On adaptivity and minimax optimality of two-sided nearest neighbors

Tathagata Sadhukhan* Cornell University

Manit Paul* The Wharton School

Raaz Dwivedi Cornell Tech

Abstract

Nearest neighbor (NN) algorithms have been extensively used for missing data problems in recommender systems and sequential decision-making systems. Prior theoretical analysis has established favorable guarantees for NN when the underlying data is sufficiently smooth and the missingness probabilities are lower bounded. Here we analyze NN with non-smooth non-linear functions with vast amounts of missingness. In particular, we consider matrix completion settings where the entries of the underlying matrix follow an latent non-linear factor model, with the non-linearity belonging to a Hölder function class that is less smooth than Lipschitz. Our results establish following favorable properties for a suitable two-sided NN: (1) The mean squared error (MSE) of NN adapts to the smoothness of the non-linearity, (2) under certain regularity conditions, the NN error rate matches the rate obtained by an oracle equipped with the knowledge of both the row and column latent factors, and finally (3) NN's MSE is non-trivial for a wide range of settings even when several matrix entries might be missing deterministically. We support our theoretical findings via extensive numerical simulations and a case study with data from a mobile health study, HeartSteps.

1 INTRODUCTION

Latent factor models are ubiquitous in recommendation systems, panel data settings, sequential decision-making problems, and in various other scenarios. Matrix completion is a crucial problem in this context. Suppose $\Theta = ((\theta_{i,j})) \in \mathbb{R}^{n\times m}$ denotes the matrix of ground truths and $X \in \mathbb{R}^{n\times m}$ denotes the observed matrix. Let $A_{i,j}$ be the indicator variable denoting whether the (i,j)-th element of the matrix has been observed or not. We have the following model,

$$X_{i,j} = \begin{cases} \theta_{i,j} + \epsilon_{i,j} & \text{if } A_{i,j} = 1, \\ * & \text{if } A_{i,j} = 0. \end{cases} \tag{1}$$

Here $\epsilon_{i,j}$ is mean zero noise. The primary objective of matrix completion problem is to estimate the ground truths $\theta_{i,j}$ for both the missing as well as non-missing entries. Without any assumption on the matrix Θ this is a very difficult problem as there are a large number (nm) of unknown parameters as opposed to number of observations in this problem. To make this problem feasible it is generally assumed that the matrix Θ has an implicit low dimensional structure i.e. there are row latent factors $u_1, \cdots, u_n \in \mathbb{R}^{d_1}$, column latent factors $v_1, \cdots, v_m \in \mathbb{R}^{d_2}$, and a latent function f such that the following holds,

$$\theta_{i,j} = f(u_i, v_j) \quad \forall \quad (i,j) \in [n] \times [m].$$

A very popular choice of a bilinear latent function f is $f(u,v) = \langle u, v\rangle$. In this case, we have the decomposition $\Theta = UV^T$ where U, V are the matrices containing the row and column latent factors respectively. More generally, there are a large number of works (refer to Xu et al. (2013), Jain and Dhillon (2013), Zhong et al. (2015), Chiang et al. (2015), Lu et al. (2016), Guo (2017), Eftekhari et al. (2018), Ghassemi et al. (2018), Chiang et al. (2018), Arkhangelsky et al. (2019), Bertsimas and Li (2020), Agarwal et al. (2020), Agarwal et al. (2021), Burkina et al. (2021)) on matrix completion which assume that the ground truth matrix Θ can be decomposed as $U\Sigma V^T$ where U, V are the covariance matrices comprising of row and column latent factors respectively. However the setting when f is unknown and non-linear which is the main focus of this work, has received relatively little attention in the literature. Nearest neighbor (NN) algorithms have

[*] These authors contributed equally to this work

Proceedings of the 28[th] International Conference on Artificial Intelligence and Statistics (AISTATS) 2025, Mai Khao, Thailand. PMLR: Volume 258.

been observed to perform well in this set-up. There are many variants of NN algorithm in literature which people have tried in this context. NN algorithms approximate the L_2 distance between the rows and columns in the latent functional space and use those estimated distances to obtain a fixed-radius NN estimator. One of the most prominent works in this domain is that of Dwivedi et al. (2022a) who analyses the performance of row (user)-nearest neighbor with the objective of performing counterfactual inference in sequential experiments under the assumption that non-linear latent function f is Lipschitz in both the coordinates u and v. Another work by Dwivedi et al. (2022b) also ventures into this regime assuming the latent function f is a Lipschitz function satisfying certain convexity conditions and studies the performance of a doubly-robust nearest neighbor. Apart from this, Yu (2022) works with an unknown Hölder-continuous latent function f and has introduced a novel algorithm (a variant of the vanilla two-sided NN) which attains the minimax optimal non-parametric rate in a moderate regime assuming the knowledge of column latent factors.

We study the performance of the two-sided NN (TS-NN) method under the assumption that the latent function f is Hölder-smooth, none of the row or column latent factors are observed, and the entries of the matrix are missing not at random (MNAR). This non-parametric setting considers a much more general model class than the low rank bilinear class of functions. The assumption of the non-parametric model class, such as the one we analyse in our work, has been previously studied in Song et al. (2016), Li et al. (2019), Dwivedi et al. (2022b), and Yu (2022). Similar models have been previously widely studied in graphon estimation literature (with binary observations and symmetric matrix). Gao et al. (2015), Gao et al. (2016), Klopp et al. (2017), and Xu (2018) are some of the relevant references. Moreover, the MNAR regime is much closer to reality as compared to the missing completely at random (MCAR) regime. For instance in movie recommendation system, a user who does not like the action genre is very less likely to see movies with heavy action. Schnabel et al. (2016), Ma and Chen (2019), Zhu et al. (2019), Sportisse et al. (2020a), Sportisse et al. (2020b), Wang et al. (2020), Yang et al. (2021), Bhattacharya and Chatterjee (2021), and Agarwal et al. (2021) are some of the several works in the literature which have considered the MNAR regime. Thus our work serves as a unification of these two domains of research.

Our contributions The main finding of our work is that not only does the two-sided nearest neighbor method adapt to the smoothness of the latent function f but it also attains the minimax optimal non-parametric rate in an intermediate regime. In other words, even without the prior knowledge of the latent factors, the performance of the TS-NN is as good as the oracle algorithm which has access to all the latent factors. Our analysis also shows that the TS-NN algorithm is robust to the missingness pattern of the matrix and can yield minimax optimal rate even when some entries of the matrix are missing deterministically. Our work contributes to the growing literature on understanding properties/robustness of nearest neighbor methods and handling missing data problems with non-smooth data and deterministic missingness.

Organization We start with the description of the model and various underlying assumptions in Sec. 2. In Sec. 3, we review the two-sided nearest neighbor algorithm. We discuss the theoretical guarantees of the performance of the method in Sec. 4 and support the theoretical results with extensive simulation studies and real data analysis in Sec. 5.

Notations We denote the set $\{1, \cdots, n\}$ by $[n]$. We use $a_n = O(b_n)$ or $a_n \ll b_n$ to imply that there exists a constant $c > 0$ such that $a_n \le cb_n$. We use $a_n = o(b_n)$ to imply that $a_n/b_n \to 0$ as $n \to \infty$. We use $a_n = \Omega(b_n)$ to mean $b_n = O(a_n)$ and $a_n = \omega(b_n)$ to mean $b_n = o(a_n)$. The notation $a_n = \Theta(b_n)$ is used when both $a_n = O(b_n)$ and $a_n = \Omega(b_n)$ hold true. We use $\mathcal{U}$ to denote the set of row latent factors and $\mathcal{V}$ to denote the set of column latent factors. In our results, we use c to denote universal constant (independent of m, n, model parameters), that might take a different value in every appearance.

2 PROBLEM SET-UP

We have the data matrix $X \in \mathbb{R}^{n\times m}$ coming from the data-generating model (1). The objective is to estimate the ground truth matrix Θ given the data matrix. We make the following assumptions regarding the data-generating mechanism and the structure of the ground truths $\theta_{i,j}$ for all $(i,j) \in [n] \times [m]$.

Assumption 1 (Non-linear factor model)**.** *Conditioned on the latent factors $u_1, \cdots, u_n$ and $v_1, \cdots, v_m$ the ground truth has the following low-rank representation,*

$$\theta_{i,j} = f(u_i, v_j) \quad \forall \quad (i,j) \in [n] \times [m],$$

where f is (λ, L) Hölder function for $\lambda \in (0,1]$ i.e. for $x, x' \in \mathrm{Domain}(f)$,

$$|f(x) - f(x')| \le L||x - x'||_\infty^\lambda.$$

Assum. 1 describes the non-parametric model where the ground truth matrix Θ is described in terms of the latent factors $\mathcal{U}, \mathcal{V}$ using the latent function f. This allows for a potentially non-linear relationship between the row and column latent factors.

Assumption 2 (Sub-Gaussian noise)**.** *The noise terms $\{\epsilon_{i,j}\}$ are independent of each other, the latent factors, and the missingness indicators $\{A_{i,j}\}$. Moreover $\{\epsilon_{i,j}\}$ are sub-Gaussian random variables with $\mathbb{E}[\epsilon_{i,j}] = 0$, $\mathrm{Var}(\epsilon_{i,j}) = \sigma^2$.*

Assumption 3 (Row and column latent factors)**.** *The row latent factors $u_1, \cdots, u_n$ are sampled independently from $\mathrm{Uniform}[0,1]^{d_1}$. The column latent factors $v_1, \cdots, v_m$ are sampled independently from $\mathrm{Uniform}[0,1]^{d_2}$.*

Assum. 3 is made for the ease of presentation. The assumption of sampling the row and column latent factors from the unit hypercubes can be easily relaxed to any compact set in d_1 and d_2 dimensions (respectively). Moreover the entire analysis can be done for any arbitrary sampling distribution if we replace the tail bounds of the uniform distribution with that of the arbitrary distribution.

We note that if both Assum. 1 and Assum. 3 hold, then the latent function f is bounded as it is well known that a Hölder-continuous function on a compact domain is always bounded. In all the results that we discuss in this paper both these assumptions are required to hold true and thus we assume that $|f(x)| \leq M$ for all $x \in \mathrm{Domain}(f)$.

3 ALGORITHM

We now describe the two-sided nearest neighbor (TS-NN) algorithm in this section. To approximate the L_2 distance in the latent functional space we consider the following oracle distance:

$$d^2_{row}(i,i') = \frac{1}{m}\sum_{j\in[m]}(f(u_i,v_j) - f(u_{i'},v_j))^2,$$
$$d^2_{col}(j,j') = \frac{1}{n}\sum_{i\in[n]}(f(u_i,v_j) - f(u_i,v_{j'}))^2,$$

for all $i, i' \in [n]$ and for all $j, j' \in [m]$. Here $d^2_{row}(i,i')$ serves as a proxy for the distance between the row latent factors $u_i, u_{i'}$. Similarly $d^2_{col}(j,j')$ serves as proxy for the distance between the column latent factors $v_j, v_{j'}$. However since the the latent function f is unknown it is not possible to exactly compute these distances between the rows and the columns. Therefore we use the observed entries of the matrix X to approximate the distances $d^2_{row}(i,i')$ and $d^2_{col}(j,j')$ via the following data-driven analogues:

$$\widehat{d}^2_{row}(i,i') = \frac{\sum_{j\in[m]}(X_{i,j} - X_{i',j})^2 A_{i,j}A_{i',j}}{\sum_{j\in[m]} A_{i,j}A_{i',j}} - 2\sigma^2,$$
$$\widehat{d}^2_{col}(j,j') = \frac{\sum_{i\in[n]}(X_{i,j} - X_{i,j'})^2 A_{i,j}A_{i,j'}}{\sum_{i\in[n]} A_{i,j}A_{i,j'}} - 2\sigma^2.$$

Given the distances above, the two-sided nearest neighbor algorithm with tuning parameters $\boldsymbol{\eta} = \{\eta_{row}, \eta_{col}\}$ (TS-NN($\boldsymbol{\eta}$)) consists of the following steps:

1. Compute the pairwise row and column distance estimates $\widehat{d}^2_{row}(i,i')$ and $\widehat{d}^2_{col}(j,j')$ for all $i, i' \in [n]$ and for all $j, j' \in [m]$ and use those to construct the following neighborhoods,
$$\mathcal{N}_{row}(i) = \{i' \in [n] : \widehat{d}^2_{row}(i,i') \leq \eta^2_{row}\}, \qquad (2)$$
$$\mathcal{N}_{col}(j) = \{j' \in [m] : \widehat{d}^2_{col}(j,j') \leq \eta^2_{col}\}.$$
2. Average the outcomes across the the two sets of neighbors:
$$\widehat{\theta}_{i,j} = \frac{\sum_{i'\in\mathcal{N}_{row}(i);j'\in\mathcal{N}_{col}(j)} X_{i',j'}A_{i',j'}}{|\mathcal{N}_{row,col}(i,j)|},$$
where $\mathcal{N}_{row,col}(i,j) = \{(i',j') | i' \in \mathcal{N}_{row}(i),\ j' \in \mathcal{N}_{col}(j),\ A_{i',j'} = 1\}$.

We note that the variance of the noise terms, σ^2, in the definition of $\widehat{d}^2_{row}(i,i'), \widehat{d}^2_{col}(j,j')$, is without loss of generality and only to simplify the algebraic expressions henceforth. In practice, this term is not used and one can verify below that the algorithm is unaffected if we remove $2\sigma^2$ from the display above and replace $\eta^2_{row}, \eta^2_{col}$ with $\eta^2_{row} - 2\sigma^2, \eta^2_{col} - 2\sigma^2$.

4 THEORETICAL GUARANTEES

In this section, we present our main results that characterize the performance of the two-sided nearest neighbor algorithm under the assumptions discussed in Sec. 2. The main error metric in this work is the mean-squared-error (MSE) of the estimates $\widehat{\theta}_{i,j}$:

$$\mathrm{MSE} := \frac{1}{mn}\sum_{i\in[n],j\in[m]}\left(\widehat{\theta}_{i,j} - f(u_i,v_j)\right)^2. \qquad (3)$$

We first discuss in Sec. 4.1 the behavior of the MSE of the two-sided nearest neighbor method under the simpler setting where $A_{i,j} \overset{iid}{\sim} \mathrm{Ber}(p)$ for some $0 < p \leq 1$. This is the MCAR assumption mentioned in Assum. 4. Thereafter in Sec. 4.2 we discuss the performance of the two-sided nearest neighbor algorithm under a more general setting where we show that the algorithm stays minimax optimal even if there is arbitrary/deterministic missingness in some entries of the matrix.

4.1 Missing completely at random (MCAR)

Our first result provides a guarantee when the missingness is independent of the underlying means, a setting referred to as MCAR in the matrix completion and causal inference literature.

