Visualizing high dimensional datasets using Partiview

Dinoj Surendran*
Computer Science Department
University of Chicago

Stuart Levy †
Experimental Technologies Group
National Center for Supercomputing Applications
University of Illinois at Urbana-Champaign.

ABSTRACT

A standard method of visualizing high-dimensional data is reducing its dimensionality to two or three using some algorithm, and then creating a scatterplot with data represented by labelled and/or colored dots. Two problems with this approach are (1) dots do not represent data well, (2) reducing to just three dimensions does not make full use of several dimensionality-reduction algorithms. We demonstrate how Partiview can be used to solve these problems, in the context of handwriting recognition and image retrieval.

CR Categories: H.5.m [Information Interfaces and Presentation]: Miscellaneous; I.5.3 [Pattern Recognition]: Clustering; I.5.4 [Pattern Recognition]: Applications;

Keywords: dimensionality reduction, glyphs, high dimensional data visualization, optical character recognition, image retrieval, information visualization

1 INTRODUCTION

Partiview [1] is an interactive 3D visualization tool written by the second author primarily for astronomy-related applications; the first author uses it as an everyday tool in machine learning research. This poster describe two problems in standard methods of visualizing high-dimensional data that can be solved using Partiview.

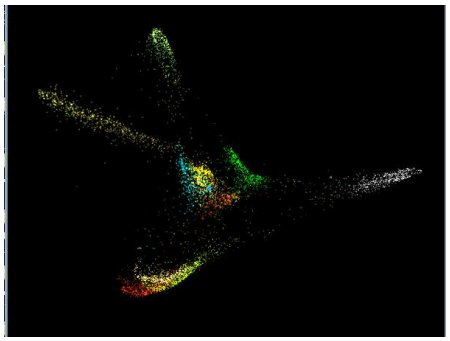

Figure 1: Handwritten digits from the MNIST/LeCun database (available from http://yann.lecun.com/exdb/mnist) clustered by the Laplacian Eigenmaps algorithm [2].

2 PROBLEM 1: DOTS POORLY REPRESENT DATA

For example, Figure 1 shows how well a clustering algorithm works on a dataset of handwritten digits. Different digits are represented by 'points' of different colors. Points of the same color are clustered together, indicating that the clustering algorithm is doing fairly well at recognizing digits. However, it would be useful to examine more closely where the algorithm fails, i.e. where differently-colored digits are near each other.

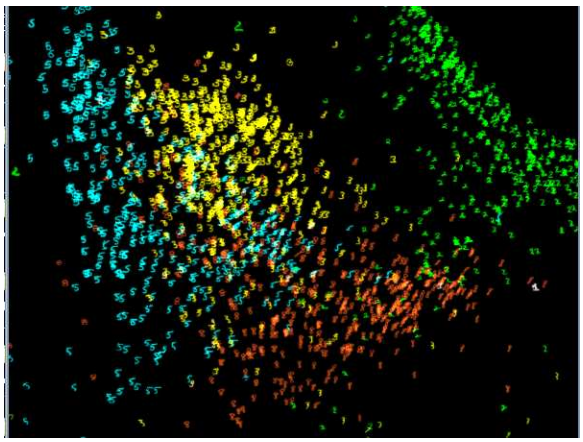

Figure 2: A closeup of the points in the center of Figure 1.

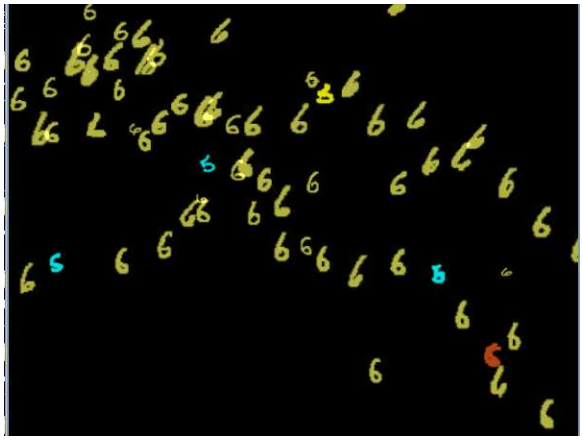

Figure 3: A closeup of the points in one of the 'arms' of Figure 1.

Partiview permits images to be placed at specific 3d coordinates, always facing the user; they can also be placed with a fixed orientation but this is not as useful for information visualization. In the case where each datum is a handwritten digit, we can represent a digit by itself, i.e. an image of the original digit. Our first demo is precisely this; as evidenced from Figures 2 and 3, which are closeups of Figure 1. For example, Figure 3 shows that the algorithm mistakes 5's with closed loops, and 3's with thin tops, as 6's.

