On the effects of dimensionality on data analysis with neural networks

M. Verleysen[1], D. François[2], G. Simon[1], V. Wertz[2*]

Université catholique de Louvain
[1] DICE, 3 place du Levant, B-1348 Louvain-la-Neuve, Belgium
[2] CESAME, 4 av. G. Lemaitre, B-1348 Louvain-la-Neuve, Belgium
{verleysen, simon}@dice.ucl.ac.be, {francois, wertz}@auto.ucl.ac.be

Abstract. Modern data analysis often faces high-dimensional data. Nevertheless, most neural network data analysis tools are not adapted to high-dimensional spaces, because of the use of conventional concepts (as the Euclidean distance) that scale poorly with dimension. This paper shows some limitations of such concepts and suggests some research directions as the use of alternative distance definitions and of non-linear dimension reduction.

1. Introduction

In the last few years, data analysis has become a specific discipline, sometimes far from its mathematical and statistical origin, where understanding of the problems and limitations coming from the data themselves is often more valuable than developing complex algorithms and methods. The specificity of modern data mining is that *huge* amounts of data are considered. There are new fields where data mining becomes crucial (medical research, financial analysis, etc.); furthermore, collecting huge amount of data often becomes easier and cheaper.

A main concern in that direction is the *dimensionality* of data. Think of each measurement of data as one observation, each observation being composed of a set of variables. It is very different to analyze 10000 observations of 3 variables each, than analyzing 100 observations of 50 variables each! One way to get some feeling of this difficulty is to imagine each observation as a point in a space whose dimension is the number of variables. 10000 observations in a 3-dimensional space most probably form a structured shape, one or several clouds, from which it is possible to extract some relevant information, like principal directions, variances of clouds, etc. On the contrary, at first sight 100 observations in a 50-dimensional space do not represent anything specific, because the number of observations is too low.

Nevertheless, many modern data *have* this unpleasant characteristic of being high-dimensional. And despite the above difficulties, there *are* ways to analyze the data,

* MV is a Senior research associate at the Belgian FNRS. GS is funded by the Belgian FRIA. The work of DF and VW is supported by the Interuniversity Attraction Pole (IAP), initiated by the Belgian Federal State, Ministry of Sciences, Technologies and Culture. The scientific responsibility rests with the authors.

J. Mira (Ed.): IWANN 2003, LNCS 2687, pp. 105-112, 2003.

and extract information from observations. If the current data analysis methodologies are not adapted to high-dimensional, sparse data, then it is our duty to develop adapted methods, even if some well-admitted concepts must be questioned. In particular, artificial neural network methods, now widely and successfully used in data analysis, should be faced to high-dimensional data and modified if necessary.

This paper makes no pretence of presenting generic solutions to this problem; the current state-of-the-art is far from that. However, we will illustrate some surprising facts (Section 2) about high-dimensional data in general, and about the use of neural networks in this context (Section 3). In particular, we will show that the use of standard notions as the Euclidean distance, the nearest neighbor, and more generally similarity search, is not adapted to high-dimensional spaces. There is thus a need for alternative solutions; this paper only gives paths to future developments (Section 4), to a new research activity that could influence considerably the field of neural networks for data mining in the next few years.

2. Some weird facts about high-dimensional space

High dimensional spaces do in fact escape from our mental representations. What we take for granted in dimension one, two or three, because we can figure it out quite easily, might not actually hold in higher dimensions. Let's highlight some weird facts.

2.1. The empty space phenomenon

Scott and Thompson [1] first noticed some counter-intuitive facts related to high dimensional Euclidean spaces, and described what they called the "empty space phenomenon".

Fact 1. The volume of a hyper-sphere of unit radius goes to zero as dimension growths. The volume of a sphere of radius r in d dimensions is given by:

$$V(d) = \frac{\pi^{d/2}}{\Gamma(d/2+1)} r^d \,. \qquad (1)$$

Figure 1a shows the volume for $r = 1$; we see that the volume rapidly decreases with d. So, in higher dimension, a unit sphere is nearly empty.

