The Variational Coupled Gaussian Process Dynamical Model

Dmytro Velychko, Dominik Endres
Theoretical Neuroscience Group, Department of Psychology
Philipps-University, 35032 Marburg, Germany
velychko@staff.uni-marburg.de,dominik.endres@uni-marburg.de

Abstract

We present a full variational treatment of the Coupled Gaussian Process Dynamical Model (CGPDM), which is a non-parametric, modular dynamical movement primitive model. Our work builds on similar developments in Gaussian state-space models, but we obviate the need for sampling, which results in a fast deterministic approximation for the posterior of latent states. We illustrate the performance of our model on synthetic data.

1 Motivation

Planning and execution of full-body movements is a formidable control problem. Modular movement primitives (MP) have been suggested as a means to simplify this control problem while retaining a sufficient degree of control flexibility for a wide range of task, see Bizzi et al. [2008] for a recent review. 'Modular' in this context usually refers to the existence of an operation which allows for the combination of (simple) primitives into (complex) movements. Thus, MPs are a type of compressed representation of the movements of the end-effectors (arms, legs etc).

A particularly well-developed type of MP in robotics is the dynamical movement primitive (DMP) [Schaal, 2006]. In this approach, each primitive is encoded by a second order differential equation with guaranteed stability properties and learnable parameters. However, the form of the differential equation remains fixed during learning, which can potentially reduce the representational capacity of DMPs. To lift this restriction, we devised a model that learns MPs comprised of coupled dynamical systems and associated kinematics mappings, where *both* components are learned. We build on the Coupled Gaussian Process Dynamical Model (CGPDM) by Velychko et al. [2014], which combines the advantages of modularity and flexibility in the dynamics, at least theoretically. In a CGPDM, the temporal evolution functions for latent dynamical systems are drawn out of a Gaussian process (GP) prior [Rasmussen and Williams, 2005, Wang et al., 2008]. These dynamical systems are then coupled probabilistically, the result is mapped onto observations by functions drawn from another GP. One drawback of the CGPDM is its fully non-parametric nature, which leads to a prohibitive run-time scaling for large datasets. We improve this scaling by employing sparse variational approximations [Titsias, 2009, Titsias and Lawrence, 2010, Frigola et al., 2013, 2014] and obviating the need for sampling. In our model, each MP is effectively parametrized by a small set of inducing variables, leading to a compact representation. This compactness is important for real-world applicability of the model, since there might be more primitives than muscles (or actuators) across tasks, as pointed out by Bizzi and Cheung [2013]: the motor 'code' might be sparse and overcomplete, similar to the sparse codes in early vision [Földiák and Endres, 2008].

