Latent Force Models: Bridging the Divide between Mechanistic and Data Modelling Paradigms

Neil D. Lawrence
University of Sheffield

Max Planck Institute for Intelligent Systems, Stuttgart

21st July 2015

Outline

Machine Learning

Gaussian Processes

Motion Capture Example

ODE Model of Transcriptional Regulation

Discussion and Future Work

Outline

Machine Learning

Gaussian Processes

Motion Capture Example

ODE Model of Transcriptional Regulation

Discussion and Future Work

What is Machine Learning?

data

- data: observations, could be actively or passively acquired (meta-data).

What is Machine Learning?

data +

- data: observations, could be actively or passively acquired (meta-data).

What is Machine Learning?

data + model

- data: observations, could be actively or passively acquired (meta-data).
- model: assumptions, based on previous experience (other data! transfer learning etc), or beliefs about the regularities of the universe. Inductive bias.

What is Machine Learning?

data + model =

- data: observations, could be actively or passively acquired (meta-data).
- model: assumptions, based on previous experience (other data! transfer learning etc), or beliefs about the regularities of the universe. Inductive bias.

What is Machine Learning?

data + model = prediction

- data: observations, could be actively or passively acquired (meta-data).
- model: assumptions, based on previous experience (other data! transfer learning etc), or beliefs about the regularities of the universe. Inductive bias.
- prediction: an action to be taken or a categorization or a quality score.

Historical Perspective

- A data driven approach to Artificial Intelligence.
- Inspired by attempts to model the brain (the connectionists).
- A community that transcended traditional boundaries (psychology, statistical physics, signal processing)
- Led to an approach that dominates in the modern data-rich world.

Two Dominant Approaches

- Machine Learning as Optimization:

Two Dominant Approaches

- Machine Learning as Optimization:
    - Formulate your learning Problem as an optimization problem.

Two Dominant Approaches

- Machine Learning as Optimization:
  - Formulate your learning Problem as an optimization problem.
  - Typically intractable, so minimize a *relaxed* version of the cost function.

Two Dominant Approaches

- Machine Learning as Optimization:
  - Formulate your learning Problem as an optimization problem.
  - Typically intractable, so minimize a *relaxed* version of the cost function.
  - Prove characteristics of the resulting solution.

Two Dominant Approaches

- Machine Learning as Optimization:
  - Formulate your learning Problem as an optimization problem.
  - Typically intractable, so minimize a *relaxed* version of the cost function.
  - Prove characteristics of the resulting solution.
- Machine Learning as Probabilistic Modelling:

Two Dominant Approaches

- Machine Learning as Optimization:
  - Formulate your learning Problem as an optimization problem.
  - Typically intractable, so minimize a *relaxed* version of the cost function.
  - Prove characteristics of the resulting solution.
- Machine Learning as Probabilistic Modelling:
  - Formulate your learning problem as a probabilistic model.

Two Dominant Approaches

- Machine Learning as Optimization:
  - Formulate your learning Problem as an optimization problem.
  - Typically intractable, so minimize a *relaxed* version of the cost function.
  - Prove characteristics of the resulting solution.
- Machine Learning as Probabilistic Modelling:
  - Formulate your learning problem as a probabilistic model.
  - Relate variables through probability distributions.

Two Dominant Approaches

- Machine Learning as Optimization:
  - Formulate your learning Problem as an optimization problem.
  - Typically intractable, so minimize a *relaxed* version of the cost function.
  - Prove characteristics of the resulting solution.
- Machine Learning as Probabilistic Modelling:
  - Formulate your learning problem as a probabilistic model.
  - Relate variables through probability distributions.
  - If *Bayesian*, treat parameters with probability distributions.

Two Dominant Approaches

- Machine Learning as Optimization:
  - Formulate your learning Problem as an optimization problem.
  - Typically intractable, so minimize a *relaxed* version of the cost function.
  - Prove characteristics of the resulting solution.
- Machine Learning as Probabilistic Modelling:
  - Formulate your learning problem as a probabilistic model.
  - Relate variables through probability distributions.
  - If *Bayesian*, treat parameters with probability distributions.
  - Required integrals often intractable: use approximations (MCMC, variational etc).

Modelling Assumptions

- Modelling assumptions are either included as:

Modelling Assumptions

- Modelling assumptions are either included as:
  - a regularizer (optimization) or

Modelling Assumptions

- Modelling assumptions are either included as:
  - a regularizer (optimization) or
  - in the probability distribution (probabilistic approach).

Modelling Assumptions

- Modelling assumptions are either included as:
  - a regularizer (optimization) or
  - in the probability distribution (probabilistic approach).
- Typical assumptions: sparsity, smoothness.

Styles of Machine Learning

Background: interpolation is easy, extrapolation is hard

- Urs Hölzle keynote talk at NIPS 2005.
  - Emphasis on massive data sets.
  - Let the data do the work—more data, less extrapolation.
- Alternative paradigm:
  - Very scarce data: computational biology, human motion.
  - How to generalize from scarce data?
  - Need to include more assumptions about the data (e.g. invariances).

General Approach

Broadly Speaking: Two approaches to modeling

data modeling

mechanistic modeling

General Approach

Broadly Speaking: Two approaches to modeling

data modeling	*mechanistic modeling*
let the data "speak"	

General Approach

Broadly Speaking: Two approaches to modeling

data modeling	*mechanistic modeling*
let the data "speak"	impose physical laws

General Approach

Broadly Speaking: Two approaches to modeling

data modeling	*mechanistic modeling*
let the data “speak”	impose physical laws
data driven	

General Approach

Broadly Speaking: Two approaches to modeling

data modeling	*mechanistic modeling*
let the data “speak”	impose physical laws
data driven	knowledge driven

General Approach

Broadly Speaking: Two approaches to modeling

data modeling	*mechanistic modeling*
let the data "speak"	impose physical laws
data driven	knowledge driven
adaptive models	

General Approach

Broadly Speaking: Two approaches to modeling

data modeling	*mechanistic modeling*
let the data “speak”	impose physical laws
data driven	knowledge driven
adaptive models	differential equations

General Approach

Broadly Speaking: Two approaches to modeling

data modeling	*mechanistic modeling*
let the data “speak”	impose physical laws
data driven	knowledge driven
adaptive models	differential equations
digit recognition	

General Approach

Broadly Speaking: Two approaches to modeling

data modeling	*mechanistic modeling*
let the data “speak”	impose physical laws
data driven	knowledge driven
adaptive models	differential equations
digit recognition	climate, weather models

General Approach

Broadly Speaking: Two approaches to modeling

data modeling	*mechanistic modeling*
let the data "speak"	impose physical laws
data driven	knowledge driven
adaptive models	differential equations
digit recognition	climate, weather models

Weakly Mechanistic

General Approach

Broadly Speaking: Two approaches to modeling

data modeling	*mechanistic modeling*
let the data "speak"	impose physical laws
data driven	knowledge driven
adaptive models	differential equations
digit recognition	climate, weather models

Weakly Mechanistic

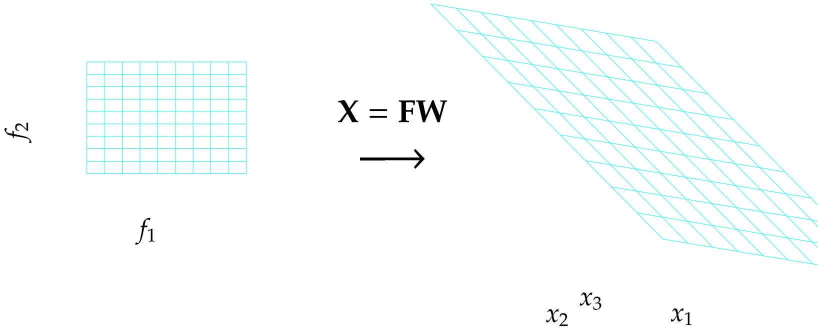
Strongly Mechanistic

Weakly Mechanistic vs Strongly Mechanistic

- Underlying data modeling techniques there are *weakly mechanistic* principles (e.g. smoothness).
- In physics the models are typically *strongly mechanistic*.
- In principle we expect a range of models which vary in the strength of their mechanistic assumptions.
- Latent Force Models are one part of this spectrum: add further mechanistic ideas to weakly mechanistic models.

Linear Dimensionality Reduction

- Find a lower dimensional plane embedded in a higher dimensional space.
- The plane is described by the matrix $\mathbf{W} \in \Re^{p \times q}$.



Dimensionality Reduction

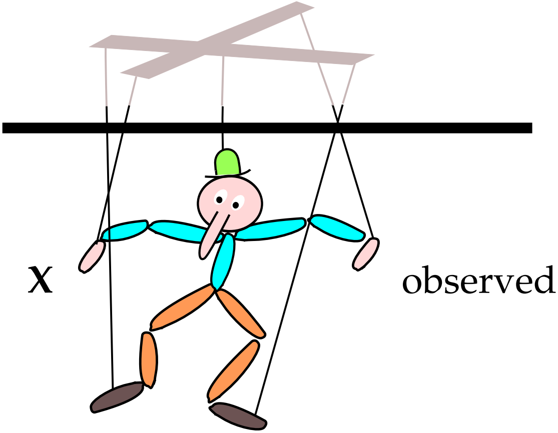
- Linear relationship between the data, **X**, and a reduced dimensional representation, **F**.

$$\mathbf{X} = \mathbf{F}\mathbf{W} + \boldsymbol{\epsilon},$$

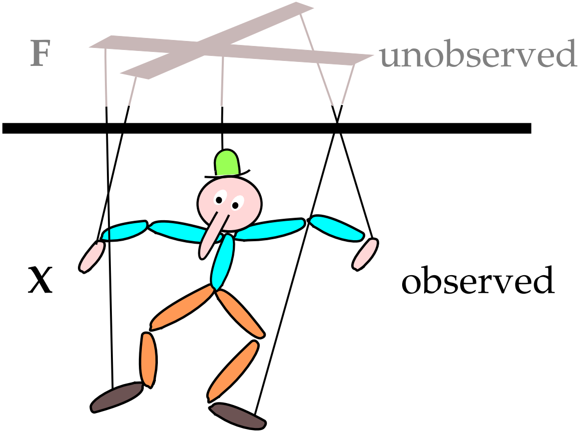
$$\boldsymbol{\epsilon} \sim \mathcal{N}\left(\mathbf{0}, \boldsymbol{\Sigma}\right)$$

- Problem is we don't know what **F** should be!

Marionette Analogy



Marionette Analogy



$\mathbf{F}$ is a Latent Variable

- Define a *probability distribution* for $\mathbf{F}$.
- Marginalize out $\mathbf{F}$ (integrate over).
- Optimize with respect to $\mathbf{W}$.
- For Gaussian distribution, $\mathbf{F} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
    - and $\Sigma = \sigma^2\mathbf{I}$ we have probabilistic PCA (Tipping and Bishop, 1999; Roweis, 1998).
    - and Σ constrained to be diagonal, we have factor analysis.

Dimensionality Reduction: Temporal Data

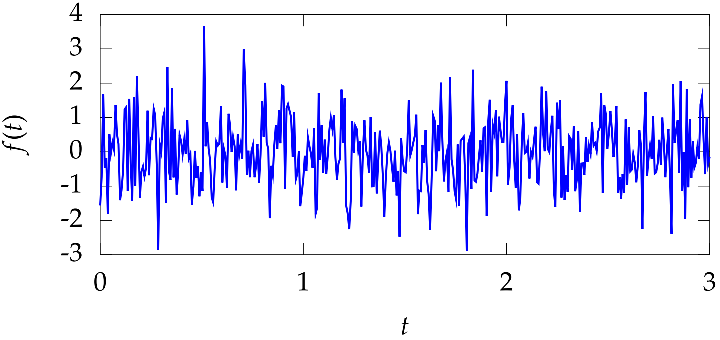


Figure: PCA: Pure sampling from a Gaussian does not retain temporal effects.

Dimensionality Reduction: Temporal Data

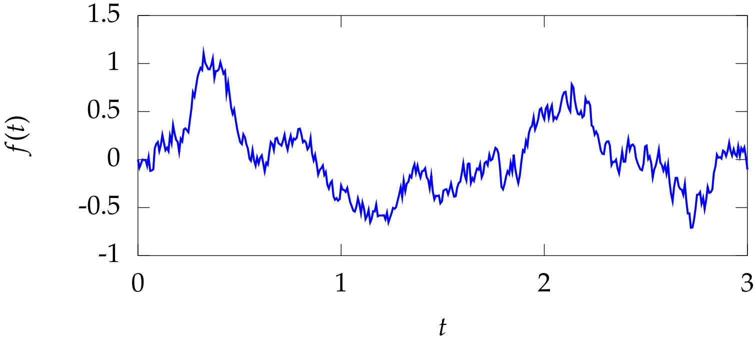


Figure: Kalman filter (Rauch-Tung-Striebel smoother) is Markov-Gaussian (non smooth).

Dimensionality Reduction: Temporal Data

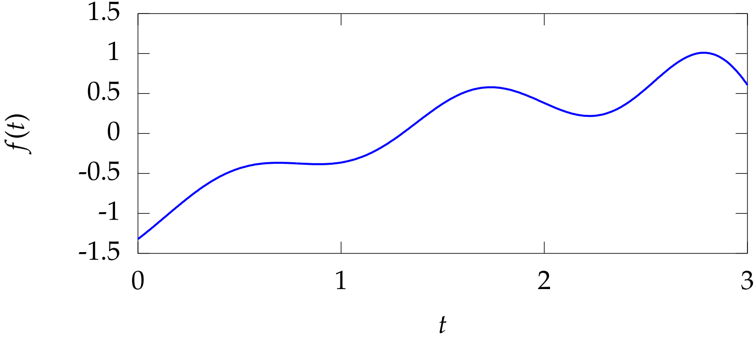


Figure: General Gaussian processes allow for priors over *smooth* functions.

Outline

Machine Learning

Gaussian Processes

Motion Capture Example

ODE Model of Transcriptional Regulation

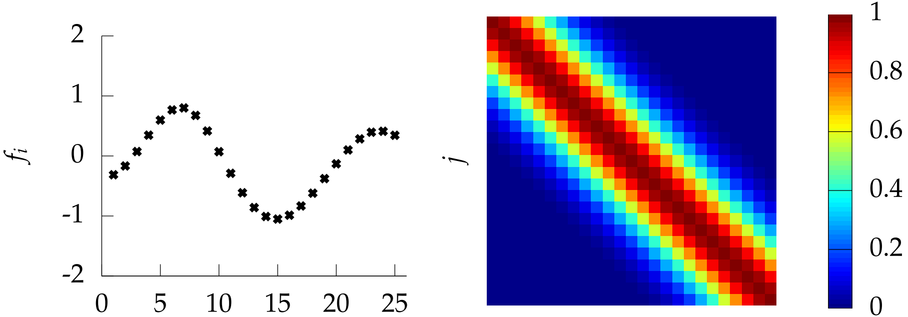
Discussion and Future Work

Sampling a Function

Multi-variate Gaussians

- We will consider a Gaussian with a particular structure of covariance matrix.
- Generate a single sample from this 25 dimensional Gaussian distribution, $\mathbf{f} = [f_1, f_2 \ldots f_{25}]$.
- We will plot these points against their index.

