ESANN 2016 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 27-29 April 2016, i6doc.com publ., ISBN 978-287587027-8.
Available from http://www.i6doc.com/en/.

A State-Space Model on Interactive Dimensionality Reduction

Ignacio Díaz[1], Abel A. Cuadrado[1] and Michel Verleysen[2] *

1- Electrical Engineering Dept. University of Oviedo
Edif. Dept. 2, campus de Viesques s/n 33204, Gijón, SPAIN
2- Univ. Catholique de Louvain - Machine Learning Group
ICTEAM/ELEN - Place du Levant, 3 1348 Louvain-la-Neuve, Belgium

Abstract. In this work, we present a conceptual approach to the convergence dynamics of interactive dimensionality reduction (iDR) algorithms from the perspective of a well stablished theoretical model, namely *state-space* theory. The expected benefits are twofold: 1) suggesting new ways to import well known ideas from the state-space theory that help in the characterization and development of iDR algorithms and 2) providing a conceptual model for user interaction in iDR algorithms, that can be easily adopted for future interactive machine learning (iML) tools.

1 Introduction

The communities of data visualization and machine learning have been becoming aware of the amazing opportunities of bringing together intelligent algorithms and human perception. An emerging field in the epicenter of this common place is the development of steerable machine learning algorithms [1]. Dimensionality reduction (DR) techniques provide a way to find latent low dimensional structures in high dimensional data, resulting in a mapping from a high dimensional space on a low dimensional space that makes it possible to visualize items arranged in an ordered way, following a spatialization principle (close $\approx$ similar) allowing the user to interpret and interact with the data. Interaction provides *feedback* in the visualization process, resulting in a virtuous cycle where the user is part of the loop and drives the process to increase knowledge and focus on the interesting patterns or aspects of the problem –see [2] for an enlightening model of the visualization process. Despite traditional interaction mechanisms (zoom, pan, focus & context, etc.) help the user to carry out this process, the transformations induced on the view are quite basic and far from "intelligent". In the last few years, some works have proposed more advanced interaction schemes, involving direct manipulation of the intelligent data analysis algorithms used for visualization [3, 4, 5, 6, 7]. More recently, closely related to the paradigm proposed in [1], an interactive version of DR (iDR) based on visualizing and interacting with intermediate results during convergence was proposed for the analysis of time varying data or correlation analysis, [8, 9].

In this work, we present a conceptual approach to iDR convergence dynamics from the perspective of a well stablished theoretical model, namely *state-space*

*The authors would like to thank financial support from the Spanish Ministry of Economy (MINECO) and FEDER funds from the EU under grant DPI2015-69891-C2-2-R.

theory. The expected benefits are twofold: 1) suggesting new ways to import well known ideas from the state-space theory and 2) providing a conceptual model for user interaction in iDR, that can be easily adopted for future iML tools.

2 Conceptual formulation of interactive DR

Configuration space. Let $P = \{\mathbf{p}_i\}_{i=1,...,Q}$ be a set of points $\mathbf{p}_i = (p_i^1, ..., p_i^n) \in \mathbb{R}^n$. We define the *configuration vector* of set P as the vector composed of the nQ scalar coordinates of all the points $\mathbf{p}_i$, that will be denoted as

$$\mathbf{p} = (p_1^1, p_1^2, ..., p_1^n, p_2^1, p_2^2, ..., p_2^n, ..., p_Q^1, p_Q^2 ..., p_Q^n) \in \mathbb{R}^{nQ}$$

The configuration vector $\mathbf{p}$ is a single point of the *configuration space* $\mathbb{R}^{nQ}$ that uniquely defines the spatial layout of the points $\mathbf{p}_i$.