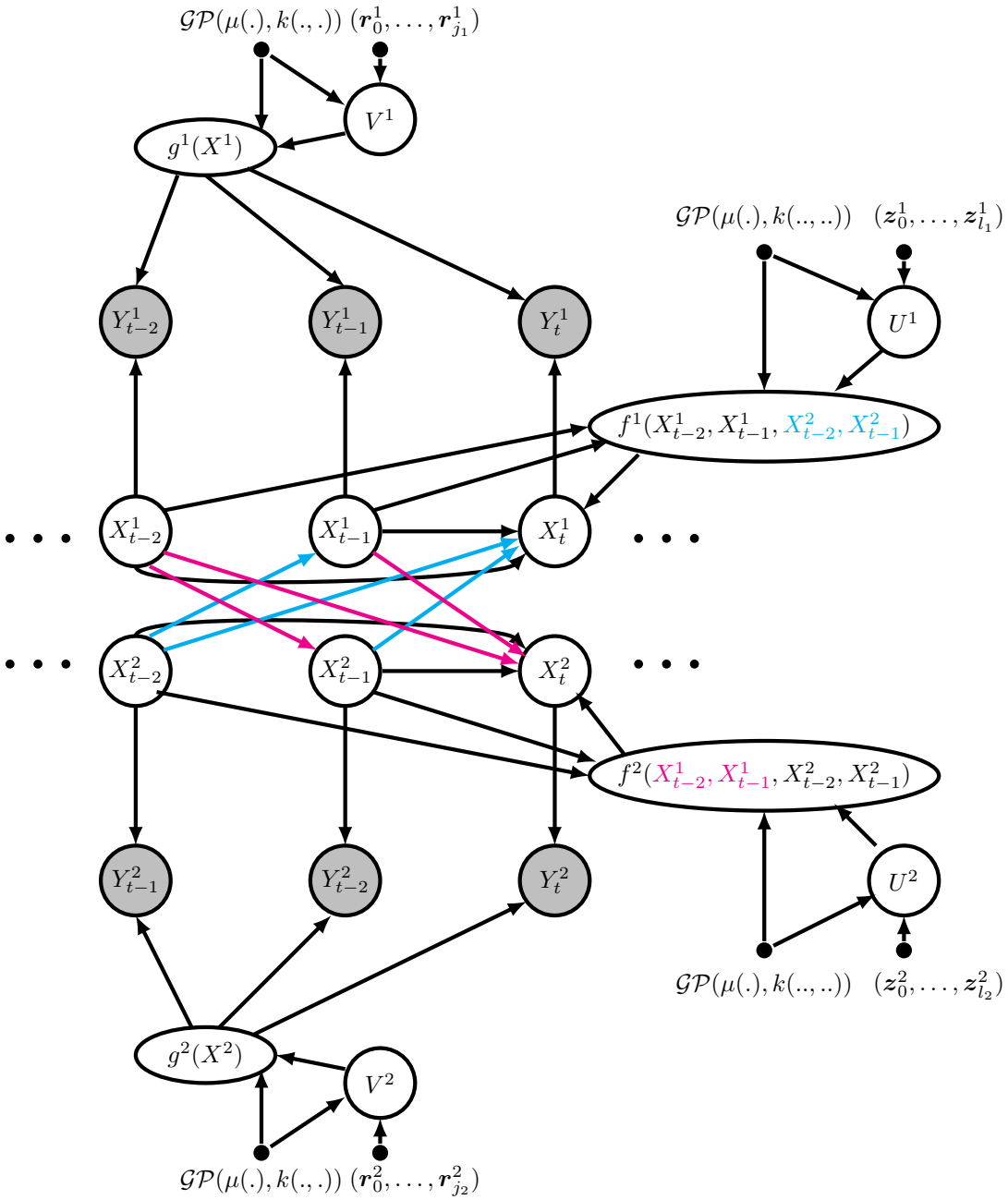

Figure 1: The graphical model representation of the augmented Coupled Gaussian Process Dynamical Model (CGPDM). Shown is a model with 2 parts, indicated by the superscripts $m \in \{1, 2\}$. Latent, vector-valued dynamical random variables X_t^m generate (vector-valued) observations Y_t^m via functions $g^m(\boldsymbol{x}_t^1)$ drawn from a $\mathcal{GP}$ augmented with inducing points $\boldsymbol{r}_{j_m}^m$ and associated values $\boldsymbol{v}_{j_m}^m$ such that $g^m(\boldsymbol{r}^m) = \boldsymbol{v}^m$ and $(\boldsymbol{v}_0^m, \ldots, \boldsymbol{v}_{j_m}^m) \in \text{range}(V^m)$ [Titsias, 2009]. Gaussian noise η is added to the observations, i.e. $\boldsymbol{y}_t^m = g^m(\boldsymbol{x}_t^m) + \eta$ (not shown). The second-order mean dynamics functions $f^m(X_{t-2}^1, X_{t-1}^1, X_{t-2}^2, X_{t-1}^2)$, which govern the temporal evolution of the mean of X_t^m, $\boldsymbol{x}_t^m = f^m(\boldsymbol{x}_{t-2}^1, \boldsymbol{x}_{t-1}^1, \boldsymbol{x}_{t-2}^2, \boldsymbol{x}_{t-1}^2)$, are also drawn from a $\mathcal{GP}$ augmented with $(\boldsymbol{z}_0^2, \ldots, \boldsymbol{z}_{l_2}^2)$ and $(\boldsymbol{u}_0^m, \ldots, \boldsymbol{u}_{l_m}^m) \in \text{range}(U^m)$. The cyan and magenta connections couple part one (top half) to part two of the model (bottom half) by product-of-experts [Hinton, 1999] combination of their individual predictions with variable coupling strengths.

