Clustering on Manifolds: an Application to Scenario Analysis using Principal Component Analysis

D. Mandelli[1†], A. Yilmaz[2] and T. Aldemir[1]
[1] *Nuclear Engineering Program, The Ohio State University, Columbus, OH, 43202*
[2] *Photogrammetric Computer Vision Laboratory, The Ohio State University, Columbus, OH, 43202*
[†] *Corresponding author: mandelli.1@osu.edu*

INTRODUCTION

A recent trend in the nuclear power engineering field is the implementation of heavily computational time consuming algorithms [1] and codes [2, 3] for both design and safety analysis. The new generation of system analysis codes aim to embrace several phenomena such as thermo-hydraulic, structural behavior, system dynamics and human behavior, as well as uncertainty quantification and sensitivity analyses associated with these phenomena.

The use of dynamic probabilistic risk assessment (PRA) methodologies, such as the dynamic event tree (DET) methodology [1, 4], allows a systematic approach to uncertainty quantification. The major challenges in using dynamic methodologies are the heavier computational and memory requirements compared to the classical event-tree/fault-tree analysis to achieve better and more systematic description of hardware/process/software/firmware/human interactions. Each branch of a DET contains time evolutions of a large number of variables and a large number of scenarios arising from a single initiating event. Such large amounts of information are usually very difficult to organize in order to identify the main trends in scenario evolution and the main risk contributors for each initiating event.

Clustering methodologies [5], such as the Mean-Shift algorithm [6, 7], offer powerful tools that can help the user to identify groups of scenarios that are representative of the original data set and, thus, can reduce the effort involved in data analysis. This is particularly useful to identify commonalities among scenarios that have similar temporal behavior but different outcomes or to identify differences among data sets generated for different initial conditions or system configurations.

In order to decrease the computational time of the clustering algorithms, data dimensionality reduction is an effective approach. The rationale is to identify the dependencies in the original data set and perform the clustering on a reduced data set without loosing information on system behavior.

In this article, a data reduction algorithm based on the local implementation of the Principal Component Analysis (PCA) [8] is described and applied to a large data set generated by a DET. The paper also shows how the data reduction decreases the computational time in clustering. We use the Mean-Shift algorithm as a clustering tool but the proposed methodology can be applied to any data clustering algorithm.

CLUSTERING AND DIMENSIONALITY REDUCTION

Clustering applied to scenario analysis is the process of organizing scenarios into groups (i.e., clusters) whose members have similar behavior. In the clustering using the Mean-shift algorithm, each scenario s_i is represented as the multidimensional vector

$$s_i = [s_i(0), s_i(1), s_i(2), \ldots, s_i(T)] \tag{1}$$

where $s_i(t)$ is an M-tuple which contains values of M chosen system variables $(x_1, \ldots, x_M)$ sampled at time $t = 1, \ldots, T$. Note that the dimensionality of each scenario is $M \cdot T$ and can be extremely high for complex systems (i.e., high number of state variables and high sample instants).

These M variables are often heavily correlated and, consequently, the information contained in the set of M state variables comprising the full state space can still be maintained by using a set of N variables where $N < M$. The objective of the data reduction process is to determine those N variables by finding the correlations among the original M state variables to achieve dimensionality reduction[1].

Dimensionality reduction is the process of finding a bijective mapping function $\mathfrak{F}$:

$$\mathfrak{F} : \mathbb{R}^D \mapsto \mathbb{R}^d \text{ (where } d < D) \tag{2}$$

which maps the data points from the D-dimensional space into a reduced d-dimensional space (i.e., embedding on a manifold) in such a way that the distances between each point and its neighbors are preserved. In our applications $D = M + 1$, i.e., M state variables plus time t.

Linear algorithms for dimensionality reduction, such as PCA [8] or multidimensional scaling (MDS) [9], have the advantage that they are easier to implement but can only identify linear correlations among state variables. In order to overcome this limitation, it is possible to partition the original data set into smaller subsets and apply MDS or PCA to each of these subsets (i.e., local analysis). This approach assumes that each of the subsets are characterized by linear correlation among variables and, thus, part of the dimensionality reduction process is to find those subsets such that the correlation among variables can be considered linear.

As reported in [10], the local application of MDS (i.e., the ISOMAP algorithm) showed good dimensionality reduction results. This paper focuses on the local application of PCA using the same dataset used in [10].

[1] Note that those N variables are not necessarily a subset of the original M variables but, more likely, a combination of those M state variables.

