CS-E4075 Special course on Gaussian processes: Session #5 Latent variable models

Charles Gadd

Aalto University

charles.gadd@aalto.fi

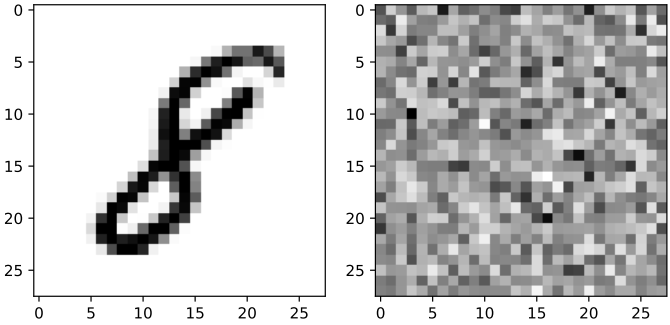
Monday 25.01.2021

Agenda for today

- Introduction
  - Why are LVMs useful?
  - Definition of LVMs
- Gaussian process latent variable models
  - Principal Component Analysis
  - Probabilistic PCA
  - Dual probabilistic PCA
  - GPLVM
- Multi-output models
  - Intrinsic Model of Coregionalisation
  - Semiparametric Latent Factor Model
  - Linear Model of Coregionalisation

Why are LVMs useful?

- Data has structure



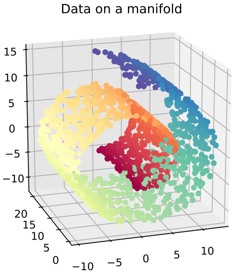
- ... the dimension of this space is large ($D = 784$)
- ... you would never sample this digit randomly

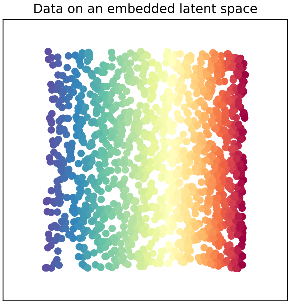
Why are LVMs useful?

- These samples lie on a **very** narrow manifold in $\mathbb{R}^{28\times 28}$
- We should only require enough dimensions to describe the digit sufficiently
  - e.g. shape and distortions (rotation, translation, stretching)
- The number of these dimensions is called the *intrinsic dimensionality* and is often significantly smaller than the number of features.
- Its often far easier to perform your inference on this embedded manifold

Swiss roll example

Moving from $\mathbb{R}^3$ to $\mathbb{R}^2$





Some notation: Features $\mathrm{Y} \in \mathbb{R}^{n \times D}$, latent variables $\mathrm{X} \in \mathbb{R}^{n \times d}$

Definition of LVMs

Definition: Dimensionality reduction - Learning a projection onto a lower dimensional embedding.

Definition: Manifold learning - Learning this embedding **and** a *pre-image map* $g : \mathrm{x} \rightarrow y$

A latent variable model is of the form:

$$\mathrm{y} = g\left(\mathrm{x}\right) + \epsilon$$

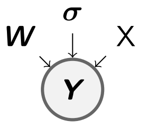
Often such models make assumptions of

- Independence across latent samples
- Conditional independence across features **given** latent samples

The Gaussian process latent variable model (GPLVM)

Principal Component Analysis (Recap)
Formulating GPLVM

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

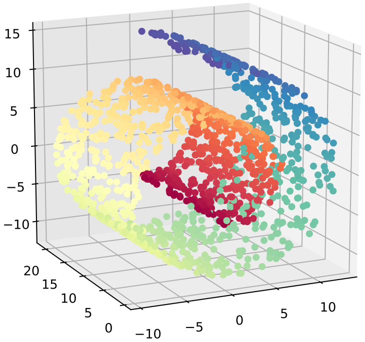


- Linear projection
- ... that projects to a new co-ordinate system
- ... such that the new basis is comprised of *principal components*
- ... which span the directions of greatest variance

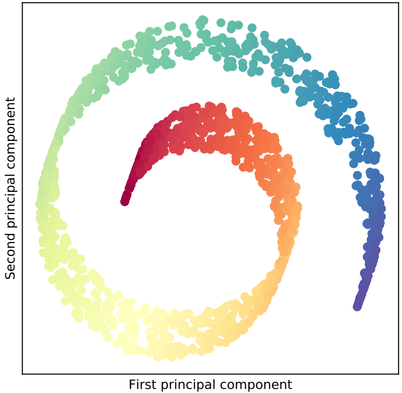
Only works well if data lies on a plane in high dimensional space

