A New Look at Nearest Neighbours: Identifying Benign Input Geometries via Random Projections

Kaban, Ata

Journal of Machine Learning Research 1–16

A New Look at Nearest Neighbours: Identifying Benign Input Geometries via Random Projections

Ata Kabán A.Kaban@cs.bham.ac.uk
School of Computer Science, The University of Birmingham,
Edgbaston, B15 2TT, Birmingham, UK

Abstract

It is well known that in general, the nearest neighbour rule (NN) has sample complexity that is exponential in the input space dimension d when only smoothness is assumed on the label posterior function. Here we consider NN on randomly projected data, and we show that, if the input domain has a small "metric size", then the sample complexity becomes exponential in the metric entropy integral of the set of normalised chords of the input domain. This metric entropy integral measures the complexity of the input domain, and can be much smaller than d – for instance in cases when the data lies in a linear or a smooth nonlinear subspace of the ambient space, or when it has a sparse representation. We then show that the guarantees we obtain for the compressive NN also hold for the dataspace NN in bounded domains; thus the random projection takes the role of an analytic tool to identify benign structures under which NN learning is possible from a small sample size. Numerical simulations on data designed to have intrinsically low complexity confirm our theoretical findings, and display a striking agreement in the empirical performances of compressive NN and dataspace NN. This suggests that high dimensional data sets that have a low complexity underlying structure are well suited for computationally cheap compressive NN learning.

Keywords: Nearest Neighbours, Random projection, Sample Complexity

1. Introduction

A fundamental question in machine learning is the following: What makes it possible to generalise from few training points? Here we consider this question for nearest neighbour learning – that is, general learning of an unrestricted function class. To this end we will study the compressive version of this learning method where we make use of a result of Klartag & Mendelson (2005) that established a connection between random projection and empirical process theory. This connection brings us some notions of metric complexity that we use to progress our understanding of what kinds of data domains permit good generalisation from fewer training points in the case of nearest neighbour learning.

The setting of general unrestricted learning is of importance for several reasons. The nearest neighbour (NN) rule is extremely simple and intuitive, and widely used. Obtaining results on its sample complexity is of broad interest. While the study of complexity-constrained function classes was studied extensively in statistical machine learning theory, nearest neighbour type methods are of much recent interest (Gottlieb et al. , 2014; Chaudhuri & Dasgupta , 2014).

As we shall see, the lack of constraints on the function class will allow us to quantify the impact of complexity properties of the data domain on the generalisation of NN. In particular, a look at its compressive version reveals a connection between compressed sensing and compressed learning that completely puzzled previous attempts. Indeed, with the impressive advances in compressed sensing it has been tempting to work on the premise that data that has sparse representation must be easier to learn from – so tools taken from compressed sensing such as the restricted isometry property could be used. It then became clear that sparse representation of the data was irrelevant for tasks like linear or convex classification (Bandeira et al. , 2014; Durrant & Kabán , 2013) as long as most points have a large margin. Sparsity was also found unnecessary for compressive linear regression (Kabán , 2014).

In this paper we show that for nearest neighbour learning, contrary to learning of certain restricted parametric function classes studied before, a sparse representation does make learning easier i.e. makes it possible to generalise from fewer training points. Moreover, sparse representation is just one example of a much wider characterisation of "metric size" that governs the sample complexity of nearest neighbour learning.

2. Preliminaries and Tools

Definition 1 (Packing number) *Let $(T, \|\cdot\|)$ be a totally bounded pseudo metric space. Let $\alpha > 0$. We say that T is α-separated if $\forall a, b \in T, a \neq b$, we have $\|a, b\| \geq \alpha$.*
The α-packing number of T is defined as the maximum cardinality of the α-separated subsets of T, i.e. $N_{\|\cdot\|}(\alpha, T) = \max\{|T'| : T' \text{ is } \alpha\text{-separable}, T' \subset T\}$. *When the pseudometric is clear from the context we can omit the subscript.*

Definition 2 (α-entropy number) *The α-entropy number of T is defined as the log of the packing number, $H(\alpha, T) = \log N(\alpha, T)$.*

Definition 3 (Metric entropy) *The function $H(\cdot, T)$ is called the metric entropy of T.*

Theorem 4 *[Klartag & Mendelson (2005)]*[1]
Let $\mathcal{X} \subset \mathcal{R}^d$. Let R be a $k \times d, k < d$ random projection matrix with i.i.d. Gaussian or Rademacher entries with mean 0 and variance σ^2. Consider the set of all normalised chords between point pairs of $\mathcal{X}$: $T = \left\{ \frac{a-b}{\|a-b\|} : a, b \in \mathcal{X} \right\}$, with $\|\cdot\|$ being the Euclidean distance, and define the metric entropy integral

$$\gamma(T) = \int_0^1 \sqrt{H(\alpha, T)} d\alpha \tag{1}$$

where $H(\alpha, T)$ is the α-entropy number of T w.r.t. the Euclidean distance.

Then, $\exists c$ absolute constant s.t. $\forall \zeta, \delta \in (0, 1)$, if

$$k \geq c\zeta^{-2}(\gamma^2(T) + \log(2/\delta)) \tag{2}$$

1. In fact, this is a simplified version cf. Boucheron et al. (2013) (Thm 13.15), with $\gamma(T)$ being a more user-friendly upper bound on the γ_2 functional that upper bounds the supremum of a stochastic process with subgaussian increments. The result in Klartag & Mendelson (2005) allows R with i.i.d. entries from any subgaussian distribution.

then R is an ζ-isometry on $\mathcal{X}$ with high probability, i.e. with probability at least $1-\delta$ we have:

$$(1-\zeta)k\sigma^2\|x-x'\|^2 \leq \|Rx-Rx'\|^2 \leq (1+\zeta)k\sigma^2\|x-x'\|^2, \quad \forall x,x' \in \mathcal{X} \tag{3}$$

This uniform bound represents a great generalisation of the Johnson-Lindenstrauss lemma (JLL) and allows the set of points $\mathcal{X}$ to be infinite as long as $\gamma(T)$ is finite.

In the special case when $\mathcal{X}$ is a finite set of N points then $\gamma^2(T) \in \Omega(\log(N))$, as implied by the JLL. It also recovers the Restricted Isometry Property as a special case, since for s-sparse vectors $\gamma^2(T) \leq 2s\log(d/(2s))$ (Boucheron et al. , 2013). Other low complexity structures covered by this result include certain smooth manifolds, and metric spaces with finite doubling dimension. The latter was exploited in Indyk & Naor (2007) for approximate nearest neighbour search to reduce the computational complexity.

