Universidad Tecnológica de Pereira
Grupo de Investigación en Automática

Definition and Composition of Motor Primitives using Latent Force Models and Hidden Markov Models

Diego Agudelo-España Mauricio A. Álvarez
{dialagudelo,malvarez}@utp.edu.co

Maestría en Ingeniería Eléctrica

Universidad Tecnológica de Pereira

Grupo en Ecuaciones Diferenciales y Aplicaciones (GREDYA)
November 30, 2016

Outline

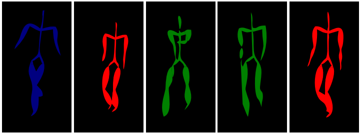
Motivation

- How can we represent movement for robot control and motion synthesis problems?

Photo taken from http://mocap.cs.cmu.edu

Motivation

- The use of a limited set of modifiable templates seems to be the only way to cope with the **high dimensionality nature** of movement representation and robot control problems [Schaal, 1999].
- **Biological evidence** suggests that movements performed by living beings can be represented as a combination of a limited number of motor primitives [Flash and Hochner, 2005] which are the basic building blocks of more complex movements.
- The main concerns about motor primitives are logically **how to define them** and **how to combine them** for building complex movements.



What is Machine Learning?

Aim: endow computers with the ability to "learn" from "data".

Data + Model = Prediction

- Data = observations.
- Model = assumptions, based on previous experience, or beliefs about the regularities of the universe.
- Prediction = an action to be taken or a categorization or a quality score.

Taken from the GPSS lectures by Neil D. Lawrence (http://gpss.cc/gprs14/assets/session1.pdf)

What is Machine Learning?

Aim: endow computers with the ability to "learn" from "data".

Data + Model = Prediction

- Data = observations.
- Model = assumptions, based on previous experience, or beliefs about the regularities of the universe.
- Prediction = an action to be taken or a categorization or a quality score.

Taken from the GPSS lectures by Neil D. Lawrence (http://gpss.cc/gprs14/assets/session1.pdf)

What is Machine Learning?

Aim: endow computers with the ability to "learn" from "data".

Data + Model = Prediction

- Data = observations.
- Model = assumptions, based on previous experience, or beliefs about the regularities of the universe.
- Prediction = an action to be taken or a categorization or a quality score.

Taken from the GPSS lectures by Neil D. Lawrence (http://gpss.cc/gprs14/assets/session1.pdf)

What is Machine Learning?

Aim: endow computers with the ability to "learn" from "data".

Data + Model = Prediction

- Data = observations.
- Model = assumptions, based on previous experience, or beliefs about the regularities of the universe.
- Prediction = an action to be taken or a categorization or a quality score.

Taken from the GPSS lectures by Neil D. Lawrence (http://gpss.cc/gprs14/assets/session1.pdf)

Gaussian Processes

A Gaussian Process (GP) is a collection of random variables, any finite number of which follows a joint Gaussian distribution [Rasmussen, 2006]. A GP is fully specified by a mean function $m(\mathbf{x})$ and a covariance function $k(\mathbf{x}, \mathbf{x}')$,

$$f(\mathbf{x}) \sim \mathcal{GP}(m(\mathbf{x}), k(\mathbf{x}.\mathbf{x}')),$$

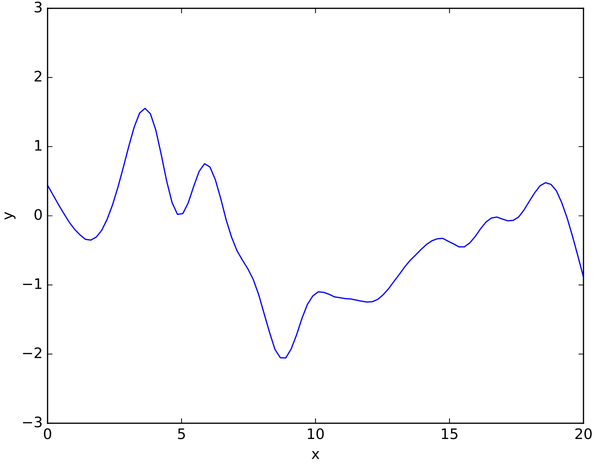
Interestingly, any finite set of random variables $\{f(x_i), f(x_j), \ldots, f(x_k)\}$ over f is distributed according to

$$\begin{pmatrix} f(x_i) \\ f(x_j) \\ \vdots \\ f(x_k) \end{pmatrix} \sim \mathcal{N}\left(\begin{pmatrix} m(x_i) \\ m(x_j) \\ \vdots \\ m(x_k) \end{pmatrix}, \begin{pmatrix} k(x_i, x_i) & k(x_i, x_j) & \ldots & k(x_i, x_k) \\ k(x_j, x_i) & k(x_j, x_j) & \ldots & k(x_j, x_k) \\ \vdots & \vdots & \ddots & \vdots \\ k(x_k, x_i) & k(x_k, x_j) & \ldots & k(x_k, x_k) \end{pmatrix}\right),$$

as a consequence of the marginal of a GP being a Gaussian distribution.

