Approximations

Neil D. Lawrence

GPRS
19th–22nd January 2015

Outline

Regression

Bayesian Perspective

Gaussian Processes

Multiple Output Processes

Latent Force Models

Approximations

Dimensionality Reduction

Outline

Regression

Bayesian Perspective

Gaussian Processes

Multiple Output Processes

Latent Force Models

Approximations

Larger Datasets

Approximations in GPs

- Two main challenges:
  - Computational complexity and storage of exact inference $\mathcal{O}(n^3)$ and $\mathcal{O}(n^2)$ respectively.
  - Non Gaussian likelihoods making requisite integrals intractable.
- In this section we address these challenges.

Variational Compression

(Lawrence, 2007; Titsias, 2009)

- Complexity of standard GP:
  - $\mathcal{O}(n^3)$ in computation.
  - $\mathcal{O}(n^2)$ in storage.
- Via low rank representations of covariance:
  - $\mathcal{O}(nm^2)$ in computation.
  - $\mathcal{O}(nm)$ in storage.
- Where m is user chosen number of *inducing* variables. They give the rank of the resulting covariance.

Variational Compression

(Lawrence, 2007; Titsias, 2009)

- Complexity of standard GP:
  - $\mathcal{O}(n^3)$ in computation.
  - $\mathcal{O}(n^2)$ in storage.
- Via low rank representations of covariance:
  - $\mathcal{O}(nm^2)$ in computation.
  - $\mathcal{O}(nm)$ in storage.
- Where m is user chosen number of *inducing* variables. They give the rank of the resulting covariance.

Variational Compression

(Lawrence, 2007; Titsias, 2009)

- Complexity of standard GP:
  - $\mathcal{O}(n^3)$ in computation.
  - $\mathcal{O}(n^2)$ in storage.
- Via low rank representations of covariance:
  - $\mathcal{O}(nm^2)$ in computation.
  - $\mathcal{O}(nm)$ in storage.
- Where m is user chosen number of *inducing* variables. They give the rank of the resulting covariance.

Variational Compression

- Inducing variables are a compression of the real observations.
- They can live in space of $\mathbf{f}$ or a space that is related through a linear operator (Álvarez et al., 2010) — could be gradient or convolution.
- There are inducing variables associated with each set of hidden variables, $\mathbf{x}^i$.
- **Importantly** conditioning on inducing variables renders the likelihood independent across the data.
  - It turns out that this allows us to variationally handle uncertainty on the kernel (including the inputs to the kernel).
  - It also allows standard scaling approaches: stochastic variational inference Hensman et al. (2013), parallelization Gal et al. (2014) and work by Zhenwen Dai on GPUs to be applied: an *engineering* challenge?

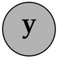
Inducing Variable Approximations

- Date back to (Williams and Seeger, 2001; Smola and Bartlett, 2001; Csató and Opper, 2002; Seeger et al., 2003; Snelson and Ghahramani, 2006). See Quiñonero Candela and Rasmussen (2005) for a review.
- We follow variational perspective of (Titsias, 2009).
- This is an augmented variable method, followed by a collapsed variational approximation (King and Lawrence, 2006; Hensman et al., 2012).

Augmented Variable Model: Not Wrong but Useful?

Augment standard model with a set of m new inducing variables, $\mathbf{u}$.

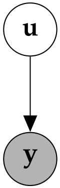
$$p(\mathbf{y}) = \int p(\mathbf{y}, \mathbf{u}) \mathrm{d}\mathbf{u}$$



Augmented Variable Model: Not Wrong but Useful?

Augment standard model with a set of m new inducing variables, $\mathbf{u}$.

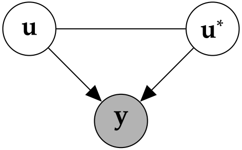
$$p(\mathbf{y}) = \int p(\mathbf{y}|\mathbf{u})p(\mathbf{u})\mathrm{d}\mathbf{u}$$



Augmented Variable Model: Not Wrong but Useful?

Important: Ensure inducing variables are *also* Kolmogorov consistent (we have m^* other inducing variables we are not *yet* using.)

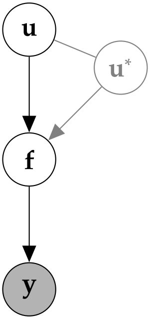
$$p(\mathbf{u}) = \int p(\mathbf{u}, \mathbf{u}^*)\mathrm{d}\mathbf{u}^*$$



Augmented Variable Model: Not Wrong but Useful?

Assume that relationship is through $\mathbf{f}$ (represents 'fundamentals'—push Kolmogorov consistency up to here).

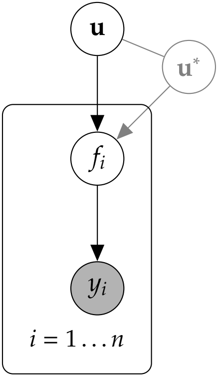
$$p(\mathbf{y}) = \int p(\mathbf{y}|\mathbf{f})p(\mathbf{f}|\mathbf{u})p(\mathbf{u})\mathrm{d}\mathbf{f}\mathrm{d}\mathbf{u}$$



Augmented Variable Model: Not Wrong but Useful?

Convenient to assume factorization (*doesn't* invalidate model—think delta function as worst case).

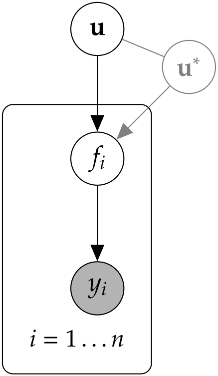
$$p(\mathbf{y}) = \int \prod_{i=1}^{n} p(y_i|f_i) p(\mathbf{f}|\mathbf{u}) p(\mathbf{u}) \mathrm{d}\mathbf{f} \mathrm{d}\mathbf{u}$$



Augmented Variable Model: Not Wrong but Useful?

Focus on integral over $\mathbf{f}$.

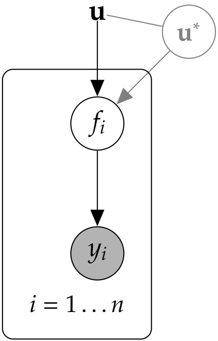
$$p(\mathbf{y}) = \int \int \prod_{i=1}^{n} p(y_i|f_i)p(\mathbf{f}|\mathbf{u})\mathrm{d}\mathbf{f}p(\mathbf{u})\mathrm{d}\mathbf{u}$$



Augmented Variable Model: Not Wrong but Useful?

Focus on integral over $\mathbf{f}$.

$$p(\mathbf{y}|\mathbf{u}) = \int \prod_{i=1}^{n} p(y_i|f_i)p(\mathbf{f}|\mathbf{u})\mathrm{d}\mathbf{f}$$



Variational Bound on $p(\mathbf{y}|\mathbf{u})$

$$\log p(\mathbf{y}|\mathbf{u}) = \log \int p(\mathbf{y}|\mathbf{f})p(\mathbf{f}|\mathbf{u})\mathrm{d}\mathbf{f}$$
$$= \int q(\mathbf{f}) \log \frac{p(\mathbf{y}|\mathbf{f})p(\mathbf{f}|\mathbf{u})}{q(\mathbf{f})}\mathrm{d}\mathbf{f} + \mathrm{KL}\left(q(\mathbf{f})\,\|\,p(\mathbf{f}|\mathbf{y},\mathbf{u})\right)$$

(Titsias, 2009)

- Example, set $q(\mathbf{f}) = p(\mathbf{f}|\mathbf{u})$,

$$\log p(\mathbf{y}|\mathbf{u}) \geq \log \int p(\mathbf{f}|\mathbf{u}) \log p(\mathbf{y}|\mathbf{f})\mathrm{d}\mathbf{f}.$$

$$p(\mathbf{y}|\mathbf{u}) \geq \exp \int p(\mathbf{f}|\mathbf{u}) \log p(\mathbf{y}|\mathbf{f})\mathrm{d}\mathbf{f}.$$

Variational Bound on $p(\mathbf{y}|\mathbf{u})$

$$\begin{aligned}\log p(\mathbf{y}|\mathbf{u}) &= \log \int p(\mathbf{y}|\mathbf{f})p(\mathbf{f}|\mathbf{u})\mathrm{d}\mathbf{f} \\ &= \int q(\mathbf{f}) \log \frac{p(\mathbf{y}|\mathbf{f})p(\mathbf{f}|\mathbf{u})}{q(\mathbf{f})}\mathrm{d}\mathbf{f} + \mathrm{KL}\left(q(\mathbf{f})\,\|\,p(\mathbf{f}|\mathbf{y},\mathbf{u})\right)\end{aligned}$$

(Titsias, 2009)

- Example, set $q(\mathbf{f}) = p(\mathbf{f}|\mathbf{u})$,

$$\log p(\mathbf{y}|\mathbf{u}) \geq \log \int p(\mathbf{f}|\mathbf{u}) \log p(\mathbf{y}|\mathbf{f})\mathrm{d}\mathbf{f}.$$

$$p(\mathbf{y}|\mathbf{u}) \geq \exp \int p(\mathbf{f}|\mathbf{u}) \log p(\mathbf{y}|\mathbf{f})\mathrm{d}\mathbf{f}.$$

Optimal Compression in Inducing Variables

- Maximizing lower bound minimizes the KL divergence (information gain):

$$\mathrm{KL}\left(p(\mathbf{f}|\mathbf{u})\,\|\,p(\mathbf{f}|\mathbf{y},\mathbf{u})\right) = \int p(\mathbf{f}|\mathbf{u}) \log \frac{p(\mathbf{f}|\mathbf{u})}{p(\mathbf{f}|\mathbf{y},\mathbf{u})} \mathrm{d}\mathbf{u}$$

- This is minimized when the information stored about $\mathbf{y}$ is stored already in $\mathbf{u}$.
- The bound seeks an *optimal compression* from the *information gain* perspective.
- If $\mathbf{u} = \mathbf{f}$ bound is exact ($\mathbf{f}$ d-separates $\mathbf{y}$ from $\mathbf{u}$).

Choice of Inducing Variables

- Optimizing the bound directly not always practical.
- Free to choose whatever heuristics for the inducing variables.
- Can quantify which heuristics perform better through checking lower bound.

Factorizing Likelihoods

- If the likelihood, $p(\mathbf{y}|\mathbf{f})$, factorizes

$$p(\mathbf{y}|\mathbf{u}) \geq \exp \int p(\mathbf{f}|\mathbf{u}) \log \prod_{i=1}^{n} p(y_i|f_i) \mathrm{d}\mathbf{f}.$$

- Then the bound factorizes.
- Now need a choice of distributions for $\mathbf{f}$ and $\mathbf{y}|\mathbf{f}$ ...

Factorizing Likelihoods

- If the likelihood, $p(\mathbf{y}|\mathbf{f})$, factorizes

$$p(\mathbf{y}|\mathbf{u}) \geq \exp \int p(\mathbf{f}|\mathbf{u}) \log \prod_{i=1}^{n} p(y_i|f_i) \mathrm{d}\mathbf{f}.$$

- Then the bound factorizes.
- Now need a choice of distributions for $\mathbf{f}$ and $\mathbf{y}|\mathbf{f}$ ...

Factorizing Likelihoods

- If the likelihood, $p(\mathbf{y}|\mathbf{f})$, factorizes

$$p(\mathbf{y}|\mathbf{u}) \geq \exp \int p(\mathbf{f}|\mathbf{u}) \sum_{i=1}^{n} \log p(y_i|f_i) \mathrm{d}\mathbf{f}.$$

- Then the bound factorizes.
- Now need a choice of distributions for $\mathbf{f}$ and $\mathbf{y}|\mathbf{f}$ ...

Factorizing Likelihoods

- If the likelihood, $p(\mathbf{y}|\mathbf{f})$, factorizes

$$p(\mathbf{y}|\mathbf{u}) \geq \exp \int p(\mathbf{f}|\mathbf{u}) \sum_{i=1}^{n} \log p(y_i|f_i) \mathrm{d}\mathbf{f}.$$

- Then the bound factorizes.
- Now need a choice of distributions for $\mathbf{f}$ and $\mathbf{y}|\mathbf{f}$ ...

Factorizing Likelihoods

- If the likelihood, $p(\mathbf{y}|\mathbf{f})$, factorizes

$$p(\mathbf{y}|\mathbf{u}) \geq \exp \sum_{i=1}^{n} \int p(f_i|\mathbf{u}) \log p(y_i|f_i) \mathrm{d}\mathbf{f}.$$

- Then the bound factorizes.
- Now need a choice of distributions for $\mathbf{f}$ and $\mathbf{y}|\mathbf{f}$ ...

Factorizing Likelihoods

- If the likelihood, $p(\mathbf{y}|\mathbf{f})$, factorizes

$$p(\mathbf{y}|\mathbf{u}) \geq \exp \sum_{i=1}^{n} \int p(f_i|\mathbf{u}) \log p(y_i|f_i) \mathrm{d}\mathbf{f}.$$

- Then the bound factorizes.
- Now need a choice of distributions for $\mathbf{f}$ and $\mathbf{y}|\mathbf{f}$ ...

Factorizing Likelihoods

- If the likelihood, $p(\mathbf{y}|\mathbf{f})$, factorizes

$$p(\mathbf{y}|\mathbf{u}) \geq \prod_{i=1}^{n} \exp \int p(f_i|\mathbf{u}) \log p(y_i|f_i) \mathrm{d}\mathbf{f}.$$

- Then the bound factorizes.
- Now need a choice of distributions for $\mathbf{f}$ and $\mathbf{y}|\mathbf{f}$ ...

Factorizing Likelihoods

- If the likelihood, $p(\mathbf{y}|\mathbf{f})$, factorizes

$$p(\mathbf{y}|\mathbf{u}) \geq \prod_{i=1}^{n} \exp \int p(f_i|\mathbf{u}) \log p(y_i|f_i) \mathrm{d}\mathbf{f}.$$

- Then the bound factorizes.
- Now need a choice of distributions for $\mathbf{f}$ and $\mathbf{y}|\mathbf{f}$ ...

Factorizing Likelihoods

- If the likelihood, $p(\mathbf{y}|\mathbf{f})$, factorizes

$$p(\mathbf{y}|\mathbf{u}) \geq \prod_{i=1}^{n} \exp \left\langle \log p(y_i|f_i) \right\rangle_{p(f_i|\mathbf{u})}$$

- Then the bound factorizes.
- Now need a choice of distributions for $\mathbf{f}$ and $\mathbf{y}|\mathbf{f}$ ...

Factorizing Likelihoods

- If the likelihood, $p(\mathbf{y}|\mathbf{f})$, factorizes

$$p(\mathbf{y}|\mathbf{u}) \geq \prod_{i=1}^{n} \exp \left\langle \log p(y_i|f_i) \right\rangle_{p(f_i|\mathbf{u})}$$

- Then the bound factorizes.
- Now need a choice of distributions for $\mathbf{f}$ and $\mathbf{y}|\mathbf{f}$ ...

Gaussian $p(y_i|f_i)$

For Gaussian likelihoods:

$$\left\langle \log p(y_i|f_i) \right\rangle_{p(f_i|\mathbf{u})} = -\frac{1}{2}\log 2\pi\sigma^2 - \frac{1}{2\sigma^2}\left(y_i - \left\langle f_i \right\rangle\right)^2 - \frac{1}{2\sigma^2}\left(\left\langle f_i^2 \right\rangle - \left\langle f_i \right\rangle^2\right)$$

Gaussian $p(y_i|f_i)$

For Gaussian likelihoods:

$$\langle \log p(y_i|f_i)\rangle_{p(f_i|\mathbf{u})} = -\frac{1}{2}\log 2\pi\sigma^2 - \frac{1}{2\sigma^2}\left(y_i - \langle f_i\rangle\right)^2 - \frac{1}{2\sigma^2}\left(\left\langle f_i^2\right\rangle - \langle f_i\rangle^2\right)$$

Implying:

$$p(y_i|\mathbf{u}) \geq \exp\left\langle \log c_i\right\rangle \mathcal{N}\left(y_i|\left\langle f_i\right\rangle, \sigma^2\right)$$

Gaussian Process Over $\mathbf{f}$ and $\mathbf{u}$

Define:

$$q_{i,i} = \mathrm{var}_{p(f_i|\mathbf{u})}\left(f_i\right) = \left\langle f_i^2 \right\rangle_{p(f_i|\mathbf{u})} - \left\langle f_i \right\rangle_{p(f_i|\mathbf{u})}^2$$

We can write:

$$c_i = \exp\left(-\frac{q_{i,i}}{2\sigma^2}\right)$$

If joint distribution of $p(\mathbf{f}, \mathbf{u})$ is Gaussian then:

$$q_{i,i} = k_{i,i} - \mathbf{k}_{i,\mathbf{u}}^\top \mathbf{K}_{\mathbf{u},\mathbf{u}}^{-1} \mathbf{k}_{i,\mathbf{u}}$$

c_i is not a function of $\mathbf{u}$ but *is* a function of $\mathbf{X}_\mathbf{u}$.

Lower Bound on Likelihood

Substitute variational bound into marginal likelihood:

$$p(\mathbf{y}) \geq \prod_{i=1}^{n} c_i \int \mathcal{N}\left(\mathbf{y}| \left\langle \mathbf{f} \right\rangle, \sigma^2 \mathbf{I}\right) p(\mathbf{u}) \mathrm{d}\mathbf{u}$$

Note that:

$$\left\langle \mathbf{f} \right\rangle_{p(\mathbf{f}|\mathbf{u})} = \mathbf{K}_{\mathbf{f},\mathbf{u}} \mathbf{K}_{\mathbf{u},\mathbf{u}}^{-1} \mathbf{u}$$

is *linearly* dependent on $\mathbf{u}$.

Deterministic Training Conditional

Making the marginalization of $\mathbf{u}$ straightforward. In the Gaussian case:

$$p(\mathbf{u}) = \mathcal{N}\left(\mathbf{u}|\mathbf{0}, \mathbf{K}_{\mathbf{u},\mathbf{u}}\right)$$

$$\int p(\mathbf{y}|\mathbf{u})p(\mathbf{u})\mathrm{d}\mathbf{u} \geq \prod_{i=1}^{n} c_i \int \mathcal{N}\left(\mathbf{y}|\mathbf{K}_{\mathbf{f},\mathbf{u}}\mathbf{K}_{\mathbf{u},\mathbf{u}}^{-1}\mathbf{u}, \sigma^2\right) \mathcal{N}\left(\mathbf{u}|\mathbf{0}, \mathbf{K}_{\mathbf{u},\mathbf{u}}\right) \mathrm{d}\mathbf{u}$$

Deterministic Training Conditional

Making the marginalization of $\mathbf{u}$ straightforward. In the Gaussian case:

$$p(\mathbf{u}) = \mathcal{N}\left(\mathbf{u}|\mathbf{0}, \mathbf{K}_{\mathbf{u},\mathbf{u}}\right)$$

$$\int p(\mathbf{y}|\mathbf{u})p(\mathbf{u})\mathrm{d}\mathbf{u} \geq \prod_{i=1}^{n} c_i \mathcal{N}\left(\mathbf{y}|\mathbf{0}, \sigma^2\mathbf{I} + \mathbf{K}_{\mathbf{f},\mathbf{u}}\mathbf{K}_{\mathbf{u},\mathbf{u}}^{-1}\mathbf{K}_{\mathbf{u},\mathbf{f}}\right)$$

Deterministic Training Conditional

Making the marginalization of $\mathbf{u}$ straightforward. In the Gaussian case:

$$p(\mathbf{u}) = \mathcal{N}\left(\mathbf{u}|\mathbf{0}, \mathbf{K}_{\mathbf{u},\mathbf{u}}\right)$$

$$\int p(\mathbf{y}|\mathbf{u})p(\mathbf{u})\mathrm{d}\mathbf{u} \geq \prod_{i=1}^{n} c_i \mathcal{N}\left(\mathbf{y}|\mathbf{0}, \sigma^2\mathbf{I} + \mathbf{K}_{\mathbf{f},\mathbf{u}}\mathbf{K}_{\mathbf{u},\mathbf{u}}^{-1}\mathbf{K}_{\mathbf{u},\mathbf{f}}\right)$$

Maximize log of the bound to find covariance function parameters,

$$L \geq \sum_{i=1}^{n} \log c_i + \log \mathcal{N}\left(\mathbf{y}|\mathbf{0}, \sigma^2\mathbf{I} + \mathbf{K}_{\mathbf{f},\mathbf{u}}\mathbf{K}_{\mathbf{u},\mathbf{u}}^{-1}\mathbf{K}_{\mathbf{u},\mathbf{f},}\right)$$

Deterministic Training Conditional

Making the marginalization of $\mathbf{u}$ straightforward. In the Gaussian case:

$$p(\mathbf{u}) = \mathcal{N}\left(\mathbf{u}|\mathbf{0}, \mathbf{K}_{\mathbf{u},\mathbf{u}}\right)$$

$$\int p(\mathbf{y}|\mathbf{u})p(\mathbf{u})\mathrm{d}\mathbf{u} \geq \prod_{i=1}^{n} c_i \mathcal{N}\left(\mathbf{y}|\mathbf{0}, \sigma^2\mathbf{I} + \mathbf{K}_{\mathbf{f},\mathbf{u}}\mathbf{K}_{\mathbf{u},\mathbf{u}}^{-1}\mathbf{K}_{\mathbf{u},\mathbf{f}}\right)$$

Maximize log of the bound to find covariance function parameters,

$$L \geq \sum_{i=1}^{n} \log c_i + \log \mathcal{N}\left(\mathbf{y}|\mathbf{0}, \sigma^2\mathbf{I} + \mathbf{K}_{\mathbf{f},\mathbf{u}}\mathbf{K}_{\mathbf{u},\mathbf{u}}^{-1}\mathbf{K}_{\mathbf{u},\mathbf{f},}\right)$$

Deterministic Training Conditional

Making the marginalization of $\mathbf{u}$ straightforward. In the Gaussian case:

$$p(\mathbf{u}) = \mathcal{N}\left(\mathbf{u}|\mathbf{0}, \mathbf{K}_{\mathbf{u},\mathbf{u}}\right)$$

$$\int p(\mathbf{y}|\mathbf{u})p(\mathbf{u})\mathrm{d}\mathbf{u} \geq \prod_{i=1}^{n} c_i \mathcal{N}\left(\mathbf{y}|\mathbf{0}, \sigma^2\mathbf{I} + \mathbf{K}_{\mathbf{f},\mathbf{u}}\mathbf{K}_{\mathbf{u},\mathbf{u}}^{-1}\mathbf{K}_{\mathbf{u},\mathbf{f}}\right)$$

Maximize log of the bound to find covariance function parameters,

$$L \approx \log \mathcal{N}\left(\mathbf{y}|\mathbf{0}, \sigma^2\mathbf{I} + \mathbf{K}_{\mathbf{f},\mathbf{u}}\mathbf{K}_{\mathbf{u},\mathbf{u}}^{-1}\mathbf{K}_{\mathbf{u},\mathbf{f},}\right)$$

- If the bound is normalized, the c_i terms are removed.
- This results in the projected process approximation (Rasmussen and Williams, 2006) or DTC (Quiñonero Candela and Rasmussen, 2005). Proposed by (Smola and Bartlett, 2001; Seeger et al., 2003; Csató and Opper, 2002; Csató, 2002).