Rather curiously the result of Klartag & Mendelson in its full generality has not yet been introduced to statistical machine learning theory. However, there are several works in the literature that generalised JLL specifically for subspace embeddings (Sarlós , 2006) and for manifold embeddings (Baraniuk & Wakin , 2007; Clarkson , 2007; Verma , 2011) that have implications for unsupervised learning of smooth manifolds from compressive data.

Here we are interested in capturing more generally the low complexity input space structures that allow a reduction in sample complexity for supervised classification by nearest neighbours. This complements recent interest and progress in exploiting some appropriate notion of *intrinsic dimension* for analysing supervised learning tasks. For example, the doubling dimension was used for analysing tree-based regression in (Verma et al. , 2009) and (Kpotufe & Dasgupta , 2012). Since nearest neighbours is an unrestricted nonparametric method, the generality of Theorem 4 will be beneficial.

3. Generalisation and sample complexity of compressive nearest-neighbour classification

Let $S = \{(x_1, y_1), \ldots, (x_N, y_N)\}$ be a training set drawn i.i.d. from some unknown distribution $\mathcal{D}$ over the input-output domain $\mathcal{X} \times \mathcal{Y}$ where $\mathcal{Y} = \{0, 1\}$ for classification problems, and we take $\mathcal{X} = [-1, 1]^d$. $\mathcal{D}_X$ will denote the marginal distribution over the inputs $\mathcal{X}$. Further, denote by $\eta : \mathcal{R}^d \to \mathcal{R}$ the true conditional probability of the labels, i.e. $\eta(x) = \Pr(Y = 1|X = x)$. Since we consider general learning of an unconstrained function class, some form of Lipschitz-like assumption is known to be needed on $\eta(\cdot)$ for learnability (Shalev-Shwartz & Ben-David , 2014).

3.1. Nearest Neighbour

Consider the nearest neighbour classifier of S, which will be denoted as h_S. This is a nonparametric method that, given an input point $x \in \mathcal{X}$ it looks up its nearest neighbour, denoted $N(x) \in S$ and returns its label, $h_S(x) = Y_{N(x)}$. The generalisation error of h_S is defined as $err(h_S) = \mathrm{E}_{(x,y)\sim\mathcal{D}}[h_S(x) \neq y]$, where (x, y) is a query point drawn independently from and identically distributed as the training points. The Bayes-optimal classifier will be denoted as h^*.

It is well known (Cover & Hart , 1967) that the nearest neighbour (NN) classifier converges to at most twice the Bayes error as the number of training points grows to infinity.

Non-asymptotic (finite sample) analyses of NN are more recent (Shalev-Shwartz & Ben-David , 2014; Gottlieb et al. , 2014; Chaudhuri & Dasgupta , 2014).

The recent work of Chaudhuri & Dasgupta (2014) obtained finite sample bounds for K-NN under weaker assumptions than previous studies, which subsume Hölder-continuity (hence also Lipschitzness) as a special case. Under the Tsybakov noise condition their rates match the optimal rates known for nonparametric classification. But the associated sample complexity is still exponential in the data dimension d.

In turn, our interest in this work is to identify input domains for which the $\mathcal{O}(\exp(d))$ sample complexity of NN can be reduced. This is somewhat in the spirit of the work in (Gottlieb et al. , 2013), where the authors show for regularised linear function classes that improved error bounds are achievable when the data lies close to a low dimensional linear subspace, and furthermore, in general metric spaces, for classifiers realised by Lipschitz functions they give improved error bounds that scale with the doubling dimension of the space.

In this work we will take the ambient space to be Euclidean, but the intrinsic structure can be non-Euclidean. The use of random projections will allow us to preserve and uncover benign intrinsic structures for (compressive) NN learning that characterise the generalisation, sample complexity and convergence rates of (compressive) nearest neighbour.

As a first attempt at doing this, we will make the standard Lipschitz continuity assumption on the label posterior function, and will work on bounded input domains. A particularly simple and insightful approach is in (Shalev-Shwartz & Ben-David , 2014) under the assumption that η is Lipschitz with constant L, which we will build on here. While we reckon that these conditions are considerably stronger than those in (Chaudhuri & Dasgupta , 2014), it makes the technical work needed to obtain new insights more straightforward. In particular, under these conditions we can show that we can replace d with a metric entropy integral. We are also able to relax the Lipschitz assumption to the relatively recently proposed 'probabilistic Lipschitzness' condition at the expense of an extra additive term in the excess risk. This condition is particularly well suited for various extension to semi-supervised settings where a condition of more generative flavour is appropriate. Whether it would be possible to obtain our results under weaker or different assumptions, e.g. those in (Chaudhuri & Dasgupta , 2014) remains for further research.

To keep the exposition simple, and keep the focus on our new findings, for the purposes of this work we will limit ourself to NN, although KNN can be analysed in the same way e.g. following Shalev-Shwartz & Ben-David (2014). Non-asymptotic analysis has the advantage of giving generalisation guarantees for any finite sample size N. In particular, for the above setting, the following was obtained in (Shalev-Shwartz & Ben-David , 2014):

Theorem 5 (Shalev-Shwartz & Ben-David (2014): Theorem 19.3) *Let $\mathcal{X} = [0,1]^d, \mathcal{Y} = \{0,1\}$, and $\mathcal{D}$ a distribution over $\mathcal{X} \times \mathcal{Y}$ for which the conditional probability function is L-Lipschitz. Let h_S denote the nearest neighbour rule applied to the training set $S \sim \mathcal{D}^N$. Then,*

$$E_S[err(h_S)] \leq 2err(h^*) + 4L\sqrt{d}N^{-\frac{1}{d+1}} \tag{4}$$

which implies the sample complexity

$$N \geq \left(\frac{4L\sqrt{d}}{\epsilon}\right)^{d+1} \in \tilde{\Omega}(\exp(d)) \tag{5}$$

to guarantee $\mathrm{E}_S[err(h_S)] \leq 2err(h^*) + \epsilon$.

From this result, taken together with the No Free Lunch theorem, it was concluded (Shalev-Shwartz & Ben-David , 2014; Urner , 2013) that the exponential sample complexity of NN with the input dimension d is essential, and not just a byproduct of the proof technique used.

