MANCHESTER 1824

The University of Manchester

Hierarchical Gaussian Process Latent Variable Models

Neil D. Lawrence neill@cs.man.ac.uk and Andrew J. Moore A.Moore@dcs.shef.ac.uk

School of Computer Science
University of Manchester, U.K.

Overview

The Gaussian process latent variable model (GP-LVM) is a powerful approach for probabilistic modelling of high dimensional data through dimensional reduction. In this paper we extend the GP-LVM through hierarchies. A hierarchical model (such as a tree) allows us to express conditional independencies in the data as well as the manifold structure. We first introduce Gaussian process hierarchies through a simple dynamical model, we then extend the approach to a more complex hierarchy which is applied to the visualisation of human motion data sets.

Introduction

- GP-LVM: an effective to probabilistic modelling of high dimensional data, assumes it lies on a manifold.
- An alternative to manifold representations: develop a latent variable model with sparse connectivity.
- Example: tree structured models for images [14, 4, 1], object recognition [3, 6] and human pose estimation [9, 10, 7].
- Tree structures offer a convenient way to specify conditional independencies in the model.
- We will show how we can construct our dimensionality reduction in a hierarchical way, exploiting the advantages of expressing conditional independencies and low dimensional non-linear manifolds.

Probabilistic Dimensional Reduction

- Formulate a latent variable model, with lower latent than data dimension, $q < d$.
  - Latent space prior distribution $p(\mathbf{X})$.
  - Mapping from latent ($\mathbf{x}_n$) to data space ($\mathbf{y}_n$)

$$y_{ni} = f_i(\mathbf{x}_n; \mathbf{W}) + \epsilon_n,$$

  $\mathbf{W}$ is a matrix of mapping parameters.
- For linear mappings and Gaussian priors: recover probabilistic PCA [11].
- For non-linear mapping: unclear how to propagate prior distribution to data space.

Dual Approach

- Place Gaussian Process prior over the mappings.
- Marginalise mappings:
  - for linear kernel a dual probabilistic PCA is recovered.
  - for non-linear kernel — a non linear probabilistic PCA.

GP-LVM

- Several advantages to marginalising the mapping:
  - *e.g.* adding dynamical priors in the latent space [13, 12]
  - constraining points in the latent space [8].
- Here we further exploit this characteristic, proposing the hierarchical Gaussian process latent variable model
- Introduce it by simple (one layered) hierarchical model for *dynamics.*

Dynamics via a Simple Hierarchy

- Standard latent space dynamical prior: $p(\mathbf{X}) = p(\mathbf{x}_1)\prod_{t=2}^{T} p(\mathbf{x}_t|\mathbf{x}_{t-1})$.
- Combine a with the GP-LVM likelihood and seek a *maximum a posteriori* (MAP) solution.
- Wang et al. [13] *autoregressive* Gaussian process prior to augment the GP-LVM with dynamics.
- Consider an *regressive* Gaussian process implementation of dynamics.
  - We place a Gaussian process prior over the latent space, the inputs are given by the time frame, $\mathbf{t}$.
  - Removes requirement for uniform sampling.
  - Allows the path in latent space to bifurcate.

Notation

- Motion capture sequence, $\mathbf{Y} = [\mathbf{y}_{1,:}, \dots, \mathbf{y}_{T,:}]^{\mathrm{T}} \in \Re^{T\times d}$
- GP-LVM Likelihood

$$p(\mathbf{Y}|\mathbf{X}) = \prod_{j=1}^{d} N(\mathbf{y}_{:,j}|\mathbf{0}, \mathbf{K}_x), \qquad (1)$$

- Latent variables, $\mathbf{X} = [\mathbf{x}_1, \dots, \mathbf{x}_T]^{\mathrm{T}} \in \Re^{T\times q}$,
- Kernel Matrix

$$k_x(\mathbf{x}_i, \mathbf{x}_j) = \sigma^2_{\text{rbf}} \exp\left(-\frac{\left\|\mathbf{x}_i - \mathbf{x}_j\right\|^2}{2l_x^2}\right) + \sigma^2_{\text{white}}\delta_{ij},$$

- Place a prior over the elements of $\mathbf{X}$.

