arXiv:2202.11817v1 [stat.ML] 23 Feb 2022

Benefit of Interpolation in Nearest Neighbor Algorithms

Yue Xing, Qifan Song, and Guang Cheng[y]

Abstract. In some studies [e.g., 43] of deep learning, it is observed that over-parametrized deep neural networks achieve a small testing error even when the training error is almost zero. Despite numerous works towards understanding this so-called \double descent" phenomenon [e.g., 8, 10], in this paper, we turn into another way to enforce zero training error (without over-parametrization) through a data interpolation mechanism. Specifically, we consider a class of interpolated weighting schemes in the nearest neighbors (NN) algorithms. By carefully characterizing the multiplicative constant in the statistical risk, we reveal a U-shaped performance curve for the level of data interpolation in both classification and regression setups. This sharpens the existing result [11] that zero training error does not necessarily jeopardize predictive performances and claims a counter-intuitive result that a mild degree of data interpolation actually strictly improve the prediction performance and statistical stability over those of the (un-interpolated) k-NN algorithm. In the end, the universality of our results, such as change of distance measure and corrupted testing data, will also be discussed.

Key words. Interpolation, Nearest Neighbors Algorithm, Regret Analysis, Double descent, Regression, Classification

1. Introduction.

Statistical learning algorithms play a central role in modern data analysis and artificial intelligence. A supervised learning algorithm aims at constructing a predictor h, which uses training samples to predict testing data. Given a loss function and the class of candidate predictors H, the function h 2 H is usually selected by minimizing the empirical loss. To avoid over-fitting, classical learning theory [1, 7] suggests controlling the capacity of model space H (e.g., VC-dimension, fat-shattering dimension, Rademacher complexity), and discourages data interpolation. On the other hand, recent deep learning applications reveal a completely different phenomenon from classical understanding, that is, with heavily over-parameterized neural network models, when the training loss reaches zero, the testing performance is still sound. For example, [25, 32, 43] demonstrated experiments where deep neural networks have a small generalization error even when the training data are perfectly fitted and gave discussions towards this phenomenon. This counter-intuitive phenomenon motivates various theoretical studies to investigate the testing performance of estimators belonging to the \over-fitting regime." For instance, [2, 3, 5, 14, 29, 39] established generalization bounds for shallow neural networks with a growing number of hidden nodes; [6, 10, 15, 16, 24] provided the theoretical characterization of the interpolated estimators in some particular models, e.g. linear regression and binary classification. Some later studies also characterize how the learning curve changes in different models [18], or its universality beyond natural training [28]. All these aforementioned results deliver proper explanations for the phenomenon that \(proper) over-fitting does not hurt prediction" under parametric modelings.

Inspired by these studies, this paper aims to sharpen the existing results on the relation-ship between over-fitting and testing performance in a nonparametric estimation procedure

Department of Statistics, Purdue University

[y] Department of Statistics, University of California, Los Angeles

and provide insights into the phenomenon, \over-fitting does not hurt prediction", from a different perspective. To be more specific, we study the interpolated nearest neighbors algorithm (interpolated-NN, [9]), which is a nonparametric regression/classification estimator that interpolates the training data. [9, 11] have proved the rate optimality of interpolated-NN regression estimation. Beyond rate optimality claim, so far, there is no insight about the behavior of nearest neighbor estimations within the overfitting regimes, such as how different interpolation schemes affect the characteristics of interpolated-NN.

We conduct a comprehensive analysis to quantify the risk of interpolated-NN estimators, including its convergence rate and, more importantly, the associated multiplicative constant. The interpolated-NN estimator, h, is indexed by a parameter , which directly represents the level of interpolation. As increases, the value of h(x) is more influenced by data point value $Y^1(x)$ in the sense that $\lim_{!1} h(x) = Y^1(x)$, where $X^1(x)$ denotes the nearest neighbor of x and $Y^1(x)$ is the response of $X^1(x)$. As the interpolation level increases within a proper range, the rate of convergence of the squared bias term and the variance term in the variance-bias decomposition are not affected. Under proper smoothness conditions, as increases, the multiplicative constants associated with the squared bias and the variance terms decrease and increase, respectively. More importantly, when is small, the decrease of squared bias dominates the growth of variance; hence mild level interpolation strictly improves interpolated-NN compared with non-interpolated k-NN estimator. Overall, the risk of interpolated-NN, as a function of interpolation level , is U-shaped. Therefore, within the over-fitting regime, the risk of interpolated NN will first decrease and then increase concerning the interpolation level. A graphic illustration can be found in Figure 1 below with data dimension d = 2. Besides, we conduct a similar analysis to study the statistical stability [35] of interpolated-NN classifier and obtain similar results where the stability of interpolated-NN is U-shaped in the interpolation level as well. Our major finding, which in spirit claims that increasing degree of overfitting potentially benefit testing performance, is somewhat similar to the recently discovered \double-descent phenomenon" [8]. A graphical illustration of the \double-descent phenomenon" is shown in Figure 2. In the first regime of the double-descent phenomenon, the \classical regime", the model complexity is below some \interpolation threshold" and the population loss is U-shaped when the model complexity increases. If the model complexity

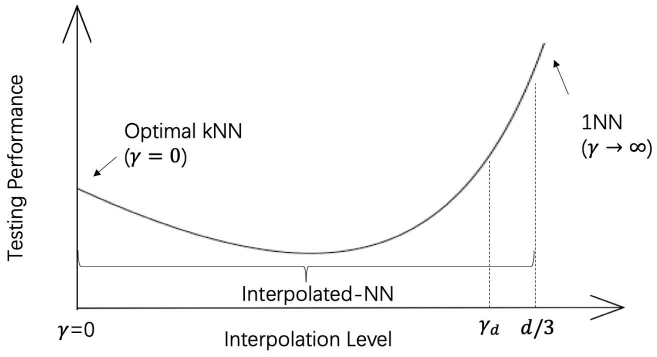

Figure 1. Relationship between testing performance and level of interpolation in interpolated-NN.

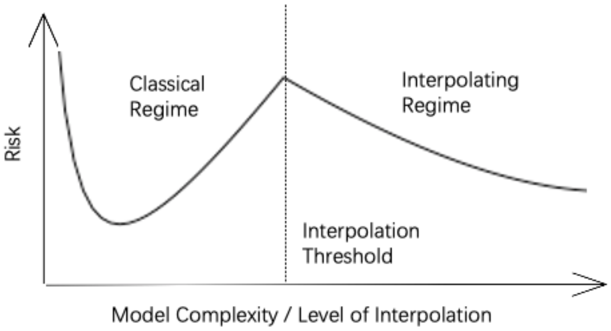

Figure 2. Double-descent Phenomenon in modern machine learning theory

keeps increasing, in the second regime, \overfitting regime", the training loss will be exactly zero, and the generalization performance will get better in the model complexity. In both the double-descent phenomenon and our result, the estimator gets improved with a proper level of interpolation. However, besides the fact that our work focuses on nonparametric models while the double-descent phenomenon results are mostly on parametric models, it is noteworthy to emphasize the crucial differences between this double descent phenomenon and our result. The literature of the double-descent phenomenon studies the change of interpolated estimators' performance for the level of over-parameterization, while our work focuses on the level of interpolation. Although both are related to model over-fitting, they are not equivalent. The level of over-parameterization refers to the complexity of model space H. In contrast, the level of interpolation is merely a measure of the estimating procedure (i.e., how much $h(x)$ is affected by the nearest neighbor of x) and has nothing to do with the complexity or dimensionality of H. Hence, in the double-descent literature, one compares the risk of estimator $h()$ across different dimensional setups, while in our work, the data dimension is fixed. It is worth noting that some recent works (e.g., 26) discover the double-descent phenomenon for nonparametric regression as well (e.g., kernel ridge regression), which also considers that the dimension of model space increases.

Besides the main contributions, there are some minor contributions. First, we notice that the convergence rate given in [9] for classification tasks is sub-optimal, and we improve it to optimal rate convergence via technical improvements. Second, we study interpolated-NN under some other scenarios, e.g., change on distance metric and corruptions in testing data attributes. To be more specific, (1) We consider how the choice of distance metric used affects the asymptotic behavior of interpolated-NN. Traditional NN estimation usually utilizes L_2 distance to determine the neighborhood set, and we will investigate whether or not the effect of interpolation is universal among different choices of distance metric. (2) Since interpolation-NN leads to a rather rugged and non-smooth regression estimation function, i.e., the nonparametric estimation function $h(x)$ will have a big jump, especially around the training samples (refer to Figure 9 in Section A), we will also investigate in whether interpolation affects the prediction performance when testing data is slightly perturbed.

Finally, we want to emphasize that this paper does not aim to promote the practical use of this interpolation method, given that k-NN is more user-friendly. Instead, our study on the interpolated-NN algorithm is used to describe the role of interpolation in generalization ability precisely, which is a new and interesting phenomenon motivated by the deep neural network practices.

The structure of the paper is as follows: in Section 2, we introduce interpolated-NN and the level of interpolation in this method. We further present theorems regarding to rate optimality and multiplicative constant in Section 3 and 4 respectively. A measure of statistical instability is also considered in Section 3 and 4. In Section 4, in addition to the main theorems, some discussions relating to double descent, or more specifically, about the interpolation regime, will be given in Section 4.6, followed by numerical experiments in Section 5 and conclusion in Section 6.

2. Interpolation in Nearest Neighbors Algorithm.

In this section, we first review the notations and the detailed algorithm of interpolated-NN. Denote $(X_i; Y_i)$ as training samples for $i = 1; :::; n$. Given a data point x for testing, define $R_{k+1}(x)$ to be the distance between x and its $(k + 1)$th nearest neighbor. Without loss of generality, let $X^1(x); : : : ; X^k(x)$ de-note the unsorted k nearest neighbors of x, and write as $X^1; : : : ; X^k$ for simplicity. Denote $fR_i(x)g^k_{i=1}$ and $fY^i(x)g^k_{i=1}$ ($fR_ig^k_{i=1}$ and $fY^ig^k_{i=1}$ in short) as distances between x and X^i and the corresponding responses.

For regression models, the aim is to estimate $E(Y jX = x)$. Denote (x) as the target function, where $(x) = E(Y jX = x)$, and ${}^2(x) = V ar(Y jX = x)$. A NN regression estimator at x, based on nearest k neighbors, is defined as

$$b_{k;n}(x) = \overset{k}{\underset{i=1}{X}} W_i(x)Y^i; \tag{2.1}$$

where $W_i(x)$ (write as W_i later) represents the (data-dependent) weight for each neighbor. The weighting scheme satisfies $W_i\ \ 0$ and $P^k_{i=1} W_i = 1$. For traditional k-NN, $W_i = 1=k$ for all $i = 1; :::; k$.

For binary classification, denote $(x) = P(Y = 1jX = x)$, with $g(x) = 1f(x) > 1=2g$ as the Bayes classifier. The interpolated-NN classifier is further defined as

$$bg_{k;n}(x) = \Bigg(\begin{matrix} 1 & P^k_{i=1} W_iY^i > 1=2 \\ 0 & P^k_{i=1} W_iY^i\ \ 1=2 \end{matrix} \quad :$$

The asymptotics of traditional k-NN algorithm have been extensively studied in the literature (e.g., 17, 20, 22, 33, 35, 37, 40). Variants of k-NN have been proposed to improve the convergence, stability or computational performance, e.g.,optimally weighted NN [31], locally weighted NN [13], stabilized NN [35], distributed NN [21], pre-processed 1NN [41], and robust NN [38].

An interpolating algorithm will interpolate training data, i.e., the fitted value/label is the same as the response/label of the training data. From this aspect, traditional k-NN does not interpolate unless $k = 1$. To incorporate interpolation with NN algorithm, we follow [9] to adopt the interpolated weighting scheme below:

$$(2.2) \qquad W_i = \frac{R_i}{P\ R_j} = \frac{(R_i=R_{k+1})}{P_{j=1}(R_j=R_{k+1})} := \frac{(R_i=R_{k+1})}{P_{j=1}(R_j=R_{k+1})}\ {}_{j=1};$$

for $i = 1;:::;k$ and some 0. Rewrite $b_{k;n}$ and $b_{k;n}$ as $b_{k;n;}$ and $b_{k;n;}$ to emphasize the choice of . In interpolated-NN, the function is chosen as $(t) = t$. In general, if the positive function satisfies $\lim_{t!0} (t) = 1$ and decreases in t, through replacing the in W_i, we can create different interpolating weighting schemes.

The parameter 0 controls the level of interpolation: with a larger > 0, the al-gorithm will put more weights on the closer neighbors, especially the nearest neighbor. In particular, when $= 0$, the interpolated-NN reduces to the common k-NN, and when $= 1$, interpolated-NN reduces to 1-NN. [9] showed that given any fixed $2\ R^+$, the interpolated estimator using (2.2) is minimax rate-optimal for mean squared error in regression, but only provided a suboptimal upper bound for the Regret of binary classification tasks.

To evaluate the predictive performance of a NN algorithm, we adopt the conventional measures as follows: assume the random testing data X and its corresponding response variable Y follow the same distribution as the training data, then

$$\text{Regression:}\quad MSE(k;n;) = E((b_{k;n;}(X) \qquad (X))^2);$$
$$\text{Classification:}\quad Regret(k;n;) = P\ (b_{k;n;}(X) = Y) \qquad P\ (g(X) = Y);$$

where E and P are the expectation and probability measure with respect to the joint distribution of training data and testing data.

For regression, MSE is adopted to evaluate the predictive accuracy of mean responses. For classification, the Regret measures the difference between the testing accuracies of the estimated classifier and the oracle Bayes classifier, i.e., the excessive mis-classification rate. The Regret can be equivalently rewritten as

$$(2.3) \qquad Regret(k;n;) = E\,[2j(X) \qquad 1{=}2jP\,(b_{k;n;}(X) = Y)];$$

that is, the Regret can be viewed as a weighted mis-classification rate, where less weight is assigned on the region closer to decision boundary $fx : (x) = 1{=}2g$.

3. Rate Optimality of Interpolated-NN.

In this section, the convergence rate results for interpolated-NN are provided. We show that both MSE (of regression task) and Regret (of classification task) converge at the optimal rate. Besides, the statistical instability (defined by [35]) of interpolated-NN is evaluated, revealing that interpolated-NN is as stable (in terms of asymptotic rate) as k-NN.

3.1. Model Setup.

The theorems in this section are developed under assumptions as those in [17] and [9]. We state these assumptions as follows:

A.1 X is a d-dimensional random variable on a compact set and satisfies the following regularity condition [4]: there exists positive $(c_0; r_0)$ such that for any x in the support X ,

$$(X \setminus B(x;r)) \quad c_0(B(x;r));$$

for any $0 < r \quad r_0$, where denotes the Lebesgue measure on R^d.

A.2 The density of X is between $[m_x; M_x]$ for some constants $0 < m_x \quad M_x < 1$.

A.3 Smoothness condition: $j(x) \qquad (y)j \quad Akx \qquad yk$ for some > 0.

A.4 For classication, the model satises Tsybakov margin condition [36]: $P(j(X) \quad 1=2j < t)$ Bt

A.5 For regression, the variance of noise is nite, i.e., $^2(x) \quad M < 1$ for any $x \; 2 \; X$.

Assumption A.1 regulates the shape of X and avoids spiky support, it essentially ensures that for any $x \; 2 \; X$, all its k nearest neighbors are suciently close to x with high probability. If X is compact and convex, this regularity condition is automatically satised. Assumption A.2 regularizes the neighborhood set for any testing sample $x \; 2 \; X$: the upper bound of the density prevents neighbors from clustering at x, i.e., R_i's converge to 0 too fast; the lower bound of the density prevents the neighbors from being too far from x. Assumption A.1 and A.2 together guarantee that $R_i \quad (k=n)^{1=d}$ in probability. If this lower bound assumption is violated and the density of x goes to zero, classical k-NN may not perform well, and a locally-weighted NN [13] allows the weight assignment to depend on x, is preferred. As shown in [13], locally-weighted NN improves the empirical performance of k-NN. Readers of interest can design interpolated locally-weighted NN algorithm and study its asymptotic behavior as those in our later theorems. The constants $c_0; r_0; A; B; ;$ are distribution-specic (i.e., can change w.r.t. the distribution of $(X; Y)$ or the dimension d) but are independent to n. Among them, only and are the two key values that aect the convergence rate (w.r.t. n) of the NN estimator.

For regression tasks, assumption A.1-A.3 and A.5 are commonly used in the literature (e.g., 9), under which one can prove that the NN type regression estimators achieve the minimax rate $O(n \quad ^{2=(2+d)})$, via the technical tools of [36]. For assumption A.3, a smaller value of , i.e., a less smooth true , implies that estimating is more dicult. Assumption A.5 is imposed to restrict the variation of y given x. Assumption A.5 only requires that the variance at each x is nite, which is a fairly weak condition.

For classication tasks, the smoothness condition A.3 and margin condition A.4 usually appear in the Regret analysis of k-NN [e.g., 9, 17, 35]. These two conditions together describe how large the probability measure is near the decision boundary $fxj(x) = 1=2g$. Under A.1 to A.4, it is well known that the optimal rate for Regret of nonparametric classication is $O(n \; ^{(+1)=(2+d)})$ due to [36]. Note that our theorem allows that = 0, i.e., it applies to k-NN as well.

Besides the detailed analysis in the following sections, we also provide a concrete example with an intuitive explanation of why interpolation does not hurt the convergence of interpolated-NN in Section A.

3.2. Rate of Convergence. In the following theorem, we present the convergence rate of interpolated-NN for both regression and classication under a mild level of data interpolation, i.e., is within some suitable range. This theorem is a rened result of [9], which only obtains the optimal convergence rate of MSE for regression tasks. Note that our rate for MSE is the same as in [9]. We include both regression and classications results in the following theorem for the sake of completeness.

