Random Projection for the k-NN Rule : A Theoretical Report

k-NN ルールのためのランダムプロジェクション:理論的考察

Random Projection for the k-NN Rule: A Theoretical Report

シュシマ　クマリ[1]

Sushma Kumari

概要

Random projection is one of the most computationally efficient dimension reduction techniques having low error rates and theoretical guarantees to approximately preserve the distances. In applied domain, random projection boosts the performance of the k-NN rule on high-dimensional data. However, there is no theoretical reasoning as to why random projection based k-NN rule has improved error rates. In this article, we review the method of combining random projection for the k-NN rule, and further present a few related questions.

1 Introduction

To provide accurate information, data with large number of attributes are being recorded on a daily basis in various fields of data analysis, remote sensing and bioinformatics. The number of attributes measured for an observation is its dimension. The primary objective of a data analysis problem is to process and extract valuable information from data but high dimensionality of the data poses several challenges to the accuracy and efficiency of an algorithm.

In high-dimensional space, many traditional machine learning algorithms such as the k-NN rule breaks down due to the *curse of dimensionality*. The curse of dimensionality, a term coined by Bellman in 1961, is a set of all the counter-intuitive phenomenon encountered in the high-dimensional settings which are otherwise not present in the low-dimensional settings. Two of the main obstacles that we face in the high-dimensional space are: sparsity and proximity of the data. The data sparsity problem can be understood by a simple example of finding a sample representation of $[0, 1]^d$. For $d = 1$, the given space is an interval and 10 samples is a very well representation of it. In two-dimensional space, 10 samples will still be a reasonable representation. It can be seen from the Figure 1 that data are scattered and scanty in three-dimensional space. As dimension grows further, the volume of the space grows exponentially causing data to become sparse. It means that the data must grow exponentially for an accurate representation of the space. This is also called the Hughes effect, which essentially states that a sub-exponential growth

[1] 武蔵野大学工学部数理工学科

of data points is required to maintain the consistency of many data algorithms. On the other hand, a distance function loses its ability to distinguish two different objects in high-dimensional space because of another aspect of curse of dimensionality called the distance concentration. As a consequence, all data points are almost equidistant and the notion of 'proximity' is lost. A detailed study on distance concentration for various metrics can be found in [8].

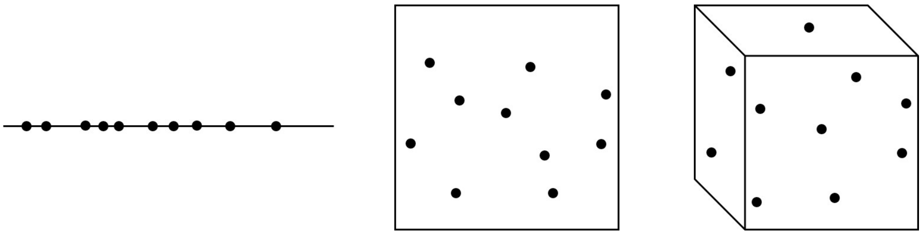

図 1　An illustration of the curse of dimensionality

Due to distance concentration, almost all distance-based machine learning algorithms, for example the k-NN rule become unsuitable for the high-dimensional settings and thus, reducing the data dimension is an essential need. Considering that our intuition and analysis in low-dimensional space are more developed, in this regard, several dimension reduction techniques such as Principal Component Analysis (PCA) and Linear Discriminant Analysis (LDA) were invented to find a low-dimensional approximation of high-dimensional data in order to avoid the curse of dimensionality. However, the computational cost of implementing these conventional techniques is still very high on high-dimensional data. The drawbacks of these techniques led to the development of a more computationally efficient method called random projection. Random projection [12] projects the original high-dimensional data onto a low-dimensional subspace while approximately preserving pairwise distances as follows from the Johnson-Lindenstrauss lemma. Generally random projection is used as a data pre-processing step in combination to the k-NN rule, which makes the algorithm processing fast and also increases its applicability on high-dimensional data. For all that, random projection is relatively a new technique and the link between its experimental success and its theoretical properties is mostly unexplained. This is the motivation of this article. Here, we investigate theoretical questions originated while examining the combination of random projection and k-NN rule. We would like to make a note that this article does not present any new findings but is an attempt to study the problems of random projections from a theoretical perspective.