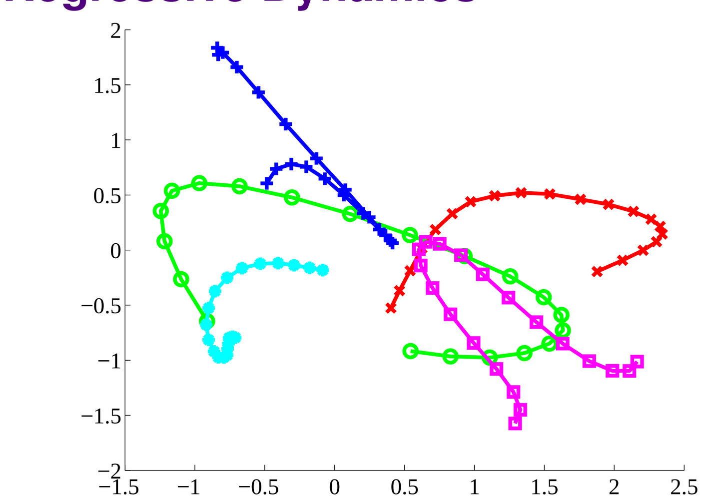
$$p(\mathbf{X}|\mathbf{t}) = \prod_{i=1}^{q} N(\mathbf{x}_{:,i}|\mathbf{0}, \mathbf{K}_t), \qquad (2)$$

  $\mathbf{t} \in \Re^{T\times 1}$ is vector of sample times.
- Kernel Matrix

$$k_t(t_i, t_j) = \varsigma^2_{\text{rbf}} \exp\left(-\frac{(t_i - t_j)^2}{2l_t^2}\right) + \varsigma^2_{\text{white}}.$$

  Samples shown below.

Regressive Dynamics

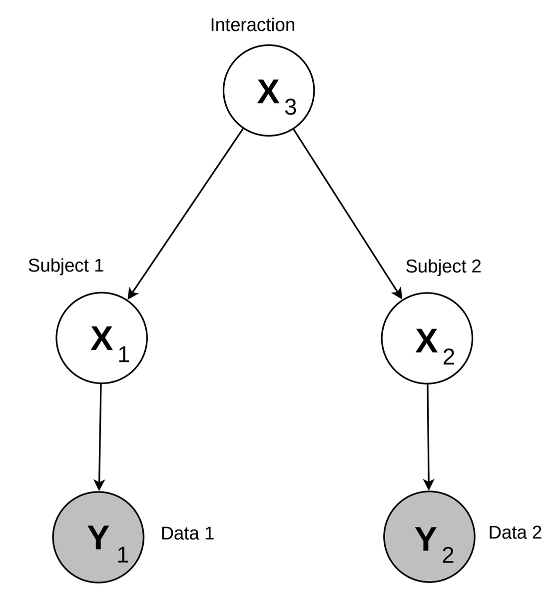


Typical sample paths for the RBF covariance function as temporal prior over the latent space.

More Complex Hierarchies

- This is a simple hierarchy. A Gaussian process prior on the latent space of a Gaussian process.
- Can we create more complex hierarchies and still find MAP solutions?
- We consider a motion capture example with multiple subjects interacting.
- Use a simple tree structure to model the subjects.

Subject Interaction



A simple hierarchy for capturing interaction between two subjects where $\mathbf{Y}_1$ is the data associated with subject 1, $\mathbf{Y}_2$ is that of subject 2. Each of these variable sets is then controlled by latent variables, $\mathbf{X}_1$ and $\mathbf{X}_2$. These latent variables are in turn controlled by $\mathbf{X}_3$.

Joint Probability

- The joint probability:

$$p(\mathbf{Y}_1, \mathbf{Y}_2) = \int p(\mathbf{Y}_1|\mathbf{X}_1) \int p(\mathbf{Y}_2|\mathbf{X}_2) \int p(\mathbf{X}_1, \mathbf{X}_2|\mathbf{X}_3)\, d\mathbf{X}_3 d\mathbf{X}_2 d\mathbf{X}_1$$

- Has an intractable integral.
- We therefore turn to MAP solutions for finding the values of the latent variables.
- Maximise

$$\log p(\mathbf{X}_1, \mathbf{X}_2\mathbf{X}_3|\mathbf{Y}_1, \mathbf{Y}_2) = \log p(\mathbf{Y}_1|\mathbf{X}_1) + \log p(\mathbf{Y}_2|\mathbf{X}_2) + \log p(\mathbf{X}_1, \mathbf{X}_2|\mathbf{X}_3)$$