Formulation in state-space theory. Let's consider a DR algorithm that takes a set of points $X = \{\mathbf{x}_i\}_{i=1,...,Q}$, being $\mathbf{x}_i \in \mathbb{R}^D$, in the input data space, and yields a set of projections $Y = \{\mathbf{y}_i\}_{i=1,...,Q}$, being $\mathbf{y}_i \in \mathbb{R}^d$, in a low-dimensional visualization space $V \in \mathbb{R}^d$. Considered as systems that evolve in time during the convergence stage, DR algorithms can be viewed as *dynamical systems*. A convenient description of DR dynamic behavior is the following state-space equation

$$\dot{\mathbf{y}} = \mathbf{f}(\mathbf{y}, \mathbf{u}) \tag{1}$$

whose *state* is the *projection configuration vector*, defined as

$$\mathbf{y} = (y_1^1, y_1^2, ..., y_1^d, y_2^1, y_2^2, ..., y_2^d, ..., y_Q^1, y_Q^2 ..., y_Q^d) \in \mathbb{R}^{dQ}$$

that describes the current projections $\mathbf{y}_i$, and its input or exogenous variable is the *context configuration vector*

$$\mathbf{u} = (\mathbf{x}, \mathbf{w}) = (x_1^1, x_1^2, ..., x_1^D, x_2^1, x_2^2, ..., x_2^D, ..., x_Q^1, x_Q^2 ..., x_Q^D, w^1, w^2, ..., w^m)$$

which is composed of the input data points $\mathbf{x}_i$, plus the set of adjustable parameters $w^1, w^2, ..., w^m$ specific to the DR algorithm. The DR algorithm has a cost function $J(\mathbf{y}, \mathbf{u})$ that depends on the projections $\mathbf{y}$, the input data $\mathbf{x}$ and the DR parameters $\mathbf{w}$. During the DR convergence J is optimized for $\mathbf{y}$, within a given context $\mathbf{u}$.

To express this optimization from a computational framework, assuming small time increments Δt, equation (1) can be approximated into the discrete form $\Delta\mathbf{y}/\Delta t = \mathbf{f}(\mathbf{y}, \mathbf{u})$, that is, $\Delta\mathbf{y} = \Delta t \cdot \mathbf{f}(\mathbf{y}, \mathbf{u})$, which turns into

$$\mathbf{y}(t+1) = \mathbf{y}(t) + \Delta t \cdot \mathbf{f}(\mathbf{y}(t), \mathbf{u}(t)) \tag{2}$$

In the last expression it is made evident that, at every step, the DR algorithm takes an initial projection $\mathbf{y}(t)$ and evolves to a new (updated) projection $\mathbf{y}(t+1)$. It can also be noted that equation (2) conceptually resembles a gradient descent approach, that is at the heart of many non-convex DR algorithms (see e.g. [10]).

In this case, the term $f(\mathbf{y}, \mathbf{u})$ can be interpreted as the gradient of the cost function

$$\mathbf{f}(\mathbf{y}, \mathbf{u}) \approx \frac{\partial J(\mathbf{y}, \mathbf{u})}{\partial \mathbf{y}}$$

Under the hypothesis of a stable algorithm, the state equation (1) reaches a steady state for $\dot{\mathbf{y}} = \mathbf{0}$ resulting in the following condition expressed in implicit form

$$\mathbf{0} = \mathbf{f}(\mathbf{y}^0, \mathbf{u}^0) \tag{3}$$

The previous expression states that the context $\mathbf{u}^0$ forces a final projection $\mathbf{y}^0$ that results from a new equilibrium state, although such projection might not be unique. An important consequence of equation (3) is that a change in the *context* $\mathbf{u}^0$ may induce a change in the steady-state projection $\mathbf{y}^0$. This fact is indeed the key to bring interactivity into the DR visualization.

A further consequence of this approach is the possibility to analyze the DR from a dynamic point of view. Small variations around the equilibrium point 0 allow us to consider the linear model $\dot{\mathbf{y}} = \mathbf{A}\mathbf{y} + \mathbf{B}\mathbf{u}$, which paves the way for rigorous local analysis of stability and dynamical behavior of iDR convergence, based on eigenmode analysis of the state matrix $\mathbf{A} = \partial\, \mathbf{f}(\mathbf{y}, \mathbf{u})/\partial \mathbf{y}|_0$.