Theorem 3.1. Assume $d \quad 3 \quad C > 0$ for some constant $C > 0$ and $\quad 0$. For regression, under A.1-A.3 and A.5,

$$\mathrm{MSE}(;n) := \min_k \mathrm{MSE}(k;;n) = O(n^{\ 2=(2+d)}):$$

For classification, under A.1-A.3 and A.4 if < 2,

$$\mathrm{Regret}(;n) := \min_k \mathrm{Regret}(k;;n) = O(n^{\ (+1)=(2+d)}):$$

In addition, denote $()$ as the smallest even number that is greater than $+ 1$. If 2, when d $() > 0$,

$$\mathrm{Regret}(;n) = O(n^{\ (+1)=(2+d)}):$$

In below we discuss the critical steps of proving Theorem 3.1. For regression, MSE at x can be decomposed into

$$E(b_{k;n;}(x) \quad (x))^2 \quad A^2 E \left(\sum_{i=1}^{k} W_i \|X^i \quad x\| \right)^2 + E \sum_{i=1}^{k} W^2 (Y_i(X^i) \quad (X^i))^2 \ ;$$

and our convergence rate of MSE is thus based on careful derivations of upper bounds for EW_i, EW_i^2, and $E\|X^i \quad x\|$. Although interpolation introduces obvious biased prediction at the training data points by forcing $b(X_i) = Y_i$, after taking expectation over both training and testing data, such a bias effect is averaged out. For classification, we note that $(X^i; Y^i)$'s for $i = 1; :::; k$ are i.i.d. samples within the ball $B(x; r)$ conditional on $R^{k+1} = r$. Consequently, we can apply (non-uniform) Berry-Esseen Theorem to approximate $P(b_{k;n;}(x) \neq g(x))$ by normal probability, i.e.,

$$(3.1) \qquad P(b_{k;n;}(x) < 1=2 | g(x) = 1) \qquad \left(\frac{1=2 \quad E(b_{k;n;}(x))}{\sqrt{Var(b_{k;n;}(x))}} \right) \ ;$$

which is related to the mean and variance of $b_{k;n;}(x)$. Similar to the regression case, the pointwise mis-classification rate is barely affected by data interpolation when $x = X_i$, thus the Regret of interpolated-NN has the same rate as the traditional k-NN. For different regions (< 2 and 2), different restrictions on are imposed for technical simplicity. This restriction is used to control the reminder term that appears in the non-uniform Berry-Esseen Theorem. A detailed proof for the Regret convergence part of Theorem 3.1 is postponed to the supplementary material. For the MSE convergence part, we refer readers to [9, 11].

Our technical contributions of Theorem 3.1 (and Theorem 4.1 later) lie in the adaptations of the existing analysis framework of [17, 31].

(a) Compared to Theorem 4.5 in [9] that is derived based on the Chebyshev's inequality, the Berry-Esseen Theorem applied in Theorem 3.1 leads to a tighter Regret bound.

(b) By the definition of W_i, the weighted samples W_iY_i are no longer independent with each other, so some transformations are made to enable the use of concentration inequalities in the approximation (3.1).

The insight of Theorem 3.1 is that under proper a interpolation scheme and reasonable overfitting level, data interpolation does not hurt the rate optimality of the NN algorithm for both regression and classification tasks.

3.3. Statistical Stability. In this section, we explore how interpolation affects the statistical stability of NN algorithms in classification. For a stable classification method, it is expected that with high probability, the classifier can yield the same prediction label when being trained by different data sets sampled from the same population. Therefore, [35] introduced a type of statistical stability, named as classification instability (CIS), to quantify how stable the classifier is. Denote D_1 and D_2 as two i.i.d. training sets of the same sample size n. Then CIS for interpolated-NN is defined as

$$\mathrm{CIS}_{k;n}() = P_{D_1;D_2;X}\left(\hat{g}_{k;n;}(X;D_1) = \hat{g}_{k;n;}(X;D_2)\right);$$

where $\hat{g}_{k;n;}(x;D_j)$ is the predicted label of $\hat{g}_{k;n;}$ at x when the training data set is D_j for $j = 1;2$. Therefore, the CIS can be viewed as a counterpart of the variance measure of nonpara-metric regression estimator: $E_X Var(\hat{b}_{k;n;}(X)) = \frac{1}{2}E_{D_1;D_2;X}(\hat{b}_{k;n;}(X;D_1) \quad \hat{b}_{k;n;}(X;D_2))^2$.

From the formula of CIS, a larger value of CIS indicates that the classifier is less statistically stable. Both misclassification rate and classification instability should be taken into account when evaluating the merits of any classification algorithm. Thus, our theory aims to characterize the CIS of interpolated-NN under the choice of k that attains its optimal classification performance (i.e., the Regret). Besides, it is noteworthy that CIS is different from the algorithmic stability in the literature [12, 19, 23], where the two data sets D_1 and D_2 are identical except for one sample, rather than independent identically distributed.

The following theorem studies the CIS of interpolated-NN. In short, we show that the CIS for interpolated-NN and k-NN converge under the same rate:

Theorem 3.2. Under A.1, A.2, A.3 and A.4, if satisfies the conditions stated in Theorem 3.1, when taking $k = cn^{1=(2+d)}$ for any constant $c > 0$, we have

$$\mathrm{CIS}_{k;n}() = O\, n^{\;=(2+d)}:$$

Note that $k \quad n^{1=(2+d)}$ is the choice of k, which attains the optimal rate of Regret for both k-NN and interpolated-NN.

Based on Theorem 3.2, the rate of CIS of interpolated-NN is the same as traditional k-NN (i.e., taking $= 0$) when we either (i) select the best k value for k-NN and apply it to both k-NN and interpolated-NN, or (ii) select the optimal k for k-NN and interpolated-NN respectively. Also, this rate of CIS matches the minimax lower bound described below. Let P be the marginal distribution of X, and P denotes the joint distribution of $(X;y)$ determined by P and conditional mean function , then the following proposition holds:

Proposition 3.3 (Theorem 4 in [35]). Let $P_{;}$ be the set of probability distributions that contains all distributions of form P, where P and satisfy A.1, A.2, A.3 and A.4. If $< d$, then there exists some constant $C > 0$ such that for any estimator $\hat{}$,

$$\sup_{2P_{;}} \mathrm{CIS}(\hat{}) \quad Cn^{\;=(2+d)}:_P$$

4. Quantification of Interpolation Effect. Our results presented in Section 3 show that interpolated-NN retains the rate minimaxity in terms of MSE, Regret, and statistical stability.

To further reveal the subtle relationship between data interpolation and estimation performance, we sharply quantify the multiplicative constants associated with the convergence rates of MSE, Regret, and statistical instability of interpolated-NN algorithm.

4.1. Model Assumptions. To facilitate our theoretical investigation, a slightly dierent set of assumptions are imposed:

A.1' X is a d-dimensional random variable on a compact R^d manifold X with boundary @X.

A.2' The density of X is in $[m_x; M_x]$ for some $0 < m_x \;\; M_x < 1$, and twice dierentiable.

A.3' For classication, the set S = fxj(x) = 1=2g is non-empty. There exists an open subset U_0 in R^d which contains S such that, for an open set containing X (dened as U), is continuous on $U n U_0$.

A.4' For classication, there exists some constant $c_x > 0$ such that when j(x) 1=2j c_x, has bounded fourth-order derivative; when (x) = 1=2, the gradient _(x) = 0 when (x) = 1=2, and with restriction on x 2 @X, @(x) =— 0 if (x) = 1=2, where @(x) denotes the restriction of to @X. Also the derivative of (x) within restriction on the boundary of support is non-zero.

A.5' For regression, the second-order derivative of is smooth for all x.

A.6' For regression, $\sup_{x2X} {}^2(x) \;\; M < 1$ and ${}^2(x)$ is twice-continuously dierentiable.

The smoothness condition A.4' describes how smooth the function is, which aects the minimax performance of k-NN given a class of 's. c_x is a distribution-specic value. The existence of nonzero c_x and the high-order smoothness condition allows us to rigorously analyze the near decision boundary region where most of the mispredictions are made (Step 2 in the proof of Theorem 4.1 in Appendix E.2). Note that the validity of our results doesn't depend on the magnitude of c_x. Assumptions A.3' and A.4' describe how far the samples are away from fxj(x) = 1=2g. The assumptions are mostly derived from the framework established by [31]. Note that the additional smoothness required in and f is needed to facilitate the asymptotic study of the interpolated weighting scheme. We also want to point out that these assumptions are generally stronger than A.1 to A.5, but they allow us to obtain a sharper bound. Using A.1 to A.5, one can only obtain some upper bound rate for the approximation error in (3.1), while using A.1' to A.6' one can obtain the exact value of the approximation error in (3.1) (except for high order remainder terms).

Remark 1. There is a heuristic relationship between smoothness conditions A.1-A.4 and A.1'-A.4'. First of all, Assumption A.1' to A.4' also imply the marginal condition A.4 with = 1. In addition, as mentioned by [17], A.2 in fact implies

$$j(x) \quad (B(x; r))j \;\; Lr; \tag{4.1}$$

while [31] showed the approximation E(X) (x_0) + tr[•(x_0)E(X x_0)(X $x_0)^>$] under A.2' and A.4'. Therefore, by matching result (4.1) and the above approximation, we can view A.1'-A.4' as a special case of conditions A.1-A.4 under smoothness parameter = 2 and = 1. Furthermore, for classication, the minimax rate under conditions A.1-A.4 is $O(n^{\;(+1)=(2+d)})$, which becomes $O(n^{\;4=(4+d)})$ under (;) = (2; 1). It coincides with con-vergence rates result [31] under condition set A.1'-A.4'.

4.2. Main Theorem. The following theorem examines the asymptotics of MSE and Regret of interpolated-NN and k-NN under Assumptions A.1'-A.6'. We rst dene some useful quantities. Let P_1 and P_2 (and f_1, f_2) be the conditional distributions (and densities) of X given $Y = 0; 1$ respectively, and $_{1;2}$ be the marginal probability $P(Y = 0)$ and $P(Y = 1)$. Denote $P = {}_1P_2 + {}_2P_2$, $P = {}_1P_1 \quad {}_2P_2$, $f(x) = {}_1f_1(x) + {}_2f_2(x)$, and also denote $(x) = {}_1f_1(x) \quad {}_2f_2(x)$. Dene

$$a(x) = \frac{1}{f(x)^{1+2=d}d}\left\{\sum_{j=1}^{d}[\_{j}(x)f_{j}(x) + \bullet_{jj}(x)f(x)=2]\right\}:$$

Then the following decomposition of MSE/Regret holds:

Theorem 4.1. Assume $d \quad 3 \quad C > 0$ for some constant C and $\quad 0$. For regression, suppose that assumptions A.1', A.2', A.5', and A.6' hold. If k satises $n \quad k \quad n^{1 \quad 4=d}$ for some > 0, then

$$\text{(4.2)} \qquad \text{MSE}(k; n;) = \underbrace{\frac{(d \quad)^{2}}{d(d \quad 2)}\frac{1}{k}E[\ (X)^2]}_{\text{Variance}}$$

$$\text{(4.3)} \qquad + \underbrace{\frac{d(d \quad)^{2}}{(\ + 2 \quad)^{2}}\frac{(d \ +2)^2}{d^{2}}E\left[a^{2}(X)\right]E^{2}(R_{1}^{2}|X)}_{\text{Bias}} + \text{Remainder};$$

where Remainder = o(MSE(k; n;)).

For classication, under A.1' to A.4', the excess risk w.r.t. becomes

$$\text{(4.4)} \quad \text{Regret}(k; n;) = \underbrace{\frac{(d \quad)^{2}}{d(d \quad 2)}\frac{1}{4k}\int_{S}\frac{f(x_0)}{k\_(x_0)k}d\text{Vol}^{d \quad 1}(x_0)}_{\text{Variance}}$$

$$\text{(4.5)} \qquad + \underbrace{\frac{(d \quad)^{2}}{(d + 2 \quad)^{2}}\frac{(d \ +2)^2}{d^{2}}\int_{S}\frac{f(x_0)a^{2}(x_0)}{k\_(x_0)k}E^{2}(R_{1}^{2}|X = x_0)d\text{Vol}^{d \quad 1}(x_0)}_{\text{Bias}}$$

$$+\text{Remainder};$$

where Remainder = o(Regret(k; n;)).

The proof of Theorem 4.1 is postponed to Section E.3 of the supplementary material. The basic ideas are the same as Theorem 3.1. Since the stronger conditions of A.1' to A.6' enables a detailed Taylor expansion on MSE/Regret, the multiplicative constants associated with variance and bias can be gured out.

Similar as in [31], when taking $k = n^{4=(d+4)}$, $E(R_1^2|X) = O(n^{\quad 2=(d+4)})$, thus it is easy to see that the MSE and Regret of interpolated-NN are of $O(n^{\quad 4=(d+4)})$, which is the same rate of k-NN, optimal weighted NN [31], stabilized-NN [35], and distributed-NN [21].

Theorem 4.1 provides an exact variance-bias decomposition for MSE and Regret, except for a negligible remainder term, which enables us to quantify the eect of interpolation carefully.

Given fixed (k; n) satisfying the conditions in Theorem 4.1, the coefficient in the variance terms (4.2) and (4.4) are increasing functions in , i.e.,

$$\frac{1}{d}\frac{(d\quad)^2}{(d\quad 2)} = \frac{1}{k}\;1+\frac{2}{d(d\quad 2)}\;k\;\frac{1}{1+}\;d^2\;\frac{2}{;}\;{}^{k}$$

which behaves like a quadratic function around = 0; the coefficient in the bias terms in (4.3) and (4.5) are decreasing functions in , i.e.,

$$\frac{(d\quad)^2}{(d+2\quad)^2}\;\frac{(d+2)^2}{d^2} = 1\;\frac{2}{d^2+2d\quad d}\;{}^{2}\;1\,\frac{4}{d^2+2d}$$

which behaves like a linear decreasing function around = 0. Therefore, it is clear that tuning the interpolation level leads to a trade-off between the bias and variance of interpolated-NN estimation: interpolated-NN will introduce a larger variance and reduce the bias when increasing . In addition, given a small value of , since a quadratic increase of variance ($O(^2)$) is always dominated by a linear decrease of bias ($O()$), it is possible for interpolated-NN to achieve a better performance than k-NN.

The above analysis can be extended to a general class of weight functions, and the same variance-bias trade-off occurs, in the sense that a weighting scheme allocating more weights on closer neighbors tends to have a smaller bias and larger variance. The detailed analysis can be found in Section I of the supplementary material.

4.3. U-shaped Asymptotic Performance of Interpolated-NN. In this section, we aim to refine our results stated in Theorem 4.1, and characterize the asymptotic performance of interpolated-NN with respect to the interpolation level . Two choices of k are considered. In first choice, k is -independent and only relies on the value of n. To be more specific, we choose k = arg min MSE(k; n; = 0) (or k = arg min Regret(k; n; = 0) for classification) which is the optimal choose for k-NN algorithm. This same k value is used regardless of the interpolation level, and we study how the performance of interpolation-NN changes as increases. In the second choice, we allow k to depend on interpolation level as well (hence is denoted by k), such that k = arg min MSE(k; n;) (or k = arg min Regret(k; n;) for classification). In other words, the k value is optimally tuned with respect to the interpolation level.

Under the first choice of k, we quantify the asymptotic performance of interpolated-NN using k-NN as the benchmark reference. Define

$$PR(d;\,) := \frac{1}{+d=4}\;\frac{(d\quad)^2}{d(d\quad 2)} + 4\,\frac{d\,(d\quad)^2(d+2)^2\;1}{d^2(d+2\quad)^2}\;:$$

Then the following result holds:

Corollary 4.2. Under conditions in Theorem 4.1, for regression, when k is chosen to minimize the MSE of k-NN, then for any 2 [0; d=3), the performance ratio satisfies,

$$\frac{MSE(k;n;\,)}{MSE(k;n;0)}\;!\;PR(d;\,):$$

For classication, when k is chosen to minimize the Regret of k-NN, then for any 2 [0; d=3), the performance ratio becomes

$$\frac{\text{Regret}(k; n;)}{\text{Regret}(k; n; 0)} \;!\; PR(d;):$$

When k is chosen based on k-NN, the interpolation aects regression and classication in exactly the same manner through PR(d;). In particular, this ratio exhibits an interesting U-shape curve of for any xed d: as increases from 0, PR(d;) rst decreases and then increases. Therefore, within the range $(0; _d)$ for some threshold value $_d$ which depends on dimension d only, the asymptotic performance ratio PR(d;) < 1. It is noteworthy that although PR(d;), as a function of , is U-shaped on 2 R, our Corollary 4.2 is restricted to that < d=3. Thus $_d$ is the minimum value between =3 and the second root of function PR(d;) = 1. When PR(d;) < 1, the interpolated-NN is strictly better than the k-NN. Moreover, the eect of interpolation is asymptotically invariant for any distributions of (X; y) as long as the assumptions hold, and PR(d;) ! 1 as the dimension d grows to innity.

In the second choice, we let the interpolated-NN utilize the optimal k with respect to . From Theorem 4.1, k $k_0(:= k_{=0})$ $n^{\frac{4}{d+4}}$, but $k=k_0$ > 1 for > 0, i.e., interpolated-NN needs to employ slightly more neighbors than k-NN to achieve the best performance.

Corollay 4.3 below asymptotically compares interpolated-NN and traditional k-NN in terms of the above measures. It turns out that the performance ratio (the ratio of two MSE's or two Regret's), asymptotic converges to

$$PR^0(d;) := 1 + \frac{2}{d(d \quad 2)}^{d+4\;\frac{4}{}} \;(d + \frac{2\;(d \quad)}{)^2}\;\frac{2\;(d \quad 2)^2\;d+4}{+\;d^2}\;\frac{d}{}\;;$$

which is a function of d and only, and is independent of the underlying data distribution.

Corollary 4.3. Under conditions in Theorem 4.1, for any 2 [0; d=3),

$$\frac{\text{MSE}(n;)}{\text{MSE}(n; 0)} \;!\; PR^0(d;); \qquad \text{and} \qquad \frac{\text{Regret}(n;)}{\text{Regret}(n; 0)} \;!\; PR^0(d;); \qquad \text{as } n \;!\; 1:$$

Note that MSE(n; 0)/Regret(n; 0) is the optimum MSE/Regret for k-NN.

Denote ^0_d as the threshold such that $PR^0(d;) < 1$ for any < ^0_d. From Corollary 4.3, starting from 1 when = 0, the performance ratio $PR^0(d;)$ will rst decrease in then increase; see Figure 1 in introduction. Some further calculations can show that ^0_d < d=3 when d 3; ^0_d = d=3 when d 4. It can also be gured out that whether interpolated-NN is better or not does not depends on the distribution of (X; y).

In addition, comparing the results in Corollary 4.2 and 4.3, since interpolated-NN selects the k to minimize its Regret in $PR^0(d;)$, the region of where $PR^0(d;)$ is smaller than 1 is wider than that for PR(d;) as in Corollary 4.2, i.e., $^0_d > _d$.

Remark 2. It is easy to show that the limiting values of P R and PR^0 converge to 1 in d, i.e., $\lim_{d!1}[\min_{<d=3} PR(d;)] = 1$ and $\lim_{d!1}[\min_{<d=3} PR^0(d;)] = 1$. This indicates that high dimensional model benets less from interpolation, or said dierently, high dimensional model is less aected by data interpolation. This phenomenon can be explained by the fact that, as d increases, $R_i=R_{k+1}$! 1 for i = 1; :::; k due to high dimensional geometry.