Assumption 4 (MCAR missingness)**.** *The indicators $A_{i,j}$ are drawn i.i.d. Ber(p), and independently of the latent factors and the noise.*

As highlighted in prior works, MCAR assumption, while rare in practice, provides an initial understanding of the algorithm's effectiveness as a function of the amount of missingness (captured by a single parameter p in MCAR) and the factors and noise distributions. We are now ready to state our guarantee.

Theorem 1. *Under Assum. 1 to 4 and for any fixed $\delta \in (0,1)$, the MSE of TS-NN($\boldsymbol{\eta}$) satisfies*

$$\mathrm{MSE} \leq c_{0,\delta}\Bigg(\eta_{row}^2 + \eta_{col}^2 + \frac{c}{\sqrt{m}} + \frac{c}{\sqrt{n}} + \frac{c_{1,\delta}\sigma^2 L^{(d_1+d_2)/\lambda}}{pmn\Big(\eta_{row}^2 - \frac{c}{\sqrt{m}}\Big)^{\frac{d_1}{2\lambda}}\Big(\eta_{col}^2 - \frac{c}{\sqrt{n}}\Big)^{\frac{d_2}{2\lambda}}}\Bigg),$$

with probability at least $1-\delta$, where $c_{0,\delta} = \frac{c(1+\delta/7)}{(1-\delta/7)^2}$ and $c_{1,\delta} = \frac{c\log(\frac{14}{\delta})}{(1-\delta/7)}$.

Thm. 1 provides an explicit upper bound (with a high probability) on the mean-squared error of the two-sided nearest neighbor algorithm. The proof of Thm. 1 is discussed in App. A. In order to get superior MSE decay rates, we optimize the above upper bound with respect to $\boldsymbol{\eta}$. An immediate consequence of this exercise is the following corollary.

Corollary 1. *Under Assum. 1 to 4 for $n = \omega(\max\{m^{\frac{d_1}{2\lambda+d_2}}, m^{\frac{4\lambda}{d_1+d_2-2\lambda}}\})$ and $n = O(\min\{m^{\frac{2\lambda+d_1}{d_2}}, m^{\frac{d_1+d_2-2\lambda}{4\lambda}}\})$, TS-NN($\boldsymbol{\eta}$) with $\eta_{row} = \eta_{col} = \Theta((mn)^{\frac{-\lambda}{2\lambda+d_1+d_2}})$ achieves the non-parametric minimax optimal rate,*

$$\mathrm{MSE} = O\Big((mn)^{\frac{-2\lambda}{2\lambda+d_1+d_2}}\Big).$$

The proof of Cor. 1 is provided in App. B. When the dimension of the row and the column latent space are equal (i.e. $d_1 = d_2 = d$), Cor. 1 implies that the two-sided nearest neighbor algorithm with $\eta_{row} = \eta_{col} = \Theta((mn)^{-\frac{\lambda}{2(\lambda+d)}})$ achieves the minimax optimal rate in the moderate regime of $n = \omega(\max\{m^{\frac{d}{2\lambda+d}}, m^{\frac{4\lambda}{2(d-\lambda)}}\})$ and $n = O(\min\{m^{\frac{2\lambda+d}{d}}, m^{\frac{2(d-\lambda)}{4\lambda}}\})$ (for the lipschitz case ($\lambda = 1$) this regime becomes $n = \omega(\max\{m^{\frac{d}{2+d}}, m^{\frac{4}{2(d-1)}}\})$ and $n = O(\min\{m^{\frac{2+d}{d}}, m^{\frac{2(d-1)}{4}}\})$).

Cor. 1 states that it is possible to achieve the optimal minimax non-parametric rate of the oracle algorithm in a moderate regime when $n = \omega(\max\{m^{\frac{d_1}{2\lambda+d_2}}, m^{\frac{4\lambda}{d_1+d_2-2\lambda}}\})$ and $n = O(\min\{m^{\frac{2\lambda+d_1}{d_2}}, m^{\frac{d_1+d_2-2\lambda}{4\lambda}}\})$. Cor. 1 implies that in the Lipschitz case for $\lambda = 1$, $d_1 = d_2$ and $m = n$ the MSE of two-sided nearest neighbor achieves the rate $O(n^{-\frac{2}{d+1}})$ in a certain intermediate regime. Under the same setting Yu (2022)'s NN estimator achieves the same rate in an intermediate regime using the additional knowledge of column latent factors. This rate is better than the MSE rate of $O(n^{-\frac{2}{d+2}})$ achieved by the row nearest neighbor method under the same setting. However under some additional convexity assumptions doubly-robust nearest neighbor (Dwivedi et al. (2022b)) achieves the MSE rate of $O(n^{-\frac{4}{d+4}})$ in $\lambda = 1$ case which is better than that of the two-sided nearest neighbor .

We note that theoretically, the performance (measured in terms of MSE) of TS-NN algorithm is never worse than the row or column nearest neighbor counterparts. This is because one can choose η_{col} small enough so that $\mathcal{N}_{col}(j) = \{j\}$ for all $j \in [m]$. Then the two-sided nearest neighbor method simplifies to the row nearest neighbor algorithm. Similarly by choosing η_{row} small enough TS-NN recovers the column NN algorithm.

4.2 Missing not at random (MNAR)

The minimax optimality of the two-sided nearest neighbor algorithm holds in much more general missingness patterns than the MCAR setup. To formalize this claim, we introduce our next assumption.

Assumption 5 (MNAR missingness)**.** *Conditioned on the latent factors $\mathcal{U}, \mathcal{V}$, the indicators $A_{i,j}$ are drawn from Ber$(p_{i,j})$ independent of each other and independent of all other randomness.*

Note that such an assumption, used in prior works like Agarwal et al. (2021), allows the missingness to depend on unobserved latent factors, and thus falls under the category of missing not at random.

Next we require a sufficient condition on the number of neighbors. Recall the definitions of $\mathcal{N}_{row}(i), \mathcal{N}_{col}(j), \mathcal{N}_{row,col}(i,j)$ discussed in Steps 1, 2 of the TS-NN($\boldsymbol{\eta}$) algorithm in Sec. 3.

Assumption 6 (Minimum number of nearest neighbors)**.** *There exists a function $g : (0,1] \to \mathbb{R}_+$ such that for any $\delta \in [0,1)$ the event E_δ defined as*

$$E_\delta = \bigcap_{(i,t)\in[n]\times[m]} \left\{ \frac{|\mathcal{N}_{row,col}(i,j)|}{|\mathcal{N}_{row}(i)||\mathcal{N}_{col}(j)|} \geq g(\delta) \right\},$$

satisfies $\mathbb{P}(E_\delta | \mathcal{U}, \mathcal{V}) \geq 1-\delta$.

Assum. 6 essentially guarantees the presence of a certain minimum number of nearest neighbors, which in turn aids in carrying out valid statistical inference.

The following theorem discusses the performance of the algorithm in the general setup.

Theorem 2. *Under Assum. 1 to 3, 5, and 6 for any fixed $\delta \in (0,1)$, the MSE of TS-NN($\boldsymbol{\eta}$) satisfies,*

$$\mathrm{MSE} \leq c'_{0,\delta}\Bigg(\eta_{row}^2 + \eta_{col}^2 + \frac{c}{\sqrt{m}} + \frac{c}{\sqrt{n}} + \frac{c_{1,\delta}\sigma^2 L^{(d_1+d_2)/\lambda}}{mn\Big(\eta_{row}^2 - \frac{c}{\sqrt{m}}\Big)^{\frac{d_1}{2\lambda}}\Big(\eta_{col}^2 - \frac{c}{\sqrt{n}}\Big)^{\frac{d_2}{2\lambda}}}\Bigg).$$

with probability at least $1-\delta$, where $c'_{0,\delta} = \frac{c}{g(\delta)(1-\delta/7)}$ and $c_{1,\delta} = \frac{c\log(\frac{14}{\delta})}{(1-\delta/7)}$.

The proof of Thm. 2 is discussed in App. C. To our knowledge, no theoretical analysis exists for any method when both the row and column latent factors are unknown, the latent function is non-linear, and (λ, L) Hölder-continuous with $\lambda < 1$ (i.e., the function is not Lipschitz), and the missingness is not at random. To this end, Thm. 2 is a first result of its kind.

As before, if we optimize the upper bound in Thm. 2 with respect to $\boldsymbol{\eta}$ to obtain explicit MSE decay rates. When $g(\delta) \geq c$, we can once again deduce that for $n = \omega(\max\{m^{\frac{d_1}{2\lambda+d_2}}, m^{\frac{4\lambda}{d_1+d_2-2\lambda}}\})$ and $n = O(\min\{m^{\frac{2\lambda+d_1}{d_2}}, m^{\frac{d_1+d_2-2\lambda}{4\lambda}}\})$, TS-NN($\boldsymbol{\eta}$) with $\eta_{row} = \eta_{col} = \Theta((mn)^{\frac{-\lambda}{2\lambda+d_1+d_2}})$ achieves the minimax rate $\mathrm{MSE} = O\Big((mn)^{\frac{-2\lambda}{2\lambda+d_1+d_2}}\Big)$.

Let us now illustrate a few examples where Assum. 6 is satisfied. First note that if $p_{i,j} \geq p > 0$ for all $(i,j) \in [n] \times [m]$. Then using Chernoff bound, we can show that conditioned on the latent factors, $|\mathcal{N}_{row,col}(i,j)| \geq (1-\delta)p|\mathcal{N}_{row}(i)||\mathcal{N}_{col}(j)|$ holds for all $(i,j) \in [n] \times [m]$ with a high probability. Thus Assum. 6 is satisfied with $g(\delta) = (1-\delta)p$ and hence TS-NN($\boldsymbol{\eta}$) achieves minimax optimal rate in an intermediate regime. Notably, this setting recovers the guarantee of Cor. 1 where $p_{i,j} = p$ as a special case.

More generally, note that

$$|\mathcal{N}_{row,col}(i,j)| \geq (1-\delta) \sum_{i' \in \mathcal{N}_{row}(i), j' \in \mathcal{N}_{col}(j)} p_{i',j'}$$

with high probability (follows from the concentration of a sum of weighted Bernoulli random variables (Lemma 2, Dwivedi et al. (2022b)). Hence, if

$$\sum_{i' \in \mathcal{N}_{row}(i), j' \in \mathcal{N}_{col}(j)} p_{i',j'} \geq c|\mathcal{N}_{row}(i)||\mathcal{N}_{col}(j)|, \tag{4}$$

holds with a high probability for some constant c, Assum. 6 shall hold and we can apply Thm. 2 to guarantee the minimax optimality of the TS-NN($\boldsymbol{\eta}$) algorithm. The condition (4) is pretty general and can arise in many settings. For instance, suppose there are underlying iid variables $B_{i,j} \sim \mathrm{Ber}(1/2)$ for all $(i,j) \in [n] \times [m]$, independent of everything else and we have $A_{i,j} = 0$ if $B_{i,j} = 0$, and $A_{i,j} \sim \mathrm{Ber}(p_{i,j})$ if $B_{i,j} = 1$. Suppose $p_{i,j} \geq p > 0$ for all $(i,j) \in [n] \times [m]$ such that $B_{i,j} = 1$. It can be easily checked that with a high probability there exists a subset $S \subset \mathcal{N}_{row}(i) \times \mathcal{N}_{col}(j)$ such that $|S| \geq (|\mathcal{N}_{row}(i)||\mathcal{N}_{col}(j)|)/2$ and $\sum_{i' \in \mathcal{N}_{row}(i), j' \in \mathcal{N}_{col}(j)} p_{i',j'} \geq \sum_{(i',j') \in S} p_{i',j'}$ which is greater than or equal to $p|S| \geq (p/2)|\mathcal{N}_{row}(i)||\mathcal{N}_{col}(j)|$. Thus the required condition is satisfied with $c = (p/2)$. These examples suggest that even with around 50% deterministic missingness ($p_{i,j} = 0$) in the data, we will continue to get optimal results by using TS-NN($\boldsymbol{\eta}$) algorithm. To summarise, as long as conditioned on the latent factors $|\mathcal{N}_{row,col}(i,j)| \geq g(\delta)|\mathcal{N}_{row}(i)||\mathcal{N}_{col}(j)|$ holds for all $(i,j) \in [n] \times [m]$ for some $g(\delta) \in (0,1]$ with a high probability, the minimax optimality of the two-sided nearest neighbor algorithm can be established.

5 EXPERIMENTS

In this section, we illustrate the practical usability of TS-NN to complement our theoretical findings with empirical evidence via two vignettes: one with synthetic data and another a case study with the real-life dataset HeartSteps. All of our tests have been run on a MacBook Pro with an M2 chip and 32 GB of RAM.

5.1 Simulation Study

We use the following $(\lambda, 2)$ Hölder smooth $f : \mathbb{R}^2 \to \mathbb{R}$:

$$f(u,v) = |u+v|^\lambda \mathrm{sgn}(u+v),$$

and generate i.i.d. latent factors and noise as follows:

$$u_i \sim \mathrm{Unif}[-0.5, 0.5]; \quad v_j \sim \mathrm{Unif}[-0.5, 0.5];$$
$$\epsilon \sim \mathcal{N}(0, \sigma_\epsilon^2); \quad \theta_{i,j} = f(u_i, v_j),$$

and simulate two distinct missingness mechanisms:

$$\text{MCAR:} \quad A_{i,j} \overset{iid}{\sim} \mathrm{Ber}(0.75),$$
$$\text{MNAR:} \quad A_{i,j}(u_i, v_j) \sim \mathrm{Ber}(p_{i,j}(u_i, v_j))$$
$$p_{i,j}(u_i, v_j) = \begin{cases} 0 & \text{with prob } 0.2 \\ \frac{2}{5} + \frac{1}{5}\mathbb{I}(u_i + v_j > 0) & \text{with prob } 0.8 \end{cases}$$

The latter formulation, captures an MNAR setting, where data 20% of entries are deterministically missing and for 80% points, larger signals will have a larger probability of being observed. For example, in a movie recommendation system, if a user strongly likes or dislikes a movie, they are more inclined to give a rating than if they have mixed feelings about it.

We present results for a couple of different signal-to-noise ratio in our simulations, which is defined as:

$$\mathrm{SNR} = \sqrt{\frac{\sum_{i=1}^{n}\sum_{j=1}^{m} f^2(u_i, v_j)}{mn\sigma_\epsilon^2}}.$$

Baselines We compare TS-NN with the vanilla NNs, namely row nearest neighbors (Row-NN) and its column counterpart (Col-NN) Li et al. (2019); Dwivedi et al. (2022a) each of which only use one set of neighbors, and then its doubly robust variant (DR-NN) Dwivedi et al. (2022b) which uses both row and column neighbors. We also include the conventional matrix completion methods Universal Singular Value Thresholding (USVT) Chatterjee (2015) and SoftImpute Hastie et al. (2015).

We also provide the performance of Oracle TS-NN (O-TS-NN), which has access to both row and column latent factors but doesn't know the f. It uses the latent vectors in place of observed rows and columns to compute the inter-row and inter-column distances, eventually yielding the neighborhood set.