3.2. Compressive Nearest Neighbour

Let R be a $k \times d, k < d$ matrix with i.i.d. entries drawn from a subgaussian distribution such as a Gaussian or a Rademacher distribution. We will create and work with the compressed training set $S_R = \{(Rx_1, y_1), \ldots, (Rx_N, y_N))\}$.

Denote by $N_R(x)$ the training point $x' \in S$ such that Rx' is the nearest neighbour of Rx after random projection. Of course, this is not the same as $N(x)$ in general. Rather it may be thought of as an approximate nearest neighbour – indeed, Indyk & Naor (2007) has shown that it is an $(1+\epsilon)$-nearest neighbour, provided that k is chosen to be of the order of the metric entropy of the input space. While their motivation was to create an approximate nearest neighbour algorithm to reduce the computation time and has not considered the sample complexity, our purpose in the sequel is the latter.

The nearest neighbour classifier in the compressive space receives S_R, and will be denoted by $h^R_{S_R}$. We are interested in the distribution of its expected generalisation error, $\mathrm{E}_{S\sim\mathcal{D}^N}[err(h^R_{S_R})] = \mathrm{E}_S[\mathrm{E}_{(x,y)\sim\mathcal{D}}[h^R_{S_R}(Rx) \neq y]]$, as a random function of R.

3.3. Main Result

Theorem 6 *Let $\mathcal{X} = [-1,1]^d, \mathcal{Y} = \{0,1\}$, and $\mathcal{D}$ a distribution over $\mathcal{X} \times \mathcal{Y}$ for which the conditional probability function is L-Lipschitz. Let R be a $k \times d$ random matrix, $k < d$, with i.i.d. Gaussian or Rademacher entries with mean 0 and parameter σ^2. Let $h^R_{S_R}$ denote the nearest neighbour rule applied to the randomly projected training set S_R where $S \sim \mathcal{D}^N$. Then $\forall \delta, \zeta \in (0,1)$, with probability at least $1-\delta$ over the random draws of R, the expected generalisation error of compressive nearest neighbour is upper bounded as:*

$$E_S[err(h^R_{S_R})] \leq 2err(h^*) + 2\sqrt{2}\left(L\sqrt{d}\sqrt{\frac{1+\zeta}{1-\zeta}}\right)^{\frac{k}{k+1}} (eN)^{-\frac{1}{k+1}}\sqrt{k} \tag{6}$$

provided that $k \in \Omega(\zeta^{-2}(\gamma^2(T) + \log(2/\delta)))$.

It may be useful to point out that the $\sqrt{d}$ factor is not essential: As it will be apparent from the proof shortly, in our bounds the dependence on d came in only because the input space in the original data space was taken as $\mathcal{X} = [-1,1]^d$ so the maximal length of any point is $2\sqrt{d}$. If instead we take $\mathcal{X} = \mathcal{B}(0,\rho)$, the ball of radius ρ, then $\sqrt{d}$ would get replaced by ρ and so the error becomes independent of the ambient dimension d. We give this as a corollary.

Corollary 7 *Let $\mathcal{X} = \mathcal{B}(0,\rho) \in \mathcal{R}^d, \mathcal{Y} = \{0,1\}$, and all conditions identical to those in Theorem 6. Then $\forall \delta, \zeta \in (0,1)$, with probability at least $1-\delta$ over the random draws of R, the expected generalisation error of compressive nearest neighbour is upper bounded as:*

$$E_S[err(h^R_{S_R})] \leq 2err(h^*) + 2\sqrt{2}\left(L\rho\sqrt{\frac{1+\zeta}{1-\zeta}}\right)^{\frac{k}{k+1}}(eN)^{-\frac{1}{k+1}}\sqrt{k} \tag{7}$$

provided that $k \in \Omega(\zeta^{-2}(\gamma^2(T) + \log(2/\delta)))$.

The sample complexity of compressive nearest neighbour is an immediate corollary:

Corollary 8 *The following sample size guarantees that $E_S[err(h^R_{S_R})] \leq 2err(h^*) + \epsilon$ w.p. $1-\delta$:*

$$N \geq \frac{1}{e}\left(\frac{2\sqrt{2}\sqrt{k}}{\epsilon}\right)^{k+1}\left(L\rho\sqrt{\frac{1+\zeta}{1-\zeta}}\right)^{k} = \tilde{\Omega}(exp(\gamma(T))) \tag{8}$$

provided that $k \in \Omega(\zeta^{-2}(\gamma^2(T) + \log(2/\delta)))$.

Proof [of Theorem 6]

$$\begin{aligned}
\mathrm{E}_S[err(h^R_{S_R})] &= \mathrm{E}_{S\sim\mathcal{D}^N}[\Pr_{(x,y)\sim\mathcal{D}}(Y_{N_R(x)} \neq y)] &(9)\\
&= \mathrm{E}_S\mathrm{E}_{x,y}[\mathbf{1}(y=1)\mathbf{1}(Y_{N_R(x)}=0) + \mathbf{1}(y=0)\mathbf{1}(Y_{N_R(x)}=1)] &(10)\\
&= \mathrm{E}_S\mathrm{E}_x[\mathrm{E}_{y|x}[\mathbf{1}(y=1)\mathbf{1}(Y_{N_R(x)}=0) + \mathbf{1}(y=0)\mathbf{1}(Y_{N_R(x)}=1)]] &(11)\\
&= \mathrm{E}_{S,x}[\eta(x)\mathbf{1}(Y_{N_R(x)}=0) + (1-\eta(x))\mathbf{1}(Y_{N_R(x)}=1)] &(12)
\end{aligned}$$

since $\mathrm{E}_{y|x}[\mathbf{1}(y=1)] = \eta(x)$ by definition, and using the linearity of expectation. Here, $\mathbf{1}(\cdot)$ is 1 if its argument is true and 0 otherwise.