2 The model

A CGPDM is basically a number of GPDMs (the 'parts') run in parallel, with coupling between the latent space dynamics. Fig. 1 shows a model with 2 parts (top and bottom half of figure). Each part $m \in \{1, 2\}$ is comprised of a latent, second-order dynamics model $f^m(X^1_{t-2}, X^1_{t-1}, X^2_{t-2}, X^2_{t-1})$ drawn from a $\mathcal{GP}(\mu(.), k(., .))$. We chose a second-order model, because our target application is human movement modeling, and the literature indicates (e.g. [Taubert et al., 2012]) that a second-order model is a good choice for this task. In a CGPDM, the $\mathcal{GP}(\mu(.), k(., .))$ has a constant zero mean function $\mu(.) = 0$ and a kernel that is derived with product-of-experts (PoE, [Hinton, 1999]) coupling between the latent spaces of the different parts, as described by [Velychko et al., 2014]: each part generates a Gaussian prediction about every part (i.e. including itself). Let $\boldsymbol{x}_t^{m,n}$ be the mean of the prediction of part m about part n at time index t, and $\sigma^2_{m,n}$ its variance. Following the standard PoE construction of multiplying the densities of the individual predictions and renormalizing, one finds

$$\begin{aligned} p(\boldsymbol{x}_t^n | \boldsymbol{x}_i^{:,n}, \sigma^2_{:,n}) &\propto \prod_m \mathcal{N}\left(\boldsymbol{x}_t^n | \boldsymbol{x}_i^{m,n}, \sigma^2_{m,n}\right) \\ p(\boldsymbol{x}_t^n | \boldsymbol{x}_t^{:,n}, \sigma_n^2) &= \frac{\exp\left[-\frac{1}{2\sigma_n^2}\left(\boldsymbol{x}_t^n - \sigma_n^2 \sum_m \frac{\boldsymbol{x}_t^{m,n}}{\sigma^2_{m,n}}\right)^2\right]}{(2\pi\sigma_n^2)^{\frac{\dim(X^n)}{2}}} \end{aligned} \tag{1}$$

where $\dim(X^n)$ is the dimensionality of the latent space of part n and σ_n^2 is the total predictive variance:

$$\sigma_n^2 = \left(\sum_m \sigma^{-2}_{m,n}\right)^{-1}. \tag{2}$$

Velychko et al. [2014] showed that the individual predictions $\boldsymbol{x}_t^{m,n}$ can be marginalized out in closed form, if each of them is generated by a function drawn from a GP with mean zero and kernel $k_{m,n}(., .)$. In that case, one obtains a model where the dynamics function $f^n(...)$ of each part n is drawn from a GP with zero mean function and kernel

$$k_n\left(\boldsymbol{X}, \boldsymbol{X}'\right) = \sigma_n^2 \delta(\boldsymbol{X}, \boldsymbol{X}') + \sigma_n^4 \sum_m \frac{k_{m,n}(\boldsymbol{X}^m, \boldsymbol{X}^{m\prime})}{\sigma^4_{m,n}} \tag{3}$$

where $\boldsymbol{X}^m$ is obtained by stacking the latent variable values $\boldsymbol{x}^m_{t-2}, \boldsymbol{x}^m_{t-1}$ into a vector and $\boldsymbol{X}$ is the tuple of all $\boldsymbol{X}^m$. In the graphical model in fig. 1, this marginalization has been carried out. The form of eqn. 3 indicates the function of the coupling variances: the smaller a given variance, the more important is the prediction of the generating part. Thus, the $\sigma^2_{m,n}$ can not only be learned from data, but also modulated *after* learning to generate movements with different dynamics [Velychko et al., 2014]. The latent states $\boldsymbol{x}_t^m$ generate Gaussian-distributed observations $Y_t^m = \boldsymbol{y}_t^m$

The basic CGPDM exhibits the usual cubic runtime scaling with the number of datapoints, which prohibits learning from large datasets. We therefore developed a sparse variational approximation, following the treatment in [Titsias, 2009]. We augment the model with inducing points $\boldsymbol{r}^m_{j_m}$ and associated values $\boldsymbol{v}^m_{j_m}$ such that $g^m(\boldsymbol{r}^m) = \boldsymbol{v}^m$ for the latent-to-observed mappings $g^m(X^m)$, and condition the probability density of the function values of $g^m(.)$ on these points/values, which we assume to be a sufficient statistic for $g^m(X^m)$. We apply the same augmentation strategy to reduce the computational effort for learning the dynamics mappings, which are induced by $(\boldsymbol{z}^m_0, \ldots, \boldsymbol{z}^m_{l_m})$ and $(\boldsymbol{u}^m_0, \ldots, \boldsymbol{u}^m_{l_m}) \in \text{range}(U^m)$, see fig. 1. To learn the model, we optimize the proposal posterior distribution $q(\boldsymbol{X})$ to maximize the "evidence lower bound" (ELBO) $\mathcal{L}(q)$ [Bishop, 2006]:

$$\mathcal{L}(q) = \int q(\boldsymbol{X}) \log \frac{p(\boldsymbol{X}, \boldsymbol{Y})}{q(\boldsymbol{X})} d\boldsymbol{X} \tag{4}$$