Deterministic Training Conditional

Making the marginalization of $\mathbf{u}$ straightforward. In the Gaussian case:

$$p(\mathbf{u}) = \mathcal{N}\left(\mathbf{u}|\mathbf{0}, \mathbf{K}_{\mathbf{u},\mathbf{u}}\right)$$

$$\int p(\mathbf{y}|\mathbf{u})p(\mathbf{u})\mathrm{d}\mathbf{u} \geq \prod_{i=1}^{n} c_i \mathcal{N}\left(\mathbf{y}|\mathbf{0}, \sigma^2\mathbf{I} + \mathbf{K}_{\mathbf{f},\mathbf{u}}\mathbf{K}_{\mathbf{u},\mathbf{u}}^{-1}\mathbf{K}_{\mathbf{u},\mathbf{f}}\right)$$

Maximize log of the bound to find covariance function parameters,

- If the bound is normalized, the c_i terms are removed.
- This results in the projected process approximation (Rasmussen and Williams, 2006) or DTC (Quiñonero Candela and Rasmussen, 2005). Proposed by (Smola and Bartlett, 2001; Seeger et al., 2003; Csató and Opper, 2002; Csató, 2002).

Fully Independent Training Conditional

Define c_i' to be

$$c_i' = c_i \exp\left(\frac{\mathbf{y}_i^2 q_{i,i}}{2}\right) = \exp\left(\frac{q_{i,i}(\mathbf{y}_i^2 - \sigma^{-2})}{2}\right)$$

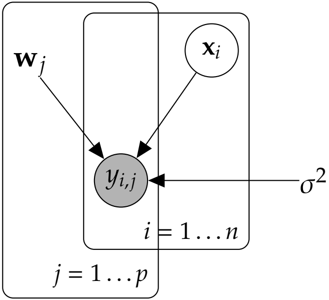
Then rewrite the bound:

$$\sum_{i=1}^{n} \log c_i' + \log \mathcal{N}\left(\mathbf{y}|\mathbf{0}, \sigma^2\mathbf{I} + \mathrm{diag}\left(\mathbf{Q}_{\mathbf{f},\mathbf{f}}\right) + \mathbf{K}_{\mathbf{f},\mathbf{u}}\mathbf{K}_{\mathbf{u},\mathbf{u}}^{-1}\mathbf{K}_{\mathbf{u},\mathbf{f}}\right)$$

where

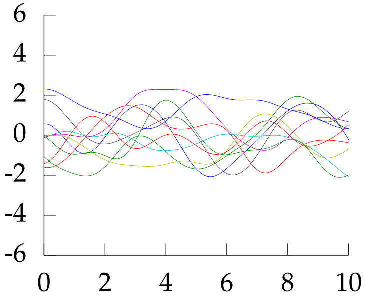
$$\mathbf{Q}_{\mathbf{f},\mathbf{f}} = \mathrm{cov}\left(\mathbf{f}\mathbf{f}^\top\right)_{p(\mathbf{f}|\mathbf{u})} = \mathbf{K}_{\mathbf{f},\mathbf{f}} - \mathbf{K}_{\mathbf{f},\mathbf{u}}\mathbf{K}_{\mathbf{u},\mathbf{u}}^{-1}\mathbf{K}_{\mathbf{u},\mathbf{f}}$$

In FITC the $\log c_i'$ terms could be negative or positive.

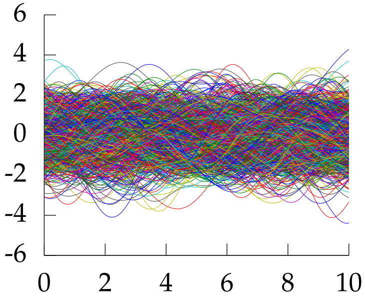


$\mathbf{w}_j$
$\mathbf{x}_i$
$y_{i,j}$
σ^2
$i = 1 \dots n$
$j = 1 \dots p$

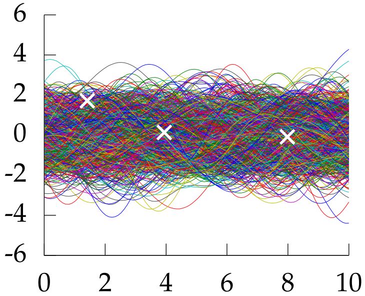
Gaussian Processes: Extremely Short Overview



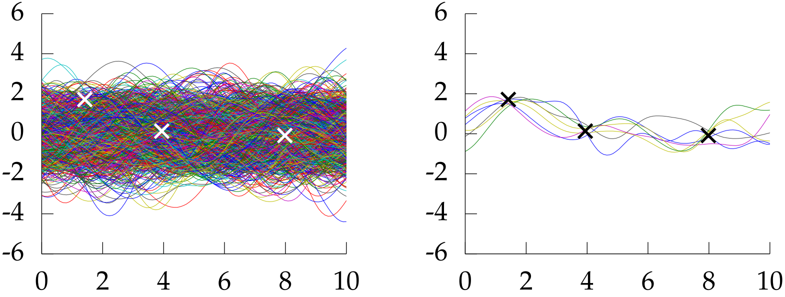
Gaussian Processes: Extremely Short Overview



Gaussian Processes: Extremely Short Overview



Gaussian Processes: Extremely Short Overview



GP Regression

Analytical tractability of the posterior distribution is assured:

- Gaussian prior:
$$\mathbf{f} \sim \mathcal{N}\left(\mathbf{0}, \mathbf{K}_{\mathbf{ff}}\right)$$
- Gaussian likelihood:
$$\prod_{i=1}^{n} p(y_i|f_i) = \mathcal{N}\left(\mathbf{y}|\mathbf{f}, \sigma_i^2\mathbf{I}\right)$$
- Gaussian posterior:
$$p(\mathbf{f}|\mathbf{y}) \propto \mathcal{N}\left(\mathbf{f}|\mathbf{0}, \mathbf{K}_{\mathbf{ff}}\right)\mathcal{N}\left(\mathbf{y}|\mathbf{f}, \sigma_i^2\mathbf{I}\right)$$

Bernoulli Distribution

- A mathematical switch allows us to write a probability table as a function.

$$P(Y = 1) = \pi$$

$$P(Y = 0) = (1 - \pi)$$

- Write as a function

$$P(Y = y) = \pi^y (1 - \pi)^{1-y}$$

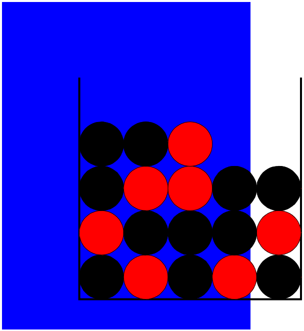
- Can think of this construction as a “mathematical switch”. Known as the Bernoulli distribution.
- Widely used in classification algorithms: π parameter is made to be dependent on “inputs”.

Binomial Distribution

- Generalization of Bernoulli to multiple trials.
- Jakob Bernoulli: black and red balls in an urn. Proportion of red is π.
- Sample with replacement. Binomial gives the distribution of number of reds, y, from S extractions

$$P(y|\pi, S) = \frac{S!}{y!(S-y)!}\pi^{y}(1-\pi)^{(S-y)}$$

- Mean is given by $S\pi$ and variance $S\pi(1-\pi)$.



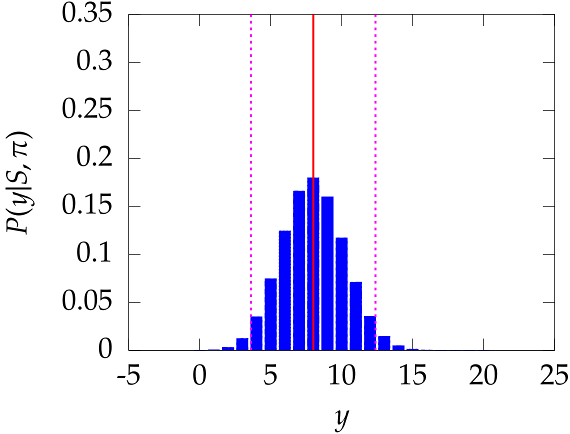


Figure : The binomial distribution for $\pi = 0.4$ and $S = 20$. Mean is shown as red line, 2 standard deviations are magenta.

The Gamma Density

- Density over positive real values.

$$
\begin{aligned}
p(y|a, b) &= \frac{b^a}{\Gamma(a)} y^{a-1} \exp\left(-by\right) \\
&= \mathcal{G}\left(y|\mu, \sigma^2\right)
\end{aligned}
$$

- Mean is $\frac{a}{b}$ and variance is $\frac{a}{b^2}$.
- Also available in multivariate as the Wishart (positive definite matrices).

Gamma PDF I

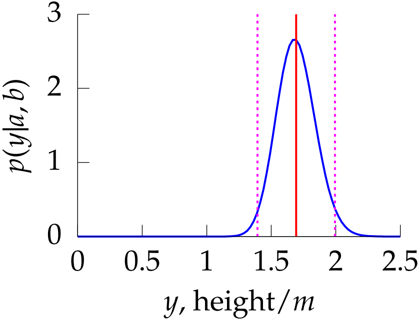


Figure : The Gamma PDF with $a = 127$ and $b = 75$. Here it represents the heights of a population of students and constrains them positive.

Gamma PDF I

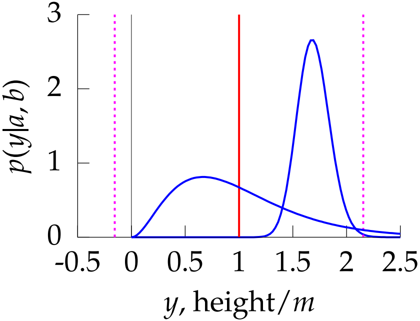


Figure : The Gamma PDF with $a = 127$ and $b = 75$ alongside a Gamma PDF with $a = 3$ and $b = 3$.

Categorical Distribution

Multiple outcomes, example: die roll.

die role	probability	$\mathbf{y}$
1	π_1	[1 0 0 0 0 0]
2	π_2	[0 1 0 0 0 0]
3	π_3	[0 0 1 0 0 0]
4	π_4	[0 0 0 1 0 0]
5	π_5	[0 0 0 0 1 0]
6	π_6	[0 0 0 0 0 1]

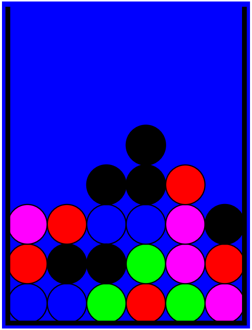
$$P(\mathbf{y}) = \prod_{i=1}^{k} \pi_i^{y_i}$$

Multinomial Distribution

- Generalization of categorical to multiple trials.
- Generalization of binomial to multiple outcomes. Proportion of each colour ball is now π_i.
- Sample with replacement. Multinomial gives the distribution of number of each of k different balls, y, from S extractions

$$P(y|\pi, S) = \frac{S!}{\prod_{i=1}^{k} y_i!} \prod_{i=1}^{k} \pi_i^{y_i}$$

- Mean for each colour is given by $S\pi_i$ and variance $S\pi_i(1 - \pi_i)$.



Distributions as Functions

- Probability distribution with a simple table can be limiting.
- The Poisson Distribution — a distribution a a function
- First published by **Siméon Denis Poisson** (1781-1840) in 1837.
- Defined over the space of all non-negative integers.
- This set is countably infinite: impossible to summarise in a table!
- The Poisson distribution is therefore defined as

$$P\left(y|\mu\right) = \frac{\mu^y}{y!}\exp\left(-\mu\right). \tag{4}$$

where y is any integer from 0 to ∞, and μ is a parameter of the distribution.

A Poisson with $\mu = 2$

- To work out the probability of y in a Poisson $\mu = 2$ we can start filling a table.
- The values in a table are computed from (4)

y	0	1	2	...
$P(y)$	0.135	0.271	0.271	...

Table : Some values for the Poisson distribution with $\mu = 2$.

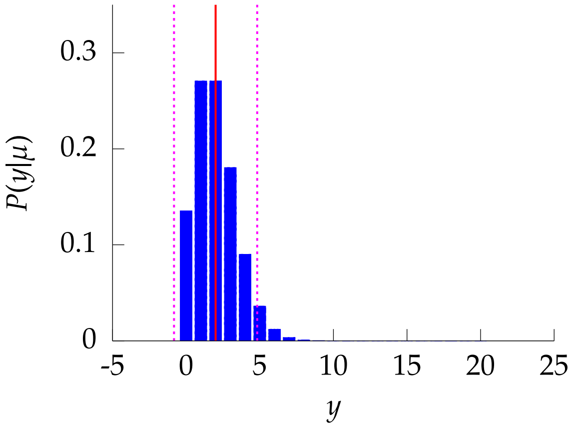


Figure : The Poisson distribution for $\mu = 2$. Mean is given by μ (red line), standard deviation is given by $\sqrt{\mu}$ (magenta lines show 2 standard deviations).

Gaussian Noise

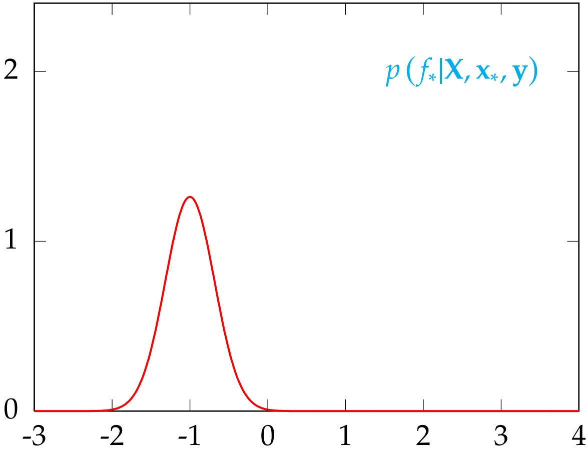


Figure : Inclusion of a data point with Gaussian noise.

Gaussian Noise

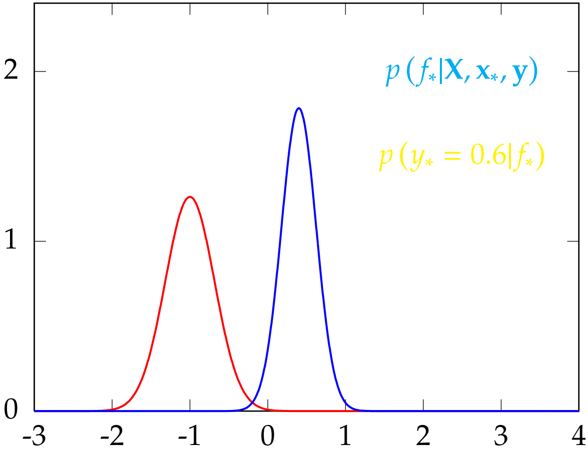


Figure : Inclusion of a data point with Gaussian noise.

Gaussian Noise

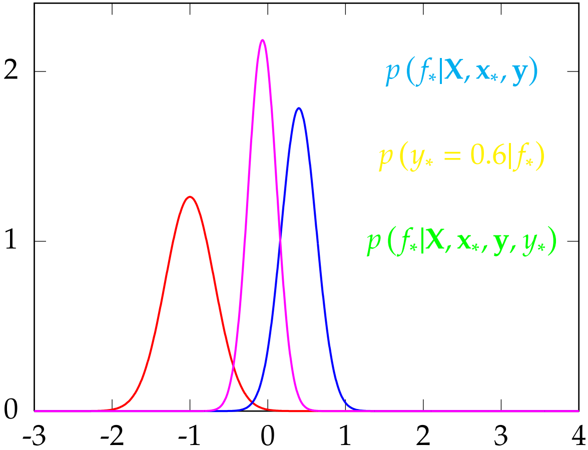


Figure : Inclusion of a data point with Gaussian noise.

Classification Noise Model

Probit Noise Model

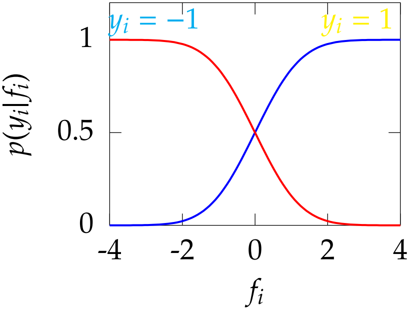


Figure : The probit model (classification). The plot shows $p\left(y_i|f_i\right)$ for different values of y_i. For $y_i = 1$ we have
$p\left(y_i|f_i\right) = \phi\left(f_i\right) = \int_{-\infty}^{f_i} \mathcal{N}\left(z|0, 1\right) \mathrm{d}z$.

Ordinal Noise Model

Ordered Categories

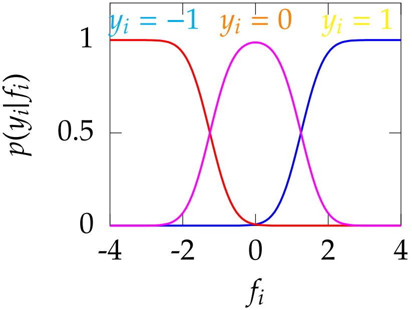


Figure : The ordered categorical noise model (ordinal regression). The plot shows $p(y_i|f_i)$ for different values of y_i. Here we have assumed three categories.

Null Category Noise Model

Classification with a Missing Category

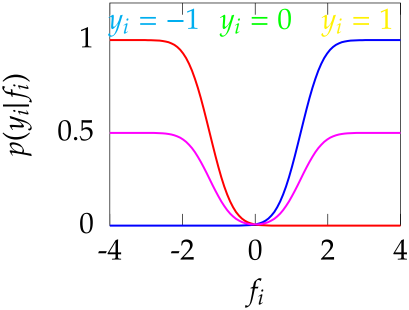


Figure : The null category noise model (semi-supervised learning). The plot shows $p(y_i|f_i)$ for different values of y_i. Here we have assumed three categories.

Non-linear Response Functions

- Non Gaussian likelihood:

$$p(y_i|f_i) = \Phi(f_i)$$

- Exact computation of the posterior is no longer possible analytically.

$$p(\mathbf{f}|\mathbf{y}) = \frac{p(\mathbf{f}) \prod_{i=1}^{n} p(y_i|f_i)}{\int p(\mathbf{f}) \prod_{i=1}^{n} p(y_i|f_i) \mathrm{d}\mathbf{f}}$$

Link Functions

- Take the output of our function, $f(\cdot)$ use as:
  - Success probability in binomial distribution.
  - Rate function in Poisson likelihood.
  - shape parameter of Gamma distribution.
- Problem: $f(\cdot)$ defined over real line.
- Needs to be squashed down to 0-1 or constrained positive.

Link Functions

- Log link function, model the log rate.

$$\log \lambda(\mathbf{x}) = f(\mathbf{x})$$

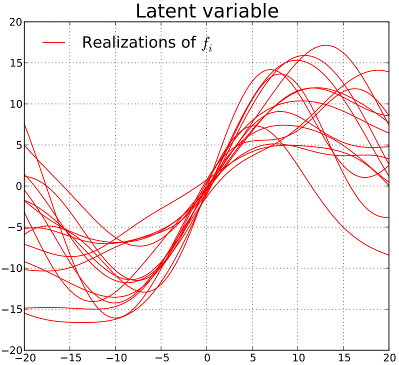
- Logit link function, model the log odds.

$$\frac{\log \pi(\mathbf{x})}{\log(1 - \pi(\mathbf{x}))} = f(\mathbf{x})$$

Generative Model

- From a generative perspective we often naturally think of the inverse link:
$$\lambda(\mathbf{x}) = \exp(f(\mathbf{x}))$$
$$\pi(\mathbf{x}) = \frac{1}{1 + \exp(-f(\mathbf{x}))}$$
- Can make some assumptions of the link function clearer. For example log additive link function:
$$\log \lambda(\mathbf{x}) = f_1(\mathbf{x}) + f_2(\mathbf{x})$$
is a product of functions:
$$\lambda(\mathbf{x}) = \exp(f_1(\mathbf{x})) \exp(f_2(\mathbf{x}))$$

Example: Logit/Probit Link Function



Laplace Approximation

- Second order Taylor expansion at mode of log likelihood.
- First suggested by Laplace for his English dice example.
- How Laplace independently (of de Moivre) reinvented the Gaussian density.

Laplace Approximation

$$\log p(\mathbf{f}|\mathbf{y}) = \log p(\mathbf{y}|\mathbf{f}) + \log p(\mathbf{f}) + \text{const}$$

$$\log p(\mathbf{f}|\mathbf{y}) = \log p(\mathbf{y}|\mathbf{f}) - \frac{1}{2}\mathbf{f}^\top \mathbf{K}_{\mathbf{ff}}^{-1}\mathbf{f}$$

- Find MAP estimate $\hat{\mathbf{f}}$. This is mean of Gaussian approximation.
- Find Hessian of this system.
- Covariance of approximation is $-\mathbf{H}^{-1}$.

$$\mathbf{H} = \left(\frac{\mathrm{d}^2 \log p(\mathbf{y}|\mathbf{f})}{\mathrm{d}f_i \mathrm{d}f_j}\right)_{ij} - \mathbf{K}_{\mathbf{ff}}^{-1}$$

Expectation Propagation: General Case

- Exact (intractable) posterior:

$$p(\mathbf{f}|\mathbf{y}) = \frac{p(\mathbf{f}) \prod_{i=1}^{n} p(y_i|f_i)}{\int p(\mathbf{f}) \prod_{i=1}^{n} p(y_i|f_i) \mathrm{d}\mathbf{f}}$$

- EP posterior approximation:

$$q(\mathbf{f}|\mathbf{y}) = \frac{\prod_{i=1}^{K} t_i(f_i)}{Z_{EP}}$$

Expectation Propagation: Gaussian Approximation

Consider the special case:

$$p(y_i|f_i) \approx t_i(f_i) = Z_i \mathcal{N}\left(\tilde{\mu}_i|f_i, \tilde{\sigma}_i^2\right)$$

Here Z_i is a scaling factor so t_i is unnormalized.

If

$$p(\mathbf{f}) \sim \mathcal{N}\left(\mathbf{0}, \mathbf{K}_{\mathbf{f},\mathbf{f}}\right).$$

No approximation needed.

