Nearest Neighbour Based Estimates of Gradients: Sharp Nonasymptotic Bounds and Applications

Guillaume Ausset[†‡] **Stephan Clémençon**[†] **François Portier**[†]
{guillaume.ausset, stephan.clemencon, francois.portier}@telecom-paris.fr
LTCI[†], Télécom Paris[†], Institut Polytechnique de Paris [†], BNP Paribas[‡]

Abstract

Motivated by a wide variety of applications, ranging from stochastic optimization to dimension reduction through variable selection, the problem of estimating gradients accurately is of crucial importance in statistics and learning theory. We consider here the classical regression setup, where a real valued square integrable r.v. Y is to be predicted upon observing a (possibly high dimensional) random vector X by means of a predictive function $f(X)$ as accurately as possible in the mean-squared sense and study a nearest-neighbour-based pointwise estimate of the gradient of the optimal predictive function, the regression function $m(x) = \mathbb{E}[Y \mid X = x]$. Under classical smoothness conditions combined with the assumption that the tails of $Y - m(X)$ are sub-Gaussian, we prove nonasymptotic bounds improving upon those obtained for alternative estimation methods. Beyond the novel theoretical results established, several illustrative numerical experiments have been carried out. The latter provide strong empirical evidence that the estimation method proposed here performs very well for various statistical problems involving gradient estimation, namely dimensionality reduction, stochastic gradient descent optimization and disentanglement quantification.

1 INTRODUCTION

In this paper, we place ourselves in the usual regression setup, one of the flagship predictive problems in statistical learning. Here and throughout, (X, Y) is a pair of random variables defined on the same probability space $(\Omega,\ \mathcal{F},\ \mathbb{P})$ with unknown probability distribution P: the r.v. Y is real valued and square integrable, whereas the (supposedly continuous) random vector X takes its values in $\mathbb{R}^D$, with $D \geq 1$, and models some information *a priori* useful to predict Y. Based on a sample $\mathcal{D}_n = \{(X_1, Y_1),\ \ldots,\ (X_n, Y_n)\}$ of $n \geq 1$ independent copies of the generic pair (X, Y), the goal pursued is to build a Borelian mapping $f : \mathbb{R}^d \to \mathbb{R}$ that produces, on average, a good prediction $f(X)$ of Y. Measuring classically its accuracy by the squared error, the learning task then boils down to finding a predictive function f that is solution of the risk minimization problem $\min_f \mathcal{R}_P(f)$, where

$$\mathcal{R}_P(f) = \mathbb{E}\left[(Y - f(X))^2\right]. \tag{1}$$

Of course, the minimum is attained by the regression function $m(X) = \mathbb{E}[Y \mid X]$, which is unknown, just like Y's conditional distribution given X and the risk (1). The empirical risk minimization (ERM) strategy consists in solving the optimization problem above, except that the unknown distribution P is replaced by an empirical estimate based on the training data D_n, such as the raw empirical distribution $\hat{P}_n = (1/n)\sum_{i \leq n} \delta_{X_i}$ typically, denoting by δ_x the Dirac mass at any point x, and minimization is restricted to a class $\mathcal{F}$ supposed to be rich enough to include a reasonable approximant of m but not too complex (*e.g.* of finite VC dimension) in order to control the fluctuations of the deviations between the empirical and true distributions uniformly over it. Under the assumption that the random variables Y and $f(X)$, $f \in \mathcal{F}$, have sub-Gaussian tails, the analysis of the performance of empirical risk minimizers (*i.e.* predictive functions obtained by least-squares regression) has been the subject of much interest in the literature, see *e.g.* Györfi et al., 2002, Massart, 2007, Boucheron et al., 2013 or Lecué and Mendelson, 2016 (and refer to *e.g.* Lugosi and Mendelson, 2016 for alternatives to the ERM approach in non sub-Gaussian situations).

Proceedings of the 24[th] International Conference on Artificial Intelligence and Statistics (AISTATS) 2021, San Diego, California, USA. PMLR: Volume 130.

In this paper, we are interested in estimating accurately the (supposedly well-defined) gradient $\nabla m(x)$ by means of the popular k nearest neighbour (k-NN) approach, see *e.g.* Chapter in Devroye et al., 1996 or Biau and Devroye, 2015. The *gradient learning* issue has received increasing attention in the context of local learning problems such as classification or regression these last few years, see *e.g.* Mukherjee and Wu, 2006; Mukherjee and D.-X. Zhou, 2006. Because it provides a valuable information about the local structure of a dataset in a high-dimensional space, an accurate estimator of the gradient of a predictive function can be used for various purposes such as dimensionality reduction or variable selection (see *e.g.* Dalalyan et al., 2008; Hristache, Juditsky, Polzehl, et al., 2001; Hristache, Juditsky, and Spokoiny, 1998; Trivedi et al., 2014; Xia, 2007; Xia et al., 2002; Ye and Xie, 2012), the partial derivative w.r.t. a given variable being a natural indicator of its importance regarding prediction. The previous references are all concerned with outer products of gradients so as to recover some dimension-reduction subspace. Estimators of the gradients have also been proposed for zeroth-order optimization (see *e.g.* Berahas et al., 2020; Nesterov and Spokoiny, 2017; Wang et al., 2018) and can benefit from good convergence properties.

Whereas the use of standard nonparametric methods for gradient estimation is documented in the literature (see De Brabanter et al., 2013; Delecroix and Rosa, 1996; Fan and Gijbels, 1996 for the use of local polynomial with kernel smoothing techniques, Gasser and Müller, 1984 for the so-called Gasser-Muller alternative, S. Zhou and Wolfe, 2000 for the use of regression spline and Mukherjee and D.-X. Zhou, 2006 for the estimation on a reproducing kernel Hilbert space with kernel smoothing), it is the purpose of the present article to investigate the performance of an alternative local averaging method, the popular k-NN method. As it provides piecewise constant estimates, it is easier to conceptualize for the practitioner and, more importantly; the neighbourhoods determined by the parameter k are data-driven and often more consistent than those defined by the bandwidth in the kernel setting, especially in high dimensions.

Here we investigate the behaviour of the estimator of the (supposedly sparse) gradient of the regression function at a given point $x \in \mathbb{R}^D$, obtained by solving a regularized local linear version of the k-NN problem with a Lasso penalty. Precisely, nonasymptotic bounds for the related estimation error are established. Whereas k-NN estimators of the regression function have been extensively analysed (see *e.g.* Biau, Cérou, et al., 2010; Jiang, 2019; Kpotufe, 2011 and the references therein), the result stated in this paper is the first of this type to the best of our knowledge.

The relevance of the approach promoted is then illustrated by several applications. A variable selection algorithm that exploits the local nature of the gradient estimator proposed is first exploited to refine the popular random forest algorithm (see Breiman, 2001): by exploiting the node estimate of the gradient we are able to direct better the choice of cuts. Very simple to implement and accurate, as supported by the various numerical experiments carried out, it offers an attractive and flexible alternative to existing traditional methods such as PCA or the more closely related method of Dalalyan et al., 2008, allowing for a local reduction of the dimension rather than implementing a global preprocessing of the data. We next show how a rough statistical estimate of the gradient of any smooth objective function based on the estimation principle previously analysed in the context of regression can be exploited in a basic gradient descent algorithm. We exploit the local structure of the algorithm to be able to reuse past computations in order to calculate our estimator and jump to a better local minimum at each gradient step as well. Finally, we give an example of the usefulness of a sparse gradient estimate when the gradient is believed to be truly sparse: we use our estimator to retrieve the direction of interest for a specific attribute inside a disentangled representation and show how this can be used as an *ad hoc* measure of disentanglement.

