Journal of Machine Learning Research 27 (2026) 1-49 Submitted 10/24; Revised 1/26; Published 4/26

Node Regression on Latent Position Random Graphs via Local Averaging

Martin Gjorgjevski MARTIN.GJORGJEVSKI@GRENOBLE-INP.FR
GIPSA-Lab, CNRS
11 Rue de Mathématiques
Grenoble, 38000, France

Nicolas Keriven NICOLAS.KERIVEN@CNRS.FR
IRISA, CNRS
263 av. du Général Leclerc
Rennes, 35000, France

Simon Barthelmé SIMON.BARTHELME@GRENOBLE-INP.FR
GIPSA-Lab, CNRS
11 Rue de Mathématiques
Grenoble, 38000, France

Yohann De Castro YOHANN.DE-CASTRO@EC-LYON.FR
Institut Universitaire de France
Institut Camille Jordan
École Centrale de Lyon
36 Avenue Guy de Collongue
69134 Écully, France

Editor: Bharath Sriperumbudur

Abstract

Node regression consists in predicting the value of a graph label at a node, given observations at the other nodes. We perform a theoretical study where the graph is generated by a Latent Position Model: each node has a latent position and the probability of connection depends on the distance between latent positions.

We begin by studying the simplest estimator: averaging the label at all neighboring nodes. We show that in Latent Position Models this estimator tends to a Nadaraya-Watson estimator in the latent space, with the same rate of convergence.

One issue with this estimator is that it averages over all neighbors of a node, which may be too large or too small a region depending on the graph model. An alternative consists in first estimating the "true" distances between latent positions, then injecting these into a classical Nadaraya-Watson estimator. This enables averaging in regions either smaller or larger than the typical graph neighborhood. We show that this method can achieve standard nonparametric rates even when the graph neighborhood is too large or too small.

Keywords: generalization bounds, graph machine learning, random graphs, nonparametric statistics, kernel regression

1. Introduction

Given an undirected graph with $n+1$ vertices and an adjacency matrix $\mathbf{A} = [a_{i,j}]$ where all but the $(n+1)$-st node have labels y_i, the node regression problem addresses the prediction of the (continuous valued) label y_{n+1} of the remaining node[1]. This framework represents a simplified version of the so-called *transductive Semi-Supervised Learning (SSL)* problem on graphs (Song et al., 2021), where the labels of some nodes on a graph are known and the goal is to predict the labels of the other nodes in the same graph. While some SSL theoretical works focus on how to best exploit a *large* quantity of unlabeled nodes, this simplified framework with only one unlabeled node is closer to classical Machine Learning, where generalization is computed (in expectation) for one unknown sample only. As we will see, this allows us to draw parallels between Graph Machine Learning (ML) and "regular" ML, and better isolate the effects of the graph structure on the problem. Despite the vastness of the Graph ML literature, this simplified framework has rarely been studied. While there are numerous works on unsupervised tasks such as node clustering (Athreya et al., 2018; Abbe, 2018) and community detection (Fortunato, 2010; Zhao et al., 2012), supervised tasks have received less attention in this framework. To our knowledge, the only authors to study this framework are Tang et al. (2013), within the context of Reproducing Kernel Hilbert Spaces. Here we will study an even simpler, arguably more foundational approach: a simple 1-hop averaging, mimicking the classical Nadaraya-Watson estimator in the graph context.

We consider the node regression problem with the goal of establishing generalization bounds in the context of random graphs. Specifically, we work with the **Latent Position Model** (LPM) (Hoff et al., 2002), where each node i is associated to a *latent, unknown* variable $\boldsymbol{x}_i \in Q \subseteq \mathbb{R}^d$. An edge between nodes i and j occurs with a probability that depends on the distance $\|\boldsymbol{x}_i - \boldsymbol{x}_j\|$ of the latent positions of nodes i and j, and occurrences are independent conditionally on the latent positions. Like often in the literature, our random graph model will essentially depend on a parameter that we call the **length-scale** h_g. This represents the "typical scale" of the model[2]: intuitively speaking, nodes with latent positions at distance below length-scale h_g are highly likely to be connected, and vice-versa. As mentioned above, in addition to the graph, we observe continuous labels y_i on the first n nodes of the graph. We will assume that the labels y_i are noisy observations of some deterministic function of the latent positions $\boldsymbol{x}_i$, allowing for a direct comparison between node regression and classical (nonparametric) regression.

Graphical Nadaraya-Watson Estimator (GNW) In this paper, we focus on a simple (arguably, the simplest non-trivial) estimator for the missing label of node $n+1$, which computes the average of the labels over all of its neighbors, i.e.,

$$\hat{y}_{n+1} = \frac{\sum_{j=1}^{n} y_j a_{j,n+1}}{\sum_{j=1}^{n} a_{j,n+1}} \tag{1}$$

The estimator (1) resembles the *Nadaraya-Watson* (NW) estimator, a fundamental estimator for nonparametric regression (Tsybakov, 2008), but where the "soft" distance kernel

1. Our assumption that the regression node is numbered node $n+1$ is made purely out of notational convenience
2. It may be found under other names in the literature, e.g. "kernel bandwidth".

$k(\boldsymbol{x}_{n+1}, \boldsymbol{x}_i)$ usually computed in NW is here replaced by the graph edges (recall that the $\boldsymbol{x}_i$'s are unknown in our context). Therefore, we decide to call estimator (1) the **Graphical Nadaraya-Watson** (GNW) estimator. Note that, although we had to pick a name for the estimator (1) because to our knowledge it did not bear any particular name as a standalone estimator, the "1-hop averaging" principle is of course far from new and appears in many contexts (e.g. as an aggregation function in Graph Neural Networks (Kipf and Welling, 2017)).

As we will see, the hidden geometrical structure of the LPM allows us to study and compare the GNW estimator with the classical NW estimator with techniques from classical nonparametric regression. There is however one major difference between the two. In the classical nonparametric settings, the statistician is free to select a parameter of NW known as the *bandwidth* τ, which sets the spatial scale over which the NW estimator performs averaging, e.g. through a kernel $\phi\left(\frac{\|\boldsymbol{x}_i - \boldsymbol{x}_j\|}{\tau}\right)$ with a decreasing function ϕ. Thus, in the classical literature of nonparametric regression, for a given regression function f (or more generally a function class $\mathcal{F}$) and a noise level σ^2, there exists an optimal bandwidth $\tau_\star$ that minimizes the risk. On the contrary, in our setting, the neighborhood "size" is imposed by the graph, which depends on a **length-scale** h_g *that is not user-chosen.* The length-scale h_g, like the latent positions themselves, is assumed to be *latent (unknown) and fixed* parameter of the LPM.

Note that the GNW estimator as given by (1) depends on unknown parameters of the LPM such as the positions $\boldsymbol{x}_i \in Q \subseteq \mathbb{R}^d$, $i \in [n+1]$ and the length-scale $h_g > 0$. In Sec. 2.3 we show that the error[3] of the GNW estimator with length-scale h_g is surprisingly comparable to that of a NW estimator with *fixed bandwidth* $\tau := h_g$. In other words, replacing a fixed kernel with the corresponding Bernoulli variables does not (asymptotically) degrade the performance of the estimator. As a consequence, GNW is nearly optimal if the length-scale h_g of the LPM is sufficiently close to the optimal bandwidth $\tau_\star$ for the corresponding nonparametric regression problem.

On the other hand, if this is not the case, if the length-scale h_g is far away from the ideal bandwidth $\tau_\star$, GNW will perform poorly. Namely, there are two unfavorable scenarios for GNW - the *under averaging regime* $h_g \ll \tau_\star$ (node averaging is performed on a scale significantly smaller than the optimal one) and the *over averaging regime* $h_g \gg \tau_\star$ (node averaging performed on a scale too large relative to the underlying label). In order to address this problem, in the second part of the paper we study a broad family of estimators collectively titled the **Estimated Nadaraya-Watson** (ENW) estimator.

Estimated Nadaraya-Watson Estimator (ENW) In the aforementioned regimes of under-averaging ($h_g \ll \tau_\star$) and over-averaging ($h_g \gg \tau_\star$), the GNW estimator (1) is sub-optimal. This suboptimality stems from the fact that the node neighborhood given by the graph *does not translate to an appropriate* neighborhood in the latent space. A natural approach is therefore, to attempt to construct a more appropriate neighborhood, based on the topology of the graph and the observed signal.

The construction of this neighborhood is divided in two stages. The first stage is a *distance recovery algorithm* $\mathcal{A}$, that is, an algorithm that estimates the latent distances $\|\boldsymbol{x}_i - \boldsymbol{x}_j\|$, based on the observed adjacency matrix $\mathbf{A}$. The second stage simply uses the

3. For an appropriately defined notion of risk, adapted to Latent Position Models

approximated distances to compute the regular Nadaraya-Watson estimator with tunable bandwidth τ in the hope that, if the estimated distances are sufficiently close to the true ones, then the optimal bandwidth $\tau_\star$ (approximately known or, in practice, estimated by cross-validation) leads to a better result than the previous GNW. We call this estimation procedure the $\mathcal{A}$-**Estimated Nadaraya-Watson** estimator ($\mathcal{A}$-ENW), where the adjective *estimated* refers to the distances between the latent positions. In Section 3, our theoretical analysis will decouple these stages, allowing for separate treatment of the two problems. Our contribution is with regard to the second step, that is, the stability of the Nadaraya-Watson estimator to perturbations of the distances between the design points. We provide a risk bound for $\mathcal{A}$-ENW in terms of the probability of the algorithm $\mathcal{A}$ to land within a prescribed tolerance level of the true positions. Concerning the algorithm $\mathcal{A}$ itself, we do not make any novel contribution *per se* (as this is slightly out-of-scope here), but we build on some existing algorithms $\mathcal{A}$ from the literature and point out instances in which $\mathcal{A}$-ENW outperforms GNW both in the *under averaging* ($h_g \ll \tau_\star$) and in the *over averaging* ($h_g \gg \tau_\star$) regimes. In particular, in some instances we can achieve *standard nonparametric rates* from classical nonparametric regression literature.

1.1 Background on the Latent Position Model

A random graph consists of a vertex set $V = [n]$ and a *random* edge set $\mathcal{E} \subseteq V \times V$. The study of random graphs begins with the Erdös-Renyi model (Gilbert, 1959), where each edge occurs independently with probability $0 \leq p \leq 1$. However, real world networks do not have the same distributional properties as the Erdös-Renyi model, for example it has been observed that the distribution of degrees in real world networks follows a power law (Albert and Barabási, 2002). Such observations prompted research into models that can better capture the topology of real world networks, yielding richer models of random graphs. One such model for studying community structure is called the Stochastic Block Model (Holland et al., 1983). Here nodes belong to latent communities and the probability that two nodes i, j are linked depends only on the communities of the nodes C_i, C_j. This model has been studied extensively from a theoretical point of view (Abbe, 2018). Another popular model is the random geometric graph (Penrose, 2003), where nodes are associated to *latent positions* $\boldsymbol{x}_i$. Here nodes i and j are linked if the distance between their latent positions $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ is within a prescribed threshold $r > 0$, i.e. if $\|\boldsymbol{x}_i - \boldsymbol{x}_j\|_2 < r$. These two models can be unified in the Latent Position Model (Hoff et al., 2002). In this model each node i is associated to a latent position[4] $\boldsymbol{x}_i \in Q$, where Q is the *latent space*. The probability of having an edge between nodes i and j is then given by

$$\mathbb{P}\left(a_{i,j} = 1 | \boldsymbol{x}_i, \boldsymbol{x}_j\right) = k(\boldsymbol{x}_i, \boldsymbol{x}_j) \tag{2}$$

where $k \colon Q \times Q \to [0, 1]$ is (in general) a nonparametric *link* function.

To relate the node regression problem with the classical theory of (non-parametric) regression, we will suppose that $Q \subseteq \mathbb{R}^d$ and that the link function has the following shape

$$\mathbb{P}\left(a_{i,j} = 1 | \boldsymbol{x}_i, \boldsymbol{x}_j\right) = k(\boldsymbol{x}_i, \boldsymbol{x}_j) = \alpha K\left(\frac{\|\boldsymbol{x}_i - \boldsymbol{x}_j\|}{h_g}\right) \tag{3}$$

4. these positions may be deterministic or i.i.d. draws from some distribution

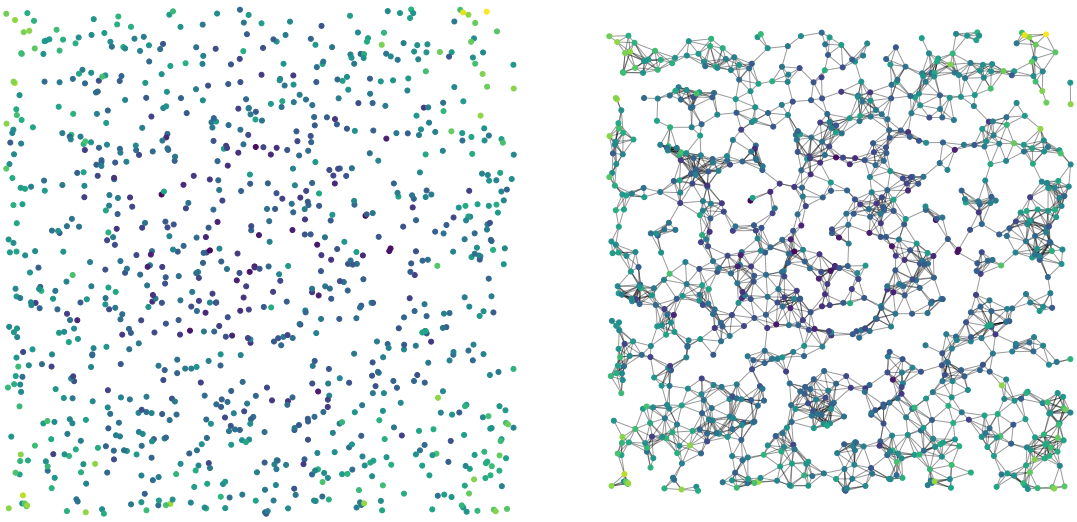


Figure 1: Sampling a LPM. Left: latent positions drawn uniformly on $[-1,1]^2$, with colors representing label values (brighter means higher). Right: the corresponding random geometric graph with $h_g = 0.1$.

Here $0 < \alpha \leq 1$, $h_g > 0$ and $K\colon [0,\infty) \to [0,1]$ with $K(0) = 1$. When we observe a LPM graph with link function (3), we do not assume to know anything but the graph itself: we **do not have access** to the latent positions $\boldsymbol{x}_i, i \in [n]$, nor to the parameters α, h_g. The function K is in general assumed to be decreasing, and we will add the assumption that it is compactly supported and non-vanishing in a neighborhood of 0 (see Assumption 8). In the literature, the parameters α, h_g are generally used to model *sparsity* in random graphs, that is, the relative number of edges with respect to n. Loosely speaking, the number of edges of a LPM sampled graph with n nodes, latent dimension d and parameters α, h_g is expected to scale like $\sum_{1\leq i<j\leq n} a_{i,j} \sim \alpha h_g^d n^2$. For large n asymptotics, when α, h_g are fixed relative to n, the expected number of edges is in $\Omega(n^2)$, and the random graph is said to be *dense*. When the number of edges is in $\mathcal{O}(n)$, the graph is *sparse*. In-between those two rates, the graph is *relatively sparse*. Most real-world graphs are observed to be relatively sparse or sparse. To model this, taking a decreasing multiplicative factor α when n increases is e.g. more common in the SBM literature, while using a decreasing length-scale h_g is more common in the geometric graph literature. For the GNW, we take both parameters h_g and α into account, and we show that our results hold for any relatively sparse graph, as soon as the expected degrees grow with the number of nodes, *even if this growth is arbitrarily slow*. For $\mathcal{A}$-ENW, we fix $\alpha = 1$ in order to facilitate the analysis; we leave the case of α decreasing with n for future work.

1.2 Framework and Notation

We denote the indicator of a set S by $\mathbb{I}[S]$, the Lebesgue measure on $\mathbb{R}^d$ by m and the volume of the unit ball in $\mathbb{R}^d$ by v_d. The standard Euclidean distance between $\boldsymbol{x}, \boldsymbol{z} \in \mathbb{R}^d$

is denoted by $\|\boldsymbol{x}-\boldsymbol{z}\|$. We use the comparisons operators $a_n \lesssim b_n$ when there exists some constant $c>0$ independent of n and h_g such that $ca_n \leq b_n$, and we use $\gtrsim$ in an analogous manner. Similarly, we use standard Big O notation which is by default assumed to be in relation to the node size n, unless explicitly stated otherwise.

We introduce the notation

$$\mathbf{X}_n = [\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n] \in \mathbb{R}^{d\times n} \tag{4}$$

to denote the matrix that contains the latent positions of nodes 1 through n (the labeled nodes), and

$$\mathbf{X}_{n+1} = [\mathbf{X}_n, \boldsymbol{x}_{n+1}] = [\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n, \boldsymbol{x}_{n+1}] \in \mathbb{R}^{d\times(n+1)} \tag{5}$$

to denote the extended matrix that contains the latent position of the regression node, node $(n+1)$. The observed label $\boldsymbol{y} = [y_1, \ldots, y_n]^t$ is given by

$$y_i = f(\boldsymbol{x}_i) + \epsilon_i \tag{6}$$

where $f\colon Q \to \mathbb{R}$ is a regression function that belongs to a Hölder class (See Assumption 9) and

$$\boldsymbol{\epsilon} = [\epsilon_1, \ldots, \epsilon_n]^t \tag{7}$$

is the label additive noise vector with independent entries of finite variance (See Assumption 2). An LPM graph can be represented by the $(n+1)\times(n+1)$ adjacency matrix $\mathbf{A} = \left[a(\boldsymbol{x}_i, \boldsymbol{x}_j)\right]_{1\leq i,j\leq n+1}$, where the indicator of an edge between nodes i and j is given by

$$a(\boldsymbol{x}_i, \boldsymbol{x}_j) = \mathbb{I}\left[U_{i,j} \leq k(\boldsymbol{x}_i, \boldsymbol{x}_j)\right] \tag{8}$$

where

$$\boldsymbol{\mathcal{U}} = \left[U_{i,j}\right]_{1\leq i,j\leq (n+1)} \tag{9}$$

are uniform variables on $[0,1]$, for $i \neq j$ and $U_{i,i} = c > 1$ (by convention, this prevents self edges). The entries of $\boldsymbol{\mathcal{U}}$ are independent for distinct pairs (i_1, j_1) and (i_2, j_2) with $i_1 < j_1$ and $i_2 < j_2$, and satisfying the symmetric constraint $U_{i,j} = U_{j,i}$ (imposed by the symmetry of the adjacency matrix $\mathbf{A}$). The matrix $\boldsymbol{\mathcal{U}}$ is also independent of $\mathbf{X}_{n+1}$ and $\boldsymbol{\epsilon}$. Throughout the paper, we will assume that the latent positions are either fixed or they are i.i.d. samples with density p with support $Q \subseteq \mathbb{R}^d$. In the latter case, the local edge density and the local expected degree at a point $\boldsymbol{x} \in \mathbb{R}^d$ are given by

$$c(\boldsymbol{x}) = \int_{\mathbb{R}^d} k(\boldsymbol{x}, \boldsymbol{z}) p(\boldsymbol{z}) dz, \qquad d(\boldsymbol{x}) = nc(\boldsymbol{x}) \tag{10}$$

respectively. For $\boldsymbol{x} \in \mathbb{R}^d$, we define the operator $S(\cdot, \boldsymbol{x})$ on the set of bounded and measurable functions by

$$S(f, \boldsymbol{x}) = \begin{cases} \frac{\int f(\boldsymbol{z}) k(\boldsymbol{x}, \boldsymbol{z}) p(\boldsymbol{z}) dz}{c(\boldsymbol{x})} & \text{if } c(\boldsymbol{x}) > 0 \\ 0 & \text{otherwise} \end{cases} \tag{11}$$

Furthermore, we denote by

$$\delta_i = \|\boldsymbol{x}_i - \boldsymbol{x}_{n+1}\| \tag{12}$$

the distance between the ith latent variable and the one of the node of interest $n+1$.

1.3 Differences between Classical Nonparametric Regression and Node Regression in LPMs

1.3.1 Risks

The (nonparametric) regression problem in its simplest form can be stated as estimating a *regression* function $f\colon Q \to \mathbb{R}$ based on a sample $\mathcal{D} \coloneqq (\mathbf{X}_n, \boldsymbol{y}) = \{(\boldsymbol{x}_1, y_1), \ldots, (\boldsymbol{x}_n, y_n) | \boldsymbol{x}_i \in Q, y_i \in \mathcal{Y} \subseteq \mathbb{R}\}$ where $\boldsymbol{x}_i$ are either deterministic points from a domain $Q \subseteq \mathbb{R}^d$ or are i.i.d. samples from a distribution with density p, supported on $Q \subseteq \mathbb{R}^d$ and

$$y_i = f(\boldsymbol{x}_i) + \epsilon_i \tag{13}$$

with ϵ_i i.i.d. centered, finite variance noise variables. The nonparametric literature uses the nomenclature of fixed and random design for the case of deterministic samples and random samples $\boldsymbol{x}_i$, respectively. An estimator $\hat{f} = \hat{f}_{\mathcal{D}}$ is any *random (measurable)* function $\hat{f}\colon Q \to \mathbb{R}$ that depends on the data $\mathcal{D}$. Traditionally, assuming observations of the form (13), in the case of fixed design the quality of the estimator $\hat{f}$ is measured by the *risk*

$$\mathcal{R}\big(\hat{f}(\boldsymbol{x}_{n+1}), f(\boldsymbol{x}_{n+1})\big) = \mathbb{E}_{\boldsymbol{\epsilon}}\left[\big(\hat{f}(\boldsymbol{x}_{n+1}) - f(\boldsymbol{x}_{n+1})\big)^2\right]$$

Under the random design assumption there are two notions of risks: *point-wise* and *global*. For a (non-random) point $\boldsymbol{x}_{n+1} \in Q$, the *point-wise risk* is given by

$$\mathcal{R}\big(\hat{f}(\boldsymbol{x}_{n+1}), f(\boldsymbol{x}_{n+1})\big) = \mathbb{E}_{\mathbf{X}_n, \boldsymbol{\epsilon}}\left[\big(\hat{f}(\boldsymbol{x}_{n+1}) - f(\boldsymbol{x}_{n+1})\big)^2\right] \tag{14}$$

which captures statistical information about the particular point $\boldsymbol{x}$ of the domain Q. The *global risk* is given by

$$\mathcal{R}\big(\hat{f}, f\big) = \mathbb{E}_{\mathbf{X}_{n+1}, \boldsymbol{\epsilon}}\left[\big(\hat{f}(\boldsymbol{x}_{n+1}) - f(\boldsymbol{x}_{n+1})\big)^2\right] \tag{15}$$

where $\boldsymbol{x}_{n+1}$ is an out of sample example not used in the training process. Note that the global and the point-wise risk are related by the equation

$$\mathcal{R}\big(\hat{f}, f\big) = \int \mathcal{R}\big(\hat{f}(\boldsymbol{x}), f(\boldsymbol{x})\big) p(\boldsymbol{x}) d\boldsymbol{x} \tag{16}$$

The global risk (15) can be interpreted as the average of the point-wise risk (14) over the data distribution p.