EP Posterior Approximation

$$q(\mathbf{f}|\mathbf{y}) = \frac{\prod_{i=1}^{n} t(f_i)p(\mathbf{f})}{Z_{EP}} = \mathcal{N}\left(\mathbf{f}|\boldsymbol{\mu}, \boldsymbol{\Sigma}\right)$$

Site functions provide "fake Gaussian observations" with target value $\hat{\mu}_i$ and observation variance $\hat{\sigma}_i^2$.

$$Z_{EP} = \prod_{i=1}^{n} Z_i \int \prod_{i=1}^{n} \mathcal{N}\left(\hat{\mu}_i | f_i, \hat{\sigma}_i^2\right) p(\mathbf{f}) \mathrm{d}\mathbf{f}$$

EP Posterior Approximation

$$q(\mathbf{f}|\mathbf{y}) = \frac{\prod_{i=1}^{n} Z_i \mathcal{N}\left(\hat{\mu}_i | f_i, \hat{\sigma}_i^2\right) p(\mathbf{f})}{Z_{EP}} = \mathcal{N}\left(\mathbf{f}|\boldsymbol{\mu}, \boldsymbol{\Sigma}\right)$$

Site functions provide "fake Gaussian observations" with target value $\hat{\mu}_i$ and observation variance $\hat{\sigma}_i^2$.

$$Z_{EP} = \prod_{i=1}^{n} Z_i \int \prod_{i=1}^{n} \mathcal{N}\left(\hat{\mu}_i | f_i, \hat{\sigma}_i^2\right) p(\mathbf{f}) \mathrm{d}\mathbf{f}$$

Site approximations

- Given initial site approximations: $t_j(f_j)$ for $j \neq i$.
- Need to set

$$t_i(f_i) \approx p(y_i|f_i)$$

$$p(y_i|f_i)p(\mathbf{f}) \prod_{j \neq i} t_j(f_j) \approx p(\mathbf{f}) \prod_{j=1}^{n} t_j(f_j)$$

$$p(y_i|f_i) \int p(\mathbf{f}) \prod_{j \neq i} t_j(f_j) \mathrm{d} f_{j \neq i} \approx \int p(\mathbf{f}) \prod_{j=1}^{n} t_j(f_j) \mathrm{d} f_{j \neq i}$$

$$p(y_i|f_i) q_{\setminus i}(f_i) \approx \mathcal{N}\left(f_i|\hat{\mu}_i, \hat{\sigma}_i^2\right) \hat{Z}_i$$

Cavity Distribution

$$q_{\setminus i}(f_i) = \frac{\prod_{j \neq i} t(f_j) p(\mathbf{f})}{\int \prod_{j \neq i} t(f_j) p(\mathbf{f})} \mathrm{d}\mathbf{f}$$

Tilted Distribution

$$\hat{p}_i(f_i|y_i) = \frac{p(y_i|f_i)q_{\backslash i}(f_i)}{\hat{Z}}$$

where

$$\hat{Z}_i = \int p(y_i|f_i)q_{\backslash i}(f_i)\mathrm{d}f_i$$

Minimization of the KL divergence

$$\hat{\mu}_i, \hat{\sigma}_i = \text{argmin}_{\hat{\mu}_i, \hat{\sigma}_i} \text{KL}\left(\frac{p(y_i|f_i)q_{\backslash i}(f_i)}{\hat{Z}} \,\|\, \mathcal{N}\left(f_i|\hat{\mu}_i, \hat{\sigma}_i^2\right)\right)$$

This is the KL between *tilted distribution* and *marginal of approximation*.
Since the approximation is Gaussian, KL is minimal when:

- $\hat{\mu}_i = \langle f_i \rangle_{p(y_i|f_i)q_{\backslash i}(f_i)}$
- $\hat{\sigma}_i^2 = \langle f_i \rangle^2_{p(y_i|f_i)q_{\backslash i}(f_i)} - \tilde{\mu}_i^2$

Scale of Site Approximation

- Since the approximation is un-normalized, we set scale as follows:

$$\hat{Z}_i = \int p(y_i|f_i)q_{\setminus i}(f_i)\mathrm{d}f_i$$

Classification Noise Model

Probit Noise Model

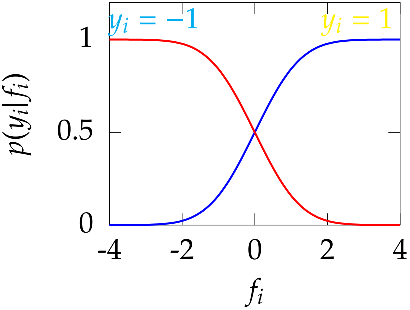


Figure : The probit model (classification). The plot shows $p(y_i|f_i)$ for different values of y_i. For $y_i = 1$ we have
$p(y_i|f_i) = \phi(f_i) = \int_{-\infty}^{f_i} \mathcal{N}(z|0,1)\,\mathrm{d}z$.

Classification

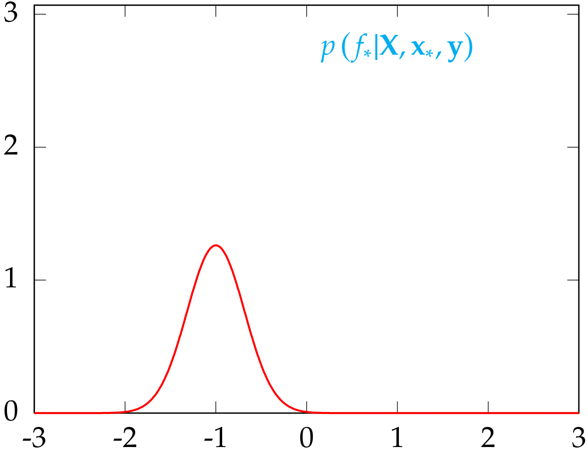


Figure : An EP style update with a classification noise model.

Classification

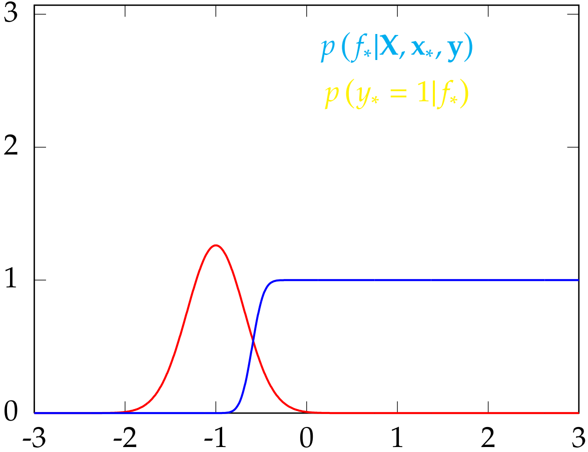


Figure : An EP style update with a classification noise model.

Classification

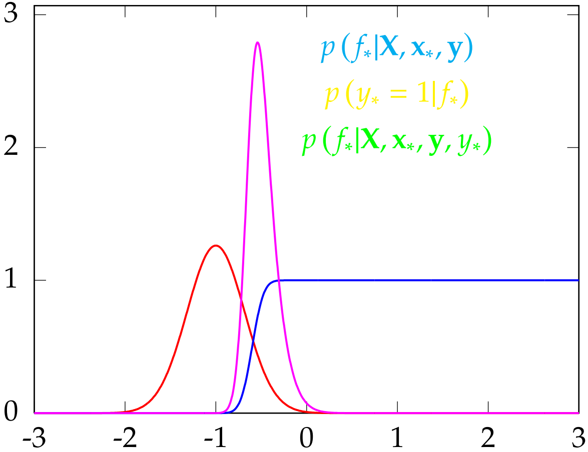


Figure : An EP style update with a classification noise model.

Classification

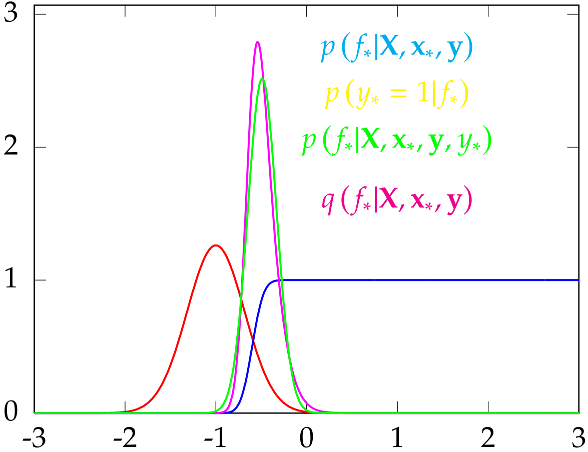


Figure : An EP style update with a classification noise model.

Ordered Categories

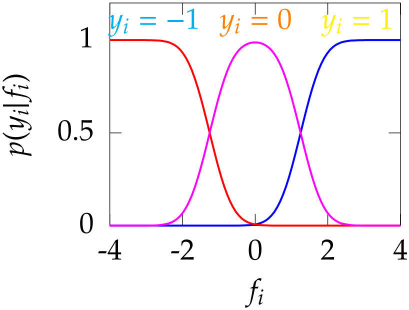


Figure : The ordered categorical noise model (ordinal regression). The plot shows $p(y_i|f_i)$ for different values of y_i. Here we have assumed three categories.

Ordinal Regression

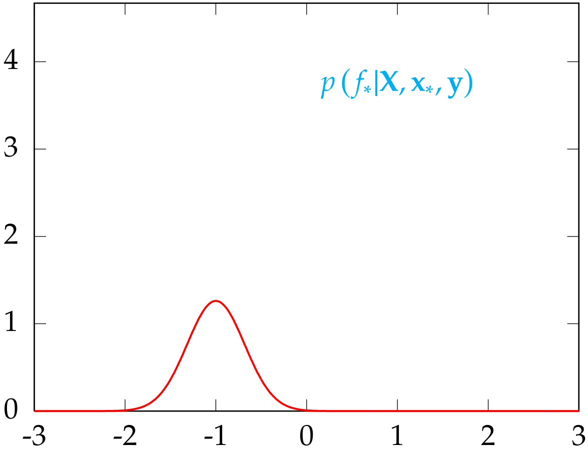


Figure : An EP style update with an ordered category noise model.

Ordinal Regression

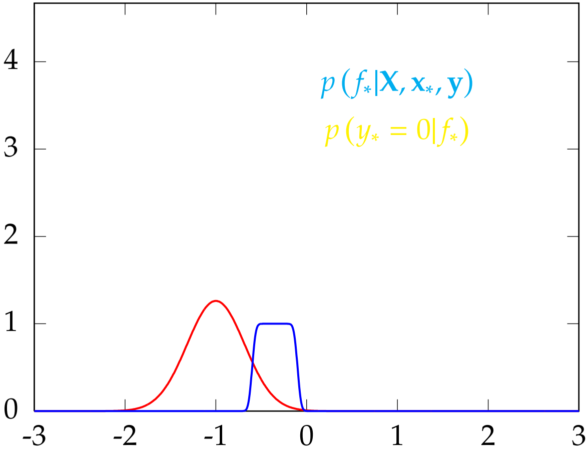


Figure : An EP style update with an ordered category noise model.

Ordinal Regression

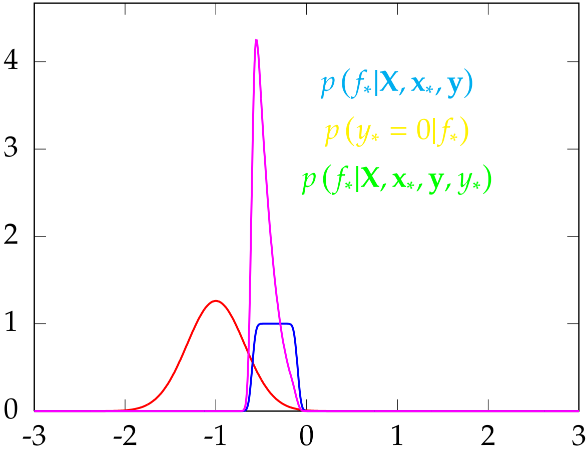


Figure : An EP style update with an ordered category noise model.

Ordinal Regression

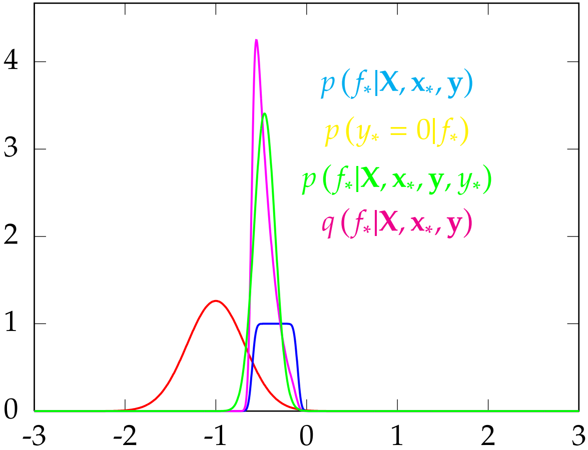


Figure : An EP style update with an ordered category noise model.

Null Category Noise Model

Classification with a Missing Category

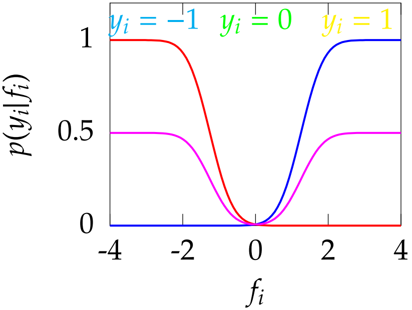


Figure : The null category noise model (semi-supervised learning). The plot shows $p\,(y_i|f_i)$ for different values of y_i. Here we have assumed three categories.

Semi-supervised Learning

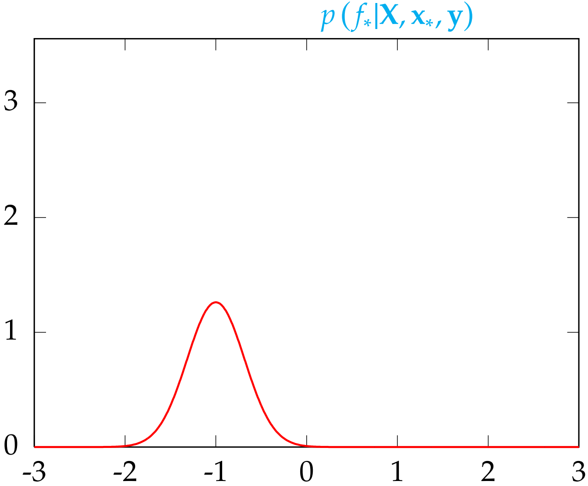


Figure : An EP style update with an null category noise model.

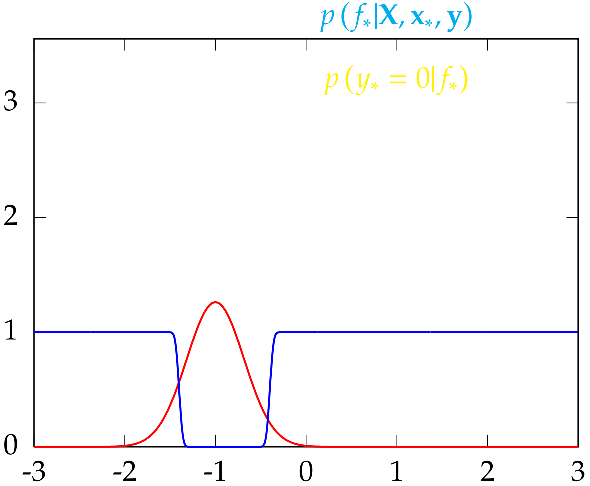


Figure : An EP style update with an null category noise model.

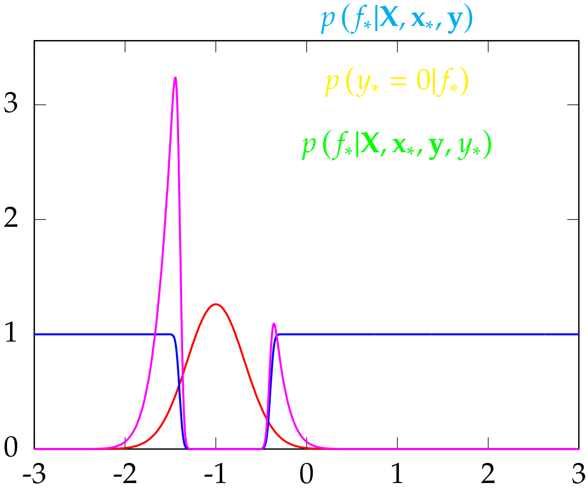


Figure : An EP style update with an null category noise model.

Semi-supervised Learning

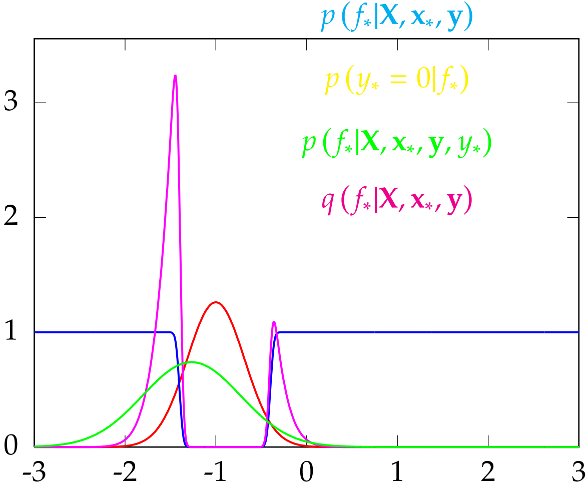


Figure : An EP style update with an null category noise model.

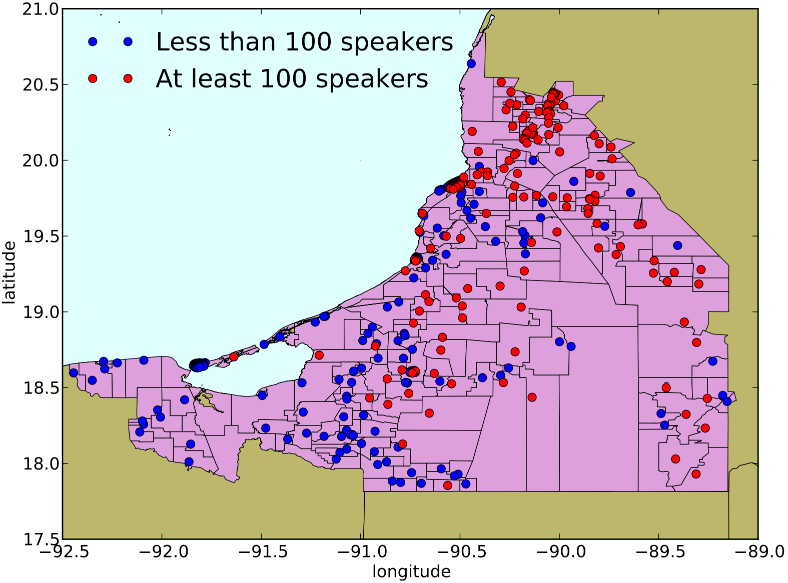
Predictions

- Predictive distribution of $q(f_*|\mathbf{y})$ is also Gaussian:

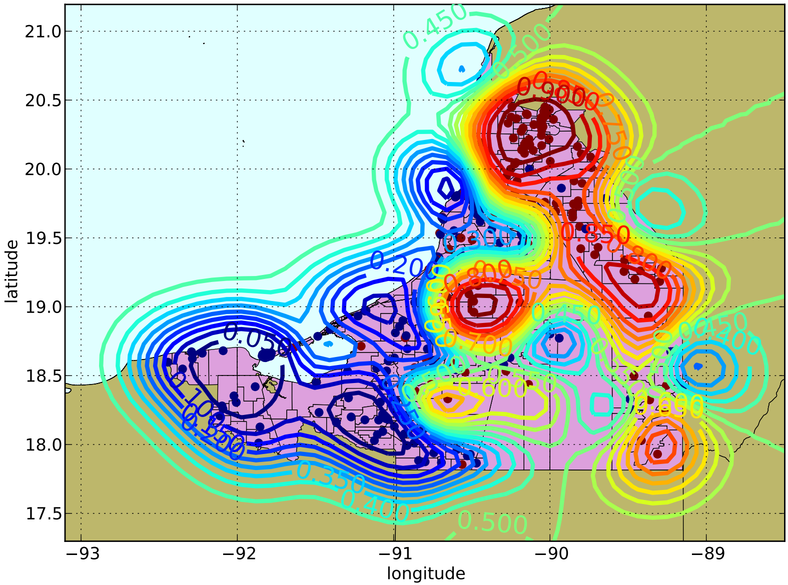
$$\langle f_* \rangle_{q(f_*|\mathbf{y})} = \mathbf{k}_*^\top \left(\mathbf{K}_{\mathbf{f},\mathbf{f}} + \boldsymbol{\Sigma} t\right)^{-1} \tilde{\boldsymbol{\mu}}$$

$$\mathrm{var}\left(f_*\right) = k_{*,*} - \mathbf{k}_*^\top \left(\mathbf{K}_{\mathbf{f},\mathbf{f}} + \boldsymbol{\Sigma}_t\right)^{-1} \mathbf{k}_*$$

Example: People who speak an indigenous language



Example: People who speak an indigenous language



Posterior variance update

- Complexity is dominated by the computation of the posterior covariance:

$$\boldsymbol{\Sigma} = \left(\mathbf{K}_{\mathbf{f},\mathbf{f}}^{-1} + \boldsymbol{\Sigma}_t^{-1}\right)^{-1}$$