No representation of uncertainty

Principal Component Analysis

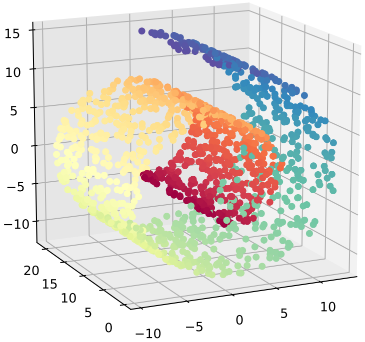


(a) Swiss roll data

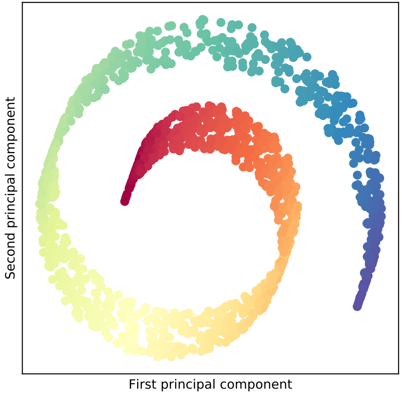


(b) PCA embedding

Principal Component Analysis



(a) Swiss roll data



(b) PCA embedding

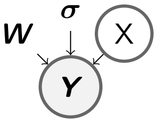
This is not optimal. This embedding does not capture all of the variance!

Probabilistic PCA

Formulating GPLVM

Likelihood

$$p\left(\boldsymbol{Y} \middle| \boldsymbol{W}, \mathrm{X}, \sigma\right) \sim \prod_{i=1}^{n} \mathcal{N}\left(\boldsymbol{y}_i \middle| \boldsymbol{W}\,\mathrm{x}_i, \sigma^2 \boldsymbol{I}\right)$$

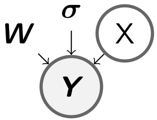


Probabilistic PCA

Formulating GPLVM

Likelihood

$$p\left(\boldsymbol{Y} | \boldsymbol{W}, \mathrm{X}, \sigma\right) \sim \prod_{i=1}^{n} \mathcal{N}\left(\boldsymbol{y}_i | \boldsymbol{W}\,\mathrm{x}_i, \sigma^2 \boldsymbol{I}\right)$$

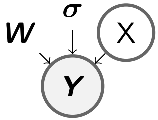


Place a conjugate Gaussian prior over latent space $\mathcal{Z}$

$$p\left(\mathrm{X}\right) \sim \prod_{i=1}^{n} \mathcal{N}\left(\mathrm{x}_i \,| 0, \boldsymbol{I}\right)$$

Likelihood

$$p\left(\boldsymbol{Y}|\boldsymbol{W},\mathrm{X},\sigma\right) \sim \prod_{i=1}^{n} \mathcal{N}\left(\boldsymbol{y}_i|\boldsymbol{W}\mathrm{x}_i,\sigma^2\boldsymbol{I}\right)$$



Place a conjugate Gaussian prior over latent space $\mathcal{Z}$

$$p\left(\mathrm{X}\right) \sim \prod_{i=1}^{n} \mathcal{N}\left(\mathrm{x}_i\,|0,\boldsymbol{I}\right)$$

... and integrate over latent variables $\boldsymbol{z}$ to obtain the marginal likelihood

$$p\left(\boldsymbol{Y}|\boldsymbol{W},\sigma\right) = \prod_{i=1}^{n} \int p\left(\boldsymbol{y}_i|\boldsymbol{W},\mathrm{x}_i,\sigma\right) p\left(\mathrm{x}_i\right) d\,\mathrm{x}$$

Marginal likelihood

$$p\left(\boldsymbol{Y}|\boldsymbol{W},\sigma\right)=\prod_{i=1}^{n}\int p\left(\boldsymbol{y}_i|\boldsymbol{W},\mathrm{x}_i,\sigma\right)p\left(\mathrm{x}_i\right)d\,\mathrm{x}$$

Using scaling and summation results

$$\boxed{\alpha\mathcal{N}\left(\mu,\Sigma^2\right)=\mathcal{N}\left(\alpha\mu,\alpha^2\Sigma^2\right)}\qquad\boxed{\sum_i\mathcal{N}\left(\mu_i,\Sigma_i^2\right)=\mathcal{N}\left(\sum_i\mu_i,\sum_i\Sigma_i^2\right)}$$