Experiment Setup We set the number of rows and columns equal, i.e., $m = n$ so that Row and Col-NN perform similarly and we omit one of them. In all NN methods, hyper-parameter tuning is done via cross-validation and SoftImpute is implemented over a log lambda grid and the best MSE was reported (details in App. D.1). USVT is implemented using the `filling` R package. We repeat the experiment 10 times and plot the mean MSE (3) along with 1 standard deviation for it (which can be too small to notice), as a function of n in Fig. 1 for $\lambda = 0.75$ and Fig. 3 in Appendix for $\lambda = 0.5$. We also provide a least squares fit for $\log(\mathrm{MSE})$ with respect to $\log n$ and report the slope of the regression line in the plot. If the slope of the line is -0.9 then, MSE decreases in the order of $n^{-0.9}$. We show it for quantifying the MSE decay of algorithms in simulation studies.

Results Following our Cor. 1 and Thm. 2, the theoretical MSE decay rate of TS-NN becomes $\mathcal{O}\left(n^{-\frac{4\lambda}{2\lambda+2}}\right)$ in setup where $n = m$ and $d_1 = d_2 = 1$. We note that the empirical MSE decay rates of TS-NN are better than the theoretical ones. TS-NN and its oracle version O-TS-NN show the best MSE decay rates among all the baseline algorithms. Infact in the MCAR setup, TS-NN shows similar MSE decay rate as O-TS-NN for both the lambdas 0.75 and 0.5.

Overall, TS-NN and DR-NN perform better than the one-sided counterparts. On the other hand, USVT and SoftImpute exhibit no-to-weak MSE decay. We observe that only TS-NN maintains its non-trivial error decay while transitioning from MCAR to MNAR setup. While DR-NN is competitive to TS-NN for MCAR, it fairs much worse for MNAR. We highlight that TS-NN exhibits the minimum MSE among all the baselines in both settings for all $n \geq 100$.

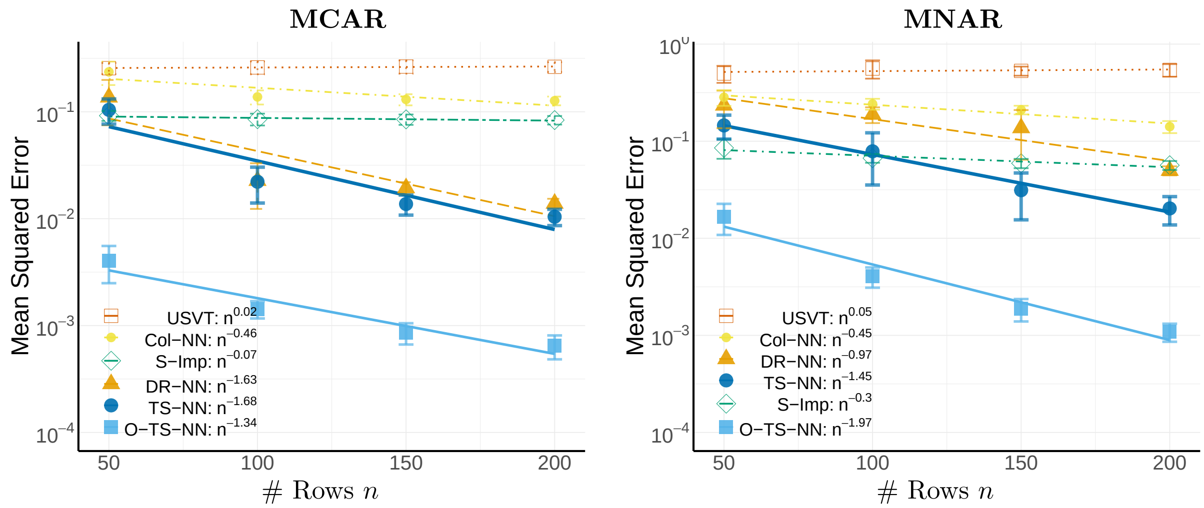
Sensitivity to Smoothness of latent function Next, we verify the adaptivity of TS-NN to the smoothness of the underlying signal (quantified by λ), an implication of Thm. 2 in a high SNR regime to highlight the change in decay rates in low-sample size. (Note when the SNR is not too high, the effect is not as pronounced, e.g., when we change $\lambda = 0.75$ in Fig. 1(a) to $\lambda = 0.5$ in Fig. 3.)

Overall, we see an improvement in the estimation accuracy of TS-NN as the smoothness of f increases. Interestingly in MCAR, we obtain empirical MSE decay rates of $n^{-0.79}, n^{-0.93}$ and $n^{-1.05}$ at $\lambda = 0.5, 0.75$ and 1; the trends is consistent with the theoretical rates of $n^{-0.67}, n^{-0.86}$ and n^{-1} respectively. Theoretical rates are obtained by plugging the λ in $\mathcal{O}\left(n^{-\frac{4\lambda}{2\lambda+2}}\right)$. Even in MNAR, at $\lambda = 0.6, 0.8$ and 1 we get empirical MSE decay rates of $n^{-0.79}, n^{-0.87}$ and n^{-1} where the theoretical rates are $n^{-0.75}, n^{-0.89}$ and n^{-1} respectively.

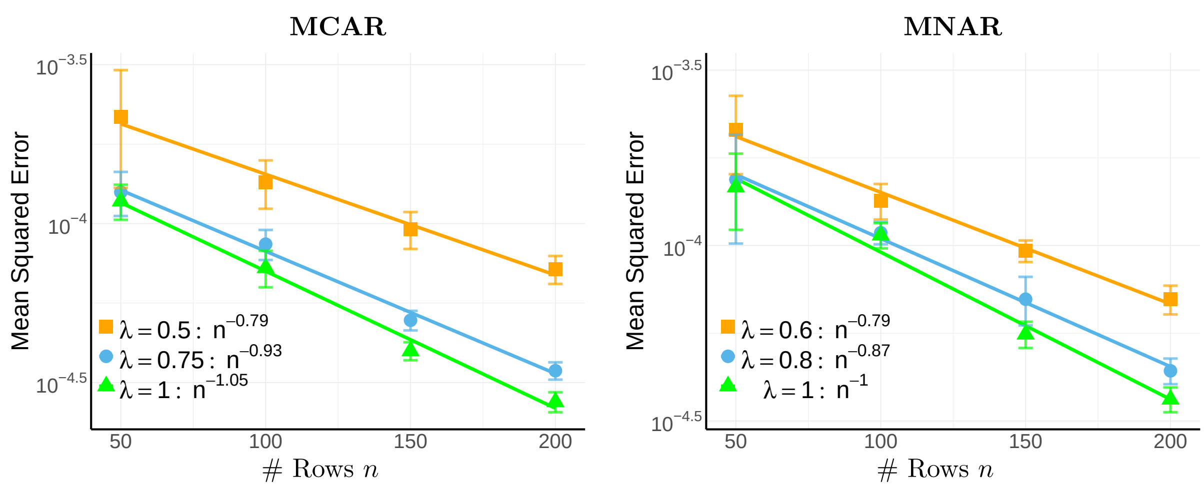
Overall, we observe that TS-NN's performance is adaptive with respect to the smoothness parameter λ and the missingness mechanism being MCAR / MNAR, consistent with our theory, namely Cor. 1 and Thm. 2. Our simulations provide evidence of the resilience of TS-NN towards arbitrary missingness/intervention patterns while delivering MSE-decay rates similar to that of the oracle-TS-NN. (However, as expected the oracle's performance is strictly better (primarily in terms of constant scaling factors), compared to TS-NN.)

5.2 Case-study with HeartSteps

HeartSteps Klasnja et al. (2019) is a micro-randomized trial aimed at improving participants' walking activity via mobile notifications, resulting in a health intervention dataset spanning 6 weeks with 37 users. Klasnja et al. (2019) looked for healthy sedentary adults who intended to improve their fitness and walking. They did the recruitment in between August 2015 and January 2016 via fliers and facebook ads. Selected applicants were invited to an interview and were provided with a Jawbone tracker and HeartSteps app installed on their phones for tracking physical activity pre and post-intervention. The underlying recommender sys-



(a) MSE error rates for various algorithms for $\lambda = 0.75$-smooth function and SNR ≈ 1.41.



(b) Variations in TS-NN MSE with smoothness parameter λ for SNR ≈ 31.

Figure 1: **MSE comparison for various benchmarks.** Results are averaged across 10 runs.

tem could send notifications (driven by user's context) to user's phone up to five times a day, at user-specific times. For every decision point when the participant is available(refer to App. D.2 for details), delivery of the notification was randomized with the following probabilities: 0.4 no notification, 0.3 walking suggestion, and 0.3 anti-sedentary suggestion. This means there is a 60% chance of sending a notification to a HeartSteps participant at each decision time. The final quantity of interest is the log of the user's step counts within 30 minutes after the decision point.

For us, a matrix of interest will have rows denoting users, columns denoting decision time(for a randomized decision), and entries will be log(step counts) after a certain intervention. The intervention is whether a notification (walking or anti-sedentary) was sent or not sent, resulting in 2 types of intervention. We will focus on one intervention, say "notification sent" to understand the generality of missing matrix completion setups that Assum. 6 encapsulates. If a user is available at a decision time point, they receive a notification with a probability of 0.6. Also, a user is available on average around 80% of the decision times. So, we have around 20% deterministic missingness in the matrix corresponding to the intervention "notification sent". This exactly matches the prescription of the example in Sec. 4.2 satisfying Assum. 6 even with deterministic missingness. Following the same chain of arguments, we have condition (4) satisfied over here with $c = 0.6/2 = 0.3$. Therefore, Assum. 6 holds over here, making the HeartSteps a MNAR dataset which can be tackled by TS-NN.

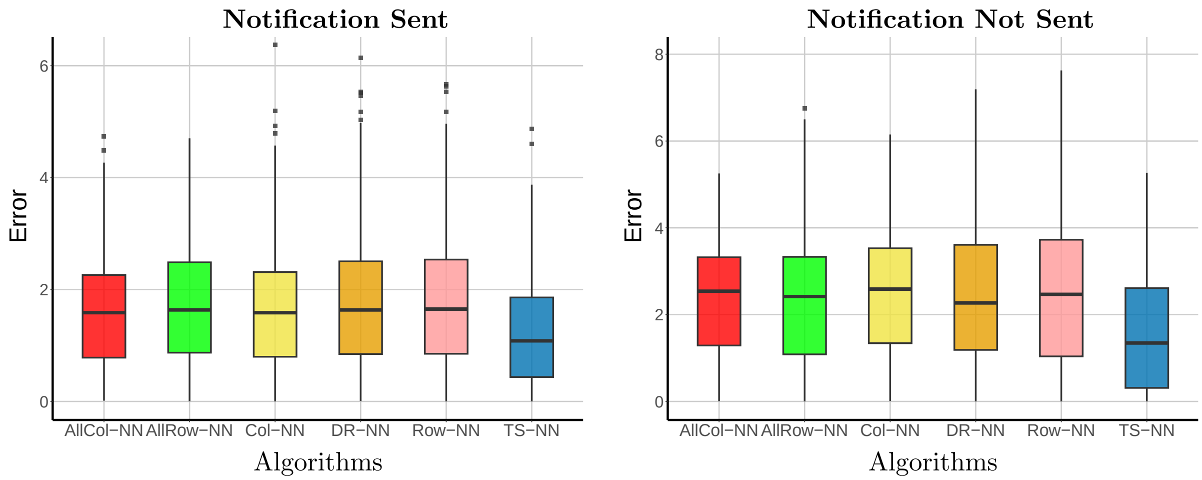


Figure 2: **Held-out data root MSE (RMSE) for various algorithms in HeartSteps.** RMSE of USVT and SoftImpute were too high to be competitive and are omitted for plot clarity.

We aim to estimate the counterfactual of both the interventions on each user at every time point, with HeartSteps data from this Github repository. At a macro-level, we want to do matrix completion of the 2 missing matrices arising due to different interventions. We work with 37×210 matrix as majority of the users did not experience more than 210 decision times due to availability issues. Also in this real-life case study, we ignore all the context information available in the HeartSteps, which reduces the SNR. We treat this experiment as an empirical demonstration of the usefulness of our methodology for estimation. Since true underlying counterfactuals are unknown to us, we use a 5 fold blocked cross-validation approach to evaluate all the algorithms. We divide the rows/users of the matrix into 5 folds, and in each fold of CV, we hold out the entries in the last 40 decision times of the rows in that particular fold as our test dataset. The remaining entries are used to train our TS-NN and other benchmark algorithms.

For NN based algorithms, we do not allow an entry to be its self-neighbor (since we are comparing the estimate to the observed entry). After training, the differences between the entries of test dataset and their estimates were recorded and presented as a boxplot in Fig. 2. Both USVT and SoftImpute performed poorly and are dropped from the figure for clarity. For an alternative baseline, we took the simple “allRow-NN” (and “allCol-NN”) which basically takes all the available rows (and columns) as neighbors. Fig. 2 shows how well different NN strategies perform in estimating the held-out test dataset. The small size of the dataset (37 rows and 210 columns) is causing problems for one-sided NNs to show better results than the baseline counterparts. One-sided NNs had a proclivity towards smaller neighborhoods during training which performed poorly during test matrix prediction. Only TS-NN is convincingly beating the baseline algorithms allRow-NN and allCol-NN, with the best median error and least amount of error spread.

6 DISCUSSION

We have studied the performance of the two-sided nearest neighbor in the setting where the latent function f is Hölder smooth and the row and column latent factors are unknown. We have seen that it is possible to achieve the optimal minimax non-parametric rate of the oracle algorithm using the two-sided nearest neighbor in certain scalings of rows and columns for a wide range of MCAR and MNAR missingness. Thus the error rate of TS-NN does not suffer from the lack of knowledge of row and column latent factors. The simulations and the real data analysis support the theoretical guarantee derived in this work.

In this work, we analyzed the adaptivity for functions less smooth than Lipschitz. In some settings, functions might have higher-order smoothness (when f belongs in a smooth reproducing kernel Hilbert space). Analyzing whether nearest neighbors adapt to the model smoothness in such settings is an interesting venue for future work.

We assumed independence of row/column latent factors and the exogenous noise, both which are often violated in real life. For example, in movie recommendation systems, a user’s perception and rating are both susceptible to their peer group’s preference. In other causal panel data settings, the columns of the matrix denote time and the missingness is dependent on the assigned treatments. For settings with such network interference or dependence over time arises due to spillover effects or due to sequentially assigned treatments, designing a correctly adjusted TS-NN is another interesting direction.

References

Anish Agarwal, Devavrat Shah, and Dennis Shen. Synthetic interventions. *arXiv preprint arXiv:2006.07691*, 2020. (Cited on page 1.)

Anish Agarwal, Munther Dahleh, Devavrat Shah, and Dennis Shen. Causal matrix completion. *arXiv preprint arXiv:2109.15154*, 2021. (Cited on pages 1, 2, and 4.)