Gaussian Distribution Sample

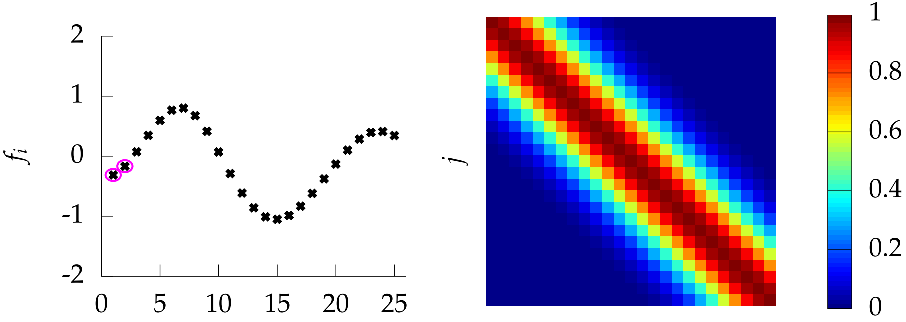


(a) A 25 dimensional correlated random variable (values ploted against index)

(b) colormap showing correlations between dimensions.

Figure: A sample from a 25 dimensional Gaussian distribution.

Gaussian Distribution Sample

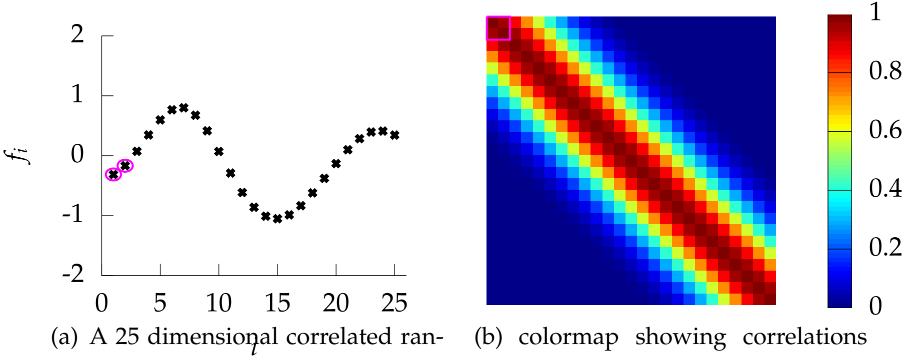


(a) A 25 dimensional correlated random variable (values ploted against index)

(b) colormap showing correlations between dimensions.

Figure: A sample from a 25 dimensional Gaussian distribution.

Gaussian Distribution Sample

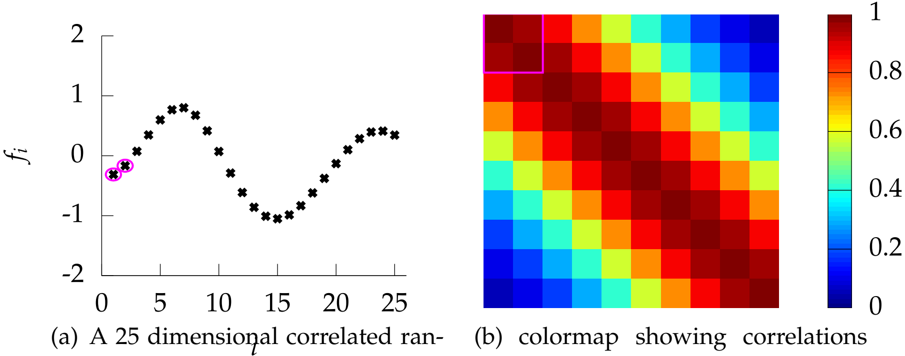


(a) A 25 dimensional correlated random variable (values ploted against index)

(b) colormap showing correlations between dimensions.

Figure: A sample from a 25 dimensional Gaussian distribution.

Gaussian Distribution Sample

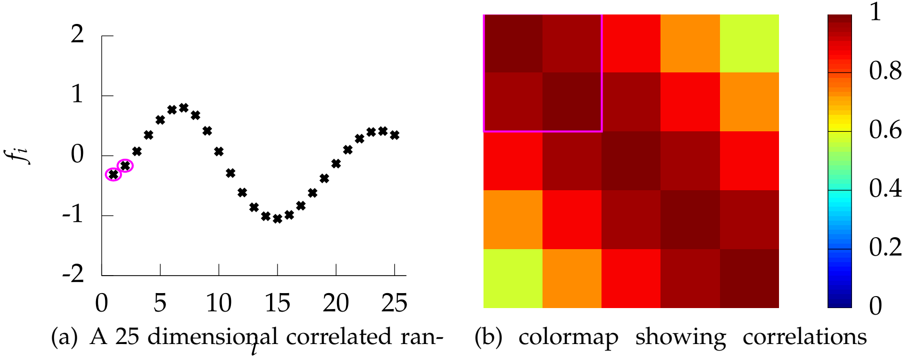


(a) A 25 dimensional correlated random variable (values ploted against index)

(b) colormap showing correlations between dimensions.

Figure: A sample from a 25 dimensional Gaussian distribution.

Gaussian Distribution Sample

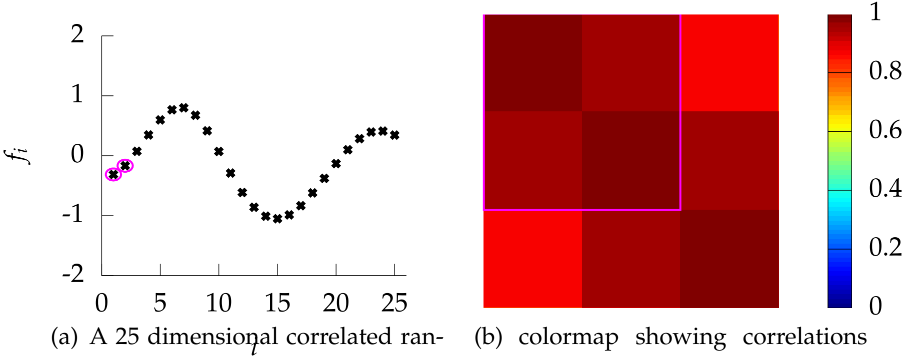


(a) A 25 dimensional correlated random variable (values ploted against index)

(b) colormap showing correlations between dimensions.

Figure: A sample from a 25 dimensional Gaussian distribution.

Gaussian Distribution Sample

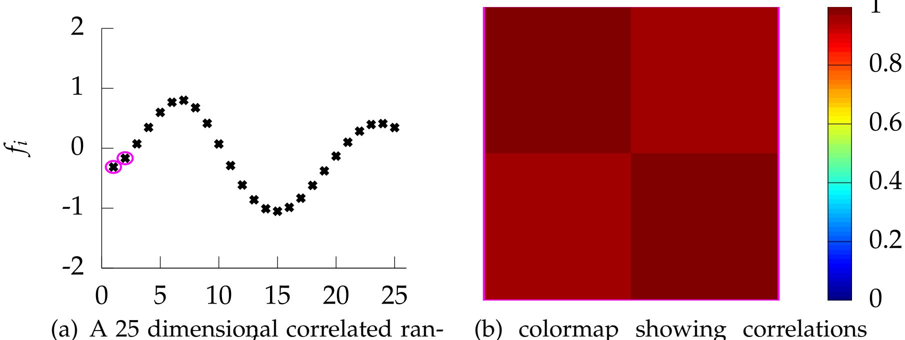


(a) A 25 dimensional correlated random variable (values ploted against index)

(b) colormap showing correlations between dimensions.

Figure: A sample from a 25 dimensional Gaussian distribution.

Gaussian Distribution Sample

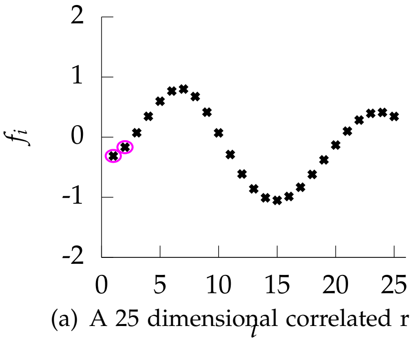


(a) A 25 dimensional correlated random variable (values ploted against index)

(b) colormap showing correlations between dimensions.

Figure: A sample from a 25 dimensional Gaussian distribution.

Gaussian Distribution Sample



(a) A 25 dimensional correlated random variable (values ploted against index)

$$\begin{bmatrix} 1 & 0.96587 \\ 0.96587 & 1 \end{bmatrix}$$

(b) correlation between f_1 and f_2.

Figure: A sample from a 25 dimensional Gaussian distribution.

Covariance Function

The covariance matrix

- Covariance matrix shows correlation between points f_i and f_j if i is near to j.

Covariance Function

The covariance matrix

- Covariance matrix shows correlation between points f_i and f_j if i is near to j.
- Less correlation if i is distant from j.

Covariance Function

The covariance matrix

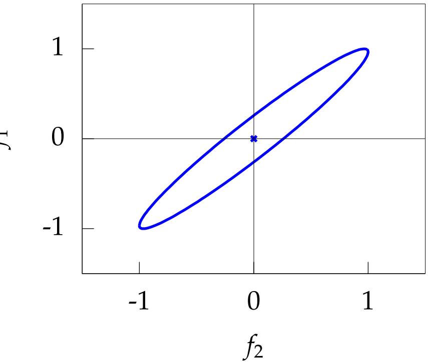
- Covariance matrix shows correlation between points f_i and f_j if i is near to j.
- Less correlation if i is distant from j.
- Our ordering of points means that the *function appears smooth*.

Covariance Function

The covariance matrix

- Covariance matrix shows correlation between points f_i and f_j if i is near to j.
- Less correlation if i is distant from j.
- Our ordering of points means that the *function appears smooth*.
- Let's focus on the joint distribution of two points from the 25.

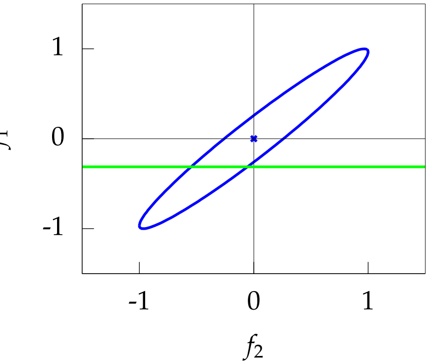
Prediction of f_2 from f_1



$$\begin{bmatrix} 1 & 0.96587 \\ 0.96587 & 1 \end{bmatrix}$$

- The single contour of the Gaussian density represents the joint distribution, $p(f_1, f_2)$.

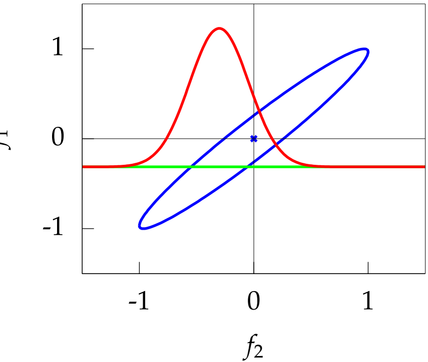
Prediction of f_2 from f_1



$$\begin{bmatrix} 1 & 0.96587 \\ 0.96587 & 1 \end{bmatrix}$$

- The single contour of the Gaussian density represents the joint distribution, $p(f_1, f_2)$.
- We observe that $f_1 = -0.313$.

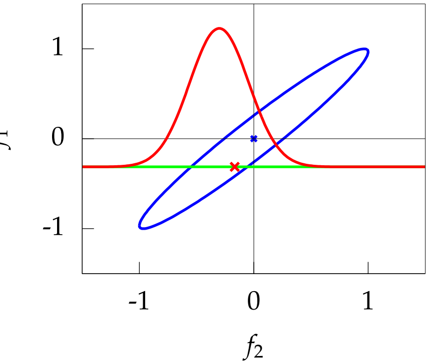
Prediction of f_2 from f_1



$$\begin{bmatrix} 1 & 0.96587 \\ 0.96587 & 1 \end{bmatrix}$$

- The single contour of the Gaussian density represents the joint distribution, $p(f_1, f_2)$.
- We observe that $f_1 = -0.313$.
- Conditional density: $p(f_2|f_1 = -0.313)$.

Prediction of f_2 from f_1



$$\begin{bmatrix} 1 & 0.96587 \\ 0.96587 & 1 \end{bmatrix}$$

- The single contour of the Gaussian density represents the joint distribution, $p(f_1, f_2)$.
- We observe that $f_1 = -0.313$.
- Conditional density: $p(f_2|f_1 = -0.313)$.

Prediction with Correlated Gaussians

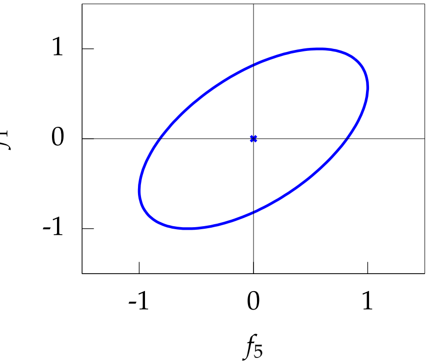
- Prediction of f_2 from f_1 requires *conditional density*.
- Conditional density is *also* Gaussian.

$$p(f_2|f_1) = \mathcal{N}\left(f_2|\frac{k_{1,2}}{k_{1,1}}f_1, k_{2,2} - \frac{k_{1,2}^2}{k_{1,1}}\right)$$

where covariance of joint density is given by

$$\mathbf{K} = \begin{bmatrix} k_{1,1} & k_{1,2} \\ k_{2,1} & k_{2,2} \end{bmatrix}$$

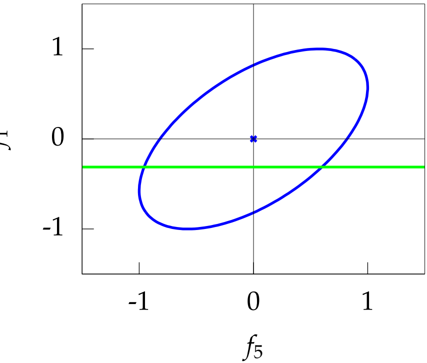
Prediction of f_5 from f_1



$$\begin{bmatrix} 1 & 0.57375 \\ 0.57375 & 1 \end{bmatrix}$$

- The single contour of the Gaussian density represents the joint distribution, $p(f_1, f_5)$.

