Graphical Kernel Ridge Regression in Latent Position Models

Martin GJORGJEVSKI[1] Nicolas KERIVEN[2] Simon BARTHELMÉ[1] Yohann DE CASTRO[3]

[1]GIPSA-Lab, CNRS 11 Rue de Mathématiques Grenoble, 38000, France

[2]IRISA, CNRS 263 av. du Général Leclerc Rennes, 35000, France

[3] Institut Universitaire de France Institut Camille Jordan École Centrale de Lyon 36 Avenue Guy de Collongue 69134 Écully, France

Résumé – Étant donné un graphe avec un signal sur les nœuds, nous nous intéressons à la prédiction des valeurs de signal manquantes à l'aide de la topologie du graphe et du signal disponible aux nœuds. Dans un travail précédent, nous avons montré qu'en modélisant le graphe selon un modèle de positions latentes, sous certaines hypothèses, l'estimation de la position latente associée à l'estimateur de Nadaraya-Watson permet d'atteindre des taux de convergence minimax lorsque la régularité du signal sous-jacent est faible. Dans ce travail, nous étudions une méthode basée sur la régularisation de la matrice d'adjacence, étroitement liée à la régression Kernel Ridge à partir d'une estimation non paramétrique, que nous appelons « Régression Graphique Kernel Ridge » (GKRR). Pour l'estimateur GKRR, nous démontrons des taux de convergence inférieurs aux taux KRR classiques. Nous soutenons qu'il s'agit d'un phénomène plus général : les algorithmes spectraux sur les LPM ont une variance strictement plus grande que les algorithmes spectraux classiques sur les matrices à noyau.

Abstract – Given a graph with signal on the nodes, we are interested in prediction of missing signal values using the graph topology and the available node signal. In a previous work we have shown that modeling the graph as Latent Position Model, under certain assumptions, latent position estimation along with the Nadaraya-Watson estimator can achieve minimax rates of convergence when the regularity of the underlying signal is low. In this work we consider a method based on regularizing the adjacency matrix, closely related with the Kernel Ridge Regression from nonparametric estimation, which we title Graphical Kernel Ridge Regression (GKRR). For the GKRR estimator we prove convergence rates that are slower than classical KRR rates. We argue that this is a more general phenomenon - spectral algorithms on LPMs have strictly larger variance than classical spectral algorithms on kernel matrices.

1 Introduction

We study the problem of *node regression* — given a graph $\mathcal{G}$ with vertex set $[n+1] = \{1, 2, \dots, n, n+1\}$, and signal $y_i \in \mathbb{R}$ on nodes $1 \leq i \leq n$, we want to infer a missing value $y_{n+1} \in \mathbb{R}$ using the observed signal $\boldsymbol{y}$ and the graph topology of $\mathcal{G}$. There are many approaches to this problem [2, 13, 20]. In recent years there has been a rapid development of *Graph Machine Learning* methods, including deep learning and representation learning [15, 17]. Nevertheless, there are relatively few works on theoretical comparison of algorithms and architectures, especially in the context of random graphs. In particular, traditional notions of statistical machine learning are underdeveloped in the context of random graphs. In this work we aim to expand upon this relatively understudied topic.

A statistical machine learning analysis in the context of graphs is problematic due to absence of classical notions of *empirical* and *true risk*. The Latent Position Model (LPM) [16] is a popular method to circumvent this issue. In this model it is assumed that the observed graph arises from an *unobserved point cloud* — each node i is represented by *an unobserved, hidden* vector $\boldsymbol{x}_i \in \mathbb{R}^d$.

In the LPM framework, both the graph $\mathcal{G}$ and the signal $\boldsymbol{y}$ can be assumed to depend on the latent positions $\mathbf{X}_{n+1} = [\boldsymbol{x}_1, \dots, \boldsymbol{x}_n, \boldsymbol{x}_{n+1}] \in \mathbb{R}^{d\times(n+1)}$, allowing for a comparison between problems in the graph learning and classical statistical learning setting. In a previous work [13] we have adopted this approach to the Nadaraya-Watson estimator, where we have shown that under certain conditions, position estimation algorithms in conjunction with the Nadaraya-Watson estimator can yield *optimal minimax* rates of convergence over the class of Hölder functions with exponent $\frac{1}{2} < a \leq 1$.