This paper is organized in the following way: we first give an introduction to random projection in the Section 2, followed by an overview of statistical learning fundamentals and k-NN learning rule in the Section 3. We then present a random projection based k-NN learning model in the Section 4 and conclude this article by presenting a few research directions in the Section 5.

2 Random projection and its developments

Random projection is a powerful technique because of its simple design, low computational cost, and solid theoretical foundations. It is an effective method against the curse of dimensionality and has numerous applications in compressed sensing, machine learning and many data mining tasks. The idea of random projection is based on the Johnson-Lindenstrauss lemma, which states that a set of n data points can be projected to a lower-dimensional space in such a way that the pairwise distances are preserved with high probability.

Theorem 1 (Johnson-Lindenstrauss lemma [12]). *Let X be a set of n data points in Euclidean space $(\mathbb{R}^d, ||.||)$ and let $\varepsilon \in (0,1)$. For $p \geq O(\varepsilon^{-2}\log n)$, there is a linear function $f : \mathbb{R}^d \to \mathbb{R}^p$ such that*

$$(1-\varepsilon)||x-y||^2 \leq ||f(x)-f(y)||^2 \leq (1+\varepsilon)||x+y||^2, \textit{ for every } x, y \in X.$$

The Johnson-Lindenstrauss lemma essentially says that we can choose the target dimension p independent of original dimension d such that the relative distance between data points are preserved up to a factor of $(1 \pm \varepsilon)$. An elementary proof of the Johnson-Lindenstrauss lemma was given by Dasgupta and Gupta [4] which improved the bound on p as given by,

$$p \geq \frac{4}{(\varepsilon^2/2) - (\varepsilon^3/3)} \log n.$$

The expression $p \geq O(\varepsilon^{-2}\log n)$ provides a very important theoretical bound on the projecting dimension and had been a motivation for many works done in this direction [4, 1].

Another interesting remark is that the lemma shows the existence of a projection mapping but does not state its explicit definition. A brief review of literature show that orthogonal random matrices satisfy the conditions of lemma and give desirable transformation. In particular, random projection can be primarily interpreted as a random matrix and low-dimensional projection is achieved by simple matrix multiplication. Algorithmically, random projection can be implemented in the following way:

(i) Let X, a $n \times d$ matrix, be the given data of n sample points in a d-dimensional space $\mathbb{R}^d$.

(ii) We choose a suitable projecting dimension p according to the bounds stated in the Johnson-Lindenstrauss lemma (p is very small compared to d), and construct a random matrix R of dimension $d \times p$.

(iii) Then, the p-dimensional projection is defined as $Z = XR$.

The above projection of the d-dimensional data to a p-dimensional data Z has the computational complexity of order $O(dpn)$. Indeed, there are some computations involved in calculating and storing projection matrix R, but the computational cost can be reduced further if R is sparse. In simpler words, if R has only t non-zero entries, then the complexity is of order $O(tpn)$. Therefore, a number of studies were focused on making the random projection matrix more sparse to exploit its computational advantages. We present some sparse random projection matrices which were constructed based on the Johnson-Lindenstrauss lemma.

(i) *Gaussian projection matrix*: The simplest projection matrix can be constructed by choosing each entry r_{ij} of projection matrix R as an independently identically distributed standard Gaussian variable. Note that this is not a sparse matrix.