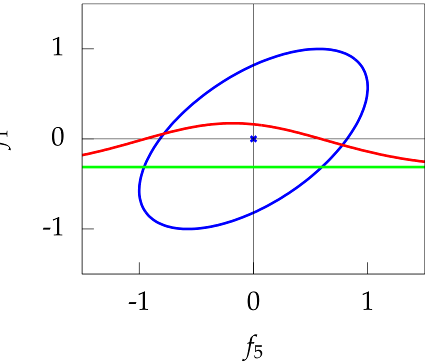
Prediction of f_5 from f_1



$$\begin{bmatrix} 1 & 0.57375 \\ 0.57375 & 1 \end{bmatrix}$$

- The single contour of the Gaussian density represents the joint distribution, $p(f_1, f_5)$.
- We observe that $f_1 = -0.313$.

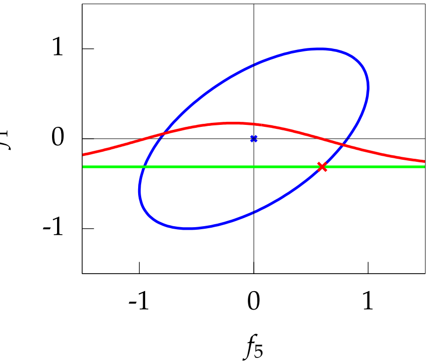
Prediction of f_5 from f_1



$$\begin{bmatrix} 1 & 0.57375 \\ 0.57375 & 1 \end{bmatrix}$$

- The single contour of the Gaussian density represents the joint distribution, $p(f_1, f_5)$.
- We observe that $f_1 = -0.313$.
- Conditional density: $p(f_5|f_1 = -0.313)$.

Prediction of f_5 from f_1



$$\begin{bmatrix} 1 & 0.57375 \\ 0.57375 & 1 \end{bmatrix}$$

- The single contour of the Gaussian density represents the joint distribution, $p(f_1, f_5)$.
- We observe that $f_1 = -0.313$.
- Conditional density: $p(f_5|f_1 = -0.313)$.

Prediction with Correlated Gaussians

- Prediction of $\mathbf{f}_*$ from $\mathbf{f}$ requires multivariate *conditional density*.
- Multivariate conditional density is *also* Gaussian.

$$p(\mathbf{f}_*|\mathbf{f}) = \mathcal{N}\left(\mathbf{f}_*|\mathbf{K}_{*,\mathbf{f}}\mathbf{K}_{\mathbf{f},\mathbf{f}}^{-1}\mathbf{f}, \mathbf{K}_{*,*} - \mathbf{K}_{*,\mathbf{f}}\mathbf{K}_{\mathbf{f},\mathbf{f}}^{-1}\mathbf{K}_{\mathbf{f},*}\right)$$

- Here covariance of joint density is given by

$$\mathbf{K} = \begin{bmatrix}\mathbf{K}_{\mathbf{f},\mathbf{f}} & \mathbf{K}_{*,\mathbf{f}}\\ \mathbf{K}_{\mathbf{f},*} & \mathbf{K}_{*,*}\end{bmatrix}$$

Prediction with Correlated Gaussians

- Prediction of $\mathbf{f}_*$ from $\mathbf{f}$ requires multivariate *conditional density*.
- Multivariate conditional density is *also* Gaussian.

$$p(\mathbf{f}_*|\mathbf{f}) = \mathcal{N}\left(\mathbf{f}_*|\boldsymbol{\mu}, \boldsymbol{\Sigma}\right)$$

$$\boldsymbol{\mu} = \mathbf{K}_{*,\mathbf{f}}\mathbf{K}_{\mathbf{f},\mathbf{f}}^{-1}\mathbf{f}$$

$$\boldsymbol{\Sigma} = \mathbf{K}_{*,*} - \mathbf{K}_{*,\mathbf{f}}\mathbf{K}_{\mathbf{f},\mathbf{f}}^{-1}\mathbf{K}_{\mathbf{f},*}$$

- Here covariance of joint density is given by

$$\mathbf{K} = \begin{bmatrix}\mathbf{K}_{\mathbf{f},\mathbf{f}} & \mathbf{K}_{*,\mathbf{f}}\\ \mathbf{K}_{\mathbf{f},*} & \mathbf{K}_{*,*}\end{bmatrix}$$

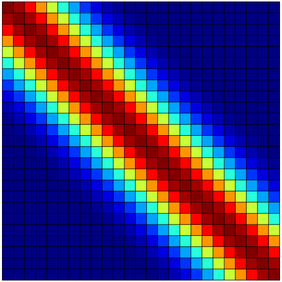
Covariance Functions

Where did this covariance matrix come from?

Exponentiated Quadratic Kernel Function (RBF, Squared Exponential, Gaussian)

$$k(t, t') = \alpha \exp\left(-\frac{\|t - t'\|_2^2}{2\ell^2}\right)$$

- Covariance matrix is built using the *inputs* to the function t.
- For the example above it was based on Euclidean distance.
- The covariance function is also know as a kernel.



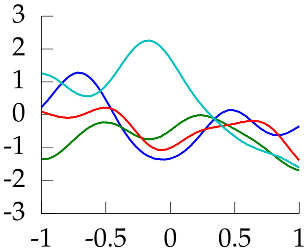
Covariance Functions

Where did this covariance matrix come from?

Exponentiated Quadratic Kernel Function (RBF, Squared Exponential, Gaussian)

$$k(t, t') = \alpha \exp\left(-\frac{\|t - t'\|_2^2}{2\ell^2}\right)$$

- Covariance matrix is built using the *inputs* to the function t.
- For the example above it was based on Euclidean distance.
- The covariance function is also know as a kernel.



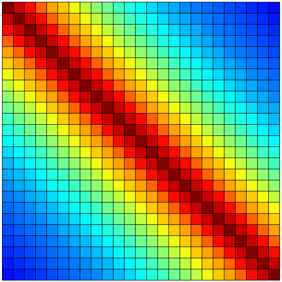
Covariance Functions

Where did this covariance matrix come from?

Ornstein-Uhlenbeck (stationary Gauss-Markov) covariance function

$$k(t, t') = \alpha \exp\left(-\frac{|t - t'|}{2\ell^2}\right)$$

- In one dimension arises from a stochastic differential equation. Brownian motion in a parabolic tube.
- In higher dimension a Fourier filter of the form $\frac{1}{\pi(1+x^2)}$.



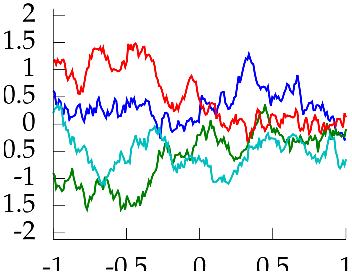
Covariance Functions

Where did this covariance matrix come from?

Ornstein-Uhlenbeck (stationary Gauss-Markov) covariance function

$$k\left(t, t'\right) = \alpha \exp\left(-\frac{\left|t - t'\right|}{2\ell^2}\right)$$

- In one dimension arises from a stochastic differential equation. Brownian motion in a parabolic tube.
- In higher dimension a Fourier filter of the form $\frac{1}{\pi(1+x^2)}$.



Mechanical Analogy

Back to Mechanistic Models!

- These models rely on the latent variables to provide the dynamic information.
- We now introduce a further dynamical system with a *mechanistic* inspiration.
- Physical Interpretation:
  - the latent functions, $f_i(t)$ are q forces.
  - We observe the displacement of p springs to the forces.,
  - Interpret system as the force balance equation, $\mathbf{XD} = \mathbf{FS} + \boldsymbol{\epsilon}$.
  - Forces act, e.g. through levers — a matrix of sensitivities, $\mathbf{S} \in \Re^{q \times p}$.
  - Diagonal matrix of spring constants, $\mathbf{D} \in \Re^{p \times p}$.
  - Original System: $\mathbf{W} = \mathbf{SD}^{-1}$.

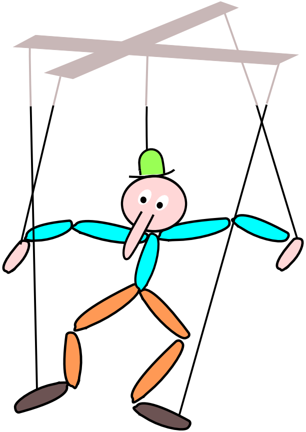
Extend Model

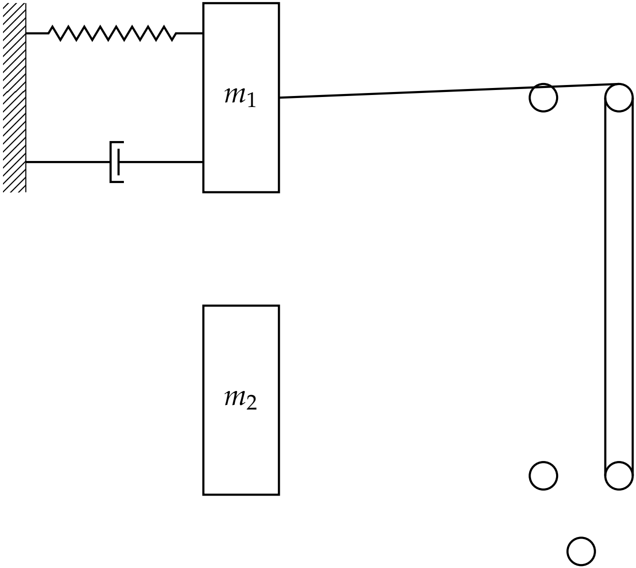
- Add a damper and give the system mass.

$$\mathbf{F}\mathbf{S} = \ddot{\mathbf{X}}\mathbf{M} + \dot{\mathbf{X}}\mathbf{C} + \mathbf{X}\mathbf{D} + \boldsymbol{\epsilon}.$$

- Now have a second order mechanical system.
- It will exhibit inertia and resonance.
- There are many systems that can also be represented by differential equations.
    - When being forced by latent function(s), $\{f_i(t)\}_{i=1}^{q}$, we call this a *latent force model*.

Marionette




m_1
m_2

Mass Spring Damper Analogy

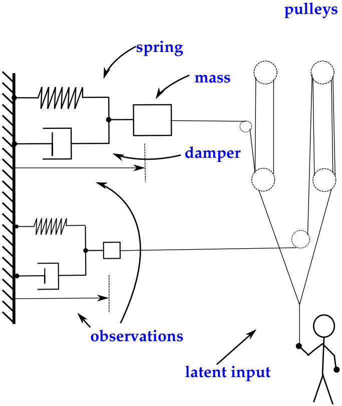


Figure: Mass spring damper analogy, an unobserved force drives multiple oscillators.

Mass Spring Damper Analogy

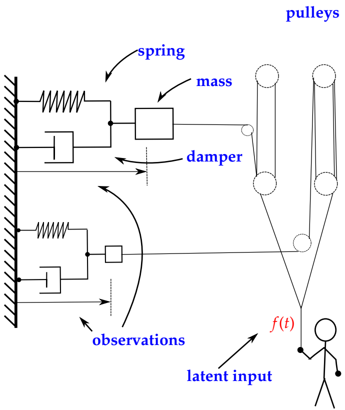


Figure: Mass spring damper analogy, an unobserved force drives multiple oscillators.

Mass Spring Damper Analogy

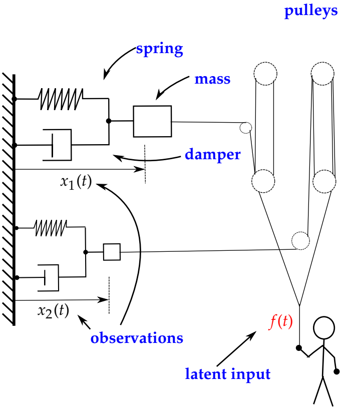


Figure: Mass spring damper analogy, an unobserved force drives multiple oscillators.

Mass Spring Damper Analogy

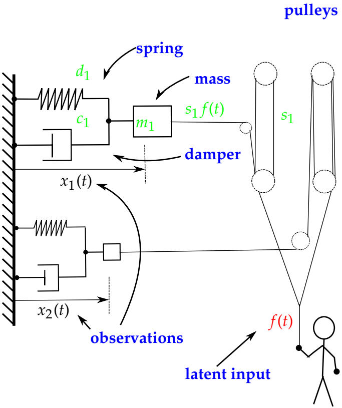


Figure: Mass spring damper analogy, an unobserved force drives multiple oscillators.

Mass Spring Damper Analogy

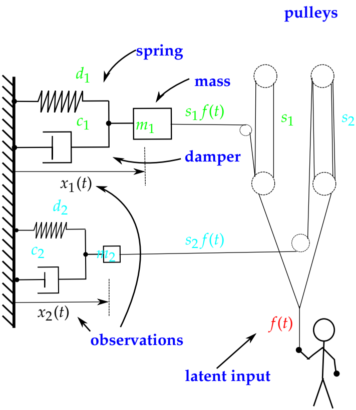


Figure: Mass spring damper analogy, an unobserved force drives multiple oscillators.