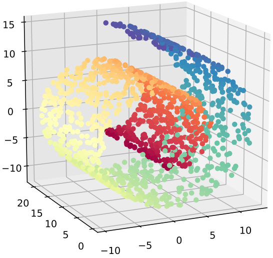
we can derive this

$$p\left(\boldsymbol{Y}|\boldsymbol{W},\sigma\right)\sim\prod_{i=1}^{n}\int\mathcal{N}\left(\boldsymbol{y}_i|\boldsymbol{W}\,\mathrm{x}_i,\sigma^2\boldsymbol{I}\right)\mathcal{N}\left(\mathrm{x}_i\,|0,\boldsymbol{I}\right)d\,\mathrm{x}_i$$

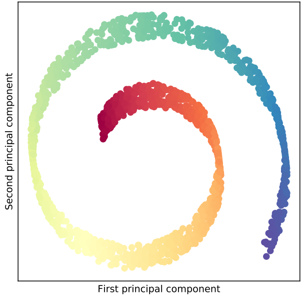
$$p\left(\boldsymbol{Y}|\boldsymbol{W},\sigma\right)\sim\prod_{i=1}^{n}\mathcal{N}\left(\boldsymbol{y}_i|0,\boldsymbol{W}\boldsymbol{W}^T+\sigma^2\boldsymbol{I}\right)$$

Probabilistic PCA - Swiss roll example

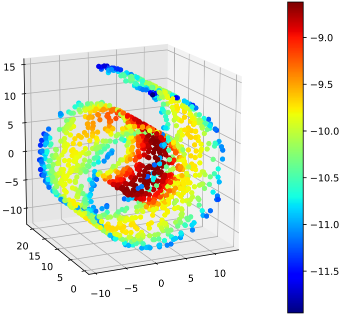
Formulating GPLVM



(a) Swiss roll data



(b) PCA embedding



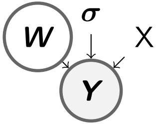
(c) Log-likelihood

Dual PPCA

Formulating GPLVM

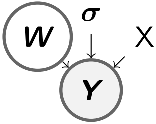
Likelihood

$$p\left(\boldsymbol{Y} \middle| \boldsymbol{W}, \mathrm{X}, \sigma\right) \sim \prod_{d=1}^{D} \mathcal{N}\left(\boldsymbol{y}_i \middle| \boldsymbol{W}\mathrm{X}, \sigma^2 \boldsymbol{I}\right)$$



Likelihood

$$p\left(\boldsymbol{Y}|\boldsymbol{W}, \mathrm{X}, \sigma\right) \sim \prod_{d=1}^{D} \mathcal{N}\left(\boldsymbol{y}_i|\boldsymbol{W}\mathrm{X}, \sigma^2\boldsymbol{I}\right)$$



Place a conjugate Gaussian prior over the space of linear transformations

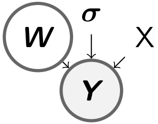
$$p\left(\boldsymbol{W}\right) \sim \prod_{i=1}^{D} \mathcal{N}\left(\boldsymbol{w}_i|0, \boldsymbol{I}\right)$$

Dual PPCA

Formulating GPLVM

Likelihood

$$p\left(\boldsymbol{Y}|\boldsymbol{W}, \mathrm{X}, \sigma\right) \sim \prod_{d=1}^{D} \mathcal{N}\left(\boldsymbol{y}_i|\boldsymbol{W}\mathrm{X}, \sigma^2 \boldsymbol{I}\right)$$



Place a conjugate Gaussian prior over the space of linear transformations

$$p\left(\boldsymbol{W}\right) \sim \prod_{i=1}^{D} \mathcal{N}\left(\boldsymbol{w}_i|0, \boldsymbol{I}\right)$$

... and integrate over transformation matrix $\boldsymbol{W}$ to obtain the marginal likelihood

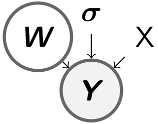
$$p\left(\boldsymbol{Y}|\mathrm{X}, \sigma\right) = \prod_{d=1}^{D} \int p\left(\boldsymbol{y}_i|\boldsymbol{W}, \mathrm{X}, \sigma\right) p\left(\boldsymbol{W}\right) dW$$

Dual PPCA

Formulating GPLVM

Likelihood

$$p\left(\boldsymbol{Y} | \boldsymbol{W}, \mathrm{X}, \sigma\right) \sim \prod_{d=1}^{D} \mathcal{N}\left(\boldsymbol{y}_i | \boldsymbol{W}\mathrm{X}, \sigma^2 \boldsymbol{I}\right)$$



Place a conjugate Gaussian prior over the space of linear transformations

$$p\left(\boldsymbol{W}\right) \sim \prod_{i=1}^{D} \mathcal{N}\left(\boldsymbol{w}_i | 0, \boldsymbol{I}\right)$$