In contrast, when considering a node regression estimator, we will consider the risk taken with respect to the randomness of the edges, the additive noise on the label, and the latent positions (when they are treated as random variables). In other words if $\hat{f}$ is a node regression estimator (i.e. it depends on $\mathcal{D}_g = (\mathbf{A}, y)$, the adjacency matrix $\mathbf{A}$ and the graph label $\boldsymbol{y}$), we define the pointwise risk

$$\mathcal{R}_g(\hat{f}(\boldsymbol{x}_{n+1}), f(\boldsymbol{x}_{n+1})) = \mathbb{E}_{\boldsymbol{\mathcal{U}}, \boldsymbol{\epsilon}}\left[\Big(\hat{f}(\boldsymbol{x}_{n+1}) - f(\boldsymbol{x}_{n+1})\Big)^2\right] \tag{17}$$

and, in the case of random latent positions $\mathbf{X}_{n+1}$, the global risk, as

$$\mathcal{R}_g(\hat{f}, f) = \mathbb{E}_{\mathbf{X}_{n+1}, \boldsymbol{\mathcal{U}}, \boldsymbol{\epsilon}}\left[\Big(\hat{f}(\boldsymbol{x}_{n+1}) - f(\boldsymbol{x}_{n+1})\Big)^2\right] \tag{18}$$

In Equation (18) the expectation is taken as before over the latent positions $\mathbf{X}_{n+1}$ (which include the latent position of the regression node), the additive label noise $\boldsymbol{\epsilon}$, but also *the random matrix* $\boldsymbol{\mathcal{U}}$ which, we recall, is used along with $\mathbf{X}_{n+1}$ to generate the random adjacency matrix $\mathbf{A}$ through (8). It is often convenient to write the expectation this way instead of the conditional expectation $\mathbb{E}_{\mathbf{X}_{n+1}}\mathbb{E}_{\mathbf{A}|\mathbf{X}_{n+1}}$, since $\mathbf{X}_{n+1}$ and $\boldsymbol{\mathcal{U}}$ are independent. Sometimes we will adopt the shortcut $\boldsymbol{x} = \boldsymbol{x}_{n+1}$ in the notation above, with the understanding that random edges "link" the point $\boldsymbol{x}$ with all the others using the last column of $\boldsymbol{\mathcal{U}}$ as before: $a(\boldsymbol{x}_i, \boldsymbol{x}) = \mathbb{I}[U_{i,n+1} \leq k(\boldsymbol{x}_i, \boldsymbol{x})]$. Again, the risk (18) is the pointwise risk (17) integrated with respect to $\boldsymbol{x}_{n+1}$.

1.3.2 Estimators

Nadaraya-Watson A classical approach for the regression problem is the weighted average *Nadaraya-Watson* estimator, for which a modern theoretical analysis may be found in (Tsybakov, 2008; Györfi et al., 2002)

$$\hat{f}_{\mathrm{NW},\tau}(\boldsymbol{x}) = \begin{cases} \frac{\sum_{i=1}^n y_i \phi\left(\frac{\|\boldsymbol{x}-\boldsymbol{x}_i\|}{\tau}\right)}{\sum_{i=1}^n \phi\left(\frac{\|\boldsymbol{x}-\boldsymbol{x}_i\|}{\tau}\right)} & \text{if } \sum_{i=1}^n \phi\left(\frac{\|\boldsymbol{x}-\boldsymbol{x}_i\|}{\tau}\right) \neq 0 \\ 0 & \text{otherwise} \end{cases} \tag{19}$$

Here, $\phi\colon \mathbb{R} \to \mathbb{R}$ is called a *kernel* function. Some popular choices for ϕ are the *rectangular* kernel ($\phi(z) = \mathbb{I}\left[|z| \leq 1\right]$), *Gaussian* kernel ($\phi(z) = e^{-z^2}$), the *sinc* kernel ($\phi(z) = \frac{\sin(\pi z)}{\pi z}$). Other common choices are discussed in (Tsybakov, 2008). The parameter $\tau > 0$ is called the *bandwidth* and it controls the scale on which the data is being averaged. This parameter needs to be chosen carefully, as too small values of τ produce estimates of high variance (overfitting), while too large values of τ give highly biased estimators (underfitting), an instance of the *Bias-Variance trade-off*, a well known phenomenon in statistics and classical machine learning. The Nadaraya-Watson estimator is a *local* estimator, in that a prediction for a point $\boldsymbol{x}$ will depend on the distances of the samples $\boldsymbol{x}_i \in \mathcal{D}$ from the point of interest $\boldsymbol{x}$; NW is averaging the observations $\{y_i | \boldsymbol{x}_i \in \mathcal{D}\}$, giving higher weights to observations y_i with covariates $\boldsymbol{x}_i$ close to the point $\boldsymbol{x}$. Therefore, the NW is a reasonable estimator for regression functions that vary smoothly across the domain Q. More precisely, a natural class for the regression function f is the *Hölder class* $\Sigma(a, L)$ (Tsybakov, 2008) given by

$$\Sigma(a, L) = \left\{ g\colon \mathbb{R}^d \to \mathbb{R} |\ \textit{for all } \boldsymbol{x}, \boldsymbol{z} \in \mathbb{R}^d,\ |g(\boldsymbol{x}) - g(\boldsymbol{z})| \leq L\|\boldsymbol{x} - \boldsymbol{z}\|^a \right\}, \tag{20}$$

for $a \in [0, 1], L > 0$. The larger the parameter a, the smoother the function is. Indeed, for $a = 1$, one recovers the class of *Lipschitz* functions.

Minimax rates of NW The standard nonparametric rate in terms of the bandwidth is

$$\mathcal{R}\left(\hat{f}_{\mathrm{NW},\tau}, f\right) \leq C_1 \tau^{2a} + \frac{C_2}{n\tau^d} \tag{21}$$

where $C_1, C_2 > 0$ depend on the variance σ^2 of the noise $\boldsymbol{\epsilon}$, and the Hölder constant L, but not on the sample size n. In the large n regime, optimizing this rate in terms of τ, one gets that

$$\inf_{\tau > 0} \mathcal{R}\left(\hat{f}_{\mathrm{NW},\tau}, f\right) \leq C_\star n^{-\frac{2a}{2a+d}} \tag{22}$$

obtained for bandwidth $\tau_\star$ of order $n^{-\frac{1}{2a+d}}$ (Tsybakov, 2008; Györfi et al., 2002). It can also be shown that the rate (22) is optimal in a minimax sense for the Hölder class $\Sigma(a, L)$ (Tsybakov, 2008), i.e. given the prior that the regression function f belongs in the Hölder class $\Sigma(a, L)$, asymptotic improvements are only possible on the multiplicative constant $C_\star$, but not on the rate (22). In presence of additional smoothness of the regression function f, one can improve upon the rate (22). For example, for an appropriate class of m-times differentiable functions on $\mathbb{R}^d$, the rate of convergence is $n^{-\frac{2m}{2m+d}}$, see (Stone, 1982).

Graphical NW In this paper, we do *not* observe the positions $\boldsymbol{x}_i$, but we observe instead a random graph with $n+1$ nodes sampled according to a LPM with kernel function (3). We assume that for all but the last node there is a label of the form (13). Denoting $\boldsymbol{x} = \boldsymbol{x}_{n+1}$ for convenience, we introduce the (random) *empirical* degree

$$\hat{d}(\boldsymbol{x}) = \sum_{i=1}^{n} a(\boldsymbol{x}, \boldsymbol{x}_i) \tag{23}$$

where we recall that $a(\boldsymbol{x}, \boldsymbol{x}_i)$ are the random edges between the nodes $n+1$ and i taken as (8). With this notation, the GNW estimator (1) is given by

$$\hat{f}_{\mathrm{GNW}}(\boldsymbol{x}) = \begin{cases} \frac{1}{\hat{d}(\boldsymbol{x})} \sum_{i=1}^{n} y_i a(\boldsymbol{x}, \boldsymbol{x}_i) & \text{if } \hat{d}(\boldsymbol{x}) > 0 \\ 0 & \text{otherwise} \end{cases} \tag{24}$$

Note that the only edges of interest for the Graphical Nadaraya-Watson estimator are those adjacent to node $(n+1)$, so in the discussion of GNW we will not be concerned with the remaining edge variables $a(\boldsymbol{x}_i, \boldsymbol{x}_j)$ for $1 \leq i, j \leq n$. Since the edges $a(\boldsymbol{x}, \boldsymbol{x}_i)$ are Bernoulli variables with expectation $\alpha K\left(\frac{\|\boldsymbol{x}-\boldsymbol{x}_i\|}{h_g}\right)$, GNW can be considered as a quite noisy version of NW, where the true weights $\alpha K\left(\frac{\|\boldsymbol{x}-\boldsymbol{x}_i\|}{h_g}\right)$ are replaced by 1 with probability $\alpha K\left(\frac{\|\boldsymbol{x}-\boldsymbol{x}_i\|}{h_g}\right)$ and by 0 with complementary probability. Given the potentially high variance introduced by such nonlinear perturbations, it is somewhat surprising that GNW achieves the NW-rate (21) for $\tau \coloneqq h_g$, as we will show in Section 2.

Estimated NW As mentioned in the introduction, in order to address some shortcomings of GNW, we also study a broad family of node regression estimators that are built using a plug-in estimator of the latent distances. Specifically, we use an estimator of either latent distances or latent positions, then plug those estimates in a classical NW estimator. In addition to the observed label $\boldsymbol{y}$ on the first n nodes (13), namely $\boldsymbol{y} = [y_1, \ldots, y_n]^t$ and the adjacency matrix $\mathbf{A}$, we also assume that there exist an algorithm $\mathcal{A}$ that takes in the observed graph with adjacency matrix $\mathbf{A}$ as an input[5] and outputs a vector $\tilde{\boldsymbol{\delta}} = \left[\tilde{\delta}_1, \ldots, \tilde{\delta}_n\right]$, an *estimation of the distances* $\boldsymbol{\delta} = [\delta_1, \ldots, \delta_n]$ where δ_i is given by (12). The $\mathcal{A}$-Estimated Nadaraya-Watson is given by

$$\hat{f}^{\mathcal{A}}_{\mathrm{ENW},\tau}(\boldsymbol{x}_{n+1}) = \begin{cases} \dfrac{\sum_{i=1}^{n} y_i \phi\left(\frac{\tilde{\delta}_i}{\tau}\right)}{\sum_{i=1}^{n} \phi\left(\frac{\tilde{\delta}_i}{\tau}\right)} & \text{if} \quad \displaystyle\sum_{i=1}^{n} \phi\left(\frac{\tilde{\delta}_i}{\tau}\right) > 0 \\ 0 & \text{otherwise} \end{cases} \tag{25}$$

5. potentially depending on some hyperparameters.

where $\phi\colon [0,\infty) \to \mathbb{R}$ and $\tau > 0$ are *user chosen*. The theoretical analysis of $\mathcal{A}$-ENW is conducted in Sec. 3.

1.4 Related work

To the best of our knowledge, the node regression problem has not been studied in our framework. In particular, the metrics proposed in equations (17) and (18) are novel. However, there is a significant literature on convergence results to related problems in a slightly different context from ours. One can distinguish between problems in which the graph is *constructed* from available data and those where the graph is being implicitly imposed by the problem (our framework falls within the latter paradigm).

The graph is constructed According to (Song et al., 2021), a large part of the Graph Semi Supervised Learning literature falls in this category. The underlying assumption is classical: we have access to predictors $\boldsymbol{x}_i \in \mathbb{R}^d$, and (at least) some labels or signal values y_i. We may seek to denoise the observed signal (in smoothing problems), or to predict missing labels from observed labels (in semi-supervised learning). The graph enters only as a construction for performing these tasks.

A family of approaches construct a graph from the predictors in order to enforce some type of regularity on f (an early reference is Mallet (1989), even earlier references most likely exist). In Machine Learning this was popularised as Laplacian smoothing (Smola and Kondor (2003)). Recent works that study rates of convergence for this class of estimators include (Green et al., 2021; Shi et al., 2023) and (Rosa and Rousseau, 2024) in a Bayesian setting, among others.

We remark that these results are close in spirit (but not identical) to ours — the major difference being that they are concerned with classical signal smoothing or nonparametric regression — they operate on graphs by the *choice* of the practitioner. As such, the practitioner has full control over the construction of the graph $\mathcal{G}_n$, in particular, they are free to choose the bandwidth of the kernel which governs the implicit neighborhood size. In our setting the graph is *imposed* on the practitioner, as per equation (3). The fact that the bandwidth is imposed rather than user chosen will have deep implications for the theoretical analysis, as we shall see in Sections 2 and 3.

The graph is imposed by the problem There are several works that consider signal smoothing and node regression problems in this scenario. In (Belkin et al., 2004), the authors consider the graph SSL setting, and provide guarantees of convergence. However, the generalization bounds are provided in terms of a distribution on $[n] \times \mathbb{R}$ of the node-signal pairs and the resulting bounds depend on the spectrum of the graph in question, which is unsatisfactory from a statistical point of view. The authors in (Kovac and Smith, 2011) propose a penalized least squares method, where the penalization is in terms of ℓ_1 norm over the edges of the graph, albeit the focus is on optimization algorithms rather than rates of convergence. In (Wang et al., 2016), the authors propose a method of trend filtering on graphs and guarantees for the in-sample MSE are given. (Li et al., 2018) address the node regression problem, where in addition to the graph, node attributes are observed. A notable difference to our work (besides the availability of node attributes) is that the graph

is not considered to be random; the sole element of randomness is the additive noise on the signal.

When the graph is imposed by the problem, it is arguably more natural to consider the observed graph as a sample from a random graph model. Accordingly, a large amount of literature considers random graph models for node-level tasks. For example, graph clustering (community detection) has been studied in the context of Stochastic Block Models, for instance, in (Snijders and Nowicki, 1997; Abbe, 2018; Gao et al., 2022) and classification has been considered in (Tang et al., 2013), where the authors draw connections with kernel methods and Reproducing Kernel Hilbert Spaces (RKHS). Recently, convergence of Graph Neural Networks on large random graphs has been studied in (Cordonnier et al., 2024). As large graphs in the real world tend to be sparse (Albert and Barabási, 2002), a significant effort in the community detection literature is dedicated to understanding statistical properties of graphs with low expected degree (Oliveira, 2009; Lei and Rinaldo, 2015; Le et al., 2017). Another line of work specific to the LPMs studies algorithms for recovering latent positions or latent distances (Arias-Castro et al., 2018; Chen et al., 2020; Giraud et al., 2023; Dani et al., 2022; Shalizi and Asta, 2024). We will leverage some results established in this literature to demonstrate that $\mathcal{A}$-ENW achieves the standard nonparametric rate (22) in certain under-averaging ($h_g \ll \tau_\star$) and over-averaging ($h_g \gg \tau_\star$) regimes.

1.5 Outline

In Section 2 we show that under classical assumptions on the regression function f and the kernel K, the *Graphical Nadaraya-Watson* (GNW) estimator achieves the same risk rates as those of the *Nadaraya-Watson* estimator. A precise formulation of this statement can be found in Theorems 16 and 17. We follow an approach inspired by the classical bias-variance decomposition, but we use instead two quantities which we call *bias and variance proxies*, which are close but not equal to the exact bias and variance. The bias and variance proxies have simpler expressions and are easier to study. Under minimal assumptions on the additive noise, we show that the *variance proxy* of GNW is inversely proportional to the expected degree; a precise statement is in Sec. 2.1. In Sec. 2.2 we study the bias proxy. To do so, we require more assumptions on the kernel K as well as the distribution of latent positions p. Finally, in Sec. 2.3, we conclude the GNW analysis by combining the bias and variance analysis.

In Section 3 we study the two-stage estimator that consists in *estimating* the latent distances by some user-chosen algorithm $\mathcal{A}$ and then plugging those estimated distances into the classical NW (often with a bandwidth parameter τ_{CV} that is chosen by cross-validation by the user). Our analysis treats these two steps separately. In Sec 3.1 we show that Nadaraya-Watson with bandwidth τ and maximum perturbation of the distances $\Delta \geq 0$ preserves the classical rate (21) as long as $\Delta \lesssim \tau$ (see Theorem 22) and, building on that result, we prove a bound on the risk of $\mathcal{A}$-ENW in terms of the *probability of success* of the Algorithm $\mathcal{A}$ (see Theorem 23). In Sec. 3.2 we give several examples of existing literature on distance estimation algorithms $\mathcal{A}$ in LPMs to derive risk bounds on $\mathcal{A}$-ENW. We point out certain *under-averaging* and *over-averaging* regimes in which distance recovery can yield optimal nonparametric rates for $\mathcal{A}$-ENW.

In Section 4 we corroborate our theoretical results by numerical experiments. We consider two simple position recovery algorithms that achieve (sometimes only empirically) optimality in the under-averaging and over-averaging regimes respectively.

2. The Graphical Nadaraya-Watson (GNW) estimator

In this section we adopt the random design setting, i.e. we assume that the latent positions $\mathbf{X}_{n+1}$ are i.i.d. samples with density p supported on $Q \subseteq \mathbb{R}^d$. We will work conditionally on node $n+1$ having latent position $\boldsymbol{x}_{n+1} = \boldsymbol{x}$. The goal of this section is to provide a bound on the *global risk of GNW* (18). The approach we take is thus to provide an upper bound of (17) and then to integrate it in order to obtain a bound on the global risk (18). There will often be a need to take expectations with respect to the random matrix $\boldsymbol{\mathcal{U}}$ that generates the random edges (9), the latent positions $\mathbf{X}_n$ (4) and the additive noise $\boldsymbol{\epsilon}$ (7). In lieu of writing $\mathbb{E}_{\mathbf{X}_n,\boldsymbol{\mathcal{U}},\boldsymbol{\epsilon}}[\cdot]$, we will simply use the notation $\mathbb{E}[\cdot]$.

Recall that the labels are given by $y_i = f(\boldsymbol{x}_i) + \epsilon_i$. We make the following two general assumptions on the regression problem.

Assumption 1 *There exists $B > 0$ such that*

$$\|f\|_\infty \coloneqq \sup_{\boldsymbol{z} \in Q} |f(\boldsymbol{z})| \leq B < \infty$$

Assumption 2 *The additive noise $\boldsymbol{\epsilon}$ is such that its entries are independent random variables and*

$$\mathbb{E}[\epsilon_i] = 0 \text{ and } \max_{i \in [n]} \mathbb{E}\left[\epsilon_i^2\right] \leq \sigma^2 < \infty$$

Assumption 1 is somewhat restrictive but it holds in various settings. For example, if the domain Q is compact and there is a continuity assumption on f, then Assumption 1 is satisfied. The classical setup in (Györfi et al., 2002) includes this assumption. Assumption 2 is the most general assumption under model (13): while it is classical to assume stronger tail control of the distribution of the noise, here we just assume that it has finite variance.

We will follow a bias-variance decomposition inspired approach. For $\boldsymbol{x} \in Q$ we introduce a *variance proxy* and a *bias proxy* at $\boldsymbol{x}$:

$$v(\boldsymbol{x}) = \mathbb{E}\left[\left(\hat{f}_{\mathrm{GNW}}(\boldsymbol{x}) - S(f, \boldsymbol{x})\right)^2\right] \tag{26}$$

$$b(\boldsymbol{x}) = S(f, \boldsymbol{x}) - f(\boldsymbol{x}) \tag{27}$$

where $S(f, \boldsymbol{x})$ is the operator given by (11). We remark that these variance and bias proxies do *not* correspond to the exact variance and bias, but are simpler to manipulate. In fact, for the true bias and variance, we have the following result.

Proposition 3 *Let* $\mathrm{Bias}\left[\hat{f}_{\mathrm{GNW}}(\boldsymbol{x})\right]$ *and* $\mathrm{Var}\left[\hat{f}_{\mathrm{GNW}}(\boldsymbol{x})\right]$ *denote the standard bias and variance of $\hat{f}_{\mathrm{GNW}}(\boldsymbol{x})$, i.e.*

$$\mathrm{Bias}\left[\hat{f}_{\mathrm{GNW}}(\boldsymbol{x})\right] = \mathbb{E}\left[\hat{f}_{\mathrm{GNW}}(\boldsymbol{x})\right] - f(\boldsymbol{x}) \text{ and}$$

$$\mathrm{Var}\left[\hat{f}_{\mathrm{GNW}}(\boldsymbol{x})\right] = \mathbb{E}\left[\left(\hat{f}_{\mathrm{GNW}}(\boldsymbol{x}) - \mathbb{E}\left[\hat{f}_{\mathrm{GNW}}(\boldsymbol{x})\right]\right)^2\right]$$

If Assumptions 1 and 2 hold, then

$$0 \leq \left[b(\boldsymbol{x}) - \mathrm{Bias}(\hat{f}_{\mathrm{GNW}}(\boldsymbol{x}))\right]^2 \leq B^2 \exp(-2d(\boldsymbol{x}))$$

and

$$0 \leq v(\boldsymbol{x}) - \mathrm{Var}\left[\hat{f}_{\mathrm{GNW}}(\boldsymbol{x})\right] \leq B^2 \exp(-2d(\boldsymbol{x}))$$

The rest of this Section is dedicated to bounding the variance and bias proxies, in order to obtain a bound on the global risk. The variance proxy (26) governs the statistical fluctuation of $\hat{f}_{\mathrm{GNW}}(\boldsymbol{x})$ around the quantity $S(f, \boldsymbol{x})$. Its analysis is relying principally on probability techniques such as concentration inequalities. We provide a bound of this term in Sec. 2.1. The main result of the *sharp variance bound* given in Theorem 4, which states that $v(\boldsymbol{x})$ behaves like $1/d(\boldsymbol{x})$.

The bias proxy (27) on the other hand, measures the proximity of quantity $S(f, \boldsymbol{x})$ towards the regression function $f(\boldsymbol{x})$. Unlike the variance proxy (26) which can be universally controlled by concentration inequalities, the bias proxy (27) must be controlled on a case-by-case basis. To this end, we will focus on radial kernels (3). In this scenario, for compactly supported link functions, $S(f, \boldsymbol{x})$ approximates $f(\boldsymbol{x})$ uniformly within precision $\mathcal{O}(h_g^a)$, where $0 < a \leq 1$ is the Hölder exponent of the regression function f and h_g is the length-scale of the random graph kernel. This dependence is described in Sec. 2.2.