On the other hand, [31] worked out a general form of Regret and MSE for a weighted NN estimator and thereafter proposed the optimally weighted nearest neighbors algorithm (OWNN) by minimizing Regret for classication with respect to the values of weight W_i's. We extend their results to regression and compare them with interpolated-NN.

Combining Theorem 4.3 with [31], we have

$$\frac{R(n;\mathrm{OWNN})}{R(n;)} \;!\; 2^{\frac{4}{d+4}}\,\frac{d+2}{d+4}^{\frac{2d+4}{d+4}}\;\; 1+\frac{2}{d(d\quad 2)}^{\frac{4}{d+4}}\;\; \frac{(d\quad)^2}{(d+2\quad)^2}\,\frac{(d+2)^2}{d^2}^{\frac{d}{d+4}} \;;$$

which is always smaller than 1. Here $R(n;\mathrm{OWNN})$ denotes the MSE/Regret of OWNN given its optimum k, and $R(n;)$ denotes the one of interpolated-NN given its own optimum k choice. Again, the above ratio reects the dierence in convergence for OWNN and interpolated-NN, at the multiplicative constant level. This ratio converges to 1 as d diverges (just as the case of $PR(d;)$; see Remark 2). Thus, under ultra high dimensional setting, the performance dierences among k-NN, interpolated-NN, and OWNN are almost negligible even at the multiplicative constant level when $n \; ! \;\; 1$.

4.4. Statistical Stability. Similar to MSE and Regret, we perform a precise quantication analysis for the convergence of CIS: Theorem 4.4 below illustrates how CIS is aected by interpolation. In short, if interpolated-NN and k-NN algorithms share the same value of k, then interpolated-NN is less stable than k-NN; otherwise, the interpolated-NN will be more stable for $\; 2\; (0;{}^0_d)$ if k is allowed to be chosen optimally based on .

The following theorem quanties CIS for general choices of $(k;)$:

Theorem 4.4. Under the conditions in Theorem 4.1, the CIS of interpolated-NN is derived as

$$\mathrm{CIS}_{k;n}() = (1+o(1))\,p\frac{B\,1}{\frac{1}{}\,p\quad k}\; s\overline{\frac{(d\quad)^{2}}{d(d\quad 2)}}:$$

The proof of Theorem 4.4 is postponed to Section F in the supplementary material. Compared with Theorem 4.1, one can gure out that CIS is related to the variance term in total Regret, but not related to the bias term. This is because the two data sets D_1 and D_2 are drawn independently from the same distribution, thus the estimators share the same bias, while how dierent the predictions depends on the variance.

In Section 3.3, we show that under either (i) using the optimum k's for these two methods respectively; or (ii) using the same k (the optimum for k-NN) for both k-NN and interpolated-NN, CIS for both k-NN and interpolated-NN converges in the same rate. Based on Theorem 4.4, we can further compare the multiplicative constants as in Corollary 4.5 below:

Corollary 4.5. Following the conditions in Theorem 4.4, when the same k value is used for k-NN and interpolated-NN (as long as the k satises conditions in Theorem 4.1), then as $n \; ! \;\; 1$,

$$\frac{\mathrm{CIS}_{k;n}()}{\mathrm{CIS}_{k;n}(0)} \;!\;\; s\overline{\frac{(d\quad)^2}{d(d\quad 2)}}\;\; 1:$$

On the other hand, if we choose optimum k's for k-NN and interpolated-NN respectively, i.e., k = arg min $_k$Regret(k; n;), when n ! 1 , we have $_2$

$$\frac{\mathrm{CIS}_{k\ ;n}()}{\mathrm{CIS}_{k_0;n}(0)} \quad ! \quad \mathrm{PR}^0(d;):$$

Therefore, when 2 (0;^{0}),$_d$interpolated-NN with optimal k has higher accuracy and stability than k-NN at the same time.

From Corollary 4.5, the interpolated-NN is not as stable as k-NN if both algorithms use the same number of neighbors. However, this is not the case if an optimal k is tuned w.r.t . An intuitive explanation is that under the same k, k-NN has a smaller variance (more stable) given equal weights for all k neighbors. On the other hand, by choosing an optimum k, interpolated-NN can achieve a much smaller bias, which osets its performance lost in the variance through enlarging k.

4.5. Eect of Corrupted Testing Data. When applying machine learning algorithms in practice for classication, the testing data may be corrupted, and its distribution diers from the training data. For example, the testing data may be randomly perturbed due to the inaccuracy of data collection (e.g., an image has noise due to an optical sensor system malfunction). In some other scenarios, the testing data may be deliberately modied to induce the learning algorithm to make a wrong decision (e.g., spam email sender will try to hack the email lter system). From the denition of interpolated-NN, the prediction is forced to jump to a label if the testing sample is suciently closed to some training data, and the estimated regression function () could be rather rugged (refer to Figure 9). These obser-vations potentially imply a certain degree of adversarial instability in the sense that a small change of x may lead to a huge change of b $_{;n;}$(x). This$_k$ motivates us to investigate the performance of interpolated-NN further when encountering random perturbed/adversarial at-tacked input testing samples. In short, when the corruption is independent with training data, i.e., random perturbation or black-box attack, a small positive improves the performance. When the corruption is data-dependent, i.e., white-box attack, interpolated-NN is vulnerable. Some existing literature focused on design attack / adversarial robust nearest neighbors-type algorithms, e.g., [34, 38, 40], and they worked on un-interpolated NN algorithms.

Denote ! as the corruption level of testing data, i.e., instead of observing the testing data x, we observe another value $\tilde{x}$ which is inside an L_2 ball B(x; !) centering at x with radius !. Three types of corruptions are considered in this paper: (1) random perturbation: $\tilde{x}_{rand}$ is randomly drawn from the ball B(x; !); (2) black-box attack: the adversary has no information on our training data and the trained model, hence it trains a dierent machine e() using an independent dataset. The adversarial attack thereafter is designed as $\tilde{x}_{black}$ = arg max$_{z2B(x;!)}$ e(z) if (x) < 1=2 and $\tilde{x}_{black}$ = arg min$_{z2B(x;!)}$ e(z) if (x) > 1=2; (3) white-box attack: the adversary has full access to the trained model b $_{;k;}$, and designed white-box attack as $\tilde{x}_{white}$ = arg max$_{z2B(x;!)}$ b $_{;k;}$(z) if (x) < 1=2 and $\tilde{x}_{white}$ = arg min$_{z2B(x;!)}$ b $_{;k;}$(z) if (x) > 1=2. $_n$ $_n$ $_n$

We rst display the sucient condition when the testing Regret still converges in the rate of n $^{4=(d+4)}$.

Corollary 4.6 (Informal description). Under conditions in Theorem 4.1, taking k n$^{4=(d+4)}$, for corruption scheme $\tilde{x}_{rand}$ and $\tilde{x}_{black}$, when ! = O(n $^{2=(d+4)}$), the Regret P(b $_{k;n;}$(X) =

Y) P (g(X) = Y) = $O(n^{4=(d+4)})$. When ! = $o(n^{2=(d+4)})$, for corruption scheme ✗$_{\text{rand}}$ and ✗$_{\text{black}}$, the Regret ratio (Regret of interpolated-NN over Regret of k-NN) is the same as the one for un-corrupted data. When $!^3$ = $o(n^{4=(d+3)})$, the regret decreases when slightly increases from zero.

For corruption scheme ✗$_{\text{white}}$, when ! = $O(n^{2=(d+4)})$ ^ $o(n^{1=d})$, the Regret P ($b_{k;n;}$(X) = Y) P (g(X) = Y) = $O(n^{4=(d+4)})$ for any satisfying conditions in Theorem 4.1. When ! = $o(n^{2=(d+4)})$ ^ $o(n^{1=d})$, the Regret ratio (Regret of interpolated-NN over Regret of k-NN) is the same as the one for un-corrupted data.

For the formal representations for random perturbation and black-box attack, we postpone them to Section H of the supplementary material.

Corollary 4.6 reveals that when the corruption level is small enough, we can still benet from interpolation since the performance ratio between interpolated-NN and k-NN is unchanged. However, interpolated-NN is more vulnerable to the white-box attack.

On the opposite, when ! is not suciently small, the testing data corruptions lead to a sub-optimal convergence rate, especially under white-box attacks. The reason is that unless 2 f0; 1g, there are always (innitely many as n increases) training samples, denoted as D^0, whose labels are dierent from the Bayes classier. Then for any testing sample x, such that B (x; !) overlaps with D^0, the white-box attack can be designed as ✗$_{\text{white}}$ 2 B (x; !) \ D^0, and will yield a wrong prediction, as the prediction of interpolated-NN always interpolate D^0.

For k-NN, such a problem will not happen since the predictor always takes an average among k training samples. Based on [9], the misclassied labels are dense in interpolated-NN. However, for (1) random perturbation and (2) black-box attack, the phenomenon is dierent:

Corollary 4.7. Under the conditions of Theorem 4.1, for white-box attack ✗$_{\text{white}}$, when !=($n^{1=d}$) ! 1 , there exists some constant c > 0 (depending on f and) such that the Regret is asymptotically greater than c. For random perturbation ✗$_{\text{rand}}$ and black-box attack ✗$_{\text{black}}$, when !=($n^{1=d}$) ! 1 , the Regret is in O($!^2$).

4.6. Bless of Interpolation and Eect of Distance Metric. As shown by the results in the previous section, data interpolation leads to a more accurate and stable performance than traditional k-NN when k = k and 2 (0; ^{0}). Similarly, in the literature of double descent, [e.g., 10, 24] use a linear regression model to demonstrate a U-shaped curve (w.r.t. data dimension d) of MSE in the over-tting regime (i.e., the number of linear covariates is larger than the number of observations).

To compare interpolated-NN and linear regression, although we observe a U-shaped performance ratio curve for both interpolated-NN estimator and ridgeless estimator in high dimensional linear regression when over-tting occurs, they benet from interpolation in dierent ways. For the ridgeless estimator in the regression problem, an increasing over-parametrization level causes a larger bias, but the coecients all tend to zero. Therefore, following the analysis in [10] and [24], the variance is very small and leads to a descent of MSE in the over-tting regime. This descent in MSE in this interpolation regime is the second descent of MSE in the double descent phenomenon.

On the other hand, for interpolated-NN, the benet of increasing the level of interpolation is the reduction of bias. As a result, although the double descent phenomenon is observed in many different estimation procedures [e.g., 8], a detailed study is needed to comprehensively understand how each machine learning technique enjoys the benefit of interpolation.

Besides, it is worth mentioning that the benefit of interpolation in interpolated-NN is not affected by the choice of distance metric. In the previous discussions, the L_2 norm is used to determine the neighborhood set and measure the distance R_i. If we replace it by a different norm, e.g., L_1 or L_{inf}, it turns out that the effect of interpolation is the same as under L_2 measure:

Corollary 4.8. Assume the optimum k's for k-NN and interpolated-NN are chosen, respectively. Under assumptions in Corollary 4.3, if L_p ($p \quad 1$) distance is used to calculate the distances among all data points, as well as for deciding interpolation weighting scheme, then the performance ratio between interpolated-NN and k-NN shares the same shape as in Corol-lary 4.3, and CIS ratio shares the same shape as in Corollary 4.4.

5. Numerical Experiments.

In this section, several simulation studies are presented to justify our theoretical discoveries for regression, classification, and stability of the interpolated-NN algorithm, together with some real data analysis.

5.1. Simulations.

In Section 4.3, two scenarios of the performance of interpolated-NN are presented: (1) interpolated-NN utilizes the same k as k-NN; (2) k is chosen optimally for each . In Section 5.1.1, the experiment setups are described in details and the numerical results are presented for scenario (1). The numerical results for scenario (2) are postponed to Section 5.1.2.

5.1.1. k is Chosen Optimally for $= 0$.

We aim to estimate the performance ratio curve via numerical simulations and compare it with the theoretical curve $PR(d;)$ (in Corol-lary 4.2). We take training sample size $n = 2048$ and use 5000 testing samples to evaluate MSE/Regret/CIS for all simulations. This procedure is repeated 500 times to obtain the mean and standard deviation of the performance ratios. As observed in [31], the empirical performance ratio is not stable, with 0:02 difference to the theoretical value when $n = 1000$. Since our goal of this simulation is to empirically validate the performance ratios rather than promoting an interpolated-NN based method, instead of tuning k via cross-validation over training data, we select k which minimizes the MSE/Regret over independently simulated large testing data set.

For regression simulation, each attribute of X follows i.i.d. uniform distribution on $[\quad 1; 1]$ with $d = 2$ or 5, and $Y = (x) + "$ with $" \quad N(0; 1)$ where

$$(x) = \frac{e^{x^> w}}{e^{x^> w} + e^{\quad x^> w}}; \qquad (5.1)$$

$$w_i = i \quad d{=}2 \quad 0{:}5 \qquad i = 1; :::; d:$$

A sequence of levels of interpolation $=d = 0; 0{:}05; 0{:}1; : : : ; 0{:}35$ are used to evaluate the performance of interpolated-NN.

The results are summarized in Figure 3. The trends for theoretical value and simulation value are close. The small difference is mostly caused by the small order terms in the asymp-totic result and shall vanish if larger n is used. Note that $=d = 0{:}35$ is outside our theoretical range $=d < 1{=}3$, but the empirical performance is still reasonable.

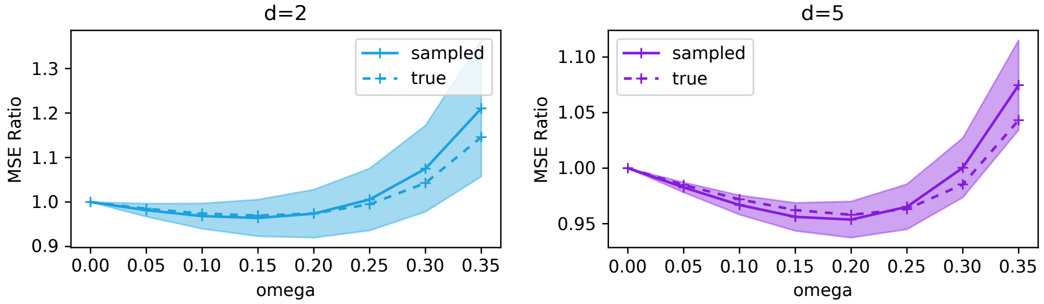

Figure 3. Performance Ratio in Regression when k is Chosen Optimally for k-NN

For classification, two models are generated. For the first model, each dimension of P_1, the marginal distribution of X given $Y = 0$, follows independent standard Cauchy. The first $\lfloor d/2 \rfloor$ dimensions of P_2, the marginal distribution of X given $Y = 1$, follows independent standard Cauchy, and the remaining dimensions of P_2 follow independent standard Laplace distribution. Through the design of the first model, the two classes have the same center at the origin. In the second model, each dimension of P_1 follows independent $N(0;1)/2 + N(3;4)/2$, and each dimension of P_2 follows independent $N(1.5;1)/2 + N(4.5;4)/2$. For both P_1 and P_2 in the second model, the distributions are multi-modal.

The CIS of interpolated-NN classifier is estimated by calculating the proportion of testing samples that have different prediction labels, that is

$$\widehat{\mathrm{CIS}}() = \frac{1}{n}\sum_{i=1}^{n} 1(\hat{g}_{n,k}(x_i; D_1) = \hat{g}_{n,k}(x_i; D_2));$$

where D_1, D_2 are two independent training data sets, and x_i's are independently sampled testing data points.

Similar to Figure 3, the classification results are summarized in Figures 4 and 5. Similar to the regression setup, the theoretical value and empirical value trends are closed for the Regret ratio. Compared with the Regret ratio, the standard deviation of the CIS ratio is much larger due to the randomness of the two training sample sets; thus, its trend is slightly away from the theoretical value.

5.1.2. k is Chosen Optimally for each . For all the three models in Section 5.1.1, the performance ratios are also calculated when interpolated-NN uses the k value, which optimizes its testing performance. The formula for the performance ratio in this scenario can be found in Corollary 4.3. The results are shown in Figures 6, 7 and 8. Similar to the results in Section 5.1.1, the empirical curves are mostly closed to the theoretical curves.

5.2. Real Data Analysis. In the real data experiments, we compare the classification accuracy of interpolated-NN with k-NN when k is optimally tuned (via cross-validation) with respect to values.

