. Therefore, rather

than directly minimizing $L(w)$, we resort to selecting data-driven weights by the K-fold cross-validation criterion. However, it is obvious that nodes in the network are connected by edges, and partitioning nodes would require removing edges, destroying the network structure. Therefore, the traditional K-fold cross-validation used for model averaging is no longer applicable. Here, we propose a K-fold edge cross-validation criterion for networks to select the model weights for link prediction. The key idea for the K-fold edge cross-validation is to split the nodal pairs into K groups and treat each group as a testing set to evaluate the model performance. Then we describe the calculation of the K-fold edge cross-validation criterion and how to conduct link prediction with data-driven weights in detail. The proposed method proceeds as follows.

Step 1: Divide the nodal pairs in Ψ_1 into K groups equally. Let $G_k,\ k = 1, \ldots, K$, denote the set of the nodal pairs in the kth group.

Step 2: For $k = 1, \ldots, K$,

(a) Exclude the nodal pairs in the kth group from Ψ_1 and use the remaining nodal pairs in Ψ_1 to calculate the estimators of $Z_{(m)}$ and α in the mth model ($m = 1, \ldots, M$), which are $\widetilde{Z}_{(m)}^{[-k]}$ and $\widetilde{\alpha}_{(m)}^{[-k]}$, respectively.

(b) Calculate the predictions for observations within the kth group for each model. That is, we calculate the prediction of $P \circ S^{[k]}$ for the mth model by $\widetilde{P}_{(m)}^{[k]} = f_1\left(\widetilde{\alpha}_{(m)}^{[-k]} 1_N^\top + 1_N \widetilde{\alpha}_{(m)}^{[-k]\top} + \widetilde{Z}_{(m)}^{[-k]} \widetilde{Z}_{(m)}^{[-k]\top}\right) \circ S^{[k]}$, where $\circ$ denotes the Hadamard product of two matrics, $f_1(x) = K/\{(1+\exp(-x))(K-1)\}$, and $S^{[k]} = \left(S_{ij}^{[k]}\right) = \left(\mathbb{1}_{(i,j)\in G_k}\right) \in \mathbb{R}^{N\times N}$, where $\mathbb{1}_{(i,j)\in G_k}$ is an indicator function that equals 1 if the pair (i, j) is in the set G_k and 0 otherwise.

Step 3: Construct the K-fold edge cross-validation criterion $CV(w) = |\Psi_1|^{-1}\sum_{k=1}^{K}\left\|A^{[k]} - \widetilde{P}^{[k]}(w)\right\|_F^2$, where $A^{[k]} = A \circ S^{[k]}$ and $\widetilde{P}^{[k]}(w) = \sum_{m=1}^{M} w_m \widetilde{P}^{[k]}_{(m)}$.

Step 4: Select the model weights by minimizing the K-fold edge cross-validation criterion, i.e., $\widehat{w} = \text{argmin}_{w\in\mathcal{W}} CV(w)$, with $w = (w_1, \ldots, w_M)^\top$ being a weight vector in the unit simplex in $\mathbb{R}^M$, i.e., $\mathcal{W} = \left\{w \in [0,1]^M : \sum_{m=1}^{M} w_m = 1\right\}$. Thus, we can construct an averaging prediction for P_{ij}, $(i,j) \in \Psi_2$ through $\widehat{P}_{ij}(\widehat{w}) = \sum_{m=1}^{M} \widehat{w}_m \widehat{P}_{(m),ij}$.

Minimizing $CV(w)$ can be transformed into a quadratic programming problem about w. Specifically, let $h_{ijk} := (\widetilde{P}^{[k]}_{(1),ij}, \ldots, \widetilde{P}^{[k]}_{(M),ij})^\top$, $H := \sum_{k=1}^{K}\sum_{i,j} h_{ijk}h_{ijk}^\top$, and $h := 2\sum_{k=1}^{K}(\langle A^{[k]}, \widetilde{P}^{[k]}_{(1)}\rangle, \ldots, \langle A^{[k]}, \widetilde{P}^{[k]}_{(M)}\rangle)^\top$, where $\langle\cdot,\cdot\rangle$ denotes the sum of all elements in the Hadamard product of two matrices. Then we have $\min_{w\in\mathcal{W}} CV(w) \Leftrightarrow \min_{w\in\mathcal{W}}\left(w^\top H w - h^\top w\right)$, which is a quadratic function of w. Therefore, we can use quadratic programming to solve the K-fold edge cross-validation weights.

2.3 Theoretical property

In this section, we first present an analysis of the asymptotic optimality of the proposed averaging prediction when the candidate models are all misspecified. When the candidate set includes models with large enough latent dimensions, as will be shown later, the proposed method assigns all weights to these models. Then we provide the convergence rate of the weight estimator towards the infeasible optimal weight vector. For the single layer network, Theorem 1 shows that the empirical K-fold edge cross-validation weights asymptotically minimize the loss function $L(w)$. The assumptions required for Theorem 1 are discussed as follows. All limiting processes in this section are with respect to $|\Psi_1| \to \infty$.

Assumption 1. Suppose that $M \leq N$. For $(i,j) \in \Psi_1$, there exists a limiting value $P^*_{(m),ij}$ for $\widehat{P}_{(m),ij}$ such that $\sum_{(i,j)\in\Psi_1}(\widehat{P}_{(m),ij} - P^*_{(m),ij})^2 = O_p(NM)$ uniformly for $m = 1,\ldots,M$.

Assumption 1 guarantees that the estimator of $P_{(m),ij}$ in each candidate model has a limit $P^*_{(m),ij}$. Here, $P^*_{(m),ij}$ could be regarded as a pseudo-true value, which is not necessarily equal to the true value. Notice that $\widehat{P}_{(m),ij}$ and $\widetilde{P}^{[-k]}_{(m),ij}$ have the same limiting values $P^*_{(m),ij}$ because $|\Psi_1|$ and $|\Psi_1| - |\Psi_1|/K$ have the same order for any $K \in \{2,\ldots,|\Psi_1|\}$. The existing literature on the LSM (Zhang et al., 2022; Ma et al., 2020) showed that the PGD estimation error satisfies $\|\widehat{\Theta}_{(m)} - \Theta\|_F^2 = O_p(Nm)$ with probability at least $1 - N^{-C}$ for some constant $C > 0$. A union bound over $m = 1,\ldots,M$ then yields $\max_{1\leq m\leq M}\sum_{(i,j)\in\Psi_1}(\widehat{P}_{(m),ij} - P^*_{(m),ij})^2 = O_p(NM)$ with probability at least $1 - MN^{-C}$, suggesting that Assumption 1 is reasonable for fixed M, and also for diverging M as long as $M = o(N^C)$ for some constant $C > 0$. A more detailed discussion is provided in Section S3.1 of the Supplementary Material.

Next, we introduce some notations associated with the limiting value $P^*_{(m),ij}$. The averaging prediction based on the limiting value is $P^*_{ij}(w) = \sum_{m=1}^M w_m P^*_{(m),ij}$. Similarly, the loss function based on the limiting value is defined as $L^*(w) = \sum_{(i,j)\in\Psi_1}\{P^*_{ij}(w) - P_{ij}\}^2$. The minimum loss in the class of averaging estimators based on the limiting value is $\xi^* = \inf_{w\in\mathcal{W}} L^*(w)$. The following assumption is about an upper bound on the expected nodal degree D, which is defined to satisfy $D = N\max_{ij} P_{ij}$.

Assumption 2. $NM\xi^{*-1} = o(1)$ and $N\max\{D,\log N\}\xi^{*-1} = O(1)$.

Assumption 2 constraints that ξ^* grows faster than NM, and $N\max\{D,\log N\}\xi^{*-1}$ is finite. The former is similar to Assumption A3 of Ando and Li (2017) and Assumption 5 of Zhang and Liu (2023). This assumption rules out the scenario where a candidate model has a large enough latent dimension, in the sense that its dimension of the latent vectors is greater

than or equal to the true dimension. To better understand its implications, we note that this condition essentially requires the dimensions omitted by the candidate models to contain significant signals. We acknowledge that this assumption may have limitations in transitional regimes. For instance, if the candidate dimension is close to the true dimension or if the omitted latent factors are weak, the resulting risk ξ^* might be smaller than NM. In such cases, the system effectively shifts from a misspecified regime toward a regime where the model has large enough latent dimensions. The theoretical properties for the case where candidate models include models with large enough latent dimensions will be discussed later.

We now justify that the weights selected by the K-fold edge cross-validation criterion are asymptotically optimal. In other words, the K-fold edge cross-validation weights asymptotically minimize the prediction loss.