2.1 A Variance Bound

The goal of this subsection is to bound the variance proxy $v(\boldsymbol{x})$ (26) in terms of the expected degree $d(\boldsymbol{x})$ (10). Theorem 4 shows that $v(\boldsymbol{x})$ is of order $\mathcal{O}\left(\frac{1}{d(\boldsymbol{x})}\right)$. Later, in Section 2.2 we will show how the local degree $d(\boldsymbol{x})$ (10) depends on the parameters h_g and α in the kernel function (3).

Theorem 4 ***(Sharp Variance Bound)***
Suppose that Assumptions 1 and 2 hold. Then

$$v(\boldsymbol{x}) \leq \frac{9B^2 + 2\sigma^2}{d(\boldsymbol{x})}$$

Before presenting a proof of Theorem 4, let us briefly discuss its implication for convergence of GNW in various graph sparsity regimes. Formally this notion is defined in terms of an infinite sequence of graphs indexed by their node sizes n which tend to infinity. As mentioned in Section 1.1, we may consider various graph sparsity regimes in terms of the number of edges relative to the node size n. Recall that when the total number of edges is in $\mathcal{O}(n)$ the graph is considered to be sparse, whereas when this number is in $\Omega(n^2)$, the graph is considered dense. In-between these two regimes, the graph is *relatively sparse*. Since GNW is only concerned with 1-hop neighborhood of the regression node, it makes sense to introduce the notion of *local* sparsity regime of a graph sequence. For node $(n+1)$ with latent position $\boldsymbol{x}_{n+1} = \boldsymbol{x}$, the graph (sequence) is said to be locally sparse (resp. locally dense) at node $(n+1)$ if $d(\boldsymbol{x}) \in \mathcal{O}(1)$ (resp. $d(\boldsymbol{x}) \in \Omega(n)$). The graph is said to be locally relatively sparse at node $(n+1)$ if it is neither locally sparse nor locally dense at node $(n+1)$.

Theorem 4 shows that $\hat{f}_{\mathrm{GNW}}$ concentrates towards $S(f, \boldsymbol{x})$ in the *locally relatively sparse* and the *locally dense* regime. Indeed, for $n \to \infty$, the variance proxy $v(\boldsymbol{x}) \to 0$ as soon as the local degree (10) $d(\boldsymbol{x}) \in \omega(1)$. In comparison, most methods in the literature require a certain growth rate of the local degree (10) $d(\boldsymbol{x})$ in order to provide a theoretical guarantee: for instance, a classical threshold is logarithmic degrees, $d(\boldsymbol{x}) \in \Omega(\log(n))$ (Oliveira, 2009; Lei and Rinaldo, 2015).

Proof of Theorem 4 For convenience of notation, let

$$Z \coloneqq \mathbb{I}\left[\hat{d}(\boldsymbol{x}) > 0\right] \tag{28}$$

Note that by Definition (24) and Equation (28)

$$(1 - Z)\hat{f}_{\mathrm{GNW}}(\boldsymbol{x}) = 0 \tag{29}$$

or equivalently

$$Z\hat{f}_{\mathrm{GNW}}(\boldsymbol{x}) = \hat{f}_{\mathrm{GNW}}(\boldsymbol{x}) \tag{30}$$

Keeping in mind that Z is $\{0, 1\}$-valued variable signifying the occurrence of an edge incident to node $(n + 1)$, we have

$$\mathbb{E}\big[1 - Z\big] = \mathbb{P}\big(a(\boldsymbol{x}, \boldsymbol{x}_1) = a(\boldsymbol{x}, \boldsymbol{x}_2) = \cdots = a(\boldsymbol{x}, \boldsymbol{x}_n) = 0\big) = (1 - c(\boldsymbol{x}))^n \tag{31}$$

Additionally, we have $Z^2 = Z$ and $(1 - Z)^2 = 1 - Z$, so using Equations (29) and (30), we get

$$\begin{aligned}
v(\boldsymbol{x}) &= \mathbb{E}\left[\Big(\hat{f}_{\mathrm{GNW}}(\boldsymbol{x}) - S(f, \boldsymbol{x})\Big)^2 Z\right] + \mathbb{E}\left[\Big(\hat{f}_{\mathrm{GNW}}(\boldsymbol{x}) - S(f, \boldsymbol{x})\Big)^2 (1 - Z)\right] \\
&= \mathbb{E}\left[\Big(\hat{f}_{\mathrm{GNW}}(\boldsymbol{x})Z - S(f, \boldsymbol{x})Z\Big)^2\right] + \mathbb{E}\left[\Big(\hat{f}_{\mathrm{GNW}}(\boldsymbol{x})(1 - Z) - S(f, \boldsymbol{x})(1 - Z)\Big)^2\right] \\
&= \mathbb{E}\Big[\Big(\hat{f}_{\mathrm{GNW}}(\boldsymbol{x}) - S(f, \boldsymbol{x})Z\Big)^2\Big] + S^2(f, \boldsymbol{x})\mathbb{E}\left[1 - Z\right] \\
&= \mathbb{E}\Big[\Big(\hat{f}_{\mathrm{GNW}}(\boldsymbol{x}) - S(f, \boldsymbol{x})Z\Big)^2\Big] + S^2(f, \boldsymbol{x})\big(1 - c(\boldsymbol{x})\big)^n
\end{aligned} \tag{32}$$

where we used Equation (31) in line (32). In particular, we get

$$v(\boldsymbol{x}) \leq \mathbb{E}\Big[\Big(\hat{f}_{\mathrm{GNW}}(\boldsymbol{x}) - S(f, \boldsymbol{x})Z\Big)^2\Big] + e^{-d(\boldsymbol{x})}S^2(f, \boldsymbol{x}) \tag{33}$$

From Equation (33) it follows that we only need to focus on control of

$$\mathbb{E}\Big[\Big(\hat{f}_{\mathrm{GNW}}(\boldsymbol{x}) - S(f, \boldsymbol{x})Z\Big)^2\Big] \tag{34}$$

We will show that the term (34) is of order $\mathcal{O}(\frac{1}{d(\boldsymbol{x})})$, and hence, in Equation (33) we may substitute $e^{-d(\boldsymbol{x})}$ with $\frac{1}{d(\boldsymbol{x})}$. The key insight is that the expression within the expectation of Equation (34) has a representation as a sum of identically distributed, uncorrelated variables whose variance is easy to compute. We now derive this representation, using a method that we title the **decoupling trick**.

The decoupling trick For $I \subseteq [n]$, let

$$R_I(\boldsymbol{x}) = \begin{cases} \frac{1}{|I|+\sum_{j\notin I} a(\boldsymbol{x},\boldsymbol{x}_j)}, \ I \neq \emptyset \\ \frac{1}{\hat{d}(\boldsymbol{x})}, \ I = \emptyset \text{ and } \hat{d}(\boldsymbol{x}) > 0 \\ 0, \text{ otherwise} \end{cases}$$

For $I \neq \emptyset$, consider the graph $\mathcal{G}_I$ obtained by adding the edges $\{(n+1, i) | i \in I\}$ to the original LPM graph $\mathcal{G}$ (of course, not all edges of the form $(n+1, i), i \in I$ need to exist in the original graph). Then $1/R_I(\boldsymbol{x}) = |I| + \sum_{j\notin I} a(\boldsymbol{x}, \boldsymbol{x}_j)$ can be thought of as counting the number of neighbors of node $n+1$ in the modified graph $\mathcal{G}_I$. For $I = \emptyset$, observe that when $\hat{d}(\boldsymbol{x}) > 0$, $R_\emptyset(\boldsymbol{x}) = 1/\hat{d}(\boldsymbol{x})$ whereas when $\hat{d}(\boldsymbol{x}) = 0$, $R_\emptyset(\boldsymbol{x}) = 0$, and hence one can easily see that

$$\hat{f}_{\mathrm{GNW}}(\boldsymbol{x}) = \sum_{i=1}^{n} y_i a(\boldsymbol{x}, \boldsymbol{x}_i) R_\emptyset(\boldsymbol{x}) \tag{35}$$

At this point we placed the inconvenience of having a bracket in the definition (24) into the variable $R_\emptyset(\boldsymbol{x})$. For convenience of notation we define

$$R_i(\boldsymbol{x}) \coloneqq R_{\{i\}}(\boldsymbol{x}) \tag{36}$$

Taking into account the fact that $a(\boldsymbol{x}, \boldsymbol{x}_i)$ is a Bernoulli variable, i.e. it takes values in $\{0, 1\}$, it follows that for all $i \in [n]$

$$R_\emptyset(\boldsymbol{x}) a(\boldsymbol{x}, \boldsymbol{x}_i) = R_i(\boldsymbol{x}) a(\boldsymbol{x}, \boldsymbol{x}_i) \tag{37}$$

Indeed, if $a(\boldsymbol{x}, \boldsymbol{x}_i) = 0$ then both sides of Equation (37) are 0. Otherwise, $a(\boldsymbol{x}, \boldsymbol{x}_i) = 1$ and both sides in Equation (37) equal $R_i(\boldsymbol{x})$. Moreover, $R_i(\boldsymbol{x})$ is independent of $a(\boldsymbol{x}, \boldsymbol{x}_i)$. More generally we have the following observation.

Lemma 5 ***(Decoupling trick)*** *For all pairs of* ***disjoint*** *subsets $I, J \subseteq [n]$ we have*

$$R_J(\boldsymbol{x}) \prod_{i\in I} a(\boldsymbol{x}, \boldsymbol{x}_i) = R_{I\cup J}(\boldsymbol{x}) \prod_{i\in I} a(\boldsymbol{x}, \boldsymbol{x}_i)$$

and $R_{I\cup J}(\boldsymbol{x})$ is independent of $\{a(\boldsymbol{x}, \boldsymbol{x}_i) | i \in I\}$.

Proof If $\prod_{i\in I} a(\boldsymbol{x}, \boldsymbol{x}_i) = 0$ then there is nothing to prove. If $\prod_{i\in I} a(\boldsymbol{x}, \boldsymbol{x}_i) \neq 0$, then by the fact that $a(\boldsymbol{x}, \boldsymbol{x}_i)$ are Bernoulli variables we get $a(\boldsymbol{x}, \boldsymbol{x}_i) = 1$ for all $i \in I$. As $I \subseteq [n] \setminus J$, we have

$$R_J(\boldsymbol{x}) = \frac{1}{|J| + \sum_{i\notin J} a(\boldsymbol{x}, \boldsymbol{x}_i)} = \frac{1}{|I| + |J| + \sum_{i\notin I\cup J} a(\boldsymbol{x}, \boldsymbol{x}_i)} = R_{I\cup J}(\boldsymbol{x})$$

The second part of the lemma follows from modeling assumptions. ■

Plugging in Equation (37) into Equation (35), we get

$$\hat{f}_{\mathrm{GNW}}(\boldsymbol{x}) = \sum_{i=1}^{n} y_i a(\boldsymbol{x}, \boldsymbol{x}_i) R_i(\boldsymbol{x}) \tag{38}$$

Moreover, summing Equation (37) over $i \in [n]$ gives

$$\begin{aligned}\sum_{i=1}^{n} a(\boldsymbol{x}, \boldsymbol{x}_i) R_i(\boldsymbol{x}) &= \sum_{i=1}^{n} a(\boldsymbol{x}, \boldsymbol{x}_i) R_\emptyset(x) \\ &= \hat{d}(\boldsymbol{x}) R_\emptyset(\boldsymbol{x}) \\ &= \mathbb{I}\left[\hat{d}_n(\boldsymbol{x}) > 0\right] = Z\end{aligned}$$

we get

$$Z = \sum_{i=1}^{n} a(\boldsymbol{x}, \boldsymbol{x}_i) R_i(\boldsymbol{x}) \tag{39}$$

Using Equations (38) and (39) we have

$$\begin{aligned}\hat{f}_{\mathrm{GNW}}(\boldsymbol{x}) - S(f, \boldsymbol{x}) Z &= \sum_{i=1}^{n} y_i a(\boldsymbol{x}, \boldsymbol{x}_i) R_i(\boldsymbol{x}) - S(f, \boldsymbol{x}) \sum_{i=1}^{n} a(\boldsymbol{x}, \boldsymbol{x}_i) R_i(\boldsymbol{x}) \\ &= \sum_{i=1}^{n} (y_i - S(f, \boldsymbol{x})) a(\boldsymbol{x}, \boldsymbol{x}_i) R_i(\boldsymbol{x})\end{aligned} \tag{40}$$

In Appendix A we show that the summands in the right-hand side of Equation (40) are uncorrelated and consequently we obtain the following expression for the quantity (34).

Lemma 6 *We have*

$$\mathbb{E}\left[\left(\hat{f}_{\mathrm{GNW}}(\boldsymbol{x}) - S(f, \boldsymbol{x}) Z\right)^2\right] = \sum_{i=1}^{n} \mathbb{E}\left[\left(y_i - S(f, \boldsymbol{x})\right)^2 a(\boldsymbol{x}, \boldsymbol{x}_i) R_i^2(\boldsymbol{x})\right]$$

The proof of Lemma 6 may be found in the Appendix; it uses the decoupling trick 5 along with the fact[6] that $\mathbb{E}\left[\left(y_i - S(f, \boldsymbol{x})\right) a(\boldsymbol{x}, \boldsymbol{x}_i)\right] = 0$. Since

$$|S(f, \boldsymbol{x})| \leq \|f\|_\infty \tag{41}$$

one can deduce from Assumption 1 that $|S(f, \boldsymbol{x})| \leq B$. For $i \in [n]$, we have

$$\begin{aligned}\mathbb{E}_{\boldsymbol{\epsilon}}\left[\left(y_i - S(f, \boldsymbol{x})\right)^2\right] &= \left[\left(f(\boldsymbol{x}_i) - S(f, \boldsymbol{x})\right)^2\right] + \mathbb{E}_{\boldsymbol{\epsilon}}\left[\epsilon_i^2\right] \\ &\leq 4B^2 + \sigma^2\end{aligned} \tag{42}$$

Plugging in the bound (42) into Lemma 6, we get

6. This fact is the reason why we work directly with the random design; verbatim analysis for the fixed design does not satisfy this.

$$\begin{aligned}\mathbb{E}\left[\left(\hat{f}_{\mathrm{GNW}}(\boldsymbol{x}) - S(f,\boldsymbol{x})Z\right)^2\right] &\leq (4B^2+\sigma^2)\mathbb{E}\left[\sum_{i=1}^{n} a(\boldsymbol{x},\boldsymbol{x}_i)R_i^2(\boldsymbol{x})\right] \\ &= (4B^2+\sigma^2)\mathbb{E}\left[\sum_{i=1}^{n} a(\boldsymbol{x},\boldsymbol{x}_i)R_{\emptyset}^2(\boldsymbol{x})\right] \\ &= (4B^2+\sigma^2)\mathbb{E}\left[\frac{1}{\hat{d}(\boldsymbol{x})}\mathbb{I}\left[\hat{d}(\boldsymbol{x})>0\right]\right]\end{aligned} \tag{43}$$

The empirical degree (23) $\hat{d}(\boldsymbol{x})$ is a binomial random variable. Using a specific result for such variables, namely Lemma 4.1 in (Györfi et al., 2002), we have the following result[7]

$$\mathbb{E}\left[\frac{1}{\hat{d}(\boldsymbol{x})}\mathbb{I}\left[\hat{d}(\boldsymbol{x})>0\right]\right] \leq \frac{2}{d(\boldsymbol{x})}$$

Finally,

$$\mathbb{E}\left[\left(\hat{f}_{\mathrm{GNW}}(\boldsymbol{x}) - S(f,\boldsymbol{x})Z\right)^2\right] \leq \frac{8B^2+2\sigma^2}{d(\boldsymbol{x})} \tag{44}$$

Plugging the bounds (41) and (44) into Equation (33), we get the desired result. ■

Theorem 4 is essentially tight, at least in the presence of additive noise, i.e. we have the following lemma.

Lemma 7 *Suppose that* $\min_{i\in[n]}\mathbb{E}[\epsilon_i^2] \geq \sigma_0^2 > 0$. *Then*

$$v(\boldsymbol{x}) \geq \frac{\sigma_0^2\left(1-e^{-d(\boldsymbol{x})}\right)^2}{d(\boldsymbol{x})}$$

The proof of Lemma 7 can be found in the appendix.

We remark that the Theorem 4 holds for Latent Position Models with general nonparametric kernel functions $k\colon Q\times Q\to[0,1]$, as long as the condition $d(\boldsymbol{x})>0$ holds. Indeed, the proof of Theorem 4 is independent of the form of k. However, $S(f,\boldsymbol{x})$ (11) depends on f and on the kernel function k (2). When $k(\boldsymbol{x}_i,\boldsymbol{x}_j)$ depends on the distance $\|\boldsymbol{x}_i-\boldsymbol{x}_j\|$ as in (3), $S(f,\boldsymbol{x})$ is a good approximant of $f(\boldsymbol{x})$, as we show in Sec. 2.2.

7. Binomial random variables are known for their strong concentration around their mean, which in the case of $\hat{d}(\boldsymbol{x})$ is $d(\boldsymbol{x})$. Therefore, a similar result with potentially smaller constant upper bound for the expression $\mathbb{E}\left[\frac{d(\boldsymbol{x})}{\hat{d}(\boldsymbol{x})}\mathbb{I}\left[\hat{d}(\boldsymbol{x})>0\right]\right]$ is possible by means of concentration inequalities. We have instead chosen to use Lemma 4.1 in (Györfi et al., 2002), which simplifies the presentation. In addition, marginal improvements of multiplicative constants, although interesting, do not change the conclusions of our results.

2.2 Bias and Risk of GNW

In Sec. 2.1 we considered a LPM graph with general kernel function k. As mentioned before in (3), we will suppose that the kernel is *radial*, i.e.

$$k(\boldsymbol{x}, \boldsymbol{z}) = \alpha K\left(\frac{\|\boldsymbol{x} - \boldsymbol{z}\|}{h_g}\right) \tag{45}$$

with $0 < \alpha \leq 1$ and $h_g > 0$. These two parameters are considered to be **unknown and fixed**. There are two important questions that need to be addressed. First, under which conditions on α and h_g is $S(f, \boldsymbol{x})$ a good approximation of $f(\boldsymbol{x})$? In other words, how does the bias proxy (27) depend on α and h_g? Second, our bound for the variance proxy (26) is in terms of the local degree $d(\boldsymbol{x})$. Therefore, it is important to understand how the local degree $d(\boldsymbol{x})$ depends on the parameters α and h_g. We address these questions in this section. Proofs for this Section can be found in the Appendix B.

In order to control the bias proxy (27) we will need to assume regularity conditions on the regression function f, the kernel function K and on the density p.

Assumption 8 ***(Box assumption)***
There exists $M_1, M_2 > 0$ s.t. for all $t \in [0, \infty)$

$$\frac{1}{2}\mathbb{I}\left[t \leq M_1\right] \leq K(t) \leq \mathbb{I}\left[t \leq M_2\right]$$

Assumption 9 ***(Regularity of the regression function)***
There exist $0 < a \leq 1$ and $L > 0$ such that for all $\boldsymbol{x}, \boldsymbol{z} \in Q$

$$|f(\boldsymbol{x}) - f(\boldsymbol{z})| \leq L\|\boldsymbol{x} - \boldsymbol{z}\|^a$$

Assumption 10 ***(Regularity of the domain)***
There exist $r_0, c_0 > 0$ such that for all $\boldsymbol{x} \in Q = \operatorname{supp}(p)$, and all $r \leq r_0$,

$$m\left(Q \cap B_r\left(\boldsymbol{x}\right)\right) \geq c_0 m\left(B_r\left(\boldsymbol{x}\right)\right)$$

Here m is the Lebesgue measure on $\mathbb{R}^d$.

Finally, Assumptions 11 and 12 cover different type of distributions for the latent positions.

Assumption 11 ***(Density Assumption 1)***
There exists $p_0 > 0$ such that for all $\boldsymbol{x} \in Q$

$$p(\boldsymbol{x}) \geq p_0$$

Assumption 12 ***(Density Assumption 2)***
There exist $0 < b \leq 1$ and $S > 0$ such that $p \in \Sigma(b, S)$ and

$$\int p^{1/2}(\boldsymbol{x}) dx < \infty$$

Assumptions 8 and 9 are rather classical in the context of the NW estimator. The NW estimator performs poorly in low density regions and near the boundary of the support of the data distribution. The intuitive explanation for this behavior is that because there are on average fewer observations in such a region, the variance of the estimator is greater. Under Assumptions 10, 11 and 12, the problematic regions are not too large. Assumption 10 is the most technical one, but it is satisfied in many instances considered in the classical regression setting. Clearly, $\mathbb{R}^d$ satisfies Assumption 10 with $r_0 = \infty$, $c_0 = 1$, and it is not difficult to show that the Cube $Q_d = [-1, 1]^d$ satisfies the regularity Assumption 10 with $r_0 = 1$, $c_0 = \frac{1}{2^d}$ and so does every closed and convex subset[8] of $\mathbb{R}^d$ (for some $r_0, c_0 > 0$). Another broad class of sets which satisfy this property and are used in the regression context in $\mathbb{R}^d$ are those that satisfy *interior cone condition* (Wendland, 2004). A set Q satisfies an interior cone condition with cone C if for all points $\boldsymbol{x} \in Q$, one can rotate and translate C to a cone $C_{\boldsymbol{x}}$ with a vertex in $\boldsymbol{x}$ such that $C_{\boldsymbol{x}} \subseteq Q$. A typical example of Assumption 11 is the uniform distribution (over a convex body), whereas Assumption 12 covers non-compactly supported, but smooth distributions such as the Gaussian.

Under Assumptions 8 and 9, the bias proxy (27) is uniformly bounded over Q by Lemma 13.

Lemma 13 ***(Bias control lemma)*** *Suppose that Assumptions 8 and 9 hold. Then*

$$\sup_{\boldsymbol{x} \in Q} |S(f, \boldsymbol{x}) - f(\boldsymbol{x})| \leq 2LM_2^a h_g^a$$

The problematic vertices for GNW are those whose latent positions fall in a low density region or are near the boundary of the support Q.

Our next lemma lower-bounds the expected degree of a node, to control the risk:

Lemma 14 ***(Local degree bound)***
Suppose that Assumption 8 and 10 hold. If $M_1 h_g < r_0$ and $\boldsymbol{x} \in Q$ is such that

$$p_0(\boldsymbol{x}) \coloneqq \inf_{\substack{\boldsymbol{z} \in Q \\ \|\boldsymbol{x} - \boldsymbol{z}\| \leq M_1 h_g}} p(\boldsymbol{z}) > 0 \tag{46}$$

Then

$$\frac{1}{d(\boldsymbol{x})} \leq \frac{C}{n\alpha h_g^d p_0(\boldsymbol{x})}$$

where $C = \frac{2}{c_0 v_d M_1^d}$ and v_d is the volume of the $d-$dimensional unit ball.