Gaussian Process priors and Latent Force Models

Driven Harmonic Oscillator

- For Gaussian process we can compute the covariance matrices for the output displacements.
- For one displacement the model is

$$m_k \ddot{x}_k(t) + c_k \dot{x}_k(t) + d_k x_k(t) = b_k + \sum_{i=0}^{q} s_{ik} f_i(t), \qquad (1)$$

  where, m_k is the kth diagonal element from $\mathbf{M}$ and similarly for c_k and d_k. s_{ik} is the i, kth element of $\mathbf{S}$.
- Model the latent forces as q independent, GPs with exponentiated quadratic covariances

$$k_{f_i f_l}(t, t') = \exp\left(-\frac{(t-t')^2}{2\ell_i^2}\right)\delta_{il}.$$

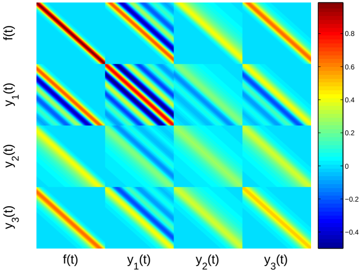
Covariance for ODE Model

- Exponentiated Quadratic Covariance function for $f(t)$

$$x_j(t) = \frac{1}{m_j\omega_j}\sum_{i=1}^{q} s_{ji}\exp(-\alpha_j t)\int_0^t f_i(\tau)\exp(\alpha_j\tau)\sin(\omega_j(t-\tau))\mathrm{d}\tau$$

- Joint distribution for $x_1(t)$, $x_2(t)$, $x_3(t)$ and $f(t)$. Damping ratios:

ζ_1	ζ_2	ζ_3
0.125	2	1



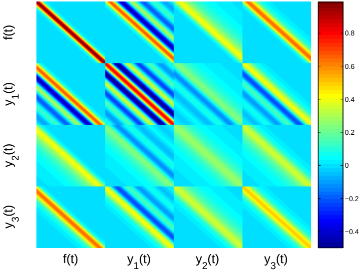
Covariance for ODE Model

- Analogy

$$x = \sum_i \mathbf{e}_i^\top \mathbf{f}_i \quad \mathbf{f}_i \sim \mathcal{N}(\mathbf{0}, \Sigma_i) \rightarrow x \sim \mathcal{N}\left(0, \sum_i \mathbf{e}_i^\top \Sigma_i \mathbf{e}_i\right)$$

- Joint distribution for $x_1(t)$, $x_2(t)$, $x_3(t)$ and $f(t)$. Damping ratios:

ζ_1	ζ_2	ζ_3
0.125	2	1



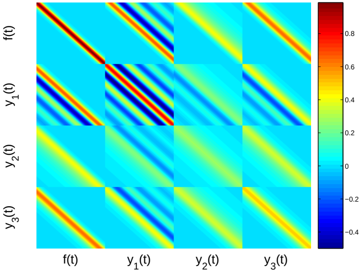
Covariance for ODE Model

- Exponentiated Quadratic Covariance function for $f(t)$

$$x_j(t) = \frac{1}{m_j\omega_j}\sum_{i=1}^{q} s_{ji}\exp(-\alpha_j t)\int_0^t f_i(\tau)\exp(\alpha_j\tau)\sin(\omega_j(t-\tau))\mathrm{d}\tau$$

- Joint distribution for $x_1(t)$, $x_2(t)$, $x_3(t)$ and $f(t)$. Damping ratios:

ζ_1	ζ_2	ζ_3
0.125	2	1



Joint Sampling of $x(t)$ and $f(t)$

- `lfmSample`

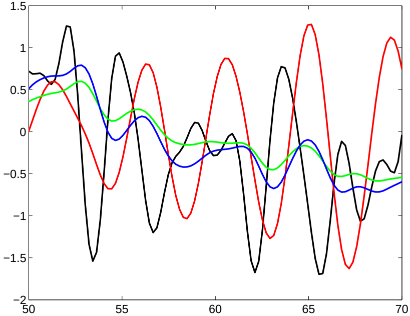


Figure: Joint samples from the ODE covariance, *black*: $f(t)$, *red*: $x_1(t)$ (underdamped), *green*: $x_2(t)$ (overdamped), and *blue*: $x_3(t)$ (critically damped).

Joint Sampling of $x(t)$ and $f(t)$

- `lfmSample`

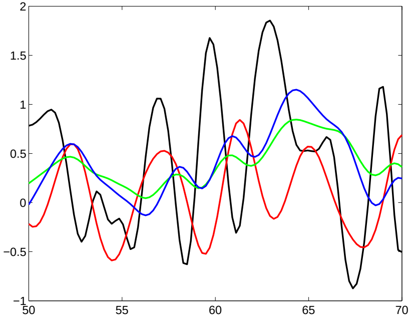


Figure: Joint samples from the ODE covariance, *black*: $f(t)$, *red*: $x_1(t)$ (underdamped), *green*: $x_2(t)$ (overdamped), and *blue*: $x_3(t)$ (critically damped).

Joint Sampling of $x(t)$ and $f(t)$

- `lfmSample`

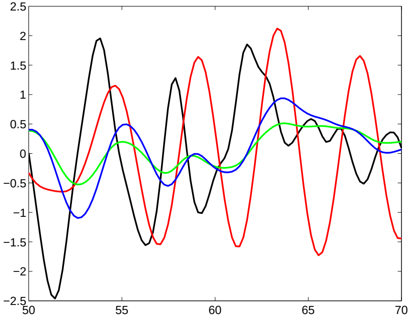


Figure: Joint samples from the ODE covariance, *black*: $f(t)$, *red*: $x_1(t)$ (underdamped), *green*: $x_2(t)$ (overdamped), and *blue*: $x_3(t)$ (critically damped).

Joint Sampling of $x(t)$ and $f(t)$

- `lfmSample`

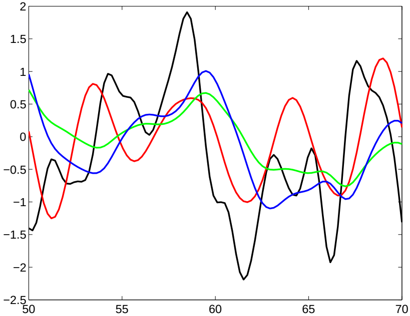


Figure: Joint samples from the ODE covariance, *black*: $f(t)$, *red*: $x_1(t)$ (underdamped), *green*: $x_2(t)$ (overdamped), and *blue*: $x_3(t)$ (critically damped).

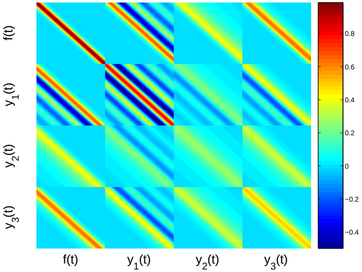
Covariance for ODE

- Exponentiated Quadratic Covariance function for $f(t)$

$$x_j(t) = \frac{1}{m_j\omega_j}\sum_{i=1}^{q} s_{ji}\exp(-\alpha_j t)\int_0^t f_i(\tau)\exp(\alpha_j\tau)\sin(\omega_j(t-\tau))\mathrm{d}\tau$$

- Joint distribution for $x_1(t)$, $x_2(t)$, $x_3(t)$ and $f(t)$.
- Damping ratios:

ζ_1	ζ_2	ζ_3
0.125	2	1



Outline

Machine Learning

Gaussian Processes

Motion Capture Example

ODE Model of Transcriptional Regulation

Discussion and Future Work

Example: Motion Capture

Mauricio Alvarez and David Luengo (Álvarez et al., 2009, 2013)

- Motion capture data: used for animating human motion.

Example: Motion Capture

Mauricio Alvarez and David Luengo (Álvarez et al., 2009, 2013)

- Motion capture data: used for animating human motion.
- Multivariate time series of angles representing joint positions.

Example: Motion Capture

Mauricio Alvarez and David Luengo (Álvarez et al., 2009, 2013)

- Motion capture data: used for animating human motion.
- Multivariate time series of angles representing joint positions.
- Objective: generalize from training data to realistic motions.

Example: Motion Capture

Mauricio Alvarez and David Luengo (Álvarez et al., 2009, 2013)

- Motion capture data: used for animating human motion.
- Multivariate time series of angles representing joint positions.
- Objective: generalize from training data to realistic motions.
- Use 2nd Order Latent Force Model with mass/spring/damper (resistor inductor capacitor) at each joint.

Prediction of Test Motion

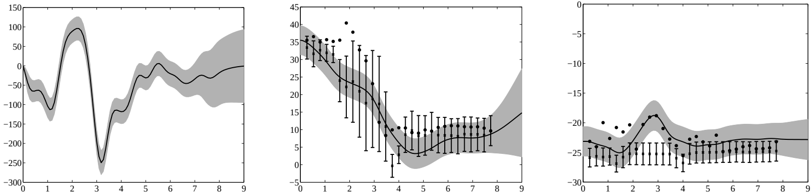
- Model left arm only.
- 3 balancing motions (18, 19, 20) from subject 49.
- 18 and 19 are similar, 20 contains more dramatic movements.
- Train on 18 and 19 and testing on 20
- Data was down-sampled by 32 (from 120 fps).
- Reconstruct motion of left arm for 20 given other movements.
- Compare with GP that predicts left arm angles given other body angles.

Mocap Results

Table: Root mean squared (RMS) angle error for prediction of the left arm's configuration in the motion capture data. Prediction with the latent force model outperforms the prediction with regression for all apart from the radius's angle.

Angle	Latent Force Error	Regression Error
Radius	4.11	4.02
Wrist	6.55	6.65
Hand X rotation	1.82	3.21
Hand Z rotation	2.76	6.14
Thumb X rotation	1.77	3.10
Thumb Z rotation	2.73	6.09

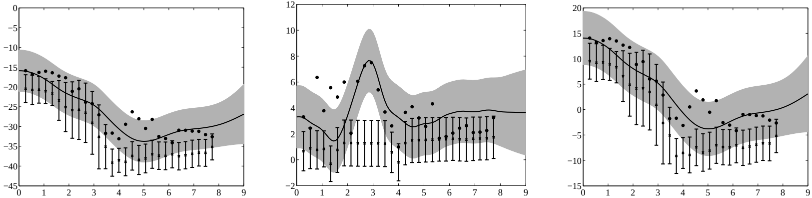
Mocap Results II



(a) Inferred Latent Force

(b) Wrist

(c) Hand X Rotation



(d) Hand Z Rotation

(e) Thumb X Rotation

(f) Thumb Z Rotation

Figure: Predictions from LFM (solid line, grey error bars) and direct regression (crosses with stick error bars).

Motion Capture Experiments

- Data set is from the CMU motion capture data base[1].
- Two different types of movements: golf-swing and walking.
- Train on a subset of motions for each movement and test on a different subset.
- This assesses the model's ability to extrapolate.
- For testing: condition on three angles associated to the root nodes and first five and last five frames of the motion.
- Golf-swing use leave one out cross validation on four motions.
- For the walking train on 4 motions and validate on 8 motions.

Motion Capture Results

Table: RMSE and R^2 (explained variance) for golf swing and walking

Movement	**Method**	**RMSE**	**R^2 (%)**
Golf swing	IND GP	21.55 ± 2.35	30.99 ± 9.67
	MTGP	21.19 ± 2.18	45.59 ± 7.86
	SLFM	21.52 ± 1.93	49.32 ± 3.03
	LFM	**18.09 ± 1.30**	**72.25 ± 3.08**
Walking	IND GP	8.03 ± 2.55	30.55 ± 10.64
	MTGP	7.75 ± 2.05	37.77 ± 4.53
	SLFM	7.81 ± 2.00	36.84 ± 4.26
	LFM	**7.23 ± 2.18**	**48.15 ± 5.66**

Outline

- Machine Learning
- Gaussian Processes
- Motion Capture Example
- ODE Model of Transcriptional Regulation
- Discussion and Future Work

Example: Transcriptional Regulation

- First Order Differential Equation

$$\frac{\mathrm{d}x_j(t)}{\mathrm{d}t} = b_j + s_j f(t) - d_j x_j(t)$$

Example: Transcriptional Regulation

- First Order Differential Equation

$$\frac{\mathrm{d}x_j(t)}{\mathrm{d}t} = b_j + s_j f(t) - d_j x_j(t)$$

- Can be used as a model of gene transcription: Barenco et al., 2006; Gao et al., 2008.

Example: Transcriptional Regulation

- First Order Differential Equation

$$\frac{\mathrm{d}x_j(t)}{\mathrm{d}t} = b_j + s_j f(t) - d_j x_j(t)$$

- Can be used as a model of gene transcription: Barenco et al., 2006; Gao et al., 2008.
- $x_j(t)$ – concentration of gene j's mRNA

Example: Transcriptional Regulation

- First Order Differential Equation

$$\frac{\mathrm{d}x_j\left(t\right)}{\mathrm{d}t} = b_j + s_j f\left(t\right) - d_j x_j\left(t\right)$$

- Can be used as a model of gene transcription: Barenco et al., 2006; Gao et al., 2008.
- $x_j(t)$ – concentration of gene j's mRNA
- $f(t)$ – concentration of active transcription factor

Example: Transcriptional Regulation

- First Order Differential Equation

$$\frac{\mathrm{d}x_j(t)}{\mathrm{d}t} = b_j + s_j f(t) - d_j x_j(t)$$

- Can be used as a model of gene transcription: Barenco et al., 2006; Gao et al., 2008.
- $x_j(t)$ – concentration of gene j's mRNA
- $f(t)$ – concentration of active transcription factor
- Model parameters: baseline b_j, sensitivity s_j and decay d_j

Example: Transcriptional Regulation

- First Order Differential Equation

$$\frac{\mathrm{d}x_j\left(t\right)}{\mathrm{d}t} = b_j + s_j f\left(t\right) - d_j x_j\left(t\right)$$

- Can be used as a model of gene transcription: Barenco et al., 2006; Gao et al., 2008.
- $x_j(t)$ – concentration of gene j's mRNA
- $f(t)$ – concentration of active transcription factor
- Model parameters: baseline b_j, sensitivity s_j and decay d_j
- Application: identifying co-regulated genes (targets)

Example: Transcriptional Regulation

- First Order Differential Equation

$$\frac{\mathrm{d}x_j(t)}{\mathrm{d}t} = b_j + s_j f(t) - d_j x_j(t)$$

- Can be used as a model of gene transcription: Barenco et al., 2006; Gao et al., 2008.
- $x_j(t)$ – concentration of gene j's mRNA
- $f(t)$ – concentration of active transcription factor
- Model parameters: baseline b_j, sensitivity s_j and decay d_j
- Application: identifying co-regulated genes (targets)
- Problem: how do we fit the model when $f(t)$ is not observed?