Theorem 1. *Under Assumptions 1 and 2, we have* $\frac{L(\widehat{w})}{\inf_{w\in\mathcal{W}} L(w)} \to 1$ *in probability.*

The optimality statement in Theorem 1 is an important property of the model averaging estimator - that the model averaging estimator based on the K-fold edge cross-validation asymptotically achieves the lowest squared loss where w is chosen in $\mathcal{W}$. The proof of Theorem 1 is presented in Section S2.1 of the Supplementary Material.

Next, we discuss the situation where the candidate set includes models with large enough latent dimensions. Specifically, denote $\mathcal{T}$ to be the subset of $\{1,\ldots,M\}$ containing the indices of models with large enough latent dimensions. For example, when the true dimension is d_0 and $d_0 \leq M$, then $\mathcal{T} = \{d_0,\ldots,M\}$. Let $\widehat{\zeta} = \sum_{m\in\mathcal{T}} \widehat{w}_m$ be the sum of K-fold edge cross-validation weights assigned to these models. Denote $\mathcal{W}^s = \{w \in \mathcal{W} : \sum_{m\notin\mathcal{T}} w_m = 1\}$ to be the subset of $\mathcal{W}$ which assigns all weights to the models with lower latent dimensions. We need the following assumption to consider the case where the candidate set includes

models with large enough latent dimensions.

Assumption 3. $NM\left\{\inf_{w\in\mathcal{W}^s} L^*(w)\right\}^{-1} = o(1)$ and $N\max\{D,\log N\}\left\{\inf_{w\in\mathcal{W}^s} L^*(w)\right\}^{-1} = O(1)$.

Assumption 3 gives the restriction on the growth rate of the minimum loss of the averaging estimator over all misspecified models. It is easy to see that Assumption 3 is equivalent to Assumption 2 when all candidate models are misspecified. Next, we show that $\widehat{\zeta}\to 1$ in probability under some regularity conditions.

Theorem 2. *Under Assumptions 1 and 3, if $\mathcal{T}$ is not empty, we have $\widehat{\zeta}\to 1$ in probability.*

Theorem 2 implies that when the candidate set includes models with large enough latent dimensions, the proposed method asymptotically assigns all weights to these models. The proof of Theorem 2 can be found in Section S2.2 of the Supplementary Material.

Next, we present the convergence rate of the K-fold edge cross-validation-based weights. We first introduce some notations. Define the squared risk function as $R(w)=\sum_{(i,j)\in\Psi_1} E(\widehat{P}_{ij}(w)-P_{ij})^2$. We note that the integrability condition for the risk function is naturally satisfied in our framework. Specifically, since both the estimated probabilities and the true Bernoulli parameters are strictly bounded within the interval from zero to one, their squared difference lies within the unit interval. Consequently, the expectation is well-defined without requiring additional integrability assumptions. Let $\xi=\inf_{w\in\mathcal{W}} R(w)$, and denote the optimal weight vector $w^0=\text{argmin}_{w\in\mathcal{W}}R(w)$. Let $\lambda_{\min}(B)$ and $\lambda_{\max}(B)$ be the minimum and maximum singular values of a general real matrix B, respectively. Arrange $\{A_{ij},\ (i,j)\in\Psi_1\}$, $\{P_{ij},\ (i,j)\in\Psi_1\}$, $\{\widetilde{P}_{ij}(w)=\sum_{k=1}^K \widetilde{P}_{ij}^{[k]}(w),\ (i,j)\in\Psi_1\}$, $\{\widetilde{P}_{(m),ij}=\sum_{k=1}^K \widetilde{P}_{(m),ij}^{[k]},\ (i,j)\in\Psi_1\}$, $\{\widehat{P}_{ij}(w),\ (i,j)\in\Psi_1\}$, $\{\widehat{P}_{(m),ij},\ (i,j)\in\Psi_1\}$, $\{P_{ij}^*(w),\ (i,j)\in\Psi_1\}$

and $\{P^*_{(m),ij},\ (i,j) \in \Psi_1\}$ in a same particular order, and denote them as vector $a_1 \in \mathbb{R}^{|\Psi_1|}$, $p_1 \in \mathbb{R}^{|\Psi_1|}$, $\widetilde{p}_1(w) \in \mathbb{R}^{|\Psi_1|}$, $\widetilde{p}_{1(m)} \in \mathbb{R}^{|\Psi_1|}$, $\widehat{p}_1(w) \in \mathbb{R}^{|\Psi_1|}$, $\widehat{p}_{1(m)} \in \mathbb{R}^{|\Psi_1|}$, $p^*_1(w) \in \mathbb{R}^{|\Psi_1|}$ and $p^*_{1(m)} \in \mathbb{R}^{|\Psi_1|}$, respectively. Denote $\Lambda_1 = (\widehat{p}_{1(1)}, \ldots, \widehat{p}_{1(M)})$, $\Omega_1 = (p_1 - \widehat{p}_{1(1)}, \ldots, p_1 - \widehat{p}_{1(M)})$, $\Lambda = \Lambda_1^\top \Lambda_1$ and $\Omega = \Omega_1^\top \Omega_1$. Theorem 3 shows the rate of $\widehat{w}$ toward the infeasible optimal weight vector w^0. The following assumptions are needed to show this theorem.

Assumption 4. There are two positive constants ρ_1 and ρ_2, such that $0 < \rho_1 < \lambda_{\min}(\Lambda/|\Psi_1|) \leq \lambda_{\max}(\Lambda/|\Psi_1|) \leq M$, in probability tending to 1.

Assumption 5. $\lambda_{\max}(\Omega/|\Psi_1|) = O_p(M)$.

Assumption 6. $N^{1-4\kappa} \max\{D, \log N\} M\xi^{-1} = o(1)$, $N^{1-4\kappa} M^2 \xi^{-1} = o(1)$, and $M = o(N^{\min\{4\kappa, 1/3\}})$ where $\kappa \in (0, 1/2)$.

Similar to the Condition (C.4) in Liao and Zou (2020), Assumption 4 puts a constraint on the singular values of $\Lambda/|\Psi_1|$. It requires that the minimum singular value is bounded away from zero by a fixed constant ρ_1, while the maximum eigenvalue is bounded by M with probability tending to 1. Assumption 5 requires that the maximum singular value of $\Omega/|\Psi_1|$ is bounded by M in probability. Assumption 6 further restricts the relationship between M, N and ξ. The third part of Assumption 6 indicates that the number of candidate models can increase with N but at a rate with a constraint. Additionally, when M is fixed, as long as the first part holds, the second and third parts hold naturally. Assumptions 1-6 are verified in Section S3 of the Supplementary Material.

Remark 3. We note that Assumption 4 is used exclusively for establishing the convergence rate result in Theorem 3, and is not required for Theorems 1 and 2. The scenario where models with dimensions d_0 and $d_0 + 1$ produce similar predicted values occurs precisely

within the correctly specified regime ($\mathcal{T} = \{d_0, \ldots, M\}$). In this regime, Theore 2 already guarantees that the total weight $\widehat{\zeta} = \sum_{m \in \mathcal{T}} \widehat{w}_m$ converges to 1 in probability, which does not rely on Assumption 4. Therefore, even if Assumption 4 is violated in this regime, the fundamental theoretical guarantees of NetMA remain fully preserved.

We acknowledge that Assumption 4 does require the candidate models to exhibit sufficient diversity in their prediction vectors, and this condition may weaken when the candidate set includes models with very close dimensions such as d_0 and $d_0 + 1$. As demonstrated in our simulation studies, the NetMA method consistently performs well even when candidate dimensions include d_0 and its neighbors, confirming that the final averaged prediction is not compromised because any convex combination of models in $\mathcal{T}$ yields similarly accurate predictions.

Theorem 3. *If w^0 is an interior point of $\mathcal{W}$, and Assumptions 1 and 4-6 are satisfied, then $\widehat{w}$ satisfies $\|\widehat{w} - w^0\| = O_p\left(\xi^{1/2}|\Psi_1|^{-1/2+\kappa}\right)$, where κ is defined in Assumption 6.*

Theorem 3 gives the convergence rate of the estimated weights $\widehat{w}$ towards the optimal weights w^0, which is associated with the sample size $|\Psi_1|$ and the minimized risk ξ. Specifically, the slower the rate of $\xi \to \infty$, the faster the rate of $\widehat{w} \to w^0$ as $|\Psi_1| \to \infty$. Note that $CV(w)$ is a convex quadratic function over the simplex $\mathcal{W}$. Moreover, as shown in the proof of Theorem 3 (Section S2.3 of the Supplementary Material), Assumption 4 ensures the uniqueness of the global minimizer. The requirement that w^0 is an interior point of $\mathcal{W}$ facilitates the asymptotic analysis by ensuring that the neighborhood of the optimal weight vector remains entirely within the feasible parameter space. This condition allows us to employ standard local perturbation techniques. From a practical standpoint, violating this interior point assumption does not negatively impact numerical performance. We solve the

optimization problem using quadratic programming which efficiently handles constraints and solutions on the boundary of the simplex. Furthermore, our method remains robust even when the interior assumption is theoretically violated. For example, Theorem 2 proves that the method asymptotically assigns all weights to the models with large enough dimensions. This phenomenon is clearly illustrated in the additional simulation results reported in Figure 4. In particular, when $M = 5$ and $d_0 = 4$, for $N = 500$, NetMA assigns all the weight to the candidate models with latent dimensions being 4 or 5, whereas the first three candidate models receive zero weight. Hence, the resulting solution lies on the boundary of the simplex, yet NetMA still achieves superior numerical performance. The proof of Theorem 3 is presented in Section S2.3 of the Supplementary Material.