Likewise for the point $(N_R(x), Y_{N_R(x)}) \in S$ we use that $\mathrm{E}[Y_{N_R(x)}|N_R(x)] = \eta(N_R(x))$, and write (12) further as the following:

$$\begin{aligned}
\mathrm{E}_S[err(h^R_{S_R})] &= \mathrm{E}_S\mathrm{E}_x[\eta(x)(1-\eta(N_R(x))) + (1-\eta(x))\eta(N_R(x))] &(13)\\
&= \mathrm{E}_S\mathrm{E}_x[(\eta(x)-\eta(N_R(x)))(2\eta(x)-1) + 2\eta(x)(1-\eta(x))] &(14)\\
&\leq \mathrm{E}_S\mathrm{E}_x[|\eta(x)-\eta(N_R(x))|\cdot|2\eta(x)-1| + 2\eta(x)(1-\eta(x))] &(15)
\end{aligned}$$

where the re-writing in eq. (14) is easy to verify, and eq. (15) use the Cauchy-Schwarz inequality. Noting that $|2\eta(x)-1| \leq 1$, this is further bounded by the following:

$$\begin{aligned}
\mathrm{E}_S[err(h^R_{S_R})] &\leq \underbrace{\mathrm{E}_S\mathrm{E}_x[|\eta(x)-\eta(N_R(x))|]}_{T_1} + 2\mathrm{E}_x[\eta(x)(1-\eta(x))]\\
&\leq T_1 + 2err(h^*) &(16)
\end{aligned}$$

because, as in (Shalev-Shwartz & Ben-David , 2014) (Lemma 19.1), $\mathrm{E}_x[\eta(x)(1-\eta(x))] \leq \min\{\eta(x), 1-\eta(x)\} = err(h^*)$ is upper bounded by the Bayes error.

It remains to further bound the first term, denoted by T_1. Define the 'good' set of random projection matrices:

$$G := \{R : \forall x, x' \in \mathcal{X}, \sqrt{1-\zeta}\sqrt{k}\sigma\|x-x'\| \leq \|Rx - Rx'\| \leq \sqrt{1+\zeta}\sqrt{k}\sigma\|x-x'\|\} \quad (17)$$

Klartag & Mendelson (2005), cf Theorem 4 guarantees that

$$\Pr(R \notin G) < \delta \quad (18)$$

provided that k scales with the "metric size" of the set of normalised chords of the input space, i.e. it satisfies eq. (2). So, w.p. $1-\delta$,

$$T_1 \quad = \quad \mathrm{E}_S\mathrm{E}_x[|\eta(x) - \eta(N_R(x))| \; | R \in G] \quad (19)$$

and bound this further. We will use the Lipschitz property of η – that will bound $|\eta(x) - \eta(N_R(x))| \leq L \cdot \|x - N_R(x)\|$. The latter is a distance in the d-dimensional space $\mathcal{X}$, and we can use the Klartag-Mendelson theorem to bound this with a distance in the k-dimensional random projection space. We then use an approach of covering with mutually disjoint boxes, similar to the analysis of nearest neighbour in (Shalev-Shwartz & Ben-David , 2014) (Theorem 19.3) – but this time it will be the smaller, k-dimensional projected input space to be covered instead of $\mathcal{X} \in \mathcal{R}^d$. Thinking ahead to this end, it is convenient to rewrite T_1 using the mentioned covering before even using the Lipschitz property of the function $\eta(\cdot)$. This way one of the resulting terms will turn out easier to bound.

Denote the input space after random projection by $[-b_R, b_R]^k$, where we will determine b_R later. We cover this set with r disjoint boxes of side length s each. A number of $r = \left(\frac{2b_R}{s}\right)^k$ boxes are sufficient for this. We leave s unspecified for now, and will choose it later to optimise the resulting error bound. The diameter of each of these rectangles is $s\sqrt{k}$.

Denote by $C_R(x)$ the box that contains Rx. Denote $S_{R|\mathcal{X}} = \{Rx_1, \ldots, Rx_N\}$ the restriction of S_R to the inputs. The box either contains no points from $S_{R|\mathcal{X}}$, or it contains the point $RN_R(x)$. We use the law of total expectation to split T_1 into these two cases:

$$\begin{aligned} T_1 \quad &= \quad \mathrm{E}_S\mathrm{E}_x[|\eta(x) - \eta(N_R(x))| \; | R \in G, C_R(x) \cap S_{R|\mathcal{X}} = \emptyset] \cdot \Pr(C_R(x) \cap S_{R|\mathcal{X}} = \emptyset) \\ &+ \quad \mathrm{E}_S\mathrm{E}_x[|\eta(x) - \eta(N_R(x))| \; | R \in G, C_R(x) \cap S_{R|\mathcal{X}} \neq \emptyset] \cdot \Pr(C_R(x) \cap S_{R|\mathcal{X}} \neq \emptyset) \end{aligned}$$

Now, by Lemma 19.2 in (Shalev-Shwartz & Ben-David , 2014), we have:

$$\Pr(C_R(x) \cap S_{R|\mathcal{X}} = \emptyset) \leq \frac{r}{eN} \quad (20)$$

and we use the following trivial bounds as well:

$$\begin{aligned} \mathrm{E}_S\mathrm{E}_x[|\eta(x) - \eta(N_R(x))| \; | R \in G, C_R(x) \cap S_{R|\mathcal{X}} = \emptyset] \leq 1 \\ \Pr(C_R(x) \cap S_{R|\mathcal{X}} \neq \emptyset) \leq 1 \end{aligned}$$

where the former holds because $\eta(\cdot)$ takes values in $[0,1]$. Using these,

$$T_1 \leq \frac{r}{eN} + \mathrm{E}_S\mathrm{E}_x[|\eta(x) - \eta(N_R(x)| \; | R \in G, C_R(x) \cap S_{R|\mathcal{X}} \neq \emptyset]$$

Finally, by using the Lipschitz property of $\eta(\cdot)$, and then the fact that $R \in G$, we get:

$$\begin{aligned} T_1 &\leq \frac{r}{eN} + L \cdot \mathrm{E}_S \mathrm{E}_x [\|x - N_R(x)\| \; | R \in G, C_R(x) \cap S_{R|\mathcal{X}} \neq \emptyset] \\ &\leq \frac{r}{eN} + L \cdot \mathrm{E}_S \mathrm{E}_x \left[\frac{\|Rx - RN_R(x)\|}{\sqrt{1-\zeta}\sqrt{k}\sigma} \right] \; | R \in G, C_R(x) \cap S_{R|\mathcal{X}} \neq \emptyset] \\ &\leq \frac{r}{eN} + \frac{Ls\sqrt{k}}{\sqrt{(1-\zeta)k}\sigma} \end{aligned} \tag{21}$$

In the last line we used that the diameter of $C_R(x)$ is $s\sqrt{k}$.

We will now optimise this bound w.r.t. the side-length of the covering boxes, s – noting that the r.h.s. of eq. (21) are the only ones that depend on s.

For convenience, introduce the shorthand

$$L_R = \frac{L}{\sqrt{(1-\zeta)k}\sigma} \tag{22}$$

Also, recall that r is a function of s, so we plug in that $r = \left(\frac{2b_R}{s}\right)^k$. Then we can write the r.h.s. of rq. (21) as the following. Define

$$\frac{1}{eN}\left(\frac{2b_R}{s}\right)^k + L_R s\sqrt{k} =: f(s) \tag{23}$$

This is a function to be minimised in s.