(ii) *Achlioptas' matrix* [1]: Initially, Achlioptas constructed a projection matrix with entries picked from $1, -1$ with probability $1/2$ each. Further, Achlioptas constructed a matrix with only one-third non-zero entries, that is, R has its entries distributed as

$$r_{ij} = \begin{cases} \sqrt{3} & \text{with probability } 1/6, \\ \sqrt{0} & \text{with probability } 2/3, \\ -\sqrt{3} & \text{with probability } 1/6 \end{cases}$$

(iii) The Achlioptas' matrix was generalized by Li et al. [10] to present a more sparse projection matrix. Let t be the sparse factor, then R has entries as,

$$r_{ij} = \begin{cases} \sqrt{t} & \text{with probability } 1/2t, \\ -\sqrt{t} & \text{with probability } 1/2t, \\ \sqrt{0} & \text{with probability } 1 - 1/t, \end{cases}$$

For $t = 2, 3$, we obtain matrices provided by Achlioptas. Li et al. proved that the conditions of the Johnson-Lindenstrauss lemma are still satisfied even when $t = \sqrt{d}$ and random projection entries are sampled with probabilities $\{1/2\sqrt{d}, 1/2\sqrt{d}, 1 - 1/\sqrt{d}\}$ to speed up the projection. In simple terms, higher the dimension, the faster is the projection but at the cost of slow convergence.

The properties of random projection have not been fully exploited so there is a potential scope for improving bound on target dimension p, and to make sparser projection matrices.

Nevertheless, here we focus on the popular application of random projection for the k-NN rule to enhance its suitability on high-dimensional data.

3 The k-Nearest Neighbor (NN) rule

In this section, we present the basic elements of statistical machine learning before discussing the k-NN rule and its consistency properties.

3.1 Fundamentals of the statistical learning

In a classification problem, the goal is to classify a new input based on the previous experiences. For an instance, the task of categorizing new upcoming emails as 'spam' or 'not-spam' is a binary classification problem. An algorithm can be trained to classify a new email based on some hypothesis assumed from a given set of emails labeled as 'spam' and 'not-spam'. Obviously, we expect the classifier function to be as accurate as possible with less error and better classification. Nonetheless there is always some uncertainty involved, therefore probabilistic framework are best for better understanding and further analysis.

Let (Ω, ρ) be a metric space, where Ω is a non-empty domain, also the source of all data points, and ρ is a metric. Let $D_n = \{(X_i, Y_i) \in \Omega \times \{0,1\} : 1 \leq i \leq n\}$ be an i.i.d. labeled random sample and let $(X, Y) \in \Omega \times \{0,1\}$ be a pair of random variables distributed according to μ. Note that, X is a data point having label Y and μ is a probability distribution on $\Omega \times \{0,1\}$, whereas a labeled sample $\sigma_n = \{(x_i, y_i) \in \Omega \times \{0,1\} : 1 \leq i \leq n\}$ is a realization of D_n. A binary classification problem is to find a Borel measurable function (called classifier) such that each data point in Ω is associated with either a label 0 or 1. For distribution μ, we can construct an optimal classifier called the Bayes classifier which has the lowest error (referred as the Bayes error) among all the classifiers,

$$err^*_\mu = \inf_{h:\Omega \to \{0,1\}} \{err_\mu(h) = \mathbb{P}(h(x) \neq y) : h \text{ is a classifier}\}$$

In almost all cases, the data distribution μ is unknown and thus the Bayes classifier cannot be constructed explicitly. We utilize the only available resource: labeled sample, to construct a family of learning rules (or classifiers) such that its error will converge to the Bayes error. A function $h_n : (\Omega \times \{0,1\})^n \times \Omega \to \{0,1\}$ is called a learning rule of size n which assigns a label y to a pair (σ_n, x). We can interpret $h_n(\sigma_n, x) = h_n(\sigma_n)(x)$ as a classifier acting on x to assign label y. The learning error probability of (h_n) is defined as

$$err_\mu(h_n) = \mathbb{P}(h_n(X) \neq Y | D_n),$$

and the average learning error over all labeled samples of size n is given by

$$\mathbb{E}[err_\mu(h_n)] = \mathbb{P}(h_n(X) \neq Y).$$

A learning rule whose error converges asymptotically to the Bayes error is called a consistent rule.