Another well known and well studied approach for classical regression is the **Kernel Ridge Regression (KRR)**, and it is natural to investigate the advantages and limitations of its LPM variant, which we title the **Graphical Kernel Ridge Regression (GKRR)**. The adjacency matrix $\mathbf{A}_{n+1}$ is given by

$$\mathbf{A}_{n+1} = \begin{bmatrix} \mathbf{A}_n & \boldsymbol{a}_{n+1} \\ \boldsymbol{a}_{n+1}^t & 1 \end{bmatrix} \in \{0,1\}^{(n+1)\times(n+1)} \quad (1)$$

where the $n \times n$ submatrix $\mathbf{A}_n$ contains the adjacency information about the labeled nodes $i \in [n]$, with ones on the diagonal and $\boldsymbol{a}_{n+1} \in \{0,1\}^n$ contains adjacency information for node $n+1$.

In this paper we will assume a special case of the LPM — that the adjacency matrix (1) is a *kernel matrix in expectation*:

$$\mathbb{E}\left[\mathbf{A}_{n+1}\right] = \mathbf{K}_{n+1} \quad (2)$$

where $\left[\mathbf{K}_{n+1}\right]_{i,j} = k(\boldsymbol{x}_i, \boldsymbol{x}_j)$ for $i, j \in [n+1]$, where $\boldsymbol{x}_i$ is the latent position of node i and k is a Positive Semi Definite (PSD) kernel with $k(\boldsymbol{x}, \boldsymbol{x}) = 1$. We will consider our *node regression problem* as a study of kernel methods with (entry-wise) noisy kernel matrices, where the noise has a particular structure. Indeed,

$$\mathbf{A}_{n+1} = \mathbf{K}_{n+1} + \mathbf{E}_{n+1} \quad (3)$$

where $\mathbf{E}_{n+1}$ is centered matrix with independent [1], albeit not

1. conditionally on the latent positions $\mathbf{X}_{n+1}$

identically distributed entries. Our aim is to explore the statistical consequences of using a noisy kernel matrix of the form (3) instead of classical kernel matrices $\mathbf{K}_{n+1}$. In this paper we limit ourselves to the most well known classical kernel method for regression, **Kernel Ridge Regression (KRR)**. In parallel with the adjacency matrix $\mathbf{A}_{n+1}$ 1, we can write

$$\mathbf{K}_{n+1} = \begin{bmatrix} \mathbf{K}_n & \boldsymbol{k}_{n+1} \\ \boldsymbol{k}_{n+1}^t & 1 \end{bmatrix} \in [0,1]^{(n+1)\times(n+1)} \tag{4}$$

The KRR estimator is constructed from (4) via

$$\hat{f}_\lambda(\boldsymbol{x}_{n+1}) = \frac{1}{n}\boldsymbol{k}_{n+1}^t(\bar{\mathbf{K}}_n + \lambda\mathbf{I})^{-1}\boldsymbol{y} \tag{5}$$

where $\bar{\mathbf{K}}_n = \frac{1}{n}\mathbf{K}_n$ and $\boldsymbol{k}_{n+1}$ are as in 4.

We study a node regression estimator which *plugs in* the entries of the matrix $\mathbf{A}_{n+1}$ (1) into the formula for KRR (5),

$$\hat{g}_\lambda(\boldsymbol{x}_{n+1}) = \frac{1}{n}\boldsymbol{a}_{n+1}^t\left(\bar{\mathbf{A}}_n + \lambda\mathbf{I}_n\right)^{-1}\boldsymbol{y} \tag{6}$$

where $\bar{\mathbf{A}}_n = \frac{1}{n}\mathbf{A}_n$ and $\boldsymbol{a}_{n+1}$ are as in (1). Inspired from Kernel Ridge Regression (KRR) estimator (5), we title the node regression estimator (6) the **Graphical Kernel Ridge (GKRR)**. In addition, we will consider the **oracle estimator**

$$\hat{h}_\lambda(\boldsymbol{x}) = \frac{1}{n}\boldsymbol{a}_{n+1}^t(\bar{\mathbf{K}}_n + \lambda\mathbf{I})^{-1}\boldsymbol{y} \tag{7}$$

This estimator has partial access to the kernel matrix $\mathbf{K}_{n+1}$ (4), namely to the principal $n \times n$ submatrix $\mathbf{K}_n$, but uses the adjacency vector $\boldsymbol{a}_{n+1}$ from $\mathbf{A}_{n+1}$ (1) for prediction of the missing value y_{n+1} of node $(n+1)$. Hence it may be considered a less stochastic version of GKRR (6).