This Lemma in combination with Theorem 4 and Lemma 13 gives a bound on the point-wise risk (17). Having established bounds on the bias (27) and variance (26) proxies, we are ready to provide a bound on the point-wise risk (17).

Theorem 15 ***(Pointwise risk bound)***
Suppose that Assumptions 8, 9, 10 hold. Furthermore, suppose that $\boldsymbol{x} \in Q$ is s.t. $p_0(\boldsymbol{x}) > 0$ where $p_0(\boldsymbol{x})$ is given by (46). If $M_1 h_g \leq r_0$ then

$$\mathcal{R}_g\left(\hat{f}_{\mathrm{GNW}}(\boldsymbol{x}), f(\boldsymbol{x})\right) \leq C_1 h_g^{2a} + \frac{C_2}{n\alpha h_g^d p_0(\boldsymbol{x})}$$

8. when compact, such sets are called convex bodies

where $C_1 = 4L^2M_2^{2a}$ *and* $C_2 = \frac{36B^2+8\sigma^2}{c_0 v_d M_1^d}$

In the next section we follow up by giving bounds on the global risk (18) in terms of the parameters α and h_g.

2.3 Global risk

Finally, to bound the global risk (18) of GNW, we would like to integrate the inequality given in Theorem 15. Unfortunately the right-hand side of this inequality depends on $p_0(\boldsymbol{x})$, a quantity that depends non trivially on the behavior of p around the point $\boldsymbol{x} \in Q$, so a direct integration does not work. However, Assumption 11 allows us to conclude that $p_0(\boldsymbol{x}) \geq p_0$ for all $\boldsymbol{x} \in Q$ and hence yields the following result.

Theorem 16 ***(Risk bound 1)***
Suppose that Assumptions 8, 9, 10 and 11 hold. If $M_1 h_g < r_0$*, we have*

$$\mathcal{R}_g\left(\hat{f}_{\mathrm{GNW}}, f\right) \leq C_1 h_g^{2a} + \frac{C_2}{n\alpha h_g^d}$$

where $C_1 = 4L^2M_2^{2a}$*,* $C_2 = \frac{36B^2+8\sigma^2}{p_0 c_0 v_d M_1^d}$*.*

Theorem 16 matches the classical rate (21) with $\tau \coloneqq h_g$. In this sense, GNW with length-scale h_g behaves like a classical NW estimator with **fixed bandwidth** $\tau \coloneqq h_g$. In particular, Assumptions 10 and 11 apply for latent positions with compactly supported distribution p, that is also lower bounded by a positive constant, in some sense a relaxation of the uniform distribution. Theorem 16 does not cover distributions supported on all of $\mathbb{R}^d$, or any set of infinite Lebesgue measure more generally. For example, the Gaussian distribution over $\mathbb{R}^d$ is not covered by Assumption 11. Such density functions must achieve arbitrary small values and hence it is not possible to control $p_0(\boldsymbol{x})$ globally in the same way as it was done with Assumption 11. However, under Assumption 12 we get the following result.

Theorem 17 ***(Risk bound 2)***
Suppose that Assumptions 8, 9, 10 and 12 hold. If $h_g < \min(r_0/M_1, 1)$ *then*

$$\mathcal{R}_g\left(\hat{f}_{\mathrm{GNW}}, f\right) \leq C_1 h_g^{\min\left(2a, b/2\right)} + \frac{C_2}{n\alpha h_g^{d+b}}$$

where $C_1 = 4L^2M_2^{2a} + (8B^2 + 2\sigma^2)S^{1/2}M_1^{b/2}\int p^{1/2}(\boldsymbol{x})dx$ *and* $C_2 = \frac{36B^2+8\sigma^2}{c_0 v_d S M_1^{d+b}}$*.*

Assumptions 10 and 12 extend the class of distributions of the latent points to Hölder continuous density with non-compact support, with the caveat that the support still needs to be geometrically regular. The cost of these assumptions is an increase both in the bias and the variance proxies. For example, when $b = 1$ and h_g is sufficiently small, $h_g^{2a} \lesssim h_g^{\min\left(2a, 1/2\right)}$ and equality holds only when $a \leq 1/4$, i.e. the regression function f is fairly hard to learn. On the other hand, $\frac{1}{n\alpha h_g^d} \lesssim \frac{1}{n\alpha h_g^{d+1}}$, and hence the variance term is always worse under Assumption 12.

The main idea behind the proof of Theorem 17 is to split the risk over a high density region i.e. where the density is $p(\boldsymbol{x}) \gtrsim h_g^b$, and its complement. Due to the integrability condition in Assumption 12, and the fact that the point-wise risk is bounded by a constant, the low density region can be handled. For the high-density region, the risk is controlled by Theorem 15.

Remark 18 ***(Twice differentiable functions)*** *In the presence of additional smoothness of the regression function, for example f being twice continuously differentiable, the classical Nadaraya-Watson estimator has bias of order τ^2, where τ is the NW bandwidth, both for pointwise (see (Wasserman, 2005), Theorem 5.65) and global risk (See (Wasserman, 2005), Theorem 5.44). The bias proxy term* (27) *seems to behave similarly to the classical bias of NW, and therefore we speculate that the bias proxy* (27) *is of order h_g^2, under appropriate assumptions on the kernel K and the density p of the latent positions. We suspect that tighter bounds for the pointwise risk* (17) *might be possible when the latent position $\boldsymbol{x}_{n+1}$ is at distance at least h_g from the boundary of the domain Q. We note that the bias of classical NW for points near the boundary is known to be of order τ (See (Wasserman, 2005), Theorem 5.65) and therefore it is not reasonable to expect GNW bias proxy bounds better than h_g for $\boldsymbol{x}_{n+1}$ near the boundary of the latent space Q. Consequently, in order to derive improved bounds on the global risk* (18)*, one should consider the behavior of*

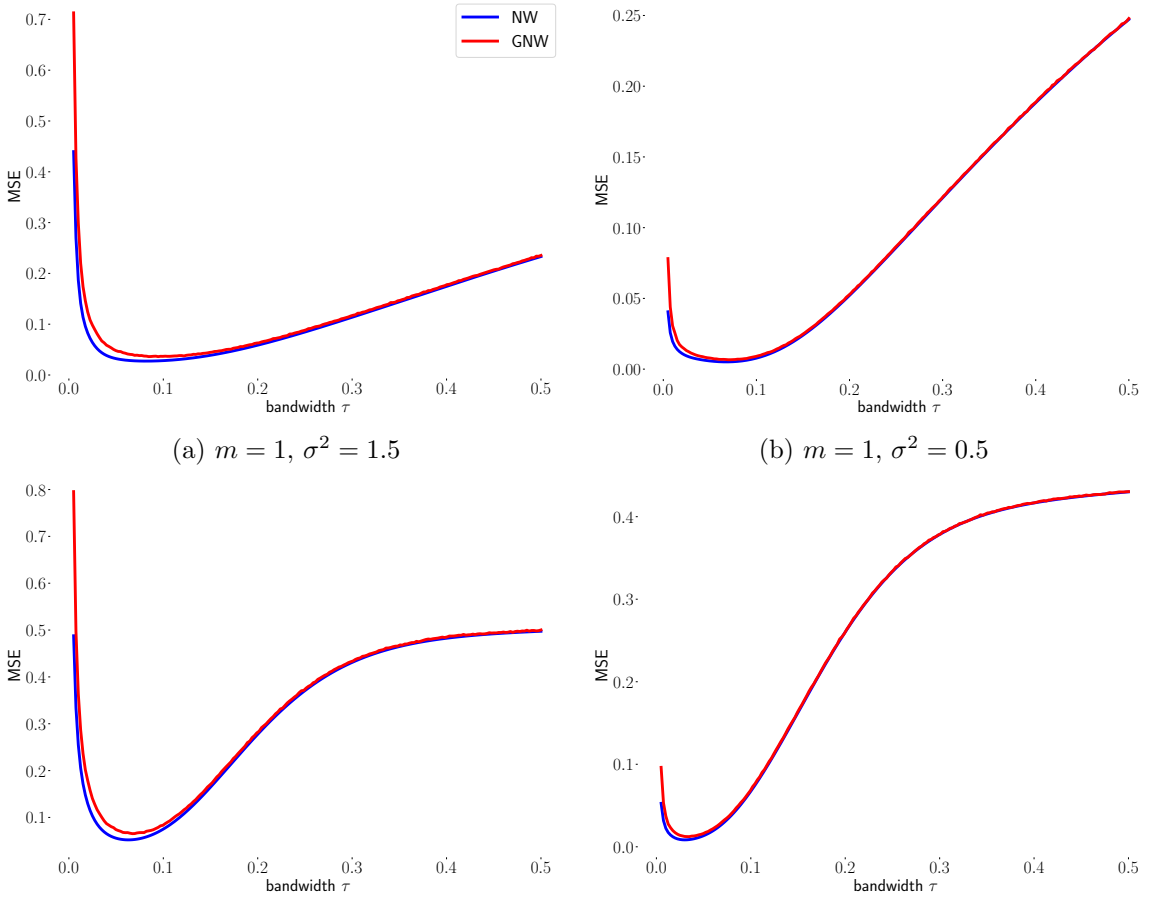
$$\int \big(S(f,\boldsymbol{x}) - f(\boldsymbol{x})\big)^2 p(\boldsymbol{x}) d\boldsymbol{x}$$

over the class of twice differentiable functions, rather than the supremum norm used in Lemma 13.

2.4 Discussion

The GNW estimator is computationally extremely cheap (with runtime $\mathcal{O}(n)$) and it has the same convergence rate[9] as the (fixed-bandwidth) NW estimator. In order to find the range of values h_g for which the GNW risk converges for large LPMs, we conduct a simplified asymptotic analysis of GNW. We suppose that the scaling factor $\alpha = 1$ in (3). In order to conclude that the risk $\mathcal{R}_g\left(\hat{f}_{\mathrm{GNW}}, f\right)$ converges to 0, we need both terms in the upper bound of Theorem 16, namely h_g^{2a} and $1/nh_g^d$ to go to 0. Note that this is equivalent to having the expected local degree (10) $d(\boldsymbol{x}) \to \infty$ and the local edge density (10) $c(\boldsymbol{x}) \to 0$ at the same time. Moreover, elementary calculus shows that the ideal bias-variance trade-off is achieved for $h_g = \tau_\star := c_{a,d} n^{-\frac{1}{d+2a}}$ and the associated rate for the risk is $C_{a,d} n^{-\frac{2a}{d+2a}}$, where $c_{a,d}$ and $C_{a,d}$ also depend on the various parameters that appear in the constants C_1 and C_2 in Theorem 16. Similar analysis may be conducted for Theorem 17. In summary, $\hat{f}_{\mathrm{GNW}}$ behaves reasonably well as soon as $h_g \to 0$ and $nh_g^d \to \infty$, with the risk depending on where this parameter h_g happens to fall. In Figure 2, we compare the in-sample MSE of the classical Nadaraya-Watson and the Graphical Nadaraya-Watson, in terms of the bandwidth parameter $\tau > 0$. As expected from our theoretical results, the curves are of very similar shape.

9. up to a multiplicative constant



(a) $m = 1$, $\sigma^2 = 1.5$ (b) $m = 1$, $\sigma^2 = 0.5$

(c) $m = 2$, $\sigma^2 = 1.5$ (d) $m = 2$, $\sigma^2 = 0.5$

Figure 2: "Bias-Variance" trade-off curves for NW, and GNW, based on the bandwidth parameter τ. For a range of values τ, we compute the in sample MSE of NW with parameter τ and compare it to the in-sample MSE of GNW with length scale h. In order to reduce variance of the GNW bias-variance curve, we use Monte Carlo techniques that average NUM_{MC} instances of edge-randomness in the LPMs (the latent positions and the signal are kept fixed over the simulation). Parameter setup: $n = 500$, $\text{NUM}_{\text{MC}} = 20$. The frequency m and the label noise σ^2 vary as specified in the caption of the sub-figures.

The problem with using the GNW estimator arises when the length-scale h_g is either too small or too large. When the length-scale h_g is too small (relative to $\tau_\star$), GNW averages labels over a neighborhood that is too small, meaning it will have low bias but high variance. We call this case the *under-averaging regime* ($h_g \ll \tau_\star$). On the other hand, a large length-scale h_g (relative to $\tau_\star$) will result in averaging on a window of size h_g, larger than the optimal window of size $\tau_\star$. This leads to low variance but high bias, and we call this case the *over-averaging regime* ($h_g \gg \tau_\star$).

We reemphasize the fact that length-scale h_g and the optimal bandwidth $\tau_\star$ are **not** user chosen parameters. The length-scale h_g is inherent to the generative process of the

graph: it influences the size of neighborhoods in the latent space and the sparsity of the graph. The optimal bandwidth $\tau_\star$ depends primarily on the sample size n, the smoothness of the regression function f, namely the Hölder constant and exponent L, a, as well as on the variance of the additive noise σ^2. The optimal bandwidth $\tau_\star$ determines the size of the window in the latent space which achieves optimal performance for the label $\boldsymbol{y}$ given by (13). In the remainder of this paper we will focus on the following question: For what pairs of values $(h_g, \tau_\star)$ can we construct a node regression estimator that achieves the optimal risk rate (21) with $\tau \coloneqq \tau_\star$?

Theorem 16 states that GNW achieves this for $h_g = \tau_\star$, and less formally, that the risk is nearly optimal when h_g is in the vicinity of $\tau_\star$. In the following section we consider the *Estimated Nadaraya-Watson* estimator, which, as we will see, achieves *standard minimax risk rates* in certain under-averaging and over-averaging regimes.

3. Nadaraya-Watson on estimated positions (ENW)

In this section we are going to consider an estimator that works on a broader range of length-scales, that still relies on a local averaging approach. In particular, the goal is to construct node regression estimators that will outperform the GNW estimator in the under and over-averaging regimes, when $h_g \ll \tau_\star$ and $h_g \gg \tau_\star$ respectively, with the ultimate goal of achieving optimal rate (21) for $\tau = \tau_\star$. We consider a LPM with kernel function (45) with $\alpha = 1$.

As mentioned in the introduction, since the main drawback of GNW is that the latent positions $\boldsymbol{x}_i$ are unknown and the bandwidth τ cannot be adjusted, we suggest combining two classical approaches from the literature: *first* estimate the latent positions — or, more precisely, estimate the *distances* from the regression node $(n+1)$ to all the others — *then* use the standard Nadaraya-Watson estimator with these estimated distances, allowing the user to freely tune the bandwidth as usual.

As there are many approaches for the first step in the literature, we suppose that the user chooses some *latent distance estimation algorithm* $\mathcal{A}$ that takes as an input the observed graph (potentially with some other hyperparameters) and returns an *estimate for the distance between the latent positions* of node $(n+1)$ and all other nodes i in the graph. The algorithm $\mathcal{A}$ needs to be deterministic, in the sense that the only random components on which it acts are the adjacency matrix $\mathbf{A}$ and the observed label $\boldsymbol{y}$, i.e. we do not cover random algorithms which require additional randomness in their execution such as additional random walks used in DeepWalk (Perozzi et al., 2014) or Node2Vec (Grover and Leskovec, 2016). By plugging these estimated distances in a NW estimator, we end up with a prediction for the regression node. We call the resulting estimator the **$\mathcal{A}$-Estimated Nadaraya-Watson** ($\mathcal{A}$-ENW).

The analysis of the performance of $\mathcal{A}$-ENW may be broken down into two problems: analyzing the precision of the distance estimation algorithm $\mathcal{A}$, and analyzing the *stability of NW* to using estimated distances instead of exact ones. The first error depends on the choice of $\mathcal{A}$ and has been studied extensively in the literature. We will give several examples in Section 3.2.

For the stability of NW, in Section 3.1 we provide a risk bound when the algorithm $\mathcal{A}$ estimates the distances $\boldsymbol{\delta}$ (12) with an additive error Δ. We show that in this case $\mathcal{A}$-ENW

achieves (up to a multiplicative constant) the classical NW rate (21) *as long as* $\tau \gtrsim \Delta$. This result is formally stated in Theorem 22. In particular, given a problem for which the optimal bandwidth is $\tau_\star$, $\mathcal{A}$-ENW can achieve the optimal NW-rate (21) with $\tau \coloneqq \tau_\star$ provided that $\tau_\star \gtrsim \Delta$.

In Section 3.2 we give several examples of algorithms $\mathcal{A}$ in the literature and their respective Δ. We find instances $(h_g, \tau_\star)$, both in the under-averaging and the over-averaging regime, for which there exist algorithms $\mathcal{A}$ such that $\mathcal{A}$-ENW achieves the optimal NW-rate (21) with $\tau \coloneqq \tau_\star$.

3.1 Risk bound on Estimated Nadaraya-Watson

In this section, in addition to the observed labels $\boldsymbol{y}$ on the first n nodes (13) $\boldsymbol{y} = [y_1, \dots, y_n]^t$ and the adjacency matrix $\mathbf{A}$, we also assume that there exists an algorithm $\mathcal{A}$ that takes the observed graph with adjacency matrix $\mathbf{A}$ as input and outputs a vector $\tilde{\boldsymbol{\delta}} = \left[\tilde{\delta}_1, \dots, \tilde{\delta}_n\right]$, an *estimation of the distances* $\boldsymbol{\delta} = [\delta_1, \dots, \delta_n]$ where δ_i is the distance between the $(n+1)$th and the ith latent variables (12). We remark that such an algorithm should be equivariant, i.e. if we relabel the nodes $[n]$ with some permutation $\pi \colon [n] \to [n]$, the algorithm $\mathcal{A}$ will permute its outputs by that same permutation. We suppose that the latent positions $\mathbf{X}_{n+1}$ are fixed; although our analysis can be easily extended to the random design case as well. The quality of the estimator $\mathcal{A}$ will be measured by

$$\Delta\left(\mathcal{A}, \mathbf{X}_{n+1}\right) \coloneqq \|\tilde{\boldsymbol{\delta}} - \boldsymbol{\delta}\|_\infty = \max_{i \in [n]} |\tilde{\delta}_i - \delta_i| \tag{47}$$

where δ_i is given by (12). Even though we state our results in terms of latent distance estimation, we can easily adapt them to *position estimation* algorithms, as described in the following remark.

Remark 19 ***(Position Estimation Algorithms)*** *If $\mathcal{B} \coloneqq \{0,1\}^{(n+1)\times(n+1)} \to \mathbb{R}^{d\times(n+1)}$ is a position estimation algorithm with $\mathcal{B}(\mathbf{A}) = [\tilde{\boldsymbol{x}}_1, \dots, \tilde{\boldsymbol{x}}_n, \tilde{\boldsymbol{x}}_{n+1}]$, where $\tilde{\boldsymbol{x}}_i$ is an estimate of the latent position $\boldsymbol{x}_i$, then one can consider the induced distance estimation algorithm $\mathcal{A}_\mathcal{B}$ given by*

$$\tilde{\delta}_i = \|\tilde{\boldsymbol{x}}_i - \tilde{\boldsymbol{x}}_{n+1}\|$$

The triangle inequality implies that

$$\Delta(\mathcal{A}_\mathcal{B}, \mathbf{X}_{n+1}) = \max_{i\in[n]} |\delta_i - \tilde{\delta}_i| \leq 2 \max_{i \in [n+1]} \|\tilde{\boldsymbol{x}}_i - \boldsymbol{x}_i\| \tag{48}$$

Hence in the case of position estimation algorithms, one can replace the metric $\Delta(\mathcal{A}_\mathcal{B}, \mathbf{X}_{n+1})$ by $\mathrm{D}(\mathcal{B}, \mathbf{X}_{n+1}) \coloneqq 2\max_{i\in[n+1]} \|\tilde{\boldsymbol{x}}_i - \boldsymbol{x}_i\|$. For position estimation algorithms $\mathcal{B}$, we use the slightly abusive notation and write $\mathcal{B}$-ENW instead of $\mathcal{A}_\mathcal{B}$-ENW.

We will analyze the performance of the Nadaraya-Watson estimator with estimated distances $\tilde{\boldsymbol{\delta}}$ in terms of the metric (47). We note that considering the maximal distance $\Delta(\mathcal{A}, \mathbf{X}_{n+1})$ (47) is somewhat pessimistic (in comparison to average distances for example), however, it results in a more susceptible theoretical analysis. In contrast to the Graphical Nadaraya-Watson estimator, the distance estimation approach allows for a choice of a kernel

function ϕ as well as a bandwidth τ. Throughout this section, we will make the following two assumptions.

Assumption 20 *The kernel function $\phi\colon [0,\infty) \to [0,1]$ is non-negative, compactly supported and non-vanishing in a neighborhood of 0, i.e. there are $M_1, M_2 > 0$ such that*

$$\frac{1}{2}\mathbb{I}\left[t \le M_1\right] \le \phi\big(t\big) \le \mathbb{I}\left[t \le M_2\right]$$

Assumption 21 *The positions $\mathbf{X}_{n+1}$ and $\tau > 0$ are such that the number of points in $\mathbf{X}_n = [\boldsymbol{x}_1, \dots, \boldsymbol{x}_n]$ in the $\frac{M_1\tau}{2}$ window around $\boldsymbol{x}_{n+1}$ satisfies*

$$M(\tau) \coloneqq \sum_{i=1}^{n} \mathbb{I}\left(\delta_i \le \frac{M_1\tau}{2}\right) \ge k_0 n \tau^d \tag{49}$$

for some $k_0 > 0$.

Assumption 20 is the same as Assumption 8, the major difference being that ϕ is user chosen whereas K is implicit in the definition of the LPM. In particular, one can *choose* the constants M_1, M_2 as well. For example, the function $\phi_0\colon [0,\infty) \to [0,1]$ given by $\phi_0(t) = \mathbb{I}(t \le 1)$ with $M_1 = M_2 = 1$ is a valid choice for ENW averaging. Assumption 21 is also standard in the NW literature. Loosely speaking, it guarantees that the points are sufficiently scattered across their support, relative to the bandwidth τ. For example, in a random sample of i.i.d. points with distribution p satisfying Assumptions 10 and 11, Assumption 21 fails to hold with probability $\mathcal{O}(e^{-n\tau^d})$.