Sparse EP

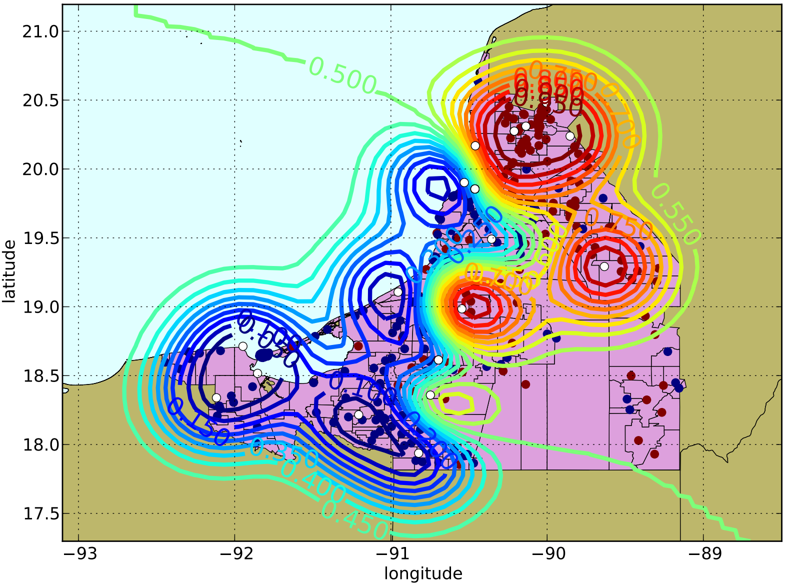
- $q(\mathbf{f}|\mathbf{y})$ is computed as before, but an sparse approximation is used instead of the exact covariance $\mathbf{K}_{\mathbf{f},\mathbf{f}}$.
- FITC approximation: $\mathcal{O}(nm^2)$

$$\mathbf{K}_{\mathbf{f},\mathbf{f}} \approx \mathbf{K}_{\mathbf{f},\mathbf{u}}\mathbf{K}_{\mathbf{u},\mathbf{u}}^{-1}\mathbf{K}_{\mathbf{u},\mathbf{f}} + \mathrm{diag}\left(\mathbf{K}_{\mathbf{f},\mathbf{f}} - \mathbf{Q}_{\mathbf{f},\mathbf{f}}\right)$$

- DTC approximation: $\mathcal{O}(nm^2)$

$$\mathbf{K}_{\mathbf{f},\mathbf{f}} \approx \mathbf{K}_{\mathbf{f},\mathbf{u}}\mathbf{K}_{\mathbf{u},\mathbf{u}}^{-1}\mathbf{K}_{\mathbf{u},\mathbf{f}}$$

EP-FITC (generalized FITC)

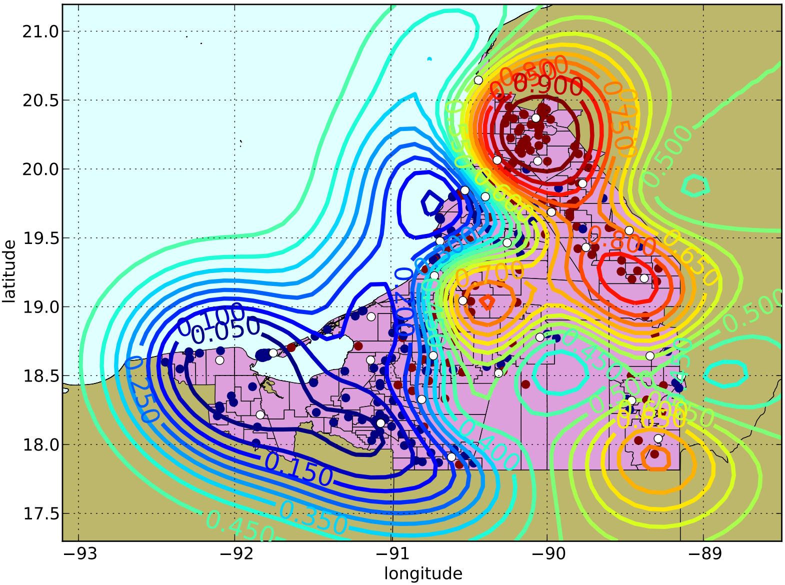


EP-DTC

Compatible with sparse variational approach:

$$\mathcal{L} = \log \mathcal{N}\left(\boldsymbol{\mu}_t | \mathbf{0}, \mathbf{Q}_{\mathbf{f},\mathbf{f}} + \boldsymbol{\Sigma}_t\right) - \frac{1}{2}\mathrm{tr}\left((\mathbf{K}_{\mathbf{f},\mathbf{f}} - \mathbf{Q}_{\mathbf{f},\mathbf{f}})\boldsymbol{\Sigma}_{t_i}\right) - Z_{EP}$$

Sparse variational + EP-DTC



Dimensionality Reduction

Neil D. Lawrence

GPRS
19th–22nd January 2015

Outline

Regression

Bayesian Perspective

Gaussian Processes

Multiple Output Processes

Latent Force Models

Approximations

Dimensionality Reduction

Outline

Regression

Bayesian Perspective

Gaussian Processes

Multiple Output Processes

Latent Force Models

Approximations

Dimensionality Reduction

Motivation for Non-Linear Dimensionality Reduction

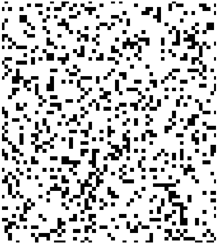
USPS Data Set Handwritten Digit

- 3648 Dimensions
  - 64 rows by 57 columns
  - Space contains more than just this digit.
  - Even if we sample every nanosecond from now until the end of the universe, you won't see the original six!

Motivation for Non-Linear Dimensionality Reduction

USPS Data Set Handwritten Digit

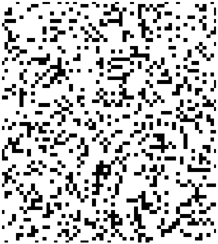
- 3648 Dimensions
  - 64 rows by 57 columns
  - Space contains more than just this digit.
  - Even if we sample every nanosecond from now until the end of the universe, you won't see the original six!



Motivation for Non-Linear Dimensionality Reduction

USPS Data Set Handwritten Digit

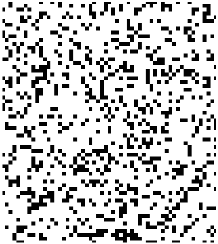
- 3648 Dimensions
  - 64 rows by 57 columns
  - Space contains more than just this digit.
  - Even if we sample every nanosecond from now until the end of the universe, you won't see the original six!



Motivation for Non-Linear Dimensionality Reduction

USPS Data Set Handwritten Digit

- 3648 Dimensions
  - 64 rows by 57 columns
  - Space contains more than just this digit.
  - Even if we sample every nanosecond from now until the end of the universe, you won't see the original six!



Simple Model of Digit

Rotate a 'Prototype'

Simple Model of Digit

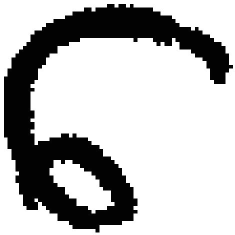
Rotate a 'Prototype'

Simple Model of Digit

Rotate a 'Prototype'

Simple Model of Digit

Rotate a 'Prototype'



Simple Model of Digit

Rotate a 'Prototype'

Simple Model of Digit

Rotate a 'Prototype'

Simple Model of Digit

Rotate a 'Prototype'

Simple Model of Digit

Rotate a 'Prototype'

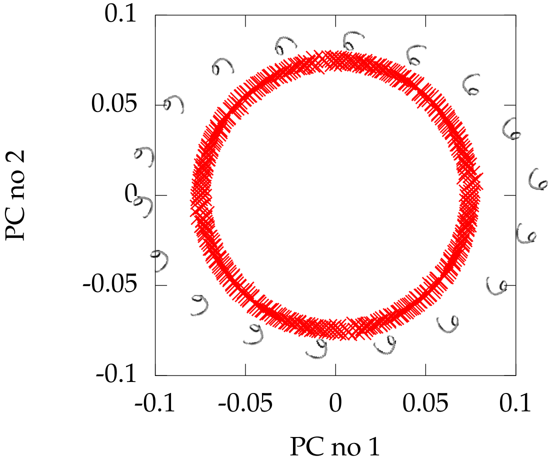
Rotate a 'Prototype'

MATLAB Demo

```
demDigitsManifold([1 2], 'all')
```

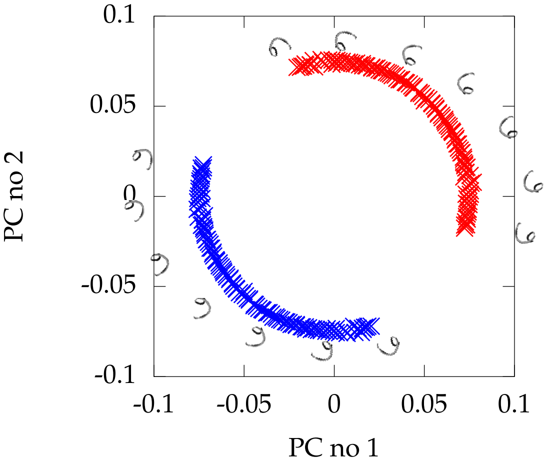
MATLAB Demo

```
demDigitsManifold([1 2], 'all')
```



MATLAB Demo

```
demDigitsManifold([1 2], 'sixnine')
```



Low Dimensional Manifolds

Pure Rotation is too Simple

- In practice the data may undergo several distortions.
  - *e.g.* digits undergo 'thinning', translation and rotation.
- For data with 'structure':
  - we expect fewer distortions than dimensions;
  - we therefore expect the data to live on a lower dimensional manifold.
- Conclusion: deal with high dimensional data by looking for lower dimensional non-linear embedding.

Existing Methods

Spectral Approaches

- Classical Multidimensional Scaling (MDS) (Mardia et al., 1979).
  - Uses eigenvectors of similarity matrix.
    - Isomap (Tenenbaum et al., 2000) is MDS with a particular proximity measure.
  - Kernel PCA (Schölkopf et al., 1998)
    - Provides a representation and a mapping — dimensional expansion.
    - Mapping is implied throught he use of a kernel function as a similarity matrix.
  - Locally Linear Embedding (Roweis and Saul, 2000).
    - Looks to preserve locally linear relationships in a low dimensional space.

Existing Methods II

Iterative Methods

- Multidimensional Scaling (MDS)
  - Iterative optimisation of a stress function (Kruskal, 1964).
  - Sammon Mappings (Sammon, 1969).
    - Strictly speaking not a mapping — similar to iterative MDS.
- NeuroScale (Lowe and Tipping, 1997)
  - Augmentation of iterative MDS methods with a mapping.

Existing Methods III

Probabilistic Approaches

- Probabilistic PCA (Tipping and Bishop, 1999; Roweis, 1998)
  - A linear method.
- Density Networks (MacKay, 1995)
  - Use importance sampling and a multi-layer perceptron.
- Generative Topographic Mapping (GTM) (Bishop et al., 1998)
  - Uses a grid based sample and an RBF network.

Existing Methods III

Probabilistic Approaches

- Probabilistic PCA (Tipping and Bishop, 1999; Roweis, 1998)
  - A linear method.
- Density Networks (MacKay, 1995)
  - Use importance sampling and a multi-layer perceptron.
- Generative Topographic Mapping (GTM) (Bishop et al., 1998)
  - Uses a grid based sample and an RBF network.

Existing Methods III

Probabilistic Approaches

- Probabilistic PCA (Tipping and Bishop, 1999; Roweis, 1998)
  - A linear method.
- Density Networks (MacKay, 1995)
  - Use importance sampling and a multi-layer perceptron.
- Generative Topographic Mapping (GTM) (Bishop et al., 1998)
  - Uses a grid based sample and an RBF network.

Existing Methods III

Probabilistic Approaches

- Probabilistic PCA (Tipping and Bishop, 1999; Roweis, 1998)
  - A linear method.
- Density Networks (MacKay, 1995)
  - Use importance sampling and a multi-layer perceptron.
- Generative Topographic Mapping (GTM) (Bishop et al., 1998)
  - Uses a grid based sample and an RBF network.

Difficulty for Probabilistic Approaches

- Propagate a probability distribution through a non-linear mapping.

A Probabilistic Non-linear PCA

- PCA has a probabilistic interpretation (Tipping and Bishop, 1999; Roweis, 1998).
- It is difficult to 'non-linearise'.

Dual Probabilistic PCA

- We present a new probabilistic interpretation of PCA (Lawrence, 2005).
- This interpretation can be made non-linear.
- The result is non-linear probabilistic PCA.

Notation

q— dimension of latent/embedded space
p— dimension of data space
n— number of data points

$$\text{centred data, } \mathbf{Y} = [\mathbf{y}_{1,:}, \ldots, \mathbf{y}_{n,:}]^\top = \left[\mathbf{y}_{:,1}, \ldots, \mathbf{y}_{:,p}\right] \in \Re^{n\times p}$$

$$\text{latent variables, } \mathbf{X} = [\mathbf{x}_{1,:}, \ldots, \mathbf{x}_{n,:}]^\top = \left[\mathbf{x}_{:,1}, \ldots, \mathbf{x}_{:,q}\right] \in \Re^{n\times q}$$

$$\text{mapping matrix, } \mathbf{W} \in \Re^{p\times q}$$

$\mathbf{a}_{i,:}$ is a vector from the ith row of a given matrix $\mathbf{A}$
$\mathbf{a}_{:,j}$ is a vector from the jth row of a given matrix $\mathbf{A}$

Reading Notation

$\mathbf{X}$ and $\mathbf{Y}$ are *design matrices*

- Covariance given by $n^{-1}\mathbf{Y}^\top\mathbf{Y}$.
- Inner product matrix given by $\mathbf{Y}\mathbf{Y}^\top$.

Linear Dimensionality Reduction

Linear Latent Variable Model

- Represent data, $\mathbf{Y}$, with a lower dimensional set of latent variables $\mathbf{X}$.
- Assume a linear relationship of the form

$$\mathbf{y}_{i,:} = \mathbf{W}\mathbf{x}_{i,:} + \boldsymbol{\epsilon}_{i,:},$$

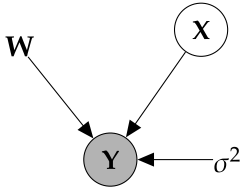
where

$$\boldsymbol{\epsilon}_{i,:} \sim \mathcal{N}\left(\mathbf{0}, \sigma^2\mathbf{I}\right).$$

Linear Latent Variable Model

Probabilistic PCA

- Define *linear-Gaussian relationship* between latent variables and data.
- **Standard** Latent variable approach:
  - Define Gaussian prior over *latent space*, $\mathbf{X}$.
  - Integrate out *latent variables*.

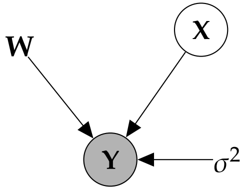


$$p(\mathbf{Y}|\mathbf{X},\mathbf{W}) = \prod_{i=1}^{n} \mathcal{N}\left(\mathbf{y}_{i,:}|\mathbf{W}\mathbf{x}_{i,:},\sigma^2\mathbf{I}\right)$$

Linear Latent Variable Model

Probabilistic PCA

- Define *linear-Gaussian relationship* between latent variables and data.
- **Standard** Latent variable approach:
  - Define Gaussian prior over *latent space*, $\mathbf{X}$.
  - Integrate out *latent variables*.

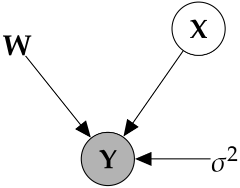


$$p(\mathbf{Y}|\mathbf{X}, \mathbf{W}) = \prod_{i=1}^{n} \mathcal{N}\left(\mathbf{y}_{i,:}|\mathbf{W}\mathbf{x}_{i,:}, \sigma^2\mathbf{I}\right)$$

Linear Latent Variable Model

Probabilistic PCA

- Define *linear-Gaussian relationship* between latent variables and data.
- **Standard** Latent variable approach:
  - Define Gaussian prior over *latent space*, $\mathbf{X}$.
  - Integrate out *latent variables*.



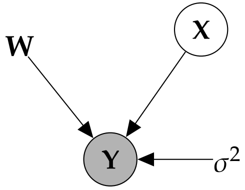
$$p(\mathbf{Y}|\mathbf{X},\mathbf{W}) = \prod_{i=1}^{n} \mathcal{N}\left(\mathbf{y}_{i,:}|\mathbf{W}\mathbf{x}_{i,:},\sigma^2\mathbf{I}\right)$$

$$p(\mathbf{X}) = \prod_{i=1}^{n} \mathcal{N}\left(\mathbf{x}_{i,:}|\mathbf{0},\mathbf{I}\right)$$

Linear Latent Variable Model

Probabilistic PCA

- Define *linear-Gaussian relationship* between latent variables and data.
- **Standard** Latent variable approach:
  - Define Gaussian prior over *latent space*, $\mathbf{X}$.
  - Integrate out *latent variables*.



$$p(\mathbf{Y}|\mathbf{X}, \mathbf{W}) = \prod_{i=1}^{n} \mathcal{N}\left(\mathbf{y}_{i,:}|\mathbf{W}\mathbf{x}_{i,:}, \sigma^2\mathbf{I}\right)$$

$$p(\mathbf{X}) = \prod_{i=1}^{n} \mathcal{N}\left(\mathbf{x}_{i,:}|\mathbf{0}, \mathbf{I}\right)$$

$$p(\mathbf{Y}|\mathbf{W}) = \prod_{i=1}^{n} \mathcal{N}\left(\mathbf{y}_{i,:}|\mathbf{0}, \mathbf{W}\mathbf{W}^{\top} + \sigma^2\mathbf{I}\right)$$

Computation of the Marginal Likelihood

$$\mathbf{y}_{i,:} = \mathbf{W}\mathbf{x}_{i,:} + \boldsymbol{\epsilon}_{i,:}, \quad \mathbf{x}_{i,:} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}), \quad \boldsymbol{\epsilon}_{i,:} \sim \mathcal{N}\left(\mathbf{0}, \sigma^2\mathbf{I}\right)$$

$$\mathbf{W}\mathbf{x}_{i,:} \sim \mathcal{N}\left(\mathbf{0}, \mathbf{W}\mathbf{W}^\top\right),$$

$$\mathbf{W}\mathbf{x}_{i,:} + \epsilon_{i,:} \sim \mathcal{N}\left(\mathbf{0}, \mathbf{W}\mathbf{W}^\top + \sigma^2\mathbf{I}\right)$$

Computation of the Marginal Likelihood

$$\mathbf{y}_{i,:} = \mathbf{W}\mathbf{x}_{i,:} + \boldsymbol{\epsilon}_{i,:}, \quad \mathbf{x}_{i,:} \sim \mathcal{N}\left(\mathbf{0}, \mathbf{I}\right), \quad \boldsymbol{\epsilon}_{i,:} \sim \mathcal{N}\left(\mathbf{0}, \sigma^2\mathbf{I}\right)$$

$$\mathbf{W}\mathbf{x}_{i,:} \sim \mathcal{N}\left(\mathbf{0}, \mathbf{W}\mathbf{W}^\top\right),$$

$$\mathbf{W}\mathbf{x}_{i,:} + \epsilon_{i,:} \sim \mathcal{N}\left(\mathbf{0}, \mathbf{W}\mathbf{W}^\top + \sigma^2\mathbf{I}\right)$$

Computation of the Marginal Likelihood

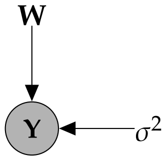
$$\mathbf{y}_{i,:} = \mathbf{W}\mathbf{x}_{i,:} + \boldsymbol{\epsilon}_{i,:}, \quad \mathbf{x}_{i,:} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}), \quad \boldsymbol{\epsilon}_{i,:} \sim \mathcal{N}\left(\mathbf{0}, \sigma^2\mathbf{I}\right)$$

$$\mathbf{W}\mathbf{x}_{i,:} \sim \mathcal{N}\left(\mathbf{0}, \mathbf{W}\mathbf{W}^\top\right),$$

$$\mathbf{W}\mathbf{x}_{i,:} + \epsilon_{i,:} \sim \mathcal{N}\left(\mathbf{0}, \mathbf{W}\mathbf{W}^\top + \sigma^2\mathbf{I}\right)$$

Linear Latent Variable Model II

Probabilistic PCA Max. Likelihood Soln (Tipping and Bishop, 1999)



$$p\left(\mathbf{Y}|\mathbf{W}\right) = \prod_{i=1}^{n} \mathcal{N}\left(\mathbf{y}_{i,:}|\mathbf{0}, \mathbf{W}\mathbf{W}^{\top} + \sigma^2\mathbf{I}\right)$$

Linear Latent Variable Model II

Probabilistic PCA Max. Likelihood Soln (Tipping and Bishop, 1999)

$$p(\mathbf{Y}|\mathbf{W}) = \prod_{i=1}^{n} \mathcal{N}(\mathbf{y}_{i,:}|\mathbf{0}, \mathbf{C}), \quad \mathbf{C} = \mathbf{W}\mathbf{W}^\top + \sigma^2\mathbf{I}$$

$$\log p(\mathbf{Y}|\mathbf{W}) = -\frac{n}{2}\log|\mathbf{C}| - \frac{1}{2}\mathrm{tr}\left(\mathbf{C}^{-1}\mathbf{Y}^\top\mathbf{Y}\right) + \text{const.}$$

If $\mathbf{U}_q$ are first q principal eigenvectors of $n^{-1}\mathbf{Y}^\top\mathbf{Y}$ and the corresponding eigenvalues are $\mathbf{\Lambda}_q$,

$$\mathbf{W} = \mathbf{U}_q\mathbf{L}\mathbf{R}^\top, \quad \mathbf{L} = \left(\mathbf{\Lambda}_q - \sigma^2\mathbf{I}\right)^{\frac{1}{2}}$$

where $\mathbf{R}$ is an arbitrary rotation matrix.

Linear Latent Variable Model II

Probabilistic PCA Max. Likelihood Soln (Tipping and Bishop, 1999)

$$p(\mathbf{Y}|\mathbf{W}) = \prod_{i=1}^{n} \mathcal{N}\left(\mathbf{y}_{i,:}|\mathbf{0}, \mathbf{C}\right), \quad \mathbf{C} = \mathbf{W}\mathbf{W}^\top + \sigma^2\mathbf{I}$$

$$\log p(\mathbf{Y}|\mathbf{W}) = -\frac{n}{2}\log|\mathbf{C}| - \frac{1}{2}\mathrm{tr}\left(\mathbf{C}^{-1}\mathbf{Y}^\top\mathbf{Y}\right) + \text{const.}$$

If $\mathbf{U}_q$ are first q principal eigenvectors of $n^{-1}\mathbf{Y}^\top\mathbf{Y}$ and the corresponding eigenvalues are $\mathbf{\Lambda}_q$,

$$\mathbf{W} = \mathbf{U}_q\mathbf{L}\mathbf{R}^\top, \quad \mathbf{L} = \left(\mathbf{\Lambda}_q - \sigma^2\mathbf{I}\right)^{\frac{1}{2}}$$

where $\mathbf{R}$ is an arbitrary rotation matrix.