Our contribution can be summarized as follows:

1. We show that the GKRR (6) converges, in a sense that takes into account the edge randomness in the graph. The convergence rate of GKRR is of order $n^{-\frac{r}{r+4}}$ where $1 \le r \le 2$ is a parameter related to the regularity of the signal.
2. We show that an oracle estimator $\hat{h}_\lambda$ (7) has larger variance than $\hat{f}_\lambda$. This oracle achieves a rate of $n^{-\frac{r}{r+2}}$, for $1 \le r \le 2$ which is still considered slow relative to KRR rates. Since GKRR $\hat{g}_\lambda$ (6) is a more stochastic version of $\hat{h}_\lambda$, this suggests that the rates of GKRR are also *slower* than those of KRR, i.e. the presence of noise (3) is detrimental to the statistical performance of KRR. Such negative results in this context are novel, to the best of our knowledge.

2 Background on LPMs

The Latent Position Model is a random graph model where each node i is represented by a vector $\boldsymbol{x}_i \in \mathbb{R}^d$, also known as a *latent position*. The vectors $\boldsymbol{x}_i$ can be treated either as random variables or as fixed, but unknown parameters, depending on the context. In both cases, they are assumed to be *unobserved*. A major interest in modeling network data as LPMs stems from the social science literature. The applications of LPMs are numerous — musical contests [8], legal [26] and illegal [3] trade, neuroscience [28], education research [24] , epidemiology [5], among others. See [18] for a review.

The LPM has also garnered attention in the theoretical statistics and machine learning literature. Most of the interest has been focused on latent position estimation [1, 12]. Convergence results for wide variety of Graph Neural Networks in this context were established by [6, 19].

In this paper, we will adopt the assumption of *Bernoulli* edges [19, 13]. Given the positions $\mathbf{X}_{n+1} = [\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n, \boldsymbol{x}_{n+1}]$, a graph with vertex set $[n+1]$ is sampled at random such that the entries $a_{i,j}$ of the adjacency matrix $\mathbf{A}_{n+1}$ (1) are given by

$$a_{i,j} = \mathbb{I}(u_{i,j} \le k(\boldsymbol{x}_i, \boldsymbol{x}_j)) \tag{8}$$

where $u_{i,j} = u_{j,i}$ are uniform on $[0,1]$ and are jointly independent for $i < j$, $i, j \in [n+1]$. Equation (8) allows us to define the *edge-randomness* in the graph, as contained entirely in the variables $\boldsymbol{\mathcal{U}}_{n+1}$.

3 Nonparametric regression

Suppose that $(\mathbf{X}_n, \boldsymbol{y}) = \{(\boldsymbol{x}_i, y_i) \colon i \in [n]\}$ are i.i.d. samples from $\mathcal{Q} \times \mathcal{Y}$, where $\mathcal{Q} \subseteq \mathbb{R}^d$ is a compact domain and $\mathcal{Y} \subseteq \mathbb{R}$. The relationship between y_i and $\boldsymbol{x}_i \sim \rho$ is modeled by

$$y_i = f(\boldsymbol{x}_i) + \epsilon_i \tag{9}$$

where ϵ_i are independent (among themselves and with $\mathbf{X}_n$), $\mathbb{E}[\epsilon_i] = 0$, $\mathbb{E}\left[\epsilon_i^2\right] = \sigma^2 < \infty$ and f is a *regression function*, such that $\int f^2(\boldsymbol{x}) d\rho(\boldsymbol{x}) < \infty$. The classical regression problem is formulated as constructing an estimate $\hat{f} = \hat{f}(\mathbf{X}_n, \boldsymbol{y}) \in L^2(\mathcal{Q}, \rho)$ based on the observed data $(\mathbf{X}_n, \boldsymbol{y})$ such that a specified performance metric is optimized. For regression, a classical choice is the *risk* $\mathcal{R}_f\left(\hat{f}\right)$, given by

$$\mathcal{R}_f\left(\hat{f}\right) = \mathbb{E}_{\boldsymbol{\epsilon}}\left[\int \left[\hat{f}(\boldsymbol{x}) - f(\boldsymbol{x})\right]^2 d\rho(\boldsymbol{x})\right]$$