3. Model Averaging Prediction for Multi-Layer Networks

A multi-layer network is a collection of various networks connecting the same set of nodes. In many applications, some complex relationships can be characterized using multi-layer networks, such as social networks of friendships, work connections, as well as networks that evolve over time. Here, we propose a model averaging method for link prediction in multi-layer networks, where the weights are determined by considering the performance of candidate models across all layers.

3.1 Model framework

Assume that the multi-layer networks are composed of T networks over a common set of N nodes. For $t = 1, \ldots, T$, the t-th layer network is represented by an adjacency matrix $A^{(t)} \in \{0, 1\}^{N \times N}$, where $A^{(t)}_{ij} = A^{(t)}_{ji} = 1$ if node i and node j are connected and $A^{(t)}_{ij} = A^{(t)}_{ji} =$

0 otherwise. Similarly, we consider using the inner-product LSM. For any $t = 1, \ldots, T$, $i, j = 1, \ldots, N$ and $i < j$, we have

$$\begin{aligned} A_{ij}^{(t)} &\sim \text{Bernoulli}\left(P_{ij}^{(t)}\right), \quad \text{with} \\ \text{logit}(P_{ij}^{(t)}) &= \Theta_{ij}^{(t)} = \alpha_i + \alpha_j + z_i^{(t)\top} z_j^{(t)}, \end{aligned} \tag{3.3}$$

where $z_i^{(t)} \in \mathbb{R}^{d_t}$. Here, d_t represents the true but unknown latent dimension of the t-th layer network, which is allowed to vary across layers. Regarding the parameters, we assume all layers share the same α, while the latent positions of nodes may vary across different layers. Specifically, the layer-invariant α_i characterizes the stable and overall popularity of the nodes, whereas the layer-specific latent position $z_i^{(t)}$ captures structural variation of the network across different layers. This setup aligns with existing studies such as Friel et al. (2016); Sewell and Chen (2017, 2015) and Sewell and Chen (2016).

For presentation simplicity, we rewrite the model in matrix form, i.e., $\Theta^{(t)} = \alpha 1_N^\top + 1_N \alpha^\top + Z^{(t)} Z^{(t)\top}$, where $Z^{(t)} = (z_1^{(t)}, \ldots, z_N^{(t)})^\top \in \mathbb{R}^{N \times d_t}$. Denote $\mathbb{Z} = \{Z^{(1)}, \ldots, Z^{(T)}\}$. To ensure the identifiability of parameters $\{\alpha, \mathbb{Z}\}$, we assume the latent variables are centered, that is $JZ^{(t)} = Z^{(t)}$.

Then we develop a PGD algorithm for multi-layer networks to estimate the Model (3.3). We first define the objective function as the negative conditional log-likelihood of $\{A^{(t)}\}_{t=1}^T$ under Model (3.3):

$$\mathcal{L}_\dagger(\alpha, \mathbb{Z}) = -\sum_{t=1}^{T}\sum_{i=1}^{N}\sum_{j=1}^{N} \log P\left(A_{ij}^{(t)} \mid \alpha, \mathbb{Z}\right) = -\sum_{t=1}^{T}\sum_{i=1}^{N}\sum_{j=1}^{N} \left\{A_{ij}^{(t)}\Theta_{ij}^{(t)} - f(\Theta_{ij}^{(t)})\right\}.$$

We need to find the estimators of α and $\mathbb{Z}$ that minimize the objective function. The original

PGD algorithm is designed for single-layer networks (Ma et al., 2020). Here, we extend it to multi-layer networks, which is similar to the algorithm proposed by Zhang et al. (2020). The procedure is summarized in Algorithm 1 in Section S4 of the Supplementary Material. We adopt the initialization method proposed by Ma et al. (2020) to obtain appropriate initial values. The original projected gradient descent algorithm is extended to the multi-layer network.

3.2 Model averaging criterion

Following Section 2.2, we consider M candidate models, where the dimension of the latent space in the mth ($m = 1, \ldots, M$) model is m. The mth candidate model is $\Theta^{(t)} = \alpha 1_N^\top + 1_N \alpha^\top + Z_{(m)}^{(t)} Z_{(m)}^{(t)\top}$, where $Z_{(m)}^{(t)} \in \mathbb{R}^{N \times m}$. Here, it is important to clarify the distinction between the true latent dimension and the candidate model dimension. As defined in Section 3.1, d_t denotes the true but unknown latent dimension of the t-th layer network, which may vary across layers in the data generating process. In contrast, m refers to the dimension of the m-th candidate model. For a specific candidate model, we fit a latent space model with a fixed dimension m across all layers. Although a single candidate model assumes a common dimension, our NetMA method approximates the complex structure of the multi-layer network by averaging multiple candidate models based on their predictive performance.

To select model weights for multi-layer networks, we propose a K-fold edge cross-validation criterion. Following the single-layer setting in Section 2.2, we partition the set of all nodal pairs Ψ into an observed set Ψ_1 and a missing set Ψ_2, where the same partition is applied across all T layers. This missing pattern is natural when the observation mechanism operates at the nodal-pair level rather than at the individual layer level. For example, in

multi-relational social network surveys, respondents are typically presented with a roster and asked about multiple types of ties (e.g., friendship, advice, collaboration) with each individual on the roster. Since the roster determines which pairs are queried, and the same roster applies to all relationship types, the set of observed pairs is identical across layers. Similarly, in multi-layer biological interaction networks, high-throughput screening experiments often test a fixed set of molecular pairs for multiple types of interactions simultaneously, so that untested pairs are missing across all interaction types. Extending the framework to accommodate layer-specific missing patterns is an interesting direction for future research.

Specifically, we aim to minimize the loss function $L_\dagger(w) = \sum_{t=1}^{T} \sum_{(i,j)\in\Psi_1} \{\widehat{P}_{ij}^{(t)}(w) - P_{ij}^{(t)}\}^2$ where the aggregated prediction $\widehat{P}_{ij}^{(t)}(w)$ is defined as $\widehat{P}_{ij}^{(t)}(w) = \sum_{m=1}^{M} w_m \widehat{P}_{(m),ij}^{(t)}$. Here, $\widehat{P}_{(m),ij}^{(t)}$ denotes the estimated connection probability from the mth candidate model in layer t, as described in Section S5 of the Supplementary Material. In light of the unattainability of the objective function $L_\dagger(w)$, we opt for the determination of data-driven weights through the K-fold edge cross-validation criterion. The detailed procedure is similar to that for single-layer networks and can be found in Section S5 of the Supplementary Material.

Furthermore, we establish the theoretical foundations of the proposed method for multi-layer networks. We demonstrate that the weights obtained via K-fold edge cross-validation are asymptotically optimal (Theorem S1) and consistently assign full weight to models with large enough latent dimensions (Theorem S2). Additionally, we derive the convergence rate of these weights towards the infeasible optimal weight vector, highlighting how the rate scales with both the number of nodal pairs and the number of layers T (Theorem S3). A comprehensive discussion of the underlying assumptions and the full technical proofs are provided in Sections S1 and S2 of the Supplementary Material.

4. Simulation Studies

In this section, we evaluate the NetMA methods by using simulation examples. In addition to NetMA, we also consider three other methods. The first one is the oracle method, which uses the true model structure and latent dimension. The second one is the "equal" version which assigns equal weights to all the candidate models. That is, if there are M candidate models, then the weight of each model is $1/M$. The third one is the best model selected by the edge cross-validation (ECV) procedure of Li et al. (2020) and Gao and Ma (2020). Specifically, we divide the nodal pairs in Ψ_1 into K groups, and then use $K-1$ groups to fit the model each time. Select the model that minimizes the loss on the hold-out set as the best model. For example, in a single-layer network, the ECV algorithm simply replaces Step 3 in Section 2.2 with $m^* = \text{argmin}_m |\Psi_1|^{-1} \sum_{k=1}^{K} \left\| A^{[k]} - \widetilde{P}^{[k]}_{(m)} \right\|_F^2$. We compare the four methods for both single-layer and multi-layer networks. The simulation results of multi-layer networks are presented in Section S6.2 of the Supplementary Material due to space constraints. The findings highlight the robustness and adaptivity of NetMA in both well-specified and misspecified model settings.