Definition 1. *A learning rule (h_n) is said to be weakly consistent with respect to μ if*

$$\mathbb{E}[err_\mu(h_n)] \to err_\mu^*, \text{ whenever } n \to \infty$$

which is essentially the convergence in probability. Whereas, (h_n) is said to be a strongly consistent rule with respect to μ if

$$err_\mu(h_n) \to err_\mu^* \text{ a.e., whenever } n \to \infty.$$

Additionally, if the weak (strong) consistency holds for all probability measures μ on $\Omega \times \{0, 1\}$, then the learning rule is universally weakly (strongly) consistent.

3.2 The k-NN rule

The k-NN rule is the most important learning rule in statistical learning, popularly studied because of its easy implementation, experimental success, wide applicability and consistency properties. The k-NN rule classifies a new input by taking a majority vote among the labels of its k nearest data points as illustrated in the Figure 2.

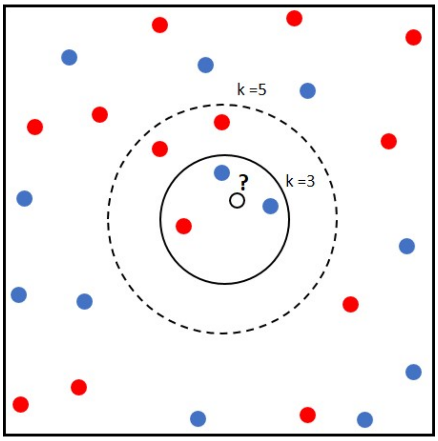

図 2 An example of k-NN: for $k = 3$, the k-NN rule assigns label 'blue' to the new input, whereas for $k = 5$, label 'red' is assigned.

The k-NN algorithm: Let (Ω, ρ) be a metric space. Given a labeled sample σ_n, a query x and integer $k \geq 1$; the problem is to predict a label for x based on σ_n.

(i) First find k number of data points from σ_n which are closest to x according to ρ, let this set be $\sigma_n^{kNN}(x) = \{x_{(1)}, \ldots, x_{(k)}\}$, which is also called the k nearest neighbor set of x.

(ii) Let $\{y_{(1)}, \ldots, y_{(k)}\}$ be the label set for $\sigma_n^{kNN}(x)$, then the k-NN rule decides the label of x as the most frequently occurring label among $\{y_{(1)}, \ldots, y_{(k)}\}$.

The k-NN rule is a proximity based learning rule, that is, data points near to the query has more value than data points at a far distance. So, we assign equal weights $1/k$ to the k nearest neighbors of x and 0 to other data points. The explicit expression of the k-NN rule can be given as,

$$h_n(x) = \begin{cases} 1 & \text{if } \frac{1}{k}\sum_{i=1}^{n} y_i \mathbb{I}_{\{x_i \in \sigma_n^{kNN}(x)\}} \geq 1/2, \\ 0 & \text{otherwise,} \end{cases}$$

where $\mathbb{I}$ is an indicator function. Historically, the k-NN rule was the first ever rule to be proven universally (weakly) consistent.

Theorem 2 (Stone's Theorem [5]). *If $n, k \to \infty$ such that k/n goes to zero, then the k-NN rule is universally weakly consistent on Euclidean space* $(\mathbb{R}^d, ||.||)$.

The main argument of the Stone's theorem was based on the geometry of Euclidean space and hence cannot be extended to general metric spaces. Nevertheless, the mathematical proofs of the Stone's theorem play a crucial role in studying the consistency of various other learning methods and histogram rules. Indeed, the universal strong consistency of the k-NN rule in Euclidean spaces was established by Devroye et al. [5]. Further, Cérou and Guyader [3] established the universally weak consistency of the k-NN rule on separable metric spaces satisfying the Lebesgue-Besicovitch differentiation property (a generalization of the fundamental theorem of calculus). On the other hand, Preiss [11] showed that the Lebesgue-Besicovitch differentiation property is equivalent to the notion of sigma-finite metric dimension of a space. The notion of the strong consistency is still not examined in such spaces. A detailed analysis on the connection between consistency and sigma-finite metric dimension can be found in [7, 2].