To get rates of convergence, we need to consider a restricted class $\mathcal{F} \subseteq L^2(\mathcal{Q}, \rho)$ as a *hypothesis space* for the regression function f. A class $\mathcal{F}$ may be considered as a prior on the signal $\boldsymbol{y}$ (9) — it incorporates some (desired od expected) smoothness property of the regression function f in (9). For a given $\mathcal{F} \subseteq L^2(\mathcal{Q}, \rho)$ we consider the *risk over* $\mathcal{F}$

$$\mathcal{R}\left(\hat{f}, \mathcal{F}\right) \coloneqq \sup_{f \in \mathcal{F}} \mathcal{R}_f\left(\hat{f}\right) \tag{10}$$

Classical *nonparametric* regression literature typically assumes that $\mathcal{F}$ is a Hölder or Sobolev space [14, 25]. In these settings, the risk over $\mathcal{F}$ (10) is bounded by a *deterministic rate* r_n in a suitable sense, either with high probability over $\mathbf{X}_n$ [4] or in expectation over $\mathbf{X}_n$ [10]. In this paper we will consider the former case.

3.1 Learning with kernels

One fruitful approach to regression is via kernel methods [11, 23]. A Positive Semi Definite (PSD) kernel $k \colon \mathcal{Q} \times \mathcal{Q} \to \mathbb{R}$ is a continuous, symmetric function, such that for any $\boldsymbol{z}_1, \ldots, \boldsymbol{z}_m \in \mathcal{Q}$ and $s_1, \ldots, s_m \in \mathbb{R}$,

$$\sum_{i,j=1}^{m} s_i s_j k(\boldsymbol{z}_i, \boldsymbol{z}_j) \ge 0 \tag{11}$$

The main theoretical benefit of working with PSD kernels k (11) is a functional analytic construction titled **Reproducing Kernel Hilbert Space (RKHS)** [27]. Mercer's theorem [27] yields the spectral decomposition

$$k(\boldsymbol{x}, \boldsymbol{z}) = \sum_{j=1}^{\infty} t_j \phi_j(\boldsymbol{x}) \phi_j(\boldsymbol{z}) \tag{12}$$

where $(t_j)_{j=1}^{\infty}$ is *decreasing sequence* of nonnegative values converging to 0 and $\phi_j \in L^2(\mathcal{Q}, \rho)$ are orthonormal in $L^2(\mathcal{Q}, \rho)$. The *nonparametric case* is the assumption that $t_j > 0$ for all $j \in \mathbb{N}$. In this case, for any $r \geq 1$ the Hilbert space $\mathcal{H}_r = \left\{ \sum_{j=1}^{\infty} \theta_j \phi_j \in L^2(\mathcal{Q}, \rho) \colon \theta_j \in \mathbb{R}, \sum_{j=1}^{\infty} \frac{\theta_j^2}{t_j^r} < \infty \right\}$ with inner product

$$\left(\sum_{j=1}^{\infty} \theta_j \phi_j, \sum_{j=1}^{\infty} \tilde{\theta}_j \phi_j \right)_{\mathcal{H}_r} = \sum_{j=1}^{\infty} \frac{\theta_j \tilde{\theta}_j}{t_j^r}$$

is an infinite dimensional space of real-valued functions. In the special case $r = 1$, $\mathcal{H} = \mathcal{H}_1$ is a space of continuous functions satisfying the *reproducing property*: for all $\boldsymbol{x} \in \mathcal{Q}$, and for all $f \in \mathcal{H}$, $f(\boldsymbol{x}) = (f, k_{\boldsymbol{x}})_{\mathcal{H}}$with $k_{\boldsymbol{x}} = \sum_{j=1}^{\infty} t_j \phi_j(\boldsymbol{x}) \phi_j \in \mathcal{H}$. The assumption

$$f \in \mathcal{H}_r(R) \coloneqq \{ f \in \mathcal{H}_r \colon ||f||_{\mathcal{H}_r} \leq R \} \tag{13}$$

is known as **source condition**, a condition frequently used in the kernel learning literature [4, 10]. For $s \geq r \geq 1$, we have $\mathcal{H}_s(R) \subseteq \mathcal{H}_r(R)$, thus the higher the parameter r, the smaller the class $\mathcal{H}_r(R)$ and the easier the problem of bounding (10).