Five data sets are considered in this experiment. The data set HTRU2 from [27] uses 17,897 samples with 8 continuous attributes to classify pulsar candidates. The data set Abalone

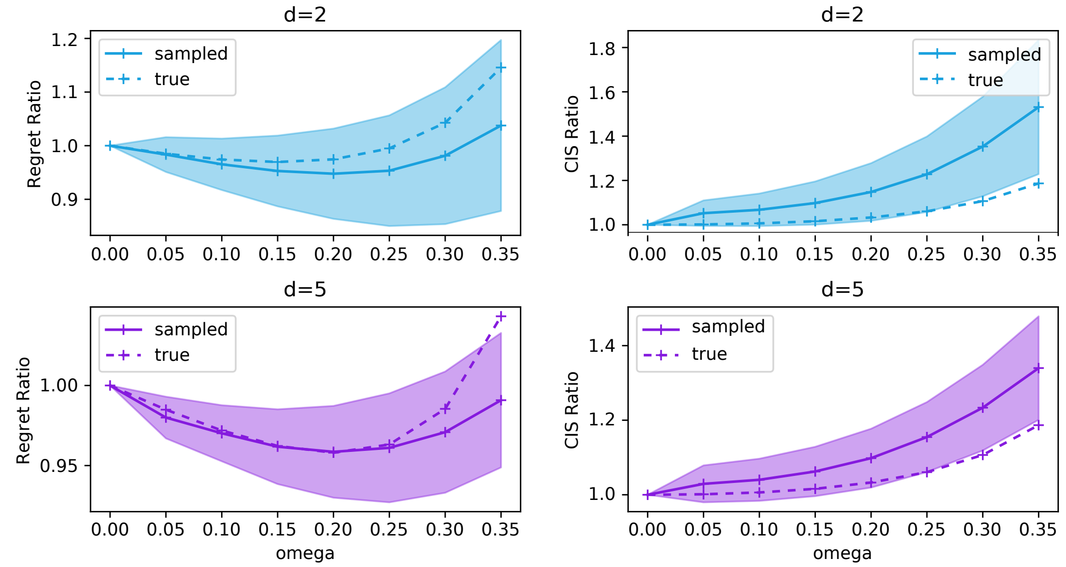

Figure 4. Performance Ratio in Classification, Model 1, when k is Chosen Optimally for k-NN

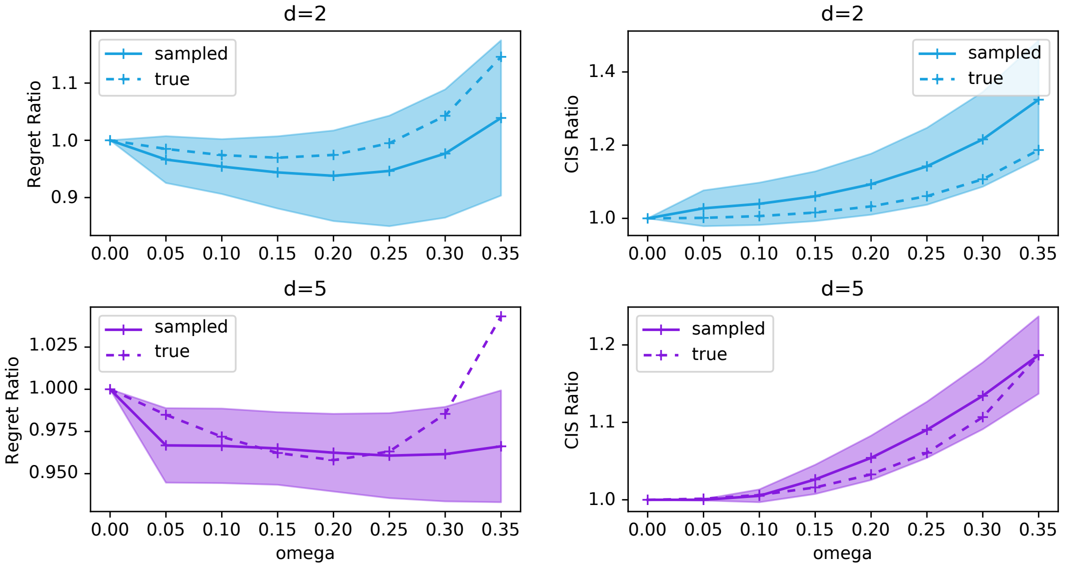

Figure 5. Performance Ratio in Classification, Model 2, when k is Chosen Optimally for k-NN

contains 4,176 samples with seven attributes. Following [38], we predict whether the number of rings is greater than 10. The data set Credit [42] has 30,000 samples with 23 attributes and predicts whether the payment will be default in the next month given the current payment

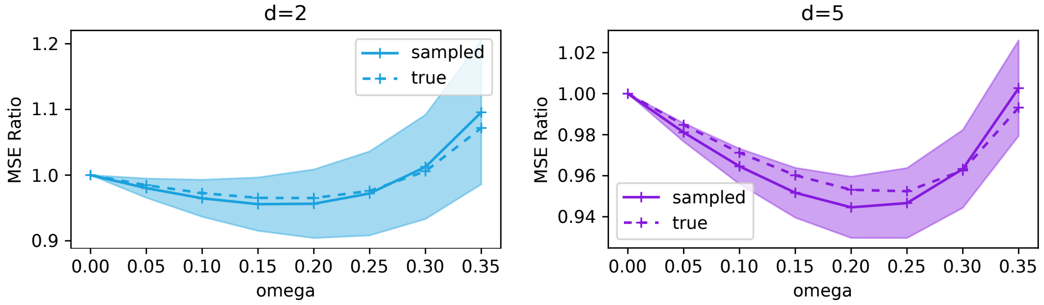

Figure 6. Performance Ratio in Regression when k is Chosen Optimally for each

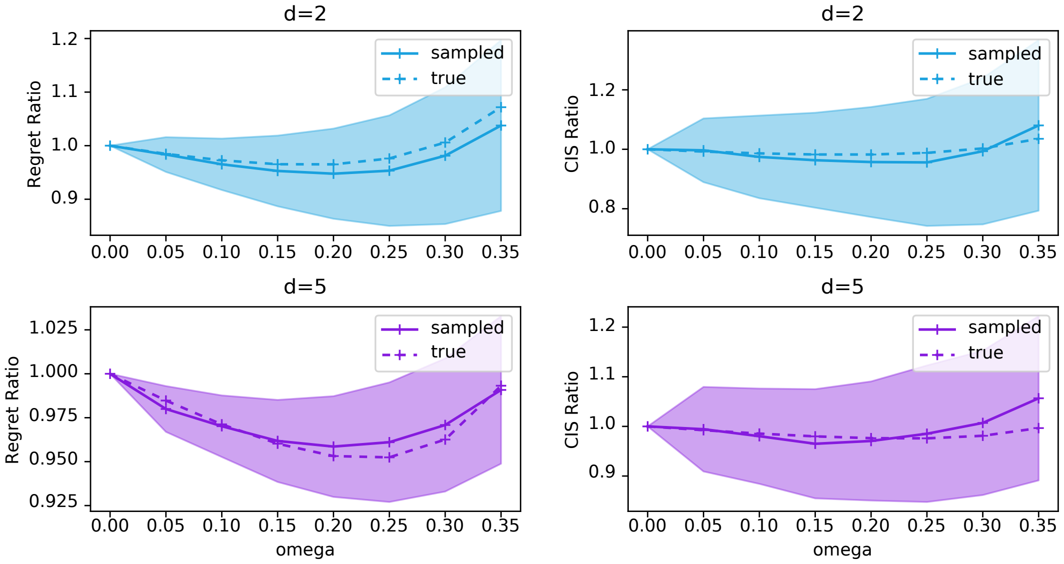

Figure 7. Performance Ratio in Classification, Model 1, when k is Chosen Optimally for each

information. The built-in data set of digits in sklearn [30] contains 1,797 samples of 88 images. For images in MNIST are 26 26, we will use part of it in our experiment. Both the data set of digit and MNIST have ten classes. Here for binary classification, we group 0 to 4 as the first and 5 to 9 as the second class.

For each data set, a proportion of data is used for training, and the rest is reserved for testing. Note that the same testing data set is also used to tune the optimal choice of k. For Abalone, HTRU2, Credit, and Digit, we use 25% data as training data and 75% as testing data. For MNIST, we randomly choose 2000 samples as training data and 1000 as testing data, which is sufficient for our comparison. The above experiment repeats 50 times, and the average testing error rate is summarized in Table 1.

For all data sets, the best testing error among interpolated-NN's with different choices of

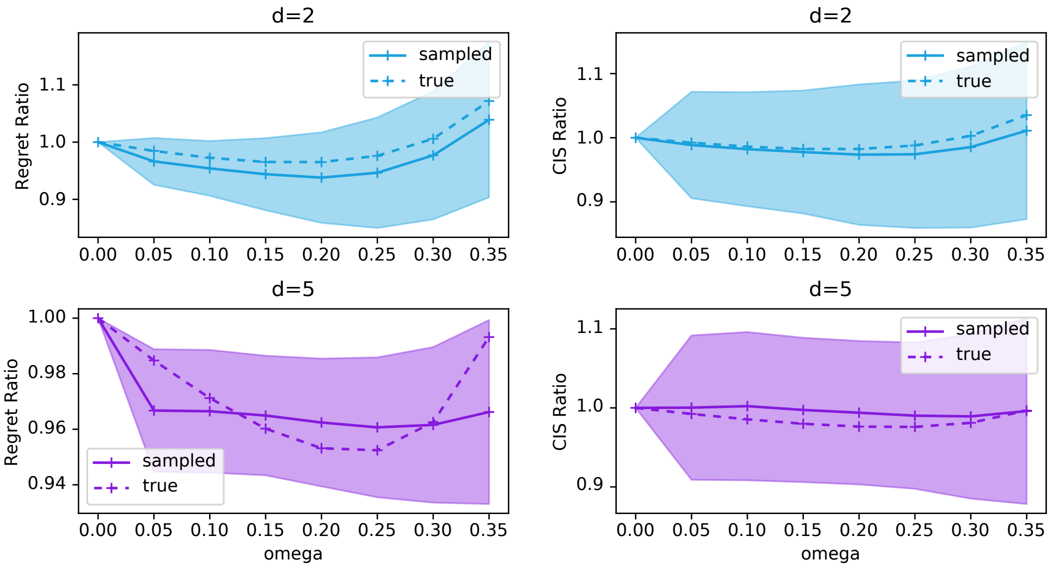

Figure 8. Performance Ratio in Classification, Model 2, when k is Chosen Optimally for each

Data	d	Error (= 0)	Error (best =d)	best =d
Abalone	7	0.22239	0.22007	0.3
HTRU2	8	0.02315	0.0226	0.2
Credit	23	0.1933	0.19287	0.05
Digit	64	0.01745	0.01543	0.25
MNIST	784	0.04966	0.04656	0.05

Table 1

Prediction Error of k-NN, interpolated-NN under the best choice of , together with the value of the best for interpolated-NN.

> 0 (column \best =d") is always smaller than the k-NN(column \ = 0"), which verifies that the nearest neighbor algorithm can benefit from mild-level interpolation.

6. Conclusion. Our work precisely quantifies how data interpolation affects the performance of the nearest neighbor algorithms beyond the rate of convergence. For both regression and classification problems, the asymptotic performance ratios between interpolated-NN and k-NN converge to the same value, independent of data distribution, and depend on d and only. More importantly, when the interpolation level =d is within a reasonable range, the interpolated-NN is strictly better than k-N, attaining both faster convergence and better stability performance.

Classical learning frameworks oppose data interpolation as it believes that over-fitting means fitting the random noise rather than the model structures. However, in the interpolated-NN, the weight degenerating occurs only on a nearly zero-measure set, and thus there is only \local over-fitting", which may not hurt the overall rate of convergence. Technically, through balancing the variance and bias, data interpolation potentially improves the overall performance. Essentially, our work precisely quantifies such a bias-variance balance. It is of great interest to investigate how our theoretical insights can be carried over to the real deep neural networks, leading to a complete picture of the double descent phenomenon.

Finally, as mentioned in this paper, different algorithms (regression, nearest neighbors algorithm) may enjoy the benefit of interpolation in different ways; thus, an in-depth explo-ration on each algorithm is necessary to obtain a comprehensive understanding of the double descent phenomenon. Nonetheless, although the mechanisms behind these results are dif-ferent, all these tell us that a carefully designed statistical method can overcome harmful over-fitting.

References.

[1] M. Anthony and P. Bartlett, Function learning from interpolation, in European Conference on Computational Learning Theory, Springer, 1995, pp. 211{221.

[2] S. Arora, S. S. Du, W. Hu, Z. Li, and R. Wang, Fine-grained analysis of optimization and generalization for overparameterized two-layer neural networks, in Proceedings of the 36th International Conference on Machine Learning, vol. 97 of Proceedings of Machine Learning Research, PMLR, 2019, pp. 322{332.

[3] S. Arora, R. Ge, B. Neyshabur, and Y. Zhang, Stronger generalization bounds for deep nets via a compression approach, arXiv preprint arXiv:1802.05296, (2018).

[4] J.-Y. Audibert and A. B. Tsybakov, Fast learning rates for plug-in classifiers, The Annals of statistics, 35 (2007), pp. 608{633.

[5] P. L. Bartlett, D. J. Foster, and M. J. Telgarsky, Spectrally-normalized margin bounds for neural networks, in Advances in Neural Information Processing Systems, 2017, pp. 6240{6249.

[6] P. L. Bartlett, P. M. Long, G. Lugosi, and A. Tsigler, Benign overfitting in lin-ear regression, Proceedings of the National Academy of Sciences, 117 (2020), pp. 30063{30070.

[7] P. L. Bartlett, P. M. Long, and R. C. Williamson, Fat-shattering and the learnability of real-valued functions, Journal of Computer and System Sciences, 52 (1996), pp. 434{452.

[8] M. Belkin, D. Hsu, S. Ma, and S. Mandal, Reconciling modern machine learning and the bias-variance trade-off, Proceedings of the National Academy of Sciences, 116 (2019), pp. 15849{15854.

[9] M. Belkin, D. Hsu, and P. Mitra, Overfitting or perfect fitting? risk bounds for classification and regression rules that interpolate, Advances in Neural Information Processing Systems 31, (2018), pp. 2300{2311.

[10] M. Belkin, D. Hsu, and J. Xu, Two models of double descent for weak features, arXiv preprint arXiv:1903.07571, (2019).

[11] M. Belkin, A. Rakhlin, and A. B. Tsybakov, Does data interpolation contradict statistical optimality?, in Proceedings of Machine Learning Research, vol. 89 of Proceedings of Machine Learning Research, PMLR, 16{18 Apr 2019, pp. 1611{1619.

[12] O. Bousquet and A. Elisseeff, Stability and generalization, Journal of machine learning research, 2 (2002), pp. 499{526.

[13] T. I. Cannings, T. B. Berrett, and R. J. Samworth, Local nearest neighbour classification with applications to semi-supervised learning, arXiv preprint arXiv:1704.00642, (2017).

[14] Y. Cao and Q. Gu, Generalization bounds of stochastic gradient descent for wide and deep neural networks, in Advances in Neural Information Processing Systems, 2019, pp. 10836{10846.

[15] Y. Cao, Q. Gu, and M. Belkin, Risk bounds for over-parameterized maximum margin classification on sub-gaussian mixtures, arXiv preprint arXiv:2104.13628, (2021).

[16] N. S. Chatterji and P. M. Long, Finite-sample analysis of interpolating linear classifiers in the overparameterized regime, Journal of Machine Learning Research, 22 (2021), pp. 1{30.

[17] K. Chaudhuri and S. Dasgupta, Rates of convergence for nearest neighbor classification, in Advances in Neural Information Processing Systems, 2014, pp. 3437{3445.

[18] L. Chen, Y. Min, M. Belkin, and A. Karbasi, Multiple descent: Design your own generalization curve, arXiv preprint arXiv:2008.01036, (2020).

[19] Y. Chen, C. Jin, and B. Yu, Stability and convergence trade-off of iterative optimization algorithms, arXiv preprint arXiv:1804.01619, (2018).

[20] T. M. Cover, Rates of convergence for nearest neighbor procedures, in Proceedings of the Hawaii International Conference on Systems Sciences, 1968, pp. 413{415.

[21] J. Duan, X. Qiao, and G. Cheng, Distributed nearest neighbor classification, arXiv preprint arXiv:1812.05005, (2018).

[22] J. Fritz, Distribution-free exponential error bound for nearest neighbor pattern classification, IEEE Transactions on Information Theory, 21 (1975), pp. 552{557.

[23] M. Hardt, B. Recht, and Y. Singer, Train faster, generalize better: Stability of stochastic gradient descent, in International Conference on Machine Learning, 2016, pp. 1225{1234.

[24] T. Hastie, A. Montanari, S. Rosset, and R. J. Tibshirani, Surprises in high-dimensional ridgeless least squares interpolation, arXiv preprint arXiv:1903.08560, (2019).

[25] Z. Li, Z.-H. Zhou, and A. Gretton, Towards an understanding of benign overfitting in neural networks, arXiv preprint arXiv:2106.03212, (2021).

[26] F. Liu, Z. Liao, and J. A. Suykens, Kernel regression in high dimension: Refined analysis beyond double descent, arXiv preprint arXiv:2010.02681, (2020).

[27] R. J. Lyon, B. Stappers, S. Cooper, J. Brooke, and J. Knowles, Fifty years of pulsar candidate selection: from simple filters to a new principled real-time classification approach, Monthly Notices of the Royal Astronomical Society, 459 (2016), pp. 1104{1123.

[28] Y. Min, L. Chen, and A. Karbasi, The curious case of adversarially robust models: More data can help, double descend, or hurt generalization, arXiv preprint arXiv:2002.11080, (2020).

[29] B. Neyshabur, S. Bhojanapalli, and N. Srebro, A pac-bayesian approach to spectrally-normalized margin bounds for neural networks, arXiv preprint arXiv:1707.09564, (2017).

[30] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Pas-

sos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, Scikit-learn: Machine learning in Python, Journal of Machine Learning Research, 12 (2011), pp. 2825{2830.

[31] R. J. Samworth, Optimal weighted nearest neighbour classifiers, The Annals of Statistics, 40 (2012), pp. 2733{2763.

[32] A. Sanyal, P. K. Dokania, V. Kanade, and P. Torr, How benign is benign overfitting?, in International Conference on Learning Representations, 2020.

[33] J. Schoenmakers, J. Zhang, and J. Huang, Optimal dual martingales, their analysis, and application to new algorithms for bermudan products, SIAM Journal on Financial Mathematics, 4 (2013), pp. 86{116.

[34] C. Sitawarin and D. Wagner, Minimum-norm adversarial examples on knn and knn-based models, arXiv preprint arXiv:2003.06559, (2020).

[35] W. W. Sun, X. Qiao, and G. Cheng, Stabilized nearest neighbor classifier and its statistical properties, Journal of the American Statistical Association, 111 (2016), pp. 1254{1265.

[36] A. B. Tsybakov, Optimal aggregation of classifiers in statistical learning, The Annals of Statistics, 32 (2004), pp. 135{166.

[37] T. Wagner, Convergence of the nearest neighbor rule, IEEE Transactions on Information Theory, 17 (1971), pp. 566{571.

[38] Y. Wang, S. Jha, and K. Chaudhuri, Analyzing the robustness of nearest neighbors to adversarial examples, in Proceedings of the 35th International Conference on Machine Learning, vol. 80 of Proceedings of Machine Learning Research, PMLR, 10{15 Jul 2018, pp. 5133{5142.

[39] B. Xie, Y. Liang, and L. Song, Diverse neural network learns true target functions, in Proceedings of the 20th International Conference on Artificial Intelligence and Statistics, vol. 54 of Proceedings of Machine Learning Research, PMLR, 20{22 Apr 2017, pp. 1216{1224.

[40] Y. Xing, Q. Song, and G. Cheng, Predictive power of nearest neighbors algorithm under random perturbation, in International Conference on Artificial Intelligence and Statistics, PMLR, 2021, pp. 496{504.

[41] L. Xue and S. Kpotufe, Achieving the time of 1-nn, but the accuracy of k-nn, in International Conference on Artificial Intelligence and Statistics, AISTATS 2018, 9-11 April 2018, Playa Blanca, Lanzarote, Canary Islands, Spain, vol. 84 of Proceedings of Machine Learning Research, PMLR, 2018, pp. 1628{1636.

[42] I.-C. Yeh and C.-h. Lien, The comparisons of data mining techniques for the predictive accuracy of probability of default of credit card clients, Expert Systems with Applications, 36 (2009), pp. 2473{2480.