In cases where the data has a natural visual representation, such

*e-mail: dinoj@cs.uchicago.edu
†e-mail: slevy@ncsa.uiuc.edu

as images in image retrieval, faces in face recognition, or (snippets of) documents in information retrieval, the choice of images for glyphs is straightforward. For example, Figures 4 and 5 show how well another dimensionality reduction algorithm groups pictures according to their semantic content. The visualization shows that while it achieves some success with grouping certain kinds of images, such as those of flags, it fails with other groups.

Of course, data in several domains have no natural visual correlate. Images would need to be created for each data point. One possibility is to have such images represent summaries of the data, such as a pie chart showing the relative frequency bigrams in a ACGT string in genomics.

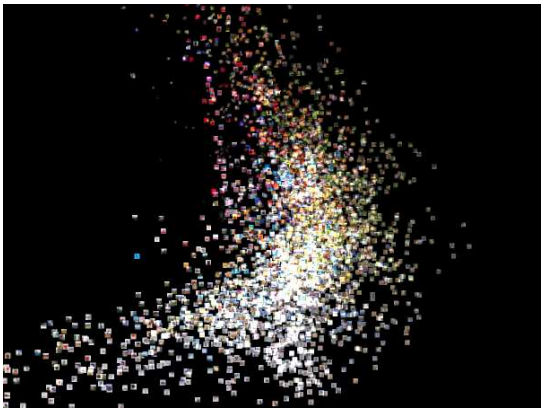

Figure 4: Images from the Corel Image Database clustered using the Locality Preserving Projections algorithm [4].

Figure 5: A close-up of the images in Figure 4.

3 Problem 2: Why reduce to just three dimensions?

If the data's inherent dimensionality is larger than three, reducing the number of dimensions to three loses, or even distorts, information. Besides, we have often found in practice that the first three dimensions produced by dimensionality-reduction algorithms capture some feature of the data that is certainly present, but not appropriate for our needs.

It would be much nicer if the algorithm only had to reduce the dimensionality of the data to, say, 10, and a good way of visualizing 10-dimensional data was available. One way of doing this is to view, as XGVis [3] does, three of the ten dimensions at a time. A more general way, that Partiview supports, is specifying a 10 x 3 matrix that defines each spatial dimension as a weighted sum of the 10 dimensions. Weights can make use of human intuition in a semi-supervised form. For example, a human expert could mark a few pairs of points as being similar or dissimilar, and the weights could then be adjusted so that these pairs are kept as near or far, respectively, from each other as possible.

4 User Experience

Partiview is an interactive industrial-strength visualization tool. The user can zoom/rotate/translate the data, and turn groups of data points on and off with a press of a button. Data points can also be turned on and off, or have their colors changed, based on the values of certain fields associated with them.

The smoothness of the user experience depends on what computer is being used, in particular on its type of graphics hardware, and if using many images, on the amount of graphics memory. A recent laptop, with a GeForce FX Go5200 graphics card with 64 Mb of memory, can handle over five thousand handwritten digits, each about 4k, and over a million points.

5 Other Features of Partiview

Partiview supports animations, and has several stereo capabilities, including red-blue, crosseyed, and chromadepth. Stereo is especially good for displaying 3-dimensional graphs. Partiview can also display some 3d models, so you can plot, for instance, a map of the internet on a spherical textured Earth.

Partiview runs on Linux, Windows, and Mac OS X. Binaries and source are available from http://niri.ncsa.uiuc.edu/partiview/ .

6 Summary

Partiview allows for easier qualitative evaluation of data clustering algorithms by representing data by images instead of points or simple glyphs. For high-dimensional data, a combination of third-party dimensionality reduction algorithms and Partiview's 'weighted dimension matrix' feature form a powerful data exploration tool.

7 Acknowledgements

Thanks to Misha Belkin and for providing details of his algorithm's results on the MNIST/LeCun dataset to visualize, and Xiaofei He for access to the Corel Image Database. Thanks also to Gina Levow, Partha Niyogi, Mike Papka, Matei Ripeanu, Rick Stevens, and Mark SubbaRao for useful discussions and suggestions.

The digits demo described here can be downloaded from http://www.cs.uchicago.edu/~dinoj/vis/digits . Several other demos can be downloaded from the same site.

References

[1] Stuart Levy. Interactive 3-D Visualization of Particle Systems with Partiview. In *"Astrophysical Supercomputing Using Particles (International Astronomical Union Symposium Proceedings Vol 208)*, 2001.

[2] Mikhail Belkin and Partha Niyogi. Laplacian eigenmaps for dimensionality reduction and data representation. *Neural Computation*, 15(6):1373–1396, 2003.

[3] A. Buja, D. F. Swayne, M. Littman, N. Dean, and H. Hofmann. Xgvis: Interactive data visualization with multidimensional scaling. *Journal of Computational and Graphical Statistics*, 2001.

[4] Xiaofei He and Partha Niyogi. Locality preserving projections. In *Proceedings of Neural Information Processing Systems 16)*, 2003.