Dimensionality Reduction for Tracking

Ali Rahimi
Intel Research Seattle
ali.rahimi@intel.com

Ben Recht
Caltech
brecht@ist.caltech.edu

Brian Ferris
University of Washington
bdferris@cs.washington.edu

Dieter Fox
University of Washington
fox@cs.washington.edu

We have been applying dimensionality reduction techniques to a variety of tracking problems. We have experimented with tracking the articulated pose of humans from video imagery, the trajectory of RFID tags from signal strength measurements, the trajectory of acoustic beacons in sensor networks, and the location of wireless device from 802.11 signal measurements. In each case, an analytic relationship between pose and measurements is either unknown or difficult to determine, so we have relied on generic dimensionality reduction or system identification algorithms. We would like to share some of the valuable lessons we have learned from these experiments, and explain how these applications have guided our development and evaluation of various dimensionality reduction algorithms. We convey the importance of taking advantage of tracking-specific prior information, of avoiding local minima, and of qualitatively evaluating the output of these algorithms, and explain why "manifold learning" is a red herring in the context of tracking.

Use time. In tracking, dimensionality reduction techniques that leverage the temporal ordering of the input signal far outperform those that do not. We have experimented with augmentations of KPCA [8], Isomap [4], and GPLVM [11] and find that they significantly outperform their time-agnostic counterparts [2, 8, 9]. Using time as an additional cue also reduces these algorithms' reliance on the brittle local neighborhood structures they require, providing additional robustness.

Question the assumptions of "Manifold Learning". A plausible generative model for our tracking problems is that latent low-dimensional poses evolve according to Markovian dynamics, and are noisily observed through a smooth unknown nonlinearity. Nonlinear system identification procedures seek to estimate the parameters of such generative models [5]. The modeling assumptions behind most manifold learning algorithms, such LLE (preserving local affine coordinates of points) or of Isomap (isometrically unfolding the manifold) do not plausibly model the measurement process in our tracking problems. While Isomap can sometimes recover the gross geometry of some of our trajectories, methods based on generative models consistently outperform it when they do not get stuck in local minima.

Avoid local minima. Although unsupervised nonlinear system identification procedures based on latent variable generative models [3, 5–7, 10, 11] capture the prior information in tracking problems well, in practice, we find that they are prone to getting stuck in local optima. These local optima are usually far from the ground truth and there are no guarantees that they are close to the global optimum. Even worse, we have no diagnostic to quantify the size of this gap. In contrast, the manifold learning literature has produced many cost functions that have no local minima. One approach is to use manifold learning algorithms to generate initial iterates for generative-model-based algorithms [2]. Another approach is to approximately optimize the posterior defined by the generative model (see below).

Find a quantitative evaluation metric. In tracking, ground truth is well-defined, and is usually available. To identify the best algorithm for a tracking task, it is helpful to quantify the deviation between the ground truth and the recovered low-dimensional coordinates. Such evaluation is rarely performed in the unsupervised system ID or manifold learning literature, perhaps because these methods are often intended as either visualization aids, or as output predictors. Because they make few *a priori* assumptions about the problem, most unsupervised algorithms can only recover the ground truth up to a coordinate transformation. We avoid unduly penalizing such gauge ambiguities by registering the output to the ground truth trajectory with an affine model, and reporting the residual error. This error measure ensures that the output is not penalized for variations in scale, translation, and rotation with respect to the ground truth, and that incorrect choice of subspace, folding, changes in topology, or nonlinear stretching are penalized.

Center the output. The low-dimensional coordinates reported by LLE tend to be squeezed into a thin sliver. This problem can often be alleviated by modifying the embedding cost function of LLE to explicitly force the mean of the output to zero. A similar re-centering trick has been suggested for kernel methods, and we recommend applying it to other spectral manifold learning algorithms as well.