... and integrate over transformation matrix $\boldsymbol{W}$ to obtain the marginal likelihood

$$p\left(\boldsymbol{Y} | \mathrm{X}, \sigma\right) = \prod_{i=1}^{D} \mathcal{N}\left(\boldsymbol{y}_i | 0, \mathrm{X}\mathrm{X}^T + \sigma^2 \boldsymbol{I}\right)$$

The dual PPCA marginal likelihood

$$p(\boldsymbol{Y}|\mathrm{X},\sigma) = \prod_{i=1}^{D} \mathcal{N}\left(\boldsymbol{y}_i|0, \mathrm{X}\mathrm{X}^T + \sigma^2 \boldsymbol{I}\right)$$

The dual PPCA marginal likelihood

$$p\left(\boldsymbol{Y} \middle| \mathrm{X}, \sigma\right) = \prod_{i=1}^{D} \mathcal{N}\left(\boldsymbol{y}_i | 0, \mathrm{X}\mathrm{X}^T + \sigma^2 \boldsymbol{I}\right)$$

The linear kernel

$$k\left(\mathrm{x}, \mathrm{x}'\right) = \theta_b^2 + \theta_v^2 (\mathrm{x} - \mathrm{c})\left(\mathrm{x} - \mathrm{c}\right)^T$$

The dual PPCA marginal likelihood

$$p\left(\boldsymbol{Y} \mid \mathrm{X}, \sigma\right) = \prod_{i=1}^{D} \mathcal{N}\left(\boldsymbol{y}_i \middle| 0, \mathrm{X}\mathrm{X}^T + \sigma^2 \boldsymbol{I}\right)$$

The linear kernel

$$k\left(\mathrm{x}, \mathrm{x}'\right) = \theta_b^2 + \theta_v^2 \left(\mathrm{x} - \mathrm{c}\right)\left(\mathrm{x} - \mathrm{c}\right)^T$$

The marginal likelihood for DPPCA is a product of D independent Gaussian processes with a linear kernel

The kernel
Formulating GPLVM

The dual PPCA marginal likelihood

$$p\left(\boldsymbol{Y} \mid \mathrm{X}, \sigma\right) = \prod_{i=1}^{D} \mathcal{N}\left(\boldsymbol{y}_i \middle| 0, \mathrm{X}\mathrm{X}^T + \sigma^2 \boldsymbol{I}\right)$$

The linear kernel

$$k\left(\mathrm{x}, \mathrm{x}'\right) = \theta_b^2 + \theta_v^2 (\mathrm{x} - \mathrm{c})\left(\mathrm{x} - \mathrm{c}\right)^T$$

The marginal likelihood for DPPCA is a product of D independent Gaussian processes with a linear kernel

We can change this kernel and obtain the GPLVM class of models

GPLVM summary

- DPPCA is a special case of GPLVM with a linear kernel
- Each dimension of the marginal can be interpreted as an independent GP
- Each dimension is *a priori* assumed independent, and identically distributed

GPLVM inference

Analytic solutions

- For PCA, PPCA and DPPCA, an analytic solution exists via solving an eigenvalue problem

GPLVM inference

Analytic solutions

- For PCA, PPCA and DPPCA, an analytic solution exists via solving an eigenvalue problem

Maximum likelihood (ML)

- Once we use a non-linear kernel analytical solutions often become intractable
- Instead we may resort to gradient based optimisation for $(\mathrm{X}, \theta, \sigma)$.[1]

$$\hat{\mathrm{X}}, \hat{\theta}, \hat{\sigma} = \underset{\mathrm{X},\theta,\sigma}{\arg\max} \left\{ \log\left(\boldsymbol{Y} \middle| \mathrm{X}, \theta, \sigma \right) \right\} \propto \underset{\mathrm{X},\theta,\sigma}{\arg\max} \left\{ -\frac{D}{2} \log |\boldsymbol{K}| - \frac{1}{2} \mathrm{tr}\left(\boldsymbol{K}^{-1} \boldsymbol{Y} \boldsymbol{Y}^T \right) \right\}$$

[1] where θ are the kernel hyper-parameters and, for example, $\boldsymbol{K} = \mathrm{X}\mathrm{X}^T + \sigma^2 \boldsymbol{I}$ in the linear kernel case

GPLVM inference - MAP

Maximum a-posteriori (MAP)

- Place a prior on latent variables Z

$$\hat{\mathrm{X}}, \hat{\theta}, \hat{\sigma} = \underset{\mathrm{X},\theta,\sigma}{\arg\max} \left\{ \log \left(\boldsymbol{Y} \middle| \mathrm{X}, \theta, \sigma \right) + \log p \left(\mathrm{X} \right) \right\}$$

- We again use gradient based optimisation
- This prior acts to regularise the latent variables

These both optimise over a huge space, but don't do as poorly as you'd expect!