Compute the derivative and equate it to zero (after checking that this will give us a minimum):

$$f'(s) = -k\frac{(2b_R)^k}{eN}s^{-(k+1)} + L_R\sqrt{k} = 0 \tag{24}$$

Solving we get the optimal s:

$$s_{opt} = (2b_R)^{\frac{k}{k+1}}(eL_RN)^{-\frac{1}{k+1}}\sqrt{k}^{\frac{1}{k+1}} \tag{25}$$

We note in passing that should we have chosen to use the Lipschitz property of $\eta(\cdot)$ directly at eq. (19) before the step of covering with boxes, the ensuing optimisation might have turned out to have no analytic solution, causing some technical inconvenience.

Plugging back, after a few lines of algebra, we get:

$$f(s_{opt}) = (2b_RL_R)^{\frac{k}{k+1}}(eN)^{-\frac{1}{k+1}}\sqrt{k}\left(\sqrt{k}^{-\frac{2k+1}{k+1}} + \sqrt{k}^{\frac{1}{k+1}}\right) \tag{26}$$

Now, one can show that the sequence

$$a_k := 2^{\frac{k}{k+1}}\left(\sqrt{k}^{-\frac{2k+1}{k+1}} + \sqrt{k}^{\frac{1}{k+1}}\right) \tag{27}$$

is decreasing and its first term is $a_1 = 2\sqrt{2}$. Therefore,

$$f(s_{opt}) < 2\sqrt{2}(b_RL_R)^{\frac{k}{k+1}}(eN)^{-\frac{1}{k+1}}\sqrt{k} \tag{28}$$

Hence, T_1 is upper bounded by this simpler expression.

We now need an estimate of b_R. This is straightforward using that $R \in G$: Since the input domain was originally $\mathcal{X} = [-1, 1]^d$ we have $\|x\| \leq \sqrt{d}$, so b_R is bounded as:

$$\begin{aligned} b_R &\leq \|Rx\| \leq \|x\|\sigma\sqrt{k}\sqrt{1+\zeta} && (29) \\ &\leq \sqrt{d}\sqrt{k}\sigma\sqrt{1+\zeta} && (30) \end{aligned}$$

Plugging this back into eq. (28), and replacing the expression of L_R from eq. (22), we obtain:

$$\begin{aligned} T_1 &\leq 2\sqrt{2}(b_R L_R)^{\frac{k}{k+1}}(eN)^{-\frac{1}{k+1}}\sqrt{k} && (31) \\ &\leq 2\sqrt{2}\left(\sqrt{d}\sqrt{\frac{1+\zeta}{1-\zeta}}L\right)^{\frac{k}{k+1}}(eN)^{-\frac{1}{k+1}}\sqrt{k} && (32) \end{aligned}$$

Note, as one intuitively expected indeed, the parameter of the entries of R, i.e. σ^2 has cancelled out.

Finally, combining eqs (16), (19) and (32) completes the proof. ■

Proof [of Corollary 8]. Eq. (32) is the error above the inevitable $2err(h^*)$in the random projection space. We require this to be less than some ϵ. Solving for N we get the sample complexity

$$N \geq \frac{1}{e}\left(\frac{2\sqrt{2}\sqrt{k}}{\epsilon}\right)^{k+1}\left(L\rho\sqrt{\frac{1+\zeta}{1-\zeta}}\right)^{k} \quad (33)$$

which is independent of the original data dimension d, and exponential in k. Plugging in the required order of k completes the proof. ■

3.3.1. Relaxing the Lipschitz assumption

The standard Lipschitz assumption we have used so far can be relaxed. One fairly recent and easily tractable alternative is the condition of 'probabilistic Lipshitzness' (PL), introduced by Urner (2013). It allows a controlled fraction of training points to have non-Lipschitz label posterior functions.

Definition [Probabilistic Lipschitzness (Urner , 2013)] Let $\phi : \mathcal{R}^+ \to [0, 1]$ be an increasing function. A function $\eta : \mathcal{X} \to [0, 1]$ is PL wrt. $\mathcal{D}_x$ if $\forall L > 0$,

$$\Pr_{x\sim\mathcal{D}_x}\left[\Pr_{x'\sim\mathcal{D}_x}[|\eta(x) - \eta(x')| > L\|x - x'\| > 0]\right] \leq \phi(1/L) \quad (34)$$

Swapping this for the standard Lipschitzness we obtain:

Corollary 9 *Let $\mathcal{X} = \mathcal{B}(0, \rho) \in \mathcal{R}^d, \mathcal{Y} = \{0, 1\}$, and $\mathcal{D}$ a distribution over $\mathcal{X} \times \mathcal{Y}$ for which the conditional probability function is probabilistically L-Lipschitz. Let R be a $k \times d$ random matrix, $k < d$, with i.i.d. Gaussian or Rademacher entries with mean 0 and variance σ^2. Let $h^R_{S_R}$ denote the nearest neighbour rule applied to the randomly projected training set S_R*

where $S \sim \mathcal{D}^N$. Then $\forall \delta, \zeta \in (0,1)$, with probability at least $1-\delta$ over the random draws of R, the expected generalisation error of compressive nearest neighbour is upper bounded as:

$$E_S[err(h^R_{S_R})] \leq 2err(h^*) + \phi(1/L) + 2\sqrt{2}\left(L\rho\sqrt{\frac{1+\zeta}{1-\zeta}}\right)^{\frac{k}{k+1}} (eN)^{-\frac{1}{k+1}}\sqrt{k} \tag{35}$$

provided that $k \in \tilde{\Omega}(\zeta^{-2}(\gamma^2(T) + \log(2/\delta)))$.

3.4. Implication for dataspace Nearest Neighbour

The exponential sample complexity of NN has never been questioned, and, as we already mentioned, the No Free Lunch theorems (see e.g. Shalev-Shwartz & Ben-David (2014)) indeed imply that this cannot be improved in general. It is now of interest to know if the low metric complexity input structures seen in the previous section would also be fortuitous for NN in the original data space? It turns out that the answer is positive. The proof presented for compressive NN can easily be modified such that the random projection only serves as an analytic tool while the NN rule runs in the original d-dimensional space.