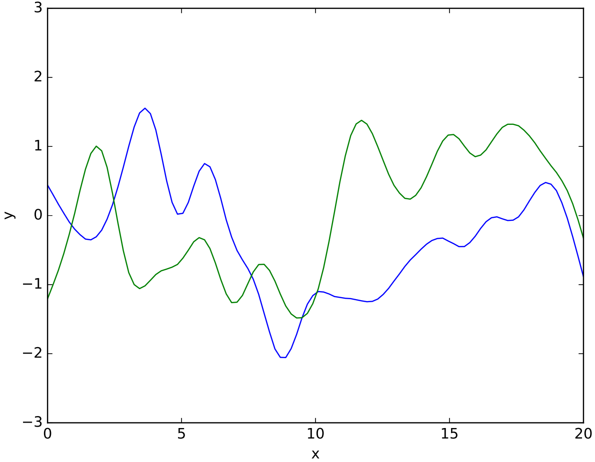
Gaussian Processes Examples

$$f(\mathbf{x}) \sim \mathcal{GP}(m(\mathbf{x}), k_{SE}(\mathbf{x}.\mathbf{x}')), \quad m(\mathbf{x}) = \mathbf{0}, \quad k_{SE}(\mathbf{x}.\mathbf{x}')) = \sigma^2 \exp\left\{ \frac{\|\mathbf{x} - \mathbf{x}'\|^2}{2l^2} \right\}$$



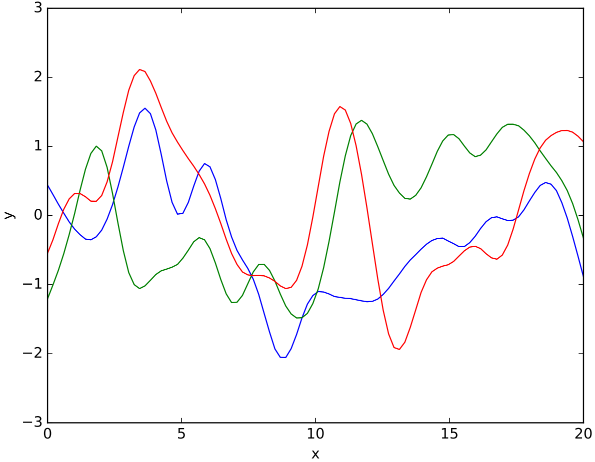
Gaussian Processes Examples

$$f(\mathbf{x}) \sim \mathcal{GP}(m(\mathbf{x}), k_{SE}(\mathbf{x}.\mathbf{x}')), \quad m(\mathbf{x}) = \mathbf{0}, \quad k_{SE}(\mathbf{x}.\mathbf{x}')) = \sigma^2 \exp\left\{ \frac{\|\mathbf{x} - \mathbf{x}'\|^2}{2l^2} \right\}$$



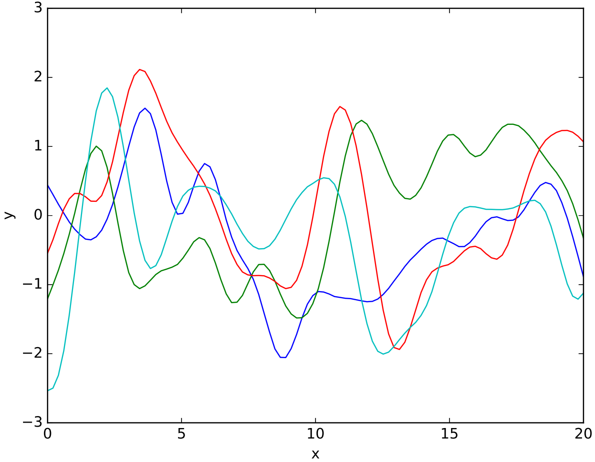
Gaussian Processes Examples

$$f(\mathbf{x}) \sim \mathcal{GP}(m(\mathbf{x}), k_{SE}(\mathbf{x}.\mathbf{x}')), \quad m(\mathbf{x}) = \mathbf{0}, \quad k_{SE}(\mathbf{x}.\mathbf{x}')) = \sigma^2 \exp\left\{ \frac{\|\mathbf{x} - \mathbf{x}'\|^2}{2l^2} \right\}$$



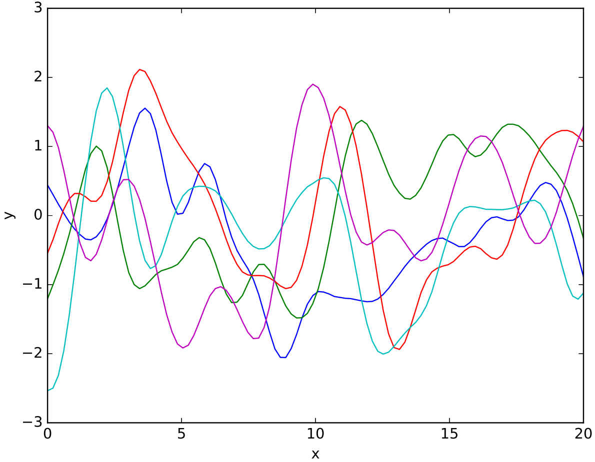
Gaussian Processes Examples

$$f(\mathbf{x}) \sim \mathcal{GP}(m(\mathbf{x}), k_{SE}(\mathbf{x}.\mathbf{x}')), \quad m(\mathbf{x}) = \mathbf{0}, \quad k_{SE}(\mathbf{x}.\mathbf{x}')) = \sigma^2 \exp\left\{\frac{\|\mathbf{x} - \mathbf{x}'\|^2}{2l^2}\right\}$$