4 Random projection for the k-NN rule

The curse of dimensionality causes the pairwise distance to concentrate, and the Euclidean metric is unable to distinguish high-dimensional objects. As a result, it is difficult to find k nearest neighbors of x as all data points seem to be at equal distance, and since k-NN rule is a distance-based algorithm, its performance degrades in high-dimensional space. Thus, random projection is employed to find a low-dimensional approximation of the original space, which is then plugged into the k-NN rule. As guaranteed by the Johnson-Lindenstrauss lemma, if the projection is done to an appropriate low-dimensional space, then the pairwise distance could be preserved up to a good accuracy.

There are several different low-dimensional approximations used for the k-NN rule as well as for other machine learning algorithms, but almost all articles do not explain why this combination will work. Although, Fraiman et al. [6] claims to give a justification to employ random projection for the k-NN rule, but the projections used is questionable. First, we present the random projection-based k-NN as in [6], and then we will analyze its results.

Definition 2 (The *RPk*-NN rule [6]). *We are given a labeled sample* $\sigma_n = \{(x_1, y_1), \ldots, (x_n, y_n)\}$*, query* x *from the Euclidean space* $(\mathbb{R}^d, ||.||)$ *and let* k *be a positive integer.*

(I) Let β *be a positive integer strictly greater than 1 such that* $\beta k \leq n$*.*

(II) Choose βk *number of closest data points to* x *among* σ_n*, let us denote it by* $\sigma_n^{\beta kNN} = \{x_{(1)}, \ldots, x_{(\beta k)}\}$*.*

(III) Find N *random projection directions* $f_1, \ldots, f_N$ *according to a non-degenerate distribution on unit sphere , essentially these directions will project to one-dimensional space*

(IV) For each f_i*, find the number of data points from* $\sigma_n^{\beta kNN}$ *which after being projected by* f_i *is among the* k *nearest neighbor set of* $f_i(x)$*. That is for each* $1 \leq i \leq N$*, find*

$$A_i = \sum_{x_{(j)} \in \sigma_n^{\beta kNN}, 1 \leq j \leq \beta k} \mathbb{I}_{\{f_i(x_{(j)}) \in \sigma_n^{kNN}(f_i(x))\}},$$

where $\mathbb{I}$ *is an indicator function taking values in* $\{0, 1\}$*. In other words, we count the number of* βk *nearest neighbor of* x*, which appears in the* k *nearest neighbor set of projected query* $f_i(x)$*.*

(V) Divide the above calculated set of $f_i(x_{(j)} \in \sigma_n^{kNN}(f_i(x))$ *into two label sets, data points having label 0, say* A_i^0 *and label 1 as* A_i^1*. Further, find its average over all projection directions,*

$$avg(A_i^0) = \frac{1}{N} \sum_{i=1}^{N} A_i^0, , \text{ and } avg(A_i^1) = \frac{1}{N} \sum_{i=1}^{N} A_i^1$$

(VI) The query x *is assigned label 1 if* $avg(A_i^0) \leq avg(A_i^1)$*, otherwise 0.*

An explicit definition of the *RPk*-NN rule can be formulated in the similar way as the k-NN rule by assigning weights $1/k$ to βk nearest neighbours of x whose one-dimensional projections are also the k nearest neighbors of projected x. The *RPk*-NN rule is fast with improved computational time, and is supplemented by theoretical consistency results.

Theorem 3 ([6]). *The RPk-NN rule is universally weakly consistent if $k/n \to 0$, as $k, n \to \infty$.*

It is also claimed that the RPk-NN rule is strongly consistent for μ with density. On the other hand, it is important to note that one-dimensional projections, according to the Johnson-Lindenstrauss lemma, may lead to high distortion. In particular, it can be derived from the Fraiman et al.'s argument that two concentric spheres in high-dimensional space can be disconnected by one-dimensional projections, which imply that the proof of consistency by Fraiman et al. is erroneous. A full argument of the falseness of this claim is given in [9].