Dmitry Arkhangelsky, Susan Athey, David A Hirshberg, Guido W Imbens, and Stefan Wager. Synthetic difference in differences. Technical report, National Bureau of Economic Research, 2019. (Cited on page 1.)

Vidmantas Bentkus. On hoeffding's inequalities. *Annals of probability*, pages 1650–1673, 2004. (Cited on pages 14, 17, and 19.)

Dimitris Bertsimas and Michael Lingzhi Li. Fast exact matrix completion: A unified optimization framework for matrix completion. *Journal of Machine Learning Research*, 21(231):1–43, 2020. (Cited on page 1.)

Sohom Bhattacharya and Sourav Chatterjee. Matrix completion with data-dependent missingness probabilities. *arXiv preprint arXiv:2106.02290*, 2021. (Cited on page 2.)

M Burkina, Ivan Nazarov, Maxim Panov, G Fedonin, and B Shirokikh. Inductive matrix completion with feature selection. *Computational Mathematics and Mathematical Physics*, 61(5):719–732, 2021. (Cited on page 1.)

Sourav Chatterjee. Matrix estimation by universal singular value thresholding. *The Annals of Statistics*, 43(1):177–214, 2015. (Cited on page 6.)

Kai-Yang Chiang, Cho-Jui Hsieh, and Inderjit S Dhillon. Matrix completion with noisy side information. *Advances in neural information processing systems*, 28, 2015. (Cited on page 1.)

Kai-Yang Chiang, Inderjit S Dhillon, and Cho-Jui Hsieh. Using side information to reliably learn low-rank matrices from missing and corrupted observations. *Journal of Machine Learning Research*, 19 (76):1–35, 2018. (Cited on page 1.)

Raaz Dwivedi, Katherine Tian, Sabina Tomkins, Predrag Klasnja, Susan Murphy, and Devavrat Shah. Counterfactual inference for sequential experiments. *arXiv preprint arXiv:2202.06891*, 2022a. (Cited on pages 2 and 6.)

Raaz Dwivedi, Katherine Tian, Sabina Tomkins, Predrag Klasnja, Susan Murphy, and Devavrat Shah. Doubly robust nearest neighbors in factor models. *arXiv preprint arXiv:2211.14297*, 2022b. (Cited on pages 2, 4, 5, 6, 18, and 22.)

Armin Eftekhari, Dehui Yang, and Michael B Wakin. Weighted matrix completion and recovery with prior subspace information. *IEEE Transactions on Information Theory*, 64(6):4044–4071, 2018. (Cited on page 1.)

Chao Gao, Yu Lu, and Harrison H Zhou. Rate-optimal graphon estimation. 2015. (Cited on page 2.)

Chao Gao, Yu Lu, Zongming Ma, and Harrison H Zhou. Optimal estimation and completion of matrices with biclustering structures. *Journal of Machine Learning Research*, 17(161):1–29, 2016. (Cited on page 2.)

Mohsen Ghassemi, Anand Sarwate, and Naveen Goela. Global optimality in inductive matrix completion. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2226–2230. IEEE, 2018. (Cited on page 1.)

Yuhong Guo. Convex co-embedding for matrix completion with predictive side information. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 31, 2017. (Cited on page 1.)

Torben Hagerup and Christine Rüb. A guided tour of chernoff bounds. *Information processing letters*, 33 (6):305–308, 1990. (Cited on pages 14, 15, and 16.)

Trevor Hastie, Rahul Mazumder, Jason D. Lee, and Reza Zadeh. Matrix completion and low-rank svd via fast alternating least squares. *Journal of Machine Learning Research*, 16(104):3367–3402, 2015. URL `http://jmlr.org/papers/v16/hastie15a.html`. (Cited on page 6.)

Prateek Jain and Inderjit S Dhillon. Provable inductive matrix completion. *arXiv preprint arXiv:1306.0626*, 2013. (Cited on page 1.)

Predrag Klasnja, Shawna Smith, Nicholas J Seewald, Andy Lee, Kelly Hall, Brook Luers, Eric B Hekler, and Susan A Murphy. Efficacy of contextually tailored suggestions for physical activity: a micro-randomized optimization trial of heartsteps. *Annals of Behavioral Medicine*, 53(6):573–582, 2019. (Cited on pages 6 and 23.)

Olga Klopp, Alexandre B Tsybakov, and Nicolas Verzelen. Oracle inequalities for network models and sparse graphon estimation. 2017. (Cited on page 2.)

Yihua Li, Devavrat Shah, Dogyoon Song, and Christina Lee Yu. Nearest neighbors for matrix estimation interpreted as blind regression for latent variable model. *IEEE Transactions on Information Theory*, 66(3):1760–1784, 2019. (Cited on pages 2 and 6.)

Jin Lu, Guannan Liang, Jiangwen Sun, and Jinbo Bi. A sparse interactive model for matrix completion with side information. *Advances in neural information processing systems*, 29, 2016. (Cited on page 1.)

Wei Ma and George H Chen. Missing not at random in matrix completion: The effectiveness of estimating missingness probabilities under a low nuclear norm assumption. *arXiv preprint arXiv:1910.12774*, 2019. (Cited on page 2.)

Rahul Mazumder, Trevor Hastie, and Robert Tibshirani. Spectral regularization algorithms for learning large incomplete matrices. *Journal of Machine Learning Research*, 11(80):2287–2322, 2010. URL http://jmlr.org/papers/v11/mazumder10a.html. (Cited on page 22.)

Tobias Schnabel, Adith Swaminathan, Ashudeep Singh, Navin Chandak, and Thorsten Joachims. Recommendations as treatments: Debiasing learning and evaluation. In Maria Florina Balcan and Kilian Q. Weinberger, editors, *Proceedings of The 33rd International Conference on Machine Learning*, volume 48 of *Proceedings of Machine Learning Research*, pages 1670–1679, New York, New York, USA, 20–22 Jun 2016. PMLR. URL http://proceedings.mlr.press/v48/schnabel16.html. (Cited on page 2.)

Dogyoon Song, Christina E Lee, Yihua Li, and Devavrat Shah. Blind regression: Nonparametric regression for latent variable models via collaborative filtering. *Advances in Neural Information Processing Systems*, 29, 2016. (Cited on page 2.)

Aude Sportisse, Claire Boyer, and Julie Josse. Imputation and low-rank estimation with missing not at random data. *Statistics and Computing*, 30(6): 1629–1643, 2020a. (Cited on page 2.)

Aude Sportisse, Claire Boyer, and Julie Josses. Estimation and imputation in probabilistic principal component analysis with missing not at random data. *Advances in Neural Information Processing Systems*, 33, 2020b. (Cited on page 2.)

Yixin Wang, Dawen Liang, Laurent Charlin, and David M Blei. Causal inference for recommender systems. In *Proceedings of the 14th ACM Conference on Recommender Systems*, pages 426–431, 2020. (Cited on page 2.)

Jiaming Xu. Rates of convergence of spectral methods for graphon estimation. In *International Conference on Machine Learning*, pages 5433–5442. PMLR, 2018. (Cited on page 2.)

Miao Xu, Rong Jin, and Zhi-Hua Zhou. Speedup matrix completion with side information: Application to multi-label learning. *Advances in neural information processing systems*, 26, 2013. (Cited on page 1.)

Chengrun Yang, Lijun Ding, Ziyang Wu, and Madeleine Udell. Tenips: Inverse propensity sampling for tensor completion. *arXiv preprint arXiv:2101.00323*, 2021. (Cited on page 2.)

Christina Lee Yu. Nonparametric matrix estimation with one-sided covariates. In *2022 IEEE International Symposium on Information Theory (ISIT)*, pages 892–897. IEEE, 2022. (Cited on pages 2 and 4.)

Kai Zhong, Prateek Jain, and Inderjit S Dhillon. Efficient matrix sensing using rank-1 gaussian measurements. In *Algorithmic Learning Theory: 26th International Conference, ALT 2015, Banff, AB, Canada, October 4-6, 2015, Proceedings 26*, pages 3–18. Springer, 2015. (Cited on page 1.)

Ziwei Zhu, Tengyao Wang, and Richard J Samworth. High-dimensional principal component analysis with heterogeneous missingness. *arXiv preprint arXiv:1906.12125*, 2019. (Cited on page 2.)

Checklist

1. For all models and algorithms presented, check if you include:
   (a) A clear description of the mathematical setting, assumptions, algorithm, and/or model. [Yes] look at Sec. 2 and Sec. 3
   (b) An analysis of the properties and complexity (time, space, sample size) of any algorithm. [Yes] Refer to App. D
   (c) (Optional) Anonymized source code, with specification of all dependencies, including external libraries. [Yes] We will provide our code in the supplements.

2. For any theoretical claim, check if you include:
   (a) Statements of the full set of assumptions of all theoretical results. [Yes] Look at Sec. 2, Assum. 5, Assum. 6 and Assum. 4.
   (b) Complete proofs of all theoretical results. [Yes] Look at App. A, App. B and App. C.
   (c) Clear explanations of any assumptions. [Yes] Look at Sec. 2, Assum. 5, Assum. 6 and Assum. 4.

3. For all figures and tables that present empirical results, check if you include:
   (a) The code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL). [Yes] We will provide our code in the supplements.
   (b) All the training details (e.g., data splits, hyperparameters, how they were chosen). [Yes] Refer to Sec. 5.1. Additional details will be provided in the revised appendix.
   (c) A clear definition of the specific measure or statistics and error bars (e.g., with respect to the random seed after running experiments multiple times). [Yes] All the metrics, no of replications, error bars have clearly specified in Sec. 5.1.
   (d) A description of the computing infrastructure used. (e.g., type of GPUs, internal cluster, or cloud provider). [Yes] All of our tests have been run on a MacBook Pro with an M2 chip and 32 GB of RAM.

4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets, check if you include:
   (a) Citations of the creator If your work uses existing assets. [Yes] We have cited the creator of HeartSteps dataset.
   (b) The license information of the assets, if applicable. [Not Applicable]
   (c) New assets either in the supplemental material or as a URL, if applicable. [Not Applicable]
   (d) Information about consent from data providers/curators. [Yes] We have used a publicly available dataset whose github link is also given in the paper.
   (e) Discussion of sensible content if applicable, e.g., personally identifiable information or offensive content. [Not Applicable]

5. If you used crowdsourcing or conducted research with human subjects, check if you include:
   (a) The full text of instructions given to participants and screenshots. [Not Applicable]
   (b) Descriptions of potential participant risks, with links to Institutional Review Board (IRB) approvals if applicable. [Not Applicable]
   (c) The estimated hourly wage paid to participants and the total amount spent on participant compensation. [Not Applicable]

Appendix

A Proof of Thm. 1

For implementing the algorithm we partition the data-set into two subsets and then use one part for learning the row and column distances, and use the other part for generating final predictions. For improving readability, we perform all the computations on a single data-set (without sample splitting) in the proof. However all the computations will continue to hold even if we partition the data-set. From here-on for notational simplicity, we write $f(i,j)$ for $f(u_i, v_j)$, $d^2(i,i')$ for $d^2_{row}(i,i')$ and $d^2(j,j')$ for $d^2_{col}(j,j')$. In the algorithm, we also perform a sub-sampling procedure after picking the full set of nearest neighbors $\mathcal{N}^s_{row}(i)$ and $\mathcal{N}^s_{col}(j)$,

$$\begin{aligned}
\mathcal{N}^s_{row}(i) &= \{i' \in [n] : \widehat{d}^2(i,i') \leq \eta^2_{row}\}, \\
\mathcal{N}^s_{col}(j) &= \{j' \in [m] : \widehat{d}^2(j,j') \leq \eta^2_{col}\}.
\end{aligned} \tag{5}$$

The sub-sampling is done by thresholding $|\mathcal{N}^s_{row}(i)|$ at $\tau n \eta_{row}^{d_1/\lambda}$ and $|\mathcal{N}^s_{col}(j)|$ at $\tau m \eta_{col}^{d_2/\lambda}$ where $\tau > 1$. We name the subsampled nearest neighbors from $|\mathcal{N}^s_{row}(i)|$ and $|\mathcal{N}^s_{col}(j)|$ as $|\mathcal{N}_{row}(i)|$ and $|\mathcal{N}_{col}(j)|$ respectively. Recall the definition of $\mathcal{N}_{row,col}(i,j)$ from Step-2 of the TS-NN($\boldsymbol{\eta}$) algorithm in Sec. 3. For the ease of proof we define the following,

$$\tilde{\theta}_{i,j} = \frac{\sum_{(i',j') \in \mathcal{N}_{row,col}(i,j)} f(u_{i'}, v_{j'})}{|\mathcal{N}_{row,col}(i,j)|} = \frac{\sum_{(i',j') \in \mathcal{N}_{row,col}(i,j)} f(i',j')}{|\mathcal{N}_{row,col}(i,j)|}. \tag{6}$$

Note that $\tilde{\theta}_{i,j}$ is essentially $\widehat{\theta}_{i,j}$ where the noisy signals $X_{i',j'}$ appearing in the numerator of $\widehat{\theta}_{i,j}$ is replaced by the ground-truths $\theta_{i',j'} = f(u_{i'}, v_{j'})$. The MSE can then be decomposed into a bias and a variance term as follows,

$$\begin{aligned}
\text{MSE} &= \frac{1}{mn} \sum_{i \in [n], j \in [m]} \left(\widehat{\theta}_{i,j} - f(i,j)\right)^2 \\
&= \frac{1}{mn} \sum_{i \in [n], j \in [m]} \left(\left(\tilde{\theta}_{i,j} - f(i,j)\right) + \left(\widehat{\theta}_{i,j} - \tilde{\theta}_{i,j}\right)\right)^2 \\
&\leq \frac{2}{mn} \sum_{i \in [n], j \in [m]} \left(\tilde{\theta}_{i,j} - f(i,j)\right)^2 + \frac{2}{mn} \sum_{i \in [n], j \in [m]} \left(\widehat{\theta}_{i,j} - \tilde{\theta}_{i,j}\right)^2 \\
&= 2\mathbb{B} + 2\mathbb{V}.
\end{aligned} \tag{7}$$

We prove the theorem by bounding the bias ($\mathbb{B}$) and the variance ($\mathbb{V}$) term separately. The main tool that we repeatedly use for bounding these terms is the following distance concentration lemma (see proof in App. A.1).