[43] C. Zhang, S. Bengio, M. Hardt, B. Recht, and O. Vinyals, Understanding deep learning requires rethinking generalization, in 5th International Conference on Learning Representations, OpenReview.net, 2017.

The supplementary material is organized as follows. In Section A, we demonstrate a con-crete example with an intuitive explanation of why interpolation does not hurt the convergence of interpolated-NN. Section B-H provide the proofs for the main theorems in the manuscript. Section I extend the class of function interpolating weighting scheme to general class of functions which enjoys the benet of variance-bias trade-o.

Appendix A. Variance-Bias Trade-o in NN Algorithms . In this section, we present an intuitive comparison between the interpolated-NN and traditional k-NN. For any weighted-NN regressor dened in (2.1), if the smooth condition $j(x_1)\quad (x_2)j\ \ Akx_1\quad x_2k_2$ holds for some and A, then we have the following bias-variance decomposition (9):

(A.1)

$$E\ \ (b_{k;n}(x)\quad (x))^2\ = Ef(E(b_{;k n}(x))\quad (x))^2g + Ef(b_{;k n}(x)\quad E(b_{k;n}(x)))^2g;\quad \text{where}$$

$$Ef(E(b_{k;n}(x))\quad (x))^2g = \text{bias}^2\ \ A^2 E \sum_{i=1}^{k} W_i kX^i\quad xk^{2};$$

$$Ef(b_{k;n}(x)\quad E(b_{k;n}(x)))^2g = \text{variance} = \sum_{i=1}^{k} E\quad W_i^2(Y^i\quad (X^i))^2\ :$$

Several insights are developed based on the decomposition in (A.1). First, if $\mathrm{Var}(Y\,jX)$ (or $E[(Y^i\quad (X^i))^2\ j\,X^i])$ is invariant among X (e.g., the regression setting with i.i.d. noise $_i$, or the classication setting with constant function), then k-NN ($W_i\ \ 1{=}k$) represents the optimal weight choice which minimizes the variance term. Second, if the weight assignment prioritizes closer neighbors, i.e., larger W_i for smaller $kX^i\quad xk$, it will lead to a smaller value for the weighted average $\sum_{i=1}^{k} W_i kX^i\quad xk$. In other words, interpolated-NN can achieve smaller upper bound for the bias term. Therefore, we argue that k-NN and interpolated-NN employ dierent strategies in reducing the upper bound of MSE. The former emphasizes reducing the variance, while the latter emphasizes reducing the bias.

The above intuitive arguments are well-validated by the following toy examples.

We take 30 training samples $x_i\ =\quad 5;\ \ 4;::::;25$ with responses generated by the three choices: (1) $y =\ 0\ x +\ $, (2) $y =\ x^2 + 0\ $, and (3) $y =\ (x\quad 10)^2{=}8 + 5$ where $N(0;1)$. In other words, the mean function (x) are (1) $(x)\quad 0$, (2) $(x)\ =\ x^2$, and (3) $(x)\ =\ (x\quad 10)^2{=}8$ respectively. The number of neighbors k is chosen to be 10. Three dierent weighting schemes are considered: (1) $(t)\ =\ 1{=}k$, (2) $(t)\ =\ 1\quad \log(t)$, and (3) $(t) =\ t\ \ ^1$, where the second and third choices are both interpolated weighting schemes, and the rst choice is simply the traditional k-NN.

The regression estimators ($b_{k;n}(x)$) under dierent choices of and are shown in Figure 9, along with the \true" curve which represent the underlying true (x). Note that we only present the $b_{k;n}(x)$ within a smaller range of x where the NN estimator does not suer from the boundary eect. There are several observations from the results of these toy examples.

First of all, interpolated weight does ensure data interpolation. As x gets closer to the any observed x_i, the estimator $b_{;n}(x_k)$ is forced towards y_i. As a consequence, $b_{;n}(x_k)$ is spiky for interpolated-NN. In contrast, the k-NN estimator is much more smooth.

Secondly, dierent weighting schemes lead to dierent balance between bias and variance of $b_{;n\,k}$ In the rst setting where 0, the NN algorithm with any weighting scheme is

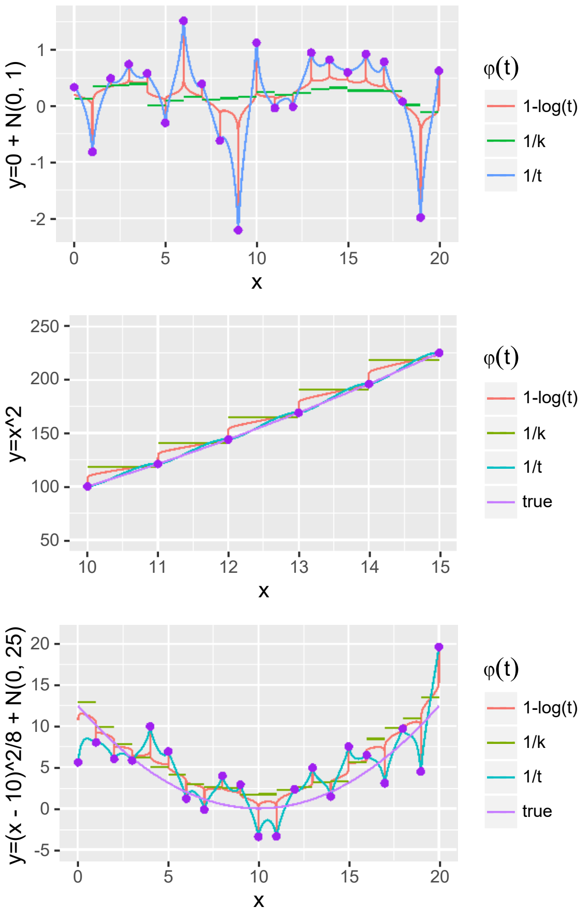

Figure 9. Three Toy Simulations: Upper: 0; Middle: (x) = x^2; Lower: (x) = (x 10)^2=8.

unbiased, hence it corresponds to the extreme situation where bias is always 0. The second model is noiseless (i.e., ^{2}=0), thus it corresponds to the opposite extreme case where variance is always 0. In the no-bias setting, k-NN performs the best, and interpolated-NN estimators lead to huge fluctuation. In the noiseless case, k-NN has the largest bias, and (t) = t ^{1} leads to the smallest bias. These observations are consistent with our arguments above, i.e., k-NN prioritizes minimizing the variance of the nearest neighbor estimator, while interpolated-NN prioritizes reducing the estimation bias instead. For the comparison between the two different interpolated weighting schemes in this example, we comment that the faster (t) increases to infinite as t ! 0, the smaller bias it will yield. Thus (t) = t ^{1} leads to smaller bias than (t) = 1 log(t), at the expense of larger estimation variance. The third model, (x) = (x 10)^2=8, involves both noise and bias. From Figure 9, the estimation of (t) = 1=k tends to stay above the true , i.e., high bias, due to the convexity of . The estimators are more rugged for interpolated-NN but fluctuate along the true (x). In this case, it is difficult to claim which one is the best visually, but the trade-off phenomenon between bias and variance is clear.

In conclusion, a faster decreasing leads to a smaller bias but a larger variance, and non-interpolated weights such as k-NN leads to a larger bias but a smaller variance.

Appendix B. Preliminary Proposition. This section provides an useful result when integrating c.d.f:

Proposition B.1. From Lemma S.1 in [35], we have for any distribution function G,

$$Z_{R}[G(\quad bu \quad a) \quad 1_{fu<0g}]du = \frac{1}{b} \quad a + Z_{R} \quad tdG(t) \quad ;$$

$$Z_{R} u[G(\quad bu \quad a) \quad 1_{fu<0g}]du = \frac{1}{b^2} \frac{a^2}{2} + \frac{1}{2} Z_{R} t^2 dG(t) + a Z_{R} \quad tdG(t):$$

Appendix C. Proof of Theorem 3.1. This section demonstrates the proof of Theorem 3.1.

Based on the same argument used in [17] and [9], conditional on $R_{k+1}(x)$, X^1 to X^k are i.i.d. variables whose support is a ball centered at x with radius $R_{k+1}(x)$, and as a consequence, $R_1(x)$ to $R_k(x)$ are conditionally independent given $R_{k+1}(x)$ as well.

The analysis of classification is very subtle especially when (x) is near 1=2. Hence, we need to have the following partition over the space X . Let p = 2k=n. Denote E = $f9R_i > r_p; i = 1; : : : ; kg$, and $ER_{k;n}(x) \qquad R(x)$ as the excess risk. Then define

$$e_{k;n;}(xjR_{k+1}) = E[(R_1=R_{k+1}) \quad (X^1)]=E(R_1=R_{k+1}) \quad ;$$

as well as

$$\overset{+}{X}_{p;}= fx \; 2 \; X j(x) > \frac{1}{2}; e(x) \; \frac{1}{2} + ; 8R_{k+1} < r_{2p}(x)g; X_{p;}=$$

$$fx \; 2 \; X j(x) < \frac{1}{2}; e(x) \; \frac{1}{2} \quad ; 8R_{k+1} < r_{2p}(x)g;$$

with the decision boundary area:

$$@_{p;} = X \; n (X_{p;} \overset{+}{[} X_{p;}):$$

Given $@_{p;}$, $X^+_{p;}$, and $X_{p;}$, similar with Lemma 8 in [17], the event of $g(x) = \;\!\!\!\!b g_{;n;k}(x)$ can be covered as:

$$1_{fg(x)=bg_{k;n;}(x)g} \quad 1_{fx2@_{p;}g} + 1_{f \; \max_{i=1;:::;k} R_i r_{2p} g} + 1_{fjb_{;n;}(x) \; _k e(xjR_{k+1})jg}:$$

When $e(xjR_{k+1}) > 1=2$ and $x \; 2 \; X^+_{p;}$, assume $b_{;n;k}(x) < 1=2$, then

$$e_{k;n;}(xjR_{k+1}) \qquad b_{k;n;}(x) > e_{;k;n;}(xjR_{k+1}) \qquad 1=2 \quad :$$

The other two events are easy to figure out.

In addition, from the definition of Regret, assume $(x) < 1=2$,

$$
\begin{aligned}
& P(\hat{g}(x) = Y j X = x) \quad (x) \\
= {} & (x)P(\hat{g}(x) = 0 j X = x) + (1 \quad (x))P(\hat{g}(x) = 1 j X = x) \quad (x) \\
= {} & (x)P(\hat{g}(x) = g(x) j X = x) + (1 \quad (x))P(\hat{g}(x) = g(x) j X = x) \quad (x) \\
= {} & (x) \quad (x)P(\hat{g}(x) = g(x) j X = x) + (1 \quad (x))P(\hat{g}(x) = g(x) j X = x) \quad (x) \\
= {} & (1 \quad 2(x))P(\hat{g}(x) = g(x) j X = x):
\end{aligned}
$$

Similarly, when $(x) > 1=2$, we have

$$P(\hat{g}(x) = Y j X = x) \quad 1 + (x) = (2(x) \quad 1)P(\hat{g}(x) = g(x) j X = x):$$

As a result, the Regret can be represented as

$$\text{Regret}(k; n;) = E(j1 \quad 2(X)jP(g(X) = \hat{g}_{k;n;}(X))):$$

For simplicity, denote $p = k=n$. We then follow the proof of Lemma 20 of [17]. Without loss of generality assume $(x) > 1=2$. Under A.1, A.2, and A.3, define

$$
\begin{aligned}
{}_0 &= \sup_x je(xjR_{k+1}) \quad (x)j = O(r\ )_{2\bar{p}} = O(k=n)_=^{d}; \\
(x) &= j(x) \quad 1=2j;
\end{aligned}
$$

then

$$e(xjR_{k+1}) \ (x) \quad {}_0 = \frac{1}{2} + ((x) \quad {}_0);$$

hence $x \ 2 \ X^{+}_{p;(x) \quad 0}$.

When $(x) > {}_0$, under A.1 and A.2, the Regret can be upper bounded as

$$P(\hat{g}_{k;n;}(X) = Y) \quad P(g(X) = Y)$$

$$2(x) \; P(r_{(k+1)} > v_{2p}) + P \quad \overset{k}{\underset{i=1}{X}} W_i Y(X^i) \quad e(xjR_{k+1}) > (x) \quad {}_0$$

$$\exp(\; k{=}8)$$

$$+2(x)P \; \overset{k}{\underset{i=1}{X}} (R_i{=}R_{k+1}) \; Y(X^i) > (e(xjR_{k+1}) + (x) \quad {}_0) \; \overset{k}{\underset{i=1}{X}} (R_i{=}R_{k+1})$$

$$= \exp(\; k{=}8)$$

$$+2(x)P \; \overset{k}{\underset{i=1}{X}} (R_i{=}R_{k+1}) \; Y(X^i) \quad kE(R_1{=}R_{k+1}) \; (X^1)$$

$$(e(xjR_{k+1}) + (x) \quad {}_0) \; \overset{k}{\underset{i=1}{X}} (R_i{=}R_{k+1}) \quad E(R_1{=}R_{k+1})$$

$$> k(e(xjR_{k+1}) + (x) \quad {}_0)E(R_1{=}R_{k+1}) \quad kE(R_1{=}R_{k+1}) \; (X^1)$$

$$= \exp(\; k{=}8) + 2(x)P \; \overset{k}{\underset{i=1}{X}} (R_i{=}R_{k+1}) \; (Y(X^i) \quad e(xjR_{k+1}))$$

$$((x) \quad {}_0) \; \overset{k}{\underset{i=1}{X}} (R_i{=}R_{k+1}) \quad kE(R_1{=}R_{k+1}) \; ! \; > \; k((x) \quad {}_0)E(R_1{=}R_{k+1}) \quad : \tag{C.1}$$

Since

$$E \; \overset{k}{\underset{i=1}{X}} (R_i{=}R_{k+1}) \; (Y(X^i) \quad e(xjR_{k+1})) = 0;$$

$$E((x) \quad {}_0) \; \overset{k}{\underset{i=1}{X}} (R_i{=}R_{k+1}) \quad kE(R_1{=}R_{k+1}) \; ! \; = 0;$$

we can use Markov inequality to the power of () to bound the probability in (C.1). Denote

$$Z_i(x) = \frac{R_i}{R_{k+1}} \quad (Y(X^i) \quad e(xjR_{k+1})) \quad ((x) \quad {}_0) \quad \frac{R_i}{R_{k+1}} \quad +((x) \quad {}_0)E \quad \frac{R_1}{R_{k+1}}$$

for simplicity. Note that

$$Var(Z_1(x)) = O((x) \quad {}_0):$$

For dierent settings of and , the following steps have the same logic but dierent details:

Case 1: < 2 and $< d=3$:. Considering the problem that the upper bound can be much greater than 1 when (x) is small, we dene $_i = 2^i{}_0$, taking $i_0 = \min f i \ 1 j \ (_i \quad _0)^2 \quad > 1=k g$. In this situation, since $E Z_1^3(x) < 1$, we can adopt non-uniform Berry-Essen Theorem for the proof:

$$P(b g_{k;n;}(X) = Y) \quad P(g(X) = Y)$$

$$= E(R_{k;n}(X) \quad R(X))1_{f(X) i_0 g} + E(R_{k;n}(X) \quad R(X))1_{f(X) > i_0 g}$$

$$2_{i_0} P((X) \quad i_0) + \exp(\quad k=8) + 4E \quad (X)1_{f_{i_0} < (X) g} \quad \frac{p \overline{k}((x \quad) \quad)_0}{p \overline{V\,ar(Z_1(X)jX)}} \;!\#$$

$$+4E \quad (X)1_{f_{i_0} < (X) g} \quad \frac{c}{k} \frac{1}{p} \frac{1}{1 + k^{3=2}((x) \qquad _0)^3}:$$

Due to -margin condition, the two terms from Berry-Essen Theorem in the above inequality become

$$E \quad (X)1_{f_i < (X) < _{i+1} g} \quad ^1 \quad \frac{c}{p \overline{k}} \frac{1}{1 + k^{3=2}((x) \quad _0)^3}$$

$$^{+1}_{i+1} \quad \frac{c_1}{p \overline{k}} \frac{1}{1 + k^{3=2}(_i \quad _0)^3}$$

$$= \frac{1}{k^{(+1)=2}} (\ ^p k_{i+1})^{+1} \quad \frac{c}{k} \frac{1}{p} \frac{1}{1 + k^{3=2}(_i \qquad _0)^3};$$

and

$$" \qquad \qquad \qquad !\#$$

$$E \quad (X)1 \quad _{f_{i_0} < (X) g} \quad \frac{p \overline{k}((x) \quad _0)}{p \overline{V ar(Z_1(X)jX)}}$$

$$h \qquad \qquad p \qquad \qquad i$$

$$E \quad (X)1_{f_{i_0} < (X) g} \quad c_3 \quad k((x) \quad ^p{}_0)$$

$$c_4 \frac{p \quad ^{+1}_{i} \quad ^{+1}}{k(\quad _i \quad _0)} \exp \quad c^2 k_3(_i \qquad _0)^2 \quad :$$

The quantity $\exp(\ c_3^2 k(_i \quad _0)^2)$ is larger than 1 if $_i \quad c_5 = {}^p k$. When $_i > c_5 = {}^p k$, for some constant $c_5 > 0$,

$$\frac{ \frac{^{+1}_{i+1}}{(\ _i \ )\ _0} \exp \quad c_3 k(_i^2 \qquad _0) \quad ^2}{\frac{^{+1}_{i}}{_0)} \exp_1 \quad c_3 k(_{i \ 1} \ ^2 \ _0)^2} \quad = 2^{+1} \frac{2^{i \quad 1} \quad 1}{2^i \quad 1} \frac{\exp \quad c_3^2 k(_i \qquad _0)^2 \quad ^i}{\exp \quad c_3 k(_{i \ 1} \quad _0)^{2^2}} \; ($$

$$2 \quad \frac{\exp \quad c_3^2 k(_i \qquad _0)^2}{\exp \quad c_3^2 k(_{i \ 1} \quad _0)^2} \quad < 1=2:$$