3 Context-based steering of DR projections

As seen in the previous section, the context $\mathbf{u}$ includes: a) the input dataset X, and b) cost-function specific parameters $\mathbf{w} = (w^1, w^2, ..., w^m)$ that the user can modify. Both components can vary during the algorithm execution, resulting in different interactive DR operation modes.

Tracking time-varying input datasets. On one side, the input dataset can be composed of *time varying* data $X(t) = \{\mathbf{x}_1(t), \ldots, \mathbf{x}_Q(t)\}$. In this case, the resulting dynamical model contains a time varying input $\mathbf{u} = \mathbf{u}(t)$, thereby adding a forced dynamics component that drives the result. In case that $\mathbf{x}$ reaches a steady state $\mathbf{x}^0$ for $t \to \infty$, the DR algorithm converges to $\mathbf{0} = \mathbf{f}(\mathbf{y}^0, (\mathbf{x}^0, \mathbf{w}))$, resulting in a final projection dependent on the steady-state input dataset $\mathbf{x}^0$ and the selected DR optimization parameters $\mathbf{w}$. Time-varying input data arise in analyses where a group of multivariate items evolve in time. Many real situations might conform to this formulation, such as the analysis of groups of patients during epidemics, maybe with control subgroups under experimental treatments, analysis of evolving social networks composed of many users, each defined by several parameters, or analysis of electric power networks dynamics under failures or special load conditions. The result of this is a dynamic DR projection whose items are continuously rearranged according to their evolving similarities. Thus, if an item $\mathbf{x}_i$ undergoes at time t a significant change in its relationships to the other items with respect to time $t-1$, its projection $\mathbf{y}_i$ will move apart revealing a change condition.

Steering DR optimization parameters. There exists a wide variety of algorithms in DR literature [11]. Most parameters in DR algorithms are related to the way the cost function is evaluated. Depending on the algorithm, one group of these parameters typically include neighborhood function parameters, such as width factors σ of gaussian components, number of neighbors k or perplexity P –see e.g. [10]. Another group of cost-function related parameters are those affecting the computation of the distance metrics between points. As described in [8], a simple but powerful interaction feature can stem from user-driven change in the input space metric $\mathbf{\Omega}$. Let's consider the following weighted norm in the input data space

$$\|\mathbf{x}_i - \mathbf{x}_j\|^2_{\mathbf{\Omega}} := \sum_r \sum_s (x_i^r - x_j^r)\omega_{rs}(x_i^s - x_j^s).$$

Using the metric induced by the previously defined weighted norm, the distances between input points $\mathbf{x}_i$ and $\mathbf{x}_j$ would be $d_{ij} = \|\mathbf{x}_i - \mathbf{x}_j\|_{\mathbf{\Omega}}$. In this paper, we shall consider the special case where the weight matrix $\mathbf{\Omega}$ is diagonal $\mathbf{\Omega} = \text{diag}(w_1, w_2, \ldots, w_n)$, where $w_q = \omega_{qq}$. Interactive user-driven changes in these weights w_i can provide insight into different kinds of analysis, including –see [8]– *correlation analysis*, by interactively weighting subsets of variables $q_1, q_2, ...q_K$ –setting the remaining weights to zero–, whereby the emergence of any ordered patterns in the resulting projection reveals dependencies among $x_{q_1}, x_{q_2}, \ldots, x_{q_K}$. Similarly, the user can do *sensitivity analysis* by changing one or M weights $\{w_{q_1}, w_{q_2}, \ldots, w_{q_M}\}$ at the same time; the points that move in the "live" projection reveal elements that differ significantly in any of the variables $x_{q_1}, x_{q_2}, \ldots, x_{q_M}$.