Theorem 10 *Let $\mathcal{X} = \mathcal{B}(0,\rho) \in \mathcal{R}^d, \mathcal{Y} = \{0,1\}$, and $\mathcal{D}$ a distribution over $\mathcal{X} \times \mathcal{Y}$ for which the conditional probability function is probabilistically L-Lipschitz. h_S denote the nearest neighbour rule, where $S \sim \mathcal{D}^N$. For any $\delta, \zeta \in (0,1)$, the expected generalisation error of h_S upper bounded as*

$$E_S[err(h_S)] \leq 2err(h^*) + \phi(1/L) + 2\sqrt{2}\left(L\rho\sqrt{\frac{1+\zeta}{1-\zeta}}\right)^{\frac{k}{k+1}} (eN)^{-\frac{1}{k+1}}\sqrt{k} \tag{36}$$

with confidence $1-\delta$, where $k \in \Omega(\zeta^{-2}(\gamma^2(T) + \log(2/\delta)))$, and $\gamma(T)$ is the metric entropy integral as defined in eq. (1).

The new sample complexity that we obtain from this, given below, recovers the known exponential scaling in d when the input space fills the domain $\mathcal{X}$. However, for low complexity input spaces $\gamma^2(T)$ will be less than d and therefore the sample complexity is improved.

Corollary 11 *The sample size required to guarantee $E_S[err(h_S)] \leq 2err(h^*) + \phi 1/L\epsilon$ w.p. $1-\delta$ is*

$$N \geq \frac{1}{e}\left(\frac{2\sqrt{2}\sqrt{k}}{\epsilon}\right)^{k+1}\left(L\rho\sqrt{\frac{1+\zeta}{1-\zeta}}\right)^{k} = \tilde{\Omega}(exp(\gamma(T))) \tag{37}$$

Proof[of Theorem 10].
The reasoning is very similar to what we have seen before in the proof of Theorem 6, so a sketch will be sufficient.

$$\begin{aligned} \mathrm{E}_S[err(h_S)] &= \mathrm{E}_S[\Pr_{(x,y)\sim D}(Y_{N(x)} \neq y)] && (38)\\ &\leq \underbrace{\mathrm{E}_S\mathrm{E}_x[|\eta(x) - \eta(N(x))|]}_{T_1} + 2\mathrm{E}_x[\eta(x)(1-\eta(x))] \\ &\leq T_1 + 2err(h^*) && (39)\end{aligned}$$

We define the good set of random projection matrices R as before in eq. (17).

Using the same steps as before, we arrive at

$$T_1 \quad \leq \quad \mathrm{E}_S \mathrm{E}_x[|\eta(x) - \eta(N(x))| \; | R \in G] \tag{40}$$

w.p. $1-\delta$, and denoting by $S_{\mathcal{X}} = \{x_1, \ldots, x_N\}$ the training inputs,

$$\begin{aligned} T_1 \quad &\leq \quad \frac{r}{eN} + \mathrm{E}_S \mathrm{E}_x[|\eta(x) - \eta(N(x)| \; | R \in G, C_R(x) \cap S_{\mathcal{X}} \neq \emptyset] \\ &\leq \quad \frac{r}{eN} + L \cdot \mathrm{E}_S \mathrm{E}_x[\|x - N(x)\| \; | R \in G, C_R(x) \cap S_{\mathcal{X}} \neq \emptyset] \end{aligned} \tag{41}$$

having used the Lipschitz property of $\eta(\cdot)$. Recall that $N(x)$ is the nearest neighbour in the original training set S.

Next, we move to the compressive space to do the covering by boxes. Notice that,

$$\|x - N(x)\| \leq \|x - N_R(x)\| \tag{42}$$

This follows trivially from the definition of nearest neighbour, and it allows us to work with the nearest neighbour in the compressive space rather than that in the original space.

The remainder of the proof is now identical to that of the previous section from eq. (21) onwards, with ρ replacing $\sqrt{d}$. ■

Discussion We should point out that the above implicit analysis of dataspace NN with no explicit random projection is aimed to provide new insight into nearest neighbour learning, without the ambition of tightness. That is, we replaced the exponential dependence on the data dimension d to exponential in k. The exponential dependence on k is essential, for the same reasons as in the original analysis of NN. The point is that, for a constant ζ, we need $k \in \tilde{\Omega}(\gamma^2(T))$, and $\gamma^2(T)$ gets smaller than d in many cases when the input space does not fill the entire d-dimensional space. For example, if the data has a sparse representation on the full domain $\mathcal{X}$ then the sample complexity of NN is exponential in the number of non-zeros in that representation and only polynomial in d and in ρ. This is because $\gamma^2(T) = \Omega(s \log(d/s))$. Also we should point out that the use of NN does not need to have any a-priori knowledge about the existence of such structures – it will just take less training points to generalise well. The role of the theory presented here is to understand why that is.

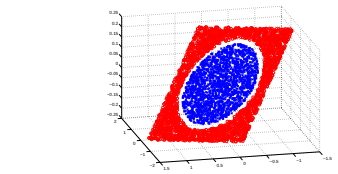
On the downside, the bound is dependent on the failure probability δ and distortion parameter ζ of a random projection, but there is no explicit random projection happening in the algorithm – this, of course, is an artifact of the analysis. Nevertheless, the obtained result qualitatively agrees with recent findings in the direction of bounding the performance of nonparametric learning in terms of a notion of intrinsic dimension such as the doubling dimension e.g. in (Kpotufe & Dasgupta , 2012; Verma et al. , 2009). In turn, contrary from those works, here a simple RP was used to uncover such benign structures. In particular, our analysis suggests that the benign input structures for compressive NN are also benign for dataspace NN. This will be verified experimentally and demonstrated to uphold in practice in the next section.

In addition, a further attractive aspect of the random projection based analysis, from the practical point of view is that in the presence of benign structures in the data set one

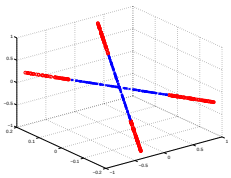
can in fact carry out the learning task in the compressive space without noticeable loss in performance, which gives us great computational savings. This point will be experimentally illustrated for NN vs. compressive NN in the next section.