Key assumption: to obtain a tractable variational posterior distribution q over the latent states $\boldsymbol{x}_t^m = (x^m_{t,1}, \ldots, x^m_{t,d_m})$, we choose a distribution that factorizes across time steps $0, \ldots, T$, parts $1, \ldots, M$ and dimensions $1, \ldots, Q_m$ within parts:

$$q(\boldsymbol{x}_0^0, \ldots, \boldsymbol{x}_T^M) = \prod_{t=0}^{T} \prod_{m=1}^{M} \prod_{i=1}^{Q_m} q(x^m_{t,i}) \tag{5}$$

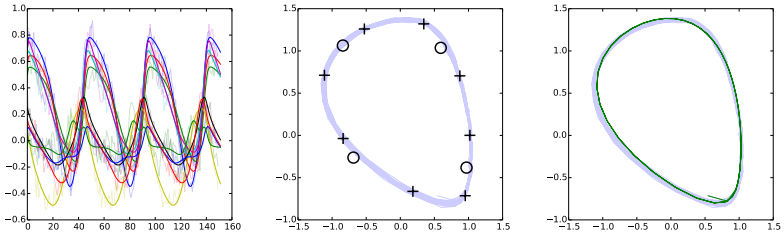

Figure 2: Learning a latent dynamical system on toy data. **Left**: training trajectories (light colors) and generated trajectories after training (solid color) for two out of 10 observed dimensions. The generated trajectories follow the mean of the training trajectories. **Middle**: latent trajectories (light blue) and learned inducing point locations (crosses: GPLVM component, circles: dynamics model). Inducing points cover the training trajectories in the latent space well, and a small number of inducing points seems to be enough for the generation of the trajectories, as can be seen on the **right**: latent trajectories (light blue) and generated trajectory after training (solid green).

and assume that the individual distributions are Gaussian:

$$q(\boldsymbol{x}_{t,i}^m) = \mathcal{N}(\mu_{t,i}^m, \sigma_{t,i}^{2,m}). \tag{6}$$

While this approximating assumption is clearly a gross simplification of the correct latent state posterior, it allows us to make analytical progress: a variational lower bound on the marginal likelihood will be a sum of two main contributions. First, the latent-to-observed component of the model is now a variational GPLVM as described by Titsias and Lawrence [2010]. Hence, the results of that paper can be reused without alteration. Second, the dynamics component. We can derive closed-form expressions for the required integrals for certain kernels $k_n\left(\boldsymbol{X}, \boldsymbol{X}'\right)$. Specifically, we use an ARD (automatic relevance detection) squared exponential kernel [Bishop, 2006] for every part-m-to-n prediction GP:

$$k_{m,n}(\boldsymbol{X}^m, \boldsymbol{X}^{m\prime}) = \sigma_f^2 \exp\left(-\frac{1}{2}\sum_i \frac{(\boldsymbol{X}_i^m - \boldsymbol{X}_i^{m\prime})^2}{\lambda_i}\right). \tag{7}$$

Since the individual $k_{m,n}(\boldsymbol{X}^m, \boldsymbol{X}^{m,\prime})$ enter linearly into the coupling kernel (eqn. 3), the averages over the $q(x_{t,i}^m)$ can be carried out before the coupling kernel is assembled, i.e. we can compute the solution for one part at a time and compute sums weighted by $\frac{\sigma_n^4}{\sigma_{m,n}^4}$ afterwards. The computations required for this are lengthy (and error-prone) but straightforward, and resemble those for the variational GPLVM of Titsias and Lawrence [2010]. We will present the details in a technical report that will soon be uploaded to the ArXiv.

3 Results

We implemented the model in `Python 2.7` using the machine-learning framework `Theano` [Bastien et al., 2012] for automatic differentiation to enable gradient-based maximization of the variational lower bound on the marginal probability of the data with the `scipy.optimize.fmin_l_bfgs_b` routine [Jones et al., 2001–]. To illustrate the performance of our model, we created artificial data by sampling form a CGPDM with 2 parts, 2 dimensions in the latent space and 10 observed dimensions per part. We learned a model of 2 parts. Fig. 2 shows learned and generated trajectories in both observed and latent space. It is evident from the plots that the model learns to generalize the training data well: numerically speaking, the explained variance on the training data is ≈ 0.93, and ≈ 0.91 on testing data not used for training. Our current work is focused on more extensive experiments with synthetic data, and a large database of human motion capture data. The latter is now feasible because the runtime complexity of the augmented sparse CGPDM scales linearly with the datapoints.