The article is organized as follows. In section 2, the estimation method and the assumptions involved in the subsequent analysis are listed. The main theoretical results of the paper are stated in section 3, while several applications of the estimation method promoted are described at length and illustrated by numerical experiments in section 4. Some concluding remarks are collected in section 5 and technical proofs are postponed to the Supplementary Material.

2 BACKGROUND - FRAMEWORK

We place ourselves in the nonparametric regression setup described in the previous section. Here and throughout, the indicator function of any event $\mathcal{E}$ is denoted by $\mathbb{1}_{\mathcal{E}}$, the cardinality of any finite set E by $\#E$. By $||x||_\infty = \max\{|x_1|, \ldots |x_D|\}$, $||x|| = |x_1| + \ldots + |x_D|$ and $||x|| = \sqrt{x_1^2 + \ldots + x_D^2}$ are meant the ℓ_∞-norm, the ℓ_1-norm and the ℓ_2-norm of any vector $x = (x_1, \ldots, x_D)$ in $\mathbb{R}^D$. Any vector x in $\mathbb{R}^D$ is identified as a column vector, the transpose of any matrix M is denoted by $M^\intercal$ and $\mathcal{B}(x, \tau) = \{z \in \mathbb{R}^D : ||x - z||_\infty \leq \tau\}$ is the (closed) ball of centre $x \in \mathbb{R}^D$ and radius $\tau > 0$.

k-NN estimation methods in regression. Let $x \in$

$\mathbb{R}^D$ be fixed and $k \in \{1, \ldots, n\}$. Define

$$\hat{\tau}_k(x) = \inf\{\tau \geq 0 : \sum_{i=1}^{n} \mathbb{1}_{\{X_i \in \mathcal{B}(x,\tau)\}} \geq k\},$$

which quantity is referred to as the k-NN radius. Indeed, observe that, equipped with this notation, $\mathcal{B}(x, \hat{\tau}_k(x))$ is the smallest ball with centre x containing k points of the sample $\mathcal{D}_n$ and the mapping $\alpha \in (0, 1] \mapsto \hat{\tau}_{\alpha n}(x)$ is the empirical quantile function related to the sample $\{\|x - X_1\|_\infty, \ldots, \|x - X_n\|_\infty\}$. The rationale behind k-NN estimation in the regression context is simplistic, the method consisting in approximating $m(x) = \mathbb{E}[Y \mid X = x]$ by $\mathbb{E}[Y \mid X \in \mathcal{B}(x, \tau)]$, the mapping m being assumed to be smooth at x, and computing next the empirical version of the approximant (*i.e.* replacing the unknown distribution P by the raw empirical distribution). This yields the estimator

$$\hat{m}_k(x) = \frac{1}{k} \sum_{i:X_i \in \mathcal{B}(x,\hat{\tau}_k(x))} Y_i, \tag{2}$$

usually referred to as the standard k-nearest neighbour predictor at x. Of course, the mapping $x \in \mathbb{R}^D \mapsto \hat{m}_k(x)$ is locally/piecewise constant, just like $x \in \mathbb{R}^D \mapsto \hat{\tau}_k(x)$. The local average $\hat{m}_k(x)$ can also be naturally expressed as

$$\hat{m}_k(x) = \arg\min_{m \in \mathbb{R}} \sum_{i:X_i \in \mathcal{B}(x,\hat{\tau}_k(x))} (Y_i - m)^2. \tag{3}$$

For this reason, the estimator (2) is sometimes referred to as the *local constant* estimator in the statistical literature. Following in the footsteps of the approach proposed in Fan, 1992, the estimation of the regression function at x can be refined by approximating the supposedly smooth function $m(z)$ around x in a linear fashion, rather than by a local constant m, since we have $m(z) = m(x) + \nabla m(x)^\intercal (z - x) + o(\|z - x\|)$ by virtue of a first-order Taylor expansion. For any point X_i close to x, one may write $m(X_i) \simeq m + \beta^\intercal (X_i - x)$ and the *local linear* estimator of $m(x)$ and the related estimator of the gradient $\beta(x) = \nabla m(x)$ are then defined as

$$\arg\min_{(m,\beta) \in \mathbb{R}^{D+1}} \sum_{i:X_i \in \mathcal{B}(x,\hat{\tau}_k(x))} (Y_i - m - \beta^\intercal (X_i - x))^2. \tag{4}$$

Because of its reduced bias, the local linear estimator (the first argument of the solution of the optimization problem above) can improve upon the local constant estimator (2) in moderate dimensions. However, when the dimension D increases, its variance becomes large and the design matrix of the regression problem is likely to have small eigenvalues, causing numerical difficulties. For this reason, we introduce here a lasso-type regularized version of (4), namely

$$(\tilde{m}_k(x), \tilde{\beta}_k(x)) \in \arg\min_{(m,\beta) \in \mathbb{R}^{D+1}} \sum_{i:X_i \in \mathcal{B}(x,\hat{\tau}_k(x))} (Y_i - m - \beta^\intercal (X_i - x))^2 + \lambda \|\beta\|_1 \tag{5}$$

for $i \in i(x) = \{j : X_j \in \mathcal{B}(x, \hat{\tau}_k(x))\}$ and where $\lambda > 0$ is a tuning parameter governing the amount of ℓ_1-complexity penalization. For the moment, we let it be a free parameter and will propose a specific choice in the next section. Focus is here on the gradient estimator $\tilde{\beta}_k(x)$, *i.e.* the second argument in (5). In the subsequent analysis, nonasymptotic bounds are established for specific choices of λ and k. The following technical assumptions are required.

Technical hypotheses. The hypothesis formulated below permits us to relate the volumes of the balls $\mathcal{B}(x, \tau)$ to their probability masses, for τ small enough.

Assumption 1 *There exists $\tau_0 > 0$ such that restriction of X's distribution on $\mathcal{B}(x, \tau_0)$ has a bounded density f_X, bounded away from zero, with respect to Lebesgue measure:*

$$b_f = \inf_{y \in B(x,\tau_0)} f_X(y) > 0$$
$$U_f = \sup_{y \in B(x,\tau_0)} f_X(y) < +\infty.$$

Suppose in addition that $U_f / b_f \leq 2$.

The constant 2 involved in the condition above for notational simplicity can be naturally replaced by any constant $1+\gamma$, with $\gamma > 0$. The next assumption, useful to control the variance term, is classical in regression, it stipulates that we have $Y = m(X) + \varepsilon$, with a sub-Gaussian residual ε independent from X.