Linear Latent Variable Model II

Probabilistic PCA Max. Likelihood Soln (Tipping and Bishop, 1999)

$$p(\mathbf{Y}|\mathbf{W}) = \prod_{i=1}^{n} \mathcal{N}(\mathbf{y}_{i,:}|\mathbf{0}, \mathbf{C}), \quad \mathbf{C} = \mathbf{W}\mathbf{W}^\top + \sigma^2\mathbf{I}$$

$$\log p(\mathbf{Y}|\mathbf{W}) = -\frac{n}{2}\log|\mathbf{C}| - \frac{1}{2}\mathrm{tr}\left(\mathbf{C}^{-1}\mathbf{Y}^\top\mathbf{Y}\right) + \text{const.}$$

If $\mathbf{U}_q$ are first q principal eigenvectors of $n^{-1}\mathbf{Y}^\top\mathbf{Y}$ and the corresponding eigenvalues are $\mathbf{\Lambda}_q$,

$$\mathbf{W} = \mathbf{U}_q\mathbf{L}\mathbf{R}^\top, \quad \mathbf{L} = \left(\mathbf{\Lambda}_q - \sigma^2\mathbf{I}\right)^{\frac{1}{2}}$$

where $\mathbf{R}$ is an arbitrary rotation matrix.

Linear Latent Variable Model II

Probabilistic PCA Max. Likelihood Soln (Tipping and Bishop, 1999)

$$p(\mathbf{Y}|\mathbf{W}) = \prod_{i=1}^{n} \mathcal{N}\left(\mathbf{y}_{i,:}|\mathbf{0}, \mathbf{C}\right), \quad \mathbf{C} = \mathbf{W}\mathbf{W}^{\top} + \sigma^2\mathbf{I}$$

$$\log p(\mathbf{Y}|\mathbf{W}) = -\frac{n}{2}\log|\mathbf{C}| - \frac{1}{2}\mathrm{tr}\left(\mathbf{C}^{-1}\mathbf{Y}^{\top}\mathbf{Y}\right) + \text{const.}$$

If $\mathbf{U}_q$ are first q principal eigenvectors of $n^{-1}\mathbf{Y}^{\top}\mathbf{Y}$ and the corresponding eigenvalues are $\mathbf{\Lambda}_q$,

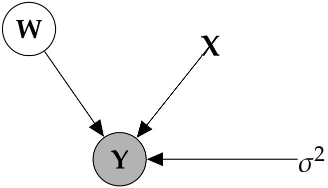
$$\mathbf{W} = \mathbf{U}_q\mathbf{L}\mathbf{R}^{\top}, \quad \mathbf{L} = \left(\mathbf{\Lambda}_q - \sigma^2\mathbf{I}\right)^{\frac{1}{2}}$$

where $\mathbf{R}$ is an arbitrary rotation matrix.

Linear Latent Variable Model III

Dual Probabilistic PCA

- Define *linear-Gaussian relationship* between latent variables and data.
- **Novel** Latent variable approach:
  - Define Gaussian prior over *parameters*, **W**.
  - Integrate out *parameters*.

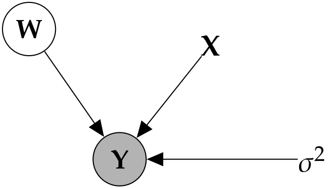


$$p\left(\mathbf{Y}|\mathbf{X},\mathbf{W}\right)=\prod_{i=1}^{n}\mathcal{N}\left(\mathbf{y}_{i,:}|\mathbf{W}\mathbf{x}_{i,:},\sigma^{2}\mathbf{I}\right)$$

Linear Latent Variable Model III

Dual Probabilistic PCA

- Define *linear-Gaussian relationship* between latent variables and data.
- **Novel** Latent variable approach:
  - Define Gaussian prior over *parameters*, $\mathbf{W}$.
  - Integrate out *parameters*.

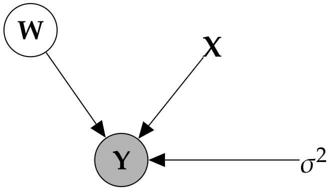


$$p(\mathbf{Y}|\mathbf{X},\mathbf{W}) = \prod_{i=1}^{n} \mathcal{N}\left(\mathbf{y}_{i,:}|\mathbf{W}\mathbf{x}_{i,:},\sigma^2\mathbf{I}\right)$$

Linear Latent Variable Model III

Dual Probabilistic PCA

- Define *linear-Gaussian relationship* between latent variables and data.
- **Novel** Latent variable approach:
  - Define Gaussian prior over *parameters*, $\mathbf{W}$.
  - Integrate out *parameters*.



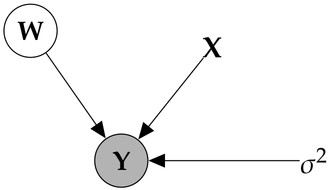
$$p(\mathbf{Y}|\mathbf{X},\mathbf{W}) = \prod_{i=1}^{n} \mathcal{N}\left(\mathbf{y}_{i,:}|\mathbf{W}\mathbf{x}_{i,:}, \sigma^2\mathbf{I}\right)$$

$$p(\mathbf{W}) = \prod_{i=1}^{p} \mathcal{N}\left(\mathbf{w}_{i,:}|\mathbf{0},\mathbf{I}\right)$$

Linear Latent Variable Model III

Dual Probabilistic PCA

- Define *linear-Gaussian relationship* between latent variables and data.
- **Novel** Latent variable approach:
  - Define Gaussian prior over *parameters*, $\mathbf{W}$.
  - Integrate out *parameters*.



$$p(\mathbf{Y}|\mathbf{X},\mathbf{W}) = \prod_{i=1}^{n} \mathcal{N}\left(\mathbf{y}_{i,:}|\mathbf{W}\mathbf{x}_{i,:}, \sigma^2\mathbf{I}\right)$$

$$p(\mathbf{W}) = \prod_{i=1}^{p} \mathcal{N}\left(\mathbf{w}_{i,:}|\mathbf{0},\mathbf{I}\right)$$

$$p(\mathbf{Y}|\mathbf{X}) = \prod_{j=1}^{p} \mathcal{N}\left(\mathbf{y}_{:,j}|\mathbf{0},\mathbf{X}\mathbf{X}^\top + \sigma^2\mathbf{I}\right)$$

Computation of the Marginal Likelihood

$$\mathbf{y}_{:,j} = \mathbf{X}\mathbf{w}_{:,j} + \boldsymbol{\epsilon}_{:,j}, \quad \mathbf{w}_{:,j} \sim \mathcal{N}\left(\mathbf{0}, \mathbf{I}\right), \quad \boldsymbol{\epsilon}_{i,:} \sim \mathcal{N}\left(\mathbf{0}, \sigma^2\mathbf{I}\right)$$

$$\mathbf{X}\mathbf{w}_{:,j} \sim \mathcal{N}\left(\mathbf{0}, \mathbf{X}\mathbf{X}^\top\right),$$

$$\mathbf{X}\mathbf{w}_{:,j} + \epsilon_{:,j} \sim \mathcal{N}\left(\mathbf{0}, \mathbf{X}\mathbf{X}^\top + \sigma^2\mathbf{I}\right)$$

Computation of the Marginal Likelihood

$$\mathbf{y}_{:,j} = \mathbf{X}\mathbf{w}_{:,j} + \boldsymbol{\epsilon}_{:,j}, \quad \mathbf{w}_{:,j} \sim \mathcal{N}\left(\mathbf{0}, \mathbf{I}\right), \quad \boldsymbol{\epsilon}_{i,:} \sim \mathcal{N}\left(\mathbf{0}, \sigma^2\mathbf{I}\right)$$

$$\mathbf{X}\mathbf{w}_{:,j} \sim \mathcal{N}\left(\mathbf{0}, \mathbf{X}\mathbf{X}^\top\right),$$

$$\mathbf{X}\mathbf{w}_{:,j} + \epsilon_{:,j} \sim \mathcal{N}\left(\mathbf{0}, \mathbf{X}\mathbf{X}^\top + \sigma^2\mathbf{I}\right)$$

Computation of the Marginal Likelihood

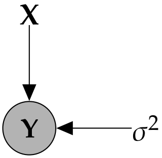
$$\mathbf{y}_{:,j} = \mathbf{X}\mathbf{w}_{:,j} + \boldsymbol{\epsilon}_{:,j}, \quad \mathbf{w}_{:,j} \sim \mathcal{N}\left(\mathbf{0}, \mathbf{I}\right), \quad \boldsymbol{\epsilon}_{i,:} \sim \mathcal{N}\left(\mathbf{0}, \sigma^2\mathbf{I}\right)$$

$$\mathbf{X}\mathbf{w}_{:,j} \sim \mathcal{N}\left(\mathbf{0}, \mathbf{X}\mathbf{X}^\top\right),$$

$$\mathbf{X}\mathbf{w}_{:,j} + \epsilon_{:,j} \sim \mathcal{N}\left(\mathbf{0}, \mathbf{X}\mathbf{X}^\top + \sigma^2\mathbf{I}\right)$$

Linear Latent Variable Model IV

Dual Probabilistic PCA Max. Likelihood Soln (Lawrence, 2004, 2005)



$$p(\mathbf{Y}|\mathbf{X}) = \prod_{j=1}^{p} \mathcal{N}\left(\mathbf{y}_{:,j}|\mathbf{0}, \mathbf{X}\mathbf{X}^\top + \sigma^2\mathbf{I}\right)$$

Linear Latent Variable Model IV

Dual PPCA Max. Likelihood Soln (Lawrence, 2004, 2005)

$$p(\mathbf{Y}|\mathbf{X}) = \prod_{j=1}^{p} \mathcal{N}\left(\mathbf{y}_{:,j}|\mathbf{0}, \mathbf{K}\right), \quad \mathbf{K} = \mathbf{X}\mathbf{X}^{\top} + \sigma^2\mathbf{I}$$

$$\log p(\mathbf{Y}|\mathbf{X}) = -\frac{p}{2}\log|\mathbf{K}| - \frac{1}{2}\mathrm{tr}\left(\mathbf{K}^{-1}\mathbf{Y}\mathbf{Y}^{\top}\right) + \text{const.}$$

If $\mathbf{U}'_q$ are first q principal eigenvectors of $p^{-1}\mathbf{Y}\mathbf{Y}^{\top}$ and the corresponding eigenvalues are $\mathbf{\Lambda}_q$,

$$\mathbf{X} = \mathbf{U}'_q\mathbf{L}\mathbf{R}^{\top}, \quad \mathbf{L} = \left(\mathbf{\Lambda}_q - \sigma^2\mathbf{I}\right)^{\frac{1}{2}}$$

where $\mathbf{R}$ is an arbitrary rotation matrix.

Linear Latent Variable Model IV

PPCA Max. Likelihood Soln (Tipping and Bishop, 1999)

$$p(\mathbf{Y}|\mathbf{X}) = \prod_{j=1}^{p} \mathcal{N}\left(\mathbf{y}_{:,j}|\mathbf{0}, \mathbf{K}\right), \quad \mathbf{K} = \mathbf{X}\mathbf{X}^\top + \sigma^2\mathbf{I}$$

$$\log p(\mathbf{Y}|\mathbf{X}) = -\frac{p}{2}\log|\mathbf{K}| - \frac{1}{2}\mathrm{tr}\left(\mathbf{K}^{-1}\mathbf{Y}\mathbf{Y}^\top\right) + \text{const.}$$

If $\mathbf{U}_q'$ are first q principal eigenvectors of $p^{-1}\mathbf{Y}\mathbf{Y}^\top$ and the corresponding eigenvalues are $\mathbf{\Lambda}_q$,

$$\mathbf{X} = \mathbf{U}_q'\mathbf{L}\mathbf{R}^\top, \quad \mathbf{L} = \left(\mathbf{\Lambda}_q - \sigma^2\mathbf{I}\right)^{\frac{1}{2}}$$

where $\mathbf{R}$ is an arbitrary rotation matrix.

Linear Latent Variable Model IV

PPCA Max. Likelihood Soln

$$p(\mathbf{Y}|\mathbf{X}) = \prod_{j=1}^{p} \mathcal{N}\left(\mathbf{y}_{:,j}|\mathbf{0}, \mathbf{K}\right), \quad \mathbf{K} = \mathbf{X}\mathbf{X}^{\top} + \sigma^2\mathbf{I}$$

$$\log p(\mathbf{Y}|\mathbf{X}) = -\frac{p}{2}\log|\mathbf{K}| - \frac{1}{2}\mathrm{tr}\left(\mathbf{K}^{-1}\mathbf{Y}\mathbf{Y}^{\top}\right) + \text{const.}$$

If $\mathbf{U}_q'$ are first q principal eigenvectors of $p^{-1}\mathbf{Y}\mathbf{Y}^{\top}$ and the corresponding eigenvalues are $\mathbf{\Lambda}_q$,

$$\mathbf{X} = \mathbf{U}_q'\mathbf{L}\mathbf{R}^{\top}, \quad \mathbf{L} = \left(\mathbf{\Lambda}_q - \sigma^2\mathbf{I}\right)^{\frac{1}{2}}$$

where $\mathbf{R}$ is an arbitrary rotation matrix.

Linear Latent Variable Model IV

PPCA Max. Likelihood Soln

$$p\left(\mathbf{Y}|\mathbf{X}\right) = \prod_{j=1}^{p} \mathcal{N}\left(\mathbf{y}_{:,j}|\mathbf{0}, \mathbf{K}\right), \quad \mathbf{K} = \mathbf{X}\mathbf{X}^{\top} + \sigma^2\mathbf{I}$$

$$\log p\left(\mathbf{Y}|\mathbf{X}\right) = -\frac{p}{2}\log\left|\mathbf{K}\right| - \frac{1}{2}\mathrm{tr}\left(\mathbf{K}^{-1}\mathbf{Y}\mathbf{Y}^{\top}\right) + \text{const.}$$

If $\mathbf{U}_q'$ are first q principal eigenvectors of $p^{-1}\mathbf{Y}\mathbf{Y}^{\top}$ and the corresponding eigenvalues are $\mathbf{\Lambda}_q$,

$$\mathbf{X} = \mathbf{U}_q'\mathbf{L}\mathbf{R}^{\top}, \quad \mathbf{L} = \left(\mathbf{\Lambda}_q - \sigma^2\mathbf{I}\right)^{\frac{1}{2}}$$

where $\mathbf{R}$ is an arbitrary rotation matrix.

Linear Latent Variable Model IV

Dual PPCA Max. Likelihood Soln (Lawrence, 2004, 2005)

$$p(\mathbf{Y}|\mathbf{X}) = \prod_{j=1}^{p} \mathcal{N}\left(\mathbf{y}_{:,j}|\mathbf{0}, \mathbf{K}\right), \quad \mathbf{K} = \mathbf{X}\mathbf{X}^{\top} + \sigma^2\mathbf{I}$$

$$\log p(\mathbf{Y}|\mathbf{X}) = -\frac{p}{2}\log|\mathbf{K}| - \frac{1}{2}\mathrm{tr}\left(\mathbf{K}^{-1}\mathbf{Y}\mathbf{Y}^{\top}\right) + \text{const.}$$

If $\mathbf{U}_q'$ are first q principal eigenvectors of $p^{-1}\mathbf{Y}\mathbf{Y}^{\top}$ and the corresponding eigenvalues are $\mathbf{\Lambda}_q$,

$$\mathbf{X} = \mathbf{U}_q'\mathbf{L}\mathbf{R}^{\top}, \quad \mathbf{L} = \left(\mathbf{\Lambda}_q - \sigma^2\mathbf{I}\right)^{\frac{1}{2}}$$

where $\mathbf{R}$ is an arbitrary rotation matrix.

Linear Latent Variable Model IV

PPCA Max. Likelihood Soln (Tipping and Bishop, 1999)

$$p(\mathbf{Y}|\mathbf{W}) = \prod_{i=1}^{n} \mathcal{N}\left(\mathbf{y}_{i,:}|\mathbf{0}, \mathbf{C}\right), \quad \mathbf{C} = \mathbf{W}\mathbf{W}^{\top} + \sigma^2\mathbf{I}$$

$$\log p(\mathbf{Y}|\mathbf{W}) = -\frac{n}{2}\log|\mathbf{C}| - \frac{1}{2}\mathrm{tr}\left(\mathbf{C}^{-1}\mathbf{Y}^{\top}\mathbf{Y}\right) + \text{const.}$$

If $\mathbf{U}_q$ are first q principal eigenvectors of $n^{-1}\mathbf{Y}^{\top}\mathbf{Y}$ and the corresponding eigenvalues are $\mathbf{\Lambda}_q$,

$$\mathbf{W} = \mathbf{U}_q\mathbf{L}\mathbf{R}^{\top}, \quad \mathbf{L} = \left(\mathbf{\Lambda}_q - \sigma^2\mathbf{I}\right)^{\frac{1}{2}}$$

where $\mathbf{R}$ is an arbitrary rotation matrix.

Equivalence of Formulations

The Eigenvalue Problems are equivalent

- Solution for Probabilistic PCA (solves for the mapping)

$$\mathbf{Y}^\top \mathbf{Y} \mathbf{U}_q = \mathbf{U}_q \boldsymbol{\Lambda}_q \qquad \mathbf{W} = \mathbf{U}_q \mathbf{L} \mathbf{R}^\top$$

- Solution for Dual Probabilistic PCA (solves for the latent positions)

$$\mathbf{Y} \mathbf{Y}^\top \mathbf{U}_q' = \mathbf{U}_q' \boldsymbol{\Lambda}_q \qquad \mathbf{X} = \mathbf{U}_q' \mathbf{L} \mathbf{R}^\top$$

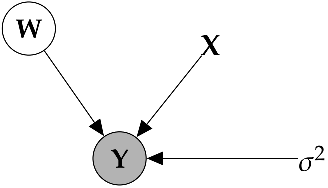
- Equivalence is from

$$\mathbf{U}_q = \mathbf{Y}^\top \mathbf{U}_q' \boldsymbol{\Lambda}_q^{-\frac{1}{2}}$$

Non-Linear Latent Variable Model

Dual Probabilistic PCA

- Define *linear-Gaussian relationship* between latent variables and data.
- **Novel** Latent variable approach:
  - Define Gaussian prior over *parameteters*, $\mathbf{W}$.
  - Integrate out *parameters*.



$$p(\mathbf{Y}|\mathbf{X},\mathbf{W}) = \prod_{i=1}^{n} \mathcal{N}\left(\mathbf{y}_{i,:}|\mathbf{W}\mathbf{x}_{i,:}, \sigma^2\mathbf{I}\right)$$

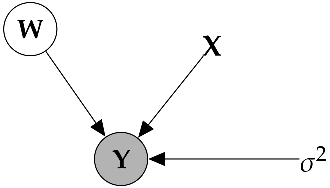
$$p(\mathbf{W}) = \prod_{i=1}^{p} \mathcal{N}\left(\mathbf{w}_{i,:}|\mathbf{0},\mathbf{I}\right)$$

$$p(\mathbf{Y}|\mathbf{X}) = \prod_{j=1}^{p} \mathcal{N}\left(\mathbf{y}_{:,j}|\mathbf{0},\mathbf{X}\mathbf{X}^{\top} + \sigma^2\mathbf{I}\right)$$

Non-Linear Latent Variable Model

Dual Probabilistic PCA

- Inspection of the marginal likelihood shows ...
  - The covariance matrix is a covariance function.
  - We recognise it as the 'linear kernel'.
  - We call this the Gaussian Process Latent Variable model (GP-LVM).

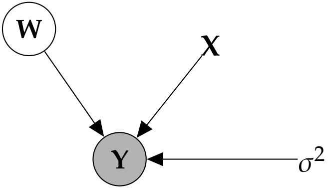


$$p\left(\mathbf{Y}|\mathbf{X}\right) = \prod_{j=1}^{p} \mathcal{N}\left(\mathbf{y}_{:,j}|\mathbf{0}, \mathbf{X}\mathbf{X}^{\top} + \sigma^2\mathbf{I}\right)$$

Non-Linear Latent Variable Model

Dual Probabilistic PCA

- Inspection of the marginal likelihood shows ...
  - The covariance matrix is a covariance function.
  - We recognise it as the 'linear kernel'.
  - We call this the Gaussian Process Latent Variable model (GP-LVM).



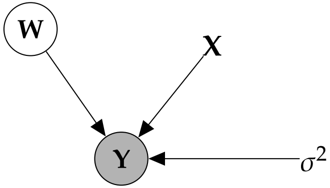
$$p(\mathbf{Y}|\mathbf{X}) = \prod_{j=1}^{p} \mathcal{N}\left(\mathbf{y}_{:,j}|\mathbf{0}, \mathbf{K}\right)$$

$$\mathbf{K} = \mathbf{X}\mathbf{X}^{\top} + \sigma^2\mathbf{I}$$

Non-Linear Latent Variable Model

Dual Probabilistic PCA

- Inspection of the marginal likelihood shows ...
  - The covariance matrix is a covariance function.
  - We recognise it as the 'linear kernel'.
  - We call this the Gaussian Process Latent Variable model (GP-LVM).



$$p(\mathbf{Y}|\mathbf{X}) = \prod_{j=1}^{p} \mathcal{N}\left(\mathbf{y}_{:,j}|\mathbf{0}, \mathbf{K}\right)$$

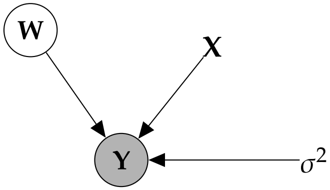
$$\mathbf{K} = \mathbf{X}\mathbf{X}^{\top} + \sigma^2\mathbf{I}$$

This is a product of Gaussian processes with linear kernels.

Non-Linear Latent Variable Model

Dual Probabilistic PCA

- Inspection of the marginal likelihood shows ...
  - The covariance matrix is a covariance function.
  - We recognise it as the 'linear kernel'.
  - We call this the Gaussian Process Latent Variable model (GP-LVM).



$$p\left(\mathbf{Y}|\mathbf{X}\right) = \prod_{j=1}^{p} \mathcal{N}\left(\mathbf{y}_{:,j}|\mathbf{0}, \mathbf{K}\right)$$

$$\mathbf{K} = ?$$

Replace linear kernel with non-linear kernel for non-linear model.

Non-linear Latent Variable Models

Exponentiated Quadratic (EQ) Covariance

- The EQ covariance has the form $k_{i,j} = k\left(\mathbf{x}_{i,:}, \mathbf{x}_{j,:}\right)$, where

$$k\left(\mathbf{x}_{i,:}, \mathbf{x}_{j,:}\right) = \alpha \exp\left(-\frac{\left\|\mathbf{x}_{i,:} - \mathbf{x}_{j,:}\right\|_2^2}{2\ell^2}\right).$$

- No longer possible to optimise wrt $\mathbf{X}$ via an eigenvalue problem.
- Instead find gradients with respect to $\mathbf{X}, \alpha, \ell$ and σ^2 and optimise using conjugate gradients.