For technical reasons we will assume that $1 \leq r \leq 2$. Namely, KRR can exploit the regularity of the source condition (13) only for $1 \leq r \leq 2$, for any $r > 2$ one gets the same performance as with $r = 2$. This is known as the **saturation effect of KRR** [22].

3.2 Kernel Ridge Regression

The **Kernel Ridge Regression** (KRR) estimator (5) can be stated as optimization problem in $\mathcal{H}$, i.e.

$$\hat{f}_{\lambda} = \operatorname{argmin}_{h \in \mathcal{H}} \left[\frac{1}{n} \sum\nolimits_{i=1}^{n} (h(\boldsymbol{x}_i) - y_i)^2 + \lambda ||h||_{\mathcal{H}}^2 \right]$$

where $\lambda \geq 0$ is a hyperparameter that determines the trade-off between fitting to the data and penalizing *complex* functions — it is a bias-variance tradeoff parameter. Early studies of KRR [7, 9] characterize this tradeoff via an upper bound on the risk (10)

$$\mathcal{R}\left(\hat{f}_{\lambda}, \mathcal{H}_r(R)\right) \lesssim \lambda^r + \frac{1}{n\lambda^4} \tag{14}$$

with probability at least $1 - n^{-\eta}$ (over $\mathbf{X}_n$), where the sign $\lesssim$ hides away some constants that depend on various parameters (k, R, r, η and σ^2) and polynomials in $\log(n)$. When there are additional *smoothness* assumptions on k, so called **fast rates** are possible. For instance, [4] shows that when the *Mercer* decomposition (12) satisfies $t_j = \Theta\left(j^{-b}\right)$ for some $b > 1$, then we have the improved upper bound

$$\mathcal{R}\left(\hat{f}_{\lambda}, \mathcal{H}_r(R)\right) \lesssim \lambda^r + \frac{1}{n\lambda^{\frac{1}{b}}} \tag{15}$$

Since the inequalities (14) and (15) hold for all $\lambda > 0$ simultaneously, they may be optimized to conclude that the **KRR risk**

$$\mathcal{R}_{\mathrm{KRR}}(\mathcal{H}_r(R)) \coloneqq \inf_{\lambda > 0} \mathcal{R}\left(\hat{f}_{\lambda}, \mathcal{H}_r(R)\right) \tag{16}$$

is of order $\mathcal{R}_{\mathrm{KRR}}(\mathcal{H}_r(R)) \lesssim n^{-\frac{r}{r+4}}$ in the general setting of (14) — a rate known as a **slow rate**, and, when the eigenvalues follow polynomial decay, in the setting of (15), it is of order $\mathcal{R}_{\mathrm{KRR}}(\mathcal{H}_r(R)) \lesssim n^{-\frac{br}{br+1}}$ — a so called **fast rate** [2]. In Section 4.1 we will show that the GKRR (6) can achieve the *slow* rate in the general setting of (14). In Subsection 4.2 we will show that the oracle (7) $\hat{h}_{\lambda}$ can not meet the *fast* rate (15) which suggests that GKRR can not meet it as well.

4 Node Regression

We observe a latent position graph $\mathcal{G}$ on $n+1$ nodes, where edges are given by (8), where the kernel k is PSD (11). In addition, we observe a signal $\boldsymbol{y} = [y_1, \ldots, y_n] \in \mathbb{R}^n$ given by (9) where $f \in \mathcal{H}_r(R)$ (13) with $1 \leq r \leq 2$. We do *not observe* the kernel matrix $\mathbf{K}_{n+1}$ or the latent positions $\mathbf{X}_{n+1}$. We will define a notion of risk analogous to (10), but that takes the random edges $\boldsymbol{\mathcal{U}}_{n+1}$ (8) into account. In this framework, an estimator $\hat{g}(\boldsymbol{x}_{n+1})$ for $f(\boldsymbol{x}_{n+1})$ *must be* constructed from the adjacency matrix $\mathbf{A}_{n+1}$ (1) and the observed signal $\boldsymbol{y}$ (9). Note that such estimators $\hat{g}_{\lambda}$ depend on the signal $\boldsymbol{y}$, the latent positions $\mathbf{X}_{n+1}$ *and* the random edges $\boldsymbol{\mathcal{U}}_{n+1}$ (8). In analogy with the nonparametric regression risk, we will consider