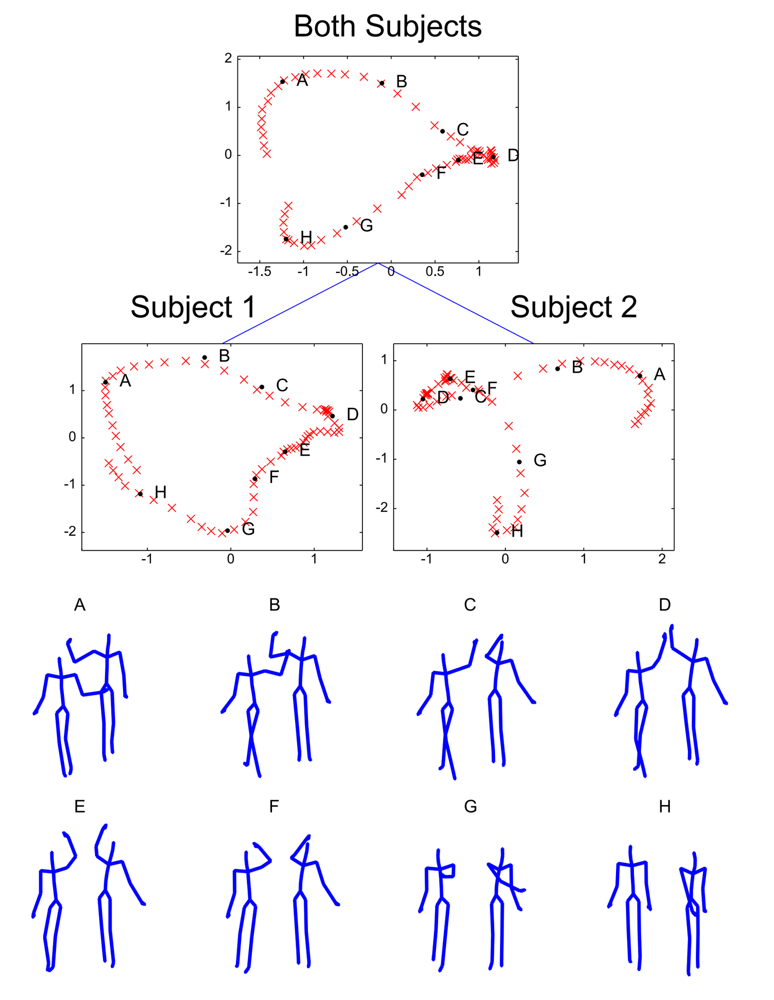
  first two terms for the subjects, third term provides coordination.

Two Interacting Subjects

- MOCAP data[a] consists of two subjects that 'high five'.
- The algorithm for optimisation of the latent variables proceeded as follows:
  1. Initialise each leaf node's latent variable set ($\mathbf{X}_1, \mathbf{X}_2$) through PCA of corresponding data set ($\mathbf{Y}_1$,$\mathbf{Y}_2$).
  2. Initialise the root node's latent variable set ($\mathbf{X}_3$) through PCA of concatenated latent variables of dependents $[\mathbf{X}_1\ \mathbf{X}_2]$.
  3. Optimise jointly the kernel parameters and latent positions $(\mathbf{X}_1, \mathbf{X}_2, \mathbf{X}_3)$.
- Used a variable sample rate to capture fast hand movements.
- The variable sample rate presents no problems for our regressive dynamics.

[a] http://mocap.cs.cmu.edu.

Interacting Subjects

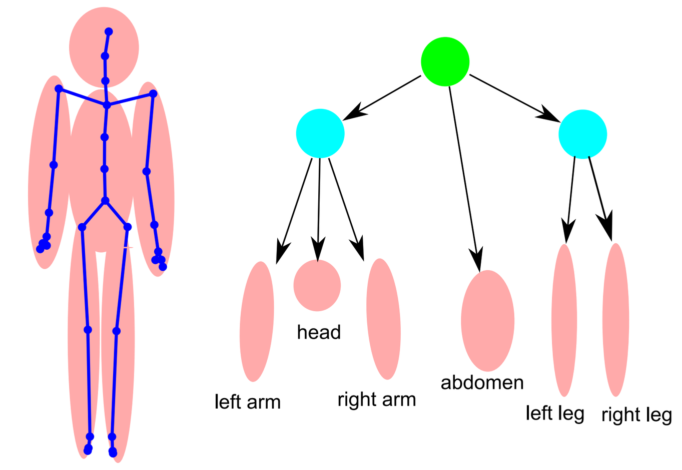


High Five example. Two subjects are modelled as they walk towards each other and 'high five'. Plot shows the simple hierarchy used to model the data. There is a regressive dynamical prior placed over the latent space of the root node. The root node then controls the two individual subjects. To illustrate the model we have taken points at time (i.e we input these values of t into the dynamical model) frames A: 85, B: 114, C:127, D: 141, E: 155, F: 170, G: 190 and H: 215. These points mapped down through the hierarchy and into the data space. In each of the plots of the two subjects, Subject 1 is on the right and Subject 2 is on the left.