In the following simulations, we consider three cases. The first case measures the effect of the dimensions of latent vectors on the prediction performance. The second case evaluates the effect of the size of the network. The third case evaluates the effect of the network density. Due to space limitations, the third case is presented in Section 6.1 of the Supplementary Material. We set $K = 10$ and the ratio of the numbers of nodal pairs in Ψ_1 and Ψ_2 as 7:3.

4.1 Single-layer networks

In the following simulation design, we generate a network adjacency matrix $A = (A_{ij}) \in \mathbb{R}^{N\times N}$, where A_{ij}s are generated from the Bernoulli distribution independently. Specifically, A_{ij} takes the value 1 with probability P_{ij}, and the value 0 with probability $1 - P_{ij}$. The probability P_{ij} is determined by $\text{logit}(P_{ij}) = \alpha_i + \alpha_j + z_i^\top z_j$, where α_i is generated from Uniform$(-1, 1)$ independently for $i = 1, \ldots, N$. For the latent vectors, we first generate a matrix $Z \in \mathbb{R}^{N\times d_0}$ such that each entry is generated from $N(0, 1)$ independently. Then we transform Z by setting $Z = JZ$ where $J = I_N - 1_N 1_N^\top/N$ and rotate Z such that $Z^\top Z \propto I_{d_0}$. Finally, scale Z such that $Z^\top Z = NI_{d_0}$. In particular, to control the expected average degree of the networks, we consider transforming P by multiplying a constant γ. For example, if we want to obtain a network with an expected average degree of $\bar{D}$, then γ equals $\bar{D}/ARS(P)$, where $ARS(P)$ refers to the average of the row sums of P.

We focus on estimating the network connection probability matrix P. Given an estimator $\widehat{P}$, the performances of the four methods are measured by the relative empirical risk function, which is calculated as

$$\widehat{R}(\widehat{w}) = \frac{1}{Q}\sum_{q=1}^{Q} \frac{\sum_{(i,j)\in\Psi_2}\left\{\widehat{P}_{ij}^{\{q\}}(\widehat{w}^{\{q\}}) - P_{ij}^{\{q\}}\right\}^2}{\sum_{(i,j)\in\Psi_2}\left\{P_{ij}^{\{q\}}\right\}^2},$$

where Q is the number of simulation replications, and $\widehat{P}_{ij}^{\{q\}}(\widehat{w}^{\{q\}})$ denotes the prediction based on the "oracle", the "equal", ECV and NetMA weights in the qth replication. In the following simulations, we set $Q = 100$, and consider a sequence of candidate models, where the dimension of the latent vectors in the mth candidate model is m, $m = 1, \ldots, M$.

Case 1 (Dimension of Latent Space). In this case, we set the size of the network as

$N = 200$, and the expected average nodal degree is 60. We vary M from 2 to 12, and let the true dimensions of the latent vectors d_0 be 4, 7 and 10, respectively.

The resulting relative risks under each d_0 are presented in Figure 1. Figure 1 shows from left to right the cases where the true dimension is 4, 7 and 10, respectively. The upper panels of Figure 1 show the change in relative risk as M changes, and the lower panels of Figure 1 show how the weight distribution changes as M varies. It can be seen that, as d_0 increases, the advantage of our method becomes more and more obvious, which is even better than the "oracle". This is mainly because when the network size and network density are fixed, the increase of d_0 introduces more parameters to be estimated. Thus, when d_0 is large, the true model may yield poor results. This phenomenon can be verified by observing the relative risk of the "oracle" from the upper panels of Figure 1 from left to right. To be more specific, in the setting of $d_0 = 4$, when all the candidate models are misspecified, "oracle" performs the best. As the candidate models gradually include the true model, the relative risk of NetMA and the "equal" drops dramatically. In the meanwhile, models with higher latent space dimensions are given higher weights. When the candidate models include the true model, the model selected by ECV performs similarly to the "oracle", while NetMA and the "equal" perform similarly, and much better than the "oracle" and ECV. In the setting of $d_0 = 7$, the "oracle" performs the best only when $M = 2$. When the dimension of the latent space in the candidate model exceeds 2, NetMA performs the best. As M increases, NetMA's superiority over the "equal" also becomes more apparent. In the setting of $d_0 = 10$, regardless of whether the candidate model contains the true model, the performance of ECV is similar to that of the "oracle", which is much worse than NetMA and the "equal". Additionally, NetMA performs better than the "equal", and the gap between NetMA and the "equal" is larger than their

gap under $d_0 = 4$ and $d_0 = 7$. It is worth noting that NetMA outperforms the oracle model, particularly when the true latent dimension is large or the network is sparse, as illustrated in Figure S1 of the Supplementary Material. This observation is consistent with recent findings in network estimation by Li and Le (2024). Heuristically, while the oracle model utilizes the true dimension, it suffers from high estimation variance due to the large number of parameters $N \times d_0$ required to be estimated from limited edge data. In contrast, NetMA combines multiple models including lower-dimensional ones. Although these simpler models introduce specification bias, they are more stable and possess lower estimation variance. In the difficult regime such as sparse networks or small sample sizes, NetMA outperforms the oracle by leveraging optimal weighting to fuse the strengths of heterogeneous candidate models, thereby achieving a superior balance between bias and variance.

Case 2 (Number of nodes). In this case, we vary the size of the networks N from 100 to 500. In order to maintain consistent density for networks of different sizes, we set the expected average nodal degree of the network to be $0.3(N-1)$. The true dimension of the latent vectors is fixed at $d_0 = 6$.

Figure 2 illustrates the relative risks for four distinct settings of Case 2, corresponding to candidate model set sizes of $M = 2,\ 4,\ 6$ and 8, respectively. When the maximum dimension of the latent space in the candidate models is 2 or 4, which is smaller than the true dimension of latent space, NetMA performs the best under small network sizes. As the network size increases, the advantage of the "oracle" becomes more evident. However, apart from the "oracle", NetMA performs the best, especially when $M = 4$, which is more pronounced. When the maximum dimension of the latent space in the candidate models is equal to the true latent space dimension ($M = 6$), NetMA performs the best for most cases. When N

