Visual Analytics:
Project Report
A Visual Exploration of Manifold Learning for Images

Mattia Bruno Stellacci
stellacci.1992018@studenti.uniroma1.it

Abstract

The following work is a report on the development and implementation of a visual analytics system intended to aid the understanding of dimensionality reduction techniques. In creating an interactive interface to various embeddings of the well-known MNIST handwritten digit dataset we propose a system which allows users to manipulate the embedded respresentations using a web-based application. The system further allows it's users to select the embedded images and to aggregate them using standard image processing algorithms to emphasize the technique's inner workings and visually aid the development of intutions.

1 Motivation, Intended users and preliminary Research

1.1 Motivation

The creation of the system was motivated by the desire to create an aid to understanding dimensionality reduction and the manifold hypothesis, as these topics seems to be strangely ubiquotous in current development in the field of Computer Science.

Reducing the extrinsic dimensionality of data has many advantages, not only for processing efficiency, but also for visualization and algebraic manipulation of higher-dimensional manifolds. Images are a prime example of the discrepancy that can exists between the intrinsic and extrinsic complexity of data. Handwritten digits specifically lend themselves to a use as toy data. Especially, because as humans we can easily comprehend them, despite hundreds of degrees of freedom. The intention was to create a system, that would not only visualize the embeddings generated by various dimensionality reduction techniques as a common cartesian chart, but would further allow the user to select points in the embedded space, manipulate them and aggregate the instances they represent using image processing techniques such as morphology. This last aspect was based on the expectation, that visualizing aggregations of the embedded instances, while knowing their position in the embedded space could yield insights on the inner workings of the embedding techniques employed, exposing chararcteristic such as linearity in the embedded space or the features represented by the degrees of freedom in the embedding.

The Manifold Hypothesis

In referring to the *Manifold Hypothesis* we intend the belief, that data recorded from the natural universe, despite a high dimensionality(think of the approx. 48000 *DOF* in a single sample of the 48KHz audio recording or 3 million *DOF* in a 1MP color image) lies on a lower-dimensional manifold. This belief has been extensively studied by papers such as (Fefferman et al., 2016)(which cites a plethora of other papers on the field) and though fascinating and interesting in it's own right, it is the basis of many techniques used in Machine Learning and Data science as these fields tend to deal with highly dimensional data. Notable examples are the random sampling over a latent space used in generative models (Razavi et al., 2019) or generating plausible interpolations/deformations of instances, exploiting the euclidean nature of the latent space (Cosmo et al., 2020). While these are very interesting examples of advanced applications of the manifold hypothesis, the application proposed here intends to lay the groundwork to it's understanding/intuition.

1.2 Goals

In an effort to create an accessible educational tool, one of the main goals was to build a powerful application that could run within the browser. As there would be image processing required for the various image aggregation techniques, the idea was to incorporate an OpenCV build for *WebAssembly* to move any image progessing to the frontend application. More on this in Section **??**.

1.3 Target Audience

Being a educational tool at it's core the application is geared towards anyone interested in deeping their understanding of dimensionality reduction or even comparing existing techniques with a specific use-case in mind. This includes Machine Learning Engineers/Data Scientists and students of the STEM fields.

Though the prototype of this application developed in the course of this project uses the well-known MNIST handwritten digit dataset as toy data to demonstrate the working principle with, the approach and large parts of the codebase could easily be adapted for custom image datasets and/or taylor made dimensionality reduction techniques in a productice machine learning environment

1.4 Related Work and Preliminary Research

While searching for related work in the field of Visual Analytics, it immediately became apparent that dimensionality reduction is a very common tool for data visualization which given it's expressive potential is unsurprising. Having a closer look at publications mentioning dimensionality reduction, a certain timeline emerges with respect to the role that it has played in scientific publications:

- 2000-2010: While Principal component analysis and Multi-dimensional scaling are methods that have been around since the last century, many publications in this period motivate/discover the techniques themselves and their potential for various applications. Many of these papers explore the mathematical properties of the lower dimensional embeddings. Examples include: The original paper presenting the Isomap embedding (Tenenbaum et al., 2000),or the Locally Linear embedding(lle)(Roweis and Saul, 2000). Contrary to this earlier focus on the techniques involved, towards the end on this decade, dimensionality reduction seems to have become pretty much standard practice for data visualization. Papers such as this survey (Zhang et al., 2010)suggest they are frequently used to visualize data in scientific publications.