LOCAL PCA

The main idea behind PCA [8] is to perform a linear mapping of the data set into a lower dimensional space such that the variance of the data in the low-dimensional representation is maximized.

This is accomplished by determining the eigenvectors and their corresponding eigenvalues of the data covariance matrix[2] S. The eigenvectors that correspond to the largest eigenvalues (i.e., the principal components) can be used as a set of basis functions. Thus, the original space is reduced to the space spanned by few eigenvectors and the original data points are projected into this new reduced space.

Figure 1 shows an example of dimensionality reduction using PCA for a data set distributed in a 2-dimensional space. After performing the eigenvalue-eigenvector decomposition of the covariance matrix, the algorithm chooses the eigenvector having the largest eigenvalue (i.e., λ_1) as subspace to project the original data. The algorithm is very easy to implement but is not able to identify non-linear correlations among variables.

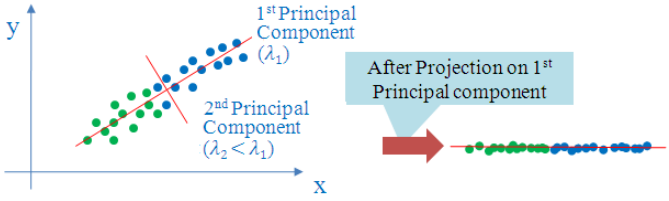

Fig. 1: Example of dimensionality reduction using PCA (reduction from $D = 2$ to $d = 1$).

In order to overcome this limitation we implemented a modified version of the PCA algorithm which performs the dimensionality reduction by analyzing the local properties of the data set as following (see Fig. 2):

1. Divide the mission time into K intervals $[t_k, t_{k+1}]$ with $k = 1, \ldots, K$
2. Consider the data points for all scenarios within a time interval $[t_k, t_{k+1}]$
3. Perform the PCA algorithm for the subset of points identified in Step 2
4. Repeat Steps 2 and 3 for all time intervals identified in Step 1
5. Identify, for each time interval, the N number of eigenvectors that maintains the local geometric properties of the original data points
6. Project, for each time interval, the original data points into the new reduced space.

[2]Given a data set in form of a matrix Z (size $D \times \Lambda$), rows correspond to data dimensions (D) and columns correspond to the number data observations (Λ), the covariance matrix S is determined as: $S = \frac{1}{\Lambda - 1} Z'Z$.

The choice of the time intervals $[t_k, t_{k+1}]$ is performed by recursively analyzing the rate of change of the covariance matrix computed in that interval. The rationale is to chose intervals where the rate of change of the covariance matrix is below a fixed threshold.

The number N of eigenvectors is determined by inspecting the sum of the corresponding N eigenvalues; N is chosen when this sum is above 90% of the overall sum of the M eigenvalues.

CASE EVALUATED

The initiating event investigated was that of a station blackout (SBO) at a U.S. PWR and the MELCOR code [3] was linked to the ADAPT tool [1] to determine the evolution for each DET scenario. The simulations using MELCOR model the transient from the occurrence of the SBO through the core melting phase and up to point of containment failure and release of radionuclides to the environment. For the purposes of this paper, we choose 8 state variables of interest (i.e., $M = 8$):

1. Seal LOCA flow rate [gpm]
2. Hydrogen mass generated [kg]
3. Core water level [m]
4. System Pressure [Pa]
5. Core vapor temperature [K]
6. Hot leg vapor temperature [K]
7. Intact core fraction [%]
8. Fuel Temperature [K]

We sampled each state variable 100 times (hence, $T = 100$) which gave us an accurate description of all the 104 transients. The resulting dimensionality of the data is equal to $100 \cdot 8 = 800$.

RESULTS

Dimensionality reduction using local PCA was performed for the data set described in the previous section by using $K = 30$. The dimensionality reduction process identified 6 variables (i.e., $N = 6$) from the original $M = 8$ state variables. The subsequent reduction in the computational time for the clustering process was19%.

In order to validate the dimensionality reduction algorithm, we compared the clusters obtained from the original and the reduced data sets. Table 1 shows a comparison of the clusters obtained by the Mean-Shift algorithm for both the original and the reduced sets and indicates that the 8 clusters obtained from both data sets agree in terms of both number of scenarios contained and cluster-to-scenario memberships.