Lemma 1 (Distance concentration)**.** *Under Assum. 1 to 3 and 5, for any $\delta \in (0,1]$ we have,*

$$\mathbb{P}(E_1 \cap E_2 | \mathcal{U}, \mathcal{V}) \geq 1 - \delta, \tag{8}$$

where,

$$E_1 = \left\{ \sup_{i \neq i'} |\widehat{d}^2(i,i') - d^2(i,i')| \leq \frac{\Delta_r}{\sqrt{\bar{\boldsymbol{p}}_{i,i'} m}} \right\} \quad \text{and} \quad E_2 = \left\{ \sup_{j \neq j'} |\widehat{d}^2(j,j') - d^2(j,j')| \leq \frac{\Delta_c}{\sqrt{\bar{\boldsymbol{p}}_{j,j'} n}} \right\}. \tag{9}$$

Here the mean of the vectors $\boldsymbol{p}_{i,i'} = [p_{i,j} p_{i',j}]_{j=1}^m$ *and* $\boldsymbol{p}_{j,j'} = [p_{i,j} p_{i,j'}]_{i=1}^n$ *are denoted by* $\bar{\boldsymbol{p}}_{i,i'}$ *and* $\bar{\boldsymbol{p}}_{j,j'}$ *respectively.* Δ_r *and* Δ_c *are constants free of* m, n*.*

Note that for MCAR missingness, the bound (9) simplifies with $\bar{\boldsymbol{p}}_{i,i'} = p^2$ and $\bar{\boldsymbol{p}}_{j,j'} = p^2$. We use the assumption of Hölder-continuity of the latent function f (Assum. 1) to obtain lower bounds on the number of nearest rows $\mathcal{N}^s_{row}(i)$ and columns $\mathcal{N}^s_{col}(j)$ under the events E_1, E_2.

Lemma 2. *The full set of row and column nearest neighbors before subsampling* ($|\mathcal{N}^s_{row}(i)|$ *and* $|\mathcal{N}^s_{col}(j)|$ *respectively) satisfy the following bounds,*

$$\mathbb{P}\left(|\mathcal{N}^s_{row}(i)| \geq (1-\delta) n \left(\frac{\eta^2_{row} - \frac{\Delta_r}{p\sqrt{m}}}{L^2} \right)^{\frac{d_1}{2\lambda}} \text{ for } i \in [n] \middle| E_1, E_2, \mathcal{U}, \mathcal{V} \right) \geq 1 - n \exp\left(-\frac{\delta^2 n}{2} \left(\frac{\eta^2_{row} - \frac{\Delta_r}{p\sqrt{m}}}{L^2} \right)^{\frac{d_1}{2\lambda}} \right) \tag{10}$$

$$\mathbb{P}\left(|\mathcal{N}^s_{col}(j)| \geq (1-\delta) m \left(\frac{\eta^2_{col} - \frac{\Delta_c}{p\sqrt{n}}}{L^2} \right)^{\frac{d_2}{2\lambda}} \text{ for } j \in [m] \middle| E_1, E_2, \mathcal{U}, \mathcal{V} \right) \geq 1 - m \exp\left(-\frac{\delta^2 m}{2} \left(\frac{\eta^2_{col} - \frac{\Delta_c}{p\sqrt{n}}}{L^2} \right)^{\frac{d_2}{2\lambda}} \right).$$

We consider the event of lower bounding the number of nearest rows and columns in the subsampled neighborhoods $\mathcal{N}_{row}(i)$ and $\mathcal{N}_{col}(j)$ as follows

$$A_1(z_1) = \bigcap_i \{ |\mathcal{N}_{row}(i)| \geq z_1 \}, \quad A_2(z_2) = \bigcap_j \{ |\mathcal{N}_{col}(j)| \geq z_2 \}.$$

where z_1, z_2 denote the lower bounds of $|\mathcal{N}^s_{row}(i)|$ and $|\mathcal{N}^s_{col}(j)|$ derived in Lem. 2. We will now show that $A_1(z_1) = \bigcap_i \{ |\mathcal{N}^s_{row}(i)| \geq z_1 \}$ and $A_2(z_2) = \bigcap_j \{ |\mathcal{N}^s_{col}(j)| \geq z_2 \}$ under the events E_1, E_2. In the regime $\eta^2_{row} \geq \Delta_r / \sqrt{m}$ and $\eta^2_{col} \geq \Delta_c / \sqrt{n}$ we have the equivalence,

$$\begin{cases} |\mathcal{N}^s_{row}(i)| \geq z_1 & \longleftrightarrow \quad |\mathcal{N}_{row}(i)| \geq z_1, \\ |\mathcal{N}^s_{col}(j)| \geq z_2 & \longleftrightarrow \quad |\mathcal{N}_{col}(j)| \geq z_2. \end{cases}$$

The above equivalence holds because the thresholding of the size of $\mathcal{N}^s_{row}(i)$ is at $\tau n \eta_{row}^{d_1/\lambda}$ which is strictly larger than the lower bound z_1. Similarly the thresholding of the size of $\mathcal{N}^s_{col}(j)$ is at $\tau n \eta_{col}^{d_2/\lambda}$ which is strictly larger than the lower bound z_2. This equivalence proves our assertion.

Proof of Lem. 2. We show that under event E_1, the first probability statement holds true. The proof of the other bound is similar. We observe that,

$$\begin{aligned} |\mathcal{N}^s_{row}(i)| &\overset{(5),(9)}{\geq} \sum_{i' \in [n]} \mathbb{I}\big[d^2(i,i') + \Delta_r / (p\sqrt{m}) \leq \eta^2_{row} \big] \\ &= \sum_{i' \in [n]} \mathbb{I}\big[d^2(i,i') \leq \eta^2_{row} - (\Delta_r / (p\sqrt{m})) \big] \\ &\overset{(A1)}{\geq} \sum_{i' \in [n]} \mathbb{I}\big[L^2 \|u_i - u_{i'}\|^{2\lambda} \leq \eta^2_{row} - (\Delta_r / (p\sqrt{m})) \big] \\ &= \sum_{i' \in [n]} \mathbb{I}\left[\|u_i - u_{i'}\| \leq \left(\frac{\eta^2_{row} - (\Delta_r / p\sqrt{m})}{L^2} \right)^{1/(2\lambda)} \right] \end{aligned}$$

Since $u_i \sim \text{Unif}([0,1]^{d_1})$ we get the following by applying Chernoff bound [Hagerup and Rüb (1990)] on $\|u_i - u_{i'}\|$,

$$\mathbb{P}\left(\|u_i - u_{i'}\| \le \left(\frac{\eta_{row}^2 - (\Delta_r/(p\sqrt{m}))}{L^2}\right)^{1/(2\lambda)}\right) \ge \left(\frac{\eta_{row}^2 - (\Delta_r/(p\sqrt{m}))}{L^2}\right)^{d_1/(2\lambda)}.$$

This implies that $\mathcal{N}_{row}(i)$ stochastically dominates $\text{Bin}(n,q)$ distribution where,

$$q = \left(\frac{\eta_{row}^2 - (\Delta_r/(p\sqrt{m}))}{L^2}\right)^{d_1/(2\lambda)}.$$

The proof of (10) is completed by using Chernoff bound [Hagerup and Rüb (1990)] on this binomial random variable and then using union bound to account for all the rows. $\square$

Let us first analyse the variance part. We consider the event A_3 for $0 < \delta < 1$ where,

$$A_3 = \{|\mathcal{N}_{row,col}(i,j)| \ge (1-\delta)p|\mathcal{N}_{row}(i)||\mathcal{N}_{col}(j)| \text{ for all } i,j \in [n]\times[m]\}$$

We apply Chernoff bound [Hagerup and Rüb (1990)] on the indicator random variables $A_{i,j}$ and union bound (to account for all $i,j \in [n]\times[m]$) to show that,

$$\mathbb{P}(A_3) \ge 1 - mn\exp\left(-\frac{\delta^2|\mathcal{N}_{row}(i)||\mathcal{N}_{col}(j)|p}{2}\right). \tag{11}$$

Under the events $A_1(z_1), A_2(z_2), A_3$, applying Hoeffding's inequality [Bentkus (2004)] on the noise terms $\epsilon_{i,j}$, yield the following for all $i,j \in [n]\times[m]$,

$$\begin{aligned} &\mathbb{P}\left(|\tilde{\theta}_{i,j} - \widehat{\theta}_{i,j}| > \zeta \,\middle|\, A_1(z_1), A_2(z_2), A_3\right) \le 2e^{-\frac{\zeta^2(1-\delta)pz_1z_2}{2\sigma^2}} \quad (12) \\ \implies \; &\mathbb{P}\left(|\tilde{\theta}_{i,j} - \widehat{\theta}_{i,j}|^2 > \frac{2\sigma^2\log(2/\delta)}{(1-\delta)pz_1z_2} \,\middle|\, A_1(z_1), A_2(z_2), A_3\right) \le \delta, \end{aligned}$$

for some $0 < \delta < 1$. This implies that conditioned on the events $A_1(z_1), A_2(z_2), A_3$, the following bound holds for the variance term $\mathbb{V}$,

$$\begin{aligned} \mathbb{P}\left(\mathbb{V} \le \frac{2\sigma^2\log(2/\delta)}{(1-\delta)pz_1z_2} \,\middle|\, A_1(z_1), A_2(z_2), A_3\right) &\overset{(7)}{=} \mathbb{P}\left(\frac{1}{mn}\sum_{i\in[n],j\in[m]}\left(\widehat{\theta}_{i,j} - \tilde{\theta}_{i,j}\right)^2 \le \frac{2\sigma^2\log(2/\delta)}{(1-\delta)pz_1z_2} \,\middle|\, A_1(z_1), A_2(z_2), A_3\right) \\ &\overset{(12)}{\ge} 1-\delta. \end{aligned}$$

Now that we have managed to bound the variance term ($\mathbb{V}$), we focus on the bias term ($\mathbb{B}$) in the decomposition of MSE.

We start by decomposing the bias term into further two parts,

$$\begin{aligned} \mathbb{B} &\overset{(6),(7)}{=} \frac{1}{nm}\sum_{i\in[n],j\in[m]}\left(\left(\frac{\sum_{i'\in\mathcal{N}_{row}(i)}\sum_{j'\in\mathcal{N}_{col}(t)} f(i',j')A_{i',j'}}{|\mathcal{N}_{row,col}(i,j)|}\right) - f(i,j)\right)^2 \quad (13) \\ &= \frac{1}{nm}\sum_{i\in[n],j\in[m]}\left(\frac{\sum_{i'\in\mathcal{N}_{row}(i)}\sum_{j'\in\mathcal{N}_{col}(j)}(f(i',j') - f(i,j))A_{i',j'}}{|\mathcal{N}_{row,col}(i,j)|}\right)^2 \\ &\overset{(i)}{\le} \frac{1}{nm}\sum_{i\in[n],j\in[m]}\frac{1}{|\mathcal{N}_{row,col}(i,j)|}\left(\sum_{i'\in\mathcal{N}_{row}(i)}\sum_{j'\in\mathcal{N}_{col}(t)}(f(i',j') - f(i,j))^2 A_{i',j'}\right) \\ &= \frac{1}{nm}\sum_{i\in[n],j\in[m]}\frac{1}{|\mathcal{N}_{row,col}(i,j)|}\left(\sum_{i'\in\mathcal{N}_{row}(i)}\sum_{j'\in\mathcal{N}_{col}(j)}(f(i',j') - f(i',j) + f(i',j) - f(i,j))^2 A_{i',j'}\right) \\ &\overset{(ii)}{\le} \frac{2}{nm}\sum_{i\in[n],j\in[m]}\frac{1}{|\mathcal{N}_{row,col}(i,j)|}\left(\sum_{i'\in\mathcal{N}_{row}(i)}\sum_{j'\in\mathcal{N}_{col}(j)}(f(i',j') - f(i',j))^2 A_{i',j'} + (f(i',j) - f(i,j))^2 A_{i',j'}\right) \\ &= \mathbb{B}_1 + \mathbb{B}_2. \end{aligned}$$

Here step-(i) follows from the AM-QM (arithmetic mean - quadratic mean) inequality that $(\sum_{i=1}^{n} a_i/n)^2 \leq \sum_{i=1}^{n} a_i^2/n$. Step-$(ii)$ follows from the basic inequality $(a+b)^2 \leq 2(a^2+b^2)$. Thus the bias term $\mathbb{B}$ can be further decomposed into two terms viz $\mathbb{B}_1$ and $\mathbb{B}_2$. The $\mathbb{B}_1$ term arises because of averaging the response across the neighboring rows and the $\mathbb{B}_2$ term arises because of averaging the response across the neighboring columns. We bound the bias term by separately obtaining bounds for $\mathbb{B}_1$ and $\mathbb{B}_2$. We start by obtaining an upper bound (with high probability) on $\mathbb{B}_2$ under the events $E_1, E_2, A_1(z_1), A_2(z_2), A_3$.

$$\begin{aligned}
\mathbb{B}_2 &\overset{(13)}{=} \frac{2}{nm} \sum_{i\in[n], j\in[m]} \frac{1}{|\mathcal{N}_{row,col}(i,j)|} \left(\sum_{i'\in\mathcal{N}_{row}(i)} \sum_{j'\in\mathcal{N}_{col}(j)} (f(i',j) - f(i,j))^2 A_{i',j'} \right) \\
&= \frac{2}{nm} \sum_{i\in[n], j\in[m]} \frac{1}{|\mathcal{N}_{row,col}(i,j)|} \left(\sum_{i'\in\mathcal{N}_{row}(i)} (f(i',j) - f(i,j))^2 \sum_{j'\in\mathcal{N}_{col}(j)} A_{i',j'} \right) \\
&\overset{(11)}{\leq} \frac{2}{nm} \sum_{i\in[n], j\in[m]} \frac{1}{(1-\delta)p|\mathcal{N}_{row}(i)||\mathcal{N}_{col}(j)|} \left(\sum_{i'\in\mathcal{N}_{row}(i)} (f(i',j) - f(i,j))^2 |\mathcal{N}_{col}(j)|p(1+\delta) \right). \qquad (14)
\end{aligned}$$

The inequality follows because with probability at least $1-\delta$ we have $\sum_{j'\in\mathcal{N}_{col}(j)} A_{i',j'} \leq |\mathcal{N}_{col}(j)|p(1+\delta)$ (by Chernoff bound[Hagerup and Rüb (1990)] on $A_{i',j'}$'s) for all $i' \in [n]$.

$$\begin{aligned}
(14) &= \frac{2}{nm} \sum_{i\in[n]} \sum_{i'\in\mathcal{N}_{row}(i)} \frac{1+\delta}{(1-\delta)|\mathcal{N}_{row}(i)|} \left(\sum_{j\in[m]} (f(i',j) - f(i,j))^2 \right) \\
&\leq \frac{2}{n} \sum_{i\in[n]} \sum_{i'\in\mathcal{N}_{row}(i)} \frac{1+\delta}{(1-\delta)|\mathcal{N}_{row}(i)|} d^2(i,i') \\
&\overset{(8)}{\leq} \frac{2}{n} \sum_{i\in[n]} \sum_{i'\in\mathcal{N}_{row}(i)} \frac{1+\delta}{(1-\delta)|\mathcal{N}_{row}(i)|} \left(\widehat{d}^2(i,i') + \frac{\Delta_r}{p\sqrt{m}} \right) \\
&\overset{(2)}{\leq} \frac{2}{n} \sum_{i\in[n]} \sum_{i'\in\mathcal{N}_{row}(i)} \frac{1+\delta}{(1-\delta)|\mathcal{N}_{row}(i)|} \left(\eta_{row}^2 + \frac{\Delta_r}{p\sqrt{m}} \right) \\
&= 2\left(\eta_{row}^2 + \frac{\Delta_r}{p\sqrt{m}} \right) \frac{1+\delta}{1-\delta}.
\end{aligned}$$