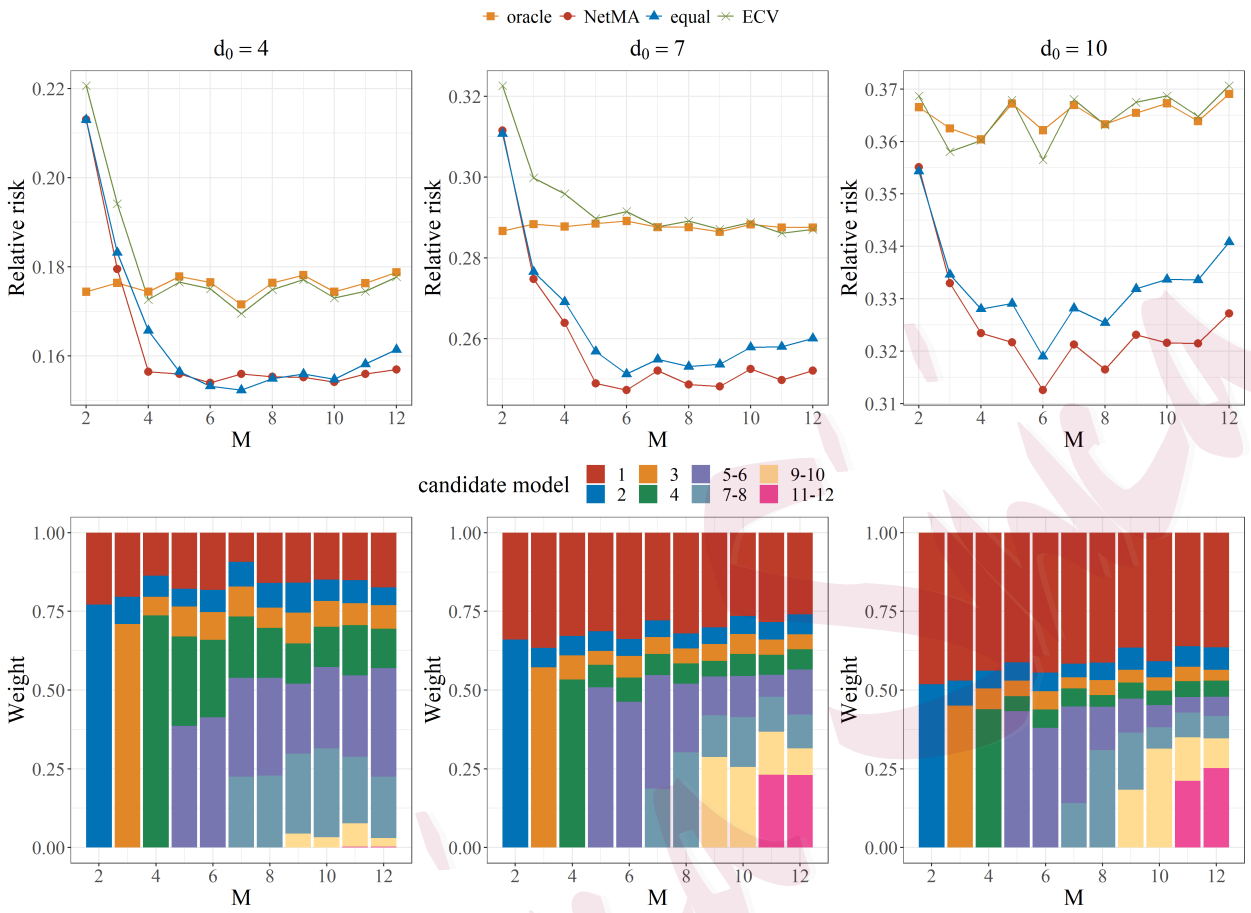

Figure 1: Relative risks and model weights for the four methods with different candidate models in single-layer networks.

is relatively large, the performance of the "oracle" and ECV becomes increasingly similar to that of NetMA, while the performance of the "equal" shows a certain gap compared to NetMA. Finally, when the candidate set is over-fitted with $M = 8$, NetMA maintains its superior performance. This demonstrates the robustness of our method to the inclusion of redundant high-dimensional models.

We further examine the performance of the proposed method when the true latent dimension is larger than the candidate model dimensions. Following the setup of Case 2, we increase the true latent dimension to $d_0 = 20$ while setting the number of candidate

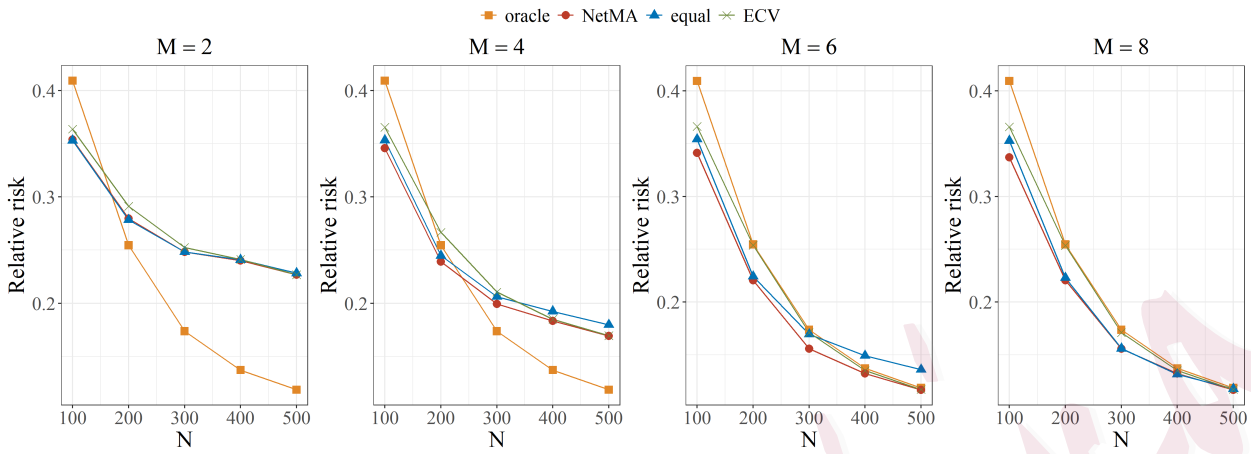

Figure 2: Relative risks of the four methods with different network sizes in single-layer networks, with the number of candidate models M varying from 2, 4, 6 to 8.

models to $M = 10$. Consequently, the candidate set does not contain any models with large enough latent dimensions. The network size N varies from 100 to 500. The results, presented in Figure 3, show that NetMA consistently achieves the lowest relative risk compared to competing methods across all sample sizes. Moreover, a comparison with Figure 2 (where $d_0 = 6$) reveals that the performance advantage of NetMA is even more substantial in this high-dimensional, misspecified setting.

To empirically validate the theoretical convergence of the weight estimator established in Theorem 2, we examine the behavior of $\widehat{\zeta}$ as the network size increases. We adopt a setting with $M = 5$ and $d_0 = 4$, ensuring the candidate models cover the true latent dimension, and vary N from 100 to 500. As illustrated in Figure 4, the estimated weight $\widehat{\zeta}$ increases monotonically with the sample size N and gradually approaches 1 as N becomes large. This finding confirms that our method can correctly identify the model structure asymptotically, providing strong empirical support for Theorem 2.

To further assess the predictive capability of our estimator, we conduct a comparative

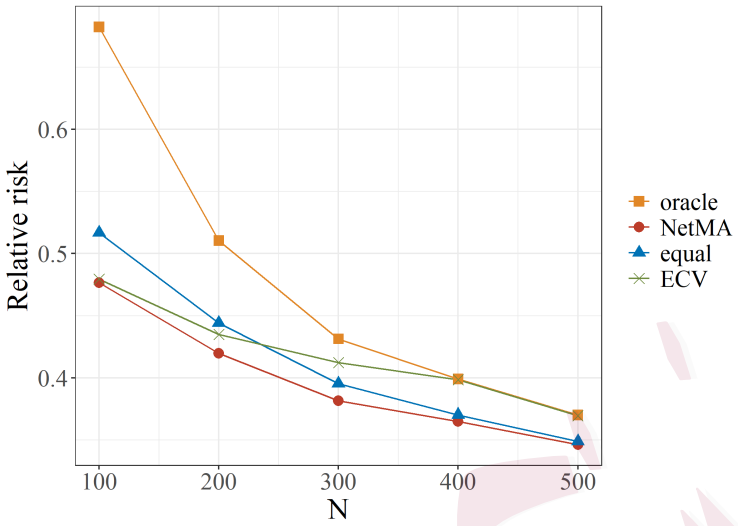

Figure 3: Relative risks of the four methods with different network sizes in single-layer networks when $M = 10$ and $d_0 = 20$.

study with the methods proposed by Li and Le (2024). Specifically, we include the Exponential Weighting method, the Non-Negative Linear mixing method, and the bounded Non-Negative Linear mixing method. We denote these methods as EXP, NNL, and NNL.bound respectively. The NNL method estimates weights without imposing a unit-sum constraint, whereas the NNL.bound method applies a further restriction to ensure the probability estimates fall within the unit interval. Figure 5 presents the relative risks across varying network sizes. The results indicate that the the unconstrained NNL method exhibits the highest relative risk among all candidates. This finding supports the discussion that the absence of a unit-sum constraint may lead to unbounded predictions. The bounded version NNL.bound mitigates this issue and improves performance. Moreover, while the EXP method demonstrates strong competitiveness, NetMA consistently achieves the lowest relative risk in most settings. Notably, as the network size N becomes large, the performance of the competing methods like EXP becomes increasingly similar to that of NetMA.

We subsequently explore two extensions in single-layer networks: (1) the inclusion of

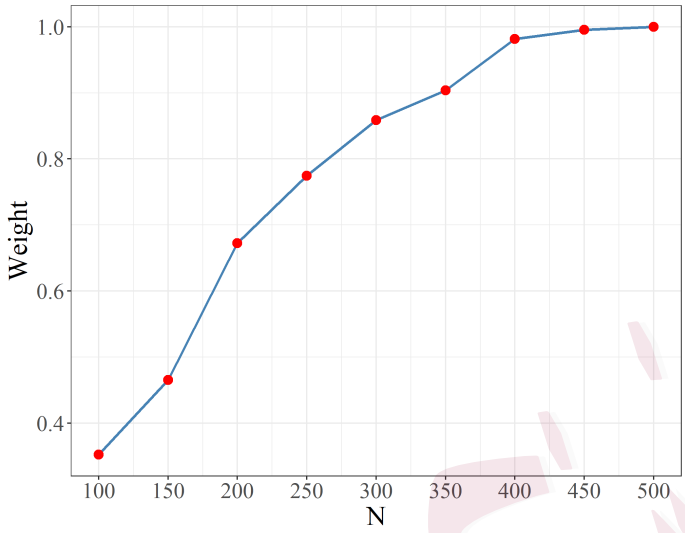

Figure 4: Sum of model weights placed on the models with large enough dimensions with different network sizes in single-layer networks when $M = 5$ and $d_0 = 4$.

edge covariates, and (2) non-random missing data mechanisms.