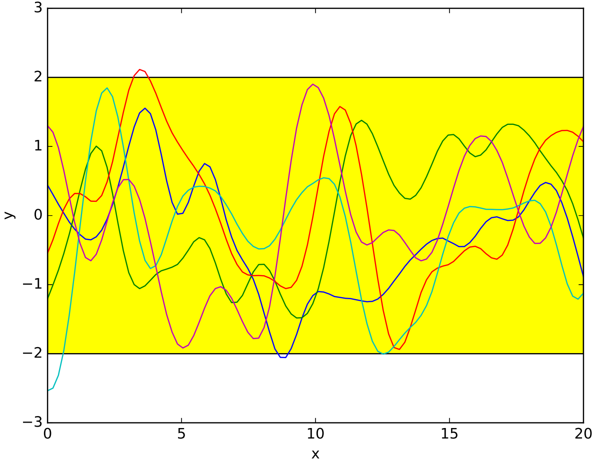
Gaussian Processes Examples

$$f(\mathbf{x}) \sim \mathcal{GP}(m(\mathbf{x}), k_{SE}(\mathbf{x}.\mathbf{x}')), \quad m(\mathbf{x}) = \mathbf{0}, \quad k_{SE}(\mathbf{x}.\mathbf{x}')) = \sigma^2 \exp\left\{ \frac{\|\mathbf{x} - \mathbf{x}'\|^2}{2l^2} \right\}$$



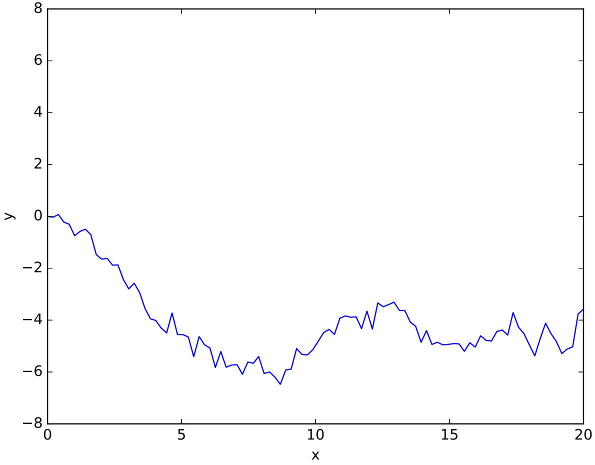
Gaussian Processes Examples

$$f(\mathbf{x}) \sim \mathcal{GP}(m(\mathbf{x}), k_{SE}(\mathbf{x}.\mathbf{x}')), \quad m(\mathbf{x}) = \mathbf{0}, \quad k_{SE}(\mathbf{x}.\mathbf{x}')) = \sigma^2 \exp\left\{ \frac{\|\mathbf{x} - \mathbf{x}'\|^2}{2l^2} \right\}$$



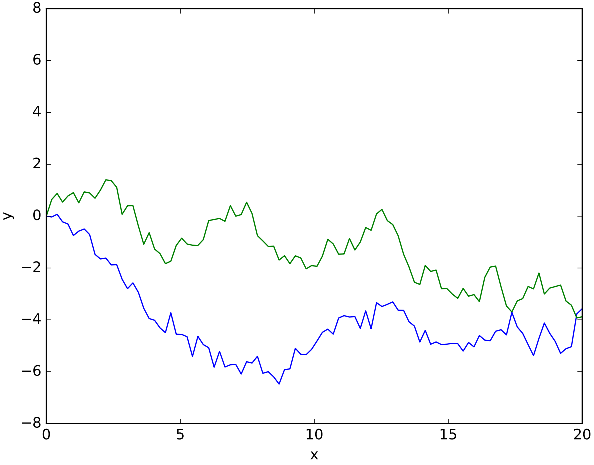
Gaussian Processes Examples

$$f(\mathbf{x}) \sim \mathcal{GP}(m(\mathbf{x}), k(\mathbf{x}.\mathbf{x}')), \quad m(\mathbf{x}) = \mathbf{0}, \quad k(\mathbf{x}.\mathbf{x}')) = \min(\mathbf{x}, \mathbf{x}')$$



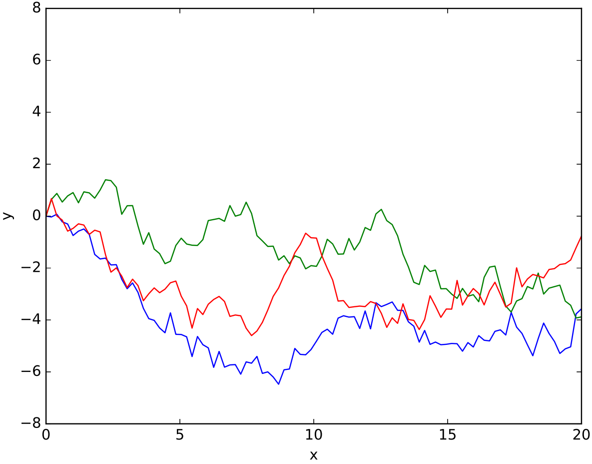
Gaussian Processes Examples

$$f(\mathbf{x}) \sim \mathcal{GP}(m(\mathbf{x}), k(\mathbf{x}.\mathbf{x}')), \quad m(\mathbf{x}) = \mathbf{0}, \quad k(\mathbf{x}.\mathbf{x}')) = \min(\mathbf{x}, \mathbf{x}')$$