GPLVM - Caveats and practical points

- Optimisation is *very* non-convex.
  - Multiple restarts, initialisation. How do we initialise?
- What is the latent dimensionality?
- Computational cost.

Bayesian GPLVM

- We may also want to place a prior on Z and integrate it out. [2]

$$p\left(\boldsymbol{Y} \mid \mathrm{X}\right) = \prod_{i=1}^{D} \mathcal{N}\left(\boldsymbol{y}_i \mid 0, \boldsymbol{K}\right)$$

$$p\left(\boldsymbol{Y}\right) = \int p\left(\boldsymbol{Y} \mid \mathrm{X}\right) p\left(\mathrm{X}\right) d\,\mathrm{X} \quad \text{and introduce} \quad p\left(\mathrm{X}\right) = \prod_{i=1}^{n} \mathcal{N}\left(\mathrm{x}_i \mid 0, \boldsymbol{I}_d\right)$$

- This is intractable as X appears non-linearly in the inverse of the kernel
- Lets try and apply the standard *variational Bayes* approach

[2] Dropping dependence on θ and σ for clarity

Bayesian GPLVM

Introduce a variational distribution

$$p\left(X|\boldsymbol{Y}\right) \approx q\left(\mathrm{X}\right) = \prod_{i=1}^{n} \mathcal{N}\left(\mathrm{x}_m \,|\mu_n, S_n\right)$$

And compute the Jensen's lower bound

$$\log p\left(\boldsymbol{Y}\right) \geq \underbrace{\sum_{i=1}^{D} \int q\left(\mathrm{X}\right) \log p\left(\boldsymbol{y}_i|\,\mathrm{X}\right) d\,\mathrm{X}}_{\text{this remains intractable}} - \underbrace{\int q\left(\mathrm{X}\right) \log \frac{q\left(\mathrm{X}\right)}{p\left(\mathrm{X}\right)} d\,\mathrm{X}}_{KL(q||p)}$$

Bayesian GPLVM

Introduce a variational distribution

$$p\left(X|\boldsymbol{Y}\right) \approx q\left(\mathrm{X}\right) = \prod_{i=1}^{n} \mathcal{N}\left(\mathrm{x}_m \,|\mu_n, S_n\right)$$

And compute the Jensen's lower bound

$$\log p\left(\boldsymbol{Y}\right) \geq \underbrace{\sum_{i=1}^{D} \int q\left(\mathrm{X}\right) \log p\left(\boldsymbol{y}_i | \,\mathrm{X}\right) d\,\mathrm{X}}_{\text{this remains intractable}} - \underbrace{\int q\left(\mathrm{X}\right) \log \frac{q\left(\mathrm{X}\right)}{p\left(\mathrm{X}\right)} d\,\mathrm{X}}_{KL(q||p)}$$

Lets apply the variational sparse methodology of [Titsias(2009)] that we learnt last lecture! [3]

[3] Slide 23 of session 4

Bayesian GPLVM (revisiting the variational sparse approach)

Lets expand our intractable integral term and augment inducing variables[4]

$$p\left(\boldsymbol{y}_i, \boldsymbol{f}_i, \mathrm{u}_i \,|\, \mathrm{X}, \mathrm{Z}\right) = p\left(\boldsymbol{y}_i | \boldsymbol{f}_i\right) p\left(\boldsymbol{f}_i | \, \mathrm{u}_i, \mathrm{X}, \mathrm{Z}\right) p\left(\mathrm{u}_i \,|\, \mathrm{Z}\right)$$

[4]**Notation reminder:** inducing inputs Z, inducing outputs U.

Bayesian GPLVM (revisiting the variational sparse approach)

Lets expand our intractable integral term and augment inducing variables[4]

$$p\left(\boldsymbol{y}_i, \boldsymbol{f}_i, \mathrm{u}_i \,|\, \mathrm{X}, \mathrm{Z}\right) = p\left(\boldsymbol{y}_i | \boldsymbol{f}_i\right) p\left(\boldsymbol{f}_i | \, \mathrm{u}_i, \mathrm{X}, \mathrm{Z}\right) p\left(\mathrm{u}_i \,|\, \mathrm{Z}\right)$$

Derive a variational approximation for the posterior

$$p\left(\boldsymbol{f}_i, \mathrm{u}_i \,|\, \boldsymbol{Y}, \mathrm{X}, \mathrm{Z}\right) \approx q\left(\boldsymbol{f}_i, \mathrm{u}_i\right) = p\left(\boldsymbol{f}_i | \, \mathrm{u}_i, \mathrm{X}, \mathrm{Z}\right) q\left(\mathrm{u}_i\right)$$

[4] **Notation reminder:** inducing inputs Z, inducing outputs U.