Assumption 2 *The zero-mean and square integrable r.v. $\varepsilon = Y - m(X)$ is independent from X and is sub-Gaussian with parameter $\sigma^2 > 0$, i.e. $\forall \lambda \in \mathbb{R}$, $\mathbb{E}[\exp(\lambda \varepsilon)] \leq \exp(-\sigma^2 \lambda^2 / 2)$.*

In order to control the bias error when estimating the gradient $\beta(z) = \nabla m(z)$ of the regression function at x, smoothness conditions are naturally required.

Assumption 3 *The function $m(z)$ is differentiable on $\mathcal{B}(x, \tau_0)$ with gradient $\beta(z) = \nabla m(z)$ and there exists $L_2 > 0$ such that for all $z \in \mathcal{B}(x, \tau_0)$,*

$$|m(z) - m(x) - \beta(x)(z - x)| \leq L_2 \|z - x\|_\infty^2.$$

Finally, a Lipschitz regularity condition is required for the density f_X.

Assumption 4 *The function f_X is L-Lipschitz at x on $\mathcal{B}(x,\tau_0)$, i.e. there exists $L>0$ such that for all $z\in B(x,\tau_0)$,*

$$|f_X(z)-f_X(x)|\leq L\|z-x\|_\infty.$$

We point out that, as the goal of this paper is to give the main ideas underlying the use of the k-NN methodology for gradient estimation rather than carrying out a fully general analysis, the ℓ_∞-norm is considered here, making the study of ℓ_1 regularization easier. The results of this paper can be extended to other norms at the price of additional work.

3 MAIN RESULT - THE k-NN BASED GRADIENT ESTIMATOR

The main theoretical result of the present paper is now stated and further discussed. Under the hypotheses listed in the previous section and for specific choices of k and λ, it provides a nonasymptotic bound for the estimator $\tilde{\beta}_k(x)$ of the gradient $\beta(x)=\nabla m(x)$ at x given by (5). Whereas nonasymptotic bounds for k-NN estimators of the regression function have been established under various smoothness assumptions (see *e.g.* Jiang, 2019 or Kpotufe, 2011), no nonasymptotic study of k-NN based estimator of the gradient of the regression function is documented in the literature. To the best of our knowledge, the result proved in this article is the first of this nature. Two key quantities are involved in the upper confidence bound given in Theorem 1, the (deterministic) radius

$$\overline{\tau}_k=\left(\frac{2k}{nb_f2^D}\right)^{1/D},$$

that upper bounds the k-NN radius on an event holding true with high probability, as well as the cardinality of the so-called local active set

$$\mathcal{S}_x=\{1\leq k\leq D\,:\,\beta_k(x)\neq 0\},$$

which, for clarity reasons, is supposed to be non-empty.

Theorem 1 *Suppose that assumptions 1, 2, 3 and 4 are fulfilled. Let $n\geq 1$ and $k\geq 1$ such that $\overline{\tau}_k\leq\tau_0$. Let $\delta\in(0,1)$ and set $\lambda=\overline{\tau}_k(\sqrt{2\sigma^2\log(16D/\delta)/k}+L_2\overline{\tau}_k^2)$. Then, we have with probability larger than $1-\delta$,*

$$\|\tilde{\beta}_k(x)-\beta(x)\|_2\leq (24)^2\sqrt{\#\mathcal{S}_x}\left(\overline{\tau}_k^{-1}\sqrt{\frac{2\sigma^2\log(16D/\delta)}{k}}+L_2\overline{\tau}_k\right), \quad (6)$$

as soon as $C_1\#\mathcal{S}_x\log(Dn/\delta)\leq k\leq C_2n$, $\overline{\tau}_k^2\leq(b_f^2/(C_3\#\mathcal{S}_xL^2)\wedge\tau_0^2)$, where C_1, C_2 and C_3 are universal constants.

The analysis of the accuracy of the nearest neighbour estimate $\hat{m}_k(x)$ classically involves the following decomposition of the estimation error

$$\hat{m}_k(x)-m(x)=(\hat{m}_k(x)-m_k(x))+(m_k(x)-m(x)), \quad (7)$$

where $m_k(x)=(1/k)\sum_{i:X_i\in\mathcal{B}(x,\hat{\tau}_k(x))}m(X_i)$. The approach developed in Jiang, 2019 essentially consists in combining this decomposition with the fact that $\hat{\tau}_k(x)\leq\overline{\tau}_k$ with high probability. By its own nature, our local linear Lasso regularized estimate of the gradient $\tilde{\beta}_k$ cannot be treated in the same way. First, in order to take advantage of the Lasso regularization in sparse situations (*i.e.* when the gradient at x depends on a small number of covariates solely), we rely on a basic inequality Hastie et al., 2015, Lemma 11.1 which is useful when analysing standard Lasso estimates. Second, we need to control the size of the neighbourhoods $\hat{\tau}_k(x)$ on an event of high probability. In this respect, we slightly deviate from the approach of Jiang, 2019: we do not rely on concentration results over VC classes but only on the Chernoff concentration bound. This way, we can relax significantly the lower bound conditions for k as the dimension D increases, see Theorem 2 below, which compares favourably with Corollary 1 in Jiang, 2019 for instance.

Balancing between the bias and the variance term of the upper bound provided in (6) we obtain that the optimal value for k is $k\sim n^{4/(4+D)}$. In this case, the bound stated above yields the rate $n^{-1/(4+D)}$. As a consequence, our bound matches the minimax rate (up to log terms) given in Stone, 1982 for the problem of the estimation of the derivative (in a L_2 sense).

Pointwise k-NN estimation of $m(x)$. Though it concerns the local estimation error, the bound in the theorem below can be viewed as a refinement of the nonasymptotic results recently established in Jiang, 2019 (see also Kpotufe, 2011), which provide uniform bounds in x. It requires a local smoothness condition for the regression function. From now on, $\|\cdot\|$ denotes any norm on $\mathbb{R}^D$.

Assumption 5 *The regression function $m(z)$ is L_1-Lipschitz at x, i.e. there exists $L_1>0$ such that for all $z\in\mathcal{B}(x,\tau_0)=\{x'\in\mathbb{R}^D\,:\,\|x'-x\|\leq\tau_0\}$,*

$$|m(x)-m(z)|\leq L_1\|x-z\|.$$

Theorem 2 *Suppose that assumptions 1, 2 and 5 are fulfilled and that $2k\leq n\tau_0b_fV_D$. Then for any $\delta\in(0,1)$ such that $k\geq 4\log(2n/\delta)$, we have with probability $1-\delta$:*

$$|\hat{m}_k(x)-m(x)|\leq\sqrt{\frac{2\sigma^2\log(4/\delta)}{k}}+L_1\left(\frac{2k}{nb_fV_D}\right)^{1/D},$$

where $V_D=\int\mathbb{1}_{\{x\in\mathcal{B}(0,1)\}}dx$ denotes the volume of the unit ball.

We obtain a weaker condition on the value of k than that obtained in Jiang, 2019 (see Corollary 1 therein), due to our different treatment of the approximation term (the second term in decomposition (7)) is different (see the argument detailed in the Supplementary Material). With $k \sim n^{2/(2+D)}$, the bound stated above yields the minimax rate $n^{-1/(D+2)}$.