- 2010-Present: With Big Data and Machine Learning becoming ever more relevant topics, dimensioality reduction gained more traction not only in it's capability to make data more comprehensible/intuitive to humans, but also as a means of counteracting the proverbial *curse of dimensionality*. Even more recently the focus seems to have once again shifted, as new models allow the enforcing of certain properties in the latent space, such as the preservation of semantics in *word2vec*(Mikolov et al., 2013) word embeddings. These machine learning based techniques, allow the transformation to be conditioned on desired properties, thereby extending the realm of possible applications.

While I couldn't find a work in the field of Visual Analytics that combines dimensionality reduction and image processing in the interactive way I envision, it is apparent that the underlying ideas are not new and that the techniques are still very relevant.

2 Implementation

2.1 Tech stack

In the following we will briefly address the technologies chosen to implement the prototype of the system proposed.

- Dataset We chose the well-known MNIST Handwritten digits dataset as toy data, as the images are small and the classification problem of handwritten digits is intuitive.

- Frontend As we decided to build the application to run in the browser, a powerful framework was required to enable the advanced features we envision. It was therefor decided that the advanced functionality of `ReactJS` would offset the additional effort involved in the inital setup.

- Charting Library: As a solution to creating charts `react-vis` was chosen. First published by Uber in 2016 and sadly depreacated in 2021, `react-vis` is a simple charting library, built for extensibility and interactity.

- Backend: In an effort to prototype rapidly, there seemed to be no incentive to handle the dimensionality reduction in the browser, as far more powerful and versatile solutions exist for Python. We therefor decided to build a simple HTTP-server based on the `bottle` package to serve the data and files. The dimensionality reduction techniques included were sourced from the `scikit-learn` library.

- Image processing To perform the image aggregation on the original instances we chose to use `OpenCV`. Initially it was our intention to perform all the processing in the browser, using a `WebAssembly` build of `OpenCV`. For practical reasons, it later became apparent that it would be much more efficient to simply refer to images by a common index(between frontend and backend) and let the frontend trigger the aggregation of instances in the backend as required.

2.2 Dimensionality Reduction Techniques

While we set out with the initial intention of incorporating approx. 5 different dimensionality reduction techniques, we limited ourseslve to 3 in the prototype. We included

- Principal Component Analysis
- Locally linear embedding
- Isomap embedding

While it is desirable to incorporate and showcase as many different techniques as possible, it became apparent in the development process, that having too many different techniques wouldn't benefit the app as much as focussing on specific techniques and tailoring the application towards showcasing their properties as well as possible.

A property that wasn't sufficiently considered during planning, was whether the incorporated techniques have an inverse transformation or not. More on this in section 4

3 The Prototype

3.1 Usage

When navigating to the page of the prototype the user immediately sees the user interface which can roughly be separated into 5 parts. In the following I will briefly address their function:

Search field

The page's header contains a field allowing the user to select the classes of digits they desire the backend to sample from, before calculating and returnin a 2-dimensional embedding. Leaving the field empty will result in all 10 digit classes being included. A drop-down field allows the user to select the dimensionality reduction technique they wish to employ.