Applications

Style Based Inverse Kinematics

- Facilitating animation through modeling human motion (Grochow et al., 2004)

Tracking

- Tracking using human motion models (Urtasun et al., 2005, 2006)

Assisted Animation

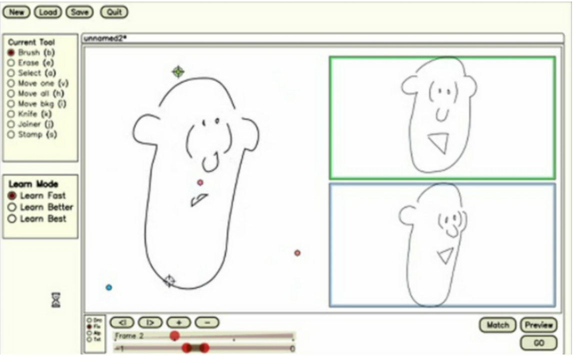
- Generalizing drawings for animation (Baxter and Anjyo, 2006)

Shape Models

- Inferring shape (e.g. pose from silhouette). (Ek et al., 2008b,a; Priacuriu and Reid, 2011a,b)

Example: Latent Doodle Space

(Baxter and Anjyo, 2006)



http://vimeo.com/3235882

Example: Latent Doodle Space

(Baxter and Anjyo, 2006)

Generalization with much less Data than Dimensions

- Powerful uncertainly handling of GPs leads to surprising properties.
- Non-linear models can be used where there are fewer data points than dimensions *without overfitting*.

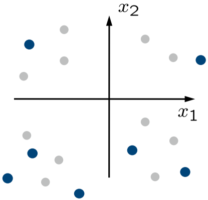
Prior for Supervised Learning

(Urtasun and Darrell, 2007)

- We introduce a prior that is based on the Fisher criteria

$$p(\mathbf{X}) \propto \exp\left\{-\frac{1}{\sigma_d^2}\operatorname{tr}\left(\mathbf{S}_w^{-1}\mathbf{S}_b\right)\right\} ,$$

with $\mathbf{S}_b$ the between class matrix and $\mathbf{S}_w$ the within class matrix



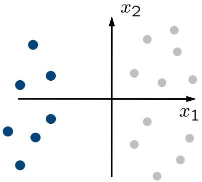
Prior for Supervised Learning

(Urtasun and Darrell, 2007)

- We introduce a prior that is based on the Fisher criteria

$$p(\mathbf{X}) \propto \exp\left\{-\frac{1}{\sigma_d^2}\mathrm{tr}\left(\mathbf{S}_w^{-1}\mathbf{S}_b\right)\right\},$$

with $\mathbf{S}_b$ the between class matrix and $\mathbf{S}_w$ the within class matrix



$$\mathbf{S}_w = \sum_{i=1}^{L} \frac{n_i}{n}(\mathbf{M}_i - \mathbf{M}_0)(\mathbf{M}_i - \mathbf{M}_0)^\top$$

where $\mathbf{X}^{(i)} = [\mathbf{x}_1^{(i)}, \cdots, \mathbf{x}_{n_i}^{(i)}]$ are the n_i training points of class i, $\mathbf{M}_i$ is the mean of the elements of class i, and $\mathbf{M}_0$ is the mean of all the training points of all classes.

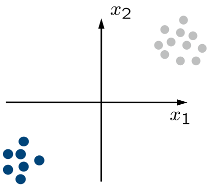
Prior for Supervised Learning

(Urtasun and Darrell, 2007)

- We introduce a prior that is based on the Fisher criteria

$$p(\mathbf{X}) \propto \exp\left\{-\frac{1}{\sigma_d^2}\operatorname{tr}\left(\mathbf{S}_w^{-1}\mathbf{S}_b\right)\right\} ,$$

with $\mathbf{S}_b$ the between class matrix and $\mathbf{S}_w$ the within class matrix



$$\mathbf{S}_w = \sum_{i=1}^{L} \frac{n_i}{n}(\mathbf{M}_i - \mathbf{M}_0)(\mathbf{M}_i - \mathbf{M}_0)^\top$$

$$\mathbf{S}_b = \sum_{i=1}^{L} \frac{n_i}{n}\left[\frac{1}{n_i}\sum_{k=1}^{n_i}(\mathbf{x}_k^{(i)} - \mathbf{M}_i)(\mathbf{x}_k^{(i)} - \mathbf{M}_i)^\top\right]$$

where $\mathbf{X}^{(i)} = [\mathbf{x}_1^{(i)}, \cdots, \mathbf{x}_{n_i}^{(i)}]$ are the n_i training points of class i, $\mathbf{M}_i$ is the mean of the elements of class i, and $\mathbf{M}_0$ is the

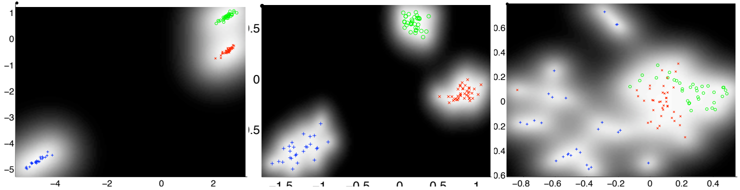
Prior for Supervised Learning

(Urtasun and Darrell, 2007)

- We introduce a prior that is based on the Fisher criteria

$$p(\mathbf{X}) \propto \exp\left\{-\frac{1}{\sigma_d^2}\mathrm{tr}\left(\mathbf{S}_w^{-1}\mathbf{S}_b\right)\right\} ,$$

with $\mathbf{S}_b$ the between class matrix and $\mathbf{S}_w$ the within class matrix



(Lu and Tang, 2014)

- First system to surpass human performance on cropped Learning Faces in Wild Data.
  `http://tinyurl.com/nkt9a38`
- Lots of feature engineering, followed by a Discriminative GP-LVM.

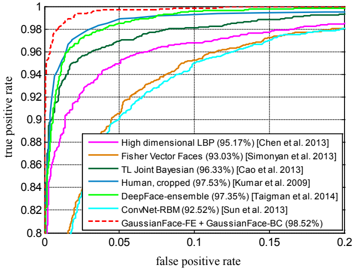


Figure 4: The ROC curve on LFW. Our method achieves the best performance, beating human-level performance.

Figure 5: The two rows present examples of matched anc mismatched pairs respectively from LFW that were incorrectly classified by the GaussianFace model.

Conclusion and Future Work

Continuous Character Control

(Levine et al., 2012)

- Graph diffusion prior for enforcing connectivity between motions.

$$\log p(\mathbf{X}) = w_c \sum_{i,j} \log K^d_{ij}$$

with the graph diffusion kernel $\mathbf{K}^d$ obtain from

$$K^d_{ij} = \exp(\beta \mathbf{H}) \qquad \text{with} \qquad \mathbf{H} = -\mathbf{T}^{-1/2}\mathbf{L}\mathbf{T}^{-1/2}$$

the graph Laplacian, and $\mathbf{T}$ is a diagonal matrix with $T_{ii} = \sum_j w(\mathbf{x}_i, \mathbf{x}_j)$,

$$L_{ij} = \begin{cases} \sum_k w(\mathbf{x}_i, \mathbf{x}_k) & \text{if } i = j \\ -w(\mathbf{x}_i, \mathbf{x}_j) & \text{otherwise.} \end{cases}$$

and $w(\mathbf{x}_i, \mathbf{x}_j) = \|\mathbf{x}_i - \mathbf{x}_j\|^{-p}$ measures similarity.

Character Control: Results

Other Topics

- Local distance preservation ▸ Details
- Dynamical models ▸ Details
- Hierarchical models ▸ Details
- Bayesian GP-LVM ▸ Details

Back Constraints I

Local Distance Preservation **(Lawrence and Quiñonero Candela, 2006)**

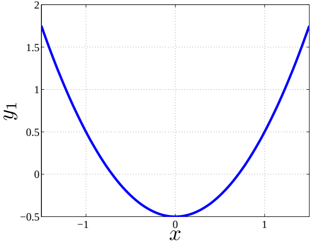
- Most dimensional reduction techniques preserve local distances.
- The GP-LVM does not.
- GP-LVM maps smoothly from latent to data space.
  - Points close in latent space are close in data space.
  - This does not imply points close in data space are close in latent space.
- Kernel PCA maps smoothly from data to latent space.
  - Points close in data space are close in latent space.
  - This does not imply points close in latent space are close in data space.

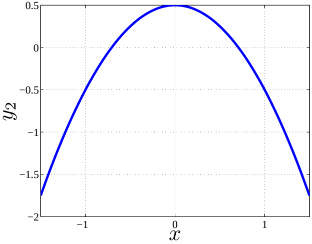
Back Constraints II

Forward Mapping (`demBackMapping` in oxford toolbox)

- Mapping from 1-D latent space to 2-D data space.

$$y_1 = x^2 - 0.5, \quad y_2 = -x^2 + 0.5$$



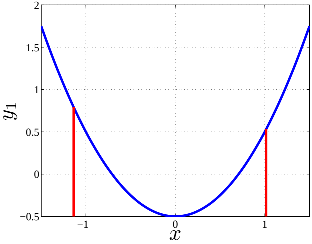


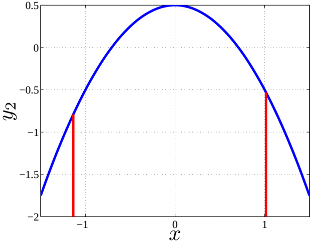
Back Constraints II

Forward Mapping (`demBackMapping` in oxford toolbox)

- Mapping from 1-D latent space to 2-D data space.

$$y_1 = x^2 - 0.5, \quad y_2 = -x^2 + 0.5$$



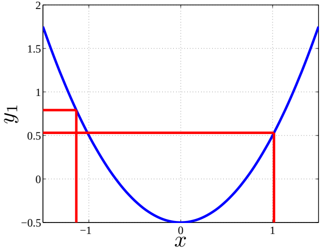


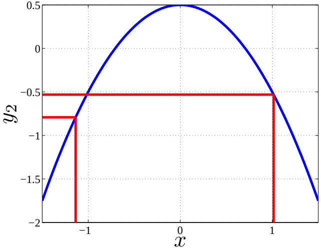
Back Constraints II

Forward Mapping (`demBackMapping` in oxford toolbox)

- Mapping from 1-D latent space to 2-D data space.

$$y_1 = x^2 - 0.5, \quad y_2 = -x^2 + 0.5$$



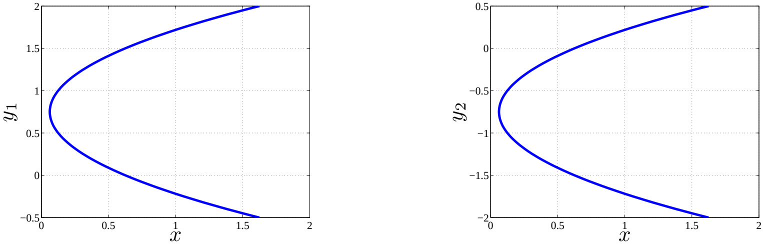


Back Constraints II

Backward Mapping (`demBackMapping` in oxford toolbox)

- Mapping from 2-D data space to 1-D latent.

$$x = 0.5\left(y_1^2 + y_2^2 + 1\right)$$

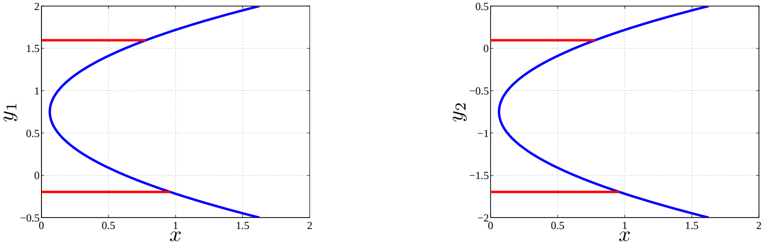


Back Constraints II

Backward Mapping (`demBackMapping` in oxford toolbox)

- Mapping from 2-D data space to 1-D latent.

$$x = 0.5\left(y_1^2 + y_2^2 + 1\right)$$

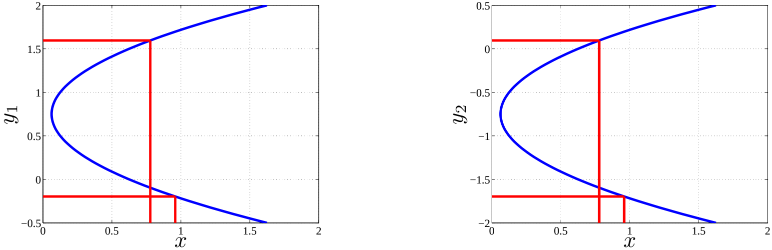


Back Constraints II

Backward Mapping (`demBackMapping` in oxford toolbox)

- Mapping from 2-D data space to 1-D latent.

$$x = 0.5\left(y_1^2 + y_2^2 + 1\right)$$



Multi-Dimensional Scaling with a Mapping

- Lowe and Tipping (1997) made latent positions a function of the data.

$$x_{i,j} = f_j\left(\mathbf{y}_{i,:}; \mathbf{v}\right)$$

- Function was either multi-layer perceptron or a radial basis function network.
- Their motivation was different from ours:
    - They wanted to add the advantages of a true mapping to multi-dimensional scaling.

Back Constraints in the GP-LVM

Back Constraints

- We can use the same idea to force the GP-LVM to respect local distances.(Lawrence and Quiñonero Candela, 2006)
  - By constraining each $\mathbf{x}_i$ to be a 'smooth' mapping from $\mathbf{y}_i$ local distances can be respected.
- This works because in the GP-LVM we maximise wrt latent variables, we don't integrate out.
- Can use any 'smooth' function:
  1. Neural network.
  2. RBF Network.
  3. Kernel based mapping.

Optimising BC-GPLVM

Computing Gradients

- GP-LVM normally proceeds by optimising

$$L(\mathbf{X}) = \log p(\mathbf{Y}|\mathbf{X})$$

with respect to $\mathbf{X}$ using $\frac{\mathrm{d}L}{\mathrm{d}\mathbf{X}}$.
- The back constraints are of the form

$$x_{i,j} = f_j(\mathbf{y}_{i,:}; \mathbf{v})$$

where $\mathbf{v}$ are parameters.
- We can compute $\frac{\mathrm{d}L}{\mathrm{d}\mathbf{v}}$ via chain rule and optimise parameters of mapping.

Motion Capture Results

`demStick1` **and** `demStick3`

Figure : The latent space for the motion capture data with (*right*) and without (*left*) back constraints.

Motion Capture Results

`demStick1` **and** `demStick3`

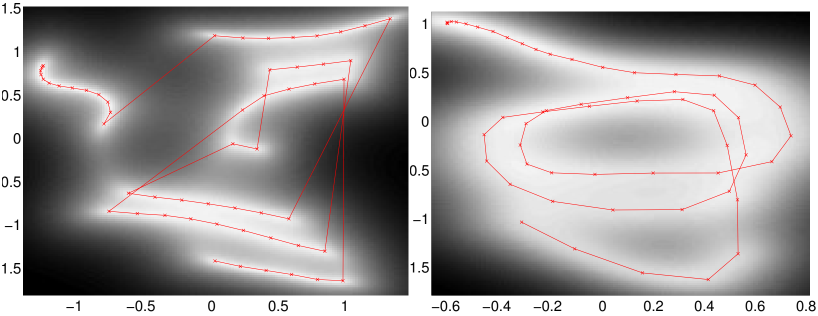
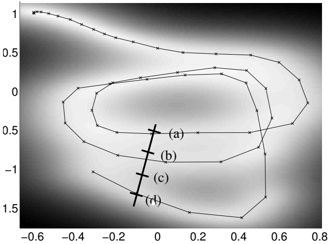


Figure : The latent space for the motion capture data with (*right*) and without (*left*) back constraints.

Stick Man Results

```
demStickResults
```



(a) (b) (c) (d)

Projection into data space from four points in the latent space. The inclination of the runner changes becoming more upright.

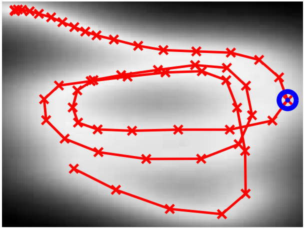
Adding Dynamics

MAP Solutions for Dynamics Models

- Data often has a temporal ordering.
- Markov-based dynamics are often used.
- For the GP-LVM
  - Marginalising such dynamics is intractable.
  - But: MAP solutions are trivial to implement.
- Many choices: Kalman filter, Markov chains *etc*..
- Wang et al. (2006) suggest using a Gaussian Process.

GP-LVM with Dynamics

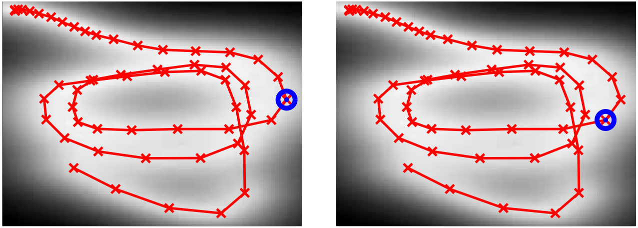
- Autoregressive Gaussian process mapping in latent space between time points.



t

GP-LVM with Dynamics

- Autoregressive Gaussian process mapping in latent space between time points.

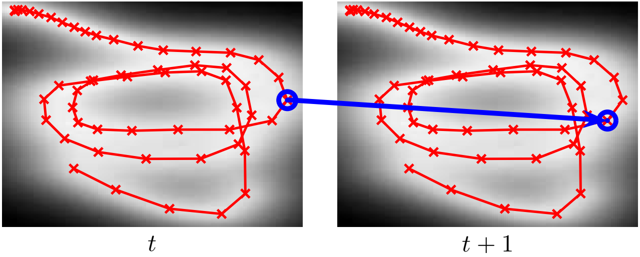


t $\quad$ $t+1$

Gaussian Process Dynamics

GP-LVM with Dynamics

- Autoregressive Gaussian process mapping in latent space between time points.



Motion Capture Results

`demStick1` **and** `demStick2`

Figure : The latent space for the motion capture data without dynamics (*left*), with auto-regressive dynamics (*right*) based on an exponentiated quadratic kernel.

Motion Capture Results

`demStick1` **and** `demStick2`

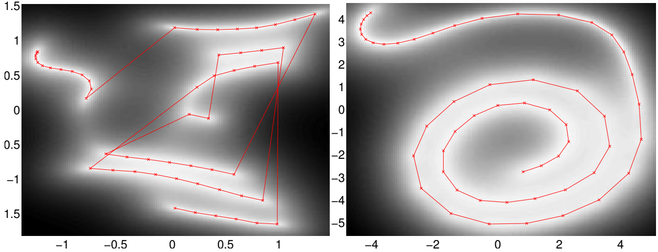
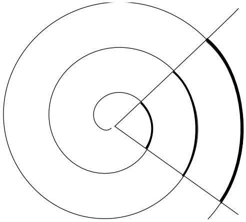


Figure : The latent space for the motion capture data without dynamics (*left*), with auto-regressive dynamics (*right*) based on an exponentiated quadratic kernel.

Regressive Dynamics

Inner Groove Distortion

- Autoregressive unimodal dynamics, $p\left(\mathbf{x}_t|\mathbf{x}_{t-1}\right)$.
- Forces spiral visualisation.
- Poorer model due to inner groove distortion.



Regressive Dynamics

Direct use of Time Variable

- Instead of auto-regressive dynamics, consider regressive dynamics.
- Take $\mathbf{t}$ as an input, use a prior $p\left(\mathbf{X}|\mathbf{t}\right)$.
- User a Gaussian process prior for $p\left(\mathbf{X}|\mathbf{t}\right)$.
- Also allows us to consider variable sample rate data.

Motion Capture Results

`demStick1`, `demStick2` **and** `demStick5`

Figure : The latent space for the motion capture data without dynamics (*left*), with auto-regressive dynamics (*middle*) and with regressive dynamics (*right*) based on an exponentiated quadratic kernel.

Motion Capture Results

`demStick1`, `demStick2` **and** `demStick5`

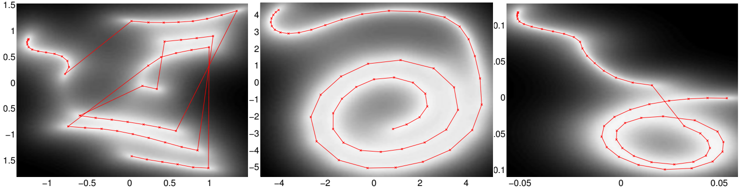


Figure : The latent space for the motion capture data without dynamics (*left*), with auto-regressive dynamics (*middle*) and with regressive dynamics (*right*) based on an exponentiated quadratic kernel.

Return

Hierarchical GP-LVM

(Lawrence and Moore, 2007)

Stacking Gaussian Processes

- Regressive dynamics provides a simple hierarchy.
  - The input space of the GP is governed by another GP.
- By stacking GPs we can consider more complex hierarchies.
- Ideally we should marginalise latent spaces
  - In practice we seek MAP solutions.

Two Correlated Subjects

(Lawrence and Moore, 2007)

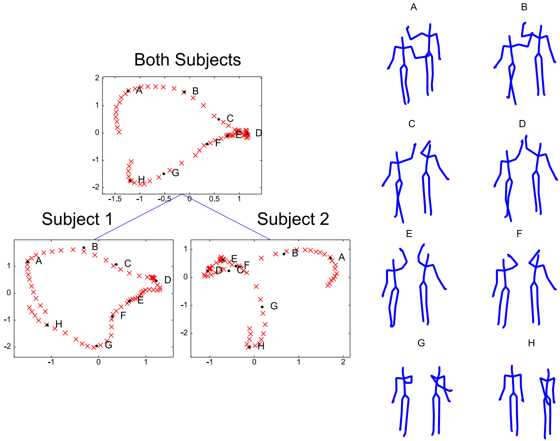


Figure : Hierarchical model of a 'high five'.

Within Subject Hierarchy

(Lawrence and Moore, 2007)

Decomposition of Body

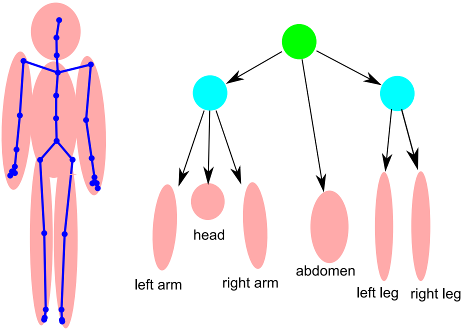


Figure : Decomposition of a subject.