Out of sample extensions and online updates. Because our goal is to map as-yet-unseen observations to poses, we need to learn out-of-sample extensions [1]. In some situations, the observation function (or the manifold of measurements) changes over time. For example, unrelated changes in the environment affect the relationship between radio frequency signal strength measurements and the pose of an object. In such case, the algorithm must incorporate unlabeled observations as they become available and update its out-of-sample mapping online.

Take advantage of low sensor noise and lack of self-intersection. In most of our experiments, sensor noise was small (except when tracking with 802.11 signal strengths), and each location within the range of the sensors produced a unique measurement. Both of these assumptions are implicit in almost all the manifold learning algorithms we have encountered (LLE, Isomap, Laplacian Eigenmaps, Hessian LLE), and almost none of the system ID algorithms we have encountered [3, 6, 10, 11]. Introducing these assumptions to nonlinear system ID can greatly simplify them, and alleviate problems with local minima, as we argue below.

At this year's NIPS, we will present an algorithm that incorporates many of these lessons. It approximately finds the MAP estimate of the observation function in a simplified generative model without getting stuck in local minima [8]. The algorithm assumes that the prior joint distribution $p_{\mathbf{X}}(\mathbf{X})$ over poses follows known dynamics, and that observations $\mathbf{Y}$ are obtained by passing poses through an unknown nonlinear transformation f that is invertible on the manifold of observations (thereby assuming that the manifold does not self-intersect, and that measurement noise is negligible). The standard change of variables formula immediately gives a likelihood over f^{-1}. When $p_{\mathbf{X}}$ is Markov or completely factored, the Shannon-McMillan-Breiman theorem guarantees consistency of the ML estimate. This likelihood can be optimized approximately using an eigenvalue relaxation when $p_{\mathbf{X}}(\mathbf{X})$ is Gaussian. An SDP relaxation that provides an approximation guarantee is also possible. We have explored methods for learning the parameters of the prior $p_{\mathbf{X}}$, are experimenting with a version of the algorithm that alleviates the self-intersection restriction, and are working on stronger bounds on our approximation result.

References

[1] Y. Bengio, O. Delalleau, N. Le Roux, J.-F. Paiement, P. Vincent, and M. Ouimet. Learning eigenfunctions links spectral embedding and kernel pca. *Neural Computation*, 10(16):2197–2219, 2004.

[2] B. Ferris, D. Fox, and N. Lawrence. Wifi-slam using gaussian process latent variable models. In *International Joint Conferences on Artificial Intelligence (IJCAI)*, 2007.

[3] Z. Ghahramani and S. Roweis. Learning nonlinear dynamical systems using an EM algorithm. In *Neural Information Processing Systems (NIPS)*, pages 431–437, 1998.

[4] O. Jenkins and M. Mataric. A spatio-temporal extension to isomap nonlinear dimension reduction. In *International Conference on Machine Learning (ICML)*, 2004.

[5] A. Juditsky, H. Hjalmarsson, A. Benveniste, B. Delyon, L. Ljung, J. Sjöberg, and Q. Zhang. Nonlinear black-box models in system identification: Mathematical foundations. *Automatica*, 31(12):1725–1750, 1995.

[6] N. D. Lawrence. Gaussian process latent variable models for visualisation of high dimensional data. In *Advances in Neural Information Processing Systems (NIPS)*, 2004.

[7] L. Ljung. *System identification: theory for the user*. Prentice-Hall, 1987.

[8] A. Rahimi and B. Recht. Estimating observation functions in dynamical systems using unsupervised regression. In *Advances in Neural Information Processing Systems (NIPS)*, 2006.

[9] A. Rahimi, B. Recht, and T. Darrell. Learning to transform time series with a few examples. *Pattern Analysis and Machine Intelligence (PAMI)*, to appear.

[10] H. Valpola and J. Karhunen. An unsupervised ensemble learning method for nonlinear dynamic state-space models. *Neural Computation*, 14(11):2647–2692, 2002.

[11] J. M. Wang, D. J. Fleet, and A. Hertzmann. Gaussian process dynamical models. In *Advances in Neural Information Processing Systems (NIPS)*, pages 1441–1448, 2005.