In the first extension, we consider Model (2.2) with edge covariates. Specifically, we generate β and each entry of the covariate matrix X from Uniform$(0, 1)$. The generation for the other parameters remains unchanged. Figure 6 shows the results of the four methods as the numbers of nodes and candidate models change. It can be observed that Figure 6 is very similar to Figure 2, i.e., our method performs best when the candidate set includes models with large enough latent dimensions, or when the candidate models do not include models with large enough latent dimensions but the number of nodes is relatively small.

In the second extension, we consider a case of non-random missingness, i.e., egocentrically sampled networks. These networks are constructed through egocentric sampling, which is a procedure where a subset of nodes is first sampled, and then the links between these nodes are recorded, while other information remains unknown (Li et al., 2023). An example of the adjacency matrix of an egocentrically sampled network is shown in Figure 7. In Figure 7, the grey area represents the observable parts, while the white area indicates the missing

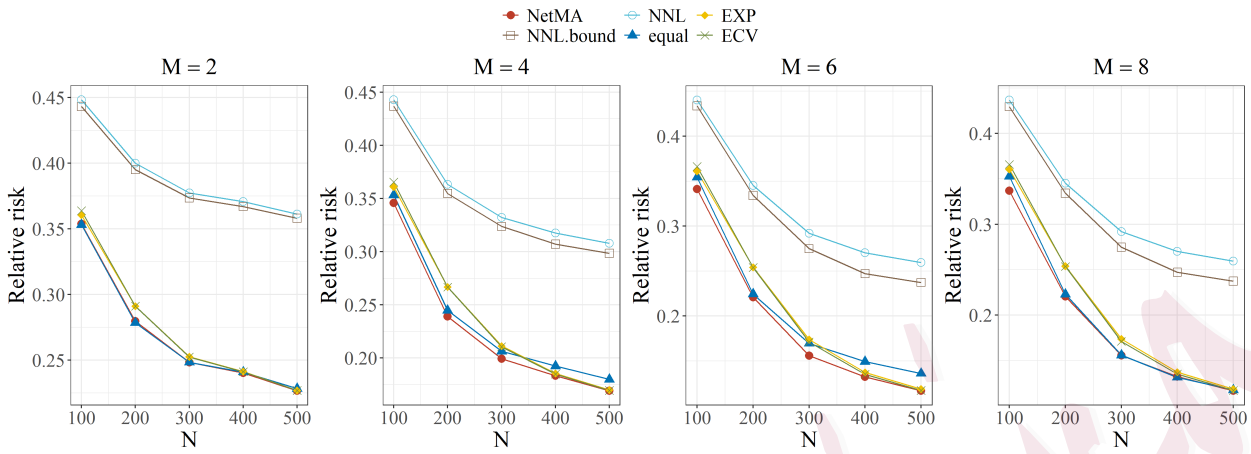

Figure 5: Comparison of relative risks between NetMA and the methods proposed by Li and Le (2024) under different network sizes, with the number of candidate models M varying from 2, 4, 6 to 8.

parts. Our goal is to predict the missing links based on the information from the observed links. In the simulation, we assume that 90% of nodes are sampled, meaning that the links between the remaining 10% of the nodes are missing. All other settings are the same as those in Case 2. Figure 8 shows the prediction results of different methods under the egocentric missing situation. It can be seen that when the candidate set does not include models with large enough latent dimensions, NetMA performs second only to the oracle, especially when the number of nodes is relatively large. When the candidate set includes models with large enough latent dimensions, NetMA performs the best, particularly when the network size is relatively small.

5. Empirical Example

In addition to the following empirical analysis based on multi-layer virtual world data, we also apply the proposed NetMA method to a single-layer real-world network constructed from the ResearchGate platform. Due to space limitations, the detailed description of this

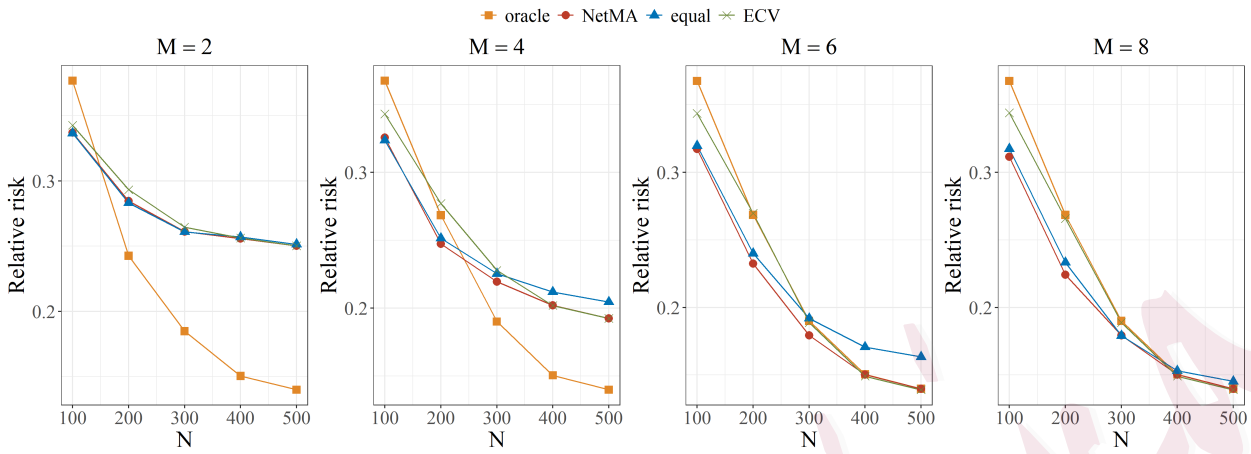

Figure 6: Relative risks of the four methods with different network sizes in single-layer networks with edge covariates.

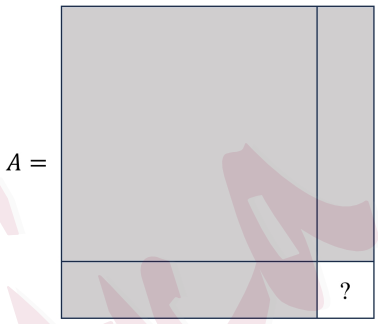

Figure 7: An illustration of the adjacency matrix of an egocentrically sampled network, where grey blocks are observed and the white block is missing.

single-layer analysis, including dataset construction, model setup, and experimental results, is provided in the Supplementary Material (Section S7.2). The results further demonstrate the superior performance and robustness of NetMA compared to ECV and simple averaging methods.

5.1 Prediction on virtual world data

In this section, we apply the proposed method NetMA to real-world data with a multi-layer network. Jankowski et al. (2017) provides a dataset which contains the record of six types

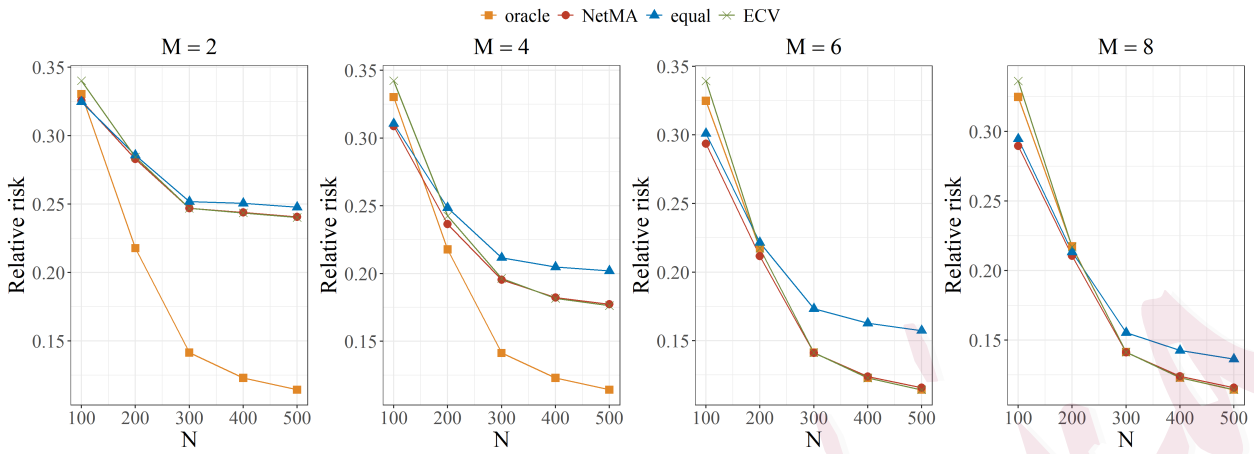

Figure 8: Relative risks of the four methods with different network sizes in single-layer networks with nonrandom missing.

of spreading events that occurred in a virtual world platform. The types of events include campaigns, friends, logins, messages, transactions and visits. Here, we extract a subset of common users present in four events and construct a multi-layer network. The network consists of four layers, representing friend, message, transaction and visit relationships. Each layer has 183 users and thus $N = 183$. Considering that some of the relations are directed, we convert the directed networks to undirected networks based on the existence of any single directional edge between nodes.