Fact 2. The ratio between of the volumes of a sphere and a cube of same radius tend towards zero with increasing dimension as illustrated in Figure 1b. In one dimension these volumes are equal, and in two dimensions the ratio is approximately 0.8, but in higher dimension we can say that the volume of a hyper-cube concentrate in its corners.

Fact 3. The ratio of volume of a sphere of radius 1 and 1-ε tend towards zero given the obvious fact that its value is equal to (1-ε) to the power d. With d as small as 20, and $e =$ 0.1, only 10% of the original radius contains 90% of the volume of the outer sphere, and so the volume of it concentrates in an outer shell. The same holds for hyper-cubes and hyper-ellipsoids as well.

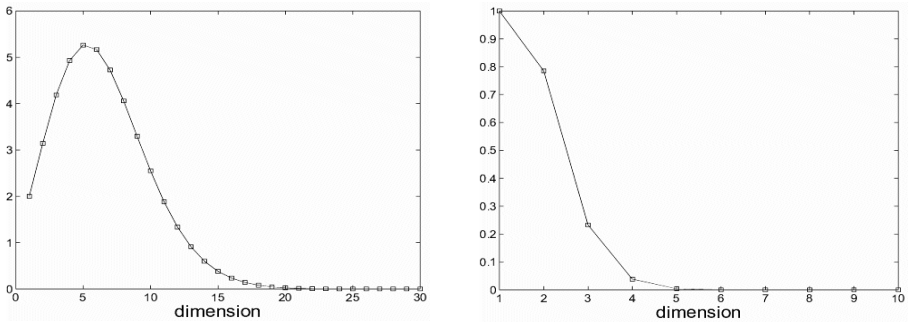

Fig. 1. Left (a): volume of the unit sphere; Right (b): ratio between the volumes of the unit sphere and the unit cube, with respect to the dimension of the space.

These observations imply that high-dimensional spaces are mostly empty. They indeed show that local neighborhoods of points are mostly empty, and that even in the case of uniform distributions, data is concentrated at the borders of the volume of interest.

2.2. The concentration of measure phenomenon

We will now have a deeper look at the behavior of the widely used Euclidean distance (i.e. the L_2-norm of the difference) when applied to high dimensional vectors.

Fact 1. The standard deviation of the norm of random vectors converges to a constant as dimension increases though the expectation of their norm growths as the square root of the dimension. More precisely, it has been proven in [2] that under i.i.d. assumption on x_i,

$$\mu_{\|x\|} = \mathrm{E}\left(\|x\|\right) = \sqrt{an - b} + \mathrm{O}(1/n) \tag{2}$$

$$\sigma^2{}_{\|x\|} = \mathrm{Var}\left(\|x\|\right) = b + \mathrm{O}\left(1/\sqrt{n}\right) \tag{3}$$

where a and b are constants depending only on the four first momentums of x_i.

Note that the same law applies to the Euclidean distance between any two points, since it happens to be a random vector too.

Fact 2. The difference between the distances of a randomly-chosen point to its furthest and nearest neighbor decreases as dimensionality increases. This can be illustrated by the asymptotic behavior of the relative contrast [3] :

$$\text{If} \quad \lim_{d\to\infty} \mathrm{Var}\left(\frac{\|x\|_k}{E\left(\|x\|_k\right)}\right) = 0 \quad \text{then} \quad \frac{Dmax^k - Dmin^k}{Dmin^k} \xrightarrow[p]{} 0 \tag{4}$$

where D_{min} and D_{max} are the distance to respectively the nearest and furthest neighbors of a particular point. Note that the hypothesis of the theorem is induced by equation (3). A more general proof of the theorem can be found in [4].

The conclusion we can draw from these observations is that, in high dimensional spaces, all points tend to be equally distant from each others, with respect to the Euclidean distance. As dimension increase, the observed distance between any two points tends towards a constant. This can be illustrated when computing the histograms of distances between random points of increasing dimensionality. It appears that (1) the mean of the histogram growths and (2) its variance shrinks.