Subject Decomposition

- We can also consider decomposition of a single subject into parts.
- Most tree based approaches to mocap modelling assume the nodes are observed and the tree reflects skeletal structure.
- Our hierarchical model is more similar to [14] where the tree structure is a *hierarchy of latent variables.*

Skeleton Decomposition

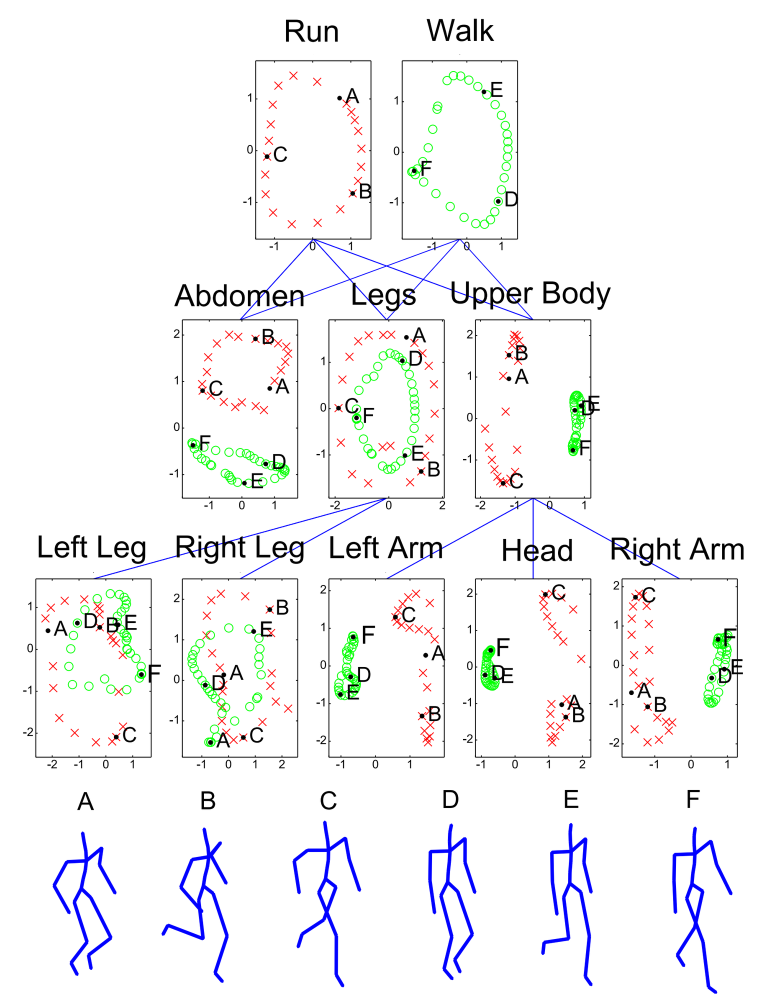


Decomposition of skeleton for hierarchical modelling. By separating the component parts of the skeleton in this manner we can model the space of natural motions for each component part and express them independently or jointly.

Decomposition in a Walk and Run

- Data set composed of a walking motion and a running motion.
- Data sub-sampled to 30 frames per second and one cycle of each motion was used.
- We modelled the subject using the decomposition shown in above.
- To reflect the fact that two different motions were present in the data we constructed a hierarchy with *two roots*. One for the run and one for the walk.
- This construction enables us to express the two motion sequences separately while sharing information lower in the hierarchy.
- We used a periodic kernel for the regressive dynamics.

Decomposition Result



Combined model of a run and a walk. The skeleton is decomposed as shown in Figure . In the plots, crosses are latent positions associated with the run and circles are associated with the walk. We have mapped three points from each motion through the hierarchy. Periodic dynamics was used in the latent spaces.