4 NUMERICAL EXPERIMENTS

In order to motivate the need for a robust estimator of the gradient, we introduce three different examples of use of our estimator compared to existing approaches. All the code to reproduce the experiments and figures can be found at `https://git.sr.ht/~aussetg/locallinear`.

As our estimator is sensitive to the choice of hyperparameters k and λ we use a local leave-one-out procedure described in Algorithm 1 for hyperparameter selection. As only the regression variable Y is observed, the regression error is used as a proxy loss in the cross-validation. The high cost of k-NN is amortized by using k-d trees, bringing the total average complexity of the nearest neighbour search down to $O(n \log n)$. In cases where the aforementioned cost is too high (n in the order of millions) it is possible to instead make use of approximate nearest neighbour schemes such as HNSW (Malkov and Yashunin, 2020). Approximate Nearest Neighbours algorithms have recently enjoyed a regain of interest and provide high accuracy at a very low computational cost (Aumüller et al., 2018).

Algorithm 1 Local Leave-One-Out

Require: x: sample point, (X, Y): training set, (K, Λ): grid
1: $X_{\text{LoO}} \leftarrow$ `Neighbourhood of` x `in` X `of size` N
2: **for** $k \in K, \lambda \in \Lambda$ **do**
3: **for** $X_i \in X_{\text{LoO}}$ **do**
4: $m_i, \beta_i \leftarrow$ `estimated gradient at` X_i `w.r.t` X, Y `using` (5)
5: **end for**
6: $\texttt{error}_{k,\lambda} \leftarrow \frac{1}{N} \sum_{i=1}^{N} (m_i - Y_i)^2$
7: **end for**
8: $k^\star, \lambda^\star \leftarrow \arg\min_{k,\lambda} \texttt{error}_{k,\lambda}$
9: **return** $k^\star, \lambda^\star$

4.1 Variable Selection

While a large number of observations is desirable the same is not necessarily the case for the individual features; a large number of features can be detrimental to the computational performance of most learning methods but also harmful to the predictive performance. In order to mitigate the detrimental impact of the high dimensionality, or *curse of dimensionality*, one can try to reduce the effective dimension of the problem. A large body of work exists on dimensionality reduction as a preprocessing step that considers the intrinsic dimensionality of X by considering for example that X lies on a lower-dimensional manifold. Those approaches only consider X in isolation and do not take into account Y which is the variable of interest. It is possible to use the information in Y to direct the dimension reduction of X, either by treating Y as side information, as is done in Bach and Jordan, 2005, or by considering the existence of an explicit *index space* such that $Y_i = g(v_1^\intercal X_i, \cdots, v_m^\intercal X_i) + \varepsilon_i$ as is done in Dalalyan et al., 2008. In the latter case, it is possible to observe that the *index space* lies on the subspace spanned by the gradient.

In contrast with the work of Dalalyan et al., 2008 our approach is local and it is therefore possible to retrieve a different subspace in different regions of $\mathbb{R}^D$. As localizing the estimator increases its variance, we choose to only identify the dimensions of interest instead of estimating the full projection matrix. We introduce *Gradient Guided Trees* in Algorithm 2 to exploit the local aspect of our estimator in order to direct the cuts in a random tree: at each step, cuts are drawn randomly with probability proportional to estimated mean absolute gradient in the cell. We demonstrate the improvements brought by guiding the cuts by the local information provided by the gradient by comparing the performance of a vanilla regression random forest with the same procedure but with local gradient information. We consider five datasets: the Breast Cancer Wisconsin (Diagnostic) Data Set introduced in Street et al., 1993; the Heart Disease dataset introduced by Detrano et al., 1989; the classic Diamonds Price dataset; the Gasoline NIR dataset introduced by Kalivas, 1997 and the Sloan Digital Sky Survey DR14 dataset of Abolfathi et al., 2018. We measure the L^2 loss by cross validation across 50 folds using the same hyperparameters for the growing of the forest in both the standard and gradient guided variants.

Algorithm 2 Node Splitting for Gradient Guided Trees

Require: (X, Y): training set, `Node`: indexes of points in the node
1: $\nabla m(X_i) \leftarrow$ `estimated gradient at` $X_i, \forall i \in$ `Node using` (5)
2: $\omega \leftarrow \sum_{i \in \texttt{Node}} |\nabla m(X_i)|$
3: $K \leftarrow$ `sample` $\sqrt{D}$ `dimensions in` $\{1, \ldots, d\}$ `with probability weights` $\propto \omega$
4: $k, c \leftarrow$ `best threshold` c `and dimension` k
5: **return** k, c

	Description		Loss	
Dataset	n	D	RF	GGF
Wisconsin	569	30	0.0352 $\pm 3.29 \cdot 10^{-4}$	$\mathbf{0.0345}$ $\pm 3.35 \cdot 10^{-4}$
Heart	303	13	0.128 $\pm 6.6 \cdot 10^{-4}$	$\mathbf{0.124}$ $\pm 8.6 \cdot 10^{-4}$
Diamonds	53940	23	680033 $\pm 3.45 \cdot 10^{9}$	$\mathbf{664265}$ $\pm 2.81 \cdot 10^{9}$
Gasoline	60	401	0.678 ± 0.451	$\mathbf{0.512}$ ± 0.347
SDSS	10000	8	$0.872 \cdot 10^{-3}$ $\pm 4.50 \cdot 10^{-6}$	$\mathbf{0.776} \cdot 10^{-3}$ $\pm 6.00 \cdot 10^{-6}$

Table 1: Performance of the two random forest algorithms on a 50-folds cross validation.

We denote by RF, Random Forests grown from standard CART trees while GGF denote *Gradient Guided Forests* grown from the *Gradient Guided Trees* previously introduced. As seen in Table 1, gradient guided split sampling consistently outperform the vanilla variant. When all variables are relevant, as is the case when the variables were carefully selected by the practitioner with prior knowledge, our variant performs similarly to the original algorithm while performance is greatly improved when only a few variables are relevant, such as in the NIR dataset (Portier and Delyon, 2014).

4.2 Gradient Free Optimization

Many of the recent advances in the field of machine learning have been made possible in one way or another by advances in optimization; both in how well we are able to optimize complex function and what type of functions we are able to optimize if only locally. Recent advances in automatic differentiation as well as advances that push the notion of *what* can be differentiated have given rise to the notion of *differentiable programming* (Innes et al., 2019) in which a significant body of work can be expressed as the solution to a minimization problem usually then solved by gradient descent.

We study here the use of the local linear estimator of the gradient in Algorithm 3 in cases where analytic or automatic differentiation is impossible, and compare it to a standard gradient free optimization technique as well as the oracle where the true gradient is known. While line 1 bears resemblance with Gaussian smoothing and could therefore be seen as analogous to gradient estimation via Gaussian smoothing (see Berahas et al., 2020), two key differences here are the subsequent local linear step as well as the fact that the samples from line 1 are not necessarily the samples used in the local linear estimator of line 4.