4. Empirical illustration

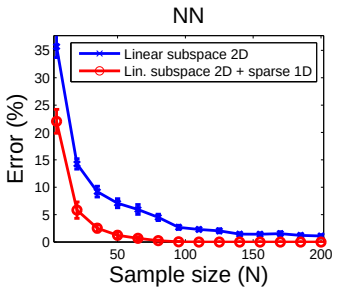
To illustrate the theory for compressive NN, and to verify that NN in the data space benefits from the low complexity input structures in the same way as its compressive counterpart, we generated some example data sets that exhibit low metric size. We use these to train and empirically test the classifiers when the training set size varies from $N = 5$ to $N = 200$. All data sets used in these experiments are $d = 100$ dimensional, so these sample sizes are rather small. We will also use a couple of 3-dimensional examples (instead of 100) for visual illustration purposes only.



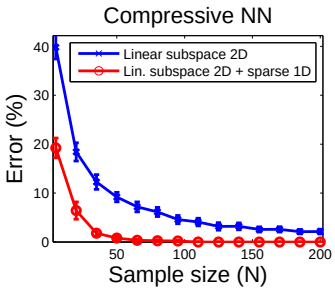
(*a*) Linear subspace



(*b*) Linear subspace & Sparse



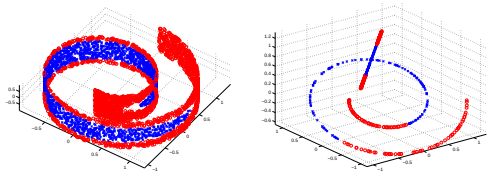
(*c*) Performance of NN



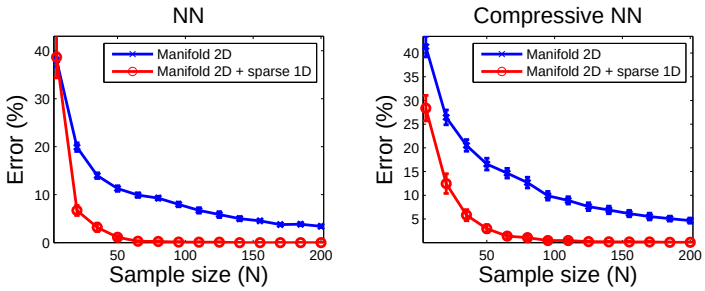
(*d*) Performance of Compressive NN

Figure 1: Empirical estimates of test error for Nearest Neighbour (NN) and Compressive NN on 100-dimensional data sets where the data lies in a 2D linear subspace, versus when in addition the data has a sparse representation in the subspace. The compressive dimension is 4 (twice the dimension of the subspace containing the data). The class labelling is such that the Bayes error is 0. The plots in the upper row illustrate 3-dimensional versions of the data sets, with the two markers / colours indicating class labels; the plots on the lower row shows the error estimates. We see that: (i) NN and Compressive NN have very similar error behaviour; (ii) Sparse representation of the input data lowers the error.

First, we generated ten $d = 100$ dimensional data sets so that the input points that lie on a 2-dimensional linear subspace. Hence for these data, $\gamma^2(T)$ coincides with the dimension of the subspace that contains the data, which is 2. An example with the ambient dimension



(a) Nonlinear subspace (b) Nonlin. subspace & Sparse



(c) Performance of NN (d) Performance of Compressive NN

Figure 2: Empirical estimates of test error for Nearest Neighbour (NN) and Compressive NN on 100-dimensional data sets where the data lies in a 2D nonlinear subspace (manifold), versus when in addition the data has a sparse representation. The compressive dimension is 4 (twice the dimension of the manifold containing the data). The class labelling is such that the Bayes error is 0. The plots in the upper row illustrate 3-dimensional versions of the data sets, with the two markers /colours indicating class labels; the plots on the lower row shows the error estimates. Again we see that: (i) NN and Compressive NN have very similar error behaviour; (ii) Sparse representation of the input data lowers the error.

being $d = 3$ is shown on Figure 1 (a). In order to test the effect of sparse representation, we further created another ten data sets where again the input points lie in a 2D linear subspace and in addition they have a sparse representation – that is, there is a linear basis in the 2D subspace containing the data, such that in that basis only one coordinate is nonzero for each point. Again, a 3-dimensional example is shown in Figure 1 (b). We labelled all data sets such that the Bayes error is zero, for reference.

For each data set, we feed N points to the classifier, and each time we test the performance on 500 held out test points. For each value of training set size N tested, we record the mean and the standard error of the percentage of test errors, over the ten data sets of the same type. These are plotted in Figure 1 (c) and (d) for the dataspace NN and the compressive NN classifiers respectively. The error bars represent one standard error w.r.t. the random draw of the training set fed to the classifier, as estimated from the ten independently drawn data sets for each experiment. For the experiments with compressive NN, we set the compressed dimension as suggested by the theory, to a multiple of $\gamma^2(T)$ – in this case to twice the dimension of the subspace containing the data, that is $k = 4$.

The results nicely confirm the expectations suggested by the theory. Note the low error rates achieved with only less the 50 points on these 100-dimensional data. This is of course because the data lies in a 2D subspace, which is a very low complexity domain. Also, it is most apparent that data with a sparse representation makes both compressive and dataspace nearest neighbour learning easier, since the error rates are lower. Moreover, both compressive NN and dataspace NN have very similar error profile, and comparable performance. Even their error bars are comparable. The latter observation may seem a bit surprising, however recall that k does not go below the intrinsic dimension of the data – hence effectively we see no performance degradation, and the gain in computation speed comes essentially for free.

Next, we did an analogous set of experiments with data generated to lie in a nonlinear subspace of the $d = 100$ dimensional ambient space. We generated ten data sets where the input points lie on a 2D swiss roll embedded in d dimensions, and an example with $d = 3$ is shown in Figure 2 (a). As before, we then created another ten data sets where again the input points lie of the 2D swiss roll, and in addition they have a sparse representation – see Figure 2 (b). With training and testing set sizes as previously, we recorded the error of NN and compressive NN on Figures 2 (c) and (d). For the latter, we have again set $k = 4$. We see the results are again in accordance with the theory. Both dataspace NN and its compressive counterpart behave in the same way, and in particular, sparse representation makes learning easier for both. This setting of k is appropriate for the particular low complexity data set, and no performance degradation is apparent for learning NN in the compressive space.

Finally, we demonstrate a set of experiments to see the effect of sparse representation. That is, the data lies on a union of linear subspaces of dimension equal to the number of nonzeros in the sparse representation. In comparison, we also show the corresponding performances when the data is contained by a single linear subspace.