Similarly we can show that with a high probability the following bound holds for the first term of the bias decomposition $\mathbb{B}_1$ under the regime $\eta_{row}^2 \geq \Delta_r/\sqrt{m}$ and under the events $E_1, E_2, A_1(z_1), A_2(z_2), A_3$,

$$\begin{aligned}
\mathbb{B}_1 &\overset{(13)}{=} \frac{2}{mn} \sum_{i\in[n], j\in[m]} \frac{1}{|\mathcal{N}_{row,col}(i,j)|} \sum_{i'\in\mathcal{N}_{row}(i)} \sum_{j'\in\mathcal{N}_{col}(j)} (f(i',j') - f(i',j))^2 A_{i',j'} \qquad (15) \\
&\overset{(11)}{\leq} \frac{2}{mn} \sum_{i\in[n], j\in[m]} \frac{1}{|\mathcal{N}_{row}(i)||\mathcal{N}_{col}(j)|(1-\delta)p} \sum_{i'\in\mathcal{N}_{row}(i)} \sum_{j'\in\mathcal{N}_{col}(j)} (f(i',j') - f(i',j))^2 A_{i',j'} \\
&\overset{(10)}{\leq} \frac{2}{mn} \sum_{i\in[n], j\in[m]} \frac{1}{(1-\delta)n\left((\eta_{row}^2 - \frac{\Delta_r}{p\sqrt{m}})/L^2 \right)^{\frac{d_1}{2\lambda}} |\mathcal{N}_{col}(j)|(1-\delta)p} \sum_{i'\in\mathcal{N}_{row}(i)} \sum_{j'\in\mathcal{N}_{col}(j)} (f(i',j') - f(i',j))^2 A_{i',j'} \\
&\overset{(i)}{\leq} \frac{2}{mn} \sum_{i\in[n], j\in[m]} \frac{L^{d_1/\lambda}}{(1-\delta)n\left(\eta_{row}^2 - \frac{\Delta_r}{p\sqrt{m}} \right)^{\frac{d_1}{2\lambda}} |\mathcal{N}_{col}(j)|(1-\delta)p} \sum_{j'\in\mathcal{N}_{col}(j)} (f(i,j') - f(i,j))^2 A_{i,j'} |\mathcal{N}_{row}(i)|
\end{aligned}$$

Here step-(i) follows by aggregating all the terms which belong to set of neighboring rows of u_i. We can further simplify the bound as follows,

$$
\begin{aligned}
(15) = & \frac{2}{mn} \sum_{i\in[n], j\in[m]} \frac{L^{d_1/\lambda}}{(1-\delta)n\Big(\eta_{row}^2 - \frac{\Delta_r}{p\sqrt{m}}\Big)^{\frac{d_1}{2\lambda}} |\mathcal{N}_{col}(j)|(1-\delta)p} \sum_{j'\in\mathcal{N}_{col}(j)} (f(i,j') - f(i,j))^2 A_{i,j'} |\mathcal{N}_{row}(i)| \\
\overset{(i)}{\leq} & \frac{2}{mn} \sum_{j\in[m]} \frac{\tau n \eta_{row}^{d_1/\lambda} L^{d_1/\lambda}}{(1-\delta)n\Big(\eta_{row}^2 - \frac{\Delta_r}{p\sqrt{m}}\Big)^{\frac{d_1}{2\lambda}} |\mathcal{N}_{col}(j)|(1-\delta)p} \sum_{j'\in\mathcal{N}_{col}(j)} \sum_{i\in[n]} (f(i,j') - f(i,j))^2 A_{i,j'} \\
\overset{(8),(ii)}{\leq} & \frac{2}{mn} \sum_{j\in[m]} \frac{\tau'}{(1-\delta)^2 |\mathcal{N}_{col}(j)| p} \sum_{j'\in\mathcal{N}_{col}(j)} (\sum_{i\in[n]} A_{i,j'}) \Big(d^2(j,j') + \frac{\Delta_c}{p\sqrt{n}}\Big) \\
\overset{(iii)}{\leq} & \frac{2}{mn} \sum_{j\in[m]} \frac{\tau'}{(1-\delta)^2 |\mathcal{N}_{col}(j)| p} \sum_{j'\in\mathcal{N}_{col}(j)} (1+\delta)np \Big(d^2(j,j') + \frac{\Delta_c}{p\sqrt{n}}\Big).
\end{aligned} \tag{16}
$$

The step-(i) is a consequence of the fact that $|\mathcal{N}_{row}(i)| \leq \tau n \eta_{row}^{d_1/\lambda}$ because of the subsampling procedure. In the inequality (ii) we used the fact that under the regime $\eta_{row}^2 \geq \Delta/\sqrt{m}$ there exists a constant $\tau' > 0$ such that,

$$
\frac{\tau n \eta_{row}^{d_1/\lambda} L^{d_1/\lambda}}{(1-\delta)n\Big(\eta_{row}^2 - \frac{\Delta_r}{p\sqrt{m}}\Big)^{\frac{d_1}{2\lambda}}} \leq \tau' \quad \text{for all} \quad n,
$$

The inequality (iii) holds because of the fact that with probability at least $1-\delta$ we have $\sum_{i\in[n]} A_{i,j'} \leq (1+\delta)np$ (using Chernoff bound [Hagerup and Rüb (1990)] on $A_{i,j'}$'s) for all $j' \in [m]$.

$$
\begin{aligned}
(16) \overset{(8)}{\leq} & \frac{2}{m} \sum_{j\in[m]} \frac{\tau'(1+\delta)}{(1-\delta)^2 |\mathcal{N}_{col}(j)|} \sum_{j'\in\mathcal{N}_{col}(j)} \Big(\widehat{d}^2(j,j') + \frac{2\Delta_c}{p\sqrt{n}}\Big) \\
\overset{(2)}{\leq} & 2\Big(\eta_{col}^2 + \frac{2\Delta_c}{p\sqrt{n}}\Big) \frac{\tau'(1+\delta)}{(1-\delta)^2}.
\end{aligned}
$$

If we put together all the bounds that we have shown till now, we get that with probability at least $1-7\delta$,

$$
\begin{aligned}
\text{MSE} & \leq 2\mathbb{V} + 2\mathbb{B}_1 + 2\mathbb{B}_2 \\
& \leq \frac{4\sigma^2 \log(2/\delta)}{(1-\delta)pz_1z_2} + 4\Big(\eta_{row}^2 + \frac{\Delta_r}{p\sqrt{m}}\Big)\frac{1+\delta}{1-\delta} + 4\Big(\eta_{col}^2 + \frac{2\Delta_c}{p\sqrt{n}}\Big)\frac{\tau'(1+\delta)}{(1-\delta)^2}.
\end{aligned}
$$

It can be easily shown that this is equivalent to the statement made in Thm. 1. This completes of the proof of the theorem.

A.1 Proof of Lem. 1

We prove the distance concentration lemma for the rows. The result for the columns will follow analogously. Recall the definitions of $\widehat{d}^2(i,i')$ and $d^2(i,i')$ from Sec. 3,

$$
\begin{aligned}
\widehat{d}^2(i,i') &= \frac{\sum_{j\in[m]} (X_{i,j} - X_{i',j})^2 A_{i,j} A_{i',j}}{\sum_{j\in[m]} A_{i,j} A_{i',j}} - 2\sigma^2, \\
d^2(i,i') &= \frac{1}{m} \sum_{j\in[m]} (f(u_i, v_j) - f(u_{i'}, v_j))^2.
\end{aligned} \tag{17}
$$

We also define the population mean over the column latent factors,

$$
\rho^*_{i,i'} = \mathbb{E}_v[(f(u_i, v) - f(u_{i'}, v))^2 | \mathcal{U}]. \tag{18}
$$

To prove Lem. 1 we derive concentration bounds of both $\widehat{d}^2(i,i') - \rho^*_{i,i'}$ and $d^2(i,i') - \rho^*_{i,i'}$. Lem. 1 then follows by applying triangle inequality on these two concentration bounds. We start by proving the concentration bounds for $\widehat{d}^2(i,i') - \rho^*_{i,i'}$. We denote the number of columns corresponding to which entries are observed in both the rows i, i' by $T_{i,i'} = \sum_{j\in[m]} A_{i,j}A_{i',j}$. For the purpose of proof we also define the stopping times $t_{(l)}(i,i')$ for observing an entry in both the rows i, i' for the l-th time. To put it rigorously we set $t_{(0)}(i,i') = 0$. For $l \geq 1$ we define iteratively,

$$t_{(l)}(i,i') = \begin{cases} \min\{t : t_{(l-1)}(i,i') < t \leq m \text{ such that } A_{i,j}A_{i',j} = 1\} & \text{if such a } j \text{ exists,} \\ m+1 & \text{otherwise.} \end{cases} \tag{19}$$

We observe that $\widehat{d}^2(i,i') - \rho^*_{i,i'}$ has the following representation.

$$\begin{aligned} \widehat{d}^2(i,i') - \rho^*_{i,i'} &\overset{(17)}{=} \frac{\sum_{j\in[m]}[(X_{i,j} - X_{i',j})^2 - 2\sigma^2 - \rho^*_{i,i'}]A_{i,j}A_{i',j}}{\sum_{j\in[m]} A_{i,j}A_{i',j}} \\ &\overset{(19)}{=} \frac{\sum_{l=1}^{T_{i,i'}} \mathbf{1}(t_l(i,i') \leq m)[(X_{i,t_l(i,i')} - X_{i',t_l(i,i')})^2 - 2\sigma^2 - \rho^*_{i,i'}]}{T_{i,i'}} \\ &= \frac{\sum_{l=1}^{T_{i,i'}} W_l}{T_{i,i'}}, \end{aligned} \tag{20}$$

where $W_l = \mathbf{1}(t_l(i,i') \leq m)[(X_{i,t_l(i,i')} - X_{i',t_l(i,i')})^2 - 2\sigma^2 - \rho^*_{i,i'}]$ for $l = 1, \cdots, T_{i,i'}$. By Hoeffding's inequality [Bentkus (2004)] on the error terms $\epsilon_{i,j}$ we can show that $\max|\epsilon_{i,j}| \leq \sigma\sqrt{2\log((2mn)/\delta)} = c_\epsilon$. This implies that $|W_l|$ is bounded above by $8D^2$ where $D = M + c_\epsilon$. Let us denote the sigma algebra containing all the information upto time t as $\mathcal{F}_t$ for $t = 1, \cdots, m$. Let us denote the sigma field generated by the stopping time $t_l(i,i')$ by $\mathcal{H}_l$. We observe the following,

$$\begin{aligned} \mathbb{E}[W_l|H_l,\mathcal{U}] =& \mathbb{E}[\mathbf{1}(t_l(i,i') \leq m)[(X_{i,t_l(i,i')} - X_{i',t_l(i,i')})^2 - 2\sigma^2 - \rho^*_{i,i'}]|H_l,\mathcal{U}] \\ =& \mathbf{1}(t_l(i,i') \leq m)\mathbb{E}[[(X_{i,t_l(i,i')} - X_{i',t_l(i,i')})^2 - 2\sigma^2 - \rho^*_{i,i'}]|H_l,\mathcal{U}] \\ \overset{(18)}{=}& \mathbf{1}(t_l(i,i') \leq m)(\rho^*_{i,i'} - \rho^*_{i,i'}) \\ =& 0. \end{aligned}$$

Thus we conclude that $\{W_l\}_{l=0}^\infty$ conditioned on the row latent factors $\mathcal{U}$ is a bounded martingale difference w.r.t. the sigma algebra $\{\mathcal{H}_l\}_{l=0}^\infty$. We shall use the Azuma martingale concentration result in this set-up.

Result 1 (Azuma martingale concentration)**.** *Consider a bounded martingale difference sequence $\{S_n\}_{n=1}^\infty$ adapted to the filtration $\{\mathcal{F}_n\}_{n=1}^\infty$ i.e. $\mathbb{E}[S_n|\mathcal{F}_{n-1}] = 0$ for all $n \in \mathbb{N}$. Suppose $|S_n| \leq M$ for all $n \in \mathbb{N}$. Then the following event holds with probability at least $1-\delta$,*

$$\left|\sum_{i=1}^n S_i\right| \leq M\sqrt{n\log(2/\delta)}.$$

We can make the following computations,

$$
\begin{aligned}
&\mathbb{P}\left(|\widehat{d}^2(i,i') - \rho^*_{i,i'}| \le 8D^2\sqrt{\frac{\log(2/\delta)}{T_{i,i'}}}, T_{i,i'} > 0|\mathcal{U}\right) \qquad (21)\\
\overset{(20)}{=}&\mathbb{P}\left(\left|\frac{\sum_{l=1}^{T_{i,i'}} W_l}{T_{i,i'}}\right| \le 8D^2\sqrt{\frac{\log(2/\delta)}{T_{i,i'}}}, T_{i,i'} > 0|\mathcal{U}\right)\\
=&\sum_{k=1}^{m}\mathbb{P}\left(\left|\frac{\sum_{l=1}^{T_{i,i'}} W_l}{T_{i,i'}}\right| \le 8D^2\sqrt{\frac{\log(2/\delta)}{k}}, T_{i,i'} = k|\mathcal{U}\right)\\
\ge&\sum_{k=1}^{m}\mathbb{P}\left(\left|\frac{\sum_{l=1}^{T_{i,i'}} W_l}{T_{i,i'}}\right| \le 8D^2\sqrt{\frac{\log(2/\delta)}{k}} \text{ for all } k\in[m], T_{i,i'} = k|\mathcal{U}\right)\\
=&\mathbb{P}\left(\left|\frac{\sum_{l=1}^{T_{i,i'}} W_l}{T_{i,i'}}\right| \le 8D^2\sqrt{\frac{\log(2/\delta)}{k}} \text{ for all } k\in[m], T_{i,i'} > 0|\mathcal{U}\right)\\
=&\mathbb{P}\left(\left|\frac{\sum_{l=1}^{T_{i,i'}} W_l}{T_{i,i'}}\right| \le 8D^2\sqrt{\frac{\log(2/\delta)}{k}} \text{ for all } k\in[m]|\mathcal{U}\right) + \mathbb{P}(T_{i,i'} > 0|\mathcal{U}) - 1\\
\overset{(i)}{\ge}&\mathbb{P}(T_{i,i'} > 0|\mathcal{U}) - m\delta\\
\overset{(ii)}{\ge}&1-(m+1)\delta.
\end{aligned}
$$