Our aim is to prove a guarantee on the $\mathcal{A}$-Estimated Nadaraya-Watson estimator with *estimated* distances $\tilde{\boldsymbol{\delta}} = \mathcal{A}(\mathbf{A})$ given by

$$\hat{f}^{\mathcal{A}}_{\mathrm{ENW},\tau}(\boldsymbol{x}_{n+1}) = \begin{cases} \dfrac{\sum_{i=1}^{n} y_i \phi\left(\frac{\tilde{\delta}_i}{\tau}\right)}{\sum_{i=1}^{n} \phi\left(\frac{\tilde{\delta}_i}{\tau}\right)} & \text{if} \quad \displaystyle\sum_{i=1}^{n} \phi\left(\frac{\tilde{\delta}_i}{\tau}\right) > 0 \\ 0 & \text{otherwise} \end{cases} \tag{50}$$

Note that even though $\boldsymbol{x}_{n+1}$ appears in the notation of (50), the latent position $\boldsymbol{x}_{n+1}$ *is not* fed into $\mathcal{A}$-ENW. This is only a convention in order to stick close to the notation of the classical ML regression literature. If the algorithm $\mathcal{A}$ is sufficiently accurate in the estimation of the latent distances, we claim that the error in the subsequent NW procedure will preserve the classical rate (21). A precise statement of this claim is given in the following theorem.

Theorem 22 *Suppose that the kernel ϕ used in* (50) *satisfies Assumption 20. Additionally, suppose that $\mathbf{X}_{n+1}$ and $\tau > 0$ satisfy Assumption 21 and that the regression function f satisfies the regularity Assumption 9, i.e. it is Hölder regular with exponent $0 < a \le 1$. Finally, suppose that the position recovery algorithm $\mathcal{A}$ satisfies*

$$\Delta(\mathcal{A}, \mathbf{X}_{n+1}) \le \frac{M_1\tau}{2}$$

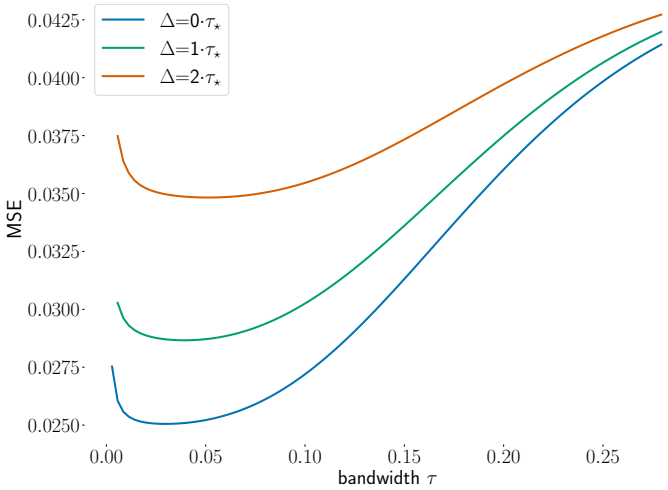


Figure 3: Bias Variance trade-off Curves for NW under perturbation. Sample size $n = 500$, label $y_i = \sin(4\pi \boldsymbol{x}_i) + \epsilon_i$ with $\epsilon_i \sim \mathcal{N}(0, 1.5)$ and $\boldsymbol{x}_i \sim \mathrm{Unif}[0, 1]$

Then

$$\mathbb{E}_{\boldsymbol{\epsilon}}\left(|\hat{f}^{\mathcal{A}}_{\mathrm{ENW},\tau}(\boldsymbol{x}_{n+1}) - f(\boldsymbol{x}_{n+1})|^2\right) \leq C_1\tau^{2a} + \frac{4\sigma^2}{k_0 n\tau^d}$$

where $C_1 = 2L^2\left[\left(\frac{M_1}{2} + M_2\right)^{2a}\right]$

Theorem 22 states that if the algorithm $\mathcal{A}$ has distance estimation error $\Delta(\mathcal{A}, \mathbf{X}_{n+1})$ (47) below $\frac{M_1\tau}{2}$, $\mathcal{A}$-ENW averaged over the additive noise (2) achieves the classical NW rate (21) (up to a multiplicative constant). However, the algorithm $\mathcal{A}$ acts on the random matrix $\mathbf{A}$ and hence even in the case when $\mathbf{X}_{n+1}$ are treated as fixed, $\Delta(\mathcal{A}, \mathbf{X}_{n+1})$ is a random variable that depends on the edge variables $\boldsymbol{\mathcal{U}}$. When the algorithm $\mathcal{A}$ fails to estimate distances within precision $\frac{M_1\tau}{2}$, we do not expect that the subsequent NW averaging procedure will yield interesting results.

We justify this claim by numerical evidence. Let us consider univariate positions $\boldsymbol{x}_i \in [0, 1]$, and perturbations given by $\boldsymbol{x}_{i,\Delta} = \boldsymbol{x}_i + \Delta \boldsymbol{u}_i$, where $\boldsymbol{u}_i \in [-1, 1]$ are uniform variables and $\Delta = k\tau_\star$, $k = 0, 1, 2$. We compute *Leave One Out prediction* values

$$\hat{f}_{\Delta,\tau}(\boldsymbol{x}_i) = \frac{\sum_{j=1, j\neq i}^n y_j \phi\left(\frac{|\boldsymbol{x}_{j,\Delta} - \boldsymbol{x}_{i,\Delta}|}{\tau}\right)}{\sum_{j=1, j\neq i}^n \phi\left(\frac{|\boldsymbol{x}_{j,\Delta} - \boldsymbol{x}_{i,\Delta}|}{\tau}\right)} \tag{51}$$

which can be interpreted as predictions for the value $f(\boldsymbol{x}_i)$ based on a Nadaraya-Watson estimator with design

$$\mathbf{X}_\Delta = \left[\boldsymbol{x}_{1,\Delta}, \ldots, \boldsymbol{x}_{i-1,\Delta}, \boldsymbol{x}_{i,\Delta}, \boldsymbol{x}_{i+1,\Delta}, \ldots \boldsymbol{x}_{n,\Delta}\right],$$

labels

$$\boldsymbol{y} = [y_1, \ldots, y_{i-1}, y_{i+1}, \ldots, y_n] \in \mathbb{R}^{n-1}$$

where $y_j = f(\boldsymbol{x}_j) + \epsilon_j$ and bandwidth τ. We then compute the Leave-One-Out (LOO) Mean Squared Error

$$\mathrm{MSE}\left(\hat{f}_{\Delta,\tau}, f\right) = \frac{1}{n}\sum_{i=1}^{n}\left(\hat{f}_{\Delta,\tau}(\boldsymbol{x}_i) - f(\boldsymbol{x}_i)\right)^2 \tag{52}$$

and we plot it as a function of τ (see Figure 3). We see that for $k = 2$, i.e. when the precision is $2\tau_\star$, the bias-variance trade-off curve does not exhibit the same behavior as for $k = 1$, which is closer to the bias-variance trade-off curve with exact positions ($k = 0$).

Therefore, we opt to control the probability of the failure of the algorithm $\mathcal{A}$ defined as

$$p_\tau(\mathcal{A}, \mathbf{X}_{n+1}) = \mathbb{P}_{\boldsymbol{\mathcal{U}}}\left(\Delta\left(\mathcal{A}\left(\mathbf{A}\right), \mathbf{X}_{n+1}\right) > \frac{M_1\tau}{2}\right) \tag{53}$$

The following result provides a bound on the pointwise risk (17), which we recall takes expectation both over additive label noise and the random edges in the graph.

Theorem 23 ***(ENW risk rate (deterministic design))*** *Suppose that the kernel ϕ used in* (50) *satisfies Assumption 20. Additionally, suppose that $\mathbf{X}_{n+1}$ and $\tau > 0$ satisfy Assumption 21 and the regression function f satisfies Assumption 9. Then*

$$\mathcal{R}_g\left(\hat{f}^{\mathcal{A}}_{\mathrm{ENW},\tau}(\boldsymbol{x}_{n+1}), f(\boldsymbol{x}_{n+1})\right) \leq C_0 p_\tau(\mathcal{A}, \mathbf{X}_{n+1}) + C_1\tau^{2a} + \frac{4\sigma^2}{k_0 n\tau^d}$$

where $C_0 = 4B^2 + 2\sigma^2$, $C_1 = 2L^2\left[\left(\frac{M_1}{2} + M_2\right)^{2a}\right]$

From Theorem 23 it follows that if $p_\tau(\mathcal{A}, \mathbf{X}_{n+1})$ is smaller than the classical NW-rate (21), $\mathcal{A}$-ENW will achieve, up to a multiplicative constant, the same rate (21) in τ. In particular, if this is true for the optimal bandwidth $\tau_\star$, then $\mathcal{A}$-Estimated Nadaraya-Watson *can achieve the optimal non-parametric NW rate* $n^{-\frac{2a}{2a+d}}$.

The LPM literature on position estimation (Arias-Castro et al., 2018; Dani et al., 2022; Giraud et al., 2023) typically establish rate of convergence for $\Delta(\mathcal{A}, \mathbf{X}_{n+1})$, i.e. there exist several results which provide rates $r_n > 0$, such that $p_{r_n}(\mathcal{A}, \mathbf{X}_{n+1})$ is overwhelmingly small, typically of order $\mathcal{O}(1/n)$ (much lower than the rate (21) for any $\tau > 0$). Since $p_\tau(\mathcal{A}, \mathbf{X}_{n+1})$ is decreasing function in τ, it follows that we can match the classical NW-rate (21) for any $\tau \gtrsim r_n$.

Several results (Arias-Castro et al., 2018), (Dani et al., 2022) indicate that the sparsity of the graph as dictated by the length-scale h_g plays a key role in establishing the rate r_n. To the best of our knowledge, theoretical understanding of the relationship between the length-scale h_g and the rate r_n in the general compactly supported kernel LPM setting is incomplete. As a consequence we cannot characterize completely the pairs of values $(h_g, \tau_\star)$ for which optimal rates (22) are achievable. In the next section we consider two approaches for the under-averaging and over-averaging regimes respectively, and point out several instances of LPMs for which we can get optimal NW performance by using $\mathcal{A}$-ENW.

3.2 Distance and Position estimation algorithms

In this section we have a glance at the existing literature on distance and position estimation algorithms and discuss some implications for the node regression problem. There are several estimators in the context of LPMs, but the theoretical analysis remains limited. We will go through a few examples, focusing on consequences for the node regression estimator $\mathcal{A}$-ENW. Given a LPM with kernel of the shape (3), the length-scale h_g will play an important role in the probability of failure (53). Recall that in the classical regression setting, for label $\boldsymbol{y} = f(\mathbf{X}_n) + \boldsymbol{\epsilon}$ with f satisfying Assumption 9, and $\boldsymbol{\epsilon}$ satisfying Assumption 2, the rate (21) achieves a minimal value $C_\star n^{-\frac{2a}{2a+d}}$ for $\tau_\star = c_\star n^{-\frac{1}{2a+d}}$, where $c_\star, C_\star > 0$ depend on L and σ^2.

3.2.1 The Shortest Path Algorithm

The Shortest Path Algorithm $\mathcal{A}_{sp}$ is the simplest and cheapest approach to distance estimation. The idea behind it is that the shortest path distance in the graph between nodes i and j should approximate (up to a scaling factor h_g) the distance of the latent positions $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$. This algorithm is the subject of study of (Arias-Castro et al., 2018). In this work it is shown that for *any* distance estimator $\hat{\boldsymbol{d}}$ based on the adjacency matrix $\mathbf{A}$, there will be some configuration of points $\mathbf{X}_{n+1}$ such that for a random geometric graph with length-scale h_g, at least half of the quantities $|\delta_i - \hat{\delta}_i|$ (12) are of order $\Omega(h_g)$. In particular, $\Delta(\mathcal{A}, \mathbf{X}_{n+1})$ is of order $\Omega(h_g)$ and hence, using our results (Theorem 23), this particular approach could yield optimal rates only in the under-averaging regime $h_g \ll \tau_\star$. To explore the implications for the downstream task of node regression, we will use the more general result Theorem 3 of (Arias-Castro et al., 2018). In our notation, they show if the latent positions $\mathbf{X}_{n+1}$ are uniformly spread out on a convex body Q, i.e.

$$\Lambda(\mathbf{X}_n) \coloneqq \sup_{\boldsymbol{x} \in Q} \min_{i \in [n]} \|\boldsymbol{x}_i - \boldsymbol{x}\| \leq \epsilon \tag{54}$$

and the kernel function K (3) is supported on $[0,1]$ and satisfies $K(t) \geq C_0(1-t)^A$ for some $C_0 > 0, A \geq 0$, then there exists $C_2 > 0$ (depending only on A and C_0) such that

$$\mathbb{P}_{\boldsymbol{\mathcal{U}}} \left(\Delta(\mathcal{A}_{sp}, \mathbf{X}_{n+1}) > C_2 \left(h_g + \left(\frac{\epsilon}{h_g} \right)^{\frac{1}{1+A}} \right) \right) \lesssim \frac{1}{n} \tag{55}$$

Building on their result, we get the following corollary. For the sake of simplicity we omit some constants in the analysis (in particular σ^2 and B), which amounts to asymptotic study with large n, where h_g is allowed to depend on n.

Corollary 24 *Suppose that $d = 1$, K is supported on $[0,1]$ with $K(t) \geq C_0(1-t)^A$ for some $C_0 > 0, 0 \leq A < 1$, and $\mathbf{X}_{n+1}$ are i.i.d. with density p that satisfies Assumptions 10 and 11. Furthermore, suppose that the regression function f satisfies Assumption 9 with $a > \frac{1+A}{2}$. Finally, suppose that $\mathbf{A}$ is an adjacency matrix of a LPM random graph with link function* (3) *s.t.*

$$\log(n) n^{-\frac{2a-A}{1+2a}} \lesssim h_g \lesssim n^{-\frac{1}{1+2a}}.$$

Then, with high probability over $\mathbf{X}_{n+1}$*,*

$$\inf_{\tau>0} \mathcal{R}_g\left(\hat{f}^{\mathcal{A}}_{\mathrm{ENW},\tau}(\boldsymbol{x}_{n+1}), f(\boldsymbol{x}_{n+1})\right) \lesssim n^{-\frac{2a}{2a+1}}$$

The time complexity of the shortest path algorithm is $\mathcal{O}\left(n\log(n)nh_g^d\right)$, in the growing degree regime $nh_g^d = \Omega(1)$ and $\mathcal{O}\left(n\log(n)\right)$ in the bounded degree regime $nh_g^d = \Theta(1)$.

Generalization to the Random Geometric Graphs in dimension $d \geq 2$ Note that for the special case of random geometric graph, $p_\tau(\mathcal{A}, \mathbf{X}_{n+1}) \in \{0,1\}$, since there is no edge randomness $\boldsymbol{\mathcal{U}}$. The approach of (Dani et al., 2022) consists in refining the shortest path distances by taking into account the number of common neighbors of the nodes. They propose a distance estimation algorithm, and building on that algorithm, they construct a position recovery algorithm $\mathcal{B}_{rgg}$. In our notation, setting $Q = [0,1]^d$ to be the d-dimensional cube with latent positions following a uniform distribution on Q, they obtain the following bound (with high probability over the sampled points $\mathbf{X}_n$)

$$\mathrm{D}\left(\mathcal{B}_{rgg}, \mathbf{X}_{n+1}\right) \leq C_d \begin{cases} (nh_g^d)^{-\frac{2}{d+1}} & \text{if} \quad 1 < nh_g^d < n^{\frac{d+1}{2d}} \\ \sqrt{\log(n)}n^{-\frac{1}{d}} & \text{if} \quad n^{\frac{d+1}{2d}} \leq nh_g^d < n \end{cases} \tag{56}$$

Note that as the degree of the LPM graph grows[10] (i.e. as the graph becomes denser), the error (56) decreases until $nh_g^d = n^{\frac{d+1}{2d}}$, after which the rate is $r_n = \sqrt{\log(n)}n^{-\frac{1}{d}}$. In particular it shows convergence of the algorithm even for sparse graphs. However, in order to obtain nonparametric rates using this algorithm, we need to be in a certain density regime. More precisely, based on the bound (56), we have the following result.

Corollary 25 *Suppose that* $d \geq 2$ *and that* $\mathbf{A}$ *is adjacency matrix of a Random Geometric Graph. If* $nh_g^d \gtrsim n^{\frac{d+1}{2(d+2a)}}$ *then with high probability over the samples* $\mathbf{X}_{n+1}$*, we have*

$$\inf_{\tau>0} \mathcal{R}_g\left(\hat{f}^{\mathcal{A}}_{\mathrm{ENW},\tau}(\boldsymbol{x}_{n+1}), f(\boldsymbol{x}_{n+1})\right) \lesssim n^{-\frac{2a}{2a+d}}$$

The algorithm $\mathcal{B}_{rgg}$ runs in $\mathcal{O}(n^\omega \log(n))$, where $\omega < 2.373$ is the matrix multiplication constant (Alman and Vassilevska Williams, 2021).

3.2.2 Localize and Refine: Optimal recovery on the Sphere

Another instance in which optimal recovery is possible is provided by (Giraud et al., 2023). This work concerns position recovery in the large length-scale regime $h_g \geq c_0 > 0$ on the sphere $\mathbb{S}^1 \subseteq \mathbb{R}^2$. Our model slightly differs from theirs, as we work with data supported on convex bodies in $\mathbb{R}^d$, which excludes surfaces like the sphere, although our results easily generalize to the case of smooth manifolds. They propose an algorithm titled *Localize and Refine* $\mathcal{B}_{LaR}$ for position recovery on the sphere with

$$\mathrm{D}(\mathcal{B}_{LaR}, \mathbf{X}_{n+1}) \leq C\sqrt{\frac{\log(n)}{n}} \tag{57}$$

10. In LPMs with compactly supported kernel functions, the degree scales like nh_g^d

Furthermore, they show that this rate is minimax optimal in their setting. Corollary 4.3 in their paper provides a specific link function K s.t.

$$p_\tau(\mathcal{B}_{LaR}, \mathbf{X}) \leq \frac{9}{n^2}$$

whenever $\tau \geq C\sqrt{\frac{\log(n)}{n}}$. In particular, when $\tau_\star \gtrsim \sqrt{\frac{\log(n)}{n}}$, $\mathcal{B}_{LaR}$-ENW achieves optimal non-parametric rates. This happens for $\frac{1}{1+2a} < \frac{1}{2}$, or equivalently for $a > 1/2$. Thus, in the over-averaging regime $\frac{h_g}{\tau_\star} \gg 1$, there are algorithms $\mathcal{B}$ such that $\mathcal{B}$-ENW achieves optimal nonparametric rates for sufficiently regular functions (Hölder exponent $a > 1/2$). This is somewhat surprising, as such graphs are extremely dense, with every degree having order $\geq c_0 n$. The time complexity of this algorithm is polynomial in the number of nodes n.

4. Numerical Experiments

We study empirically two *position recovery* algorithms[11] based on the ideas of (Arias-Castro et al., 2018) and (Giraud et al., 2023), that are intended to treat the under-averaging and over-averaging regime, respectively (see Figure 6). We restrict our attention to the cases $d = 1$ and $d = 2$ i.e, it is assumed that the latent positions are univariate, uniform i.i.d. variables on $Q = [0, 1]^d$. Throughout this section, the Kernel Function K (3) is taken to be the Gaussian $K(t) = e^{-t^2}$ and $\alpha = 1$.

The first algorithm $\mathcal{B}_{sp}$ is based on the shortest path algorithm, which yields an approximation for the distances (12). We convert these distances into position estimates by using classical Multi Dimensional Scaling (Torgerson, 1952), abbreviated as cMDS. Our implementation computes the graph distances using the Floyd-Warshwall algorithm FW, which computes all graph distances in time complexity $\mathcal{O}(n^3)$. For disconnected graphs, this algorithm will correctly calculate the graph distance of nodes i and j in different connected components to be infinite. However, since we perform cMDS on the graph distances, we require that all graph distances are finite. Indeed, one can apply cMDS on each connected component, providing embeddings for different components which are not comparable with one another. Instead, we opt to implement position recovery algorithms only for *connected graphs*. Algorithm $\mathcal{B}_{sp}$ is expected to outperform GNW in the under-averaging regime $\frac{h_g}{\tau_\star} = o(1)$. The shortest path $\mathcal{B}_{sp}$ algorithm is described in Algorithm 1

Input: Adjacency matrix $\mathbf{A}$
Compute FW($\mathbf{A}$);
Return cMDS(FW($\mathbf{A}$));
Output: Positions $\hat{\mathbf{X}} \in \mathbb{R}^{n\times 1}$

Algorithm 1: Shortest Path Position Recovery Algorithm $\mathcal{B}_{sp}$

The second algorithm $\mathcal{B}_{spectral}$ is more empirical in nature. Recall that in the over-averaging regime we have $h_g \gg \tau_\star$. The main idea is to "shrink the length scale", i.e. to produce a new adjacency matrix $\mathbf{A}_q$ that indicates if two points are within distance τ_q, where $\tau_q \ll h_g$. In order to achieve this goal, we will *denoise* the adjacency matrix $\mathbf{A}$ in the hope to get a more accurate estimate $\hat{\mathbf{K}}$ of $\mathbf{K}$. Keeping in mind that $\hat{\mathbf{K}}_{i,j}$ is an estimate of

11. Code available at `https://github.com/mgjorgje/lpm_simulation_code`.

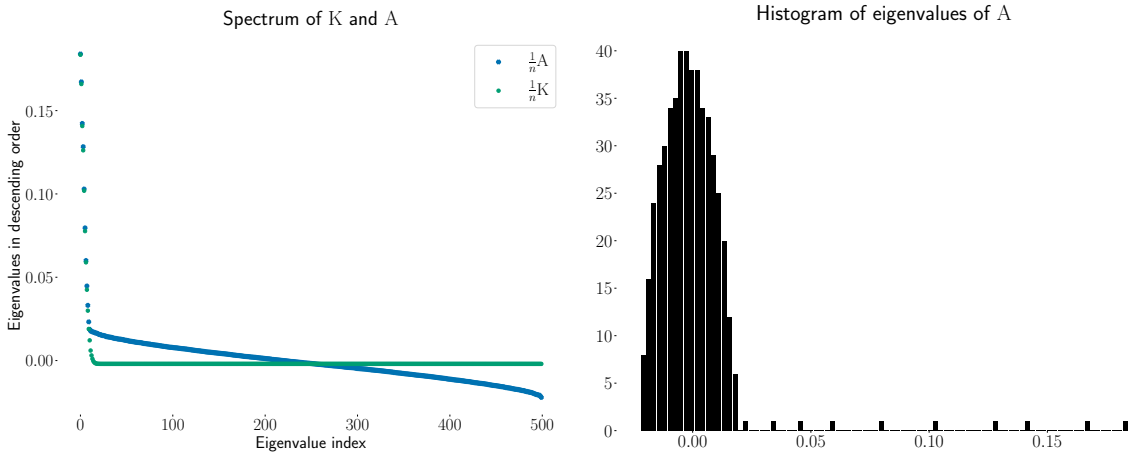
$$[\mathbf{K}]_{i,j} = K\left(\frac{\|\boldsymbol{x}_i - \boldsymbol{x}_j\|}{h_g}\right) \tag{58}$$

where K (3) is a decreasing function, we can construct $\mathbf{A}_q$ by

$$\left[\mathbf{A}_q\right]_{i,j} = \mathbb{I}\left[\hat{\mathbf{K}}_{i,j} > q\right] \tag{59}$$

Once $\mathbf{A}_q$ is constructed, we run the shortest-path algorithm $\mathcal{B}_{sp}$ (1) on $\mathbf{A}_q$.