We apply the model averaging and model selection methods to analyse the multi-layer network. Here, we consider three scenarios, namely $M = 4$, 6, and 8. We consider three scenarios for $\pi_{12} = |\Psi_1|/|\Psi_2|$: 5/5, 7/3, and 9/1. In addition, when selecting weights for NetMA and models for ECV, we take $K = 5$ and $K = 10$ for the K-fold cross-validation. The experiments are replicated 100 times, and the results for link prediction are shown in Table 1. As we can see, NetMA with 5-fold cross-validation performs the best in terms of average AUC in all scenarios. As M increases, the advantage of NetMA becomes more

pronounced. This trend is likely driven by two key factors. First, real-world networks may possess rich underlying structures that correspond to a potentially high-dimensional latent space. When the maximum dimension M is small, the candidate models might not fully capture these features, whereas a larger candidate model offers a better opportunity to approximate the complex true structure. Second, as the number of candidate models increases, the uncertainty in identifying a single best dimension may rise for standard model selection methods. In contrast, NetMA reduces the risk of relying on a single and potentially suboptimal model by integrating information from various dimensions, thus yielding a more pronounced advantage in larger search spaces. Due to space limitations, weight allocation results are presented in Supplementary Table S1. Unlike the results on the weight allocations for single-layer networks, the weight allocations for multi-layer networks are inconsistent for ECV and NetMA. Furthermore, the choice of K seems to have a great impact on the weight allocation in ECV, while its influence on NetMA is comparatively small.

Table 1: Average AUC for link prediction on virtual world data (standard errors are in brackets).

	$\lvert\Psi_1\rvert/\lvert\Psi_2\rvert$	equal	ECV ($K=5$)	ECV ($K=10$)	NetMA ($K=5$)	NetMA ($K=10$)
$M=4$	5/5	0.6979 (0.0047)	0.6872 (0.0055)	0.6859 (0.0059)	**0.7006 (0.0047)**	0.7000 (0.0046)
	7/3	0.7235 (0.0050)	0.7117 (0.0055)	0.7116 (0.0048)	**0.7246 (0.0051)**	0.7235 (0.005)
	9/1	0.7388 (0.0087)	0.7283 (0.0085)	0.7270 (0.0088)	**0.7394 (0.0087)**	0.7385 (0.0088)
$M=6$	5/5	0.6962 (0.0050)	0.6870 (0.0055)	0.6848 (0.0057)	**0.7029 (0.0046)**	0.7021 (0.0044)
	7/3	0.7220 (0.0049)	0.7111 (0.0055)	0.7106 (0.0045)	**0.7258 (0.0048)**	0.7247 (0.0048)
	9/1	0.7387 (0.0085)	0.7273 (0.0086)	0.7259 (0.0086)	**0.7417 (0.0088)**	0.7408 (0.0086)
$M=8$	5/5	0.6986 (0.0050)	0.6871 (0.0054)	0.6842 (0.0059)	**0.7063 (0.0046)**	0.7055 (0.0042)
	7/3	0.7240 (0.0048)	0.7110 (0.0052)	0.7097 (0.0054)	**0.7297 (0.0046)**	0.7285 (0.0046)
	9/1	0.7404 (0.0083)	0.7262 (0.0085)	0.7257 (0.0085)	**0.7445 (0.0083)**	0.7434 (0.0083)

6. Discussion

In this paper, we introduce a model averaging strategy for link prediction. Specifically, we focus on LSMs and allow for different dimensions of latent space. Our K-fold edge cross-validation procedures can be used to select the data-driven optimal weights for candidate models in both single-layer and multi-layer networks. The proposed method fully leverages information from multiple models and thus leads to the desirable prediction performance. Specifically, when the candidate models are misspecified, NetMA is proved to be asymptotically optimal in terms of squared errors; when the candidate model set includes models with large enough latent dimensions, it assigns all weights to these models asymptotically. Besides, we derive the rate of the NetMA-based empirical weights converging to the theoretically optimal weights. Simulation studies show the promise of the NetMA method in both single-layer networks and multi-layer networks. We also evaluate its competitive performance empirically in link prediction problems.

Although we focus on LSMs in this paper, the NetMA procedure, along with its theoretical properties, can be applied to models for any undirected and unweighted network. As Li and Le (2024) demonstrates, one can consider integrating various models for networks, such as SBM, the degree-corrected stochastic block model of Karrer and Newman (2011), and the universal singular value thresholding of Chatterjee (2012) in the future. For other types of networks, such as directed networks, we can consider candidate models designed for directed networks. For example, Zhang et al. (2022) proposed a LSM for directed networks, where $\text{logit}(P_{ij}) = \nu_i^\top \omega_j$, with $\nu_i \in \mathbb{R}^d$ and $\omega_j \in \mathbb{R}^d$ being the latent vectors of out-node i and in-node j. It makes sense to extend the model averaging method to this situation. Furthermore, in the study of networks, there are not only issues related to link prediction

but also several other problems such as community detection (Lancichinetti and Fortunato, 2009) and predicting the response observed for each node in social network (Zhu et al., 2017). It would be interesting to extend model averaging to these research problems.

Supplementary Materials

The supplementary material provides additional details including theoretical proofs, assumption verifications, algorithmic procedures, and extended simulation and empirical results. Specifically, Section S1 provides the theoretical results for multi-layer networks. Section S2 contains the proofs of Theorems 1–3 and S1–S3. Section S3 provides detailed verifications of Assumptions 1–6. Section S4 introduces the projected gradient descent algorithm for parameter estimation in multi-layer networks. Section S5 describes the procedure for selecting model weights in the multi-layer setting. Section S6 presents the simulation results for both single-layer and multi- layer networks. Finally, Section S7 reports the result of weight allocation and empirical analysis based on the ResearchGate dataset.

Acknowledgements

Jun Liao's work was partially supported by the Humanities and Social Science Foundation of the Ministry of Education of China (24YJC910004), and the National Natural Science Foundation of China (12001534). Xinyan Fan's research was supported by the National Natural Science Foundation of China (72571272, 12201626), the MOE Project of Key Research Institute of Humanities and Social Sciences (22JJD110001), and the Public Computing Cloud, Renmin University of China. Kuangnan Fang's research was supported by the National Natural Science Foundation of China (12571313,72233002), and the National Statistical Science

Research Projects(2025LD002).

References

Ando, T. and K.-C. Li (2017). A weight-relaxed model averaging approach for high-dimensional generalized linear models. The Annals of Statistics 45(6), 2654 – 2679.

Chatterjee, S. (2012). Matrix estimation by universal singular value thresholding. The Annals of Statistics 43, 177–214.

Dong, X., F. An, Z. Dong, Z. Wang, M. Jiang, P. Yang, and H. An (2021). Optimization of the international nickel ore trade network. Resources Policy 70, 101978.

Durante, D. and D. B. Dunson (2014). Nonparametric bayes dynamic modelling of relational data. Biometrika 101(4), 883–898.

Erdös, P. and A. Rényi (1959). On random graphs i. Publicationes Mathematicae Debrecen 6, 290–297.

Fragoso, T. M., W. Bertoli, and F. Louzada (2018). Bayesian model averaging: A systematic review and conceptual classification. International Statistical Review 86(1), 1–28.

Friel, N., R. Rastelli, J. Wyse, and A. E. Raftery (2016). Interlocking directorates in Irish companies using a latent space model for bipartite networks. Proceedings of the National Academy of Sciences 113(24), 6629–6634.

Gao, C. and Z. Ma (2020). Discussion of 'network cross-validation by edge sampling'. Biometrika 107(2), 281–284.

Gao, Y., X. Zhang, S. Wang, and G. Zou (2016). Model averaging based on leave-subject-out cross-validation. Journal of Econometrics 192(1), 139–151.

Gwee, X. Y., I. C. Gormley, and M. Fop (2025). A latent shrinkage position model for binary and count network data. Bayesian Analysis 20(2), 405 – 433.

Handcock, M. S., A. E. Raftery, and J. M. Tantrum (2007). Model-based clustering for social networks. Journal of the Royal Statistical Society: Series A (Statistics in Society) 170(2), 301–354.

Hansen, B. E. (2007). Least squares model averaging. Econometrica 75(4), 1175–1189.