Therefore the sum of the excess risk can be bounded. When < 2,

$$E\,(X)1_{f_i<(X)<_{i+1}g} \quad \overset{c}{k\,1 + k^{1=2}((x)} \quad \frac{1}{\qquad\qquad 0)^3}$$

$$O\ \frac{1}{k^{(+2)=2}}\ \overset{X}{}\ [\,{}^{p}k\,(_{i+1}\qquad {}_{i})]\ {}^{2}{}_{ii_0}$$

$$O\ \frac{1}{k^{(+2)=2}}\ \overset{X}{}_{ii\ _0}(\ \overset{p}{\overline{k}}\ {}_0)\quad {}^{2}\ {}_{i}(\ {}^{2)}\ 2$$

$$=\ O\ \frac{1}{k^{(+2)=2}}\,k^{(\ \ 2)=2}\ {}^{2}_{i_0}\ \ =\ O\ \frac{{}^{2}_{i_0}}{k^{2}}\ !\ ;$$

and

$$E\ \ (X)1\ \ f_{i_0}<(X)g\qquad \frac{{}^{p}k((x)\qquad {}_0)}{{}^{p}\ Var(Z_1(X)jX)}\ !\ \#$$

$$O\ \ \frac{1}{\sqrt{k}}\ \overset{X}{}_{ii_0}\ (_i\ \frac{{}^{+1}_{i+1}}{\quad {}_0)}\ \exp\ \ c^{2}k(_i\ {}_3\qquad {}_0)^2$$

$$=\ O\ \ \frac{1}{\sqrt{k}}\ {}_{i_0+1}\ \exp\ \ c_3\,k(_{i_0}\ {}_2\qquad {}_0)^2:$$

Recall that $_{i_0} > {}_0$ and $_i > 1{=}{}^{2}_{0}k$, hence when $_i = O(1{=}{}^{2}_{0}k)$, we can obtain the minimum upper bound

$$P(\hat{g}_{k;n;}(X) = Y)\quad P(g(X) = Y)\ O({}^{+1}_{0}) + O\qquad {}^{(+1)=2}\,!\ k\quad :$$

Taking k (n$^{=(2+d)}$), the upper bound becomes O(n $^{(+1)=(2+d)}$). Case $\frac{1}{}$2: < d=(): . In this case, we have

$$P(\hat{g}_{k;n;}(X) = Y)\qquad P(g(X) = Y)$$

$$=\ E(R_{k;n}(X)\quad R(X))1_{f(X)_{i_0}g} + E(R_{k;n}(X)\qquad R(X))1_{f(X)>_{i_0}g}$$

$$2_{i_0}P(\,(X)\ \ _{i_0}) + \exp(\ \ k{=}8)$$

$$+4E\ \overset{2}{\underset{4}{6}}(X)1_{f_{i_0}<(X)g}\ ((X)\ \frac{E\ \ P_{\,i=1}^{\,k}\ Z_i(X)\ \ {}^{()}}{{}_0)^{()}k^{()}E^{()}(R_1{=}R_{k+1})}\ \overset{3}{\underset{5}{7}}\ ;$$

while for some constant $c_1 > 0$,

$$E \begin{matrix} 2 \\ 6 \\ 4 \end{matrix} (X) 1_{f_{i} < (X)_{i+1} g} ((X) \quad \frac{E \, {}^{P}{}_{k}{}_{i=1} \, Z_i(X)^{()}}{{}_0)^{()} k^{()} E^{()} (R_1 = R_{k+1})} \begin{matrix} 3 \\ 7 \\ 5 \end{matrix}$$

$$E \quad (X) 1_{f_i < (X)_{i+1} g} \quad (_i \quad \frac{E \, {}^{P}{}_{k}{}_{i=1} \, Z_i(X)^{()}}{{}_0)^{()} k^{()} E^{()} (R_1 = R_{k+1})}$$

$$_{i+1} P((X) \quad _{i+1}) \, (_i \quad \frac{E \, {}^{P}{}_{k}{}_{i=1} \, Z_i(X)^{()}}{{}_0)^{()} k^{()} E^{()} (R_1 = R_{k+1})}$$

$$c_1 \quad \frac{1}{k}^{()=2} \quad {}_{i+1}^{+1} = (_i \quad {}_0)^{()};$$

where the last inequality is obtained from -margin condition and the assumption that d () > 0. Note that since () > + 1, for some constant 0 < c < 1,

$$\text{(C.2)} \qquad \frac{{}_{i+1}^{+1} = (_i \quad {}_0)^{()\,+1} = (_i}{1 \;{}_i \quad {}_0)^{()}} < c:$$

Therefore the sum of the excess risk for can be bounded, where

$$E \begin{matrix} 2 \\ 6 \\ 4 \end{matrix} (X) 1_{f_{i_0} < (X) g} ((X) \quad \frac{E \, {}^{P}{}_{k}{}_{i=1} \, Z_i(X)^{()}}{{}_0)^{()} k^{()} E^{()} (R_1 = R_{k+1})} \begin{matrix} 3 \\ 7 \\ 5 \end{matrix}$$

$$\text{(C.3)} \qquad O \quad \frac{1}{k}^{()=2} \quad \underset{i i_0}{X} \quad {}_{i+1}^{+1} = (_i \quad {}_0)^{()}$$
$$O \quad {}^{+1}_{i_0}:$$

Recall that $_{i_0} > {}_0$ and $^2 > 1 = k_{i_0}$, hence when $^2 = O(1 = k)_{i_0}$, we can obtain the minimum upper bound

$$_{;n;}(\hat{X}_k) = Y) \qquad P(g(X) = Y) = O({}_0^{+1}) + O \quad \frac{1}{k}^{(+1)=2} P(\not{b}\, ! \qquad :$$

Taking k $(n^{=(2+d)})$, the upper bound becomes $O(n^{(+1)=(2+d)})$.

Appendix D. Proof of Theorem 3.2. This section provides the proof of Theorem 3.2.

For $CIS_{k;n}()$, since we are considering two independent sets of data, it can be upper bounded by two times misclassication error:

$$\begin{aligned} CIS_{k;n}() &= 2P(\hat{g}_1(X) = g(X); \hat{g}_2(X) = g(X)) \\ &= 2E[1 \quad P(\hat{g}_2(X) = g(X)jX)]\, P(\hat{g}_1(X) = g(X)jX) \\ & 2EP(\hat{g}_1(X) = g(X)jX) \\ &= 2P(\hat{g}(X) = g(X)): \end{aligned}$$

Following the same procedures as in Theorem 3.1, when $<$ 2 and $<$ d=3, we have

$$P(\hat{g}_{k;n;}(X) = g(X))$$
$$= EP(\hat{g}_{k;n;}(X) = g(X))1_{f(X)_{i_0}g} + EP(\hat{g}_{k;n;}(X) = g(X))1_{f(X)>} \quad P((X)_{i_0}g$$
$$_{i_0}) + \exp(\; k=8) + E \; 1_{f_{i_0}<(X)g} \; " \quad \frac{p \; \frac{p}{k}((X) \quad)_0}{V\,ar(Z_1(X)jX)} \; !\#$$
$$+ E \; 1_{f_{i_0}<(X)g} \; \frac{c}{k} \frac{p\frac{1}{}}{1+} \frac{1}{k^{3=2}((X) \quad _0)^3}:$$

The two terms from Berry-Essen Theorem becomes

$$E\,1_{f_i<(X)<_{i+1}g} \; \frac{c}{k} \; \frac{1}{1+p\frac{1}{}k^{3=2}((X) \quad _0)^3} \quad (\,^p k_{i+1}) \quad k^{=2} \quad \frac{c}{k} \; \frac{1}{1+k^{p\frac{1}{}=2}} \; \frac{1}{(_i \quad _0)^3};$$

and

$$E \; 1_{f_{i_0}<(X)g} \; " \quad \frac{p \; \frac{p}{k}((X) \quad)_0}{V\,ar(Z_1(X)jX)} \; !\# \quad c_4\,p\frac{_i \quad _{+1}}{k(\;_i \quad _0)} \exp(\; c^2 k(_i \quad _0)^2):$$

Thus summing up $_i < (X) < _{i+1}$ for all $i > i_0$, we have $CIS_{k;n}()$

$$O(_0) + O(1=k)^{=2}:$$

When k $n^{=(2+d)}$, we have $CIS = O(n^{\;=(2+d)})$. The proof for $\;2$ s similar.

Appendix E. Proof of Theorem 4.1 .

This section is the proof of Theorem 4.1.

To prove Theorem 4.1, we first prove the following theorem, then represents the multiplicative constants as functions of .

Theorem E.1. Assume d $\;3$ C $>$ 0 for some constant C. For regression, suppose that assumptions A.1', A.2', A.5', and A.6' hold. If k satisfies n k $n^{1 \;\; 4=d}$ for some $>$ 0,

we have

$$MSE(k;n;) = kE \underbrace{\left[\frac{(R_1(X)=R_{k+1}(X))^2}{(\sum_{i=1}^{k}(R_i(X)=R_{k+1}(X)))^2} \;{}^2(X) \right]}_{Variance}$$
$$+ k^2 E \underbrace{\left[a^2(X)E^2 \left[\frac{R_1(X)^2(R_1(X)=R_{k+1}(X))}{\sum_{i=1}^{k}(R_i(X)=R_{k+1}(X))} \;X \right] \right]}_{Bias} + Remainder;$$

where Remainder = o(MSE(k;n;)).

For classification, under A.1' to A.4', the excess risk w.r.t. becomes

$$Regret(k;n;) = \underbrace{\frac{1}{4k} B_1 Es_{k;n;}(X)}_{Variance} + \underbrace{\int_S \frac{0}{\_(x_0)} k\_(\;f(\;)k_{k;n;}(x_0))^2 dVol^{d\;\;1}(x_0)}_{Bias} + Remainder;$$

where

$$\text{Remainder} = o(\text{Regret}(k;n;));$$
$$B_1 = \int_S \frac{f(x_0)}{k\_(x_0)k} dVol^{d\ 1}(x_0);$$
$$s^2_{k;n;}(x) = \frac{E(R_1=R_{k+1})^2}{E^2(R_1=R_{k+1})};$$
$$t_{k;n;}(x) = \frac{ER_1^2(R_1=R_{k+1})}{E(R_1=R_{k+1})}:$$

E.1. Regression. Rewrite the interpolated-NN estimate at x given the distance to the $k+1$th neighbor R_{k+1}, interpolation level as

$$S_{k;n;}(x;R_{k+1}) = \sum_{i=1}^{k} W_i Y_i;$$

where the weighting scheme is defined as

$$W_i = \frac{(R_i=R_{k+1})}{\sum_{i=1}^{k}(R_i=R_{k+1})}:$$

For regression, we decompose MSE into bias square and variance, where

$$E[(S_{k;n;}(x;R_{k+1}) \quad (x))^2|x] = E\left[\sum_{i=1}^{k} W_i((X^i) \quad (x))\right]^2 + E\left[\sum_{i=1}^{k} W_i(Y_i \quad (X^i))\right]^2;$$

in which the bias square can be rewritten as

$$E\left[\sum_{i=1}^{k} W_i((X^i) \quad (x))\right]^2 = kE(W_1((X^1) \quad (x)))^2 + (k^2 \quad k)E^2(W_1((X^1) \quad (x)));$$

and the variance can be approximated as

$$E\left[\sum_{i=1}^{k} W_i(Y_i \quad (X^i))\right]^2 = kEW_1^2(X^1)^2 = k(x)^2EW^2 +_1 o:$$

Following a procedure similar as Step 1 for classification, i.e., use Taylor expansion to approximate the bias square, we obtain that for some function a, the bias becomes

$$EW_1((X^1) \quad (x)) = a(x)EW^2 R_1^2 + o:$$

As a result, the MSE of interpolated-NN estimate given x becomes,

$$E[(S_{k;n;}(x;R_{k+1}) \quad (x))^2|x] = k(x)^2EW^2 +_1 k^2a(x)^2E^2W^2R_1^2 +_1 o:$$

Finally we integrate MSE over the whole support.

E.2. Classication. The main structure of the proof follows [31]. As the whole proof is long, we provide a brief summary in Section E.2.1 to describe things we will do in each step, then in Section E.2.2 we will present the details in each step.

E.2.1. Brief Summary. Step 1: denote i.i.d. random variables $Z_i(x; R_{k+1})$ for $i = 1; : : : ; k$ where

$$Z_i(x; R_{k+1}) = \frac{(R_i=R_{k+1}) \quad (Y(X^i) \qquad 1=2)}{E(R_i(x)=R_{k+1}(x))};$$

then the probability of classifying as 0 becomes

$$P(S_{k;n;}(x) < 1=2) = P \quad \overset{!}{\underset{i=1}{X^k}} Z_i(x; R_{k+1}) < 0 \quad :$$

The mean and variance of $Z_i(x; R_{k+1})$ can be obtained through Taylor expansion of and density function of x:

$$E(Z_1(x; R_{k+1})) = (x) \qquad \frac{1}{2} + a(x) \frac{R_1^2(R_1=R_{k+1})}{E\,E(R_1=R_{k+1})} + o\,V$$

$$ar(Z_1(x; R_{k+1})) = \frac{1}{4}\frac{(E \quad R_1 \quad R_{k+1})^2}{E \quad (R_1 = R_{k+1})} + o;$$

for some function a. The smoothness conditions are assumed in A.4 and A.5.

Note that on the denominator of Z_i, there is an expectation $E(R_i(x)=R_{k+1}(x))$. From later calculation in Corollary 4.3, the value of this expectation in fact has little changes given or without a condition of R_{k+1}, and it is little aected by x either.

Step 2: One can rewrite Regret as

$$\underset{R^d}{Z} \quad P \quad \underset{i=1}{X^k} W_iY_i \quad \frac{1}{2} \overset{!}{} \qquad 1_{f(x)<1=2g} \overset{!}{} \quad dP(x):$$

From Assumption A.2, A.4, the region where b is likely to make a wrong prediction is near $fxj(x) = 1=2g$, thus we use tube theory to transform the integral of Regret over the d-dimensional space into a tube, i.e.,

$$\underset{R^d}{Z} \quad P \quad \underset{i=1}{X^k} W_iY_i \quad \frac{1}{2} \overset{!}{} \qquad 1f(x) < 1=2g \overset{!}{} \quad dP(x)$$

$$= f1 + o(1)g \underset{S}{Z} \; Z \qquad tk\_(x_0)k \quad P(S_{k;n}(x_0^t) < 1=2) \qquad 1_{ft<0g} \quad dtdVol^{d \quad 1}(x_0) + o:$$

The term will be dened in detail later. Basically, when is within a suitable range, the integral over t will not depend on asymptotically.

Step 3: given R_{k+1} and x, the nearest k neighbors are i.i.d. random variables distributed in $B(x; R_{k+1})$, thus we use non-uniform Berry-Esseen Theorem to get the Gaussian approximation of the probability of wrong prediction:

$$Z \; Z \quad tk\_(x_0)k \;\; P(S_{k;n}(x_0^t) < 1=2) \quad 1_{ft<0g} \;\; dtdVol^d \;\; {}^1(x_0)$$

$$\overset{S}{Z} \; Z \qquad ! \qquad !$$

$$= \;\; _S \quad tk\_(x_0)kE_{R_{k+1}} \qquad \frac{p \; kEZ_1(x_0^t; R_{k+1})}{kV\,ar(Z_1(x_0^t; R_{k+1}))} \qquad 1_{ft<0g} \;\; dtdVol^d \;\; {}^1(x_0) + o:$$

Step 4: take expectation over all R_{k+1}, and integral the Gaussian probability over the tube to obtain

$$Z \; Z \qquad ! \qquad !$$

$$_S \quad tk\_(x_{\;0})kE_{R_{k+1}} \qquad \frac{p \; kEZ_1(x_0^t; R_{k+1})}{kV\,ar(Z_1(x_0^t; R_{k+1}))} \qquad 1_{ft<0g} \;\; dtd \;\; ol^d \;\; {}^1(x_0)$$

$$0 \quad 0 \qquad\qquad 1 \quad V \quad 1$$

$$Z \; Z$$

$$= \;\; _S \;\; _R \quad tk\_(x_0)k \, @@ \; \frac{tk\_(x_0)k}{q \; \frac{}{\;2\;} \; s_{k;n;}=k} \;\; \frac{E(R_1=R_{k+1}) \;\; a \; x^t)R_0^2 A \;\; 1}{q \; \frac{}{\;2\;(} \; s_{k;n;}=k} \qquad 1_{ft<0g}{}^A \, dtdVol^d \;\; {}^1(x_0)$$

$$+o$$

$$= \frac{B_1 \;\; E(R_1=R_{k+1}) \;\; {}^2}{4k \;\; E^2(R_1=R_{k+1})} + \; _S^Z \; \frac{k\_(x_0)k}{k\_(x_0)k^2} a^2(x_0) \frac{E^2(R_1=R_{k+1}) \;\; R^2}{E^2(R_1=R_{k+1})} {}_1 dVol^d \;\; {}^1(x_0) + o$$

$$= \frac{1}{8k} B_1 Es_{k;n;}^2 + \; _S^Z \; \frac{k\_(x_0)k}{2k\_x \;\; (k^2} a^2(x_0) t_{k;n;}^2 dVol^d \;\; {}^1(x_0) + o:$$

E.2.2. Details. Denote a_d is the Euclidean ball volume parameter

$$a_d = Vol(B(0; 1)) = (=2)^d=^2= \;\; (d=2 + \;1):$$

Dene p = k=n and $r_{2p} = \sup_x ER_{2k}(x)$. Denote E be the set that there exists R_i such that $R_i > r_{2p}$, then for some constant c > 0,

$$= \frac{c}{a^{1=d}\;c_d^{1=d}\;\;{}_0} \;\; n \;\; \frac{2k}{\;}{}^{1=d} \; {}^{r_{2p}} \; :$$

Hence from Claim A.5 in [9], there exist c_1 and c_2 satisfying

$$P(E) \;\; c_1k \exp(\;\; c_2k):$$

Step 1: in this step, we gure out the i.i.d. random variable in our problem, and calculate its mean and variance given x.