Acknowledgements: the authors acknowledge funding from DFG under IRTG 1901 'The Brain in Action' and the European Union Seventh Framework Program (FP7/2007 - 2013) under grant agreement no 611909 (KoroiBot).

References

F. Bastien, P. Lamblin, R. Pascanu, J. Bergstra, I. J. Goodfellow, A. Bergeron, N. Bouchard, and Y. Bengio. Theano: new features and speed improvements. Deep Learning and Unsupervised Feature Learning NIPS 2012 Workshop, 2012.

C. M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2006. ISBN 0387310738.

E. Bizzi and V. C. Cheung. The neural origin of muscle synergies. *Frontiers in Computational Neuroscience*, 7(51), 2013. ISSN 1662-5188. doi: 10.3389/fncom.2013.00051. URL http://www.frontiersin.org/computational_neuroscience/10.3389/fncom.2013.00051/abstract.

E. Bizzi, V. Cheung, A. d'Avella, P. Saltiel, and M. Tresch. Combining modules for movement. *Brain Research Reviews*, 57(1):125 – 133, 2008. ISSN 0165-0173. doi: http://dx.doi.org/10.1016/j.brainresrev.2007.08.004. URL http://www.sciencedirect.com/science/article/pii/S0165017307001774. Networks in Motion.

P. Földiák and D. Endres. Sparse coding. *Scholarpedia*, 3(1):2984, 2008. ISSN 1941-6016. URL http://www.scholarpedia.org/article/Sparse_coding.

R. Frigola, F. Lindsten, T. B. Schön, and C. Rasmussen. Bayesian inference and learning in gaussian process state-space models with particle mcmc. In C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Weinberger, editors, *Advances in Neural Information Processing Systems 26*, pages 3156–3164. Curran Associates, Inc., 2013.

R. Frigola, Y. Chen, and C. Rasmussen. Variational gaussian process state-space models. In Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K. Weinberger, editors, *Advances in Neural Information Processing Systems 27*, pages 3680–3688. Curran Associates, Inc., 2014. URL http://papers.nips.cc/paper/5375-variational-gaussian-process-state-space-models.pdf.

G. E. Hinton. Products of experts. In *Proc. ICANN'99*, volume 1, pages 1–6, 1999.

E. Jones, T. Oliphant, P. Peterson, et al. SciPy: Open source scientific tools for Python, 2001–. URL http://www.scipy.org/. [Online; accessed 2015-10-09].

C. E. Rasmussen and C. K. I. Williams. *Gaussian Processes for Machine Learning (Adaptive Computation and Machine Learning)*. The MIT Press, 2005. ISBN 026218253X.

S. Schaal. Dynamic movement primitives -a framework for motor control in humans and humanoid robotics. In H. Kimura, K. Tsuchiya, A. Ishiguro, and H. Witte, editors, *Adaptive Motion of Animals and Machines*, pages 261–280. Springer Tokyo, 2006. ISBN 978-4-431-24164-5. doi: 10.1007/4-431-31381-8_23. URL http://dx.doi.org/10.1007/4-431-31381-8_23.

N. Taubert, A. Christensen, D. Endres, and M. Giese. Online simulation of emotional interactive behaviors with hierarchical gaussian process dynamical models. In *Proceedings of the ACM Symposium on Applied Perception*, pages 25–32. ACM, 2012. doi: 10.1145/2338676.2338682.

M. K. Titsias. Variational learning of inducing variables in sparse gaussian processes. In D. A. V. Dyk and M. Welling, editors, *AISTATS*, volume 5 of *JMLR Proceedings*, pages 567–574. JMLR.org, 2009.

M. K. Titsias and N. D. Lawrence. Bayesian gaussian process latent variable model. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics, AISTATS 2010, Chia Laguna Resort, Sardinia, Italy, May 13-15, 2010*, pages 844–851, 2010. URL http://www.jmlr.org/proceedings/papers/v9/titsias10a.html.

D. Velychko, D. Endres, N. Taubert, and M. A. Giese. Coupling Gaussian process dynamical models with product-of-experts kernels. In *Proceedings of the 24th International Conference on Artificial Neural Networks, LNCS 8681*, pages 603–610. Springer, 2014. doi: 10.1007/978-3-319-11179-7_76.

J. M. Wang, D. J. Fleet, and A. Hertzmann. Gaussian process dynamical models for human motion. *IEEE Trans. Pattern Anal. Mach. Intell.*, 30(2):283–298, 2008.