2.3. The curse of dimensionality

Finally let us have a look at a not-so-weird-but-often-ignored fact, which Richard Bellman named "The Curse of Dimensionality" [5]. It refers to the huge amount of points that are necessary in high dimensions to cover an input space, for example a regular grid spanning a certain portion of the space. When filling an hypercube in 5 dimensions ($[0\ 1]^5$) with a 0.1-spaced grid, one needs no less than 100.000 points.

3. Consequences for neural network learning

The considerations developed in the previous section have important consequences on ANN (Artificial Neural Networks) learning. The following subsections give examples of such consequences in specific contexts.

3.1. Supervised learning

When modeling some process producing an output on basis of observed values for particular inputs, one has to fit a chosen model to a dataset. The more extensive the dataset, the more accurate is the model. Ideally, the dataset should span the whole input space of interest, in order to ensure that any predicted value (i.e. output of the model) is the result of an interpolation process and that no hazardous extrapolation occurs.

But one has to face the curse of dimensionality. Silverman [6] addressed the problem of finding the necessary number of training points (samples) to approximate a Gaussian distribution with fixed Gaussian kernels. His results show that the required number of samples grows exponentially with the dimension. Fukunaga [7] obtained similar results for the k-NN classifier showing that whereas 44 observations are sufficient in 4 dimensions, not less than $3.8e^{57}$ are necessary when dimension is 128.

3.2. Local models

Local artificial neural networks are often argued to be more sensitive to dimensionality than global ones. By local models, we mean approximators (or classifiers, or density estimators) made of a combination of local functions (for example Gaussian kernels). Indeed Gaussian functions also have an unexpected

behavior when extended to high-dimensional spaces. Examples of such approximators are RBFN (Radial-Basis Function Networks) and kernel methods.

When a normal distribution with standard deviation σ is assumed, the probability density function to find a point at distance r from the center of the distribution is given by [8] :

$$f(r) = \frac{r^{d-1}}{2^{d/2-1}} \frac{e^{-r^2/2\sigma^2}}{\Gamma(d/2)}, \qquad (5)$$

which is maximum for $r/\sigma = (n\text{-}1)^{0.5}$. In one dimension, it is maximum at the center of the distribution, as expected, but when dimension growths, it diverges from the center (see Figure 2), which become nearly empty, whereas the Gaussian distribution is maximal ! This shows that Gaussian kernels are not local any more in higher dimensions, and that models that have been seen as sums of local kernels do not behave as such in high dimensions.

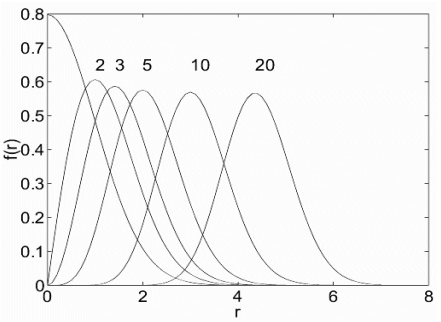

Fig. 2. Probability density of a point from a normal distribution to be at distance *r* of the center, for several space dimensions.

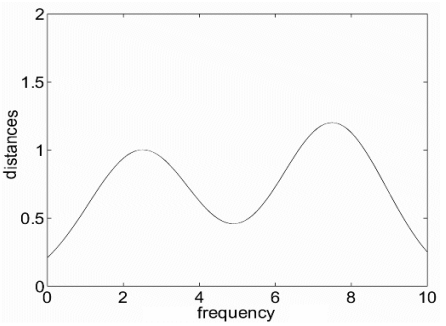

Fig. 3. Example of distance histogram in a several-clusters distributions.

This limitation to the use of local models, in particular with Gaussian kernels, seems severe. However, it should be emphasized on the fact that global models, as for example MLP (Multi-Layer Perceptrons), probably equally suffer. Indeed, in many cases, sums of sigmoids as in MLP result in functions taking significant values in a limited region of the spaces. While mathematically different, models as MLP and RBF thus often behave similarly in practice. This enforces the conviction that local and global models equally suffer from the curse of dimensionality (and related effects), while this is probably harder to prove for global models.