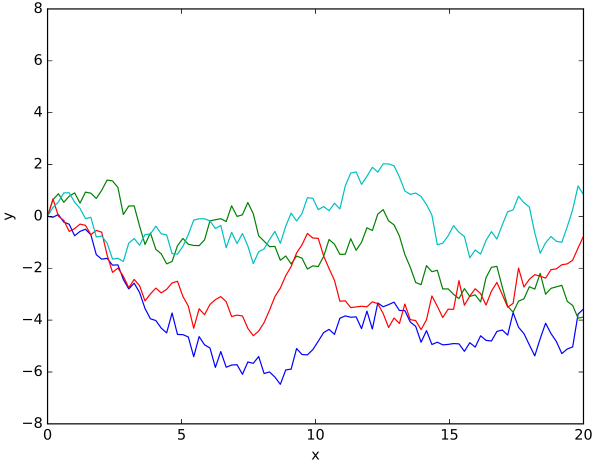
Gaussian Processes Examples

$$f(\mathbf{x}) \sim \mathcal{GP}(m(\mathbf{x}), k(\mathbf{x}.\mathbf{x}')), \quad m(\mathbf{x}) = \mathbf{0}, \quad k(\mathbf{x}.\mathbf{x}')) = \min(\mathbf{x}, \mathbf{x}')$$



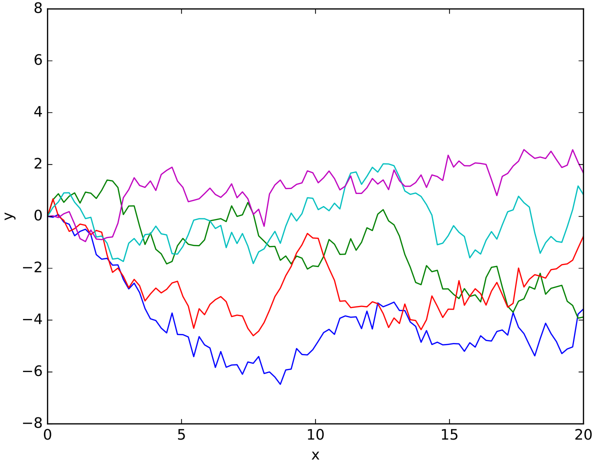
Gaussian Processes Examples

$$f(\mathbf{x}) \sim \mathcal{GP}(m(\mathbf{x}), k(\mathbf{x}.\mathbf{x}')), \quad m(\mathbf{x}) = \mathbf{0}, \quad k(\mathbf{x}.\mathbf{x}')) = \min(\mathbf{x}, \mathbf{x}')$$



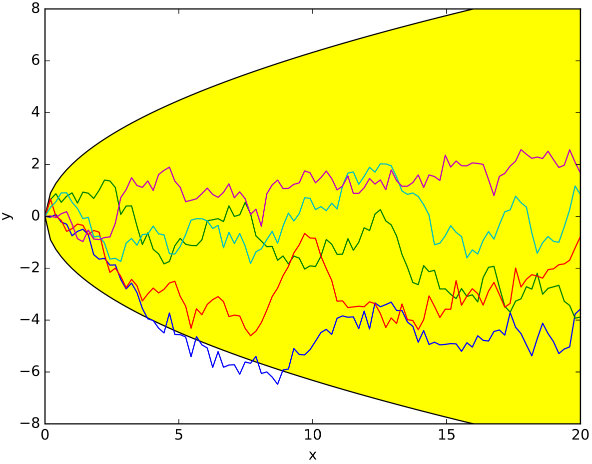
Gaussian Processes Examples

$$f(\mathbf{x}) \sim \mathcal{GP}(m(\mathbf{x}), k(\mathbf{x}.\mathbf{x}')), \quad m(\mathbf{x}) = \mathbf{0}, \quad k(\mathbf{x}.\mathbf{x}')) = \min(\mathbf{x}, \mathbf{x}')$$



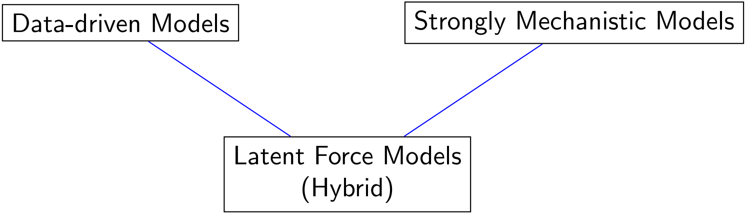
Gaussian Processes Examples

$$f(\mathbf{x}) \sim \mathcal{GP}(m(\mathbf{x}), k(\mathbf{x}.\mathbf{x}')), \quad m(\mathbf{x}) = \mathbf{0}, \quad k(\mathbf{x}.\mathbf{x}')) = \min(\mathbf{x}, \mathbf{x}')$$



Latent Force Models

The latent force model framework (LFM) was introduced in Álvarez et al. [Alvarez et al., 2009] motivated by the idea that for some phenomena a weak mechanistic assumption underlies a data-driven model.



.