We now explain the construction of the denoised matrix $\hat{\mathbf{K}}$. This construction is based on empirical observations, and theoretical analysis is out of the scope of this paper. Empirically, we observe that the eigenvalue distribution of the adjacency matrix $\mathbf{A}$ admits a Wigner-like semicircular law (Anderson et al., 2009). Indeed, $\mathbf{A} = \mathbf{K}+\mathbf{E}$ where $\mathbf{K}_{i,j}$ is given by (58) and $\mathbf{E}$ is a random matrix with centered and independent entries (conditionally on the latent positions). The spectrum of $\mathbf{A}$ is formed from a *bulk* of eigenvalues coming from $\mathbf{E}$ and only a few eigenvalues of $\mathbf{K}$ are separated from this bulk (See Figure 4). Moreover, the eigenvectors of the adjacency matrix $\mathbf{A}$ associated with these eigenvalues separated from the bulk tend to be a very good approximation for the corresponding eigenvectors of the kernel matrix $\mathbf{K}$, whereas as soon as an eigenvalue of $\mathbf{A}$ enters the bulk, its associated eigenvector is overwhelmed with noise (see Figure 5).



(a) (b)

Figure 4: Illustration for a LPM with sample size $n = 500$ and length-scale $h_g = 0.1$. Figure 4a shows a scatter plot of descending eigenvalues of $\mathbf{K}$ and $\mathbf{A}$. Interestingly, the first few eigenvalues of $\mathbf{A}$ are very close to the corresponding ordered eigenvalues of $\mathbf{K}$. Figure 4b shows a histogram of Eigenvalues of $\mathbf{A}$. The top several eigenvalues of $\mathbf{A}$ are well separated from the rest, which fall in the semicircular *bulk*.

Let $\mathbf{A} = \sum_{i=1}^n \sigma_i \boldsymbol{v}_i \boldsymbol{v}_i^t$ and $\mathbf{K} = \sum_{i=1}^n \lambda_i \boldsymbol{u}_i \boldsymbol{u}_i^t$ be the spectral decompositions of $\mathbf{A}$ and $\mathbf{K}$, respectively, where the sequences (λ_i) and (σ_i) are decreasing. For Positive Semi-Definite (PSD) matrices $\mathbf{K}$ our heuristic observation (See Figures 4 and 5) suggests that the low-rank matrix $\hat{\mathbf{K}} = \sum_{i=1}^r \sigma_i \boldsymbol{v}_i \boldsymbol{v}_i^t$ is a good approximation of $\mathbf{K}$, where r is the number of

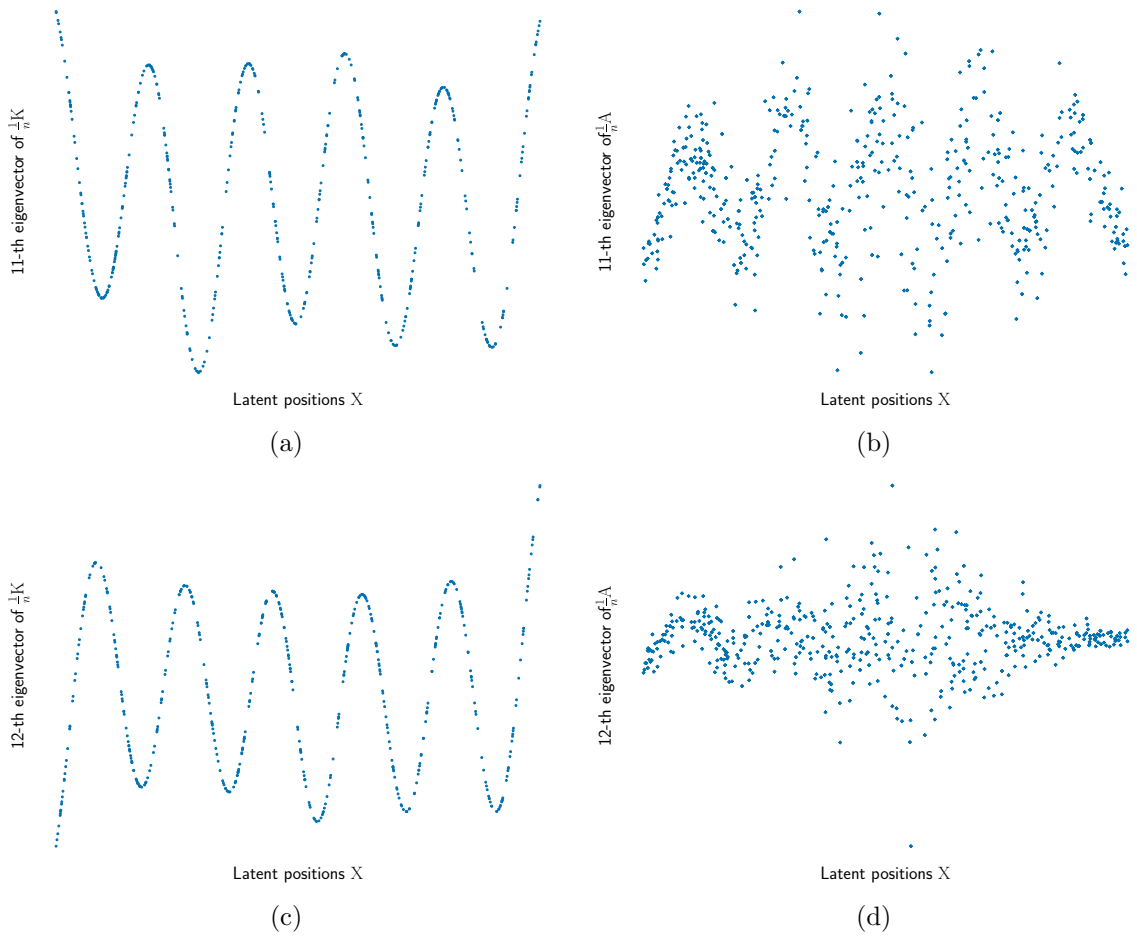


Figure 5: Scatter plots of $(\mathbf{X}_n, \boldsymbol{u}_j)$ and $(\mathbf{X}_n, \boldsymbol{v}_j)$. Figures 5a and 5b show a scatter plots of the 11th eigenvector of the matrices $\mathbf{K}$ and $\mathbf{A}$ respectively. This is the index of the last eigenvalue that separates from the bulk. Figures 5c and 5d demonstrate the same scatter plot, for the 12th eigenvector of the matrices $\mathbf{K}$ and $\mathbf{A}$, respectively. This is the index of the first eigenvalue that belongs in the bulk.

eigenvalues that are "out of the bulk". While there are many methods to estimate the bulk, here we resort to a simple *symmetrization trick*: due to the symmetry of the bulk and the fact that $\mathbf{K}$ is PSD, the *most negative* eigenvalue should be a good indication of the size of the bulk. More precisely, we keep the eigenvalues σ_i and their corresponding eigenvectors whenever $\sigma_i > -\sigma_n$, i.e. r is the index such that $\sigma_r > -\sigma_n \geq \sigma_{r+1}$, where σ_n is the smallest (most negative) eigenvalue of $\mathbf{A}$. If $\mathbf{K}$ was not PSD, one could for instance consider the spacings of the eigenvalues in order to select the threshold for the spectrum. The algorithm $\mathcal{B}_{spectral}$ is described in Algorithm 2.

Input: Adjacency matrix $\mathbf{A}$, threshold parameter q, eigenvalue tolerance ρ_0
Compute eigenvalue threshold: $r = \#\{1 \leq i \leq n : \sigma_i > -(1+\rho_0)\sigma_n\}$;
Compute low rank matrix $\hat{\mathbf{K}} = \sum_{i=1}^{r} \sigma_i \boldsymbol{v}_i \boldsymbol{v}_i^t$;
Construct new adjacency matrix $\mathbf{A}_q$ with $[\mathbf{A}_q]_{i,j} = \mathbb{I}\left[\hat{\mathbf{K}}_{i,j} \geq q\right]$;
Run $\mathcal{B}_{sp}$ on $\mathbf{A}_q$;
Output: Estimated Positions $\hat{\mathbf{X}} \in \mathbb{R}^{n \times 1}$

Algorithm 2: Spectral Position Recovery Algorithm $\mathcal{B}_{spectral}$

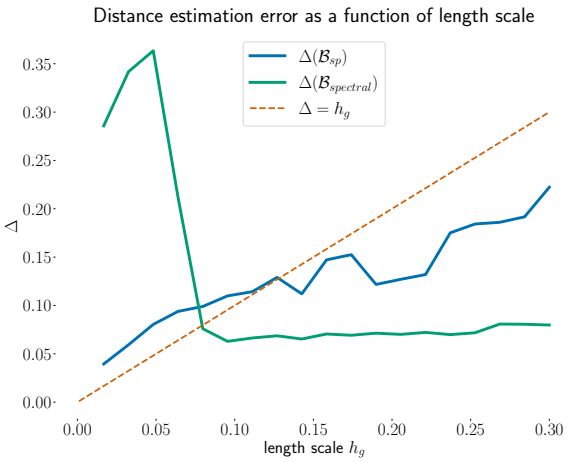


Figure 6: Empirical error of $\mathcal{B}_{sp}$ and $\mathcal{B}_{spectral}$ as a function of the length-scale h_g of the LPM. Configuration settings: $n = 500$, $\mathrm{NUM}_{\mathrm{MC}} = 20$

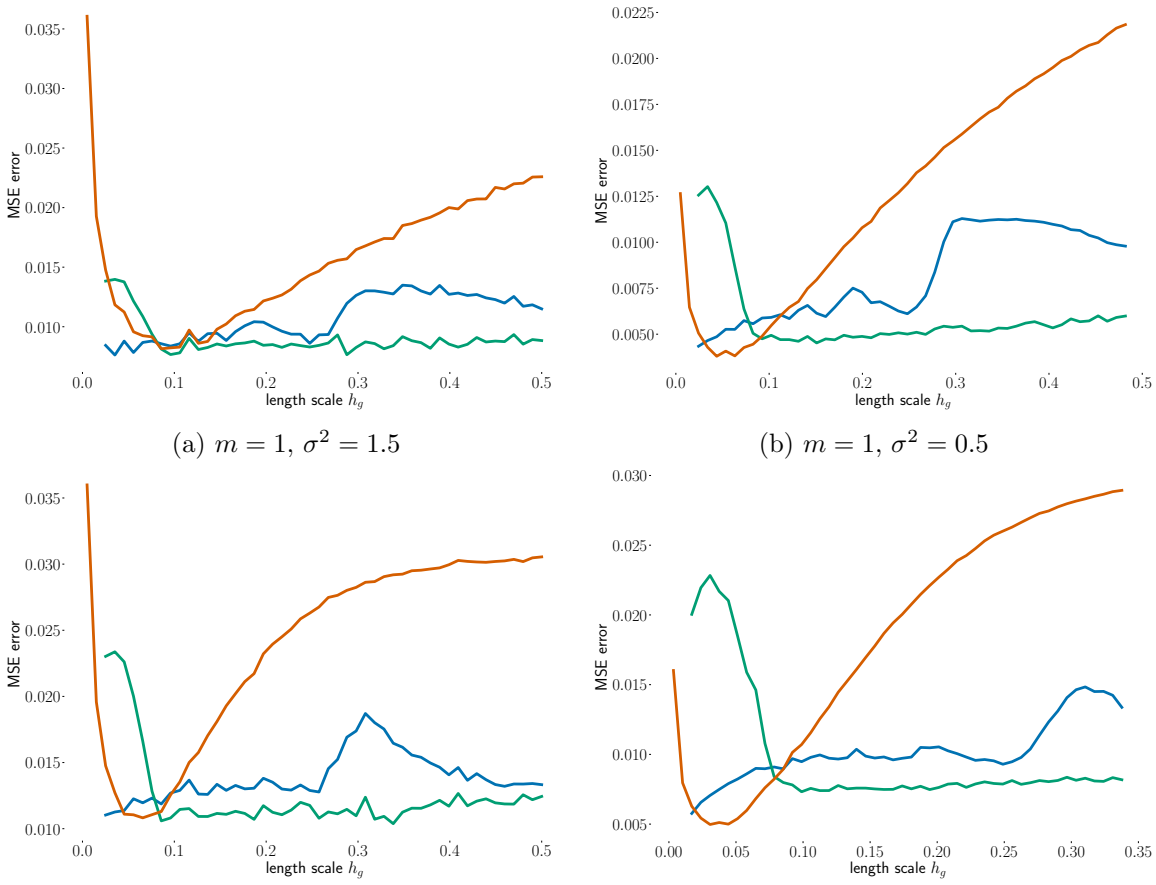
4.1 Bias-Variance trade-off curves

We compute the Bias-Variance trade-off curves in several settings. Given parameters $n \in \mathbb{N}$, $m > 0$ and $\sigma^2 > 0$, we consider the model

$$y_i = \sin(2m\pi\|\boldsymbol{x}_i\|) + \epsilon_i \tag{60}$$

where $\mathbf{X}_n = [\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n]$ are i.i.d. uniform univariate variables on $[0,1]^d$ $(d = 1, 2)$ and ϵ_i are i.i.d. Gaussian variables with variance σ^2. As m increases, so does the number of oscillations of the sine, and therefore the optimal bandwidth $\tau_\star$ needs to shrink to compensate for the irregularity of the signal $\boldsymbol{y}$.

We compute bias variance trade-off curves for GNW and ENW for a wide range of length-scales h_g. For a given set of parameters n, m, σ^2, d, we approximate $\tau_\star$ for the label $\boldsymbol{y}$ given by (60) by cross validation, i.e. we compute τ_{CV}. Then we consider the grid G of $\mathrm{NUM}_{\mathrm{PTS}} \in \mathbb{N}$ regularly spaced points in a window $W_{\tau_{\mathrm{CV}}}$ around τ_{CV} of two orders of magnitude, i.e. $W_{\tau_{\mathrm{CV}}} = [0.1\tau_{\mathrm{CV}}, 10\tau_{\mathrm{CV}}]$. The neighborhood of the leftmost endpoint of this interval could be considered as the narrow length-scale regime, whereas the neighborhood of the rightmost endpoint could be considered as the over-averaging regime. For each point p in the grid G, we generate a LPM with length-scale $h_g = p$, on which we run GNW and ENW with the algorithms 1 and 2. We report the (LOO) MSE (52) for each algorithm and for each point p. This computation is repeated $\mathrm{NUM}_{\mathrm{MC}} \in \mathbb{N}$ times to reduce the variance



(a) $m = 1$, $\sigma^2 = 1.5$

(b) $m = 1$, $\sigma^2 = 0.5$

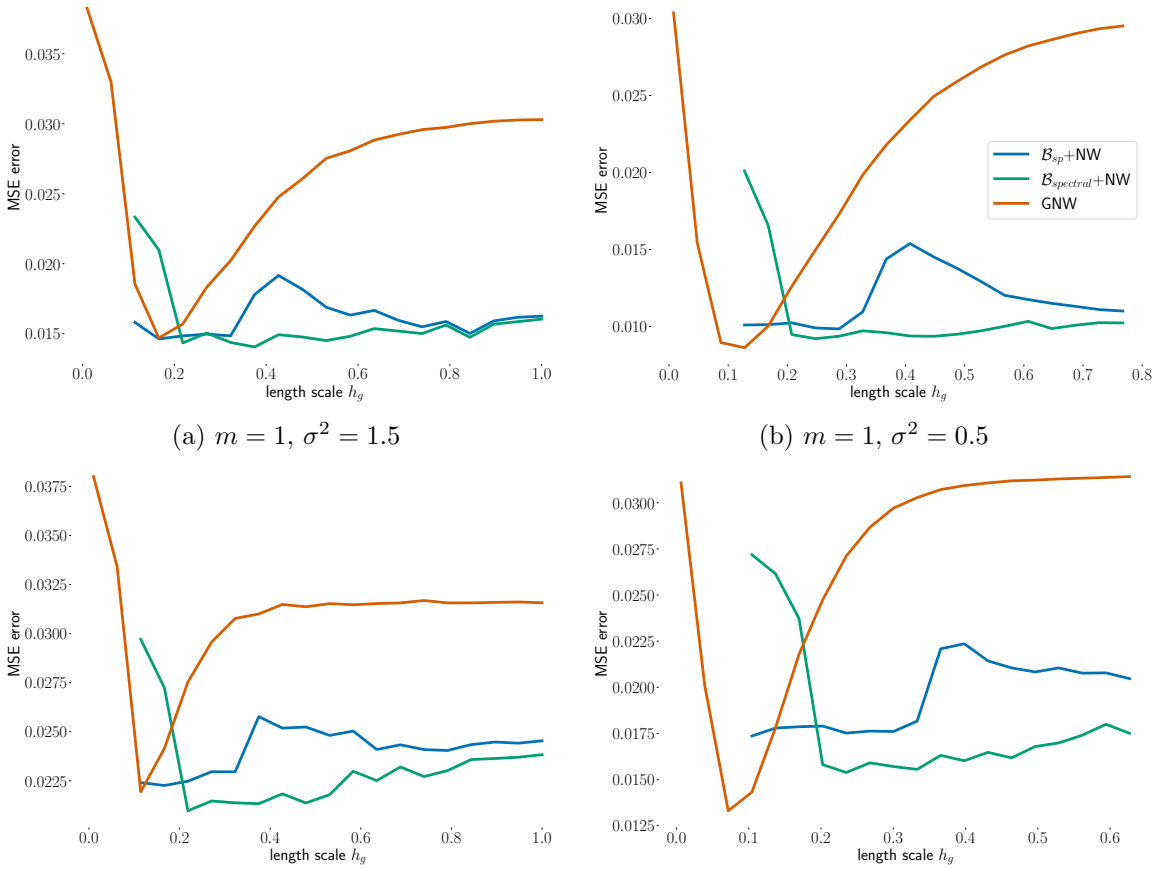
(c) $m = 2$, $\sigma^2 = 1.5$

(d) $m = 2$, $\sigma^2 = 0.5$

Figure 7: "Bias-Variance" trade-off curves for $\mathcal{B}_{sp}$-ENW, $\mathcal{B}_{spectral}$-ENW and GNW in dimension $d = 1$, based on the length-scale h_g. Parameter setup: $n = 500$, $\mathrm{NUM}_{\mathrm{MC}} = 20$, $\mathrm{NUM}_{\mathrm{PTS}} = 50$. The frequency m and the label noise σ^2 vary as specified in the caption of the sub-figures.

due to random edges. Results are displayed in Figure 7 and Figure 8 for dimensions $d = 1$ and $d = 2$, respectively.

We observe that when h_g is close to τ_{CV}, the MSE error of GNW is generally lower than that of ENW, although for certain parameters (see Figure 7a), $\mathcal{B}_{sp}$-ENW and $\mathcal{B}_{spectral}$-ENW are competitive even in this scenario. In the large length-scale regime $\mathcal{B}_{spectral}$-ENW outperforms the other algorithms by a significant margin, however $\mathcal{B}_{sp}$-ENW also shows significant improvement over GNW. For the under-averaging regime, we generally observe that $\mathcal{B}_{sp}$-ENW is the dominant algorithm, provided that the label is sufficiently regular so that τ_{CV} is above the connectivity threshold of the LPM.



(a) $m = 1$, $\sigma^2 = 1.5$ (b) $m = 1$, $\sigma^2 = 0.5$

(c) $m = 2$, $\sigma^2 = 1.5$ (d) $m = 2$, $\sigma^2 = 0.5$

Figure 8: "Bias-Variance" trade-off curves for $\mathcal{B}_{sp}$-ENW, $\mathcal{B}_{spectral}$-ENW and GNW in dimension $d = 2$, based on the length-scale h_g. Parameter setup: $n = 500$, $\text{NUM}_{\text{MC}} = 20$, $\text{NUM}_{\text{PTS}} = 50$. The frequency m and the label noise σ^2 vary as specified in the caption of the sub-figures.

5. Conclusion

We have shown that in a LPM with kernel function (3), GNW matches (up to multiplicative constant) the classical NW rate (21). In particular, GNW is effective even in the extremely narrow length-scale regime, $h_g \ll \tau_\star$, as soon as $h_g \to 0$ and $nh_g^d \to \infty$ - the same assumptions needed on τ for asymptotic convergence of NW (19). Next, in Theorem 22, we have shown that the Nadaraya-Watson estimator is stable with respect to perturbation of the design points, provided that these perturbations are not too large. Using this result, we examined several papers from the literature on position recovery, and we discussed the implications of their results for the node regression problem, with a particular focus on optimal nonparametric rates. In order to construct an estimator that achieves optimal nonparametric rates for a-Hölder continuous regression function f, it is sufficient to construct a position estimation algorithm $\mathcal{A}$ such that $p_{\tau_\star}(\mathcal{A}, \mathbf{X}_{n+1}) \leq Cn^{-\frac{2a}{d+2a}}$, where $\tau_\star \sim n^{-\frac{1}{d+2a}}$. This last question has not been treated in full generality in the literature, but rather it seems that it needs to be treated on a case-by-case basis: different algorithms work in different settings. We have the intuition that the results of position estimation of (Dani et al., 2022) based on the *local, number of common neighbors approach* should be able to extend the optimality in $d > 1$ in the under averaging regime. More detailed characterization of the possibility of obtaining standard rates (22) along with negative results in a minimax sense is left for a future work. It would be interesting to establish nonparametric regression rates for node regression in the latent position model for various smoothness classes for the regression function. For this challenging problem we believe that a combination of classical regression methods and approximate latent positions might deliver interesting results.

Empirically, we observe that if the ratio $h_g/\tau_\star$ is $\Theta(1)$ then GNW performs nearly optimally - indeed, in this case the risk GNW reaches the optimal NW-rate (22). Additionally, ENW is to be preferred in the over-averaging regime $\frac{h_g}{\tau_\star} = \omega(1)$. For sparse graphs, the problem is not as clear. Existing results (56) imply convergence of position estimation in any regime of sparsity, however it seems that GNW is simply faster in this particular case. Due to its computational complexity ($\mathcal{O}(n)$) it should be preferred for extremely sparse graphs, specifically outside the univariate case ($d > 1$). Another case in which GNW might be a good choice is in the case where fast algorithm runtime is emphasized over the statistical quality of the result. Indeed, the computational cost of position estimation is often at least $\Omega(n^2)$, so that there is a trade-off between runtime of the algorithm and its statistical performance.