The inequality in (i) follows from the application of Result 1 and the fact that for any two events A, B defined in the same probability space we have $\mathbb{P}(A\cap B) \ge \mathbb{P}(A) + \mathbb{P}(B) - 1$. In step (ii) we used that the following probability statement holds for any $\delta > 0$,

$$
\mathbb{P}\left(T_{i,i'} - 1^T\boldsymbol{p}_{i,i'} > -\sqrt{2(1^T\boldsymbol{p}_{i,i'})\log(1/\delta)}\,\middle|\,\mathcal{U},\mathcal{V}\right) \ge 1-\delta.
$$

This probability statement follows from the concentration of sum of weighted Bernoulli random variables (Lemma-2 of Dwivedi et al. (2022b)). If we use this bound on $T_{i,i'}$ in (21) we can say with probability at least $1-(m+1)\binom{n}{2}\delta$, the following event holds for all rows i, i',

$$
|\widehat{d}^2(i,i') - \rho^*_{i,i'}| \le \frac{8D^2}{\sqrt{m}}\sqrt{\frac{\log(2/\delta)}{\bar{\boldsymbol{p}}_{i,i'}\left[1-\sqrt{\frac{2\log(1/\delta)}{m\bar{\boldsymbol{p}}_{i,i'}}}\right]}},
$$

where $\bar{\boldsymbol{p}}_{i,i'} = (1^T\boldsymbol{p}_{i,i'})/m$. The above equation gives us the concentration of $\widehat{d}^2(i,i')$ about $\rho^*_{i,i'}$. To get the concentration of $d^2(i,i')$ about $\rho^*_{i,i'}$ we use the same argument as above by replacing all the probabilities $p_{i,j}$'s with 1. In particular, we can show that with probability at least $1-(m+1)\binom{n}{2}\delta$, the following event holds for all rows i, i',

$$
|d^2(i,i') - \rho^*_{i,i'}| \le \frac{8D^2}{\sqrt{m}}\sqrt{\frac{\log(2/\delta)}{1-\sqrt{\frac{2\log(1/\delta)}{m}}}}.
$$

Combining both the above inequalities we can say that with probability at least $1-2(m+1)\binom{n}{2}\delta$, the following event holds for all rows i, i',

$$
\begin{aligned}
|\widehat{d}^2(i,i') - d^2(i,i')| &\le \frac{8D^2}{\sqrt{m}}\left(\sqrt{\frac{\log(2/\delta)}{\bar{\boldsymbol{p}}_{i,i'}\left[1-\sqrt{\frac{2\log(1/\delta)}{m\bar{\boldsymbol{p}}_{i,i'}}}\right]}} + \sqrt{\frac{\log(2/\delta)}{1-\sqrt{\frac{2\log(1/\delta)}{m}}}}\right)\\
&\le \frac{\Delta_r}{\sqrt{\bar{\boldsymbol{p}}_{i,i'}m}} \quad \text{(for a suitable constant } \Delta_r).
\end{aligned}
$$

This completes the proof of the lemma.

B Proof of Cor. 1

We start with the upper bound on MSE proved in Thm. 1. Thereafter we substitute the values of z_1, z_2 in the result and carefully choose the value of the tuning parameters η_{row}, η_{col} to get the optimal MSE.

$$\begin{aligned}\text{MSE} &\leq \frac{4\sigma^2 \log(2/\delta)}{(1-\delta)pz_1z_2} + 4\left(\eta_{row}^2 + \frac{\Delta_r}{p\sqrt{m}}\right)\frac{1+\delta}{1-\delta} + 4\left(\eta_{col}^2 + \frac{2\Delta_c}{p\sqrt{n}}\right)\frac{\tau'(1+\delta)}{(1-\delta)^2} \\ &\leq C\left[\frac{1}{n\left(\eta_{row}^2 - \frac{\Delta_r}{\sqrt{m}}\right)^{\frac{d_1}{2\lambda}} m\left(\eta_{col}^2 - \frac{\Delta_c}{\sqrt{n}}\right)^{\frac{d_2}{2\lambda}}} + \eta_{row}^2 + \frac{\Delta_r}{\sqrt{m}} + \eta_{col}^2 + \frac{\Delta_c}{\sqrt{n}}\right]\end{aligned}$$

Here without loss of generality C is used to denote any arbitrary constant. From the expression of MSE it can be seen that under the regime $n = O(\min\{m^{\frac{2\lambda+d_1}{d_2}}, m^{\frac{d_1+d_2-2\lambda}{4\lambda}}\})$ (this ensures $\eta_{row,opt}^2 \geq \frac{C\Delta_r}{\sqrt{m}}$ and $|\mathcal{N}_{col}(j)| \geq z_2 \geq Cm\eta_{col,opt}^{d_2/\lambda} \geq 1$) and $n = \omega(\max\{m^{\frac{d_1}{2\lambda+d_2}}, m^{\frac{4\lambda}{d_1+d_2-2\lambda}}\})$ (this ensures $\eta_{col,opt}^2 \geq \frac{C\Delta_c}{\sqrt{n}}$ and $|\mathcal{N}_{row}(i)| \geq z_1 \geq Cn\eta_{row,opt}^{d_1/\lambda} \geq 1$), the two-sided NN estimator with $\eta_{row,opt} = \eta_{col,opt} = \Theta\left((mn)^{\frac{-\lambda}{2\lambda+d_1+d_2}}\right)$ obtains the rate,

$$\text{MSE} = O\left((mn)^{\frac{-2\lambda}{2\lambda+d_1+d_2}}\right).$$

This completes the proof of the corollary.

C Proof of Thm. 2

We know from the proof of Thm. 1 that with probability at least $1-\delta$ each of the events $E_1, E_2, A_1(z_1), A_2(z_2)$ hold true. Like in the proof of Thm. 1 we start by decomposing the MSE into a bias part and a variance part,

$$\begin{aligned}\text{MSE} &\leq \frac{2}{mn}\sum_{i\in[n],j\in[m]}\left(\tilde{\theta}_{i,j} - f(i,j)\right)^2 + \frac{2}{mn}\sum_{i\in[n],j\in[m]}\left(\widehat{\theta}_{i,j} - \tilde{\theta}_{i,j}\right)^2 \\ &= 2\mathbb{B} + 2\mathbb{V}.\end{aligned} \tag{22}$$

We bound the MSE by obtaining bounds on the bias and the variance term in the decomposition above. Let us first look at the variance part. We use A_4 to denote the event that $|\mathcal{N}_{row,col}(i,j)|\Big|\mathcal{U},\mathcal{V} \geq g(\delta)|\mathcal{N}_{row}(i)||\mathcal{N}_{col}(j)|$. From the assumption made in Thm. 2 (Assum. 6) we know that $\mathbb{P}(A_4) \geq 1-\delta$. From our previous discussions we have the following concentration bound using the Hoeffding's inequality [Bentkus (2004)] on the noise terms $\epsilon_{i,j}$'s under the events $E_1, E_2, A_1(z_1), A_2(z_2), A_4$,

$$\mathbb{P}\left(|\widehat{\theta}_{i,j} - \tilde{\theta}_{i,j}| > \zeta \Big| E_1, E_2, A_1(z_1), A_2(z_2), A_4\right) \leq 2\exp\left\{-\frac{\zeta^2 g(\delta) z_1 z_2}{2\sigma^2}\right\}. \tag{23}$$

This allows us to bound the variance term $\mathbb{V}$,

$$\begin{aligned}&\mathbb{P}\left(\mathbb{V} \leq \frac{2\sigma^2\log(2/\delta)}{g(\delta)z_1z_2}\Big| E_1, E_2, A_1(z_1), A_2(z_2), A_4\right) \\ &\overset{(22)}{=}\mathbb{P}\left(\frac{1}{mn}\sum_{i\in[n],j\in[m]}\left(\widehat{\theta}_{i,j} - \tilde{\theta}_{i,j}\right)^2 \leq \frac{2\sigma^2\log(2/\delta)}{g(\delta)z_1z_2}\Big| E_1, E_2, A_1(z_1), A_2(z_2), A_4\right) \\ &\overset{(23)}{\geq} 1 - mn\delta.\end{aligned}$$

Now let us come to the bias part. From previous discussions we know that the $\mathbb{B}$ term can be bounded above by,

$$\begin{aligned}\mathbb{B} &\overset{(22)}{\leq} \frac{2}{nm}\sum_{i\in[n],j\in[m]}\frac{1}{|\mathcal{N}_{row,col}(i,j)|}\left(\sum_{i'\in\mathcal{N}_{row}(i)}\sum_{j'\in\mathcal{N}_{col}(j)}\left(f(i',j') - f(i',j)\right)^2 A_{i',j'} + \left(f(i',j) - f(i,j)\right)^2 A_{i',j'}\right) \\ &= \mathbb{B}_1 + \mathbb{B}_2.\end{aligned} \tag{24}$$

Let us first put an upper bound (with high probability) on the second term in the bias decomposition, $\mathbb{B}_2$ under the events $E_1, E_2, A_1(z_1), A_2(z_2), A_4$. We can re-write $\mathbb{B}_2$ as,

$$\mathbb{B}_2 = \frac{2}{nm} \sum_{i\in[n],j\in[m]} \left(\sum_{i'\in\mathcal{N}_{row}(i)} (f(i',j) - f(i,j))^2 \left(\frac{\sum_{j'\in\mathcal{N}_{col}(j)} A_{i',j'}}{\sum_{i'\in\mathcal{N}_{row}(i),j'\in\mathcal{N}_{col}(j)} A_{i',j'}} \right) \right). \tag{25}$$

Let us denote the weights by $w^{i,j}_{i',j'} = A_{i',j'}/(\sum_{i'\in\mathcal{N}_{row}(i),j'\in\mathcal{N}_{col}(j)} A_{i',j'})$. We note that,

$$w^{i,j}_{i',j'} = \frac{A_{i',j'}}{\sum_{i'\in\mathcal{N}_{row}(i),j'\in\mathcal{N}_{col}(j)} A_{i',j'}} \leq \frac{1}{\mathcal{N}_{row,col}(i,j)} \overset{(A6)}{\leq} \frac{1}{g(\delta)\mathcal{N}_{row}(i)\mathcal{N}_{col}(j)}. \tag{26}$$

This implies that,

$$\frac{\sum_{j'\in\mathcal{N}_{col}(j)} A_{i',j'}}{\sum_{i'\in\mathcal{N}_{row}(i),j'\in\mathcal{N}_{col}(j)} A_{i',j'}} = \sum_{j'\in\mathcal{N}_{col}(j)} w^{i,j}_{i',j'} \overset{(26)}{\leq} \frac{1}{g(\delta)\mathcal{N}_{row}(i)}. \tag{27}$$

We use this bound in the expression of the $\mathbb{B}_2$ to get the following,

$$\begin{aligned} \mathbb{B}_2 &\overset{(25)}{=} \frac{2}{nm} \sum_{i\in[n],j\in[m]} \left(\sum_{i'\in\mathcal{N}_{row}(i)} (f(i',j) - f(i,j))^2 \left(\frac{\sum_{j'\in\mathcal{N}_{col}(j)} A_{i',j'}}{\sum_{i'\in\mathcal{N}_{row}(i),j'\in\mathcal{N}_{col}(j)} A_{i',j'}} \right) \right) \\ &\overset{(27)}{\leq} \frac{2}{g(\delta)nm} \sum_{i\in[n],j\in[m]} \frac{1}{|\mathcal{N}_{row}(i)|} \left(\sum_{i'\in\mathcal{N}_{row}(i)} (f(i',j) - f(i,j))^2 \right) \\ &= \frac{2}{g(\delta)nm} \sum_{i\in[n]} \sum_{i'\in\mathcal{N}_{row}(i)} \frac{1}{|\mathcal{N}_{row}(i)|} \left(\sum_{j\in[m]} (f(i',j) - f(i,j))^2 \right) \\ &\overset{(17)}{=} \frac{2}{g(\delta)n} \sum_{i\in[n]} \sum_{i'\in\mathcal{N}_{row}(i)} \frac{1}{|\mathcal{N}_{row}(i)|} d^2(i,i'). \end{aligned} \tag{28}$$

We can further simplify the bound using Lem. 1,

$$\begin{aligned} (28) &\overset{L1}{\leq} \frac{2}{g(\delta)n} \sum_{i\in[n]} \sum_{i'\in\mathcal{N}_{row}(i)} \frac{1}{|\mathcal{N}_{row}(i)|} \left(\hat{d}^2(i,i') + \frac{\Delta_r}{\sqrt{\bar{\boldsymbol{p}}_{i,i'} m}} \right) \\ &\overset{(2)}{\leq} \frac{2}{g(\delta)n} \sum_{i\in[n]} \sum_{i'\in\mathcal{N}_{row}(i)} \frac{1}{|\mathcal{N}_{row}(i)|} \left(\eta^2_{row} + \frac{\Delta_r}{\sqrt{\bar{\boldsymbol{p}}_{i,i'} m}} \right) \\ &= \frac{2}{g(\delta)} \left(\eta^2_{row} + \frac{\Delta_r}{\sqrt{\bar{\boldsymbol{p}}_{i,i'} m}} \right). \end{aligned}$$

Similarly under the regime $\eta^2_{row} \geq \Delta_r/\sqrt{m}$ we can bound $\mathbb{B}_1$ under the events $E_1, E_2, A_1(z_1), A_2(z_2), A_4$,

$$\begin{aligned} \mathbb{B}_1 &\overset{(24)}{=} \frac{2}{mn} \sum_{i\in[n],j\in[m]} \frac{1}{|\mathcal{N}_{row,col}(i,j)|} \sum_{i'\in\mathcal{N}_{row}(i)} \sum_{j'\in\mathcal{N}_{col}(j)} (f(i',j') - f(i',j))^2 A_{i',j'} \\ &\overset{(A6)}{\leq} \frac{2}{g(\delta)mn} \sum_{i\in[n],j\in[m]} \frac{1}{|\mathcal{N}_{row}(i)||\mathcal{N}_{col}(j)|} \sum_{i'\in\mathcal{N}_{row}(i)} \sum_{j'\in\mathcal{N}_{col}(j)} (f(i',j') - f(i',j))^2 \\ &\overset{(L2)}{\leq} \frac{2}{g(\delta)mn} \sum_{i\in[n],j\in[m]} \frac{1}{(1-\delta)n \left(\frac{\eta^2_{row} - \frac{\Delta_r}{\sqrt{m}}}{L^2} \right)^{\frac{d_1}{2\lambda}} |\mathcal{N}_{col}(j)|} \sum_{i'\in\mathcal{N}_{row}(i)} \sum_{j'\in\mathcal{N}_{col}(j)} (f(i',j') - f(i',j))^2 \\ &\overset{(i)}{\leq} \frac{2}{g(\delta)mn} \sum_{i\in[n],j\in[m]} \frac{1}{(1-\delta)n \left(\frac{\eta^2_{row} - \frac{\Delta_r}{\sqrt{m}}}{L^2} \right)^{\frac{d_1}{2\lambda}} |\mathcal{N}_{col}(j)|} \sum_{j'\in\mathcal{N}_{col}(j)} (f(i,j') - f(i,j))^2 |\mathcal{N}_{row}(i)|. \end{aligned} \tag{29}$$