Latent Force Models

The mechanistic assumptions are incorporated using dynamical systems governed by differential equations. Consider the following second order system

$$\frac{d^2 y_d(t)}{dt} + C_d \frac{dy_d(t)}{dt} + B_d y_d(t) = 0. \qquad (1)$$

C_d and B_d are known as the damper and spring coefficients respectively and $y_d(t)$ is the output of interest. The flexibility is achieved by adding a forcing term, which is governed by a set of Q latent functions $\{u_q(t)\}$ and the constants $\{S_{d,q}\}$ (sensitivities).

Latent Force Models

The mechanistic assumptions are incorporated using dynamical systems governed by differential equations. Consider the following second order system

$$\frac{d^2 y_d(t)}{dt} + C_d \frac{dy_d(t)}{dt} + B_d y_d(t) = \sum_{q=1}^{Q} S_{d,q} u_q(t). \tag{1}$$

C_d and B_d are known as the damper and spring coefficients respectively and $y_d(t)$ is the output of interest. The flexibility is achieved by adding a forcing term, which is governed by a set of Q latent functions $\{u_q(t)\}$ and the constants $\{S_{d,q}\}$ (sensitivities).

Latent Force Models

The main assumption in LFMs is that the latent functions $\{u_q(t)\}$ are modeled as GPs with Squared Exponential (SE) covariance function. As a consequence, it turns out that outputs are jointly governed by Gaussian Processes as well and their corresponding covariance functions can be derived analytically. Having $\omega_q = \sqrt{4B_d - C_d^2}/2$ and $\alpha_q = C_q/2$ the cross-covariance between outputs $y_p(t)$ and $y_q(t)$ is

$$k_{LFM}^{y_p,y_q}(t,t') = \sum_{r=1}^{Q} \frac{S_{r,p}S_{r,q}l_r\sqrt{\pi}}{8\omega_p\omega_q} k_{y_p,y_q}^{(r)}(t,t'). \quad (2)$$

See Álvarez et al. [Alvarez et al., 2009] for details.

Latent Force Models

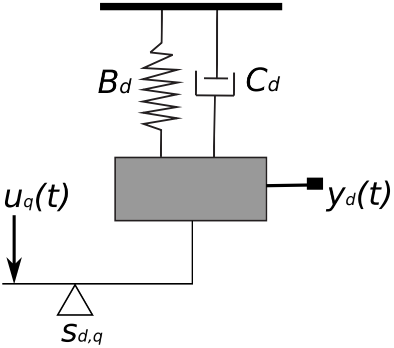
The main assumption in LFMs is that the latent functions $\{u_q(t)\}$ are modeled as GPs with Squared Exponential (SE) covariance function. As a consequence, it turns out that outputs are jointly governed by Gaussian Processes as well and their corresponding covariance functions can be derived analytically. Having $\omega_q = \sqrt{4B_d - C_d^2}/2$ and $\alpha_q = C_q/2$ the cross-covariance between outputs $y_p(t)$ and $y_q(t)$ is

$$k_{LFM}^{y_p,y_q}(t,t') = \sum_{r=1}^{Q} \frac{S_{r,p}S_{r,q}l_r\sqrt{\pi}}{8\omega_p\omega_q} k_{y_p,y_q}^{(r)}(t,t'). \quad (2)$$

See Álvarez et al. [Alvarez et al., 2009] for details.

Latent Force Models

- Main assumption: The latent functions $\{u_q(t)\}$ are modeled as GPs with Squared Exponential (SE) kernel.
- Consequence: Outputs are governed by GPs as well and their kernel functions can be derived analytically [Alvarez et al., 2009].

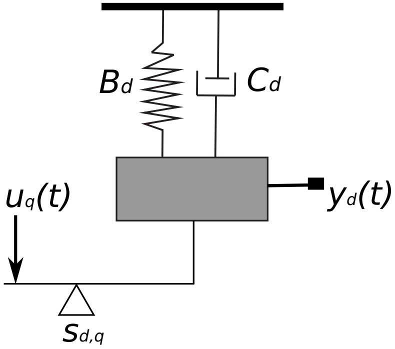


$$u_q(t) \sim \mathcal{GP}(\mathbf{0}, k_{SE}(t, t')).$$

$$y_d(t) \sim \mathcal{GP}(\mathbf{0}, k_{LFM}^{y_d, y_d}(t, t')).$$

Latent Force Models

- Main assumption: The latent functions $\{u_q(t)\}$ are modeled as GPs with Squared Exponential (SE) kernel.
- Consequence: Outputs are governed by GPs as well and their kernel functions can be derived analytically [Alvarez et al., 2009].

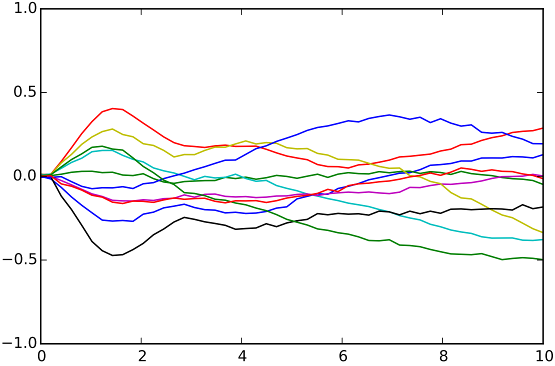


$$u_q(t) \sim \mathcal{GP}(\mathbf{0}, k_{SE}(t, t')).$$

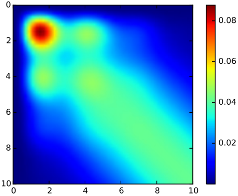
$$y_d(t) \sim \mathcal{GP}(\mathbf{0}, k_{LFM}^{y_d, y_d}(t, t')).$$

Latent Force Models Realizations

Underdamped trajectories $\{C_d = 1, B_d = 5\}$.