Hansen, B. E. and J. S. Racine (2012). Jackknife model averaging. Journal of Econometrics 167(1), 38–46.

Hoeting, J. A., D. Madigan, A. E. Raftery, and C. T. Volinsky (1999). Bayesian model averaging: A tutorial. Statistical Science 14(4), 382–417.

Hoff, P. D. (2007). Modeling homophily and stochastic equivalence in symmetric relational data. In Proceedings of the 20th International Conference on Neural Information Processing Systems, pp. 657–664.

Hoff, P. D., A. E. Raftery, and M. S. Handcock (2002). Latent space approaches to social network analysis. Journal of the American Statistical Association 97(460), 1090–1098.

Holland, P. W., K. B. Laskey, and S. Leinhardt (1983). Stochastic blockmodels: First steps. Social Networks 5(2), 109–137.

Holland, P. W. and S. Leinhardt (1981). An exponential family of probability distributions for directed graphs. Journal of the American Statistical Association 76(373), 33–50.

Jankowski, J., R. Michalski, and P. Bródka (2017). A multilayer network dataset of interaction and influence spreading in a virtual world. Scientific Data 4(1), 1–9.

Jo, W., D. Chang, M. You, and G.-H. Ghim (2021). A social network analysis of the spread of covid-19 in south korea and policy implications. Scientific Reports 11(1), 8581.

Karrer, B. and M. E. Newman (2011). Stochastic blockmodels and community structure in networks. Physical Review E 83(1), 016107.

Kim, B., K. H. Lee, L. Xue, and X. Niu (2018). A review of dynamic network models with latent variables. Statistics Surveys 12, 105.

Koren, Y., R. Bell, and C. Volinsky (2009). Matrix factorization techniques for recommender systems. Computer 42(8), 30–37.

Kossinets, G. (2006). Effects of missing data in social networks. Social networks 28(3), 247–268.

Kossinets, G. and D. J. Watts (2009). Origins of homophily in an evolving social network. American Journal of Sociology 115(2), 405–450.

Lancichinetti, A. and S. Fortunato (2009). Community detection algorithms: a comparative analysis. Physical Review E 80(5), 056117.

Li, G., M. Li, J. Wang, J. Wu, F.-X. Wu, and Y. Pan (2016). Predicting essential proteins based on subcellular localization, orthology and ppi networks. BMC Bioinformatics 17, 571–581.

Li, J., G. Xu, and J. Zhu (2023). Statistical inference on latent space models for network data. arXiv preprint arXiv:2312.06605.

Li, T. and C. M. Le (2024). Network estimation by mixing: Adaptivity and more. Journal of the American Statistical Association 119(547), 2190–2205.

Li, T., E. Levina, and J. Zhu (2020). Network cross-validation by edge sampling. Biometrika 107(2), 257–276.

Li, T., Y.-J. Wu, E. Levina, and J. Zhu (2023). Link prediction for egocentrically sampled networks. Journal of Computational and Graphical Statistics 32(4), 1296–1319.

Liao, J. and G. Zou (2020). Corrected mallows criterion for model averaging. Computational Statistics & Data Analysis 144, 106902.

Liao, J., G. Zou, Y. Gao, and X. Zhang (2021). Model averaging prediction for time series models with a diverging number of parameters. Journal of Econometrics 223(1), 190–221.

Lyu, Z., D. Xia, and Y. Zhang (2023). Latent space model for higher-order networks and generalized tensor decomposition. Journal of Computational and Graphical Statistics 32(4), 1320–1336.

Ma, Z., Z. Ma, and H. Yuan (2020). Universal latent space model fitting for large networks with edge covariates. Journal of Machine Learning Research 21(4), 1–67.

Mariadassou, M. and T. Tabouy (2020). Consistency and asymptotic normality of stochastic block models estimators from sampled data. Electronic Journal of Statistics 14(2), 3672 – 3704.

McPherson, M., L. Smith-Lovin, and J. M. Cook (2001). Birds of a feather: Homophily in social networks. Annual Review of Sociology 27(1), 415–444.

Menon, A. K. and C. Elkan (2011). Link prediction via matrix factorization. In Joint european conference on machine learning and knowledge discovery in databases, pp. 437–452. Springer.

Oh, M.-S. and A. E. Raftery (2001). Bayesian multidimensional scaling and choice of dimension. Journal of the American Statistical Association 96(455), 1031–1044.

Oh, M.-S. and A. E. Raftery (2007). Model-based clustering with dissimilarities: A Bayesian approach. Journal of Computational and Graphical Statistics 16(3), 559–585.

Óskarsdóttir, M., W. Ahmed, K. Antonio, B. Baesens, R. Dendievel, T. Donas, and T. Reynkens (2022). Social network analytics for supervised fraud detection in insurance. Risk Analysis 42(8), 1872–1890.

Pan, R., X. Chang, X. Zhu, and H. Wang (2022). Link prediction via latent space logistic regression model. Statistics and its Interface 15(3), 267–282.

Pan, R., Y. Gao, and H. Wang (2026). A latent space model for link prediction in statistical citation network. Journal of Multivariate Analysis 212, 105555.

Qiu, Y. and X. Zhang (2025). A transfer learning framework for multilayer networks via model averaging. arXiv preprint arXiv:2506.12455.

Serrat, O. (2017). Knowledge Solutions: Tools, Methods, and Approaches to Drive Organizational Performance. Springer Singapore.

Sewell, D. K. and Y. Chen (2015). Latent space models for dynamic networks. Journal of the American Statistical Association 110(512), 1646–1657.

Sewell, D. K. and Y. Chen (2016). Latent space models for dynamic networks with weighted edges. Social Networks 44,

105–116.

Sewell, D. K. and Y. Chen (2017). Latent space approaches to community detection in dynamic networks. Bayesian Analysis 12(2), 351 – 377.

Song, X., Y. Zhang, R. Pan, and H. Wang (2022). Link prediction for statistical collaboration networks incorporating institutes and research interests. IEEE Access 10, 104954–104965.

Sosa, J. and L. Buitrago (2021). A review of latent space models for social networks. Revista Colombiana de Estadística 44(1), 171–200.

Tang, W. and J. Zhu (2025). Population-level balance in signed networks. Journal of the American Statistical Association 120(550), 751–763.

Tanwar, M., R. Duggal, and S. K. Khatri (2015). Unravelling unstructured data: A wealth of information in big data. In 2015 4th International Conference on Reliability, Infocom Technologies and Optimization (ICRITO) (Trends and Future Directions), pp. 1–6.

Yang, Y. (2001). Adaptive regression by mixing. Journal of the American Statistical Association 96(454), 574–588.

Zhang, J., X. He, and J. Wang (2022). Directed community detection with network embedding. Journal of the American Statistical Association 117(540), 1809–1819.

Zhang, X. and H. Liang (2011). Focused information criterion and model averaging for generalized additive partial linear models. The Annals of Statistics 39(1), 174–200.

Zhang, X. and C.-A. Liu (2023). Model averaging prediction by K-fold cross-validation. Journal of Econometrics 235(1), 280–301.

Zhang, X., A. T. Wan, and G. Zou (2013). Model averaging by jackknife criterion in models with dependent data. Journal of Econometrics 174(2), 82–94.

Zhang, X., G. Xu, and J. Zhu (2022). Joint latent space models for network data with high-dimensional node variables. Biometrika 109(3), 707–720.

Zhang, X., S. Xue, and J. Zhu (2020). A flexible latent space model for multilayer networks. In Proceedings of the 37th International Conference on Machine Learning, pp. 11288–11297.

Zhu, L., D. Guo, J. Yin, G. Ver Steeg, and A. Galstyan (2016). Scalable temporal latent space inference for link prediction in dynamic social networks. IEEE Transactions on Knowledge and Data Engineering 28(10), 2765–2777.

Zhu, X., R. Pan, G. Li, Y. Liu, and H. Wang (2017). Network vector autoregression. The Annals of Statistics 45(3), 1096–1123.

Yan Zhang

School of Statistics and Data Science, Shanghai University of International Business and Economics, Shanghai, China

E-mail: zhangyan_elyssa@163.com

Jun Liao

Center for Applied Statistics and School of Statistics, Renmin University of China, Beijing, China

E-mail: junliao@ruc.edu.cn

Xinyan Fan

Center for Applied Statistics and School of Statistics, Renmin University of China, Beijing, China

E-mail: 20198102@ruc.edu.cn

Kuangnan Fang

School of Economics, Xiamen University, Xiamen, China; Fujian Key Laboratory of Statistical Science

E-mail: xmufkn@xmu.edu.cn

Yuhong Yang

Yau Mathematical Sciences Center, Tsinghua University, Beijing, China

E-mail: yyangsc@tsinghua.edu.cn