Introducing class knowledge. Adding extra attributes with known class information in the input data matrix gives the user the posibility to group items according to their class memberships in the projection, thereby allowing for a supervised DR. In its most basic approach, class attributes may consist of one or more columns with different discrete values for each class (e.g. using a different integer or even a 2D position for each class). Feature space transformations [12] allow improving the quality of an existing embedding in terms of both structural preservation and class separation. One simple feature extension scheme, for instance, is to augment each element $\mathbf{x}$ with an extended feature set $\bar{\mathbf{x}}_{c(\mathbf{x})}$ equal to the centroid of the class $c(\mathbf{x})$ it belongs to, thus forming an extended vector $\mathbf{x}_e = [\mathbf{x}, \bar{\mathbf{x}}_{c(\mathbf{x})}]$. The DR projection of $\mathbf{x}_e$, therefore contains class information, resulting in a more meaningful projection. A user-driven variant of this approach suitable for interaction proposed in [13], could involve a weight factor λ

$$\mathbf{x}_e(\lambda) = [(1-\lambda)\mathbf{x}, \lambda\bar{\mathbf{x}}_{c(\mathbf{x})}].$$

This approach can be seen as a particular case of weighted metrics on the extended attribute vectors $\mathbf{x}_e = [\mathbf{x}, \bar{\mathbf{x}}_{c(\mathbf{x})}]$ using $w_i = \lambda$ for the original attributes $\mathbf{x}$ and $w_j = 1 - \lambda$ for the class attributes $\bar{\mathbf{x}}_{c(\mathbf{x})}$. Letting the user modify λ at iteration level, the user can interactively control the balance between class separation and structural preservation of the original dataset to gain insight and find connections between data structure and class knowledge. Note that if only

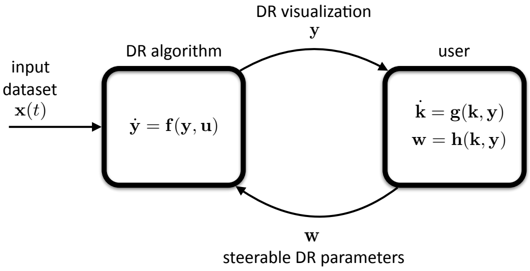
DR visualization
$\mathbf{y}$
DR algorithm
user
input
dataset
$\mathbf{x}(t)$
$\dot{\mathbf{y}} = \mathbf{f}(\mathbf{y}, \mathbf{u})$
$\dot{\mathbf{k}} = \mathbf{g}(\mathbf{k}, \mathbf{y})$
$\mathbf{w} = \mathbf{h}(\mathbf{k}, \mathbf{y})$
$\mathbf{w}$
steerable DR parameters

Fig. 1: Model describing the coupled interaction between the user and the DR algorithm

classification is looked for, the best projection happens for $\lambda = 1$, projecting all members of each class on one single point (their representative codebook) respectively.

4 A model of interaction between DR and the user

During an interactive DR session, the user's knowledge and the DR visualization evolve in a coupled manner, hopefully resulting in an increase of the user's knowledge. Inspired by the model described in [2], the knowledge of the user evolves depending on the visualization and the current knowledge state as $\dot{\mathbf{k}} = g(\mathbf{k}, \mathbf{y})$. In parallel, based on the current state of the user's knowledge $\mathbf{k}$ and the current projection state $\mathbf{y}$, the user steers the DR configuration parameters as $\mathbf{w} = \mathbf{h}(\mathbf{k}, \mathbf{y})$ to recompute the projection, according to new criteria that better meet the user's interest –for instance, depending on the problem needs, the user may wish to tune the DR algorithm to favour intrusions or extrusions [14]. In control theory, the latter equation can be seen as a *control law*, since it defines an input value $\mathbf{u}$ (depending on $\mathbf{w}$) that manipulates the dynamical system, according to some target (here, maximizing user's knowledge), based on the information of the current system state. This results in a coupled system –see Fig. 1– containing the current user's knowledge and the current projection, that models the interaction between the user and the DR algorithm, whose final result $(\mathbf{k}^0, \mathbf{y}^0)$ mainly depends on the input data $\mathbf{x}$ to be analyzed, assuming that the user is not influenced by other factors. The quality of the final knowledge $\mathbf{k}^0$ depends on a good design exploiting the synergy between the algorithm $\mathbf{f}()$ the user's mind capabilities $\mathbf{g}()$ and the interaction $\mathbf{h}()$.