Appendix: Proofs

A GNW additional results

Recall that in the analysis of GNW, unless explicitly stated otherwise, expectations are taken with respect to $\mathbf{X}_n$, $\boldsymbol{\mathcal{U}}$ and $\boldsymbol{\epsilon}$. In this section we provide details to support the theoretical results of Section 2. In particular, we prove Proposition 3, Lemma 6 and Lemma 7. En route, we will compute the expectation of GNW explicitly. Being a quotient of two random variables, the exact value of $\mathbb{E}\left[\hat{f}_{\mathrm{GNW}}(\boldsymbol{x})\right]$ may seem difficult to compute. We are able to carry out this computation due to the decoupling trick, Lemma 5.

Lemma 26 *Recall that* $R_i(\boldsymbol{x}) = \frac{1}{1+\sum_{j\neq i} a(\boldsymbol{x},\boldsymbol{x}_j)}$. *For all* $i \in [n]$ *we have*

$$\mathbb{E}\left[R_i(\boldsymbol{x})\right] = \frac{1-(1-\frac{d(\boldsymbol{x})}{n})^n}{d(\boldsymbol{x})}$$

Proof Note that $R_i(\boldsymbol{x})$, $i \in [n]$ are identically distributed. Therefore, for $i \in [n]$,

$$\mathbb{E}\left[R_i(\boldsymbol{x})\right] = \mathbb{E}\left[R_1(\boldsymbol{x})\right]$$

Recall Equation (39):

$$\sum_{i=1}^{n} a(\boldsymbol{x},\boldsymbol{x}_i)R_i(\boldsymbol{x}) = Z$$

Taking expectation and using the fact that $R_i(\boldsymbol{x})$ and $a(\boldsymbol{x},\boldsymbol{x}_i)$ are independent, we get

$$\begin{aligned}\mathbb{E}\left[\sum_{i=1}^{n} a(\boldsymbol{x},\boldsymbol{x}_i)R_i(\boldsymbol{x})\right] &= \sum_{i=1}^{n}\mathbb{E}\left[a(\boldsymbol{x},\boldsymbol{x}_i)R_i(\boldsymbol{x})\right] \\ &= \sum_{i=1}^{n}\mathbb{E}\left[a(\boldsymbol{x},\boldsymbol{x}_i)\right]\mathbb{E}\left[R_i(\boldsymbol{x})\right] \\ &= nc(\boldsymbol{x})\mathbb{E}\left[R_1(\boldsymbol{x})\right]\end{aligned} \tag{61}$$

On the other hand,

$$\mathbb{E}\left[Z\right] = \mathbb{P}\left[\sum_{i=1}^{n} a(\boldsymbol{x},\boldsymbol{x}_i) > 0\right] = 1-\mathbb{P}\left[\sum_{i=1}^{n} a(\boldsymbol{x},\boldsymbol{x}_i) = 0\right] = 1-\big(1-c(\boldsymbol{x})\big)^n \tag{62}$$

The result follows by combining Equations (39), (61) and (62). ■

Proposition 27 ***(Expectation of GNW)***

$$\mathbb{E}_{\mathbf{X}_n,\boldsymbol{\epsilon},\boldsymbol{\mathcal{U}}}\left[\hat{f}_{\mathrm{GNW}}(\boldsymbol{x})\right] = S(f,\boldsymbol{x})\left(1-\big(1-c(\boldsymbol{x})\big)^n\right)$$

where $c(\boldsymbol{x})$ *and* $S(f,\boldsymbol{x})$ *are given by* (10) *and* (11).

Proof of Proposition 27 Recall the linearized expression for $\hat{f}_{\mathrm{GNW}}$ (38) i.e. we have

$$\hat{f}_{\mathrm{GNW}}(\boldsymbol{x}) = \sum_{i=1}^{n} y_i a(\boldsymbol{x},\boldsymbol{x}_i)R_i(\boldsymbol{x})$$

Taking expectation and using Lemma 26, we get

$$\begin{aligned}
\mathbb{E}\left[\hat{f}_{\mathrm{GNW}}(\boldsymbol{x})\right] &= \sum_{i=1}^{n} \mathbb{E}\left[y_i a(\boldsymbol{x}, \boldsymbol{x}_i) R_i(\boldsymbol{x})\right] \\
&= \sum_{i=1}^{n} \mathbb{E}\left[y_i a(\boldsymbol{x}, \boldsymbol{x}_i)\right] \mathbb{E}\left[R_i(\boldsymbol{x})\right] \\
&= n\mathbb{E}[y_1 a(\boldsymbol{x}, \boldsymbol{x}_1)] \mathbb{E}\left[R_1(\boldsymbol{x})\right] \\
&= \frac{n(1-(1-c(\boldsymbol{x}))^n)}{nc(\boldsymbol{x})} \int f(\boldsymbol{z}) k(\boldsymbol{x}, \boldsymbol{z}) p(\boldsymbol{z}) dz \\
&= S(f, \boldsymbol{x})(1-(1-c(\boldsymbol{x}))^n)
\end{aligned}$$

■

Finally, the explicit computation of $\mathbb{E}[\hat{f}_{\mathrm{GNW}}]$ enables us to prove Proposition 3.

Proof of Proposition 3 In view of Proposition 27, we have

$$b(\boldsymbol{x}) - \mathrm{Bias}\left[\hat{f}_{\mathrm{GNW}}(\boldsymbol{x})\right] = S(f, \boldsymbol{x}) - \mathbb{E}\left[\hat{f}_{\mathrm{GNW}}(\boldsymbol{x})\right] = S(f, \boldsymbol{x})\left(1 - \frac{d(\boldsymbol{x})}{n}\right)^n \tag{63}$$

Next, we have

$$v(x) = \mathbb{E}\left[\hat{f}^2_{\mathrm{GNW}}(\boldsymbol{x})\right] - 2S(f, \boldsymbol{x})\mathbb{E}\left[\hat{f}_{\mathrm{GNW}}(\boldsymbol{x})\right] + S^2(f, \boldsymbol{x})$$

Again, by using Proposition 27, we get

$$\begin{aligned}
v(\boldsymbol{x}) - \mathrm{Var}(\hat{f}_{\mathrm{GNW}}(\boldsymbol{x})) &= \left(\mathbb{E}\left[\hat{f}_{\mathrm{GNW}}(\boldsymbol{x})\right]\right)^2 - 2S(f, \boldsymbol{x})\mathbb{E}\left[\hat{f}_{\mathrm{GNW}}(\boldsymbol{x})\right] + S^2(f, \boldsymbol{x}) \\
&= \left(S(f, \boldsymbol{x}) - \mathbb{E}\left[\hat{f}_{\mathrm{GNW}}(\boldsymbol{x})\right]\right)^2 \\
&= S^2(f, \boldsymbol{x})\left(1 - \frac{d(\boldsymbol{x})}{n}\right)^{2n}
\end{aligned} \tag{64}$$

The claim follows from Equations (63) and(64) and the basic inequality $1 - t \leq \exp(-t)$. ■

Next, we prove Lemma 6.

Proof of Lemma 6 We begin with a small lemma that also simplifies the notation for the actual calculation.

Lemma 28 *Suppose that $g\colon \mathbb{R}^{d+1} \to \mathbb{R}$ is a measurable function such that for all $i \in [n]$, $\mathbb{E}[g^2(\boldsymbol{x}_i, \epsilon_i)] < \infty$. For $i \in [n]$ set $F_i = g(\boldsymbol{x}_i, \epsilon_i)$. Then for all pairs of distinct indices $1 \leq i, j \leq n$ we have*

$$\mathbb{E}\left[F_i F_j a(\boldsymbol{x}, \boldsymbol{x}_i) a(\boldsymbol{x}, \boldsymbol{x}_j) R_i(\boldsymbol{x}) R_j(\boldsymbol{x})\right] = \mathbb{E}\left[F_i a(\boldsymbol{x}, \boldsymbol{x}_i)\right] \mathbb{E}\left[F_j a(\boldsymbol{x}, \boldsymbol{x}_j)\right] \mathbb{E}\left[R^2_{\{i,j\}}(\boldsymbol{x})\right]$$

Proof Using the decoupling trick 5 we have

$$F_i F_j a(\boldsymbol{x}, \boldsymbol{x}_i) a(\boldsymbol{x}, \boldsymbol{x}_j) R_i(\boldsymbol{x}) R_j(\boldsymbol{x}) = F_i F_j a(\boldsymbol{x}, \boldsymbol{x}_i) a(\boldsymbol{x}, \boldsymbol{x}_j) R^2_{\{i,j\}}(\boldsymbol{x})$$

and moreover $R_{\{i,j\}}(\boldsymbol{x})$ is independent of $\big(\boldsymbol{x}_i, \epsilon_i, a(\boldsymbol{x}, \boldsymbol{x}_i)\big)$ and $\big(\boldsymbol{x}_j, \epsilon_j, a(\boldsymbol{x}, \boldsymbol{x}_j)\big)$. Next, $\big(\boldsymbol{x}_i, \epsilon_i, a(\boldsymbol{x}, \boldsymbol{x}_i)\big)$ and $\big(\boldsymbol{x}_j, \epsilon_j, a(\boldsymbol{x}, \boldsymbol{x}_j)\big)$ are also independent by modeling assumptions[12]. As independent variables are uncorrelated, the conclusion follows. ■

Set $g(\boldsymbol{x}_i, \epsilon_i) = y_i - S(f, \boldsymbol{x})$. Using Equation (40), we have

$$\begin{aligned}\mathbb{E}\left[\Big(\hat{f}_{\mathrm{GNW}}(\boldsymbol{x}) - S(f,\boldsymbol{x})Z\Big)^2\right] &= \mathbb{E}\left[\left(\sum_{i=1}^{n}(y_i - S(f,\boldsymbol{x}))a(\boldsymbol{x},\boldsymbol{x}_i)R_i(\boldsymbol{x})\right)^2\right] \\ &= \sum_{i=1}^{n}\mathbb{E}\left[\big(g(\boldsymbol{x}_i,\epsilon_i)a(\boldsymbol{x},\boldsymbol{x}_i)R_i(\boldsymbol{x})\big)^2\right] \\ &\quad + \sum_{i\neq j}\mathbb{E}\left[g(\boldsymbol{x}_i,\epsilon_i)g(\boldsymbol{x}_j,\epsilon_j)a(\boldsymbol{x},\boldsymbol{x}_i)a(\boldsymbol{x},\boldsymbol{x}_j)R_i(\boldsymbol{x})R_j(\boldsymbol{x})\right]\end{aligned} \tag{65}$$

For $i \neq j$, applying Lemma 28 with $g\colon \mathbb{R}^{d+1} \to \mathbb{R}$ given by $g(\cdot, \star) = (f(\cdot) + \star) - S(f, \boldsymbol{x})$ along with the fact that $\mathbb{E}[g(\boldsymbol{x}_i, \epsilon_i)a(\boldsymbol{x}, \boldsymbol{x}_i)] = 0$ gives

$$\mathbb{E}\left[g(\boldsymbol{x}_i,\epsilon_i)g(\boldsymbol{x}_j,\epsilon_j)a(\boldsymbol{x},\boldsymbol{x}_i)a(\boldsymbol{x},\boldsymbol{x}_j)R_i(\boldsymbol{x})R_j(\boldsymbol{x})\right] = 0 \tag{66}$$

Finally,

$$\sum_{i=1}^{n}\mathbb{E}\left[\big(g(\boldsymbol{x}_i,\epsilon_i)a(\boldsymbol{x},\boldsymbol{x}_i)R_i(\boldsymbol{x})\big)^2\right] = \sum_{i=1}^{n}\mathbb{E}\left[(y_i - S(f,\boldsymbol{x}))^2 a(\boldsymbol{x},\boldsymbol{x}_i)R_i^2(\boldsymbol{x})\right]$$

■

Finally, we conclude the variance analysis of GNW by a proof of Lemma 7.

Proof of Lemma 7 From Equation (32), Lemma 6, as well as Equation (42), Lemma 26 and the basic inequality $1 - t \leq e^{-t}$ valid for all $t \geq 0$, we have

$$\begin{aligned} v(\boldsymbol{x}) &\geq \mathbb{E}\left[\Big(\hat{f}_{\mathrm{GNW}}(\boldsymbol{x}) - S(f,\boldsymbol{x})Z\Big)^2\right] \\ &= \sum_{i=1}^{n}\mathbb{E}_{\boldsymbol{\mathcal{U}},\mathbf{X}_n}\left[\mathbb{E}_{\boldsymbol{\epsilon}}[\big(y_i - S(f,\boldsymbol{x}_i)\big)^2]a(\boldsymbol{x},\boldsymbol{x}_i)R_i^2(\boldsymbol{x})\right] \\ &\geq \sigma_0^2 d(\boldsymbol{x})\mathbb{E}\left[R_1^2(\boldsymbol{x})\right] \\ &\geq \sigma_0^2 d(\boldsymbol{x})\Big(\mathbb{E}\left[R_1(\boldsymbol{x})\right]\Big)^2 \\ &= \sigma_0^2 d(\boldsymbol{x})\left(\frac{\Big(1 - (1 - \frac{d(\boldsymbol{x})}{n})\Big)}{d(\boldsymbol{x})}\right)^2 \\ &\geq \sigma_0^2\frac{\Big(1 - e^{-d(\boldsymbol{x})}\Big)^2}{d(\boldsymbol{x})} \end{aligned}$$

12. conditioning on $\boldsymbol{x}_{n+1} = \boldsymbol{x}$, i.e. treating the latent position of node $(n+1)$ as a deterministic quantity is crucial in this part of the proof

■

B Bias and Risk of GNW

Proof of Lemma 13 Our first claim is that under our assumptions, For $\boldsymbol{x} \in Q$, we have $c(\boldsymbol{x}) = \int K\left(\frac{\|\boldsymbol{x}-\boldsymbol{z}\|}{h_g}\right) p(\boldsymbol{z})dz > 0$ and hence the operator $S(f, \boldsymbol{x})$ (11) is non-trivial. Indeed, suppose that Assumption 8 holds. We will show that for every $\boldsymbol{x} \in Q$, $c(\boldsymbol{x}) > 0$ where Q is the support of the distribution p. Suppose that $c(\boldsymbol{x}) = 0$. Using Assumption 8 we get

$$\begin{aligned}
\alpha \int \mathbb{I}\left[\|\boldsymbol{x}-\boldsymbol{z}\| \leq M_1 h_g\right] p(\boldsymbol{z})dz &\leq 2\alpha \int \mathbb{I}\left[\|\boldsymbol{x}-\boldsymbol{z}\| \leq M_1 h_g\right] K\left(\frac{\|\boldsymbol{x}-\boldsymbol{z}\|}{h_g}\right) p(\boldsymbol{z})dz \\
&\leq 2\alpha \int K\left(\frac{\|\boldsymbol{x}-\boldsymbol{z}\|}{h_g}\right) p(\boldsymbol{z})dz \\
&= 2c(\boldsymbol{x}) \\
&= 0
\end{aligned}$$

As $\alpha > 0$, $\boldsymbol{x} \notin \text{supp}(p) = Q$, and hence our claim follows form the contrapositive. Hence, for $\boldsymbol{x} \in Q$, we have

$$\begin{aligned}
|S(f, \boldsymbol{x}) - f(\boldsymbol{x})| &= \left| \frac{\int f(\boldsymbol{z}) K\left(\frac{\|\boldsymbol{x}-\boldsymbol{z}\|}{h_g}\right) p(\boldsymbol{z})dz}{\int K\left(\frac{\|\boldsymbol{x}-\boldsymbol{z}\|}{h_g}\right) p(\boldsymbol{z})dz} - f(\boldsymbol{x}) \right| \\
&= \left| \frac{\int f(\boldsymbol{z}) K\left(\frac{\|\boldsymbol{x}-\boldsymbol{z}\|}{h_g}\right) p(\boldsymbol{z})dz}{\int K\left(\frac{\|\boldsymbol{x}-\boldsymbol{z}\|}{h_g}\right) p(\boldsymbol{z})dz} - \frac{\int f(\boldsymbol{x}) K\left(\frac{\|\boldsymbol{x}-\boldsymbol{z}\|}{h_g}\right) p(\boldsymbol{z})dz}{\int K\left(\frac{\|\boldsymbol{x}-\boldsymbol{z}\|}{h_g}\right) p(\boldsymbol{z})dz} \right| \\
&= \left| \frac{\int_Q [f(\boldsymbol{z}) - f(\boldsymbol{x})] K\left(\frac{\|\boldsymbol{x}-\boldsymbol{z}\|}{h_g}\right) p(\boldsymbol{z})dz}{\int_Q K\left(\frac{\|\boldsymbol{x}-\boldsymbol{z}\|}{h_g}\right) p(\boldsymbol{z})dz} \right| \\
&\leq L \frac{\int_Q \|\boldsymbol{z}-\boldsymbol{x}\|^\alpha K\left(\frac{\|\boldsymbol{x}-\boldsymbol{z}\|}{h_g}\right) p(\boldsymbol{z})dz}{\int_Q K\left(\frac{\|\boldsymbol{x}-\boldsymbol{z}\|}{h_g}\right) p(\boldsymbol{z})dz} \\
&\leq 2LM_2^\alpha h_g^\alpha
\end{aligned}$$

where we used Assumption 9 in the first and Assumption 8 in the second inequality. ■

Proof of Lemma 14 By Assumption 8 and the assumption that Q satisfies 10 with parameters $(r_0, c_0)-$ we have

$$\begin{aligned}
\frac{c(\boldsymbol{x})}{\alpha} &= \int K\left(\frac{\|\boldsymbol{x}-\boldsymbol{z}\|}{h_g}\right) p(\boldsymbol{z})dz \\
&\geq \frac{1}{2}\int \mathbb{I}\left[\|\boldsymbol{x}-\boldsymbol{z}\| \leq M_1 h_g\right] p(\boldsymbol{z})dz \\
&\geq \frac{p_0(\boldsymbol{x})}{2}\int \mathbb{I}\left[\|\boldsymbol{x}-\boldsymbol{z}\| \leq M_1 h_g\right] \mathbb{I}\left[\boldsymbol{z}\in Q\right] dz \\
&= \frac{p_0(\boldsymbol{x})}{2} m(Q \cap B_{M_1 h_g}(\boldsymbol{x})) \\
&\geq \frac{p_0(\boldsymbol{x})c_0}{2} m(B_{M_1 h_g}(\boldsymbol{x})) \\
&= c_0 v_d M_1^d h_g^d p_0(\boldsymbol{x})/2 > 0
\end{aligned}$$

The conclusion follows $d(\boldsymbol{x}) = nc(\boldsymbol{x})$. ■

Proof of Theorem 15 and Theorem 16 We use the bias and variance proxies to bound the risk via the following inequality

$$\mathcal{R}_g(\hat{f}_{\mathrm{GNW}}(\boldsymbol{x}), f(\boldsymbol{x})) \leq 2(v(\boldsymbol{x}) + b^2(\boldsymbol{x})) \tag{67}$$

On one hand, from Lemma 13 we see that under Assumptions 8 and 9, we have

$$|b(\boldsymbol{x})| \leq 2LM_2^a h_g^a$$

On the other hand, combining the bound in 4 along with Lemma 14, and taking into account assumptions 8, 10 and Equation (46) we arrive at the following

$$v(\boldsymbol{x}) \leq \frac{18B^2 + 4\sigma^2}{c_0 v_d M_1^d n\alpha h_g^d p_0(\boldsymbol{x})}$$

The conclusion for Theorem 15 follows form Equation (67). Under Assumption 11, Equation (46) holds with $p_0(\boldsymbol{x}) \equiv p_0$. The conclusion for Theorem 16 follows immediately from integrating the bound in Theorem 15. ■

Before we proceed with the proof of Theorem 17, we need to show that the variance of GNW can not blow up. The following lemma will be useful for ENW analysis as well.