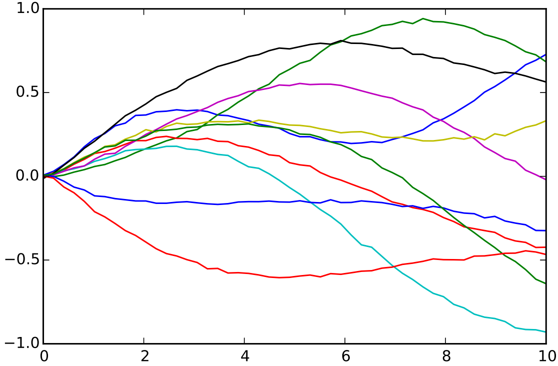
(a) Sampled trajectories



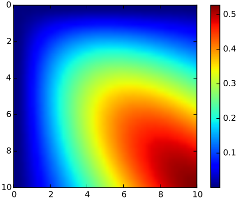
(b) Covariance matrix

Latent Force Models Realizations

Overdamped trajectories $\{C_d = 5, B_d = 1\}$.



(c) Sampled trajectories



(d) Covariance matrix

Hidden Markov Models

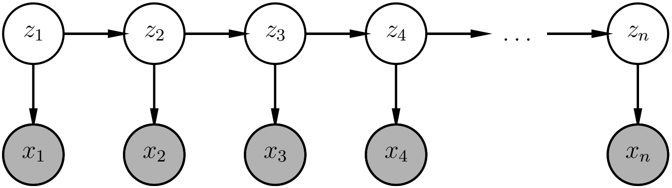


Figure 1: Probabilistic Graphical Model for HMM.

$$p(\mathbf{X}, \mathbf{Z} | \boldsymbol{\pi}, \mathbf{A}, \boldsymbol{\Theta}) = p(z_1 | \boldsymbol{\pi}) p(x_1 | z_1, \boldsymbol{\Theta}) \prod_{i=2}^{n} p(z_i | z_{i-1}, \mathbf{A}) p(x_i | z_i, \boldsymbol{\Theta}). \quad (3)$$

Where $\mathbf{A}$ is the hidden state transition matrix, $\boldsymbol{\pi}$ is the initial state probability mass function and $\boldsymbol{\Theta}$ are the emission parameters.

Hidden Markov Models

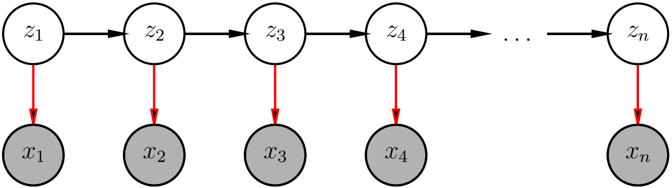


Figure 1: Probabilistic Graphical Model for HMM.

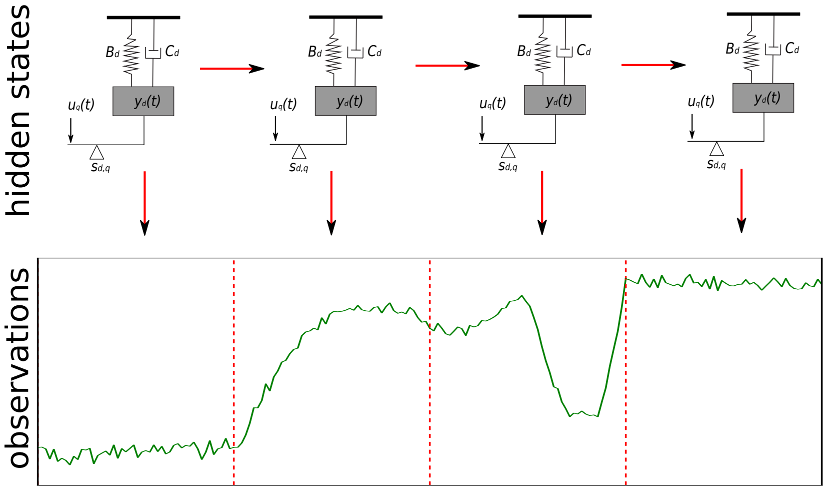
$$p(\mathbf{X}, \mathbf{Z}|\boldsymbol{\pi}, \mathbf{A}, \boldsymbol{\Theta}) = p(z_1|\boldsymbol{\pi})p(x_1|z_1, \boldsymbol{\Theta}) \prod_{i=2}^{n} p(z_i|z_{i-1}, \mathbf{A})p(x_i|z_i, \boldsymbol{\Theta}). \quad (3)$$

Where $\mathbf{A}$ is the hidden state transition matrix, $\boldsymbol{\pi}$ is the initial state probability mass function and $\boldsymbol{\Theta}$ are the emission parameters.

Hidden Markov Models

The use of novel probabilistic models as emission process broadens the horizon of potential applications and brings new challenges from the perspective of estimation.