Single Subject Run/Walk

(Lawrence and Moore, 2007)

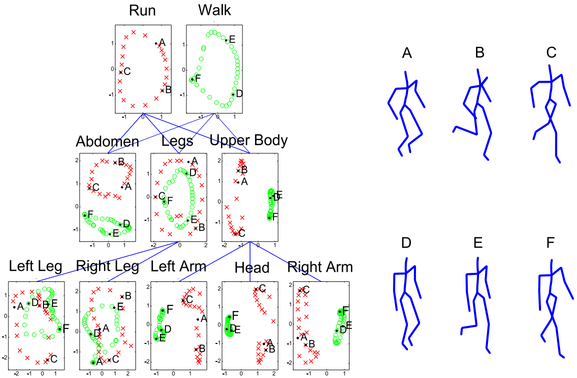


Figure : Hierarchical model of a walk and a run.

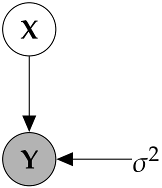
Selecting Data Dimensionality

- GP-LVM Provides probabilistic non-linear dimensionality reduction.
- How to select the dimensionality?
- Need to estimate marginal likelihood.
- In standard GP-LVM it increases with increasing q.

Integrate Mapping Function and Latent Variables

Bayesian GP-LVM

- Start with a standard GP-LVM.
- Apply standard latent variable approach:
    - Define Gaussian prior over *latent space*, $\mathbf{X}$.
    - Integrate out *latent variables*.
    - Unfortunately integration is intractable.

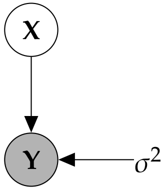


$$p\left(\mathbf{Y}|\mathbf{X}\right) = \prod_{j=1}^{p} \mathcal{N}\left(\mathbf{y}_{:,j}|\mathbf{0}, \mathbf{K}\right)$$

Integrate Mapping Function and Latent Variables

Bayesian GP-LVM

- Start with a standard GP-LVM.
- Apply standard latent variable approach:
  - Define Gaussian prior over *latent space*, $\mathbf{X}$.
  - Integrate out *latent variables*.
  - Unfortunately integration is intractable.

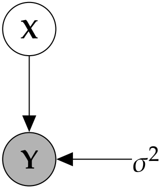


$$p(\mathbf{Y}|\mathbf{X}) = \prod_{j=1}^{p} \mathcal{N}\left(\mathbf{y}_{:,j}|\mathbf{0}, \mathbf{K}\right)$$

Integrate Mapping Function and Latent Variables

Bayesian GP-LVM

- Start with a standard GP-LVM.
- Apply standard latent variable approach:
  - Define Gaussian prior over *latent space*, $\mathbf{X}$.
  - Integrate out *latent variables*.
  - Unfortunately integration is intractable.



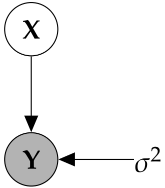
$$p(\mathbf{Y}|\mathbf{X}) = \prod_{j=1}^{p} \mathcal{N}\left(\mathbf{y}_{:,j}|\mathbf{0}, \mathbf{K}\right)$$

$$p(\mathbf{X}) = \prod_{j=1}^{q} \mathcal{N}\left(\mathbf{x}_{:,j}|\mathbf{0}, \alpha_i^{-2}\mathbf{I}\right)$$

Integrate Mapping Function and Latent Variables

Bayesian GP-LVM

- Start with a standard GP-LVM.
- Apply standard latent variable approach:
  - Define Gaussian prior over *latent space*, $\mathbf{X}$.
  - Integrate out *latent variables*.
  - Unfortunately integration is intractable.



$$p(\mathbf{Y}|\mathbf{X}) = \prod_{j=1}^{p} \mathcal{N}\left(\mathbf{y}_{:,j}|\mathbf{0}, \mathbf{K}\right)$$

$$p(\mathbf{X}) = \prod_{j=1}^{q} \mathcal{N}\left(\mathbf{x}_{:,j}|\mathbf{0}, \alpha_i^{-2}\mathbf{I}\right)$$

$$p(\mathbf{Y}|\boldsymbol{\alpha}) = ??$$

Standard Variational Approach Fails

- Standard variational bound has the form:

$$\mathcal{L} = \left\langle \log p(\mathbf{y}|\mathbf{X}) \right\rangle_{q(\mathbf{X})} + \mathrm{KL}\left(q(\mathbf{X}) \,\|\, p(\mathbf{X})\right)$$

- Requires expectation of $\log p(\mathbf{y}|\mathbf{X})$ under $q(\mathbf{X})$.

$$\log p(\mathbf{y}|\mathbf{X}) = -\frac{1}{2}\mathbf{y}^\top \left(\mathbf{K}_{\mathbf{f},\mathbf{f}} + \sigma^2\mathbf{I}\right)^{-1} \mathbf{y} - \frac{1}{2}\log\left|\mathbf{K}_{\mathbf{f},\mathbf{f}} + \sigma^2\mathbf{I}\right| - \frac{n}{2}\log 2\pi$$

- Extremely difficult to compute because $\mathbf{K}_{\mathbf{f},\mathbf{f}}$ is dependent on $\mathbf{X}$ and appears in the inverse.

Standard Variational Approach Fails

- Standard variational bound has the form:

$$\mathcal{L} = \left\langle \log p(\mathbf{y}|\mathbf{X}) \right\rangle_{q(\mathbf{X})} + \mathrm{KL}\left(q(\mathbf{X}) \,\|\, p(\mathbf{X})\right)$$

- Requires expectation of $\log p(\mathbf{y}|\mathbf{X})$ under $q(\mathbf{X})$.

$$\log p(\mathbf{y}|\mathbf{X}) = -\frac{1}{2}\mathbf{y}^\top \left(\mathbf{K}_{\mathbf{f},\mathbf{f}} + \sigma^2\mathbf{I}\right)^{-1}\mathbf{y} - \frac{1}{2}\log\left|\mathbf{K}_{\mathbf{f},\mathbf{f}} + \sigma^2\mathbf{I}\right| - \frac{n}{2}\log 2\pi$$

- Extremely difficult to compute because $\mathbf{K}_{\mathbf{f},\mathbf{f}}$ is dependent on $\mathbf{X}$ and appears in the inverse.

Standard Variational Approach Fails

- Standard variational bound has the form:

$$\mathcal{L} = \langle \log p(\mathbf{y}|\mathbf{X}) \rangle_{q(\mathbf{X})} + \mathrm{KL}\left(q(\mathbf{X}) \,\|\, p(\mathbf{X})\right)$$

- Requires expectation of $\log p(\mathbf{y}|\mathbf{X})$ under $q(\mathbf{X})$.

$$\log p(\mathbf{y}|\mathbf{X}) = -\frac{1}{2}\mathbf{y}^\top \left(\mathbf{K}_{\mathbf{f},\mathbf{f}} + \sigma^2\mathbf{I}\right)^{-1} \mathbf{y} - \frac{1}{2}\log\left|\mathbf{K}_{\mathbf{f},\mathbf{f}} + \sigma^2\mathbf{I}\right| - \frac{n}{2}\log 2\pi$$

- Extremely difficult to compute because $\mathbf{K}_{\mathbf{f},\mathbf{f}}$ is dependent on $\mathbf{X}$ and appears in the inverse.

Variational Bayesian GP-LVM

- Consider collapsed variational bound,

$$p(\mathbf{y}) \geq \prod_{i=1}^{n} c_i \int \mathcal{N}\left(\mathbf{y}| \left\langle \mathbf{f} \right\rangle, \sigma^2 \mathbf{I}\right) p(\mathbf{u}) \mathrm{d}\mathbf{u}$$

- Apply variational lower bound to the inner integral.
- Which is analytically tractable for Gaussian $q(\mathbf{X})$ and some covariance functions.

Variational Bayesian GP-LVM

- Consider collapsed variational bound,

$$p(\mathbf{y}|\mathbf{X}) \geq \prod_{i=1}^{n} c_i \int \mathcal{N}\left(\mathbf{y}|\left\langle\mathbf{f}\right\rangle_{p(\mathbf{f}|\mathbf{u},\mathbf{X})}, \sigma^2\mathbf{I}\right) p(\mathbf{u})\mathrm{d}\mathbf{u}$$

- Apply variational lower bound to the inner integral.
- Which is analytically tractable for Gaussian $q(\mathbf{X})$ and some covariance functions.

Variational Bayesian GP-LVM

- Consider collapsed variational bound,

$$\int p(\mathbf{y}|\mathbf{X})p(\mathbf{X})\mathrm{d}\mathbf{X} \geq \int \prod_{i=1}^{n} c_i \mathcal{N}\left(\mathbf{y}|\left\langle\mathbf{f}\right\rangle_{p(\mathbf{f}|\mathbf{u},\mathbf{X})}, \sigma^2\mathbf{I}\right) p(\mathbf{X})\mathrm{d}\mathbf{X}p(\mathbf{u})\mathrm{d}\mathbf{u}$$

- Apply variational lower bound to the inner integral.
- Which is analytically tractable for Gaussian $q(\mathbf{X})$ and some covariance functions.

Variational Bayesian GP-LVM

- Consider collapsed variational bound,

$$\int p(\mathbf{y}|\mathbf{X})p(\mathbf{X})\mathrm{d}\mathbf{X} \geq \int \prod_{i=1}^{n} c_i \mathcal{N}\left(\mathbf{y}|\left\langle \mathbf{f} \right\rangle_{p(\mathbf{f}|\mathbf{u},\mathbf{X})}, \sigma^2\mathbf{I}\right) p(\mathbf{X})\mathrm{d}\mathbf{X}p(\mathbf{u})\mathrm{d}\mathbf{u}$$

- Apply variational lower bound to the inner integral.
- Which is analytically tractable for Gaussian $q(\mathbf{X})$ and some covariance functions.

Variational Bayesian GP-LVM

- Consider collapsed variational bound,

$$\int p(\mathbf{y}|\mathbf{X})p(\mathbf{X})\mathrm{d}\mathbf{X} \geq \int \prod_{i=1}^{n} c_i \mathcal{N}\left(\mathbf{y}|\left\langle \mathbf{f}\right\rangle_{p(\mathbf{f}|\mathbf{u},\mathbf{X})}, \sigma^2\mathbf{I}\right) p(\mathbf{X})\mathrm{d}\mathbf{X} p(\mathbf{u})\mathrm{d}\mathbf{u}$$

- Apply variational lower bound to the inner integral.

$$\int \prod_{i=1}^{n} c_i \mathcal{N}\left(\mathbf{y}|\left\langle \mathbf{f}\right\rangle_{p(\mathbf{f}|\mathbf{u},\mathbf{X})}, \sigma^2\mathbf{I}\right) p(\mathbf{X})\mathrm{d}\mathbf{X}$$

$$\geq \left\langle \sum_{i=1}^{n} \log c_i \right\rangle_{q(\mathbf{X})}$$

$$+ \left\langle \log \mathcal{N}\left(\mathbf{y}|\left\langle \mathbf{f}\right\rangle_{p(\mathbf{f}|\mathbf{u},\mathbf{X})}, \sigma^2\mathbf{I}\right)\right\rangle_{q(\mathbf{X})}$$

$$+ \mathrm{KL}\left(q(\mathbf{X}) \,\|\, p(\mathbf{X})\right)$$

- Which is analytically tractable for Gaussian $q(\mathbf{X})$ and some covariance functions.

Variational Bayesian GP-LVM

- Consider collapsed variational bound,

$$\int p(\mathbf{y}|\mathbf{X})p(\mathbf{X})\mathrm{d}\mathbf{X} \geq \int \prod_{i=1}^{n} c_i \mathcal{N}\left(\mathbf{y}|\left\langle\mathbf{f}\right\rangle_{p(\mathbf{f}|\mathbf{u},\mathbf{X})}, \sigma^2\mathbf{I}\right) p(\mathbf{X})\mathrm{d}\mathbf{X} p(\mathbf{u})\mathrm{d}\mathbf{u}$$

- Apply variational lower bound to the inner integral.

$$\begin{aligned}\int \prod_{i=1}^{n} c_i \mathcal{N}\left(\mathbf{y}|\left\langle\mathbf{f}\right\rangle_{p(\mathbf{f}|\mathbf{u},\mathbf{X})}, \sigma^2\mathbf{I}\right)p(\mathbf{X})\mathrm{d}\mathbf{X} \\ \geq \left\langle\sum_{i=1}^{n} \log c_i\right\rangle_{q(\mathbf{X})} \\ + \left\langle\log \mathcal{N}\left(\mathbf{y}|\left\langle\mathbf{f}\right\rangle_{p(\mathbf{f}|\mathbf{u},\mathbf{X})}, \sigma^2\mathbf{I}\right)\right\rangle_{q(\mathbf{X})} \\ + \mathrm{KL}\left(q(\mathbf{X})\,\|\,p(\mathbf{X})\right)\end{aligned}$$

- Which is analytically tractable for Gaussian $q(\mathbf{X})$ and some covariance functions.

Required Expectations

- Need expectations under $q(\mathbf{X})$ of:

$$\log c_i = \frac{1}{2\sigma^2}\left[k_{i,i} - \mathbf{k}_{i,\mathbf{u}}^\top \mathbf{K}_{\mathbf{u},\mathbf{u}}^{-1} \mathbf{k}_{i,\mathbf{u}}\right]$$

and

$$\log \mathcal{N}\left(\mathbf{y}|\left\langle \mathbf{f} \right\rangle_{p(\mathbf{f}|\mathbf{u},\mathbf{Y})}, \sigma^2 \mathbf{I}\right) = -\frac{1}{2}\log 2\pi\sigma^2 - \frac{1}{2\sigma^2}\left(y_i - \mathbf{K}_{\mathbf{f},\mathbf{u}}\mathbf{K}_{\mathbf{u},\mathbf{u}}^{-1}\mathbf{u}\right)^2$$

- This requires the expectations

$$\left\langle \mathbf{K}_{\mathbf{f},\mathbf{u}} \right\rangle_{q(\mathbf{X})}$$

and

$$\left\langle \mathbf{K}_{\mathbf{f},\mathbf{u}}\mathbf{K}_{\mathbf{u},\mathbf{u}}^{-1}\mathbf{K}_{\mathbf{u},\mathbf{f}} \right\rangle_{q(\mathbf{X})}$$

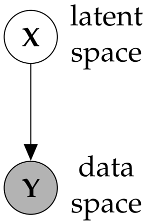
which can be computed analytically for some covariance functions.

Priors for Latent Space

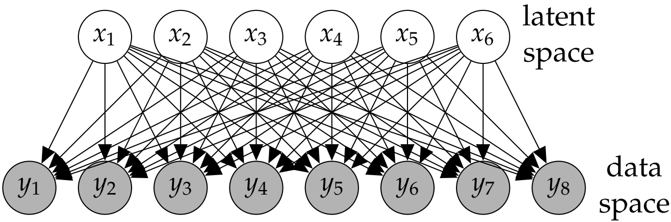
Titsias and Lawrence (2010)

- Variational marginalization of $\mathbf{X}$ allows us to learn parameters of $p(\mathbf{X})$.
- Standard GP-LVM where $\mathbf{X}$ learnt by MAP, this is not possible (see e.g. Wang et al., 2008).
- First example: learn the dimensionality of latent space.

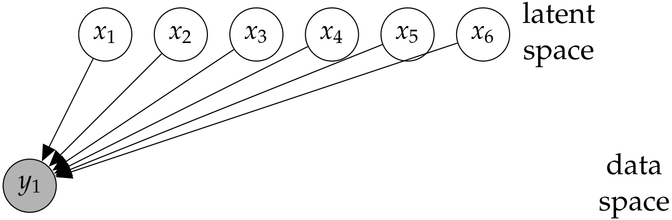
Graphical Representations of GP-LVM



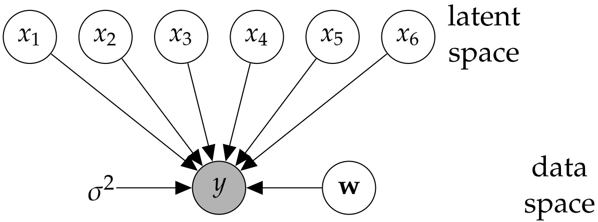
Graphical Representations of GP-LVM



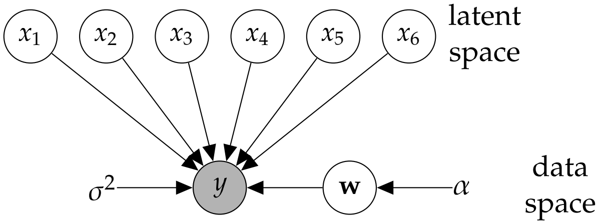
Graphical Representations of GP-LVM



Graphical Representations of GP-LVM



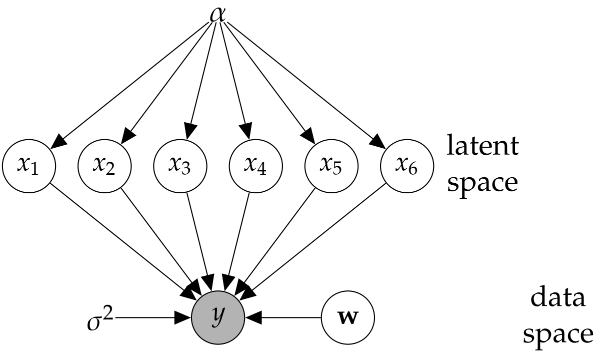
Graphical Representations of GP-LVM



$$\mathbf{w} \sim \mathcal{N}(\mathbf{0}, \alpha\mathbf{I}) \quad \mathbf{x} \sim \mathcal{N}(0, \mathbf{I})$$

$$y \sim \mathcal{N}\left(\mathbf{x}^\top \mathbf{w}, \sigma^2\right)$$

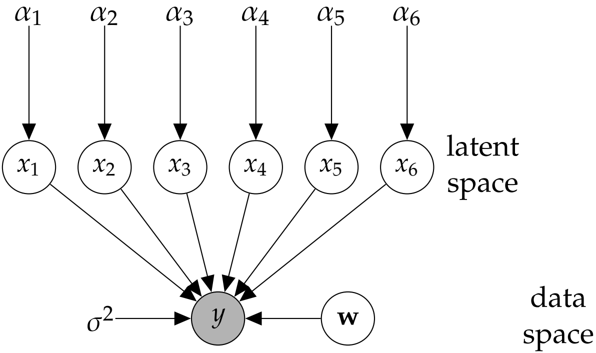
Graphical Representations of GP-LVM



$$\mathbf{w} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}) \quad \mathbf{x} \sim \mathcal{N}(0, \alpha\mathbf{I})$$

$$y \sim \mathcal{N}\left(\mathbf{x}^\top \mathbf{w}, \sigma^2\right)$$

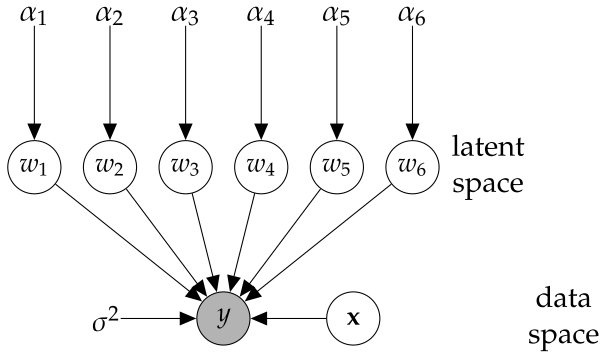
Graphical Representations of GP-LVM



$$\mathbf{w} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}) \qquad x_i \sim \mathcal{N}(0, \alpha_i)$$

$$y \sim \mathcal{N}\left(\mathbf{x}^\top \mathbf{w}, \sigma^2\right)$$

Graphical Representations of GP-LVM



$$w_i \sim \mathcal{N}(0, \alpha_i) \quad \mathbf{x} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$$

$$y \sim \mathcal{N}\left(\mathbf{x}^\top \mathbf{w}, \sigma^2\right)$$

Non-linear $f(\mathbf{x})$

- In linear case equivalence because $f(\mathbf{x}) = \mathbf{w}^\top \mathbf{x}$

$$p(w_i) \sim \mathcal{N}(\mathbf{0}, \alpha_i)$$

- In non linear case, need to scale columns of $\mathbf{X}$ in prior for $f(\mathbf{x})$.
- This implies scaling columns of $\mathbf{X}$ in covariance function

$$k(\mathbf{x}_{i,:}, \mathbf{x}_{j,:}) = \exp\left(-\frac{1}{2}(\mathbf{x}_{:,i} - \mathbf{x}_{:,j})^\top \mathbf{A}(\mathbf{x}_{:,i} - \mathbf{x}_{:,j})\right)$$

$\mathbf{A}$ is diagonal with elements α_i^2. Now keep prior spherical

$$p(\mathbf{X}) = \prod_{j=1}^{q} \mathcal{N}\left(\mathbf{x}_{:,j}|\mathbf{0}, \mathbf{I}\right)$$

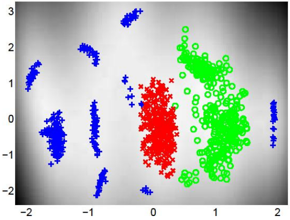
- Covariance functions of this type are known as ARD (see e.g. Neal, 1996; MacKay, 2003; Rasmussen and Williams, 2006).

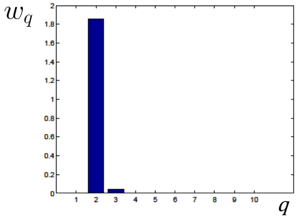
Automatic dimensionality detection

- Achieved by employing an *Automatic Relevance Determination (ARD)* covariance function for the prior on the GP mapping

- $f \sim GP(\mathbf{0}, k_f)$ with
$$k_f\left(\mathbf{x}_i, \mathbf{x}_j\right) = \sigma^2 \exp\left(-\frac{1}{2}\sum_{q=1}^{Q} w_q \left(x_{i,q} - x_{j,q}\right)^2\right)$$

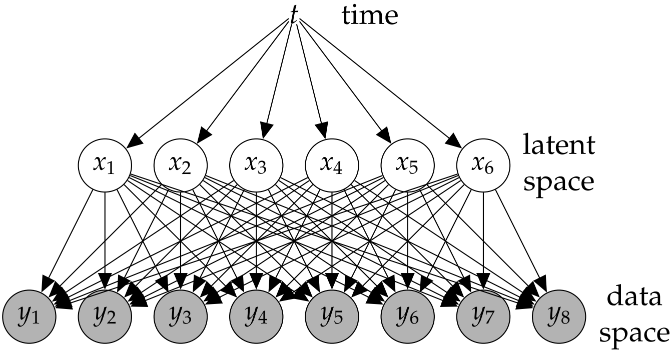
- Example





Gaussian Process Dynamical Systems

(Damianou et al., 2011)



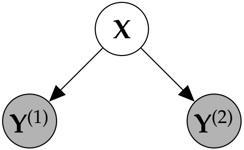
Gaussian Process over Latent Space

- Assume a GP prior for $p(\mathbf{X})$.
- Input to the process is time, $p(\mathbf{X}|t)$.