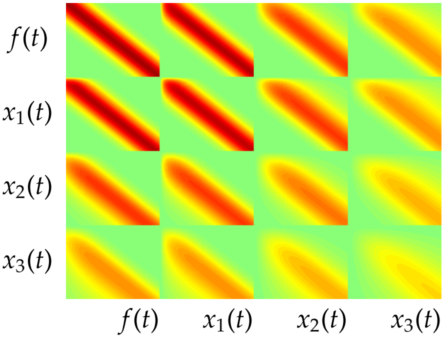
Covariance for Transcription Model

RBF covariance function for $f(t)$

$$x_i(t) = \frac{b_i}{d_i} + s_i \exp(-d_i t) \int_0^t f(u) \exp(d_i u)\, \mathrm{d}u.$$

- Joint distribution for $x_1(t)$, $x_2(t)$, $x_3(t)$, and $f(t)$.
- Here:

d_1	s_1	d_2	s_2	d_3	s_3
5	5	1	1	0.5	0.5



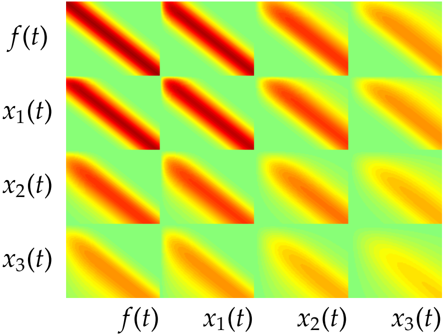
Covariance for Transcription Model

RBF covariance function for $f(t)$

$$x = b/d + \sum_i \mathbf{e}_i^\top \mathbf{f} \quad \mathbf{f} \sim \mathcal{N}\left(\mathbf{0}, \Sigma_i\right) \rightarrow x \sim \mathcal{N}\left(b/d, \sum_i \mathbf{e}_i^\top \Sigma_i \mathbf{e}_i\right)$$

- Joint distribution for $x_1(t)$, $x_2(t)$, $x_3(t)$, and $f(t)$.
- Here:

d_1	s_1	d_2	s_2	d_3	s_3
5	5	1	1	0.5	0.5



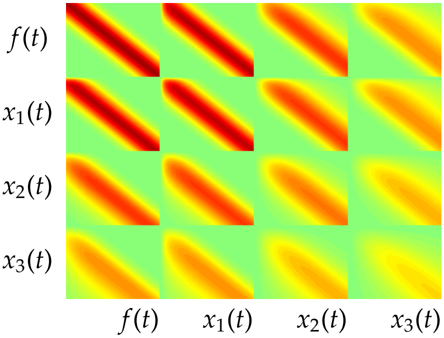
Covariance for Transcription Model

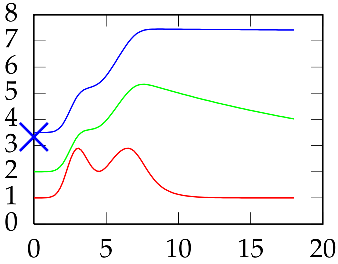
RBF covariance function for $f(t)$

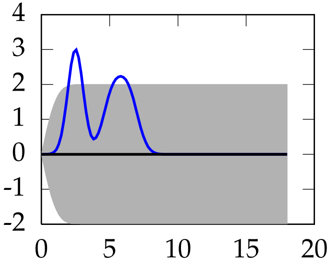
$$x_i(t) = \frac{b_i}{d_i} + s_i \exp(-d_i t) \int_0^t f(u) \exp(d_i u) \, \mathrm{d}u.$$

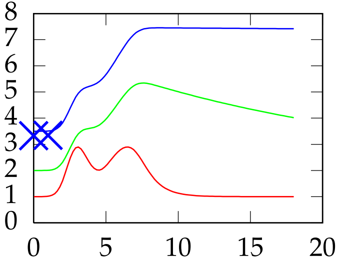
- Joint distribution for $x_1(t)$, $x_2(t)$, $x_3(t)$, and $f(t)$.
- Here:

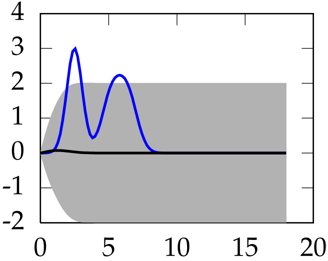
d_1	s_1	d_2	s_2	d_3	s_3
5	5	1	1	0.5	0.5

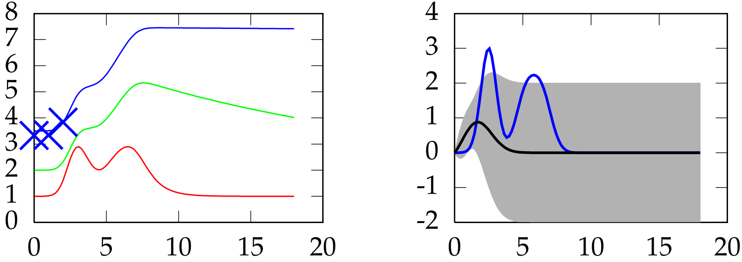


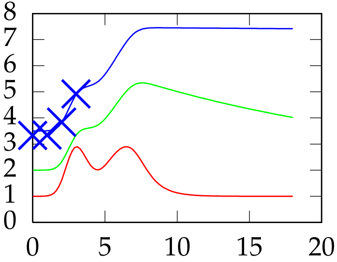

8
7
6
5
4
3
2
1
0
0
5
10
15
20

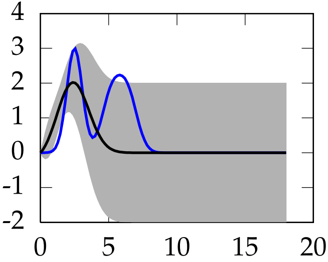

4
3
2
1
0
-1
-2
0
5
10
15
20

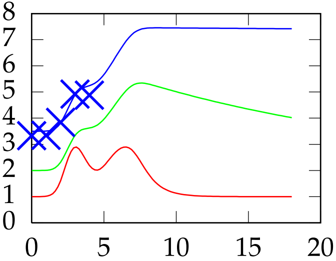

8
7
6
5
4
3
2
1
0
0
5
10
15
20

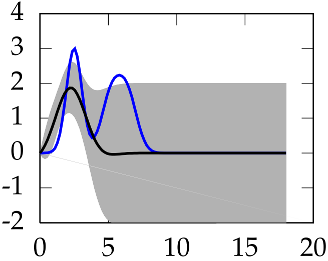

4
3
2
1
0
-1
-2
0
5
10
15
20

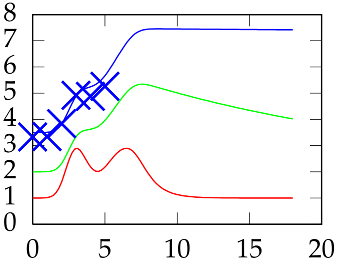

8
7
6
5
4
3
2
1
0
0
5
10
15
20
4
3
2
1
0
-1
-2
0
5
10
15
20

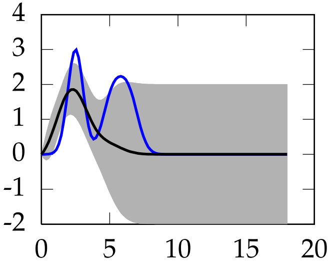

8
7
6
5
4
3
2
1
0
0
5
10
15
20

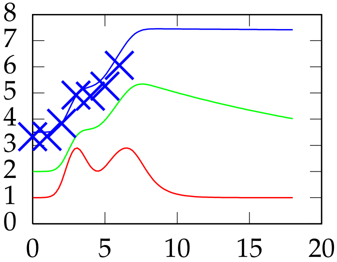

4
3
2
1
0
-1
-2
0
5
10
15
20

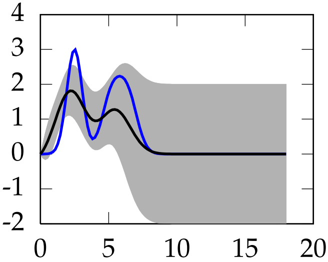

8
7
6
5
4
3
2
1
0
0
5
10
15
20

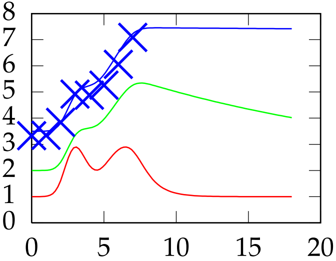

4
3
2
1
0
-1
-2
0
5
10
15
20

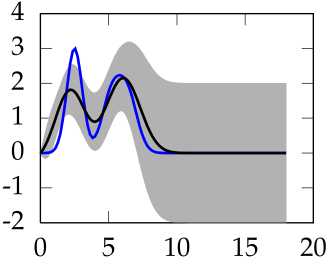

8
7
6
5
4
3
2
1
0
0
5
10
15
20

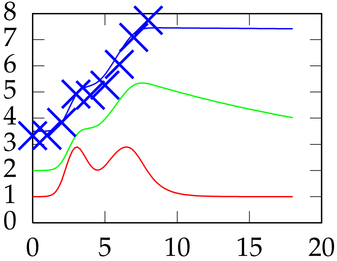

4
3
2
1
0
-1
-2
0
5
10
15
20

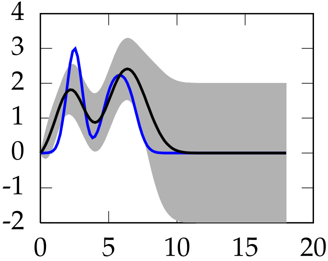

8
7
6
5
4
3
2
1
0
0
5
10
15
20

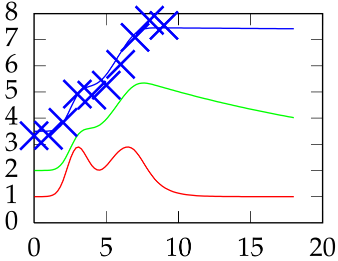

4
3
2
1
0
-1
-2
0
5
10
15
20

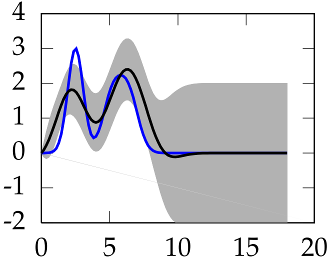

8
7
6
5
4
3
2
1
0
0
5
10
15
20

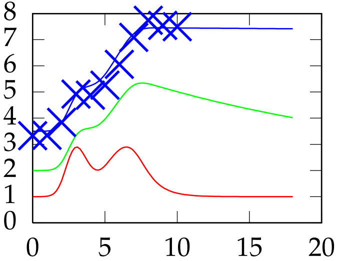

4
3
2
1
0
-1
-2
0
5
10
15
20

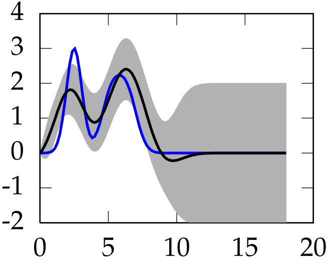

8
7
6
5
4
3
2
1
0
0
5
10
15
20

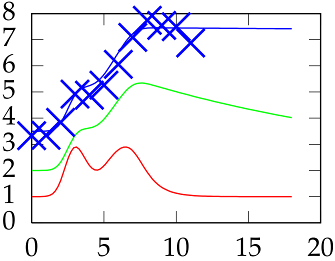

4
3
2
1
0
-1
-2
0
5
10
15
20

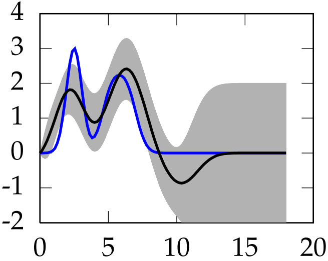

8
7
6
5
4
3
2
1
0
0
5
10
15
20

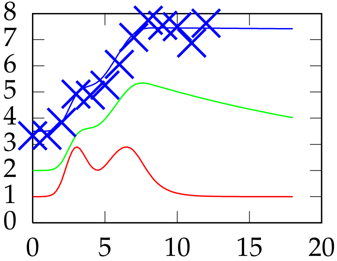

4
3
2
1
0
-1
-2
0
5
10
15
20

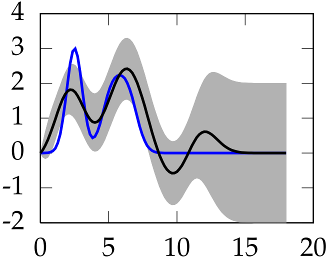

8
7
6
5
4
3
2
1
0
0
5
10
15
20

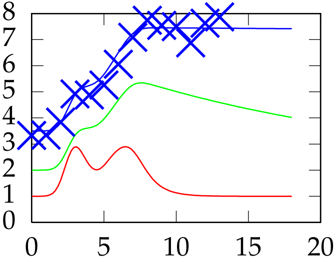

4
3
2
1
0
-1
-2
0
5
10
15
20

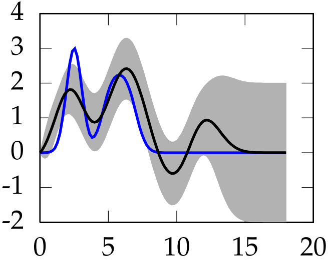

0
1
2
3
4
5
6
7
8
0
5
10
15
20

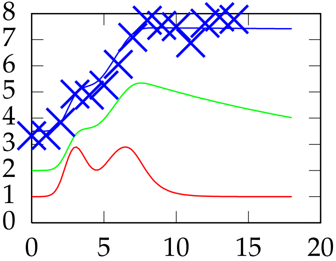

-2
-1
0
1
2
3
4
0
5
10
15
20

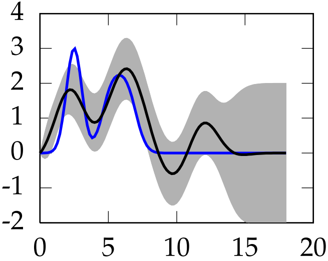

8
7
6
5
4
3
2
1
0
0
5
10
15
20

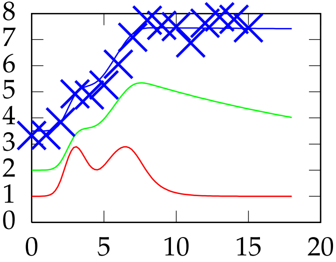

4
3
2
1
0
-1
-2
0
5
10
15
20

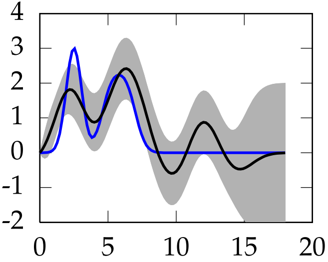

8
7
6
5
4
3
2
1
0
0
5
10
15
20

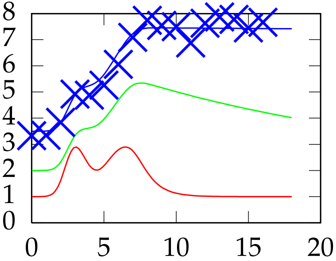

4
3
2
1
0
-1
-2
0
5
10
15
20

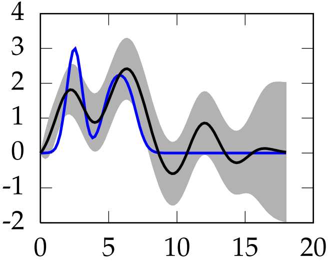

8
7
6
5
4
3
2
1
0
0
5
10
15
20

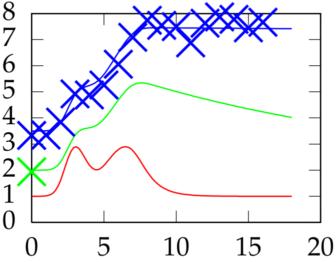

4
3
2
1
0
-1
-2
0
5
10
15
20

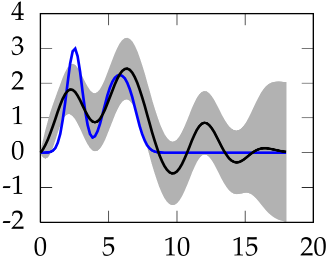

8
7
6
5
4
3
2
1
0
0
5
10
15
20

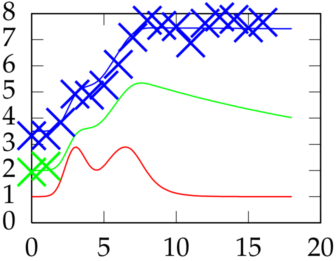

4
3
2
1
0
-1
-2
0
5
10
15
20

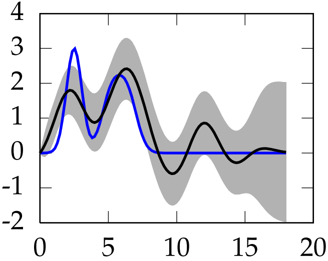

8
7
6
5
4
3
2
1
0
0
5
10
15
20

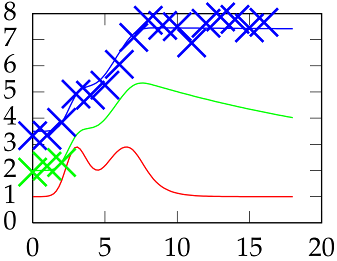

4
3
2
1
0
-1
-2
0
5
10
15
20

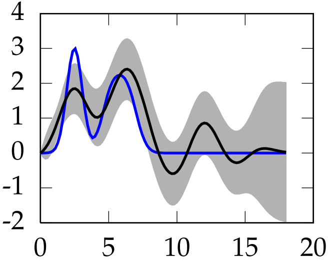

8
7
6
5
4
3
2
1
0
0
5
10
15
20

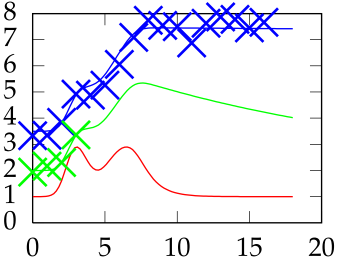

4
3
2
1
0
-1
-2
0
5
10
15
20

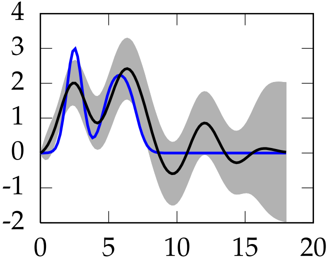

8
7
6
5
4
3
2
1
0
0
5
10
15
20

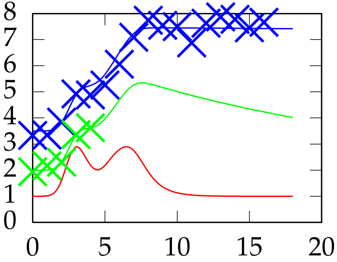

4
3
2
1
0
-1
-2
0
5
10
15
20

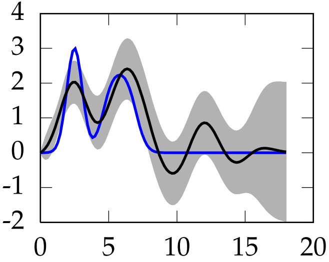

8
7
6
5
4
3
2
1
0
0
5
10
15
20

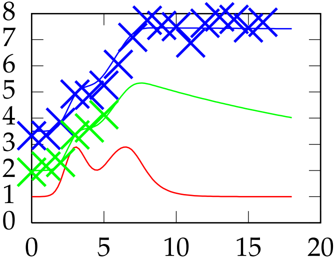

4
3
2
1
0
-1
-2
0
5
10
15
20

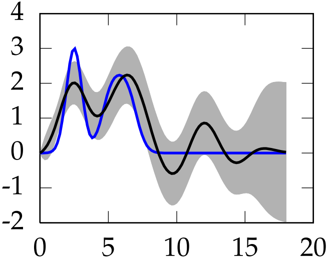

8
7
6
5
4
3
2
1
0
0
5
10
15
20

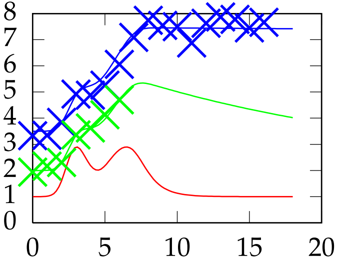

4
3
2
1
0
-1
-2
0
5
10
15
20

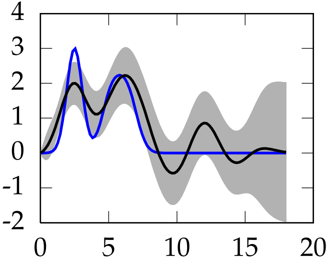

8
7
6
5
4
3
2
1
0
0
5
10
15
20

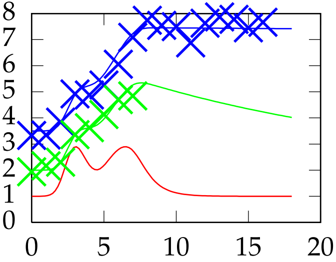

4
3
2
1
0
-1
-2
0
5
10
15
20

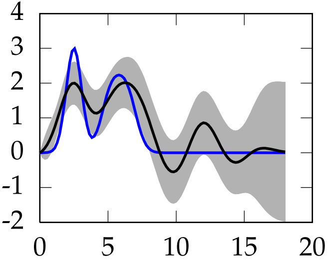

0
1
2
3
4
5
6
7
8
0
5
10
15
20

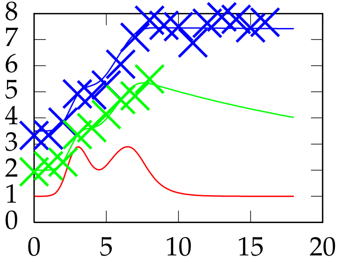

-2
-1
0
1
2
3
4
0
5
10
15
20

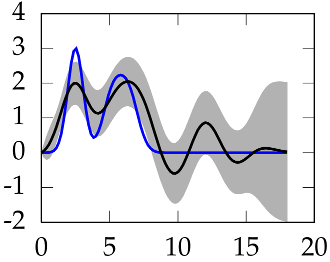

8
7
6
5
4
3
2
1
0
0
5
10
15
20

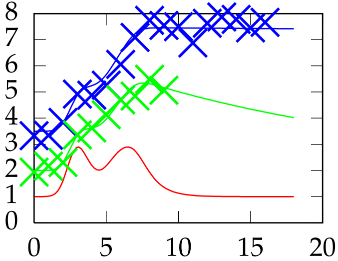

4
3
2
1
0
-1
-2
0
5
10
15
20

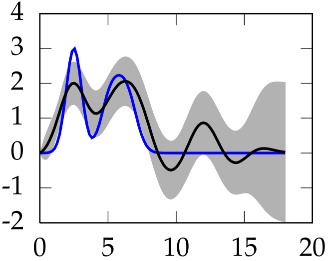

8
7
6
5
4
3
2
1
0
0
5
10
15
20

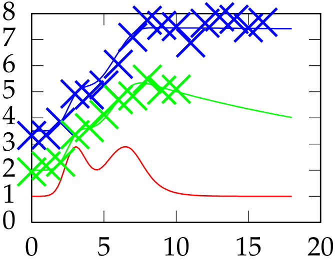

4
3
2
1
0
-1
-2
0
5
10
15
20

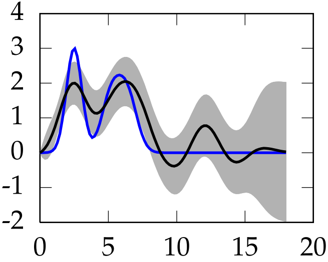

8
7
6
5
4
3
2
1
0
0
5
10
15
20

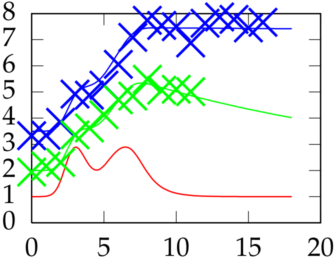

4
3
2
1
0
-1
-2
0
5
10
15
20

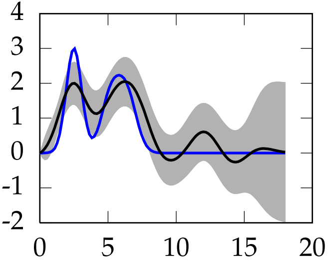

8
7
6
5
4
3
2
1
0
0
5
10
15
20

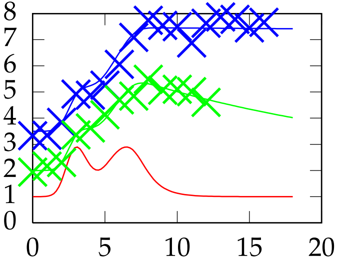

4
3
2
1
0
-1
-2
0
5
10
15
20

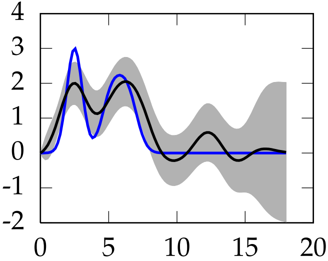

8
7
6
5
4
3
2
1
0
0
5
10
15
20

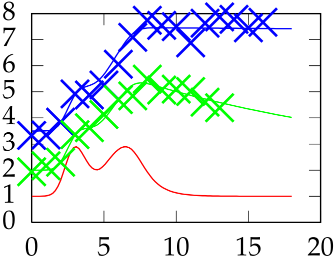

4
3
2
1
0
-1
-2
0
5
10
15
20

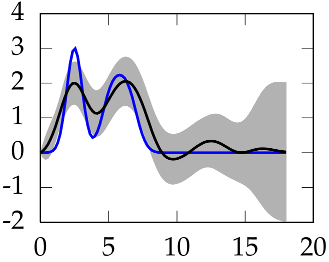

0
1
2
3
4
5
6
7
8
0
5
10
15
20

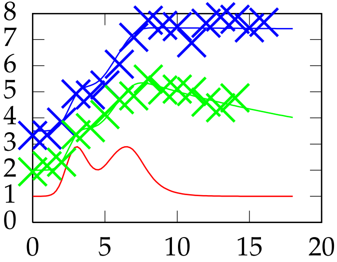

-2
-1
0
1
2
3
4
0
5
10
15
20

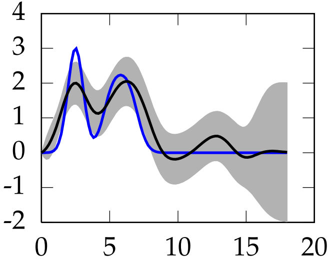

8
7
6
5
4
3
2
1
0
0
5
10
15
20

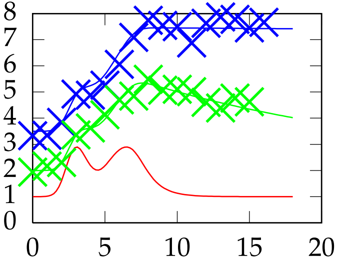

4
3
2
1
0
-1
-2
0
5
10
15
20

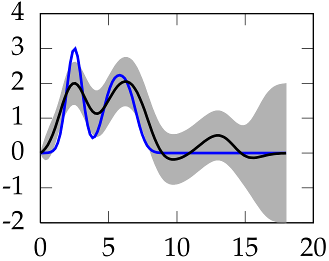

8
7
6
5
4
3
2
1
0
0
5
10
15
20

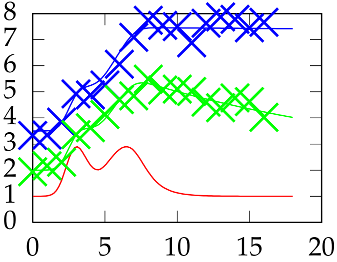

4
3
2
1
0
-1
-2
0
5
10
15
20

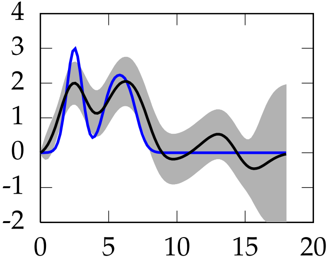

8
7
6
5
4
3
2
1
0
0
5
10
15
20

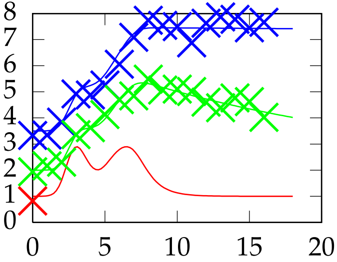

4
3
2
1
0
-1
-2
0
5
10
15
20

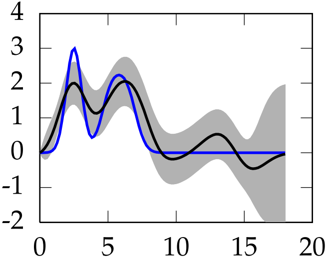