$$\begin{aligned} \mathcal{R}_f^{\mathrm{G}}(\hat{g}) &\coloneqq \mathbb{E}_{\boldsymbol{u}_{n+1}} \left[\mathcal{R}_f(\hat{g}) \right] \\ &= \mathbb{E}_{\boldsymbol{u}_{n+1}, \boldsymbol{x}_{n+1}, \epsilon} \left[\left(\hat{g}(\boldsymbol{x}_{n+1}) - f(\boldsymbol{x}_{n+1}) \right)^2 \right] \end{aligned}$$

the average risk over *both* the latent position $\boldsymbol{x}_{n+1}$ *and* the random edge vector $\boldsymbol{u}_{n+1}$, and

$$\mathcal{R}^{\mathrm{G}}(\hat{g}, \mathcal{F}) \coloneqq \sup_{f \in \mathcal{F}} \mathcal{R}_f^{\mathrm{G}}(\hat{g}) \tag{17}$$

We will work with the source condition (13), $\mathcal{F} = \mathcal{H}_r(R)$.

4.1 Graphical Kernel Ridge (GKRR)

The adjacency matrix $\bar{\mathbf{A}}_n$ in general *has negative eigenvalues*, and therefore (6) needs to be treated carefully. Indeed, $\bar{\mathbf{A}}_n + \lambda \mathbf{I}$ is invertible iff $\lambda > 0$ is not an eigenvalue of $-\bar{\mathbf{A}}_n$. The best possible performance of GKRR is

$$\mathcal{R}_{\mathrm{KRR}}^{\mathrm{G}}(\mathcal{H}_r(R)) \coloneqq \inf_{\lambda \notin \sigma\left(-\bar{\mathbf{A}}_n\right)} \mathcal{R}^{\mathrm{G}}(\hat{g}_{\lambda}, \mathcal{H}_r(R)) \tag{18}$$

Since we are learning in the LPM setting via $\mathbf{A}_{n+1}$ and $\boldsymbol{y}$, we have strictly less information than in the kernel setting, and hence, we expect that the risk $\mathcal{R}_{\mathrm{KRR}}^{\mathrm{G}}$ (18) for GKRR is at least as the risk in the nonparametric setting $\mathcal{R}^{\mathrm{KRR}}$ for KRR, i.e. $\mathcal{R}_{\mathrm{KRR}} \lesssim \mathcal{R}_{\mathrm{KRR}}^{\mathrm{G}}$. We have the following upper bound.

Theorem 4.1 *Suppose that the source condition* (13) *holds. Given $\eta > 0$, there exists C_{η} such that for any $\lambda \geq \frac{2C_{\eta}}{\sqrt{n}}$,*

$$\mathcal{R}^{\mathrm{G}}(\hat{g}_{\lambda}, \mathcal{H}_r(R)) \lesssim \lambda^r + \frac{1}{n\lambda^4}$$

2. Note that for $b > 1$ and $r \geq 1$ we have $\frac{br}{br+1} > \frac{1}{2}$, so this rate is always faster than $n^{-1/2}$

with probability at least $1 - 2n^{-\eta}$ *over the random edges* $\boldsymbol{\mathcal{U}}_n$ *and the latent variables* $\mathbf{X}_n$.

The proof of Theorem 4.1 is based on a bound of the spectral norm of $\bar{\mathbf{A}}_n - \bar{\mathbf{K}}_n$ in [21]. The GKRR rate 4.1 matches the KRR rate (14). In particular, it implies that $\mathcal{R}^{\mathrm{G}}_{\mathrm{KRR}}\left(\mathcal{H}_r\left(R\right)\right) \lesssim n^{-\frac{r}{r+4}}$. In the KRR literature, this is the rate associated with the most general assumptions on the kernel $\mathbf{K}_{n+1}$ (5), and even in this case it is known that better rates are possible [29]. At the moment we do not have a proof that a tighter upper bound is impossible, so there might be some room for improvement of Theorem 4.1. However, we have a strong argument that GKRR can not achieve the *fast rates* (15) which can be potentially extended to a broader class of spectral regularization estimators. We discuss these arguments in Subsection 4.2.