In the former case, the input data is in the full 100-dimensional space but it has a sparse representation. We vary the level of sparsity (s), i.e. the number of nonzeros per input point. In the case of compressive NN, we set $k = 2s\log(d/(2s))$, which is an upper bound on $\gamma^2(T)$ for sparse data (Boucheron et al. , 2013), which comes from the fact that sparse data with s non-zeroes in each point may be viewed as data that lies in a union on s-dimensional subspaces. Figure 3) (a)-(b) gives the average of test error estimates over ten independent realisations of the data set for each experiment. Most apparently, the error behaviour of both NN and compressive NN is again in agreement. Though we see that with small sample sizes small errors can only be achieved when the input data has a very sparse representation (i.e. very small non-zeroes in each input point), and the error grows close to random guessing s increases.

The corresponding results with data is in one subspace are seen in Figure 3) (c)-(d). The dimension of this subspace is set to the same values (s) as the levels of sparsity was earlier. That is, we have the same number of nonzero entries as before, but this time to location of these stays the same for each data point. Because of this the metric size, $\gamma^2(T)$, is smaller, namely of the order s, and we set k accordingly. We see the empirical errors are indeed lower and increase slower with s in comparison with the sparse representation case, under the same sample size conditions. That is, the metric size of the input space is indeed well predictive of the statistical difficulty of the NN classification task.

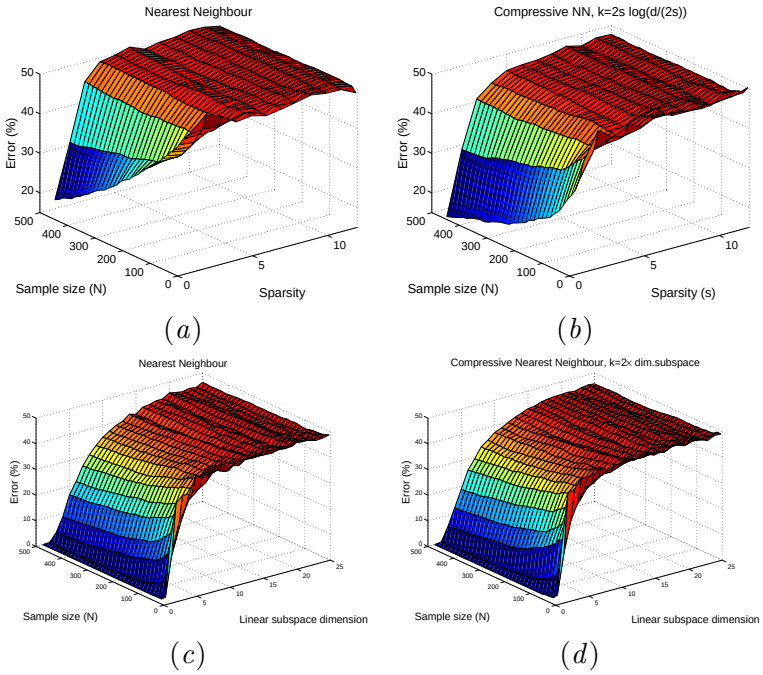


Figure 3: (a)-(b): NN and Compressive NN performance on 100-dimensional data sets that have a sparse representation, when the degree of sparsity is varied. (c)-(d): NN and Compressive NN performance on 100-dimensional data sets that lie of a single linear subspace, when the subspace dimension varies. We see that in the latter is an easier problem than the former. This is nicely captured by the values of $\gamma(T)$.

5. Conclusions

We gave the sample complexity of nearest neighbour classification as a function of the metric complexity of the input space. This agrees with the previously known exponential sample complexity in the input dimension when the input domain fills the ambient space, but it can be less for a number of low-complexity input domains. We used random projections as an analytic tool to uncover these fortuitous structures. Hence, intrinsically low complexity data sets can be efficiently learned from in the compressive space. A further implication of our analysis for theoretical research is the conjecture that the task of determining conditions that allow for a reduction in the sample complexity of the dataspace classifier, and the task of determining conditions that allow learning from compressive data may be viewed and tackled as two sides of the same coin. Further work is needed to test this conjecture in other learning settings. Further work is also needed to study the effects of a small departures from the identified fortuitous structures.

References

A.S. Bandeira, D.G. Mixon, B. Recht. Compressive classification and the rare eclipse problem. CoRR abs/1404.3203 (2014)

R. Baraniuk, M. Wakin. Random projections of smooth manifolds. Foundations of Computational Mathematics, 2007.

S. Boucheron, G. Lugosi, P. Massart. Concentration Inequalities: A Nonasymptotic Theory of Independence, Oxford University Press, 2013.

M. Cover, P. E. Hart. Nearest neighbor pattern classification. IEEE Transactions on Information Theory, 13:21-27, 1967.

K. Chaudhuri, S. Dasgupta. Rates of convergence for nearest neighbor classification. Neural Information Processing Systems (NIPS), 2014.

K. Clarkson. Tighter bounds for random projections of manifolds. Computational Geometry, 2007.

R.J. Durrant, A. Kabán. Sharp Generalization Error Bounds for Randomly-projected Classifiers. ICML 2013: 693-701.

P. Indyk, A. Naor. Nearest Neighbor Preserving Embeddings. ACM Transactions on Algorithms, vol.3, no.3, Article 31, 2007.

L. Gottlieb, A. Kontorovich, R. Krauthgamer. Adaptive Metric Dimensionality Reduction. ALT 2013.

L. Gottlieb, A. Kontorovich, P. Nisnevitch. Near-optimal sample compression for nearest neighbors, NIPS 2014.

A. Kabán. New Bounds on Compressive Linear Least Squares Regression. AISTATS 2014: 448-456

S. Kpotufe, S. Dasgupta. A tree-based regressor that adapts to intrinsic dimension. Journal of Computer and System Sciences, 78(5): 1496-1515, 2012.

B. Klartag, S. Mendelson. Empirical Processes and Random Projections. Journal of Functional Analysis 225, 2005, pp. 229-245.

T. Sarlós. Improved approximation algorithms for large matrices via random projections, 47th Annual IEEE Symposium on Foundations of Computer Science (FOCS), 2006,pp.143-152.

S. Shalev-Shwartz, S. Ben-David. Understanding Machine Learning: From Theory to Algorithms, Cambridge University Press, 2014.

R. Urner. Learning with non-Standard Supervision. PhD thesis, University of Waterloo, 2013.

N. Verma. A note on random projections for preserving paths on a manifold. UC San Diego, Tech. Report CS2011-0971, 2011.

N. Verma, S. Kpotufe, S. Dasgupta. Which spatial partition trees are adaptive to intrinsic dimension? Twenty-Fifth Conference on Uncertainty in Artificial Intelligence (UAI), 2009.