Interpolation of HD Video

Modeling Multiple 'Views'

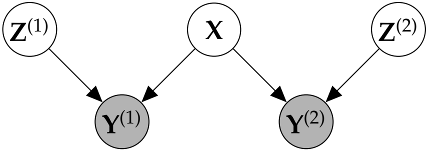
- Single space to model correlations between two different data sources, e.g., images & text, image & pose.
- Shared latent spaces: (Shon et al., 2006; Navaratnam et al., 2007; Ek et al., 2008b)



- Effective when the 'views' are correlated.
- But not all information is shared between both 'views'.
- PCA applied to concatenated data vs CCA applied to data.

Shared-Private Factorization

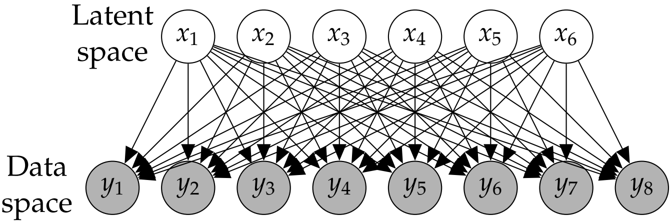
- In real scenarios, the 'views' are neither fully independent, nor fully correlated.
- Shared models
  - either allow information relevant to a single view to be mixed in the shared signal,
  - or are unable to model such private information.
- Solution: Model shared and private information (Virtanen et al., 2011; Ek et al., 2008a; Leen and Fyfe, 2006; Klami and Kaski, 2007, 2008; Tucker, 1958)



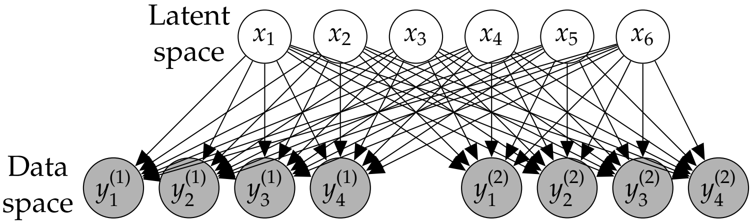
- Probabilistic CCA is case when dimensionality of $\mathbf{Z}$ matches $\mathbf{Y}^{(i)}$ (cf Inter Battery Factor Analysis (Tucker, 1958)).

Manifold Relevance Determination

Damianou et al. (2012)



Shared GP-LVM



Separate ARD parameters for mappings to $\mathbf{Y}^{(1)}$ and $\mathbf{Y}^{(2)}$.

Example: Yale faces

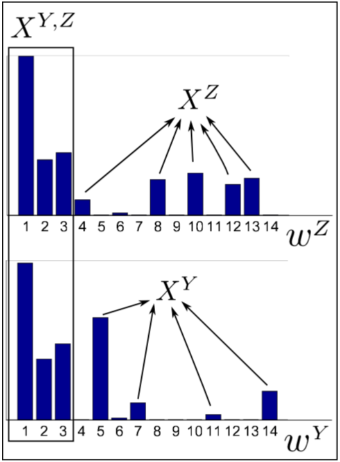
- Dataset Y: 3 persons under all illumination conditions
- Dataset Z: As above for 3 different persons
- Align datapoints $\mathbf{x}_n$ and $\mathbf{z}_n$ only based on the lighting direction

Results

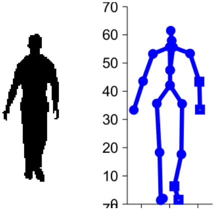
- Latent space X initialised with 14 dimensions

- Weights define a segmentation of X

- Video / demo…



Potential applications..?



Manifold Relevance Determination

References I

M. A. Álvarez and N. D. Lawrence. Sparse convolved Gaussian processes for multi-output regression. In Koller et al. (2009), pages 57–64. [PDF].

M. A. Álvarez, D. Luengo, and N. D. Lawrence. Latent force models. In van Dyk and Welling (2009), pages 9–16. [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 Teh and Titterington (2010), pages 25–32. [PDF].

O. Atteia, J.-P. Dubois, and R. Webster. Geostatistical analysis of soil contamination in the Swiss Jura. *Environ Pollut*, 86(3):315–327, 1994.

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.

W. V. Baxter and K.-I. Anjyo. Latent doodle space. In *EUROGRAPHICS*, volume 25, pages 477–485, Vienna, Austria, September 4-8 2006.

References II

T. Bayes. An essay towards solving a problem in the doctrine of chances. *Philosophical Transactions of the Royal Society*, 53:370–418, 1763. [DOI].

C. M. Bishop. *Pattern Recognition and Machine Learning*. Springer-Verlag, 2006. [Google Books] .

C. M. Bishop, M. Svensén, and C. K. I. Williams. GTM: the Generative Topographic Mapping. *Neural Computation*, 10(1):215–234, 1998. [DOI].

E. V. Bonilla, K. M. Chai, and C. K. I. Williams. Multi-task Gaussian process prediction. In J. C. Platt, D. Koller, Y. Singer, and S. Roweis, editors, *Advances in Neural Information Processing Systems*, volume 20, Cambridge, MA, 2008. MIT Press.

P. Boyle and M. Frean. Dependent Gaussian processes. In L. Saul, Y. Weiss, and L. Bouttou, editors, *Advances in Neural Information Processing Systems*, volume 17, pages 217–224, Cambridge, MA, 2005. MIT Press.

R. T. Cirz, J. K. Chin, D. R. Andes, V. de Crcy-Lagard, W. A. Craig, and F. E. Romesberg. Inhibition of mutation and combating the evolution of antibiotic resistance. *PLoS Biology*, 3(6), 2005.

S. Conti and A. O'Hagan. Bayesian emulation of complex multi-output and dynamic computer models. *Journal of Statistical Planning and Inference*, 140 (3):640–651, 2009. [DOI].

References III

J. Courcelle, A. Khodursky, B. Peter, P. O. Brown, , and P. C. Hanawalt. Comparative gene expression profiles following UV exposure in wild-type and SOS-deficient *Escherichia coli*. *Genetics*, 158:41–64, 2001.

L. Csató. *Gaussian Processes — Iterative Sparse Approximations*. PhD thesis, Aston University, 2002.

L. Csató and M. Opper. Sparse on-line Gaussian processes. *Neural Computation*, 14(3):641–668, 2002.

A. Damianou, C. H. Ek, M. K. Titsias, and N. D. Lawrence. Manifold relevance determination. In J. Langford and J. Pineau, editors, *Proceedings of the International Conference in Machine Learning*, volume 29, San Francisco, CA, 2012. Morgan Kauffman. [PDF].

A. Damianou, M. K. Titsias, and N. D. Lawrence. Variational Gaussian process dynamical systems. In P. Bartlett, F. Peirrera, C. Williams, and J. Lafferty, editors, *Advances in Neural Information Processing Systems*, volume 24, Cambridge, MA, 2011. MIT Press. [PDF].

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].

References IV

C. H. Ek, J. Rihan, P. Torr, G. Rogez, and N. D. Lawrence. Ambiguity modeling in latent spaces. In A. Popescu-Belis and R. Stiefelhagen, editors, *Machine Learning for Multimodal Interaction (MLMI 2008)*, LNCS, pages 62–73. Springer-Verlag, 28–30 June 2008a. [PDF].

C. H. Ek, P. H. Torr, and N. D. Lawrence. Gaussian process latent variable models for human pose estimation. In A. Popescu-Belis, S. Renals, and H. Bourlard, editors, *Machine Learning for Multimodal Interaction (MLMI 2007)*, volume 4892 of *LNCS*, pages 132–143, Brno, Czech Republic, 2008b. Springer-Verlag. [PDF].

Y. Gal, M. van der Wilk, and C. E. Rasmussen. Distributed variational inference in sparse Gaussian process regression and latent variable models. In Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 27, Cambridge, MA, 2014.

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].

Z. Ghahramani, editor. *Proceedings of the International Conference in Machine Learning*, volume 24, 2007. Omnipress. [Google Books] .

D. S. Goodsell. The molecular perspective: p53 tumor suppressor. *The Oncologist, Vol. 4, No. 2, 138-139, April 1999*, 4(2):138–139, 1999.

P. Goovaerts. *Geostatistics For Natural Resources Evaluation*. Oxford University Press, 1997. [Google Books] .

K. Grochow, S. L. Martin, A. Hertzmann, and Z. Popovic. Style-based inverse kinematics. In *ACM Transactions on Graphics (SIGGRAPH 2004)*, pages 522–531, 2004.

J. D. Helterbrand and N. A. C. Cressie. Universal cokriging under intrinsic coregionalization. *Mathematical Geology*, 26(2):205–226, 1994.

J. Hensman, N. Fusi, and N. D. Lawrence. Gaussian processes for big data. In A. Nicholson and P. Smyth, editors, *Uncertainty in Artificial Intelligence*, volume 29. AUAI Press, 2013. [PDF].

J. Hensman, M. Rattray, and N. D. Lawrence. Fast variational inference in the conjugate exponential family. In P. L. Bartlett, F. C. N. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 25, Cambridge, MA, 2012. [PDF].

D. M. Higdon. Space and space-time modelling using process convolutions. In C. Anderson, V. Barnett, P. Chatwin, and A. El-Shaarawi, editors, *Quantitative methods for current environmental issues*, pages 37–56. Springer-Verlag, 2002.

References VI

D. M. Higdon, J. Gattiker, B. Williams, and M. Rightley. Computer model calibration using high dimensional output. *Journal of the American Statistical Association*, 103(482):570–583, 2008.

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].

A. G. Journel and C. J. Huijbregts. *Mining Geostatistics*. Academic Press, London, 1978. [Google Books] .

A. A. Kalaitzis and N. D. Lawrence. A simple approach to ranking differentially expressed gene expression time courses through Gaussian process regression. *BMC Bioinformatics*, 12(180), 2011. [DOI].

R. Khanin, V. Viciotti, and E. Wit. Reconstructing repressor protein levels from expression of gene targets in *E. Coli*. *Proc. Natl. Acad. Sci. USA*, 103 (49):18592–18596, 2006. [DOI].

N. J. King and N. D. Lawrence. Fast variational inference for Gaussian Process models through KL-correction. In *ECML, Berlin, 2006*, Lecture Notes in Computer Science, pages 270–281, Berlin, 2006. Springer-Verlag. [PDF].

References VII

A. Klami and S. Kaski. Local dependent components analysis. In Ghahramani (2007). [Google Books] .

A. Klami and S. Kaski. Probabilistic approach to detecting dependencies between data sets. *Neurocomputing*, 72:39–46, 2008.

D. Koller, D. Schuurmans, Y. Bengio, and L. Bottou, editors. *Advances in Neural Information Processing Systems*, volume 21, Cambridge, MA, 2009. MIT Press.

J. B. Kruskal. Multidimensional scaling by optimizing goodness-of-fit to a nonmetric hypothesis. *Psychometrika*, 29(1):1–28, 1964. [DOI].

P. S. Laplace. Mémoire sur la probabilité des causes par les évènemens. In *Mémoires de mathèmatique et de physique, presentés à lAcadémie Royale des Sciences, par divers savans, & lù dans ses assemblées 6*, pages 621–656, 1774. Translated in Stigler (1986).

P. S. Laplace. *Essai philosophique sur les probabilités*. Courcier, Paris, 2nd edition, 1814. Sixth edition of 1840 translated and repreinted (1951) as *A Philosophical Essay on Probabilities*, New York: Dover; fifth edition of 1825 reprinted 1986 with notes by Bernard Bru, Paris: Christian Bourgois Éditeur, translated by Andrew Dale (1995) as *Philosophical Essay on Probabilities*, New York:Springer-Verlag.

References VIII

N. D. Lawrence. Gaussian process models for visualisation of high dimensional data. In S. Thrun, L. Saul, and B. Schölkopf, editors, *Advances in Neural Information Processing Systems*, volume 16, pages 329–336, Cambridge, MA, 2004. MIT Press.

N. D. Lawrence. Probabilistic non-linear principal component analysis with Gaussian process latent variable models. *Journal of Machine Learning Research*, 6:1783–1816, 11 2005.

N. D. Lawrence. Learning for larger datasets with the Gaussian process latent variable model. In M. Meila and X. Shen, editors, *Proceedings of the Eleventh International Workshop on Artificial Intelligence and Statistics*, pages 243–250, San Juan, Puerto Rico, 21-24 March 2007. Omnipress. [PDF].

N. D. Lawrence and A. J. Moore. Hierarchical Gaussian process latent variable models. In Ghahramani (2007), pages 481–488. [Google Books] . [PDF].

N. D. Lawrence and J. C. Platt. Learning to learn with the informative vector machine. In R. Greiner and D. Schuurmans, editors, *Proceedings of the International Conference in Machine Learning*, volume 21, pages 512–519. Omnipress, 2004. [PDF].

References IX

N. D. Lawrence and J. Quiñonero Candela. Local distance preservation in the GP-LVM through back constraints. In W. Cohen and A. Moore, editors, *Proceedings of the International Conference in Machine Learning*, volume 23, pages 513–520. Omnipress, 2006. [Google Books] . [PDF].

A. M. Lee, C. T. Ross, B.-B. Zeng, , and S. F. Singleton. A molecular target for suppression of the evolution of antibiotic resistance: Inhibition of the *Escherichia coli* RecA protein by N6-(1-Naphthyl)-ADP. *J. Med. Chem.*, 48 (17), 2005.

G. Leen and C. Fyfe. A Gaussian process latent variable model formulation of canonical correlation analysis. Bruges (Belgium), 26-28 April 2006 2006.

T. K. Leen, T. G. Dietterich, and V. Tresp, editors. *Advances in Neural Information Processing Systems*, volume 13, Cambridge, MA, 2001. MIT Press.

S. Levine, J. M. Wang, A. Haraux, Z. Popović, and V. Koltun. Continuous character control with low-dimensional embeddings. *ACM Transactions on Graphics (SIGGRAPH 2012)*, 31(4), 2012.

D. Lowe and M. E. Tipping. Neuroscale: Novel topographic feature extraction with radial basis function networks. In M. C. Mozer, M. I. Jordan, and T. Petsche, editors, *Advances in Neural Information Processing Systems*, volume 9, pages 543–549, Cambridge, MA, 1997. MIT Press.

C. Lu and X. Tang. Surpassing human-level face verification performance on LFW with GaussianFace. Technical report,

D. J. C. MacKay. Bayesian neural networks and density networks. *Nuclear Instruments and Methods in Physics Research, A*, 354(1):73–80, 1995. [DOI].

D. J. C. MacKay. *Information Theory, Inference and Learning Algorithms*. Cambridge University Press, Cambridge, U.K., 2003. [Google Books] .

K. V. Mardia, J. T. Kent, and J. M. Bibby. *Multivariate analysis*. Academic Press, London, 1979. [Google Books] .

T. P. Minka and R. W. Picard. Learning how to learn is learning with point sets. Available on-line., 1997. [URL]. Revised 1999, available at `http://www.stat.cmu.edu/~{}minka/`.

R. Navaratnam, A. Fitzgibbon, and R. Cipolla. The joint manifold model for semi-supervised multi-valued regression. In *IEEE International Conference on Computer Vision (ICCV)*. IEEE Computer Society Press, 2007.

R. M. Neal. *Bayesian Learning for Neural Networks*. Springer, 1996. Lecture Notes in Statistics 118.

J. Oakley and A. O'Hagan. Bayesian inference for the uncertainty distribution of computer model outputs. *Biometrika*, 89(4):769–784, 2002.

M. A. Osborne, A. Rogers, S. D. Ramchurn, S. J. Roberts, and N. R. Jennings. Towards real-time information processing of sensor network data using computationally efficient multi-output Gaussian processes. In *Proceedings of the International Conference on Information Processing in Sensor Networks (IPSN 2008)*, 2008.

A. D. Polyanin. *Handbook of Linear Partial Differential Equations for Engineers and Scientists*. Chapman & Hall/CRC, 1 edition, 2002.

V. Priacuriu and I. D. Reid. Nonlinear shape manifolds as shape priors in level set segmentation and tracking. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2011a.

V. Priacuriu and I. D. Reid. Shared shape spaces. In *IEEE International Conference on Computer Vision (ICCV)*, 2011b.

J. Quiñonero Candela and C. E. Rasmussen. A unifying view of sparse approximate Gaussian process regression. *Journal of Machine Learning Research*, 6:1939–1959, 2005.

C. E. Rasmussen and C. K. I. Williams. *Gaussian Processes for Machine Learning*. MIT Press, Cambridge, MA, 2006. [Google Books] .

S. Rogers and M. Girolami. *A First Course in Machine Learning*. CRC Press, 2011. [Google Books] .

References XII

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.

S. T. Roweis and L. K. Saul. Nonlinear dimensionality reduction by locally linear embedding. *Science*, 290(5500):2323–2326, 2000. [DOI].

J. W. Sammon. A nonlinear mapping for data structure analysis. *IEEE Transactions on Computers*, C-18(5):401–409, 1969. [DOI].

B. Schölkopf, A. Smola, and K.-R. Müller. Nonlinear component analysis as a kernel eigenvalue problem. *Neural Computation*, 10:1299–1319, 1998. [DOI].

M. Seeger and M. I. Jordan. Sparse Gaussian Process Classification With Multiple Classes. Technical Report 661, Department of Statistics, University of California at Berkeley,

M. Seeger, C. K. I. Williams, and N. D. Lawrence. Fast forward selection to speed up sparse Gaussian process regression. In C. M. Bishop and B. J. Frey, editors, *Proceedings of the Ninth International Workshop on Artificial Intelligence and Statistics*, Key West, FL, 3–6 Jan 2003.

A. P. Shon, K. Grochow, A. Hertzmann, and R. P. N. Rao. Learning shared latent structure for image synthesis and robotic imitation. In Weiss et al. (2006).

G. Skolidis and G. Sanguinetti. Bayesian multitask classification with Gaussian process priors. *IEEE Transactions on Neural Networks*, 22(12):2011 – 2021, 2011.

A. J. Smola and P. L. Bartlett. Sparse greedy Gaussian process regression. In Leen et al. (2001), pages 619–625.

E. Snelson and Z. Ghahramani. Sparse Gaussian processes using pseudo-inputs. In Weiss et al. (2006).

M. L. Stein. *Interpolation of Spatial Data: Some Theory for Kriging*. Springer-Verlag, 1999. [Google Books] .

S. M. Stigler. Laplace's 1774 memoir on inverse probability. *Statistical Science*, 1:359–378, 1986.

Y. W. Teh, M. Seeger, and M. I. Jordan. Semiparametric latent factor models. In R. G. Cowell and Z. Ghahramani, editors, *Proceedings of the Tenth International Workshop on Artificial Intelligence and Statistics*, pages 333–340, Barbados, 6-8 January 2005. Society for Artificial Intelligence and Statistics.

Y. W. Teh and D. M. Titterington, editors. *Artificial Intelligence and Statistics*, volume 9, Chia Laguna Resort, Sardinia, Italy, 13-16 May 2010. JMLR W&CP 9.

References XIV

J. B. Tenenbaum, V. de Silva, and J. C. Langford. A global geometric framework for nonlinear dimensionality reduction. *Science*, 290(5500): 2319–2323, 2000. [DOI].

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].

M. K. Titsias. Variational learning of inducing variables in sparse Gaussian processes. In van Dyk and Welling (2009), pages 567–574.

M. K. Titsias, A. Honkela, N. D. Lawrence, and M. Rattray. Identifying targets of multiple co-regulated transcription factors from expression time-series by Bayesian model comparison. *BMC Systems Biology*, 6(53), 2012. [DOI].

M. K. Titsias and N. D. Lawrence. Bayesian Gaussian process latent variable model. In Teh and Titterington (2010), pages 844–851. [PDF].

M. K. Titsias, N. D. Lawrence, and M. Rattray. Efficient sampling for Gaussian process inference using control variables. In Koller et al. (2009), pages 1681–1688. [PDF].

P. Tomancak, A. Beaton, R. Weiszmann, E. Kwan, S. Shu, S. E. Lewis, S. Richards, M. Ashburner, V. Hartenstein, S. E. Celniker, and G. M. Rubin. Systematic determination of patterns of gene expression during Drosophila embryogenesis. *Genome Biology*, 3(12):RESEARCH0088, 2002.

L. R. Tucker. An inter-battery method of factor analysis. *Psychometrika*, 23(2): 111–136, 1958.

R. Urtasun and T. Darrell. Discriminative Gaussian process latent variable model for classification. In Ghahramani (2007). [Google Books] .

R. Urtasun, D. J. Fleet, and P. Fua. 3D people tracking with Gaussian process dynamical models. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 238–245, New York, U.S.A., 17–22 Jun. 2006. IEEE Computer Society Press.

R. Urtasun, D. J. Fleet, A. Hertzmann, and P. Fua. Priors for people tracking from small training sets. In *IEEE International Conference on Computer Vision (ICCV)*, pages 403–410, Bejing, China, 17–21 Oct. 2005. IEEE Computer Society Press.

D. van Dyk and M. Welling, editors. *Artificial Intelligence and Statistics*, volume 5, Clearwater Beach, FL, 16-18 April 2009. JMLR W&CP 5.

S. Virtanen, A. Klami, and S. Kaski. Bayesian CCA via group sparsity. In L. Getoor and T. Scheffer, editors, *Proceedings of the International Conference in Machine Learning*, volume 28, 2011.

H. Wackernagel. *Multivariate Geostatistics: An Introduction With Applications*. Springer-Verlag, 3rd edition, 2003. [Google Books] .

References XVI

J. M. Wang, D. J. Fleet, and A. Hertzmann. Gaussian process dynamical models. In Weiss et al. (2006).

J. M. Wang, D. J. Fleet, and A. Hertzmann. Gaussian process dynamical models for human motion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(2):283–298, 2008. ISSN 0162-8828. [DOI].

Y. Weiss, B. Schölkopf, and J. C. Platt, editors. *Advances in Neural Information Processing Systems*, volume 18, Cambridge, MA, 2006. MIT Press.

C. K. I. Williams. Computation with infinite neural networks. *Neural Computation*, 10(5):1203–1216, 1998.

C. K. I. Williams and M. Seeger. Using the Nyström method to speed up kernel machines. In Leen et al. (2001), pages 682–688.

I. Woodward, M. R. Lomas, and R. A. Betts. Vegetation-climate feedbacks in a greenhouse world. *Philosophical Transactions: Biological Sciences*, 353(1365): 29–39, 1998.

K. Yu, V. Tresp, and A. Schwaighofer. Learning Gaussian processes from multiple tasks. In *Proceedings of the 22nd International Conference on Machine Learning (ICML 2005)*, pages 1012–1019, 2005.

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].