Algorithm 3 Estimated Gradient Descent

Require: x_0: initial guess, f: function $\mathbb{R}^D \to \mathbb{R}$, M: budget

1: $X \leftarrow X_1, \dots, X_M$ `with` $X_i \sim \mathcal{N}(x_0, \varepsilon \times I_D)$
2: $Y \leftarrow f(X) := f(X_1), \dots, f(X_M)$
3: **while** `not StoppingCondition` **do**
4: $\quad m, \Delta \leftarrow$ `estimated gradient at` x `w.r.t` X, Y `using` (5)
5: $\quad X \leftarrow X, X_1, \dots, X_M$ `with` $X_i \sim \mathcal{N}(\texttt{GradientStep}(x, \Delta), \varepsilon \times I_D)$
6: $\quad Y \leftarrow f(X)$
7: $\quad x \leftarrow \arg\min_{X_i}\{f(X_i)\}$
8: **end while**
9: **return** x

We first minimize the standard but challenging Rosenbrock function for different values of d. which is defined as

$$f(x) = 100 \sum_{i=1}^{d-1} (x_{i+1} - x_i)^2 + (x_i - 1)^2. \qquad (8)$$

We compare for reference our approach to the Nelder-Mead (simplex search) algorithm; a standard gradient free optimization technique. It is apparent in Figure 1 that estimating the gradient yields a significant advantage compared to traditional gradient-free techniques that usually have to rely on bounding arguments and feasible regions and therefore scale unfavourably with the dimension. As our approach uses a nearest neighbours formulation for the gradient estimate, we are able to efficiently reuse past samples in the current estimate of the gradient; this makes it possible to achieve a sufficiently accurate estimate of the gradient even in high dimensions. We compare in Figure 2 the approach developed previously to the estimators proposed by Wang et al., 2018 and Fan, 1992. As the approach proposed by Wang et al., 2018 includes the use of *mirror descent*, for fairness, we have implemented our proposed gradient descent algorithm of Algorithm 3 using our estimator as well as those of Wang et al., 2018 and Fan, 1993 (with reuse of previous samples where appropriate) for the gradient. We then reimplemented the mirror descent algorithm of Wang et al., 2018 with the previous estimators of the gradient. We observe in Figure 2 that our method compares favourably: our estimator is able to reuse past samples in its gradient estimation and has therefore access to a better gradient estimate for a

[1] The number of function evaluations does not have any meaning for the true gradient. We use here that 1 estimated gradient step $\approx$ 50 function evaluations. 5000 function evaluations therefore equate to 100 gradient steps.

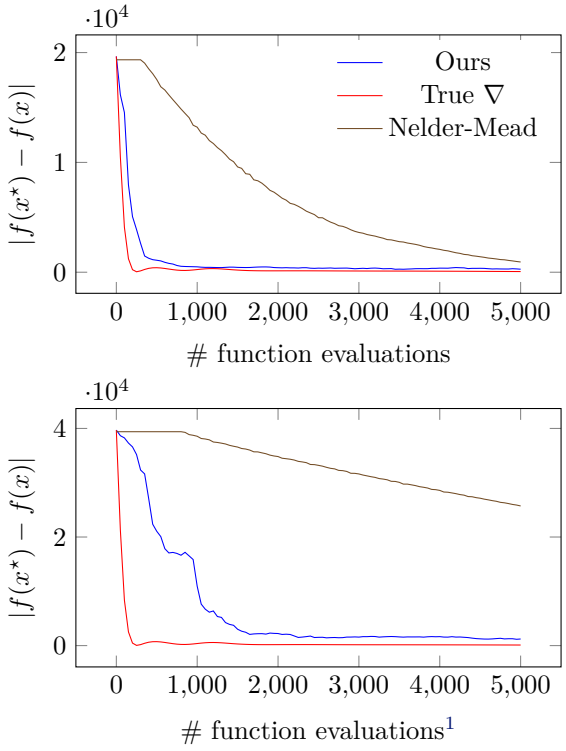

Figure 1: Nesterov Gradient Descent on the rosenbrock function for $d = 50$ (top) and $d = 100$ (bottom).

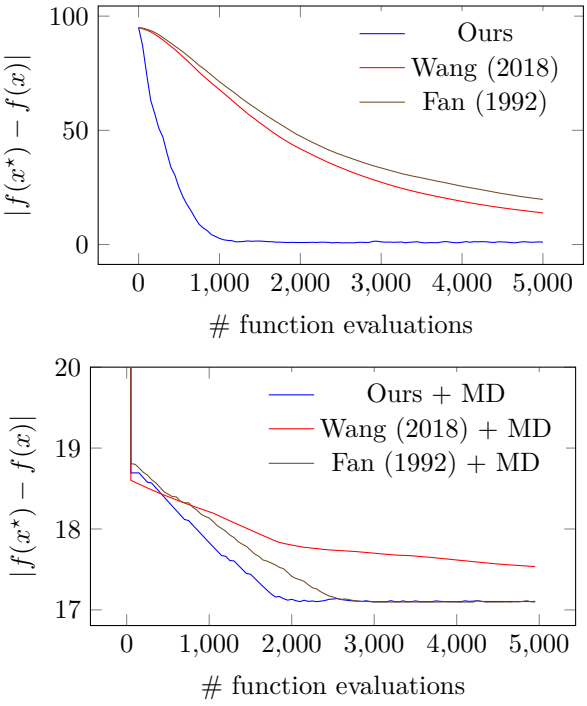

Figure 2: Nesterov Gradient Descent (top) and Mirror Gradient Descent (bottom) on the Rosenbrock function for $d = 100$.

fixed, given number of function evaluations. We apply the previous method to the minimization of the log-likelihood of a logistic model on the UCI's Adult data set, consisting of 48842 observations and 14 attributes amounting to 101 dimensions once one-hot encoded and an intercept added.

$$\mathcal{L}_\theta(X) = -\sum_i Y_i \log(1 + \exp(-\theta X_i)) - (1 - Y_i)\log(1 + \exp(\theta X_i)),\ \theta \in \mathbb{R}^{101}. \quad (9)$$

We also compare the effective CPU wall time needed to reach a given log-likelihood in order to give a more comprehensive view of the relative performance of the multiple algorithms. Given that the time per iteration can vary greatly depending on the cost of evaluations and the cost of the gradient procedures, it is important to use both the number of evaluations and the time metric jointly with the former being more relevant as the cost of individual function evaluations increases.