5 Conclusion

In this paper we have presented a state-space approach to provide a conceptual model of the iDR dynamics. We show that the dynamics of iDR can be considered as a particular case of the more general state-space formulation of dynamic systems, provided that we consider the "state" of the iDR system as a

ESANN 2016 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 27-29 April 2016, i6doc.com publ., ISBN 978-287587027-8. Available from http://www.i6doc.com/en/.

point in the configuration space. We have described the association between the iDR elements and their state-space counterparts, resulting in a consistent model that lays the basis to develop rigorous descriptions of stability and dynamical behavior of iDR convergence, based on eigenmode analysis of locally linearized models. In light of this, borrowing the idea of *state feedback* may suggest new fields of research on DR algorithm designs with tailored dynamics and robust stability properties. Finally, we believe that the ideas presented here can serve also in a conceptual plane as a model to describe in a formal way interaction schemes and the role of the user during iDR operation.

References

[1] Jaegul Choo and Haesun Park. Customizing computational methods for visual analytics with big data. *Computer Graphics and Applications, IEEE*, 33(4):22–28, 2013.

[2] J.J. Van Wijk. The value of visualization. In *16th IEEE Visualization 2005 (VIS 2005)*. IEEE Computer Society, 2005.

[3] Dong Hyun Jeong, Caroline Ziemkiewicz, Brian Fisher, William Ribarsky, and Remco Chang. iPCA: An interactive system for PCA-based visual analytics. *Computer Graphics Forum*, 28(3):767–774, 2009.

[4] Sara Johansson and Jimmy Johansson. Interactive dimensionality reduction through user-defined combinations of quality metrics. *Visualization and Computer Graphics, IEEE Transactions on*, 15(6):993–1000, 2009.

[5] Stephen Ingram, Tamara Munzner, Veronika Irvine, Melanie Tory, Steven Bergner, and T. Möller. Dimstiller: Workflows for dimensional analysis and reduction. In *Visual Analytics Science and Technology (VAST), 2010 IEEE Symposium on*, pages 3–10. IEEE, 2010.

[6] Eli T. Brown, Jingjing Liu, Carla E. Brodley, and Remco Chang. Dis-function: Learning distance functions interactively. In *Visual Analytics Science and Technology (VAST), 2012 IEEE Conference on*, pages 83–92. IEEE, 2012.

[7] A. Endert, L. Bradel, and C. North. Beyond control panels: Direct manipulation for visual analytics. *Computer Graphics and Applications, IEEE*, 33(4):6–13, July 2013.

[8] Ignacio Díaz, Abel A. Cuadrado, Daniel Pérez, Francisco J. García, and Michel Verleysen. Interactive dimensionality reduction for visual analytics. In *European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning*, Bruges, Belgium, April 2014.

[9] Serafín Alonso, Miguel A. Prada, Juan J. Fuertes, Ignacio Díaz, and Manuel Domínguez. Analysis of electricity bill data using interactive dimensionality reduction. In Lazaros Iliadis and Chrisina Jayne, editors, *16th International Conference on Engineering Applications of Neural Networks (EANN 2015)*, Island of Rhodes, Greece, September 2015.

[10] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-SNE. *Journal of Machine Learning Research*, 9:2579–2605, 2008.

[11] J.A. Lee and M. Verleysen. *Nonlinear dimensionality reduction*. Springer Verlag, 2007.

[12] Matthias Schaefer, Leishi Zhang, Tobias Schreck, Andrada Tatu, John A Lee, Michel Verleysen, and Daniel A Keim. Improving projection-based data analysis by feature space transformations. In *IS&T/SPIE Electronic Imaging*, pages 86540H–86540H. International Society for Optics and Photonics, 2013.

[13] Daniel Pérez, Leishi Zhang, Matthias Schaefer, Tobias Schreck, Daniel Keim, and Ignacio Díaz. Interactive feature space extension for multidimensional data projection. *Neurocomputing*, 150, Part B:611 – 626, 2015.

[14] John A. Lee and Michel Verleysen. Quality assessment of dimensionality reduction: Rank-based criteria. *Neurocomputing*, 72(7):1431–1443, 2009.