Bayesian GPLVM (revisiting the variational sparse approach)

Lets expand our intractable integral term and augment inducing variables[4]

$$p\left(\boldsymbol{y}_i, \boldsymbol{f}_i, \mathrm{u}_i \,|\, \mathrm{X}, \mathrm{Z}\right) = p\left(\boldsymbol{y}_i | \boldsymbol{f}_i\right) p\left(\boldsymbol{f}_i | \, \mathrm{u}_i, \mathrm{X}, \mathrm{Z}\right) p\left(\mathrm{u}_i \,|\, \mathrm{Z}\right)$$

Derive a variational approximation for the posterior

$$p\left(\boldsymbol{f}_i, \mathrm{u}_i \,|\, \boldsymbol{Y}, \mathrm{X}, \mathrm{Z}\right) \approx q\left(\boldsymbol{f}_i, \mathrm{u}_i\right) = p\left(\boldsymbol{f}_i | \, \mathrm{u}_i, \mathrm{X}, \mathrm{Z}\right) q\left(\mathrm{u}_i\right)$$

[4] **Notation reminder:** inducing inputs Z, inducing outputs U.

Bayesian GPLVM (revisiting the variational sparse approach)

Lets expand our intractable integral term and augment inducing variables[4]

$$p\left(\boldsymbol{y}_i, \boldsymbol{f}_i, \mathrm{u}_i \,|\, \mathrm{X}, \mathrm{Z}\right) = p\left(\boldsymbol{y}_i | \boldsymbol{f}_i\right) p\left(\boldsymbol{f}_i | \, \mathrm{u}_i, \mathrm{X}, \mathrm{Z}\right) p\left(\mathrm{u}_i \,|\, \mathrm{Z}\right)$$

Derive a variational approximation for the posterior

$$p\left(\boldsymbol{f}_i, \mathrm{u}_i \,|\, \boldsymbol{Y}, \mathrm{X}, \mathrm{Z}\right) \approx q\left(\boldsymbol{f}_i, \mathrm{u}_i\right) = p\left(\boldsymbol{f}_i | \, \mathrm{u}_i, \mathrm{X}, \mathrm{Z}\right) q\left(\mathrm{u}_i\right)$$

And we can use this to derive a new lower bound

$$\int q\left(\mathrm{X}\right) \log p\left(\boldsymbol{y}_i | \, \mathrm{X}\right) d\,\mathrm{X} \geq \int q\left(\mathrm{X}\right) q\left(\boldsymbol{f}_i, \mathrm{u}_i\right) \log \frac{p\left(\boldsymbol{y}_i, \boldsymbol{f}_i, \mathrm{u}_i \,|\, \mathrm{X}, \mathrm{Z}\right)}{q\left(\boldsymbol{f}_i, \mathrm{u}_i\right)} d\boldsymbol{f}_i d\,\mathrm{u}_i\, d\,\mathrm{X}$$

[4] **Notation reminder:** inducing inputs Z, inducing outputs U.

Lets expand likelihood term of our intractable integral and augment inducing variables[4]

$$p(\boldsymbol{y}_i, \boldsymbol{f}_i, \mathrm{u}_i \,|\, \mathrm{X}, \mathrm{Z}) = p(\boldsymbol{y}_i | \boldsymbol{f}_i)\, \cancel{p(\boldsymbol{f}_i | \mathrm{u}_i, \mathrm{X}, \mathrm{Z})} p(\mathrm{u}_i \,|\, \mathrm{Z})$$

Derive a variational approximation for the posterior

$$p(\boldsymbol{f}_i, \mathrm{u}_i \,|\, \boldsymbol{Y}, \mathrm{X}, \mathrm{Z}) \approx q(\boldsymbol{f}_i, \mathrm{u}_i) = \cancel{p(\boldsymbol{f}_i | \mathrm{u}_i, \mathrm{X}, \mathrm{Z})} q(\mathrm{u}_i)$$

And we can use this to derive a new lower bound

$$\int q(\mathrm{X}) \log p(\boldsymbol{y}_i | \mathrm{X})\, d\mathrm{X} \geq \int q(\mathrm{X})\, q(\boldsymbol{f}_i, \mathrm{u}_i) \log \frac{p(\boldsymbol{y}_i, \boldsymbol{f}_i, \mathrm{u}_i \,|\, \mathrm{X}, \mathrm{Z})}{q(\boldsymbol{f}_i, \mathrm{u}_i)}\, d\boldsymbol{f}_i\, d\mathrm{u}_i\, d\mathrm{X}$$

[4] **Notation reminder:** inducing inputs Z, inducing outputs U.