Denote

$$Z_i(x; R_{k+1}) = \frac{(R_i=R_{k+1}() \;\; (Y(X^i) \qquad 1=2)}{E \;\; R_i=R_{k+1})}; \tag{E.1}$$

then the dominant part we want to integrate becomes

$$P \quad S_{k;n}(x; R_{k+1}) \quad \frac{1}{2} \qquad = P$$

$$! \\ (R_i{=}R_{k+1}) \quad \sum_{i=1}^{k} (Y(X^i) \quad 1{=}2) < 0 R_{k+1}$$

$$= P \quad \frac{P_{i=1}^{k} Z_i(x; R_{k+1}) \quad kEZ_1(x; R_{k+1})}{p\, kVar(Z_1(x; R_{k+1}))} < \frac{kEZ_1(x; R_{k+1})}{p\, kVar(Z_1(x; R_{k+1}))} R_{k+1} \quad ! \quad :$$

Therefore, one can adopt non-uniform Berry-Essen Theorem to approximate the probability using normal distribution. Unlike [31] in which $EY(X^i)$ is calculated, since the i.i.d. item in non-uniform Berry-Essen Theorem is Z rather than Y, we now calculate mean and variance of Z. Under R_{k+1},

$$_{k;n;}(x; R_{k+1}) := EZ_1(x; R_{k+1}) = \frac{E(R_1{=}R_k \; R)_{+1}(Y(X^1) \qquad 1{=}2)}{(E =_1 R_{k+1})}$$

$$= \frac{E(R_1{=}R_{k+1}) \;\; ((X^1) \quad 1{=}2)}{E(R_1{=}R_{k+1})};$$

and

$$EZ_1^2(x; R^k_{+1}) = \frac{E(R_1{=}R_{k+1}) \;^2 \; (Y(X^1) \quad 1{=}2)^2}{E^2(R_1{=}R_{k+1})}$$

$$= \frac{E(R_1{=}R_{k+1}) \;^2}{4E^2(R_i{=}R_{k+1})};$$

$$^2_{k;n;}(x; R_{k+1}) := Var(Z_1(x; R_{k+1})):$$

Then the mean and variance of Z_i can be calculated as

$$_{k;n}(x_0^t; R_{k+1}) = EZ_1(x_0^t; R_{k+1}) + \frac{1}{2} = \frac{E(R_1{=}R_{k+1}) \quad (X^1)}{E(R_1{=}R_{k+1})}$$

$$= \frac{E(R_1{=}R_{k+1}) \quad (x^t_0) + (X^1 \quad x^t_0)^{>}\_(x^t_0) + 1{=}2(X^1 \quad x^t_0)^{>}\bullet(x^t_0)(X^1 \quad x^t_0)}{E(R_1{=}R_{k+1})} + o(R^3_{k+1})$$

$$= (x^t_0) + \frac{E(R_1{=}R_{k+1}) \;\; (X^1 \quad x^t_0)^{>}\_(x^t_0)}{E(R_1{=}R_{k+1})}$$

$$+ \frac{1}{2}\frac{E(R_1{=}R_{k+1}) \;\; tr[\bullet(x^t_0) \;\; (X^1 \quad x^t_0)(X^1 \quad x^t_0)^{>} \;]}{E(R_1{=}R_{k+1})} + O(R^3_{k+1}):$$

Fixing R_1 and R_{k+1}, denoting $f(jx_2; R)$ as the conditional density of X given x_2 and kX

$x_2k = R_1$, we have

$$E((X^1 \quad x_0^t)^>\_(x_0^t)jR_1)$$

$$= \int (x \quad x_0^t)^>\_(x_0^t)f(xjx_0^t; R_{k+1})dx$$

$$= \int (x \quad x_0^t)^>\_(x_0^t)[f(x_0^tjx_0^t; R_1) + f^0(x_0jx_0^t; R_1)^>(x \quad x_0^t) + o]dx$$

$$= 0 + \int (x \quad x_0^t)^>\_(x_0^t)f^0(x_0^tjx_0^t; R_1)^>(x \quad x_0^t)dx + o$$

$$= \mathrm{tr} \quad \_(x_0^t)f^0(x_0^tjx_0^t; R_1)^> \int (x \quad x_0^t)(x \quad x_0^t)^>dx \quad + o \tag{E.2}$$

and

$$\mathrm{tr} \quad \frac{1}{2}\bullet(x_0^t)E(X^1 \quad x_0^t)(X^1 \quad x_0^t)^>jR_1$$

$$= \mathrm{tr} \quad \frac{1}{2}\bullet(x_0^t) \int (x \quad x_0^t)(x \quad x_0^t)^>f(xjx_0^t; R_1)dx$$

$$= \mathrm{tr} \quad \frac{1}{2}\bullet(x_0^t) \int (x \quad x_0^t)(x \quad x_0^t)^>[f(x_0^tjx_0^t; R_1) + f^0(x_0jx_0^t; R_1)^>(x \quad x_0^t) + o]dx$$

$$= \mathrm{tr} \quad \frac{f(x_0jx_0^t; R_1)}{2}\bullet(x_0^t) \int (x \quad x_0^t)(x \quad x_0^t)^>dx + o; \tag{E.3}$$

Then taking function $a(x)$ for x such that

$$a(x_0^t)R_1^2 = E((X^1 \quad x_0^t)^>\_(x_0^t)jR_1) + \mathrm{tr} \quad \frac{1}{2}\bullet(x_0^t)E(X^i \quad x_0^t)(X^i \quad x_0^t)^>jR_1 + o:$$

The dierence caused by the value of R_1 is only a small order term.

Finally,

$$_{k;n;}(x_0; R_{k+1}) = (x_0) + tk\_(x_0)k + a(x_0)t_{k;n;}(R_{k+1}) + o:$$

Step 2: in this step we construct a tube based on the set $S = fxj(x) = 1=2g$, then gure out that the part of Regret outside this tube is a remainder term.

Assume $_{k;n}$ satises $s_{k;n;} = o(_{k;n})$ and $_{k;n} = o(s_{k;n;}k^{1=2})$, then the residual terms throughout the following steps will be $o(s_{k;n;}^2 + t_{k;n;}^2)$. Hence although the choice of $_{k;n}$ is dierent among choices of k and n, this does not aect the rate of Regret. Note that we ignore the arguments x and R_{k+1} as from A.2 and A.4, $s_{k;n;}^2(x; R_{k+1}) \quad 1=k$ and $t_{k;;n;}(x; R_{k+1})$ $(k=n)^{2=d}$ for all x while $R_{k+1} \quad (k; n)^{2=d}$ in probability.

By [31], recall that $(x) = d(_1P_1(x) \quad _2P_2(x))$, then

$$\int_{R^d} P \quad \sum_{i=1}^{k} W_iY_i \quad \frac{1}{2} \quad 1_{f(x)<1=2g} \quad dP(x)$$

$$= f1 + o(1)g \int_S \int_{k;n}^{k;n} tk\_(x_0)k \quad P(S_{k;n}(x_0^t) < 1=2) \quad 1_{ft<0g}dtdVol^{d \quad 1}(x_0) + r_1;$$

where

$$r_1 = \mathrm{Z}_{R^d n S^{k;n}} \; P \; \mathrm{X}^k_{i=1} \; W_i Y_i \; \frac{1}{2} \; ! \; 1_{f(x)<1=2g} \; ! \; dP(x)$$

$$= \mathrm{Z}_{R^d n S^{\;k;n}} E_{R_{k+1}} \; P \; \mathrm{X}^k_{i=1} \; Z_i(x; R_{k+1}) < 0 \; ! \; 1_{f(x)<1=2g} \; ! \; dP(x):$$

For r_1,

$$0 \; \mathrm{Z}_{R^d n S^{k;n} \backslash fxj(x)<1=2g} E_{R_{k+1}} \; P \; \mathrm{X}^k_{i=1} \; Z_i(x; R_{k+1}) \;\; 0 \; ! \; 1_{f(x)<1=2g} \; ! \; dP(x)$$

$$= \mathrm{Z}_{R^d n S^{k;n} \backslash fxj(x)<1=2g}$$

$$E_{R_{k+1}} \; P \; \mathrm{X}^k_{i=1} \; Z_i(x; R_{k+1}) \quad kEZ_1(x; R_{k+1}) > \quad kEZ_1(x; R_{k+1}) \; ! \; ! \; dP(x):$$

Using non-uniform Berry-Essen Theorem, when $EZ^3_1(x; R_{k+1}) < 1$, i.e., $<$ d=3, it becomes Z

$$R^d n S^{k;}{}_n \backslash fxj(x)<1=2g$$

$$E_{R_{k+1}} \; P \; \mathrm{X}^k_{i=1} \; Z_i(x; R_{k+1}) \quad kEZ_1(x; R_{k+1}) > \quad kEZ_1(x; R_{k+1}) \; ! \; ! \; dP(x)$$

$$\mathrm{Z}_{R^d n S^{k;n} \backslash fxj(x)<1=2g} E_{R_{k+1}} \; \frac{\mathrm{p}\; kEZ_1(x; R_{k+1})}{Var(Z_1(x; R_{k+1}))} \; ! \; dP(x)$$

$$+c_1 \,\mathrm{p}\, \frac{1}{k} \; \mathrm{Z}_{R^d n S^{k;n} \backslash fxj(x)<1=2g} E_{R_{k+1}} \frac{1}{1 + k^{3=2} jEZ_1(x; R_{k+1})j^3} dP(x);$$

where $(x) = 1 \quad (x)$. Since $s_{k;n;}(x; R_{k+1}) = o({}_{k;n})$, $r_1 = o(s^2_{k;n;}(x; R_{k+1}))$. By the denition of ${}_{k;n}$, we have

$$\exp({}^2_{k;n} = s^2_{k;n;}(x; R_{k+1})) = o(s^2_{k;n;}) + o(1=k);$$

$$\inf_{x 2 R^d n S^{\;k;n}} j(x) \quad 1=2j \;\; c \;\; {}_3 \; {}_{k;n}:$$

As a result, using Berstain inequality, r_1 is a smaller order term compared with $s^{2\,n}_{k;\;;}$ when $s^2_{k;n;}(R_{k+1}) = o(1)$, hence $r_1 = o(s^2_{k;n;})$.

Step 3: now we apply non-uniform Berry-Esseen Theorem. From Step 3, we have

$$\mathrm{Z}_{S} \; \mathrm{Z}^{\;k;n}_{\;k;n} \; E_{R_{k+1}} tk \,\_ (x_0)k \;\; P(S_{k;n}(x^t_0) < 1=2jR_{k+1}) \quad 1_{ft<0g} dt dVol^d \;\; {}^1(x_0)$$

$$= \mathrm{Z}_{S} \; \mathrm{Z}^{\;k;n}_{\;k;n} \; E_{R_{k+1}} tk \,\_ (x_0)k \quad \frac{\mathrm{p}\; kEZ_1(x^t_0; R_{k+1})}{kVar(Z_1(x^t_0; R_{k+1}))} \; ! \quad 1_{ft<0g} \; ! \quad dt dVol^d \;\; {}^1(x_0) + r_2;$$

where based on non-uniform Berry-Esseen Theorem:

$$P\left(\frac{\sum_{i=1}^{k} Z_i(x_0^t;R_{k+1}) - kEZ_1(x_0^t;R_{k+1})}{\sqrt{kVar(Z_1(x_0^t;R_{k+1}))}} < -\frac{kEZ_1(x_0^t;R_{k+1})}{\sqrt{kVar(Z_1(x_0^t;R_{k+1}))}}\right) - \Phi\left(-\frac{kEZ_1(x_0^t;R_{k+1})}{\sqrt{kVar(Z_1(x_0^t;R_{k+1}))}}\right)$$

$$\leq c\,\frac{kE|Z_1(x_0^t;R_{k+1})|^3}{k^{3/2}Var^{3/2}(Z_1(x_0^t;R_{k+1}))}\frac{1}{\left(1+\frac{kEZ_1(x_0^t;R_{k+1})}{\sqrt{kVar(Z_1(x_0^t;R_{k+1}))}}\right)^3};$$

and

$$|r_2| \leq \int_S\int_{-{}_{k;n}}^{{}_{k;n}} E_{R_{k+1}} t\|-(x_0)\| \frac{kE|Z_1(x_0^t;R_{k+1})|^3}{k^{3/2}Var^{3/2}(Z_1(x_0^t;R_{k+1}))}\frac{1}{\left(1+\frac{kEZ_1(x_0^t;R_{k+1})}{\sqrt{kVar(Z_1(x_0^t;R_{k+1}))}}\right)^3}dtdVol^{d-1}(x_0).$$

For r_2,

$$
\begin{aligned}
r_2 &\leq \int_S\int_{-{}_{k;n}}^{{}_{k;n}} E_{R_{k+1}} t\|-(x_0)\| \frac{kE|Z_1(x_0^t;R_{k+1})|^3}{k^{3/2}Var^{3/2}(Z_1(x_0^t;R_{k+1}))}\frac{1}{\left(1+\frac{kEZ_1(x_0^t;R_{k+1})}{\sqrt{kVar(Z_1(x_0^t;R_{k+1}))}}\right)^3}dtdVol^{d-1}(x_0)\\
&= \frac{c_1}{\sqrt{k}}\int_S\int_{-{}_{k;n}}^{{}_{k;n}} E_{R_{k+1}} t\|-(x_0)\| \frac{1}{\left(1+\frac{kEZ_1(x_0^t;R_{k+1})}{\sqrt{kVar(Z_1(x_0^t;R_{k+1}))}}\right)^3}dtdVol^{d-1}(x_0)\\
&= \frac{c_2}{\sqrt{k}}\int_S\int_{|t|<s_{k;n;}(x)} t\|-(x_0)\|dtdVol^{d-1}(x_0)\\
&\quad + \frac{c_3}{\sqrt{k}}\int_S\int_{s_{k;n;}(x)<|t|<{}_{k;n}} t\|-(x_0)\|\frac{1}{1+k^{3/2}t^3}dtdVol^{d-1}(x_0)\\
&= o(s_{k;n;}^2).
\end{aligned}
$$

Step 4: the integral becomes

$$\int_S E_{R_{k+1}} \int_{k;n}^{k;n} tk\_(x_0)k \left(p\frac{kEZ_1(x_0^t;R_{k+1})}{kVar(Z_1(x_0^t;R_{k+1}))} \right) 1_{ft<0g}\ dtd\ Vol^{d-1}(x_0)$$

$$= \int_S E_{R_{k+1}} \int_{k;n}^{k;n} tk\_(x_0)k \left(\left(q\frac{tk\_(x_0)k}{s^2_{k;n;}(x;R_{k+1})=k} \frac{E(R_1=R_{k+1})\ a(x^t_0)R^2}{q s^{k\ 2}_{;n;}(x;R_{k+1})=k} \right)^1 1_{ft<0g} \right)^1 dtd\ Vol^{d-1}(x_0)$$

$$+r_3 + o$$

$$= \int_S E_{R_{k+1}} \int_R tk\_(x_0)k \left(\left(q\frac{tk\_(x_0)k}{s^2_{k;n;}(x;R_{k+1})=k} \frac{E(R_1=R_{k+1})\ a(x^t_0)R^2}{q s^{k\ 2}_{;n;}(x;R_{k+1})=k} \right)^1 1_{ft<0g} \right)^1 dtd\ Vol^{d-1}(x_0)$$

$$+r_3 + r_4 + o$$

$$= \int_S \int_R tk\_(x^0)k \left(\left(q\frac{tk\_(x_0)k}{s^{k\ 2}_{;n;}(x)=k} \frac{E(R_1=R_{k+1})\ a(x^t_0)R^2}{q s^{k\ 2}_{;n;}(x)=k} \right)^1 1_{ft<0g} \right)^1 dtd\ Vol^{d-1}(x_0)$$

$$+r_3 + r_4 + r_5 + o$$

$$= \frac{B_1 E(R_1=R_{k+1})^2}{4k E^2(R_1=R_{k+1})} + \int_S \frac{f(x_0)}{k\_(x_0)k} a^2(x_0) \frac{E^2(R_1=R_{k+1})\ R^2}{E^2(R_1=R_{k+1})} d^1\ Vol^{d-1}(x_0) + r_3 + r_4 + r_5 + o:$$

Note that $k\_(x_0)k=k\_(x_0)k = 2f(x_0)$. The last step follows Proposition B.1 and the fact that R_{k+1} does not affect the dominant parts. The term $E((R_1=R_{k+1})\ jR_{k+1})$ is almost the same for all R_{k+1}. For the small order terms, following [31] we obtain

$$r_3 = \int_S E_{R_{k+1}} \int_{k;n}^{k;n} tk\_(x_0)k \left(p\frac{kEZ_1(x_0^t;R_{k+1})}{kVar(Z_1(x_0;R_{k+1}))} \right. \left. \left(q\frac{tk\_(x_0)k}{s^2_{k;n;}(x;R_{k+1})=k} \frac{E(R_1=R_{k+1})\ a(x^t_0)R_1}{q s^{k\ 2}_{;n;}(x;R_{k+1})=k} \right)^1 \right) dtdVol^{d-1}(x_0);$$

$$= o(s^2_{k;n;} + t^2_{k;n;});$$

and

$$r_4 = \int_S \frac{k\_(x_0)k}{k\_(x_0)k^2} E_{R_{k+1}} \int_{R n[\ k;n;\ k;n]} tk\_(x^0)k \left(\left(q\frac{tk\_(x_0)k}{s^2_{k;n;}=k} \frac{E(R_1=R_{k+1})\ a(x^t_0)R^2}{q s^2_{k;n;}=k} \right)^1 1_{fv<0g} \right)^1 dtd\ Vol^{d-1}(x_0)$$

$$= o(s^2_{k;n;}):$$

The term r_5 is the dierence between the normal probability given R_{k+1} and the one after taking expectation. Similar with [13], for each x, when $jtj < \ _{k;n}$, we have

$$E_{R_{k+1}} \overset{0}{@} q\frac{tk\_(x_0)k}{s^2_{k;n;}(x;R_{k+1})=k} \quad \frac{E(R_1=R_{k+1})\ \ a(x^t_0)R^2}{q\ s^2_{k;n;}(x;R_{k+1})=k} \overset{1}{A}$$

$$= \overset{0}{@}\ q\ \frac{tk\_(x_0)k}{s^2_{k;n;}(x;R_{k+1})=k} \quad \frac{E(R_1=R_{k+1})\ \ a(x^t_0)R^2}{q\ s^{k2}_{;n;}(x;R_{k+1})=k} \overset{1}{A} + O(kVar(t_{k;n;}(x;R_{k+1}))) + o:$$

Following step 3 in [13], we obtain

$$Var(t_{k;n;}(x;R_{k+1})) \quad \frac{1}{k^2}\sum^{k}_{j=1} E((X^1) \qquad (x))^2 = O\ \frac{1}{k}\ r^2_{p:} \ _2$$

For the case when $jtj \quad s_{k;n;} + t_{k;n;}$, dierentiate normal cdf twice still leads to very small probability, thus for each x_0, we have

$$\int tE_R \overset{0}{@} q\frac{tk\_\ \ _0 k}{0 s^k_{;n;}(x;R_{k+1})=k}\overset{(x\ )}{2} \ E(R_1=R_{k+1})\ \ a(x^t)\overset{R_{k+1}}{\frac{1}{q\ 2\ s^k_{;n;}(x;R_{k+1})=k}}\ _2 \ _0 \overset{1}{A}\ dt$$

$$= \int t\overset{}{@}\ q\ \frac{tk\_(x_0)k}{s^2_{k;n;}(x;R_{k+1})=k} \quad \frac{E(R_1=R_{k+1})\ \ a(x^t_0)R^2}{q\ s^{k2}_{;n;}(x;R_{k+1})=k} \overset{1}{A}\ dt + o(s^2_{k;n;} + t^2_{k;n;}):$$

E.3. Connecting Multiplicative Constants w.r.t . To show Theorem 4.1, we need to work out the multiplicative constants.