8
7
6
5
4
3
2
1
0
0
5
10
15
20

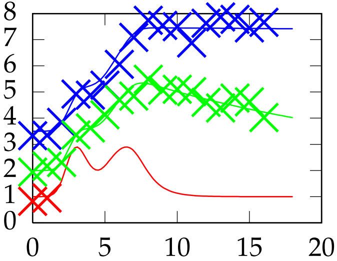

4
3
2
1
0
-1
-2
0
5
10
15
20

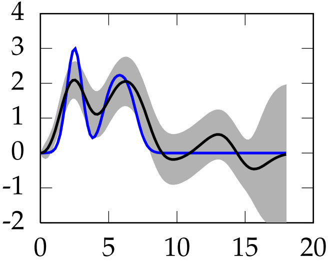

8
7
6
5
4
3
2
1
0
0
5
10
15
20

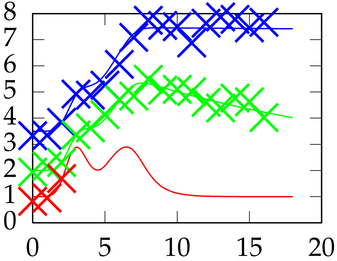

4
3
2
1
0
-1
-2
0
5
10
15
20

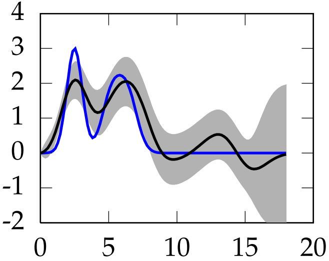

8
7
6
5
4
3
2
1
0
0
5
10
15
20

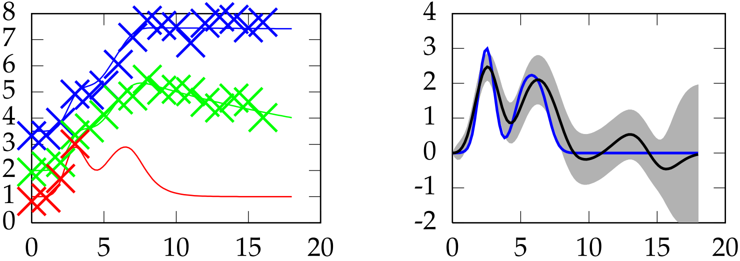

4
3
2
1
0
-1
-2
0
5
10
15
20

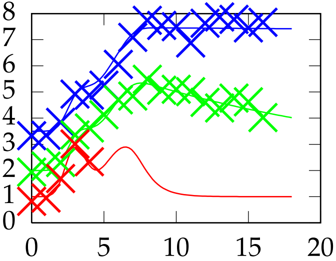

8
7
6
5
4
3
2
1
0
0
5
10
15
20

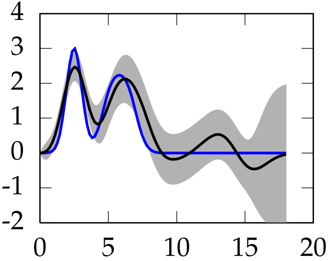

4
3
2
1
0
-1
-2
0
5
10
15
20

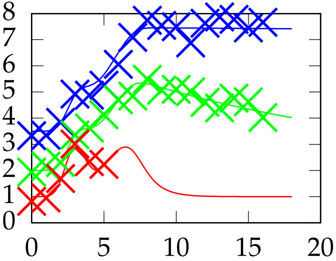

8
7
6
5
4
3
2
1
0
0
5
10
15
20

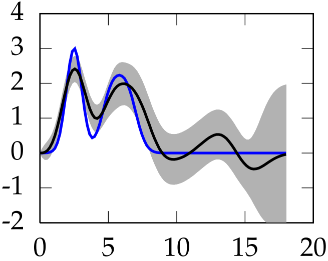

4
3
2
1
0
-1
-2
0
5
10
15
20

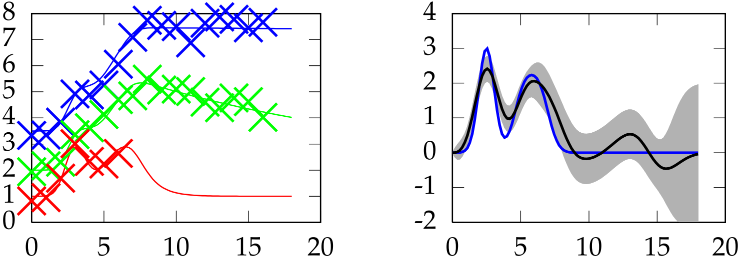

8
7
6
5
4
3
2
1
0
0
5
10
15
20
4
3
2
1
0
-1
-2
0
5
10
15
20


8
7
6
5
4
3
2
1
0
0
5
10
15
20

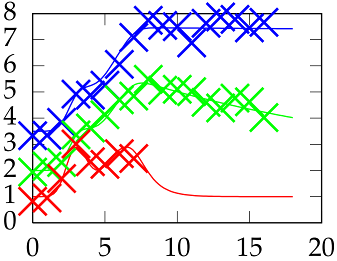

4
3
2
1
0
-1
-2
0
5
10
15
20

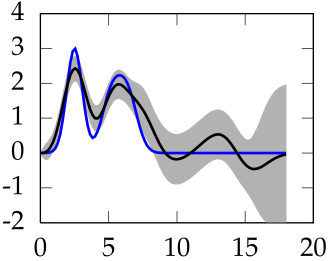

8
7
6
5
4
3
2
1
0
0
5
10
15
20

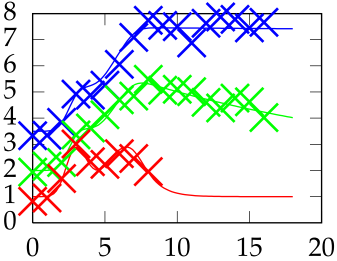

4
3
2
1
0
-1
-2
0
5
10
15
20

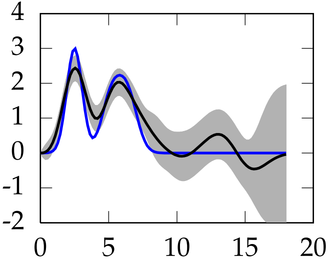


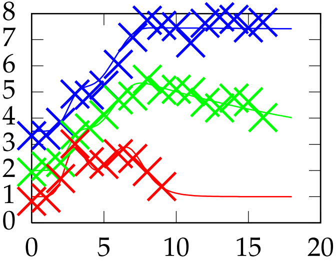
8
7
6
5
4
3
2
1
0
0
5
10
15
20
4
3
2
1
0
-1
-2
0
5
10
15
20

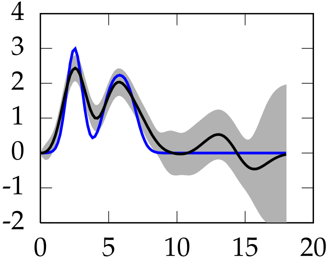

8
7
6
5
4
3
2
1
0
0
5
10
15
20

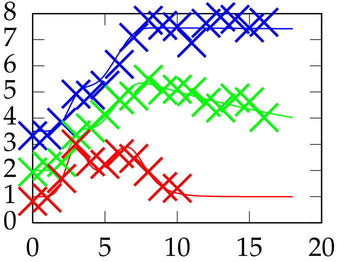

4
3
2
1
0
-1
-2
0
5
10
15
20

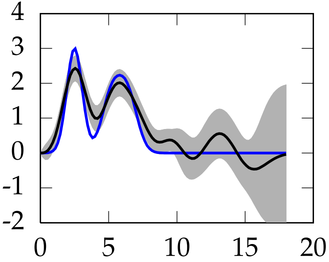

8
7
6
5
4
3
2
1
0
0
5
10
15
20

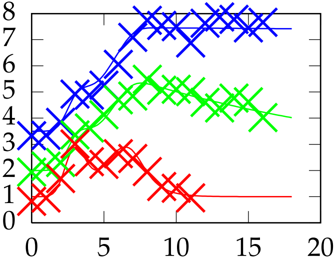

4
3
2
1
0
-1
-2
0
5
10
15
20


8
7
6
5
4
3
2
1
0
0
5
10
15
20

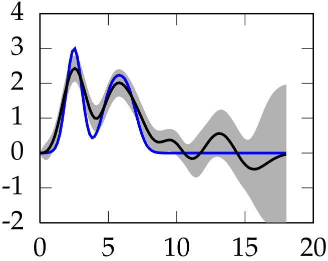

4
3
2
1
0
-1
-2
0
5
10
15
20


8
7
6
5
4
3
2
1
0
0
5
10
15
20

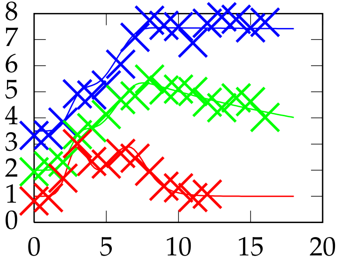

4
3
2
1
0
-1
-2
0
5
10
15
20

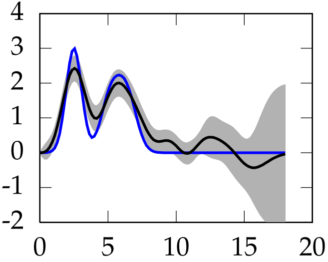


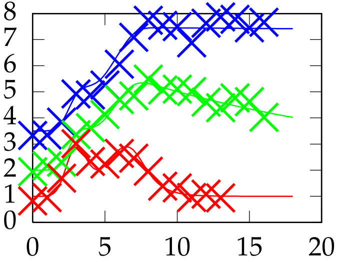
8
7
6
5
4
3
2
1
0
0
5
10
15
20

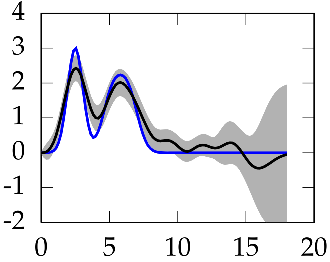


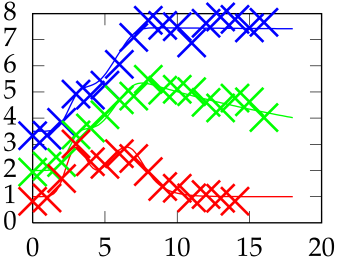
4
3
2
1
0
-1
-2
0
5
10
15
20

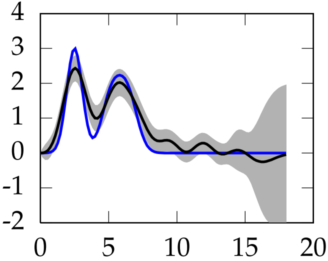

8
7
6
5
4
3
2
1
0
0
5
10
15
20

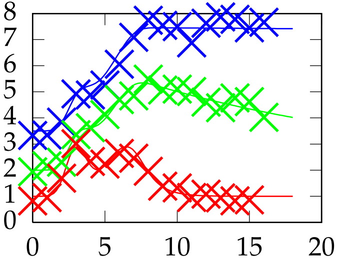

4
3
2
1
0
-1
-2
0
5
10
15
20

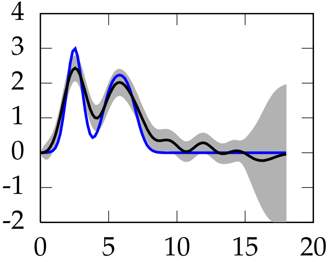

8
7
6
5
4
3
2
1
0
0
5
10
15
20

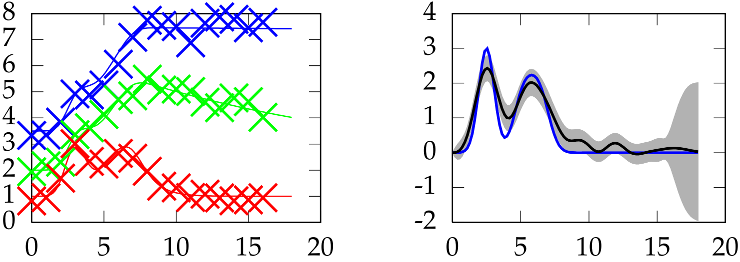

4
3
2
1
0
-1
-2
0
5
10
15
20


8
7
6
5
4
3
2
1
0
0
5
10
15
20

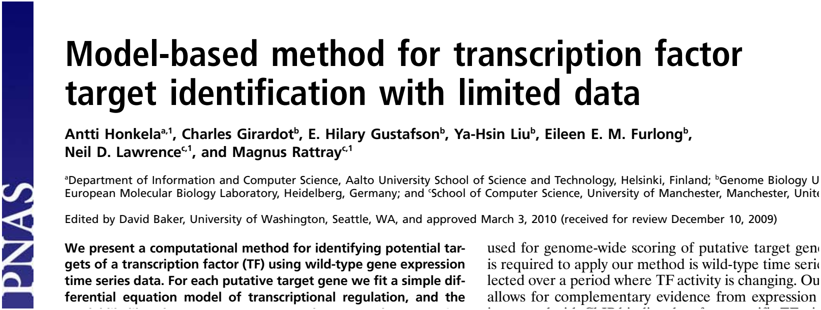

4
3
2
1
0
-1
-2
0
5
10
15
20


8
7
6
5
4
3
2
1
0
0
5
10
15
20


4
3
2
1
0
-1
-2
0
5
10
15
20


8
7
6
5
4
3
2
1
0
0
5
10
15
20
4
3
2
1
0
-1
-2
0
5
10
15
20

Cascaded Differential Equations



PNAS

Model-based method for transcription factor target identification with limited data

Antti Honkela[a,1], Charles Girardot[b], E. Hilary Gustafson[b], Ya-Hsin Liu[b], Eileen E. M. Furlong[b], Neil D. Lawrence[c,1], and Magnus Rattray[c,1]

[a]Department of Information and Computer Science, Aalto University School of Science and Technology, Helsinki, Finland; [b]Genome Biology U
European Molecular Biology Laboratory, Heidelberg, Germany; and [c]School of Computer Science, University of Manchester, Manchester, Unite

Edited by David Baker, University of Washington, Seattle, WA, and approved March 3, 2010 (received for review December 10, 2009)

We present a computational method for identifying potential targets of a transcription factor (TF) using wild-type gene expression time series data. For each putative target gene we fit a simple differential equation model of transcriptional regulation, and the

used for genome-wide scoring of putative target gen
is required to apply our method is wild-type time seri
lected over a period where TF activity is changing. Ou
allows for complementary evidence from expression

Cascaded Differential Equations

(Honkela et al., 2010)

- Transcription factor protein also has governing mRNA.
- This mRNA can be measured.
- In signalling systems this measurement can be misleading because it is activated (phosphorylated) transcription factor that counts.
- In development phosphorylation plays less of a role.
- Build a simple cascaded differential equation to model this.

Covariance for Translation/Transcription Model

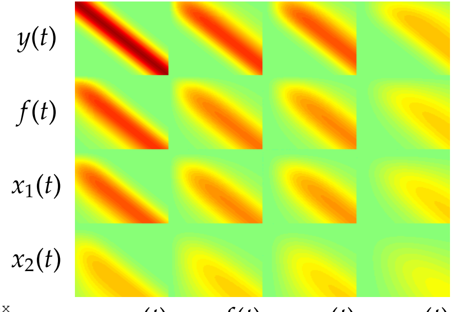
RBF covariance function for $y(t)$

$$f(t) = \sigma \exp(-\delta t) \int_0^t y(u) \exp(\delta u) \, \mathrm{d}u$$

$$x_i(t) = \frac{b_i}{d_i} + s_i \exp(-d_i t) \int_0^t f(u) \exp(d_i u) \, \mathrm{d}u.$$

- Joint distribution for $x_1(t)$, $x_2(t)$, $f(t)$ and $y(t)$.
- Here:

δ	d_1	s_1	d_2	s_2
1	5	5	0.5	0.5



Twist Results

- Use mRNA of Twist as driving input.
- For each gene build a cascade model that forces Twist to be the only TF.
- Compare fit of this model to a baseline (*e.g.* similar model but sensitivity zero).
- Rank according to the likelihood above the baseline.
- Compare with correlation, knockouts and time series network identification (TSNI) (Della Gatta et al., 2008).