4.3 Disentanglement

Disentangled representation learning aims to learn a representation of the input space such that the independent dimensions of the representation each encode separate but meaningful attributes of the original fea-

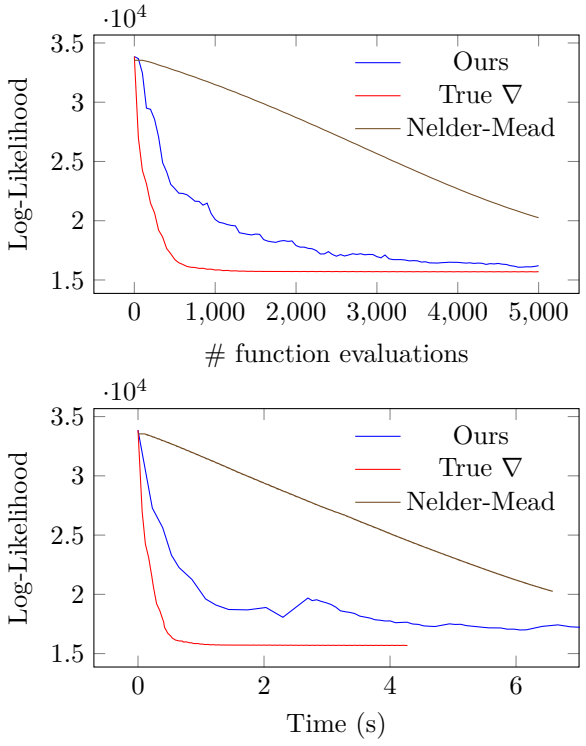

Figure 3: Log-likelihood of the logistic regression on a test set, trained by Nesterov Gradient Descent with respect to the number of evaluations (top) and time (bottom).

ture space. If the space of interest is the space of *faces*, a disentangled representation would then for example be a lower-dimensional space where one dimension encodes the sex of the subject, another its age, and so forth. We show here how our estimator can be useful for retrieving the dimensions associated with a concept in a supervised manner.

A β-VAE (Higgins et al., 2017) model is trained on the `CACD2000` dataset of celebrity faces with age labels to first build low-dimensional a representation of the images and then extract the direction relating to age. We learn $\mathcal{E}_\phi$ and $\mathcal{D}_\theta$ parameterizing q_ϕ and p_θ, to minimize the loss

$$\mathcal{L}(\theta, \phi; x, z, \beta) = \mathbb{E}_{q_\phi(z|x)}\left[\log p_\theta(x \mid z)\right] - \beta D_{KL}(q_\phi x \parallel x), \quad (10)$$

where β acts as a constraint on the representational power of the latent distribution; $\beta = 1$ leads to the standard VAE formulation of Kingma and Welling, 2014 while $\beta > 1$ increases the level of disentanglement. We use a standard symmetrical encoder-decoder architecture for the variational autoencoder, schematically presented in Figure 4. All the relevant implementation details can be found in the `Julia` code in the supplementary materials. We learn a 512-dimensional representation of the 128×128 images and encode all the `CACD2000` images. Once all the images have been encoded in $\mathbb{R}^{512}$ it is possible to use the local linear estimator of the gradient studied in this work to derive the gradient of the age with respect to the latent variable, making it possible to produce a new version of the input image that appears either older or younger as done in Figure 5. By computing a local estimate of the gradient, we are able to derive a more meaningful change when the age is not perfectly disentangled.

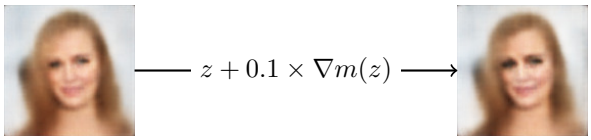

Figure 5: Extracting the direction of interest for aging.

Note that the quality of the image reconstruction and generation is here solely limited by the choice of the encoding and decoding model and is not related to the methods introduced in this paper, significant advances in the quality of the decoding have been made in the recent years and if a better quality and less blurry decoded output is desired we encourage the reader to replace the decoder with a `PixelCNN` architecture such as presented in Salimans et al., 2017. The quality of the gradient is also significantly impacted by the quality of the annotations as `CACD200` is an automatically annotated and noisy dataset.

Using our estimator it is possible to estimate the gradient ∇m of $\mathbb{E}[Y \mid Z = z]$ with respect to the latent variable Z (illustrated in the Appendix). It is then possible to analyse the sparsity of ∇m to quantify the quality of the disentanglement for varying level of β by quantifying how far from a single dimension the gradient for the age is concentrated. As the true dimension is unknown, we instead measure the angular distance to all dimensions reweighted by the magnitudes of the partial derivatives:

$$\sum_i \frac{|\hat{\nabla}_i m(x)|}{|\hat{\nabla} m(x)|} \cos(e_i, \frac{1}{n}\sum_k |\hat{\nabla} m(x)|), \quad \text{where} \quad \cos(a, b) = \frac{a \cdot b}{\|a\|\|b\|}. \quad (11)$$

We observe in Figure 6 that as β increases the age slowly become disentangled, as expected if one considers the age to be an important and independent characteristic of human faces.

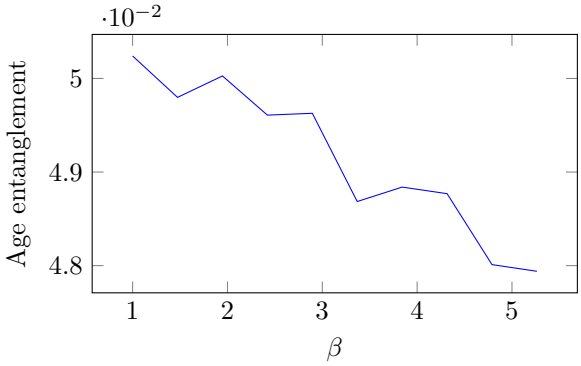

Figure 6: Quality of disentanglement with respect to the age

While not an entirely adequate metric for disentanglement, not only because disentanglement does not necessarily require the dimensions to be the one an observer expected but more importantly because this metric requires an annotated dataset; we believe this metric can be useful for practitioners. By measuring how close the estimated gradients are to the axis, with respect to an annotated dataset of characteristics of interest, a practitioner can ensure his model is sufficiently disentangled for downstream tasks such as face manipulation by a user. We also believe it is possible to design an end-to-end differentiable framework in order to force disentanglement to consider the characteristics of interest: our estimator is the solution to a convex optimization program and as such admits an adjoint; it is therefore possible to fit a local linear estimator

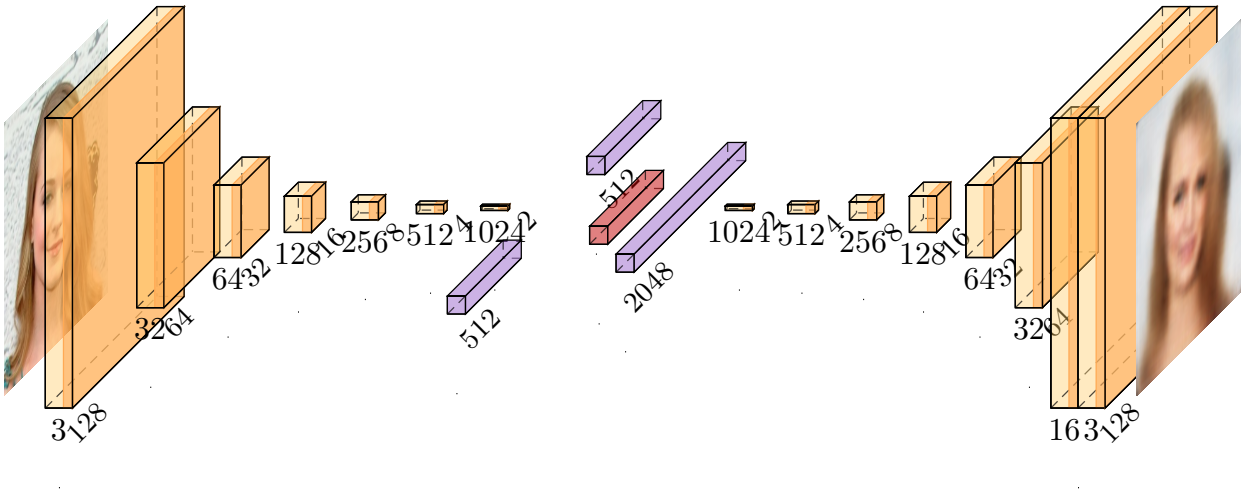

Figure 4: Encoder-Decoder Architecture used for this work

inside an automatic differentiation framework such as done in Agrawal et al., 2019.