Bayesian GPLVM

And we can use this to derive a new lower bound

$$\int q(\mathrm{X}) \log p(\boldsymbol{y}_i | \mathrm{X})\, d\mathrm{X} \geq \int q(\mathrm{X})\, q(\boldsymbol{f}_i, \mathrm{u}_i) \log \frac{p(\boldsymbol{y}_i | \boldsymbol{f}_i)\, p(\mathrm{u}_i \mid \mathrm{Z})}{q(\mathrm{u}_i)} d\boldsymbol{f}_i d\,\mathrm{u}_i\, d\mathrm{X}$$

Bayesian GPLVM

And we can use this to derive a new lower bound

$$\int q(\mathrm{X}) \log p(\boldsymbol{y}_i | \mathrm{X})\, d\mathrm{X} \geq \int q(\mathrm{X})\, q(\boldsymbol{f}_i, \mathrm{u}_i) \log \frac{p(\boldsymbol{y}_i|\boldsymbol{f}_i)\, p(\mathrm{u}_i \mid \mathrm{Z})}{q(\mathrm{u}_i)} d\boldsymbol{f}_i d\,\mathrm{u}_i\, d\mathrm{X}$$

We can now split this into an integral and a tractable KL term

$$\int q(\mathrm{X}) \log p(\boldsymbol{y}_i | \mathrm{X})\, d\mathrm{X} \geq \int q(\mathrm{X})\, q(\boldsymbol{f}_i, \mathrm{u}_i) \log p(\boldsymbol{y}_i|\boldsymbol{f}_i)\, d\boldsymbol{f}_i d\,\mathrm{u}_i\, d\mathrm{X} - \mathrm{KL}\left(q(\mathrm{u}_i) \,||\, p(\mathrm{u}_i \mid \mathrm{Z})\right)$$

Bayesian GPLVM

And we can use this to derive a new lower bound

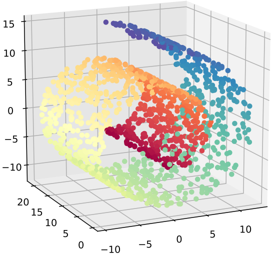
$$\int q(\mathrm{X}) \log p(\boldsymbol{y}_i | \mathrm{X})\, d\mathrm{X} \geq \int q(\mathrm{X})\, q(\boldsymbol{f}_i, \mathrm{u}_i) \log \frac{p(\boldsymbol{y}_i | \boldsymbol{f}_i)\, p(\mathrm{u}_i \mid \mathrm{Z})}{q(\mathrm{u}_i)} d\boldsymbol{f}_i d\,\mathrm{u}_i\, d\mathrm{X}$$

We can now split this into an integral and a tractable KL term

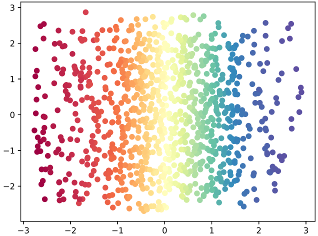
$$\begin{aligned}\int q(\mathrm{X}) \log p(\boldsymbol{y}_i | \mathrm{X})\, d\mathrm{X} \geq & \int q(\mathrm{X})\, q(\boldsymbol{f}_i, \mathrm{u}_i) \log p(\boldsymbol{y}_i | \boldsymbol{f}_i)\, d\boldsymbol{f}_i d\,\mathrm{u}_i\, d\mathrm{X} \\ & - \mathrm{KL}\left(q(\mathrm{u}_i)\, || p(\mathrm{u}_i \mid \mathrm{Z})\right)\end{aligned}$$

- Swap order of integration
- This integral is now tractable (for some kernels) as X no longer needs to be pushed through the kernel
- For more details see [Damianou(2015)] or [Titsias and Lawrence(2010)]

Bayesian GPLVM - example



(a) Swiss roll data



(b) BGPLVM embedded mean

The many flavours of GPLVM

- Shared GPLVM - map from a shared latent space to separate observation spaces
- Back constrained GPLVM - preserving locality in the image map
- Dynamic GPLVM (or GP dynamical model) - add a dynamic prior for supervised learning
- 'Deep' GPs - add a GP prior onto Z.

... and many more!