4.2 Oracle estimator lower bound

The following result suggests that fast rates are not possible for the GKRR estimator.

Proposition 1 *Recall the* ***oracle estimator*** (7) *is given by*

$$\hat{h}_\lambda(\boldsymbol{x}) = \frac{1}{n}\boldsymbol{a}^t_{n+1}\left(\bar{\mathbf{K}}_n + \lambda\mathbf{I}\right)^{-1}\boldsymbol{y}$$

There exists $k_c, K_c > 0$ *such that for all* $\mathbf{X}_n$,

$$\frac{k_c\sigma^2}{1+\lambda+n\lambda^2} \leq \mathcal{R}^{\mathrm{G}}_f\left(\hat{h}_\lambda\right) - \mathcal{R}_f\left(\hat{f}_\lambda\right) \leq \frac{K_c(R^2+\sigma^2)}{1+\lambda+n\lambda^2}$$

Proposition 1 shows that the *oracle estimator* $\hat{h}_\lambda$ which has access to the kernel matrix $\mathbf{K}_n$ (5) needs regularization $\lambda = \omega(n^{-\frac{1}{2}})$ to have (asymptotically) negligible excess risk compared to the KRR estimator. Comparing the result to the bound (15), which needs $\lambda = \mathrm{o}(1)$ and $\lambda = \omega(n^{-b})$, $b > 1$, we see that $\hat{h}_\lambda$ requires much stronger regularization for convergence. Proposition 1 also shows that the oracle $\hat{h}_\lambda$ converges at rate $n^{-\frac{r}{r+2}}$, which is considered slow in comparison to KRR. Indeed, for the highest regularity $r = 2$, $\hat{h}_\lambda$ converges at rate $n^{-\frac{1}{2}}$, whereas *fast rates* from 15 are always in $\mathrm{o}(n^{-\frac{1}{2}})$. Since GKKR (6) is a noisier version of the oracle $\hat{h}_\lambda$, we expect that its rate is not going to be better than the oracle i.e. of order $n^{-\frac{r}{r+2}}$, although currently we do not have a proof of this conjecture.

5 Discussion

In a previous work [13] we have shown that in the framework (3) under certain conditions it is possible to achieve optimal rates when the prior is that f is continuous but not necessarily with smooth derivatives.

This work hints that it is more challenging to exploit higher order of smoothness in the regression function. While the presented method GKRR converges (Theorem 4.1), it is substantially slower than its counterpart on kernel matrices, KRR (15). The bound in Proposition 1 is specific to KRR, but we suspect that this behavior extends to a broader class of spectral methods. Indeed, some **Random Matrix Theory (RMT)** results indicate that spectral information of $\mathbf{K}_n$ is *lost in the bulk* — only a small fraction of the spectrum of $\mathbf{K}_n$ can be recovered from $\mathbf{A}_n$ and therefore it is reasonable that a broader family of spectral kernel methods such as *KPCR, Landweber iteration, gradient descent with early stopping* [9, 10] are facing the same limitations as GKRR. Minimax rates in this framework as well as construction of more efficient algorithms are left as an open problem for future work.

References

[1] Ery Arias-Castro, Antoine Channarond, Bruno Pelletier, and Nicolas Verzelen. On the estimation of latent distances using graph distances, 2020.

[2] M. Belkin, I. Matveeva, and P. Niyogi. Tikhonov regularization and semi-supervised learning on large graphs. In *2004 IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 3, pages iii–1000, 2004.

[3] Giulia Berlusconi, Alberto Aziani, and Luca Giommoni. The determinants of heroin flows in europe: A latent space approach. *Social Networks*, 51:104–117, October 2017. Crime and Networks.

[4] A. Caponnetto and E. Vito. Optimal rates for the regularized least-squares algorithm. *Found. Comput. Math.*, 7(3):331–368, July 2007.

[5] A.M Chu, T.W. Chan, M.K So, and W-K. Wong. Dynamic network analysis of covid-19 with a latent pandemic space model. *International Journal of Environmental Research and Public Health*, 18(6):3195, 2021.

[6] Matthieu Cordonnier, Nicolas Keriven, Nicolas Tremblay, and Samuel Vaiter. Convergence of message-passing graph neural networks with generic aggregation on large random graphs. *Journal of Machine Learning Research*, 25(406):1–49, 2024.