Here step-(i) follows by aggregating all the terms which belong to the set of neighboring rows of u_i. We further simplify the bound as follows,

$$
\begin{aligned}
(29) &\overset{(ii)}{\leq} \frac{2}{g(\delta)mn} \sum_{j\in[m]} \frac{\tau n \eta_{row}^{d_1/\lambda}}{(1-\delta)n\left(\frac{\eta_{row}^2 - \frac{\Delta_r}{\sqrt{m}}}{L^2}\right)^{\frac{d_1}{2\lambda}} |\mathcal{N}_{col}(j)|} \sum_{j'\in\mathcal{N}_{col}(j)} \sum_{i\in[n]} (f(i,j') - f(i,j))^2 \\
&\overset{(iii),(8)}{\leq} \frac{2}{g(\delta)m} \sum_{j\in[m]} \frac{\tau'}{(1-\delta)|\mathcal{N}_{col}(j)|} \sum_{j'\in\mathcal{N}_{col}(j)} \left(d^2(j,j') + \frac{\Delta_c}{\sqrt{\bar{\boldsymbol{p}}_{j,j'} n}} \right) \\
&\overset{(8)}{\leq} \frac{2}{g(\delta)m} \sum_{j\in[m]} \frac{\tau'}{(1-\delta)|\mathcal{N}_{col}(j)|} \sum_{j'\in\mathcal{N}_{col}(j)} \left(\widehat{d}^2(j,j') + \frac{2\Delta_c}{\sqrt{\bar{\boldsymbol{p}}_{j,j'} n}} \right) \\
&\overset{(2)}{\leq} \frac{2}{g(\delta)} \left(\eta_{col}^2 + \frac{2\Delta_c}{\sqrt{\bar{\boldsymbol{p}}_{j,j'} n}} \right) \frac{\tau'}{(1-\delta)}.
\end{aligned}
$$

The step-(ii) follows since $|\mathcal{N}_{row}(i)| \leq \tau n \eta_{row}^{d_1/\lambda}$ (because of sub-sampling the neighboring rows). The inequality (iii) follows as there exists a constant $\tau' > 0$ such that,

$$
\frac{\tau n \eta_{row}^{d_1/\lambda}}{(1-\delta)n\left(\frac{\eta_{row}^2 - \frac{\Delta_r}{\sqrt{m}}}{L^2}\right)^{\frac{d_1}{2\lambda}}} \leq \tau' \text{ for all } n.
$$

Combining all the computations above we get that the MSE of the two-sided nearest neighbor is bounded above by the following with probability at least $1 - 7\delta$,

$$
\begin{aligned}
\text{MSE} &\leq 2\mathbb{V} + 2\mathbb{B}_1 + 2\mathbb{B}_2 \\
&\leq \frac{4}{g(\delta)} \left[\frac{\sigma^2 \log(2/\delta)}{z_1 z_2} + \left(\eta_{row}^2 + \frac{\Delta_r}{\sqrt{\bar{\boldsymbol{p}}_{i,i'} m}} \right) + \left(\eta_{col}^2 + \frac{2\Delta_c}{\sqrt{\bar{\boldsymbol{p}}_{j,j'} n}} \right) \frac{\tau'}{(1-\delta)} \right].
\end{aligned}
$$

It can be easily shown that this is equivalent to the statement made in Thm. 2. Moreover we observe that,

$$
\begin{aligned}
\text{MSE} &\leq \frac{4}{g(\delta)} \left[\frac{\sigma^2 \log(2/\delta)}{z_1 z_2} + \left(\eta_{row}^2 + \frac{\Delta_r}{\sqrt{\bar{\boldsymbol{p}}_{i,i'} m}} \right) + \left(\eta_{col}^2 + \frac{2\Delta_c}{\sqrt{\bar{\boldsymbol{p}}_{j,j'} n}} \right) \frac{\tau'}{(1-\delta)} \right] \\
&\leq C \left[\frac{1}{n\left(\eta_{row}^2 - \frac{\Delta_r}{\sqrt{m}}\right)^{\frac{d_1}{2\lambda}} m \left(\eta_{col}^2 - \frac{\Delta_c}{\sqrt{n}}\right)^{\frac{d_2}{2\lambda}}} + \eta_{row}^2 + \frac{\Delta_r}{\sqrt{m}} + \eta_{col}^2 + \frac{\Delta_c}{\sqrt{n}} \right].
\end{aligned}
$$

Similar to the proof of Cor. 1 we can show that for $n = \omega(\max\{m^{\frac{d_1}{2\lambda+d_2}}, m^{\frac{4\lambda}{d_1+d_2-2\lambda}}\})$ and $n = O(\min\{m^{\frac{2\lambda+d_1}{d_2}}, m^{\frac{d_1+d_2-2\lambda}{4\lambda}}\})$ the MSE of the two-sided nearest neighbor algorithm with $\eta_{row} = \eta_{col} = \Theta\left((mn)^{\frac{-\lambda}{2\lambda+d_1+d_2}}\right)$ achieves the non-parametric minimax optimal rate $O((mn)^{\frac{-2\lambda}{2\lambda+d_1+d_2}})$.

D Deferred simulation details

D.1 Deffered details about Simulation experiments

Tuning η and reporting test error We do 5 fold cross-validation to tune the η in TS-NN(other nearest neighbors are implemented in similar fashion) and report its test error. At first, matrix entries are arbitrarily assigned to 5 different folds. One of the folds is held out as the test data and the other 4 folds are used for training the NNs, denote them as $\mathcal{F}_{test}$ and $\mathcal{F}_{train}$ respectively.

We calculate the row-wise and column-wise distances $\left\{\widehat{d}^2_{row}(i,i')\right\}_{i,i'\in[n]}$ and $\left\{\widehat{d}^2_{col}(j,j')\right\}_{j,j'\in[m]}$ from the training dataset. For practical purposes, we use the following definitions of distances

$$\widehat{d}^2_{row}(i,i') = \frac{\sum_{j\in[m]}(X_{i,j}-X_{i',j})^2 A_{i,j}A_{i',j}}{\sum_{j\in[m]}A_{i,j}A_{i',j}}, \tag{30}$$

$$\widehat{d}^2_{col}(j,j') = \frac{\sum_{i\in[n]}(X_{i,j}-X_{i,j'})^2 A_{i,j}A_{i,j'}}{\sum_{i\in[n]}A_{i,j}A_{i,j'}}.$$

Now let $\widehat{\theta}_{i,j,\boldsymbol{\eta}}$ be the TS-NN($\boldsymbol{\eta}$)'s estimate of $\theta_{i,j}$ for a specified threshold $\boldsymbol{\eta} = (\eta_{row}, \eta_{col})$, using the calculated distances. We compute the grid of t η_{row}'s and η_{col}'s using certain quantiles of $\left\{\widehat{d}^2_{row}(i,i')\right\}_{i,i'\in[n]}$ and $\left\{\widehat{d}^2_{col}(j,j')\right\}_{j,j'\in[m]}$ respectively. We denote it as the $\eta_{grid,row} := \{\eta_{1,row},\ldots,\eta_{t,row}\}$ and $\eta_{grid,col} := \{\eta_{1,col},\ldots,\eta_{t,col}\}$. Since percentiles of the distances are unaffected by the addition of the same term $2\sigma^2$ to all the distances, we don't calculate $\widehat{\sigma}^2$ for $\left\{\widehat{d}^2_{row}(i,i')\right\}_{i,i'\in[n]}$ and $\left\{\widehat{d}^2_{col}(j,j')\right\}_{j,j'\in[m]}$ and work with (30). Then we tune η in the training folds as follows:

$$\eta_{tuned} = \arg\min_{\boldsymbol{\eta}\in\eta_{grid,row}\times\eta_{grid,col}} \frac{\sum_{(i,j)\in\mathcal{F}_{train}}\left(Y_{i,j}-\widehat{\theta}_{i,j,\boldsymbol{\eta}}\right)^2 A_{i,j}}{\sum_{(i,j)\in\mathcal{F}_{train}}A_{i,j}}.$$

Then the test error is calculated as the mean squared error on the test fold, denoted as $\mathcal{F}_{test}$

$$\widehat{\sigma}^2_{test} = \frac{\sum_{(i,j)\in\mathcal{F}_{test}}\left(Y_{i,j}-\widehat{\theta}_{i,j,\boldsymbol{\eta}_{tuned}}\right)^2 A_{i,j}}{\sum_{(i,j)\in\mathcal{F}_{test}}A_{i,j}}.$$

We repeat this process 5 times, each time assigning a different fold as the test fold and report the average of the $\widehat{\sigma}^2_{test}$'s as the final test error in Fig. 1 and Fig. 3.

For the $\eta_{grid,row}, \eta_{grid,col}$, we work with the percentiles ranging from 1.5 to 10(same is true for DR-NN). For one-sided NNs, we expand the percentiles range to 1.5 - 30 to make them slightly more powerful and further highlight the importance of combining row and column neighbors for matrix estimation.

Fitting SoftImpute We fit SoftImpute using the R package `softImpute` [Mazumder et al. (2010)]. SoftImpute uses nuclear norm regularization and we fit SoftImpute with λ varying over a log grid from 1 to 12. Then we report the minimum MSE among all the MSEs obtained via SoftImpute for various λ's. We choose this grid as we found that the optimum lambda is almost always lied in the interior of this grid.

Runtime Complexity of NN based methods In terms of runtime complexity for a $n\times m$ matrix, Row - NN has $\mathcal{O}(\binom{n}{2}m+n) = \mathcal{O}(n^2m+n)$. First term arises as there are $\binom{n}{2}$ combinations of rows and computing L_2 distance between each pair of rows take $\mathcal{O}(m)$ time. The second term arises due to aggregation of $\mathcal{O}(n)$ terms. Similarly Col - NN has a runtime complexity of $\mathcal{O}(m^2n+m)$. Finally, TS-NN has a runtime complexity of $\mathcal{O}(m^2n+n^2m+mn)$. DR-NN can be expressed as a linear combination of Row-NN, Col-NN and TS-NN (Dwivedi et al., 2022b), hence it also has a runtime complexity of $\mathcal{O}(m^2n+n^2m+mn)$.

Row-NN algorithm Just for clarity, the Row-NN algorithm is outlined below:

- Compute the pairwise row distance estimates $\widehat{d}^2_{row}(i,j)$ for all $i,j\in[n]$ and use it to construct the neighborhood of row i,

  $$\mathcal{N}_{row}(i) = \{j\in[n] : \hat{d}^2_{row}(i,j)\le\eta_{row}\}.$$

  Here $\eta_{row}\ge 0$ is the tuning parameter.

- The estimate of $\theta_{i,t}$ is given by the row nearest neighbor estimate,

  $$\widehat{\theta}_{i,t} = \frac{\sum_{j\in\mathcal{N}_{row}(i)}X_{j,t}A_{j,t}}{|\mathcal{N}_{row}(i)|},$$

Col-NN has the column-counterpart algorithm of the above procedure.

D.2 Deffered details of real-life case study: HeartSteps

We will now provide additional details about the HeartSteps dataset.

We observe that at each decision time point, the HeartSteps algorithm determined whether a user is available based on certain attributes like whether the user was driving a car, etc. Other features are whether the user had an active connection at or around the decision time, was not in transit and phone was not in snooze mode. We focus on the matrix completion at the “available decision times”. After screening out the “non-avaliable times”, we see that only few users have > 210 decision times, resulting in filtering out the remaining columns/ decision times from consideration. Ultimately, we work with a dataset of 37 rows/users and 210 columns/decision times. For further details, we refer the readers to Klasnja et al. (2019).

For implementation, we use 5-fold blocked cross validation. HeartSteps’ underlying recommender algorithm uses a user’s history to sequentially assign interventions. So, there is a temporal dependence of the later columns on the previous columns. To tackle that, we first of all divide the the rows of the matrix into 5 folds. Now in each iteration we fix a fold, we hold out the entries in the last 40 columns of those rows as our test dataset. Remaining entries are used for training the NNs, USVT and SoftImpute. For NNs, we use the tuned η_{row} and η_{col}(or one of them for Row-NN and Col-NN) to complete the matrix. Then we report the difference between hold-out entries and their corresponding fitted estimates. We see the estimated matrices in SoftImpute and USVT resulted in extremely high test errors compared to NN-based methods. For $\eta_{grid,row}, \eta_{grid,col}$ in one-sided NNs and DR-NN, we work with the percentiles 25-85. Low percentiles were avoided as they gave the lowest training errors but the test errors were exorbitantly high. For TS-NN $\boldsymbol{\eta}$ grid, we consider percentiles 8 - 50.

D.3 Additional Plots

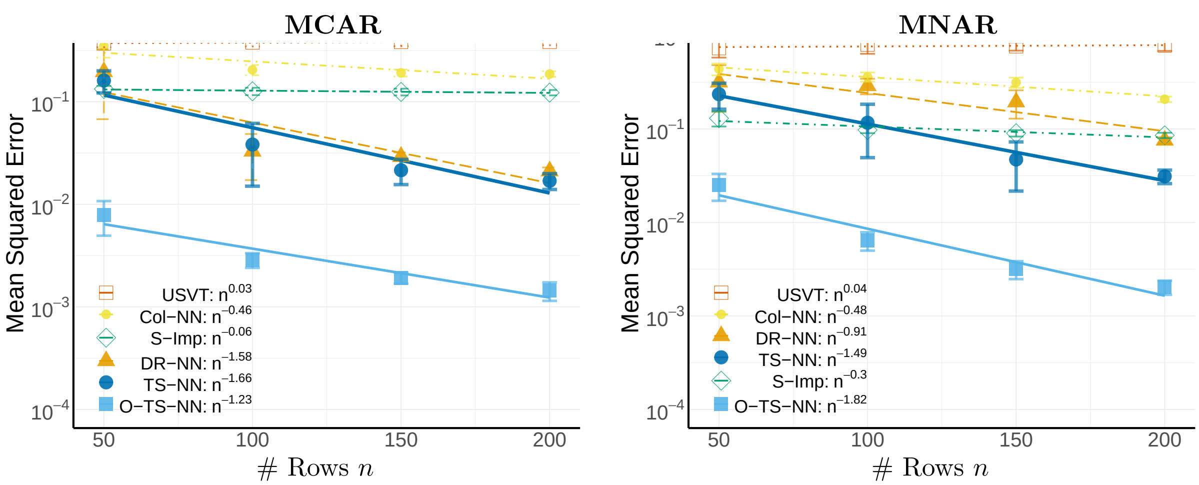


Figure 3: **MSE of different algorithms for estimating $\theta_{i,t}$ as a function of n when $\lambda = 0.5$ and SNR $= 2$.** TS-NN demonstrates quantifiable improvements over USVT, SoftImpute, and other NNs in estimation both in terms of MSE value and MSE decay rate. Moreover, TS-NN shows similar (if not better) MSE decay rates with n as compared to its oracle version in MCAR setup. Over here, we keep $n = m$ to keep the interpretation uncomplicated.