Proposed Approach: HMM + LFM



- Motor primitive representation: Latent Force Models (LFM).
- Motor primitives sequential dynamics: Hidden Markov Models (HMM).

Proposed Approach: HMM + LFM

The overall model is a HMM where the emission distribution for each hidden state is represented by a LFM. Formally,

$$p(\mathbf{x}_i | z_i, \boldsymbol{\Theta}, \boldsymbol{\chi}) = \mathcal{N}(f_i(\boldsymbol{\chi}), I\sigma^2), \tag{4}$$

$$f_i(x) \sim \mathcal{GP}(0, k_{LFM}(x, x'; \boldsymbol{\Theta}_{z_i})).$$

Where k_{LFM} represents the second order LFM kernel and σ^2 is the variance of the i.i.d Gaussian noise. $\boldsymbol{\chi} = \{\chi_1, \chi_2, \ldots, \chi_s\}$ denotes the locations where the LFMs are sample at. Therefore, a movement sequence is segmented into equal-length chunks of size $|\boldsymbol{\chi}| = s$, each of which is modeled by a particular LFM.

Advantages of HMM + LFM

- Movement realizations have some discrete and non-smooth changes on the forces and dynamics which govern the movements.
- Using a limited set of hidden primitives allows to capture the existing diversity and redundancy in the dynamics over a movement trajectory.
- Scalability. The sizes of the covariance matrices belonging to the different hidden states (LFMs) are kept fixed regardless of the observed trajectory length. In contrast with the switched dynamical LFM (SDLFM) [Alvarez et al., 2010] where the covariance matrix grows quadratically with the observation sequence length.

Experimental Results: Synthetic Data

Instance of the model used to generate synthetic trajectories.

$$\mathbf{A} = \begin{bmatrix} 0.8 & 0.1 & 0.1 \\ 0.6 & 0.3 & 0.1 \\ 0.3 & 0.2 & 0.5 \end{bmatrix}, \boldsymbol{\pi} = \begin{bmatrix} 0.1 \\ 0.3 \\ 0.6 \end{bmatrix}, \mathbf{A}^* = \begin{bmatrix} 0.83 & 0.08 & 0.09 \\ 0.63 & 0.27 & 0.1 \\ 0.27 & 0.25 & 0.48 \end{bmatrix}, \boldsymbol{\pi}^* = \begin{bmatrix} 0.0 \\ 0.4 \\ 0.6 \end{bmatrix}.$$

	Hidden State 1	Hidden State 2	Hidden State 3
Spring const.	{3., 1, 2.5, 10.}	{1., 3.5, 9.0, 5.0}	{5., 8., 4.5, 1.}
Damper const.	{1., 3., 7.5, 10.}	{3., 10., 0.5, 0.1}	{6., 5., 4., 9.}
Lengthscale	10	2	5
Spring const.*	{3.21, 1.05, 2.67, 11.09}	{0.67, 1.72, 3.39, 2.26}	{6.92, 10.66, 6.32, 2.3}
Damper const.*	{1.2, 3.36, 8.39, 9.61}	{0.5, 2.62, 1.07, 0.27}	{6.09, 5.54, 4.47, 9.76}
Lengthscale*	85.77	180.48	159.96

Table 1: LFM emission parameters.

As emission process we use a LFM with 4 outputs ($D = 4$), a single latent force ($Q = 1$) and sample locations set $\chi = \{0.1, \ldots, 5.1\}$ with $|\chi| = 20$.

Experimental Results: Synthetic Data

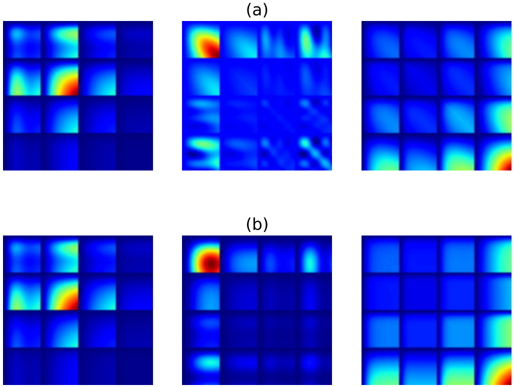


Figure 2: Actual and inferred toy experiment covariance matrices for the 3 hidden states. Top row **(a)**: LFM covariance matrices used for generating the synthetic trajectories. Bottom row**(b)**: Estimated LFM covariance matrices.

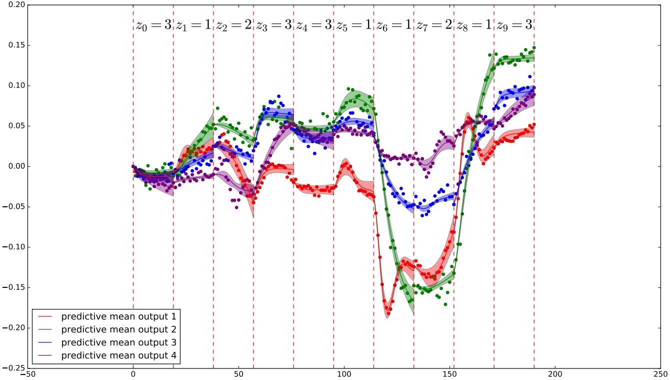


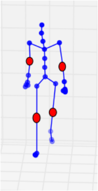
Figure 3: Primitives identification over a synthetic trajectory. In the top, the most probable hidden state sequence $\{z_0, z_1, \ldots, z_9\}$ is shown. The predictive mean after conditioning over the hidden state sequence and observations is also depicted with error bars.