[7] Cucker and Smale. Best choices for regularization parameters in learning theory: On the bias—variance problem. *Foundations of Computational Mathematics*, 2:413–428, 03 2008.

[8] Silvia D'Angelo, Thomas Brendan Murphy, and Marco Alfò. Latent space modelling of multidimensional networks with application to the exchange of votes in eurovision song contest. *The Annals of Applied Statistics*, 2018.

[9] Ernesto de Vito, Lorenzo Rosasco, and Alessandro Verri. Spectral methods for regularization in learning theory. 2006.

[10] Lee H. Dicker, Dean P. Foster, and Daniel Hsu. Kernel ridge vs. principal component regression: minimax bounds and adaptability of regularization operators, 2016.

[11] L. Gerfo, Lorenzo Rosasco, Francesca Odone, Ernesto De Vito, and Alessandro Verri. Spectral algorithms for supervised learning. *Neural Computation*, 20:1873–1897, 07 2008.

[12] Christophe Giraud, Yann Issartel, and Nicolas Verzelen. Localization in 1d non-parametric latent space models from pairwise affinities, 2023.

[13] Martin Gjorgjevski, Nicolas Keriven, Simon Barthelmé, and Yohann De Castro. Node regression on latent position random graphs via local averaging, 2024.

[14] László Györfi, Michael Kohler, Adam Krzyżak, and Harro Walk. A distribution-free theory of nonparametric regression. In *Springer Series in Statistics*, 2002.

[15] William L. Hamilton. Graph representation learning. *Synthesis Lectures on Artificial Intelligence and Machine Learning*, 14(3):1–159.

[16] Peter D Hoff, Adrian E Raftery, and Mark S Handcock. Latent space approaches to social network analysis. *Journal of the American Statistical Association*, 97(460):1090–1098, 2002.

[17] Wei Ju, Zheng Fang, Yiyang Gu, Zequn Liu, Qingqing Long, Ziyue Qiao, Yifang Qin, Jianhao Shen, Fang Sun, Zhiping Xiao, Junwei Yang, Jingyang Yuan, Yusheng Zhao, Yifan Wang, Xiao Luo, and Ming Zhang. A comprehensive survey on deep graph representation learning. *Neural Networks*, 173:106207, May 2024.

[18] Hardeep Kaur, Riccardo Rastelli, Nial Friel, and Adrian E. Raftery. Latent position network models, 2023.

[19] Nicolas Keriven, Alberto Bietti, and Samuel Vaiter. Convergence and stability of graph convolutional networks on large random graphs, 2020.

[20] Arne Kovac and Andrew D. A. C. Smith. Regression on a graph, 2009.

[21] Jing Lei and Alessandro Rinaldo. Consistency of spectral clustering in stochastic block models. *The Annals of Statistics*, 43(1), February 2015.

[22] Yicheng Li, Haobo Zhang, and Qian Lin. On the saturation effect of kernel ridge regression, 2024.

[23] Mehryar Mohri, Afshin Rostamizadeh, and Ameet Talwalkar. *Foundations of Machine Learning*. The MIT Press, 2012.

[24] T.M. Sweet, A.C. Thomas, and B.W. Junker. Hierarchical network models for education research: Hierarchical latent space models. *Journal of Educational and Behavioral Statistics*, 38(3):295–318, 2013.

[25] Alexandre B. Tsybakov. *Introduction to Nonparametric Estimation*. Springer Publishing Company, Incorporated, 1st edition, 2008.

[26] MICHAEL D. WARD, JOHN S. AHLQUIST, and ARTURAS ROZENAS. Gravity's rainbow: A dynamic latent space model for the world trade network. *Network Science*, 1(1):95–118, 2013.

[27] Holger Wendland. *Scattered Data Approximation*. Cambridge Monographs on Applied and Computational Mathematics. Cambridge University Press, 2004.

[28] J.D. Wilson, S. Cranmer, and Z.-L. Lu. A hierarchical latent space network model for population studies of functional connectivity. *Computational Brain and Behavior*, 3(4):384–399, 2020.

[29] Tong Zhang. Learning bounds for kernel regression using effective data dimensionality. *Neural Computation*, 17(9):2077–2098, 09 2005.