3.3. Similarity search and Euclidean distances

Most neural network models, as well as clustering techniques, rely on the computation of distances between vectors. For RBFN, it is the distance between a data and each kernel center. For MLP, it is the scalar product between a data and

each weight of the input layer. Both these distance measures may be related to the similarity search in clustering techniques, also used in vector quantization, LVQ, Kohonen maps, etc. Similarity search consists in finding in a dataset the closest element to a given point. In the context of clustering for example, efficient clustering is achieved when data in a cluster are similar (i.e. close with respect to the distance function) and data in different clusters are far away from each other. So, when data contain clusters, distance histograms should ideally show two peaks (as in Figure 3) : one for intra-cluster distances, and the other for extra-cluster distances. But if the distance histogram only contains one peak, or if the two peaks are close, distance-based clustering will be difficult. Unfortunately, the fact is that in high dimensions, any distance histogram tends towards a more and more concentrated peak, making the clustering task uneasy [9]. This is a direct consequence of the concentration of measure phenomenon.

4. Towards solutions

Effects of the curse of dimensionality and related limitations on neural network learning seem unavoidable in high-dimensional spaces. There are however at least two paths to explore to remedy to this situation.

4.1. Alternative distance measures

The use of the Euclidean distance between data is conventional and is rarely discussed. However, it is not obvious that another definition of distance could not be more appropriate in some circumstances, and in particular in high-dimensional spaces. In practice, any distance measure between vectors x and y (with components x_i and y_i) of the following form could be considered:

$$\|x - y\|_r = \sqrt[r]{\sum_{i=1}^{d} |x_i - y_i|^r} \quad . \tag{6}$$

In practice, (4) is applicable for any positive value of r; the asymptotical behavior of any distance definition as (6) is the same (i.e. all distances are subject to the concentration phenomenon). But the convergence rate of (4) differs for different values of r.

The intuition tells that using high values of r can mitigate the effects of loss of locality for Gaussian-like kernels. Nevertheless it has been shown [3] that lower values of r can keep the relative contrast (4) high (for a particular dimension). Unfortunately there is no known reason (other than numerical computation-related arguments) to find a lower bound for optimal r. And there is no sense in taking r = 0...Therefore it remains to find the proper to set a lower bound for r, so that an optimal and most probably dimension-dependant compromise can be found.

4.2. Non-linear projection as preprocessing

Another way to limit the effects of high dimensionality is to reduce the dimension of the working space. Data in real problems often lie on or near submanifolds of the input space, because of the redundancy between variables. While redundancy is often a consequence of the lack of information about which type of input variable should be used, it is also helpful in the case where a large amount of noise is unavoidable on the data, coming for example from measures on physical phenomena. To be convinced of this positive remark, let us just imagine that the *same* physical quantity is measured by 100 sensors, each of them adding independent Gaussian noise to the measurement; averaging the 100 measures will strongly decrease the influence of noise on the measure! The same concept applies if n sensors measure m quantities ($n > m$).

Projection of the data on submanifolds may thus help. A way to project data is to use the standard PCA (Principal Component Analysis). However PCA is linear; in most cases, submanifolds are not linear (think for example to a horseshoe distribution, as in [10]) and PCA is not efficient.

Alternative nonlinear methods exist to project data in a nonlinear way. Examples are Kohonen self-organizing maps (usually to project data onto one- or two-dimensional spaces), and methods based on distance preservation. The latter include Multi-dimensional scaling [11-12], Sammon's mapping [13], Curvilinear Component Analysis [14] and extensions [15]. All these methods are based on the same principle: if we have n data points in a d-dimensional space, they try to place n points in the m-dimensional projection space, keeping the mutual distances between any pair of points unchanged between the input space and the corresponding pair in the projection space. Of course, having this condition strictly fulfilled is impossible in the generic case (there are $n(n-1)$ conditions to satisfy with nm degrees of freedom); the methods then weight the conditions so that those on shorter distances must be satisfied more strictly than those on large distances. Weighting aims at conserving a local topology (locally, sets of input points will resemble sets of output points).

An example of successful application of the above approach in the context of financial prediction can be found in [16].