Experimental Results: Synthetic Data

The Viterbi algorithm was executed for each validation trajectory (10 out of 20 trajectories) and the resulting hidden state sequences were compared against the actual values used during the data-set generation. The correct hidden state was recovered with a success rate of 95% failing only in 10 out of 200 validation segments.

Experimental Results: Real Data

- The walking behavior of the CMU motion capture database (CMU-MOCAP)[3] is used because it exhibits a rich redundancy over dynamics as a consequence of the cyclic nature of gait.
- Specifically, the subject No. 7 was used with trials $\{01, 02, 03, 06, 07, 08, 09\}$ for training and trials $\{10, 11\}$ for validation.
- The chosen joints are both elbows and both knees since they are relevant for the walking behavior and their potential correlations might be exploited by multiple-output LFMs.



[3] The data used in this work was obtained from mocap.cs.cmu.edu.

Experimental Results: Real Data

Fixing some model parameters:

- Number of hidden states: 3.
- Number of latent forces: 3.
- The sample locations set χ was defined to cover the interval $[0.1, 5.1]$ with 20 sample locations equally spaced (i.e. $|\chi| = 20$). This is motivated by the functional division of a gait cycle into eight phases [Perry et al., 1992].

Experimental Results: Real Data

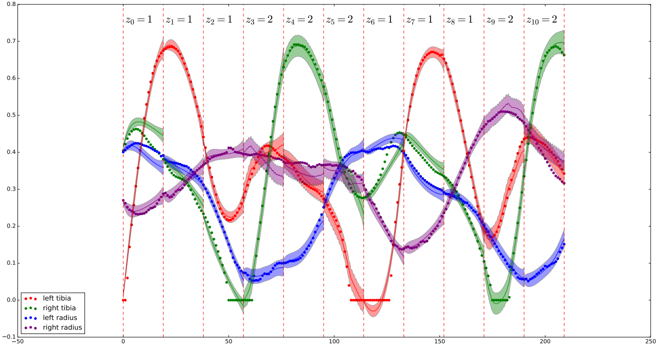


Figure 4: Primitives identification over a real walking trajectory. In the top the most probable hidden state sequence $\{z_0, z_1, \ldots, z_{10}\}$ given by the inferred model is shown. The predictive mean with error bars is depicted for the four joints.

Experimental Results: Real Data

Trial Number	Motor primitives identified
01	1, 1, 1, 2, 2, 2, 1, 1, 1, 1, 2, 2, 2, 1, 1, 1
02	1, 1, 1, 2, 2, 2, 1, 1, 1, 1, 2, 2, 2, 1, 1, 1
03	1, 1, 1, 2, 2, 2, 2, 1, 1, 1, 2, 2, 2, 2, 2, 1, 1, 1, 2, 2, 2
06	1, 1, 1, 2, 2, 2, 2, 1, 1, 1, 2, 2, 2, 2, 1, 1
07	1, 1, 1, 2, 2, 2, 2, 1, 1, 1, 2, 2, 2, 2, 2, 1, 1, 1, 2
08	1, 1, 1, 2, 2, 2, 2, 2, 1, 1, 1, 1, 2
09	1, 1, 1, 2, 1, 2, 1, 1, 1, 2, 2
10*	1, 1, 1, 2, 2, 2, 1, 1, 1, 2, 2
11*	1, 1, 1, 2, 2, 2, 2, 1, 1, 1, 2, 2, 2, 2, 2, 1, 1, 1, 2

Table 2: Motor primitives identified over walking realizations.

Conclusions and Future Work

- In this work we propose a novel probabilistic parameterization of motor primitives and their sequential composition relying on LFMs and HMMs.
- We showed how to estimate the model's parameters using the EM algorithm and, through synthetic and real data experiments, the model's capability to identify the occurrence of motor primitives after a training stage was successfully validated.
- Alternative formulations which support variable length segments are suggested to increase the model's flexibility. In particular, Hidden Semi-Markov Models (HSMM) are a natural extension.
- A further step in the validation might involve using it as a probabilistic generative model in a classification task as in the case of identifying pathological walking.

Bibliography

[Alvarez et al., 2010] Alvarez, M., Peters, J. R., Lawrence, N. D., and Schölkopf, B. (2010).
Switched latent force models for movement segmentation.
In *Advances in neural information processing systems*, pages 55–63.

[Alvarez et al., 2009] Alvarez, M. A., Luengo, D., and Lawrence, N. D. (2009).
Latent force models.
In *International Conference on Artificial Intelligence and Statistics*, pages 9–16.

[Flash and Hochner, 2005] Flash, T. and Hochner, B. (2005).
Motor primitives in vertebrates and invertebrates.
Current opinion in neurobiology, 15(6):660–666.

[Perry et al., 1992] Perry, J., Davids, J. R., et al. (1992).
Gait analysis: Normal and pathological function.
Journal of Pediatric Orthopaedics, 12(6):815.

[Rasmussen, 2006] Rasmussen, C. E. (2006).
Gaussian processes for machine learning.

[Schaal, 1999] Schaal, S. (1999).
Is imitation learning the route to humanoid robots?
Trends in cognitive sciences, 3(6):233–242.