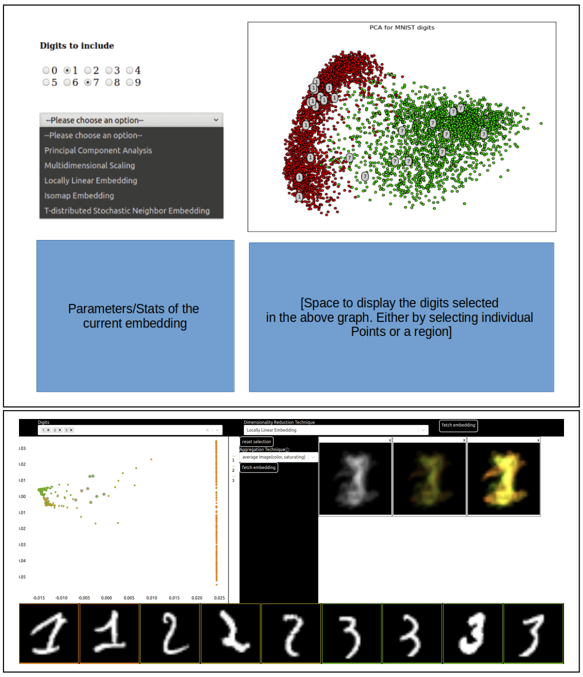

Figure 1: A picture of the proposed mockup vs. screenshot from the prototype of the application

The Chart

This section contains main chart, displaying colour-coded dots representative of all instances embedded in the calculation. The dots in the chart can be selected, either by clicking on them individually, *dragging* a rectangular selection over a region of the graph(similar to selecting multiple files in a OS's file explorer) or by clicking the legend. Hovering a point displays the instance it represents, right-clicking it will remove it from the selection if is currently selected. The scale of the axes is automatically set based on the output of the calculation and the classes are colour-coded. To the right there is a (small) legend containing all digits displayed and their color. Hovering an entry in the legend hightlights only all instances of the class it represents. Clicking on an item adds all instances in that class to the current selection.

The Aggregation Selection

To the right of the chart visualization there is a column that allows the user how they wish to aggregate the selected instances. Aggregation techniques in the prototype are currently limited to various types of average images(calculated over the user's current selction of instances). They are:

- **Average Image(Monochrome)** All images are added and the mean is calculated for every single pixel value. This can emphasize certain similarities in the input but, has a tendency to blur the results

- **Average Image(Colour, Saturating)** Mean images are calculated for per class(as above). These are then converted to colour(the colour is coordinated with the colour their class is represented by in the chart) and the resulting average images are additively overlayed. As this method tends to 'blow out'/saturate high activity areas in the image, it becomes different to interpret for large might induce false colours.

- **Average Image(Colour, normalized)** Similar to above, mean images are calculated for every class. In the aggregation step any class's mean image is converted to HSL, it's luminance is scaled to be proportionate to the current class's relative representation in the user's selection, i.e. the ratio between images pertinent to that class and the total count of images in the selection. This approach avoids saturarating the aggregated image, but results in visually 'dim' results(particularily for the darker coloured classes)

4 Insights

5 Demo

6 Conclusion

References

Luca Cosmo, Antonio Norelli, Oshri Halimi, Ron Kimmel, and Emanuele Rodolà. 2020. LIMP: Learning Latent Shape Representations with Metric Preservation Priors. volume 12348, pages 19–35. ArXiv:2003.12283 [cs, stat].

Charles Fefferman, Sanjoy Mitter, and Hariharan Narayanan. 2016. Testing the manifold hypothesis. *Journal of the American Mathematical Society*, 29(4):983–1049.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient Estimation of Word Representations in Vector Space. ArXiv:1301.3781 [cs].

Ali Razavi, Aaron van den Oord, and Oriol Vinyals. 2019. Generating Diverse High-Fidelity Images with VQ-VAE-2. ArXiv:1906.00446 [cs, stat].

Sam T. Roweis and Lawrence K. Saul. 2000. Nonlinear Dimensionality Reduction by Locally Linear Embedding. *Science*, 290(5500):2323–2326. Publisher: American Association for the Advancement of Science.

J. B. Tenenbaum, V. de Silva, and J. C. Langford. 2000. A global geometric framework for nonlinear dimensionality reduction. *Science (New York, N.Y.)*, 290(5500):2319–2323.

Junping Zhang, Hua Huang, and Jue Wang. 2010. Manifold Learning for Visualizing and Analyzing High-dimensional Data. *IEEE Intelligent Systems*, page 5401149.