5 Future prospects

Using random projection, the high-dimensional data is projected to a low-dimensional subspace, and the k nearest neighbors are sought in low-dimensional space to make a prediction. Experimental results suggest that random projection based k-NN rule is fast and computationally inexpensive, but provides no theoretical reasoning for its empirical success. The only result in this direction by Fraiman et al. seems to be defective. The k-NN rule is a universally consistent rule and as random projection (in accordance to the Johnson-Lindenstrauss lemma) approximately preserves the structure of original data, intuitively it can be expected that consistency of random projection based k-NN is also preserved. It is important to understand from a theoretical point of view as why does random projection improves the performance of the k-NN rule. Is random based k-NN rule a consistent rule? One approach to this problem is to first define a correct mathematical model for random projection based k-NN rule, and then examining its consistency based on the Stone's theorem.

Random projection have also been successfully applied with other learning algorithms such as Support Vector Machines (SVMs) and random forest classifiers, but again, the experimental results are sparse with no theoretical reasoning. With some adaptations, the SVMs and random forest classifiers can be considered as consistent rules, an interesting direction is to investigate the combination of random projection with other modern learning rules, SVMs and random forests both from an empirical as well as theoretical point of view. Will these random projection based algorithms be consistent? Random projection is a very efficient method but many theoretical properties are unexplored and a solid theoretical framework of random projection for learning rules is lacking. We hope the drawbacks identified in this article will motivate a deeper theoretical analysis of random projection to get a better understanding of its characteristics and will also broaden its applicability.

Acknowledgements

The author acknowledges the support from JSPS KAKENHI 19K20347. The author is thankful to the anonymous referees of the paper whose comments have helped to improve the manuscript.

References

[1] Achlioptas, D.; Database-friendly random projections:Johnson-Lindenstrauss with binary coins, *Journal of Computer and System Sciences* 66, 671–687 (2003).

[2] Collins, B., Kumari, S., Pestov, V.; Universal consistency of the k-NN rule in metric spaces and Nagata dimension, *ESIAM: Probability and Statistics*, 24,914–934 (2020).

[3] Cérou F., Guyader A.; Nearest Neighbor Classification in infinite dimension, *ESAIM: Probability and Statistics* , vol. 10, pp. 340-355 (2006).

[4] Dasgupta, S., Gupta, A.; An elementary proof of a theorem of Johnson and Lindenstrauss, *Random Structures & Algorithms*, 22(1), 60–65 (2003).

[5] Devroye, L., Györfi, L., Lugosi, G.; A Probabilistic Theory of Pattern Recognition, *Stochastic Modelling and Applied Probability*, Springer (1996).

[6] Fraiman, R., Justel, A., Svarc, M.; Pattern recognition via projection-based kNN rules, *Computational Statistics & Data Analysis*, 54(5), 1390–1403 (2010).

[7] Kumari, S.; Topics in Random Matrices and Statistical Machine Learn- ing, *Ph.D. thesis, Department of Mathematics, Kyoto University* (2018).

[8] Kumari, S., Jayaram, B.; Measuring Concentration of Distances-An Effective and Efficient Empirical Index, *IEEE Transactions on Knowledge and Data Engineering*, 29(2), 373–386 (2017).

[9] Kumari S.; An Analysis of the Random Projection Method for Dimensionality Reduction, *A Kyoto University Top Global Unit report, Department of Mathematics, Kyoto University* (2018).

[10] Li P., Hastie T. J., Church K. W.; Very sparse random projections, *In KDD ' 06: Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 287–296 (2006).

[11] Preiss D.; Dimension of metrics and differentiation of measures, *Heldermann*

Verlag, Berlin, pp. 565-568 (1983).

[12] Vempala, S.; The Random Projection Method, *Series in Discrete Mathematics and Theoretical Computer Science*, vol. 65. (2004).

(原稿提出: 2021 年 1 月 12 日; 修正稿提出: 2021 年 2 月 7 日)