Results for Twi using the Cascade model

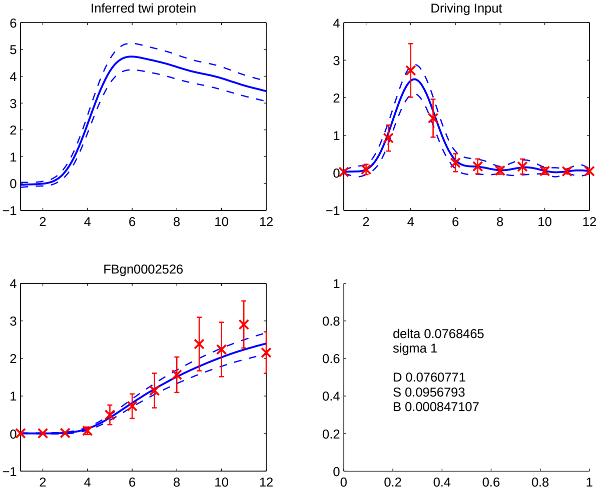
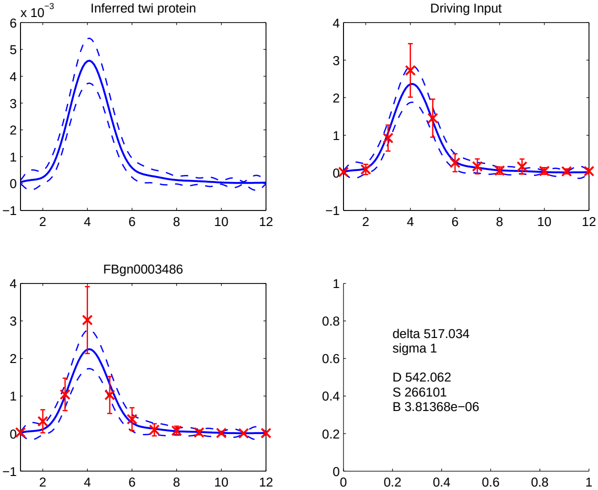


Figure: Model for flybase gene identity FBgn0002526.

Results for Twi using the Cascade model



delta 517.034
sigma 1

D 542.062
S 266101
B 3.81368e−06

Figure: Model for flybase gene identity FBgn0003486.

Results for Twi using the Cascade model

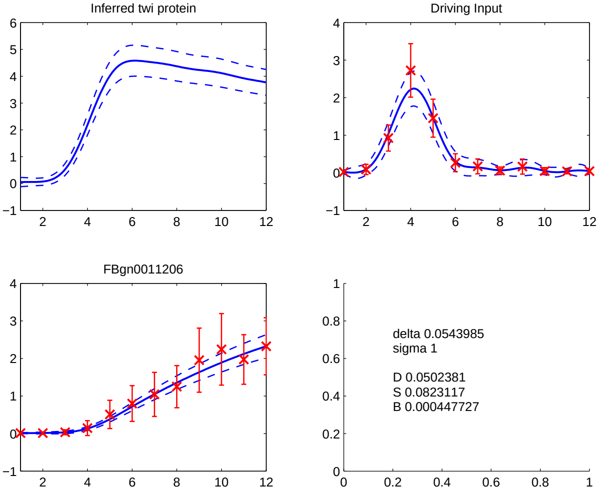
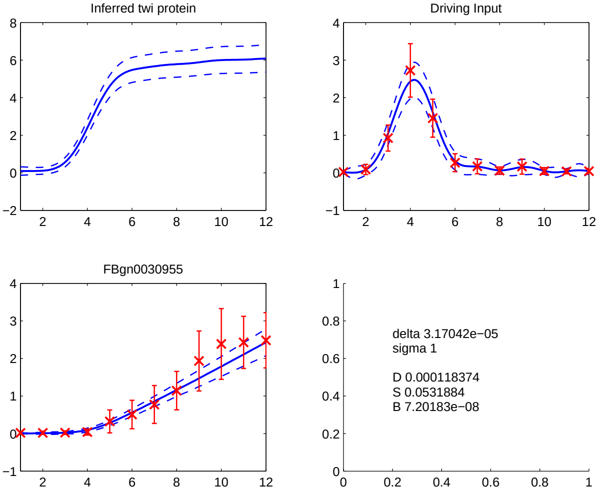


Figure: Model for flybase gene identity FBgn0011206.

Results for Twi using the Cascade model



delta 3.17042e−05
sigma 1

D 0.000118374
S 0.0531884
B 7.20183e−08

Figure: Model for flybase gene identity FBgn00309055.

Results for Twi using the Cascade model

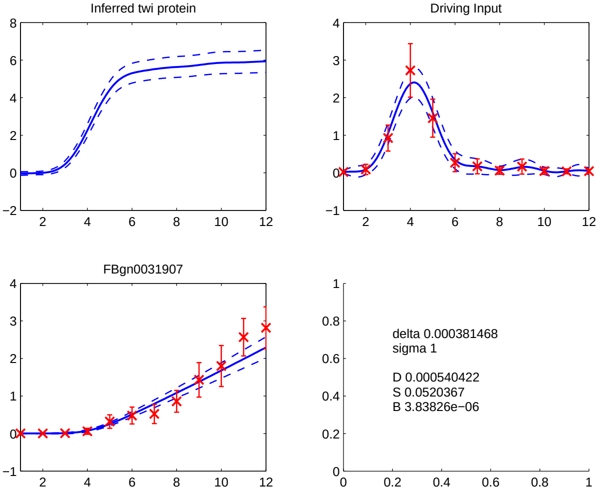
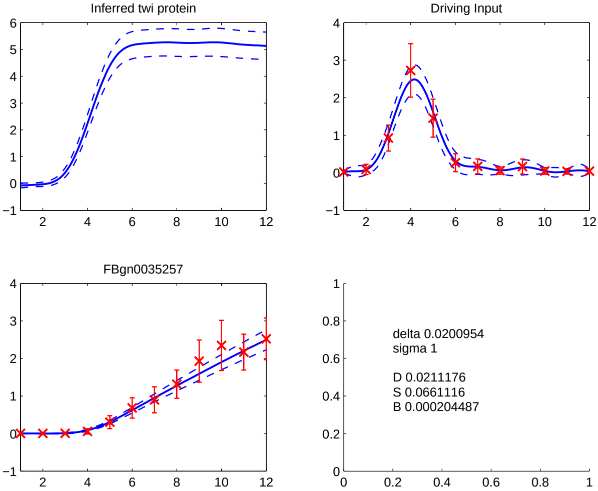


Figure: Model for flybase gene identity FBgn0031907.

Results for Twi using the Cascade model



delta 0.0200954
sigma 1

D 0.0211176
S 0.0661116
B 0.000204487

Figure: Model for flybase gene identity FBgn0035257.

Results for Twi using the Cascade model

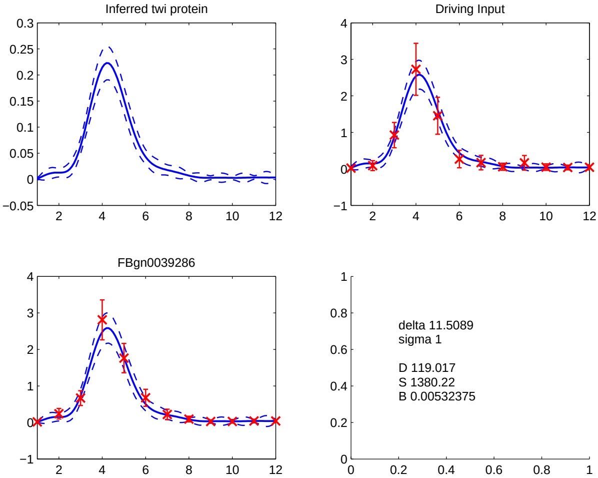
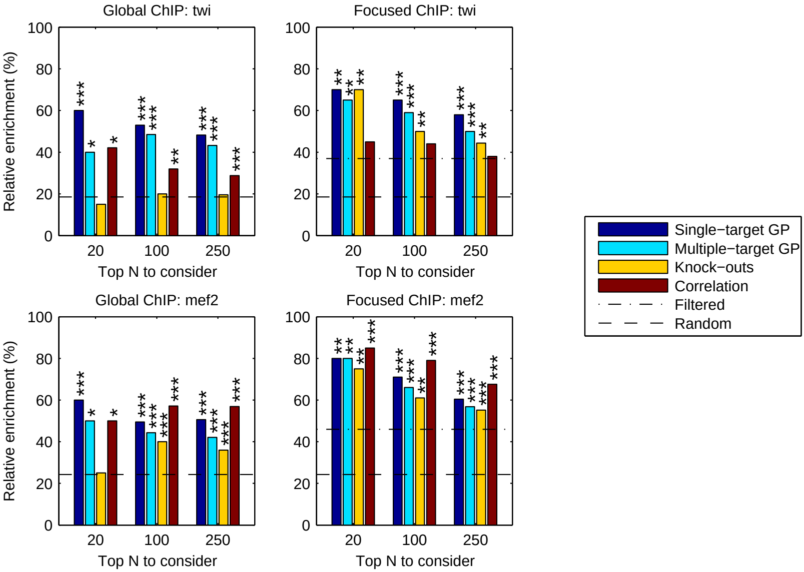


Figure: Model for flybase gene identity FBgn0039286.

Evaluation methods

- Evaluate the ranking methods by taking a number of top-ranked targets and record the number of "positives" (Zinzen et al., 2009):
    - targets with ChIP-chip binding sites within 2 kb of gene
    - (targets differentially expressed in TF knock-outs)
- Compare against
    - Ranking by correlation of expression profiles
    - Ranking by q-value of differential expression in knock-outs
- Optionally focus on genes with annotated expression in tissues of interest

Results



'***': $p < 0.001$, '**': $p < 0.01$, '*': $p < 0.05$

Summary

- Cascade models allow genomewide analysis of potential targets given only expression data.
- Once a set of potential candidate targets have been identified, they can be modelled in a more complex manner.
- We don't have ground truth, but evidence indicates that the approach *can* perform as well as knockouts.

Outline

Machine Learning

Gaussian Processes

Motion Capture Example

ODE Model of Transcriptional Regulation

Discussion and Future Work

Discussion and Future Work

- Integration of probabilistic inference with mechanistic models.
- Ongoing/other work:
    - Non linear response and non linear differential equations.
    - Scaling up to larger systems Álvarez et al. (2010a); Álvarez and Lawrence (2009).
    - Discontinuities through Switched Gaussian Processes Álvarez et al. (2010b)
    - Robotics applications.
    - Applications to other types of system, *e.g.* spatial systems Álvarez et al. (2011).
    - Stochastic differential equations Álvarez et al. (2010a).

Acknowledgements

Investigators	Neil Lawrence and Magnus Rattray
Researchers	Mauricio Álvarez, Pei Gao, Antti Honkela, David Luengo, Guido Sanguinetti, Michalis Titsias, and Jennifer Withers
Lawrence/Ratray Funding	BBSRC award "Improved Processing of microarray data using probabilistic models", EPSRC award "Gaussian Processes for Systems Identification with applications in Systems Biology", University of Manchester, Computer Science Studentship, and **Google Research Award**: "Mechanistically Inspired Convolution Processes for Learning".
Other funding	David Luengo's visit to Manchester was financed by the Comunidad de Madrid (project PRO-MULTIDIS-CM, S-0505/TIC/0233), and by the Spanish government (CICYT project TEC2006-13514-C02-01 and researh grant JC2008-00219). Antti Honkela visits to Manchester funded by PASCAL I & II EU Networks of excellence.

References I

M. A. Álvarez and N. D. Lawrence. Sparse convolved Gaussian processes for multi-output regression. In D. Koller, D. Schuurmans, Y. Bengio, and L. Bottou, editors, *Advances in Neural Information Processing Systems*, volume 21, pages 57–64, Cambridge, MA, 2009. MIT Press. [PDF].

M. A. Álvarez, D. Luengo, and N. D. Lawrence. Latent force models. In D. van Dyk and M. Welling, editors, *Proceedings of the Twelfth International Workshop on Artificial Intelligence and Statistics*, volume 5, pages 9–16, Clearwater Beach, FL, 16-18 April 2009. JMLR W&CP 5. [PDF].

M. A. Álvarez, D. Luengo, and N. D. Lawrence. Linear latent force models using Gaussian processes. Technical report, University of Sheffield, [PDF].

M. A. Álvarez, D. Luengo, and N. D. Lawrence. Linear latent force models using Gaussian processes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(11):2693–2705, 2013. [PDF].

M. A. Álvarez, D. Luengo, M. K. Titsias, and N. D. Lawrence. Efficient multioutput Gaussian processes through variational inducing kernels. In Y. W. Teh and D. M. Titterington, editors, *Proceedings of the Thirteenth International Workshop on Artificial Intelligence and Statistics*, volume 9, pages 25–32, Chia Laguna Resort, Sardinia, Italy, 13-16 May 2010a. JMLR W&CP 9. [PDF].

M. A. Álvarez, J. Peters, B. Schölkopf, and N. D. Lawrence. Switched latent force models for movement segmentation. In J. Lafferty, C. K. I. Williams, J. Shawe-Taylor, R. S. Zemel, and A. Culotta, editors, *Advances in Neural Information Processing Systems*, volume 23, pages 55–63, Cambridge, MA, 2010b. MIT Press.

M. Barenco, D. Tomescu, D. Brewer, R. Callard, J. Stark, and M. Hubank. Ranked prediction of p53 targets using hidden variable dynamic modeling. *Genome Biology*, 7(3):R25, 2006.

G. Della Gatta, M. Bansal, A. Ambesi-Impiombato, D. Antonini, C. Missero, and D. di Bernardo. Direct targets of the trp63 transcription factor revealed by a combination of gene expression profiling and reverse engineering. *Genome Research*, 18(6):939–948, Jun 2008. [URL]. [DOI].

P. Gao, A. Honkela, M. Rattray, and N. D. Lawrence. Gaussian process modelling of latent chemical species: Applications to inferring transcription factor activities. *Bioinformatics*, 24:i70–i75, 2008. [PDF]. [DOI].

A. Honkela, C. Girardot, E. H. Gustafson, Y.-H. Liu, E. E. M. Furlong, N. D. Lawrence, and M. Rattray. Model-based method for transcription factor target identification with limited data. *Proc. Natl. Acad. Sci. USA*, 107(17): 7793–7798, Apr 2010. [DOI].

References II

S. T. Roweis. EM algorithms for PCA and SPCA. In M. I. Jordan, M. J. Kearns, and S. A. Solla, editors, *Advances in Neural Information Processing Systems*, volume 10, pages 626–632, Cambridge, MA, 1998. MIT Press.

M. E. Tipping and C. M. Bishop. Probabilistic principal component analysis. *Journal of the Royal Statistical Society, B*, 6(3):611–622, 1999. [PDF]. [DOI].

R. P. Zinzen, C. Girardot, J. Gagneur, M. Braun, and E. E. M. Furlong. Combinatorial binding predicts spatio-temporal cis-regulatory activity. *Nature*, 462(7269):65–70, Nov 2009. [URL]. [DOI].