5 CONCLUSION

In this paper, we have studied the estimator of the (supposedly sparse) gradient of the regression function obtained by solving a regularized local linear version of the k-NN problem with a ℓ_1 penalty. Nonasymptotic bounds for the local estimation error have been established, improving upon those obtained for alternative methods in sparse situations. We derived nonasymptotic error bounds on a local linear estimator of the gradient based on k-nearest neighbours averaging and with a sparsity inducing L^1 penalty. Compared to previous similar estimators we show that exploiting the sparsity of the gradient improves convergence rates. Beyond its theoretical properties and its computational simplicity, the local estimation method promoted here is shown to be the key ingredient for designing efficient algorithms for variable selection and M-estimation, as supported by various numerical experiments. Hopefully, this work shall pave the way to the elaboration of novel statistical learning procedures that exploits the local structure of the gradient, and for which the theory will be extended to take into account the underlying geometry of the space in order to obtain convergence rates depending only on the true intrinsic dimension of the data such as done in Mukherjee, Wu, and D.-X. Zhou, 2010 for the kernel smoothing setting.

References

Abolfathi, B. et al. (Apr. 19, 2018). "The Fourteenth Data Release of the Sloan Digital Sky Survey: First Spectroscopic Data from the Extended Baryon Oscillation Spectroscopic Survey and from the Second Phase of the Apache Point Observatory Galactic Evolution Experiment". In: *The Astrophysical Journal Supplement Series* 235.2, p. 42 (cit. on p. 5).

Agrawal, A., B. Amos, S. Barratt, S. Boyd, S. Diamond, and J. Z. Kolter (2019). "Differentiable Convex Optimization Layers". In: *Advances in Neural Information Processing Systems 32*. Ed. by H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett. Curran Associates, Inc., pp. 9562–9574 (cit. on p. 9).

Aumüller, M., E. Bernhardsson, and A. Faithfull (July 17, 2018). *ANN-Benchmarks: A Benchmarking Tool for Approximate Nearest Neighbor Algorithms*. URL: http://arxiv.org/abs/1807.05614 (visited on 11/28/2020) (cit. on p. 5).

Bach, F. R. and M. I. Jordan (2005). "Predictive Low-Rank Decomposition for Kernel Methods". In: *Proceedings of the 22nd International Conference on Machine Learning - ICML '05*. The 22nd International Conference. Bonn, Germany: ACM Press, pp. 33–40 (cit. on p. 5).

Berahas, A. S., L. Cao, K. Choromanski, and K. Scheinberg (Apr. 1, 2020). *A Theoretical and Empirical Comparison of Gradient Approximations in Derivative-Free Optimization*. URL: http://arxiv.org/abs/1905.01332 (visited on 11/28/2020) (cit. on pp. 2, 6).

Biau, G., F. Cérou, and A. Guyader (Apr. 1, 2010). "Rates of Convergence of the Functional K-Nearest

Neighbor Estimate". In: *IEEE Transactions on Information Theory* 56.4, pp. 2034–2040 (cit. on p. 2).

Biau, G. and L. Devroye (2015). *Lectures on the Nearest Neighbor Method*. Springer Series in the Data Sciences. Springer International Publishing (cit. on p. 2).

Boucheron, S., G. Lugosi, and P. Massart (2013). *Concentration Inequalities: A Nonasymptotic Theory of Independence*. Oxford University Press (cit. on pp. 1, 12).

Breiman, L. (Oct. 1, 2001). "Random Forests". In: *Machine Learning* 45.1, pp. 5–32 (cit. on p. 2).

Dalalyan, A. S., A. Juditsky, and V. Spokoiny (Aug. 2008). "A New Algorithm for Estimating the Effective Dimension-Reduction Subspace". In: *Journal of Machine Learning Research* 9, pp. 1647–1678 (cit. on pp. 2, 5).

De Brabanter, K., J. De Brabanter, B. De Moor, and I. Gijbels (Jan. 1, 2013). "Derivative Estimation with Local Polynomial Fitting". In: *The Journal of Machine Learning Research* 14.1, pp. 281–301 (cit. on p. 2).

Delecroix, M. and A. C. Rosa (Jan. 1, 1996). "Nonparametric Estimation of a Regression Function and Its Derivatives under an Ergodic Hypothesis". In: *Journal of Nonparametric Statistics* 6.4, pp. 367–382 (cit. on p. 2).

Detrano, R., A. Janosi, W. Steinbrunn, M. Pfisterer, J. J. Schmid, S. Sandhu, K. H. Guppy, S. Lee, and V. Froelicher (Aug. 1, 1989). "International Application of a New Probability Algorithm for the Diagnosis of Coronary Artery Disease". In: *The American Journal of Cardiology* 64.5, pp. 304–310 (cit. on p. 5).

Devroye, L., L. Györfi, and G. Lugosi (1996). *A Probabilistic Theory of Pattern Recognition*. Stochastic Modelling and Applied Probability. New York: Springer-Verlag (cit. on p. 2).

Fan, J. (Dec. 1992). "Design-Adaptive Nonparametric Regression". In: *Journal of the American Statistical Association* 87.420, pp. 998–1004 (cit. on pp. 3, 6).

– (Mar. 1993). "Local Linear Regression Smoothers and Their Minimax Efficiencies". In: *The Annals of Statistics* 21.1, pp. 196–216 (cit. on p. 6).

Fan, J. and I. Gijbels (Mar. 1, 1996). *Local Polynomial Modelling and Its Applications: Monographs on Statistics and Applied Probability 66*. CRC Press. 362 pp. (cit. on p. 2).

Gasser, T. and H.-G. Müller (1984). "Estimating Regression Functions and Their Derivatives by the Kernel Method". In: *Scandinavian Journal of Statistics* 11.3, pp. 171–185 (cit. on p. 2).

Giné, E. and A. Guillou (July 1, 2001). "On Consistency of Kernel Density Estimators for Randomly Censored Data: Rates Holding Uniformly over Adaptive Intervals". In: *Annales de l'Institut Henri Poincare (B) Probability and Statistics* 37.4, pp. 503–522 (cit. on pp. 12, 13).

Györfi, L., M. Kohler, A. Krzyzak, and H. Walk (2002). *A Distribution-Free Theory of Nonparametric Regression*. Springer Series in Statistics. New York: Springer-Verlag (cit. on p. 1).

Hastie, T., R. Tibshirani, and M. Wainwright (2015). *Statistical Learning with Sparsity: The Lasso and Generalizations*. Monographs on Statistics and Applied Probability. Chapman and Hall/CRC (cit. on pp. 4, 18).