5. Conclusion

Theoretical considerations show that using classical concepts in data analysis with neural networks to process high-dimensional data may be not appropriate. The reason is that some of the underlying hypotheses, though obvious in lower dimension, are not verified any more in higher dimensions. Indeed, in practice, one observes severe performance loss with data processing algorithms when data are high dimensional. There is thus a need to adapt our models to high dimensionality. A way one can think of is to consider new similarity measures between data, other than the ancestral Euclidean distance. Another way is to reduce the dimension through projection on

(non-linear) submanifolds. In both cases, deep investigation is required in order to successfully adapt data processing tools to high dimensional data.

References

1. Scott, D.W., Thompson, J. R.: Probability density estimation in higher dimensions. In: Douglas, S.R. (ed): Computer Science and Statistics. Proceedings of the Fifteenth Symposium on the Interface, North Holland-Elsevier, Amsterdam, New York, Oxford (1983) 173–179
2. Demartines, P. : Analyse de données par réseaux de neurones auto-organisés. Ph.D. dissertation (in French), Institut National Polytechnique de Grenoble - France (1994)
3. Aggarwal, C. C., Hinneburg, A., Keim, D. A.: On the surprising behavior of distance metrics in high dimensional spaces. In: Van den Bussche, J., Vianu, V. (eds): Proceedings of Database Theory - ICDT 2001, 8th International Conference Lecture Notes in Computer Science, vol 1973. Springer, London, UK (2001) 420–434
4. Beyer K. S., Goldstein, J., Ramakrishnan, R., Shaft, U.: When is "nearest neighbor" meaningful? In: Beeri, C., Buneman, P. (eds) : Poceedings of Database Theory - ICDT '99, 7th International Conference. Lecture Notes in Computer Sciences, vol 1540, Springer, Jerusalem, Israel (1999) 217–235
5. Bellmann, R.: Adaptive Control Processes: A Guided Tour. Princeton Univ. Press (1961)
6. Silverman, B.W.: Density estimation for statistics and data analysis. Chapman & Hall (1986)
7. Fukunaga K.: Introduction to Statistical Pattern Recognition. Academic Press, Boston, MA, (1990)
8. Hérault, J., Guérin-Dugué, A., Villemain, P.: Searching for the embedded manifolds in high-dimensional data, problems and unsolved questions. Proceedings of ESANN'2002 - European Symposium on Artificial Neural Networks, d-side public, Bruges - Belgium (2002) 173–184
9. Steinbach, M., Ertoz, L., Kumar, V.: Challenges of clustering high dimensional data. New Vistas in Statistical Physics – Applications in Econo-physics, Bioinformatics, and Pattern Recognition, Springer-Verlag (2003)
10. Verleysen, M.: Learning high-dimensional data. Acc. for public. in Ablameyko, S., Goras, L., Gori, M., Piuri, V. (eds): Limitations and future trends in neural computation, IOS Press.
11. Shepard, R. N.: The analysis of proximities: Multidimensional scaling with an unknown distance function, parts I and II, *Psychometrik*a, 27 (1962) 125-140 and 219-246
12. Shepard, R.N, Carroll, J.D: Parametric representation of nonlinear data structures. In P. R. Krishnaiah (ed.): *International Symposium on Multivariate Analysi*s, Academic Press, (1965) 561-592
13. Sammon, :A nonlinear mapping algorithm for data structure analysis, *IEEE Trans. on Computer*s, C-18 (1969) 401-409
14. Demartines, P., Hérault, J.: Curvilinear Component Analysis: a self-organizing neural network for nonlinear mapping of data sets, *IEEE T. Neural Network*s,. 8-1 (1997) 148-154
15. Lee, J. A., Lendasse, A., Verleysen, M: Curvilinear Distance Analysis versus Isomap. In: Proceedings of ESANN'2002, 10th European Symposium on Artificial Neural Networks, d-side public, Bruges – Belgium, (2002) 185-192
16. Lendasse, A., Lee, J. A., de Bodt, E., Wertz, V., Verleysen, M.: Dimension reduction of technical indicators for the prediction of financial time series - Application to the Bel 20 market index. *European Journal of Economic and Social Systems*, 15-2 (2001), pp. 31-48