Note that the original data set consists of only 8 state variables chosen a priori by the user. A more complete

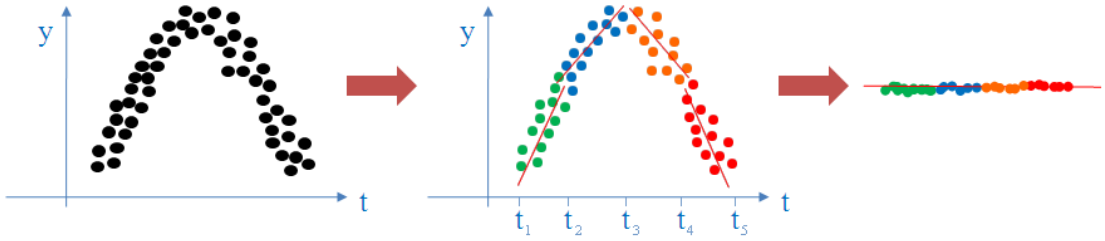

Fig. 2: Example of dimensionality reduction using local PCA (reduction from $D = 2$ to $d = 1$) with $K = 5$.

TABLE 1: Clusters obtained using Mean-Shift for both original and reduced data set. Each entry indicates the number of scenarios contained in each cluster.

Cluster	Original	Reduced	Cluster membership
1	5	5	Identical
2	49	49	Identical
3	41	41	Identical
4	3	3	Identical
5	1	1	Identical
6	1	1	Identical
7	1	1	Identical
8	3	3	Identical

application of this algorithm would be to start with the full set of state variables (e.g., the MELCOR code [3] has 50,000 data channels where each data channel corresponds to a specific state variable of a specific node of the simulator) and perform the dimensionality reduction on this full set. Due to the fact that these data channels are often heavily correlated, it is expected that much larger dimensionality and computational time reduction would be achieved with the full set.

CONCLUSIONS

This paper presents a methodology to reduce the dimensionality of a data set by locally implementing the PCA algorithm in order to reduce the computational time in the clustering process. The algorithm has been applied to a large data set generated by a DET for the SBO analysis of a PWR. The resulting dimensionality reduction led to about 19% computational time reduction in performing the clustering. It is expected that the computational time savings would be more significant when a larger number of state variables is chosen to characterize each scenario.

REFERENCES

1. B. RUTT, U. CATALYUREK, A. HAKOBYAN, K. METZROTH, T. ALDEMIR, R. DENNING, S. DUNAGAN, and D. KUNSMAN, "Distributed dynamic event tree generation for reliability and risk assessment," in "Challenges of Large Applications in Distributed Environments," IEEE (2006), pp. 61–70.
2. RELAP5-3D CODE DEVELOPMENT TEAM, *RELAP5-3D Code Manual*, Idaho National Laboratory, Idaho Falls, ID (USA) (2005).
3. R. O. GAUNTT, *MELCOR Computer Code Manual, Version 1.8.5, Vol. 2, Rev. 2*, Sandia National Laboratories, NUREG/CR-6119.
4. T. ALDEMIR, D. MILLER, M. STOVSKY, J. KIRSCHENBAUM, P. BUCCI, A. FENTIMAN, L. MANGAN, and S. ARNDT, *NUREG/CR 6901: Current state of reliability modeling methodologies for digital systems and their acceptance criteria for nuclear power plant assessments*, Division of Fuel, Engineering and Radiological Research, Office of Nuclear Regulatory Research, U.S. Nuclear Regulatory Commission, Washington, DC (2006).
5. C. M. BISHOP, *Pattern Recognition and Machine Learning (Information Science and Statistics)*, Springer (2007).
6. K. FUKUNAGA and L. HOSTETLER, "The estimation of the gradient of a density function, with applications in pattern recognition," *IEEE Transactions on Information Theory*, **21**, 1, 32–40 (1975).
7. D. MANDELLI, T. ALDEMIR, A. YILMAZ, K. METZROTH, and R. DENNING, "Scenario Aggregation and Analysis via Mean-Shift Methodology in Level 2 PRA," in "Proceedings of the American Nuclear Society (ANS) ICAPP 2010 Topical Meeting," (2010), pp. 990–994.
8. I. T. JOLLIFFE, *Principal Component Analysis*, Springer, second ed. (October 2002).
9. I. BORG and P. GROENEN, *Modern Multidimensional Scaling: Theory and Applications*, Springer-Verlag New York (2005).
10. D. MANDELLI, A. YILMAZ, and T. ALDEMIR, "Clustering Scenarios on Manifolds," in "Proceeding of American Nuclear Society (ANS)," (2011).