Higgins, I., L. Matthey, A. Pal, C. Burgess, X. Glorot, M. Botvinick, S. Mohamed, and A. Lerchner (2017). "Beta-VAE: Learning Basic Visual Concepts with a Constrained Variational Framework". In: *ICLR* (cit. on p. 8).

Hristache, M., A. Juditsky, J. Polzehl, and V. Spokoiny (Dec. 2001). "Structure Adaptive Approach for Dimension Reduction". In: *Annals of Statistics* 29.6, pp. 1537–1566 (cit. on p. 2).

Hristache, M., A. Juditsky, and V. Spokoiny (May 1998). *Direct Estimation of the Index Coefficients in a Single-Index Model*. report. INRIA (cit. on p. 2).

Innes, M., A. Edelman, K. Fischer, C. Rackauckas, E. Saba, V. B. Shah, and W. Tebbutt (July 18, 2019). *A Differentiable Programming System to Bridge Machine Learning and Scientific Computing*. URL: http://arxiv.org/abs/1907.07587 (visited on 04/30/2020) (cit. on p. 6).

Jiang, H. (July 17, 2019). "Non-Asymptotic Uniform Rates of Consistency for k-NN Regression". In: *Proceedings of the AAAI Conference on Artificial Intelligence* AAAI Technical Track: Machine Learning (Vol 33 No 01: AAAI-19, IAAI-19, EAAI-20) (cit. on pp. 2, 4, 5).

Kalivas, J. H. (June 1997). "Two Data Sets of near Infrared Spectra". In: *Chemometrics and Intelligent Laboratory Systems* 37.2, pp. 255–259 (cit. on p. 5).

Kingma, D. P. and M. Welling (2014). "Auto-Encoding Variational Bayes". In: *ICLR* (cit. on p. 8).

Kpotufe, S. (2011). "K-NN Regression Adapts to Local Intrinsic Dimension". In: *Advances in Neural Information Processing Systems 24*. Ed. by J. Shawe-Taylor, R. S. Zemel, P. L. Bartlett, F. Pereira, and K. Q. Weinberger. Curran Associates, Inc., pp. 729–737 (cit. on pp. 2, 4).

Lecué, G. and S. Mendelson (Sept. 17, 2016). *Learning Subgaussian Classes : Upper and Minimax Bounds*. URL: http://arxiv.org/abs/1305.4825 (visited on 05/22/2020) (cit. on p. 1).

Lugosi, G. and S. Mendelson (2016). "Risk Minimization by Median-of-Means Tournaments". In: *Journal of the European Mathematical Society* 22.3 (cit. on p. 1).

Malkov, Y. A. and D. A. Yashunin (Apr. 2020). "Efficient and Robust Approximate Nearest Neighbor Search Using Hierarchical Navigable Small World Graphs". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 42.4, pp. 824–836 (cit. on p. 5).

Massart, P. (2007). *Concentration Inequalities and Model Selection: Ecole d'Eté de Probabilités de Saint-Flour XXXIII - 2003*. Ed. by J. Picard. École d'Été de Probabilités de Saint-Flour. Berlin Heidelberg: Springer-Verlag (cit. on p. 1).

Mukherjee, S. and Q. Wu (Dec. 1, 2006). "Estimation of Gradients and Coordinate Covariation in Classification". In: *The Journal of Machine Learning Research* 7, pp. 2481–2514 (cit. on p. 2).

Mukherjee, S., Q. Wu, and D.-X. Zhou (Feb. 2010). "Learning Gradients on Manifolds". In: *Bernoulli* 16.1, pp. 181–207 (cit. on p. 9).

Mukherjee, S. and D.-X. Zhou (Dec. 1, 2006). "Learning Coordinate Covariances via Gradients". In: *The Journal of Machine Learning Research* 7, pp. 519–549 (cit. on p. 2).

Nesterov, Y. and V. Spokoiny (Apr. 1, 2017). "Random Gradient-Free Minimization of Convex Functions". In: *Foundations of Computational Mathematics* 17.2, pp. 527–566 (cit. on p. 2).

Portier, F. and B. Delyon (Jan. 2, 2014). "Bootstrap Testing of the Rank of a Matrix via Least-Squared Constrained Estimation". In: *Journal of the American Statistical Association* 109.505, pp. 160–172 (cit. on p. 6).

Salimans, T., A. Karpathy, X. Chen, and D. P. Kingma (Jan. 19, 2017). *PixelCNN++: Improving the PixelCNN with Discretized Logistic Mixture Likelihood and Other Modifications*. URL: http://arxiv.org/abs/1701.05517 (visited on 06/02/2020) (cit. on p. 8).

Stone, C. J. (Dec. 1982). "Optimal Global Rates of Convergence for Nonparametric Regression". In: *Annals of Statistics* 10.4, pp. 1040–1053 (cit. on p. 4).

Street, W. N., W. H. Wolberg, and O. L. Mangasarian (July 29, 1993). "Nuclear Feature Extraction for Breast Tumor Diagnosis". In: IS&T/SPIE's Symposium on Electronic Imaging: Science and Technology. Ed. by R. S. Acharya and D. B. Goldgof. San Jose, CA, pp. 861–870 (cit. on p. 5).

Talagrand, M. (Nov. 1, 1996). "New Concentration Inequalities in Product Spaces". In: *Inventiones mathematicae* 126.3, pp. 505–563 (cit. on p. 13).

Trivedi, S., J. Wang, S. Kpotufe, and G. Shakhnarovich (July 23, 2014). "A Consistent Estimator of the Expected Gradient Outerproduct". In: *Proceedings of the Thirtieth Conference on Uncertainty in Artificial Intelligence*. UAI'14. Arlington, Virginia, USA: AUAI Press, pp. 819–828 (cit. on p. 2).

Van der Vaart, A. W. and J. Wellner (1996). *Weak Convergence and Empirical Processes: With Applications to Statistics*. Springer Series in Statistics. New York: Springer-Verlag (cit. on pp. 12, 17).

Wang, Y., S. Du, S. Balakrishnan, and A. Singh (Mar. 31, 2018). "Stochastic Zeroth-Order Optimization in High Dimensions". In: *International Conference on Artificial Intelligence and Statistics*. International Conference on Artificial Intelligence and Statistics, pp. 1356–1365 (cit. on pp. 2, 6).

Xia, Y. (Dec. 2007). "A Constructive Approach to the Estimation of Dimension Reduction Directions". In: *Annals of Statistics* 35.6, pp. 2654–2690 (cit. on p. 2).

Xia, Y., H. Tong, W. K. Li, and L.-X. Zhu (2002). "An Adaptive Estimation of Dimension Reduction Space". In: *Journal of the Royal Statistical Society. Series B (Statistical Methodology)* 64.3, pp. 363–410 (cit. on p. 2).

Ye, G.-B. and X. Xie (June 1, 2012). "Learning Sparse Gradients for Variable Selection and Dimension Reduction". In: *Machine Learning* 87.3, pp. 303–355 (cit. on p. 2).

Zhou, S. and D. A. Wolfe (2000). "On Derivative Estimation In Spline Regression". In: *Statistica Sinica* 10.1, pp. 93–108 (cit. on p. 2).