Summary

- Introduced a a hierarchical version of the GP-LVM
- We use MAP approximations to fit latent variables in all different levels of the hierarchy.

Overfitting

- GP-LVM uses a large number of 'parameters' in the form of latent points. Why doesn't it overfit?
- Standard GP-LVM: parameters increase linearly $\frac{q}{d} \times N$. If $q < d$ overfitting not a problem.
- HGP-LVM: we are adding more latent variables, will we overfit?
  - Upper levels of hierarchy only regularise the leaf nodes: if the leaf nodes don't overfit neither will the model.
  - By modifying the locations of latent variables we are changing the *regularisation* of the leaf nodes.
  - If unconstrained the model would simply remove the regularisation.
  - We counter this potential problem in two ways.
    1. Provided a fixed dynamical prior at the top level.
    2. Constrained the noise variance of each non-leaf Gaussian process to 1×10^{-6}.

Other Hierarchical Models

- The model is not closely related to hierarchical PCA [2].
- Hierarchical PCA the hierarchy is not a hierarchy of latent variables but a hierarchical decomposition of the component probabilities.

Applications

- Two application areas of promise are:
  - Tracking: GP-LVM is used as a prior model in tracking, the hierarchy would allow different components to be swapped in as motion changed and 'back off'.
  - Animation: animator time is expensive. Through combining the hierarchical model with style based inverse kinematics [5] these costs could be reduced.

Acknowledgments

This work was funded by the EU FP6 PASCAL Network of Excellence under a pump priming grant. The motion capture data used in this project was obtained from mocap.cs.cmu.edu. The database was created with funding from NSF EIA-0196217. We thank Raquel Urtasun for helpful discussions.

Recreating the Experiments

The source code for re-running all the experiments detailed here is available from http://www.cs.man.ac.uk/~neill/hgplvm/, release 0.1.

References

[1] P. Awasthi, A. Gagrani, and B. Ravindran. In M. M. Veloso, editor, *Proceedings of the 20th International Joint Conference on Artificial Intelligence (IJCAI 2007)*, pages 2060–2065, 2007.

[2] C. M. Bishop and M. E. Tipping. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(3):281–293, 1998.

[3] P. F. Felzenszwalb and D. P. Huttenlocher. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 2, pages 66–73, Hilton Head Island, South Carolina, U.S.A., 13–15 Jun. 2000. IEEE Computer Society Press.

[4] X. Feng, C. K. I. Williams, and S. N. Felderhof. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(4):467–483, 2002.

[5] K. Grochow, S. L. Martin, A. Hertzmann, and Z. Popovic. In *ACM Transactions on Graphics (SIGGRAPH 2004)*, pages 522–531, 2004.

[6] S. Ioffe and D. A. Forsyth. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 2, pages 180–185, Hawaii, U.S.A., 11–13 Dec. 2001. IEEE Computer Society Press.

[7] X. Lan and D. P. Huttenlocher. In *IEEE International Conference on Computer Vision (ICCV)*, volume 1, pages 470–477, Bejing, China, 17–21 Oct. 2005. IEEE Computer Society Press.

[8] N. D. Lawrence and J. Quiñonero Candela. In W. Cohen and A. Moore, editors, *Proceedings of the International Conference in Machine Learning*, volume 23, pages 513–520. Omnipress, 2006. ISBN 1-59593-383-2.

[9] D. Ramanan and D. A. Forsyth. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 2, pages 467–474, Madison, Wisconsin, U.S.A., 18–20 Jun. 2003. IEEE Computer Society Press.

[10] L. Sigal, S. Bhatia, S. Roth, M. J. Black, and M. Isard. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 421–428, Washington, DC, U.S.A., 29 Jun.–1 Jul. 2004. IEEE Computer Society Press.

[11] M. E. Tipping and C. M. Bishop. *Journal of the Royal Statistical Society, B*, 6(3):611–622, 1999.

[12] R. Urtasun, D. J. Fleet, and P. Fua. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 238–245, New York, U.S.A., 17–22 Jun. 2006. IEEE Computer Society Press.

[13] J. M. Wang, D. J. Fleet, and A. Hertzmann. In Y. Weiss, B. Schölkopf, and J. C. Platt, editors, *Advances in Neural Information Processing Systems*, volume 18, Cambridge, MA, 2006. MIT Press.

[14] C. K. I. Williams and X. Feng. In *Proceedings Ninth International Conference on Artificial Neural Networks, ICANN'99*, 1999.