Lemma 29 *Suppose that ϵ_i satisfying Assumption 2 and $0 \leq w_i \leq 1$ are real numbers such that $\sum_{i=1}^n w_i > 0$. Then*

$$\mathbb{E}_{\boldsymbol{\epsilon}}\left[\left(\frac{\sum_{i=1}^n \epsilon_i w_i}{\sum_{i=1}^n w_i}\right)^2\right] \leq \sigma^2 \min\{\frac{1}{\sum_{i=1}^n w_i}, 1\} \tag{68}$$

Proof Note that

$$\mathbb{E}_{\boldsymbol{\epsilon}}\left[\left(\frac{\sum_{i=1}^n \epsilon_i w_i}{\sum_{i=1}^n w_i}\right)^2\right] \leq \sigma^2 \frac{\sum_{i=1}^n w_i^2}{\left(\sum_{i=1}^n w_i\right)^2} \tag{69}$$

Using $w_i \leq 1$, we get the first inequality. The second inequality easily follows from the observation that $0 \leq \frac{w_i}{\sum_{i=1}^n w_i} \leq 1$. Namely, setting $v_i = \frac{w_i}{\sum_{i=1}^n w_i}$, we have $v_i \geq 0$ and $\sum_{i=1}^n v_i = 1$, and hence $v_i \leq 1$, for all $i \in [n]$. Now

$$\frac{\sum_{i=1}^n w_i^2}{\left(\sum_{i=1}^n w_i\right)^2} = \sum_{i=1}^n v_i^2 \leq \sum_{i=1}^n v_i = 1$$

Concluding the proof. ■

Proof of Theorem 17 Using Lemma 29, we get

$$\mathcal{R}_g\left(\hat{f}_{\mathrm{GNW}}(\boldsymbol{x}), f(\boldsymbol{x})\right) \leq 2\min\{b^2(\boldsymbol{x}) + v(\boldsymbol{x}), 4B^2 + \sigma^2\}$$

The idea is to split the integral in the global risk (18) in two parts, the first where the density is sufficiently high i.e. $p(\boldsymbol{x}) \geq 2SM_1^b h_g^b$, where we use the bounds from Theorem 15 and the second, where the density is low and on which we use the bound $8B^2 + 2\sigma^2$. From Assumption 12 we have

$$\inf_{\substack{\boldsymbol{z}\in Q \\ \|\boldsymbol{x}-\boldsymbol{z}\|\leq M_1 h_g}} p(\boldsymbol{z}) \geq p(\boldsymbol{x}) - SM_1^b h_g^b$$

and hence, when $p(\boldsymbol{x}) \geq 2SM_1^b h_g^b$, we have

$$\inf_{\substack{\boldsymbol{z}\in Q \\ \|\boldsymbol{x}-\boldsymbol{z}\|\leq M_1 h_g}} p(\boldsymbol{z}) \geq SM_1^b h_g^b$$

Therefore, for $\boldsymbol{x}$ such that $p(\boldsymbol{x}) \geq 2SM_1^b h_g^b$, we have that Theorem 15 is satisfied with $p_0(\boldsymbol{x}) = SM_1 h_g^b$. We conclude as follows.

$$\begin{aligned}
\mathcal{R}_g\left(\hat{f}_{\mathrm{GNW}}, f\right) &= \int \mathcal{R}_g\left(\hat{f}_{\mathrm{GNW}}(\boldsymbol{x}), f(\boldsymbol{x})\right) p(\boldsymbol{x}) dx \\
&\leq 2\int \left(v(\boldsymbol{x}) + b^2(\boldsymbol{x})\right) \mathbb{I}\left[p(\boldsymbol{x}) \geq 2SM_1^b h_g^b\right] p(\boldsymbol{x}) dx \\
&+ (8B^2 + 2\sigma^2)\int \mathbb{I}\left[p(\boldsymbol{x}) \leq 2SM_1^b h_g^b\right] p(\boldsymbol{x}) dx \\
&\leq 4L^2 M_2^{2a} h_g^{2a} + \frac{36B^2 + 8\sigma^2}{c_0 v_d S M_1^{d+b} n\alpha h_g^{d+b}} \\
&+ (8B^2 + 2\sigma^2) S^{1/2} M_1^{b/2} h_g^{b/2} \int p^{1/2}(\boldsymbol{x}) dx
\end{aligned}$$

■

C Estimated Nadaraya-Watson

Proof of Theorem 22 Using Assumption 20, we have

$$\sum_{i=1}^{n} \phi(\tilde{\delta}_i/\tau) \geq \frac{1}{2} \sum_{i=1}^{n} \mathbb{I}(\tilde{\delta}_i \leq M_1 \tau) \tag{70}$$

Next, using the assumption that $\Delta(\mathcal{A}, \mathbf{X}_{n+1}) \leq \frac{M_1 \tau}{2}$, for all $i \in [n]$ we have

$$\frac{\tilde{\delta}_i}{\tau} = \frac{\delta_i}{\tau} + \frac{\tilde{\delta}_i - \delta_i}{\tau} \leq \frac{\delta_i}{\tau} + \frac{M_1}{2} \tag{71}$$

Hence if $\delta_i/\tau \leq M_1/2$ then $\tilde{\delta}_i/\tau \leq M_1/2 + M_1/2 = M_1$. Consequently, using Equation (70), we get that

$$\sum_{i=1}^{n} \phi(\tilde{\delta}_i/\tau) \geq \frac{1}{2} M(\tau) > 0 \tag{72}$$

In particular, under the assumptions of Theorem 22, we do not need to worry about the degenerate case $\sum_{i=1}^{n} \phi(\tilde{\delta}_i/\tau) = 0$, in which we assign value 0 by default to the estimator. We have

$$\begin{aligned}
\hat{f}^{\mathcal{A}}_{\mathrm{ENW},\tau}(\boldsymbol{x}_{n+1}) &= \frac{\sum_{i=1}^{n} y_i \phi\left(\frac{\tilde{\delta}_i}{\tau}\right)}{\sum_{i=1}^{n} \phi\left(\frac{\tilde{\delta}_i}{\tau}\right)} \\
&= \frac{\sum_{i=1}^{n} f(\boldsymbol{x}_i) \phi\left(\frac{\tilde{\delta}_i}{\tau}\right)}{\sum_{i=1}^{n} \phi\left(\frac{\tilde{\delta}_i}{\tau}\right)} + \frac{\sum_{i=1}^{n} \epsilon_i \phi\left(\frac{\tilde{\delta}_i}{\tau}\right)}{\sum_{i=1}^{n} \phi\left(\frac{\tilde{\delta}_i}{\tau}\right)} \\
&= f(\boldsymbol{x}_{n+1}) + \frac{\sum_{i=1}^{n} (f(\boldsymbol{x}_i) - f(\boldsymbol{x}_{n+1})) \phi\left(\frac{\tilde{\delta}_i}{\tau}\right)}{\sum_{i=1}^{n} \phi\left(\frac{\tilde{\delta}_i}{\tau}\right)} \qquad (73) \\
&\quad + \frac{\sum_{i=1}^{n} \epsilon_i \phi\left(\frac{\tilde{\delta}_i}{\tau}\right)}{\sum_{i=1}^{n} \phi\left(\frac{\tilde{\delta}_i}{\tau}\right)} \qquad (74)
\end{aligned}$$

We take care of the two terms separately. If $\phi(\tilde{\delta}_i/\tau) > 0$, then from Assumption 20 we get $\tilde{\delta}_i \leq M_2 \tau$, so that Equation (71) implies

$$\delta_i^a \phi(\tilde{\delta}_i/\tau) \leq \left(\tilde{\delta}_i + \frac{M_1 \tau}{2}\right)^a \phi(\tilde{\delta}_i/\tau) \leq (M_2 + M_1/2)^a \tau^a \phi(\tilde{\delta}_i/\tau) \tag{75}$$

Finally, Equation (75) yields the following bound on (73)

$$\begin{aligned}
\left| \frac{\sum_{i=1}^{n} \left(f(\boldsymbol{x}_i) - f(\boldsymbol{x}_{n+1})\right) \phi\left(\frac{\tilde{\delta}_i}{\tau}\right)}{\sum_{i=1}^{n} \phi\left(\frac{\tilde{\delta}_i}{\tau}\right)} \right| &\leq L \frac{\sum_{i=1}^{n} \delta_i^a \phi\left(\frac{\tilde{\delta}_i}{\tau}\right)}{\sum_{i=1}^{n} \phi\left(\frac{\tilde{\delta}_i}{\tau}\right)} \\
&\leq L (M_2 + M_1/2)^a \tau^a \qquad (76)
\end{aligned}$$

In order to bound the expression (74), we apply Lemma 29 with $w_i = \phi(\tilde{\delta}_i/\tau)$, yielding

$$\mathbb{E}_\epsilon \left[\left(\frac{\sum_{i=1}^n \epsilon_i \phi\left(\frac{\tilde{\delta}_i}{\tau}\right)}{\sum_{i=1}^n \phi\left(\frac{\tilde{\delta}_i}{\tau}\right)} \right)^2 \right] \leq \frac{\sigma^2}{\sum_{i=1}^n \phi(\tilde{\delta}_i/\tau)} \leq \frac{2\sigma^2}{M(\tau)} \tag{77}$$

where we have used (72) in the last inequality. We get the claimed result by combining equations (76) and (77) with the basic inequality $(a+b)^2 \leq 2(a^2+b^2)$. ■

Proof of Theorem 23 Our strategy is to analyze two cases separately: when the position recovery algorithm approximates the latent distances within precision $M_1\tau/2$ and when it fails to do so. We introduce the notation

$$S_{\mathcal{A}} = \{\Delta(\mathcal{A}(\mathbf{A}), \mathbf{X}_{n+1}) \leq \frac{M_1\tau}{2}\} \tag{78}$$

for the event that indicates success of the algorithm $\mathcal{A}$, and $S_{\mathcal{A}}^c$ for its complement. We have

$$\begin{aligned}\mathbb{E}_{\boldsymbol{\mathcal{U}},\boldsymbol{\epsilon}} \left[\left(\hat{f}^{\mathcal{A}}_{\mathrm{ENW},\tau}(\boldsymbol{x}_{n+1}) - f(\boldsymbol{x}_{n+1}) \right)^2 \right] &= \mathbb{E}_{\boldsymbol{\mathcal{U}},\boldsymbol{\epsilon}} \left[\left(\hat{f}^{\mathcal{A}}_{\mathrm{ENW},\tau}(\boldsymbol{x}_{n+1}) - f(\boldsymbol{x}_{n+1}) \right)^2 \mathbb{I}(S_{\mathcal{A}}) \right] \\ &\quad + \mathbb{E}_{\boldsymbol{\mathcal{U}},\boldsymbol{\epsilon}} \left[\left(\hat{f}^{\mathcal{A}}_{\mathrm{ENW},\tau}(\boldsymbol{x}_{n+1}) - f(\boldsymbol{x}_{n+1}) \right)^2 \mathbb{I}(S_{\mathcal{A}}^c) \right]\end{aligned}$$

When the position recovery algorithm $\mathcal{A}$ estimates the latent distances $\boldsymbol{\delta}$ with precision $\frac{M_1\tau}{2}$, then the conditions of Theorem 22 are satisfied. Hence, we have

$$\begin{aligned}\mathbb{E}_{\boldsymbol{\mathcal{U}},\boldsymbol{\epsilon}} \left[\left(\hat{f}^{\mathcal{A}}_{\mathrm{ENW},\tau}(\boldsymbol{x}_{n+1}) - f(\boldsymbol{x}_{n+1}) \right)^2 \mathbb{I}\left[S_{\mathcal{A}}\right] \right] &= \mathbb{E}_{\boldsymbol{\mathcal{U}}} \left[\mathbb{E}_{\boldsymbol{\epsilon}} \left(\hat{f}^{\mathcal{A}}_{\mathrm{ENW},\tau}(\boldsymbol{x}_{n+1}) - f(\boldsymbol{x}_{n+1}) \right)^2 \mathbb{I}(S_{\mathcal{A}}) \right] \\ &\leq \left(C_1\tau^{2a} + \frac{12\sigma^2}{M(\tau)} \right) \mathbb{E}_{\boldsymbol{\mathcal{U}}} \left[\mathbb{I}(S_{\mathcal{A}}) \right] \\ &\leq C_1\tau^{2a} + \frac{12\sigma^2}{M(\tau)} \end{aligned} \tag{79}$$

Next, we need to analyze what happens when the position recovery algorithm $\mathcal{A}$ fails. The idea will be to average over the additive noise of the label first, $\boldsymbol{\epsilon}$. In this case, we want to show that $\mathbb{E}_{\boldsymbol{\epsilon}} \left[\left(\hat{f}^{\mathcal{A}}_{\mathrm{ENW},\tau}(\boldsymbol{x}_{n+1}) - f(\boldsymbol{x}_{n+1}) \right) \right]^2$ remains bounded. If $\sum_{i=1}^n \phi(\tilde{\delta}_i/\tau) = 0$, then by definition $\hat{f}^{\mathcal{A}}_{\mathrm{ENW},\tau}(\boldsymbol{x}_{n+1}) = 0$ and

$$\mathbb{E}_{\boldsymbol{\epsilon}} \left[\left(\hat{f}^{\mathcal{A}}_{\mathrm{ENW},\tau}(\boldsymbol{x}_{n+1}) - f(\boldsymbol{x}_{n+1}) \right)^2 \right] \leq \max_{\boldsymbol{x} \in Q} |f(\boldsymbol{x})|^2 \leq B^2$$

Otherwise, we have $\sum_{i=1}^n \phi(\tilde{\delta}_i/\tau) > 0$ and Lemma 29 yields

$$\mathbb{E}_{\boldsymbol{\epsilon}} \left[\left(\hat{f}^{\mathcal{A}}_{\mathrm{ENW},\tau}(\boldsymbol{x}_{n+1}) - f(\boldsymbol{x}_{n+1}) \right)^2 \right] \leq 4B^2 + 2\sigma^2$$

Finally,

$$\begin{aligned}\mathbb{E}_{\boldsymbol{\mathcal{U}},\boldsymbol{\epsilon}}\left[\left(\hat{f}^{\mathcal{A}}_{\mathrm{ENW},\tau}(\boldsymbol{x}_{n+1})-f(\boldsymbol{x}_{n+1})\right)^2\mathbb{I}(S^c_{\mathcal{A}})\right] &= \mathbb{E}_{\boldsymbol{\mathcal{U}}}\left[\mathbb{E}_{\boldsymbol{\epsilon}}\left(\hat{f}^{\mathcal{A}}_{\mathrm{ENW},\tau}(\boldsymbol{x}_{n+1})-f(\boldsymbol{x}_{n+1})\right)^2\mathbb{I}(S^c_{\mathcal{A}})\right]\\ &\leq \left(4B^2+2\sigma^2\right)\mathbb{E}_{\boldsymbol{\mathcal{U}}}\left[\mathbb{I}(S^c_{\mathcal{A}})\right]\\ &= \left(4B^2+2\sigma^2\right)p_\tau(\mathcal{A},\mathbf{X}) \qquad (80)\end{aligned}$$

Combining Equations (79) and (80) we get the claimed result. ■

Proof of Corollary 24 We will investigate for which parameters h_g, a and d the result (55) yields standard nonparametric rates for $\mathcal{A}_{sp}$-ENW. An i.i.d. sample $\mathbf{X}_n$ with distribution p satisfying Assumptions 10 and 11 will satisfy $\epsilon = (\frac{\log(n)}{n})^{1/d}$ with overwhelming probability[13] over $\mathbf{X}_n$. Since the probability $p_\tau(\mathcal{A},\mathbf{X}_{n+1})$ (53) is decreasing in τ, we get that for

$$\tau \geq C_2\left(h_g+\left(\frac{\epsilon}{h_g}\right)^{\frac{1}{1+A}}\right) \qquad (81)$$

we have $p_\tau(\mathcal{A}_{sp},\mathbf{X}_{n+1}) \leq \frac{1}{n}$. Specifically, if we want to achieve NW optimality for $\tau_\star = c_\star n^{-\frac{1}{d+2a}}$, we need $\tau_\star$ to satisfy inequality (81). This yields $h_g \lesssim \tau_\star$ and $(\frac{\epsilon}{h_g})^{\frac{1}{1+A}} \lesssim \tau_\star$, which further limits the interval of admissible length-scales h_g:

$$\frac{\epsilon}{\tau_\star^{1+A}} \lesssim h_g \lesssim \tau_\star$$

In order for this interval to be non-empty we need $\epsilon \lesssim \tau_\star^{2+A}$. Hence, we need to have

$$\frac{\log(n)}{n} \lesssim n^{-\frac{d(2+A)}{d+2a}}$$

Keeping in mind that we are constrained to $0 < a \leq 1$, this yields

$$d(2+A) < d+2a$$

which is only possible for $d=1$ and $A > 1/2$. Conversely, it is easy to check that when $d=1$ and $a > (1+A)/2$, and h_g is as in the assumption, $\tau_\star$ satisfies Equation (81), which concludes the theorem.

■

Proof of Corollary 25 After some simple calculations, it is easy to see that for the specified values h_g, we have

$$\mathrm{D}\left(\mathcal{B}_{rgg},\mathbf{X}_{n+1}\right) \leq \frac{M_1\tau_\star}{2} = c_\star n^{-\frac{1}{2a+d}}$$

13. This can be shown by a covering number argument.

and hence $p_{\tau_\star}(\mathcal{B}_{rgg}, \mathbf{X}_{n+1}) = 0$ (with high probability over the drawn points $\mathbf{X}_{n+1}$) i.e. $\mathcal{B}_{rgg}$-ENW achieves optimal rates for $(h_g, \tau_\star)$.

■

Acknowledgments

This work was supported by ANR GRandMa (ANR-21-CE23-0006) and ERC-2024-STG-101163069 MALAGA.

References

Emmanuel Abbe. Community detection and stochastic block models: recent developments. *Journal of Machine Learning Research*, 18(177):1–86, 2018.

Réka Albert and Albert-László Barabási. Statistical mechanics of complex networks. *Reviews of Modern Physics*, 74(1):47–97, 2002.

Josh Alman and Virginia Vassilevska Williams. A refined laser method and faster matrix multiplication. In *Proceedings of the 32nd Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 522–539. SIAM, 2021. doi: 10.1137/1.9781611976465.32. URL `https://arxiv.org/abs/2010.05846`.

Greg W. Anderson, Alice Guionnet, and Ofer Zeitouni. *An Introduction to Random Matrices*. Cambridge Studies in Advanced Mathematics. Cambridge University Press, 2009.

Ery Arias-Castro, Antoine Channarond, Bruno Pelletier, and Nicolas Verzelen. On the estimation of latent distances using graph distances, 2018. URL `https://arxiv.org/abs/1804.10611`.

Avanti Athreya, Donniell E. Fishkind, Keith Levin, Vince Lyzinski, Youngser Park, Yichen Qin, Daniel L. Sussman, Minh Tang, Joshua T. Vogelstein, and Carey E. Priebe. Statistical inference on random dot product graphs: a survey. *Journal of Machine Learning Research*, 18(226):1–92, 2018. URL `https://jmlr.org/papers/v18/17-448.html`.

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.

Yu Chen, Sampath Kannan, and Sanjeev Khanna. Near-perfect recovery in the one-dimensional latent space model. In Yennun Huang, Irwin King, Tie-Yan Liu, and Maarten van Steen, editors, *Proc. of WWW*, pages 1932–1942. ACM / IW3C2, 2020.

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. URL `http://jmlr.org/papers/v25/23-0965.html`.

Varsha Dani, Josep Díaz, Thomas P. Hayes, and Cristopher Moore. Reconstruction of random geometric graphs: Breaking the omega(r) distortion barrier, 2022. URL https://arxiv.org/abs/2107.14323.

Santo Fortunato. Community detection in graphs. *Physics Reports*, 486(3–5):75–174, February 2010. ISSN 0370-1573.

Fengnan Gao, Zongming Ma, and Hongsong Yuan. Community detection in sparse latent space models. *Journal of Machine Learning Research*, 23(322):1–50, 2022. URL http://jmlr.org/papers/v23/20-1036.html.

E. N. Gilbert. Random Graphs. *The Annals of Mathematical Statistics*, 30(4):1141 – 1144, 1959.

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

Alden Green, Sivaraman Balakrishnan, and Ryan J. Tibshirani. Minimax optimal regression over sobolev spaces via laplacian regularization on neighborhood graphs, 2021. URL https://arxiv.org/abs/2106.01529.

Aditya Grover and Jure Leskovec. node2vec: Scalable feature learning for networks. In Balaji Krishnapuram, Mohak Shah, Alexander J. Smola, Charu C. Aggarwal, Dou Shen, and Rajeev Rastogi, editors, *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, August 13-17, 2016*, pages 855–864. ACM, 2016.

László Györfi, Michael Kohler, Adam Krzyzak, and Harro Walk. *A Distribution-Free Theory of Nonparametric Regression.* Springer series in statistics. Springer, 2002. ISBN 978-0-387-95441-7.

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.

Paul Holland, Kathryn B. Laskey, and Samuel Leinhardt. Stochastic blockmodels: First steps. *Social Networks*, 5:109–137, 1983.

Thomas N. Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. In *Proc. of ICLR*. OpenReview.net, 2017.

Arne Kovac and Andrew D. A. C. Smith. Nonparametric regression on a graph. *Journal of Computational and Graphical Statistics*, 20(2):432–447, 2011. doi: 10.1198/jcgs.2011.09203.

Can M. Le, Elizaveta Levina, and Roman Vershynin. Concentration and regularization of random graphs. *Random Structures & Algorithms*, 51(3):538–561, 2017. doi: 10.1002/rsa.20713.

Jing Lei and Alessandro Rinaldo. Consistency of spectral clustering in stochastic block models. *The Annals of Statistics*, 43(1):215–237, 2015. doi: 10.1214/14-AOS1274.

Tianxi Li, Elizaveta Levina, and Ji Zhu. Prediction models for network-linked data, 2018. URL https://arxiv.org/abs/1602.01192.

Jean-Laurent Mallet. Discrete smooth interpolation. *ACM Transactions on Graphics*, 8(2): 121–144, April 1989. ISSN 1557-7368. doi: 10.1145/62054.62057. URL http://dx.doi.org/10.1145/62054.62057.

Roberto Imbuzeiro Oliveira. Concentration of the adjacency matrix and of the laplacian in random graphs with independent edges, 2009. URL https://arxiv.org/abs/0911.0600.

Mathew D. Penrose. *Random geometric graphs*, volume 5. OUP Oxford, 2003.

Bryan Perozzi, Rami Al-Rfou, and Steven Skiena. Deepwalk: online learning of social representations. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '14. ACM, 2014.

Paul Rosa and Judith Rousseau. Nonparametric regression on random geometric graphs sampled from submanifolds, 2024. URL https://arxiv.org/abs/2405.20909.

Cosma Shalizi and Dena Asta. Consistency of maximum likelihood for continuous-space network models I. *Electronic Journal of Statistics*, 18(1):335 – 354, 2024. doi: 10.1214/23-EJS2169. URL https://doi.org/10.1214/23-EJS2169.

Zhaoyang Shi, Krishnakumar Balasubramanian, and Wolfgang Polonik. Adaptive and non-adaptive minimax rates for weighted laplacian-eigenmap based nonparametric regression, 2023. URL https://arxiv.org/abs/2311.00140.

Alexander J. Smola and Risi Kondor. Kernels and regularization on graphs. In Bernhard Schölkopf and Manfred K. Warmuth, editors, *Learning Theory and Kernel Machines*, pages 144–158, Berlin, Heidelberg, 2003. Springer Berlin Heidelberg. ISBN 978-3-540-45167-9.

Tom Snijders and Krzysztof Nowicki. Estimation and prediction for stochastic blockmodels for graphs with latent block structure. *Journal of Classification*, 14:75–100, 1997.

Zixing Song, Xiangli Yang, Zenglin Xu, and Irwin King. Graph-based semi-supervised learning: A comprehensive review, 2021. URL https://arxiv.org/abs/2102.13303.

Charles J. Stone. Optimal global rates of convergence for nonparametric regression. *The Annals of Statistics*, 10(4):1040–1053, 1982. ISSN 00905364, 21688966. URL http://www.jstor.org/stable/2240707.

Minh Tang, Daniel L. Sussman, and Carey E. Priebe. Universally consistent vertex classification for latent positions graphs. *The Annals of Statistics*, 41(3):1406–1430, 2013. doi: 10.1214/13-AOS1112.

Warren S. Torgerson. Multidimensional scaling: I. theory and method. *Psychometrika*, 17: 401–419, 1952. URL https://api.semanticscholar.org/CorpusID:120849755.

Alexandre B. Tsybakov. *Introduction to Nonparametric Estimation*. Springer Publishing Company, Incorporated, 1st edition, 2008. ISBN 0387790519.

Yu-Xiang Wang, James Sharpnack, Alex Smola, and Ryan J. Tibshirani. Trend filtering on graphs, 2016. URL https://arxiv.org/abs/1410.7690.

Larry A. Wasserman. *All of Nonparametric Statistics*. Springer Texts in Statistics. Springer, 2005. ISBN 978-0-387-25145-5.

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

Yunpeng Zhao, Elizaveta Levina, and Ji Zhu. Consistency of community detection in networks under degree-corrected stochastic block models. *The Annals of Statistics*, 40 (4):2266 – 2292, 2012.