For classication, given x, we know that if X follows multi-dimensional uniform distribution with density $1=f(x)$, for some constant c_d that only depends on d,

$$E(R_1=R_{k+1})\ ^2 = \frac{d}{d\quad 2} + o;$$

$$E(R_1=R_{k+1}) \quad = \frac{d}{d} + o;$$

$$E(R_1=R_{k+1})\ \ R^2_1 = E(R_1=R_{k+1})^2\ \ R^2_{k+1} = c_d \quad \frac{k}{nf(x)}^{\frac{2}{d}} \ \frac{d}{d+2} + o:$$

For regression, one more step needed compared with classication is to evaluate

$$kE\ \ ^{"}\frac{(R_1=R_{k+1})\ \ ^2}{(\ \ P^k_{i=1}(R_i=R_{k+1})\ \ )^2}\ E(X)^2 + k^2E^{\#} \qquad a^2(X)E^2\ ^{"}\frac{R^2_1(R_1=R_{k+1})}{P^k_{i=1}(R_i=R_{k+1})}^{\#!} :$$

The sum of ratios $P^k_{i=1}\ (R_i=R_{k+1})$ is hard to evaluated directly in the denominator, hence we use upper bound and lower bound on it. Since d 3 > 0, using non-uniform

Berry-Essen Theorem, given R_{k+1}, we have

$$P \quad \sum_{i=1}^{k} (R_i = R_{k+1}) \qquad kE(R_1=R_{k+1}) \quad > \qquad ! \qquad \frac{p}{kV ar((R_1=R_{k+1}) \ )} \ R_{k+1} \quad !$$

$$\frac{c}{1+ (= \ p \ k)^3}:$$

Therefore, taking $= {}_k kE(R_1=R_{k+1})$,

$$P \quad \sum_{i=1}^{k} (R_i=R_{k+1}) \quad > ({}_k + 1)kE(R_1=R_{k+1}) \ R_{k+1} \quad ! \qquad {}_k \frac{kE(R_1=R_{k+1})}{p \ \sqrt{Var((R_1=R_{k+1}) \ )}} \ R_{k+1} \quad ! \quad + \quad \frac{c}{1+ (\ p \ k_k)^3}:$$

Note that $(R_1=R_{k+1})$ is always larger than 1, hence

$$E \ " \ \frac{(R_1=R_{k+1}) \ ^2}{(P_{i=1}^{k} (R_i=R_{k+1}) \ )^2} \ \#$$

$$E_{R_{k+1}} \quad \frac{E(R_1=R_{k+1}) \ ^2}{(1 \quad {}_k)^2 k^2 E^2(R_1=R_{k+1})}$$

$$+E_{R_{k+1}} \ P \quad \sum_{i=1}^{k} (R_i=R_{k+1}) \ < (1 \quad {}_k)kE(R_1=R_{k+1}) \ R_{k+1} \quad ! \quad \frac{E(R_1=R_{k+1}) \ ^2}{k^2} \ \# \ + \ o$$

$$\frac{1}{k^2(1 \quad {}_k)^2} \frac{E(R_1=R_{k+1}) \ ^2}{E^2(R_1=R_{k+1})} + \frac{E(R_1=R_{k+1}) \ ^2}{k^2} \qquad \frac{p \ {}_k \ p\, kE(R_1=R_{k+1})}{\sqrt{Var((R_1=R_{k+1}) \ )}} \ !$$

$$+ \frac{E(R_1=R_{k+1}) \ ^2}{k^2} \ \frac{c}{1+ (\ p \ k_k)^3} + o;$$

while

$$E \ " \ \frac{(R_1=R_{k+1}) \ ^2}{(P_{i=1}^{k} (R_i=R_{k+1}) \ )^2} \ \# \qquad \frac{1}{k^2(1+ {}_k)^2} \frac{E(R_1=R_{k+1}) \ ^2}{E^2(R_1=R_{k+1})} \qquad \frac{E(R_1=R_{k+1}) \ ^2}{k^2} \ \frac{c}{1+ (\ p \ k_k)^3}$$

$$+o:$$

Hence taking ${}_k$ such that ${}_k$! 0 while ${}_k \ p\, k$! 1 , we have

$$E \ " \ \frac{(R_1=R_{k+1}) \ ^2}{P_{i=1}^{k} (R_i=R_{k+1}) \ )^2} \ \# = \frac{1}{k^2} \frac{(d \quad)^2 \ (}{d(d \quad 2)} + o;$$

and similarly

$$E \ " \ \frac{R_1^2(R_1=R_{k+1})}{P_{i=1}^{k} (R_i=R_{k+1})} \ \# = \frac{1}{k} \quad \frac{k}{nf(x)}^{\frac{2}{d}} \ \frac{d}{d+2} + o:$$

Appendix F. Proof of Theorem 4.4.

The proof is similar with Theorem 1 in [35].

From the definition of CIS, we have

$$
\begin{aligned}
CIS()=2 &= Z\, P(S_{k;n;}(x)\;\; 1=2)(1 \;\; P(S_{k;n;}(x)\;\; 1=2))\,dP(x)\; Z^{R} \\
&= \;\; P(S_{k;n;}(x)\;\; 1=2) \quad 1_{f(x)1=2g}\;\; dP(x) \\
&\quad {}^{R}Z_{R} \;\; P^{2}(S_{k;n;}(x)\;\; 1=2) \quad 1_{f(x)1=2g}\;\; dP(x):
\end{aligned}
$$

Based on the definition of $Z_i(x;R_{k+1})$ in (E.1), the derivation of ${}_{k;n;}(x;R_{k+1})$ and $s_{k;n;}(x;R_{k+1})$, follow the same procedures as in Theorem 4.1, we obtain

$$
\begin{aligned}
&Z\;\; P(S_{k;n;}(x)\;\; 1=2) \quad 1_{f(x)1=2g}dP(x) \\
&Z^{R}Z \\
={}& \;{}_{S}\;{}_{k;n}\;{}^{k;n}\; f(x^{t}_{0})\, E_{R_{k+1}} P\;\; S_{k;n;}(x^{t}_{0}) < 1=2jR_{k+1} \quad 1_{ft<0g}\;\; dtdVol^{d}\;\; {}^{1}(x_0) + o \\
&Z \qquad\qquad Z \\
={}& \;{}_{S}\; E_{R_{k+1}}\;{}_{R}\; f(x_{0}) \begin{matrix} 8 \\ < \\ : \end{matrix} @\, 0\;\; \frac{tk\_(x_0)k}{q\; \overline{s^{2}_{k;n;}=k}} \;\; \frac{E(R_1=R_{k+1})\;\; a(x^{t}_{0})R^{2}_{0}}{q\; \overline{s^{2}_{k;n;}=k}} A\, 1 \; 1\;\; 1_{ft<0g} \begin{matrix} 9 \\ = \\ ; \end{matrix} dtdVol^{d}\;\; {}^{1}(x_{0}) \\
&+o;
\end{aligned}
$$

and similarly,

$$
\begin{aligned}
&Z \\
&Z^{R}Z\,P^{2}_{k;n}(S_{k;n;}\;\; 1=2jR) \quad 1_{f(x)1=2g}dP(x) \\
={}& \qquad f(x^{t}_{0})\, P^{2}\;\; S_{k;n;}(x^{t}_{0}) < 1=2 \quad 1_{ft<0g}\;\; dtdVol^{d}\;\; {}^{1}(x_0) + o \\
&Z^{S}\;\; {}^{k;n}\; Z \\
={}& \;{}_{S}\; E_{R_{k+1}}\;{}_{R} \\
&\qquad f(x_{0}) \begin{matrix} 8 \\ < \\ : \end{matrix} {}^{2}@\, 0\;\; \frac{tk\_(x_0)k}{q\; \overline{{}^{2}\; s_{k;n;}=k}} \;\; \frac{E(R_1=R_{k+1})\;\; a(x^{t}_{0})\;\; {}^{2}_{0}}{q\; \overline{{}^{2}\; s_{k;n;}=k}\; R} A\, 1 \; 1\;\; 1_{ft<0g} \begin{matrix} 9 \\ = \\ ; \end{matrix} dtdVol^{d}\;\; {}^{1}(x_{0}) + o:
\end{aligned}
$$

Adopting Proposition B.1 and the fact that $s_{k;n;}(x;R_{k+1})$ is little changed by x and R_{k+1}, treating and ^{2} as two distribution functions, we have

$$
CIS() = p\frac{B_1}{\overline{\overline{p}}}\frac{1}{k}\,E\; q\,\overline{s^{2}_{k;n;}(X)} + o = p\frac{B_1}{\overline{p}}\frac{1}{k}\, s\,\overline{1 + \frac{2}{d(d\;\;\; 2)}} + o:
$$

Appendix G. Regret under Testing Data Corruption. This section is the proof for Regret under testing data corruption.

Proof.

Z Z k;n;! tk –(x0)k ! kE(1=2 Y1) ! 1ft<0g dtdVold 1(x0)

p kV ar(Y1)

ZS Z k;n;!

= tk –(x0)k

S 0 R 0 1 1

@@ tk_(x0)k qsign(t)!k_(x0)k b(x0)qtk;n(x0)) A 1ft<0g A dtd V old 1(x0) + o

s2 k;n s2 k;n

Z Z 0 0 1 1

= 1/2 S R tk –(x0)k @@ tk_(x0)k +q !k_(x0)k b(x0)tk;nq(x0)) A 1ft<0g A dtd V old 1(x0)

s2 k;n s2 k;n

Z Z

+ 1/2 S R tk –(x0)k

0 0 1 1

@@ tk_(x0)k q !k_(x0)k b(x0)tk;nq(x0)) A 1ft<0g A dtd V old 1(x0)

- s2 k;n s2 k;n

+r5 + o

Z

= B1/4k + 1/2 S k –(x0)k / k_(x0 k2 b(x0)2E2R1(x)2 + !2k_(x0)k2 dVold 1(x0) + r5 + o:

Z Z 0 1

2r5 = 0 2! (t 2!)k –(x0)k@ tk_(x0)k q !k_(x0)k b(x0)tk;nq(x0) A dtd V old 1(x) S0

- 2 sk;n 2 sk;n

Z Z 0 0 1

+ 2! (t + 2!)k –(x0)k@ tk_(x0)k + q !k_(x0)k b(x0)tk;n(x0) q A dtdVold 1(x) S0

2 sk;n 2 sk;n

+o

Z Z ! 0 1

= 2 ! tk –(x0)k@ tk_(x0)k q s2k;n b(x0)tk;n(x0) q s2k;n A dtdVold 1(x) + o S0

Z Z ! 0 1

= 2 0 tk –(x0)k@ tk_(x0)k q s2k;n b(x0)tk;n(x0) q s2k;n A dtdVold 1(x) S0

Z Z 0 0 1

+2 ! tk –(x0)k@ tk_(x0)k q s2k;n b(x0)tk;n(x0) q s2k;n A dtdVold 1(x) + o; S0 and some further calculation reveals that

$$2r_5 = \quad 2 \quad Z \; Z_{!}^{\;0} \quad tk\_(x_0)k@ \overset{0}{} \frac{tk\_(x_0)k}{q \; s^2_{k;n}} \quad \frac{b(x_0)t_{k;n}(x_0)}{q \; s^2_{k;n}} \overset{1}{A} dtdVol^{d \quad 1}(x_0) \quad S$$

$$+2 \quad Z_0 \; Z \quad tk\_(x \;\;)k@_{0} \; \overset{0}{} \frac{tk\_(x_0)k}{q \; s^2_{k;n}} \quad \frac{b(x_0)t_{k;n}(x_0)}{q \; 2} \overset{1}{A} dtdVol^{d \quad 1}(x \;\;) + o \, S_0 \quad !$$
$$s_{k;n}$$

$$= \; 2 \quad Z \; Z_{!}^{\;0} \quad tk\_(x_0)k@ \overset{0}{} \frac{tk\_(x_0)k}{q \; s^2_{k;n}} \quad \frac{b(x_0)t_{k;n}(x_0)}{q \; s^2_{k;n}} \overset{1}{A} \; S$$

$$+@ \overset{0}{} \frac{tk\_(x_0)k}{q \; s^2_{k;n}} + \frac{b(x_0)t_{k;n}(x_0)}{q \; s^2_{k;n}} \overset{1}{A} dtdVol^{d \quad 1}(x \;\;) \quad Z_0$$

$$+!^{\,2}_{\;S} \quad k\_(x_0)kdVol^{d \quad 1}(x_0) + o:$$

Taking gradient on r_5 w.r.t. $t_{k;n}$, the gradient is positive. As a result, when introducing interpolation and xing k, r_5 becomes smaller. ■

Appendix H. Formal Representations for Corollary 4.6. This section is a formal representations for Corollary 4.6.

Corollary H.1. Under the conditions stated in Corollary 4.6, for random perturbation: if $!^3 = o(n \;\; ^{4=(d+3)})$, the following result holds:

$$\text{Regret}(k; n; \,) = \underbrace{\frac{(d \quad)^{\;2}}{d(d \quad 2)} \frac{1}{4k} \; Z_S \; \frac{f(x \;\;)}{k\_(x \;\;)k_0} dVol^{d \quad 1}(x_0)}_{\text{Variance}}$$

$$+ \underbrace{\frac{(d \quad)^2 \quad (d+2)^2}{(d+2 \quad)^2 \quad d^2} \; Z_S \; \frac{f(x \;\;)a(x \;\;)^2}{k(x \;\;)k}}_{\text{Bias}} \; E(\;\; ^2 jX = x \;\;)d \overset{R \qquad V}{} ol^{d \quad 1}(x \;\;)$$

$$+ \underbrace{\frac{1}{2} \; Z_S \; \frac{!^2}{d} k\_(x_0)kdVol^{d \quad 1}(x_0)}_{\text{Corruption}} + \text{Remainder}:$$

The corruption is not related to .

black-box attack: for simplicity, we use instead of e. In this case, if $!^3 = o(n \;\; ^{4=(d+3)})$, the regret becomes

$$\mathrm{Regret}(k;n;\ ) = \underbrace{\frac{(d\quad)^{2}}{d(d\quad 2)}\frac{1}{4k}\int_S \frac{f(x_0)}{k\_(x_0)k} dVol^{d\ \ 1}(x_0)}_{\mathrm{Variance}}$$

$$+\underbrace{\frac{(d\quad)^2}{(d+2\quad)^2}\frac{(d+2)^2}{d^2}\int_S \frac{f(x_0)a(x_0)^2}{k\_(x_0)k} E(R_1^2 j X = x_0) dVol^{d\ \ 1}(x_0)}_{\mathrm{Bias}}$$

$$+\mathrm{Corruption} + \mathrm{Remainder};$$

where Corruption decreases when slightly increases from zero.

Appendix I. Variance-Bias Trade-o in General Weighting Schemes. This section discusses the variance-bias trade-o in general weighting schemes.

Besides OWNN and interpolated-NN, we found that the benet from the variance-bias trade-o exists for a general class of weights (not essential to be interpolated). Similar as for interpolated-NN, when is closed to zero, the increase of variance is approximately a quadratic function in , while bias is linearly reduced.

Corollary I.1. Denote x as $R_1 = R_{k+1}$ and $(x;\ ) : [0;1]\ [0;1)\ !\ [0;1)$ as a function such that $(x;0)\ 1$, and taking $^0(x;\ ) = @=@$. If

$$:= \int_0^1 {}^0(x;0)x^{d+1}dx \int_0^1 x^{d\ \ 1}dx \int_0^1 {}^0(x;0)x^{d\ \ 1}dx \int_0^1 x^{d+1}dx < 0;$$

then when sightly increasing (to cause interpolation / allocating more weight on closer neighbors), it is guaranteed that the overall MSE / Regret will get decreased.

Proof of Corollary I.1. When is chosen that $\mathrm{R}_0^1 (x;\ )^3 x^{d\ \ 1}dx$ is nite, the ratios of variance and bias in weighted-NN using $(x;\ )$ and $(x;0)$ become

$$\frac{\mathrm{R}_0^1 (x;\ )^2 x^{d\ \ 1}dx}{\mathrm{R}_0^1 (x;\ )x^{d\ \ 1}dx}{}^2 \frac{\mathrm{R}_0^1 (x;0)x^{d\ \ 1}dx^{\ 2}}{\mathrm{R}_0^1 (x;0)^2 x^{d\ \ 1}dx};$$

$$\text{and}\ \frac{\mathrm{R}_0^1 (x;\ )x^{d\ \ 1}x^2dx}{\mathrm{R}_0^1 (x;0)x^{d\ \ 1}x^2dx}{}^2 \frac{\mathrm{R}_0^1 (x;0)x^{d\ \ 1}dx}{\mathrm{R}_0^1 (x;\ )x^{d\ \ 1}dx}{}^2 : 0$$

In the context of $(x;\ ) = x$, the above two ratios refer to $s^2_{k;n;}\ = s^2_{k;n;0}$ and $t^2_{k;n;}\ = t^2_{k;n;0}$ in Theorem 4.1 respectively.

For the ratio of variances, its gradient w.r.t is 0 at = 0:

$$\frac{\mathrm{R}_0^1 (x;\ )^2 x^{d\ \ 1}dx}{\mathrm{R}_0^1 (x;\ )x^{d\ \ 1}dx}{}^2 \frac{\mathrm{R}_0^1 x^{d\ \ 1}dx^{\ 2}}{\mathrm{R}_0^1 x^{d\ \ 1}dx} \quad 1 = O(^2):$$

However, for bias, it becomes

$$\frac{\left(\int_0^1 (x;\ )x^2x^{d\ \ 1}dx\right)^2}{\left(\int_0^1 x^2x^{d\ \ 1}dx\right)^2}\ \frac{\left(\int_0^1 (x;0)x^{d\ \ 1}dx\right)^2}{\left(\int_0^1 (x;\ )x^{d\ \ 1}dx\right)^2}\quad 1$$

$$= O(\ ^2) + \frac{2\ \ \int_0^1 x^{d+1}dx\ \ \int_0^1 x^{d\ \ 1}dx}{\left(\int_0^1 x^2x^{d\ \ 1}dx\right)^2\ \ _0\int_0^1 (x;\ )x^{d\ \ 1}dx}\ ;\ _2\,0$$

As a result, the increase of variance is of O(^{2}), and the decrease of bias is a linear function of since < 0.