The applications of GPLVM

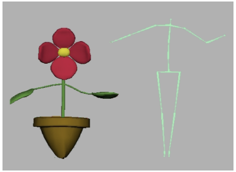


Figure: Shared GPLVM: Disney research (link)

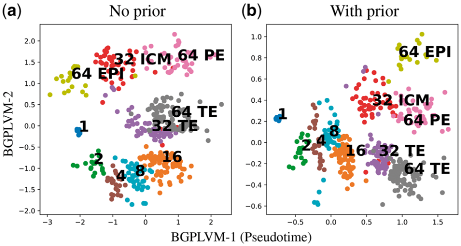


Figure: BGPLVM for single cell data

Further reading

Models

- [Lawrence(2005)] - PCA $\rightarrow$ PPCA $\rightarrow$ DPPCA $\rightarrow$ GPLVM derivation details
- [Titsias and Lawrence(2010)] - Bayesian GPLVM paper
- [Damianou(2015)] - Bayesian GPLVM thesis
- [Lawrence and Quiñonero-Candela(2006)] - Back constrained GPLVM
- [Ek and Lawrence(2009)] - Shared GPLVM

Applications

- [Yamane et al.(2010)Yamane, Ariki, and Hodgins] - Disney research Shared GPLVM for animation
- [Ahmed et al.(2019)Ahmed, Rattray, and Boukouvalas] - BGPLVM for single cell data

Multi-output Gaussian processes

Pre-requisites: The Kronecker product

Take two matrices $A \in \mathbb{R}^{m \times n}$ and $B \in \mathbb{R}^{p \times q}$

$$A = \begin{bmatrix} a_{1,1} & \dots & a_{1,n} \\ \vdots & & \vdots \\ a_{m,1} & \dots & a_{m,n} \end{bmatrix} \qquad A \otimes B = \begin{bmatrix} a_{1,1}B & \dots & a_{1,n}B \\ \vdots & & \vdots \\ a_{m,1}B & \dots & a_{m,n}B \end{bmatrix}$$

Pre-requisites: The Kronecker product

Take two matrices $A \in \mathbb{R}^{m \times n}$ and $B \in \mathbb{R}^{p \times q}$

$$A = \begin{bmatrix} a_{1,1} & \dots & a_{1,n} \\ \vdots & & \vdots \\ a_{m,1} & \dots & a_{m,n} \end{bmatrix} \qquad A \otimes B = \begin{bmatrix} a_{1,1}B & \dots & a_{1,n}B \\ \vdots & & \vdots \\ a_{m,1}B & \dots & a_{m,n}B \end{bmatrix}$$

$\boxed{\text{So what is the dimension of } A \otimes B\text{?}}$

Pre-requisites: The Kronecker product

Take two matrices $A \in \mathbb{R}^{m \times n}$ and $B \in \mathbb{R}^{p \times q}$

$$A = \begin{bmatrix} a_{1,1} & \dots & a_{1,n} \\ \vdots & & \vdots \\ a_{m,1} & \dots & a_{m,n} \end{bmatrix} \qquad A \otimes B = \begin{bmatrix} a_{1,1}B & \dots & a_{1,n}B \\ \vdots & & \vdots \\ a_{m,1}B & \dots & a_{m,n}B \end{bmatrix}$$

$\boxed{\text{So what is the dimension of } A \otimes B\text{?}}$ $A \otimes B \in \mathbb{R}^{mp \times nq}$

Pre-requisites: The Kronecker product

Take two matrices $\mathrm{A} \in \mathbb{R}^{m \times n}$ and $\mathrm{B} \in \mathbb{R}^{p \times q}$

$$\mathrm{A} = \begin{bmatrix} a_{1,1} & \dots & a_{1,n} \\ \vdots & & \vdots \\ a_{m,1} & \dots & a_{m,n} \end{bmatrix} \qquad \mathrm{A} \otimes \mathrm{B} = \begin{bmatrix} a_{1,1}\mathrm{B} & \dots & a_{1,n}\mathrm{B} \\ \vdots & & \vdots \\ a_{m,1}\mathrm{B} & \dots & a_{m,n}\mathrm{B} \end{bmatrix}$$

$\boxed{\text{So what is the dimension of } \mathrm{A} \otimes \mathrm{B}?}$ $\mathrm{A} \otimes \mathrm{B} \in \mathbb{R}^{mp \times nq}$

The inversion rule:

$$(\mathrm{A} \otimes \mathrm{B})^{-1} = \mathrm{A}^{-1} \otimes \mathrm{B}^{-1}$$

And is invertible *if and only if* both A and B are invertible.

Motivation: Multiple processes

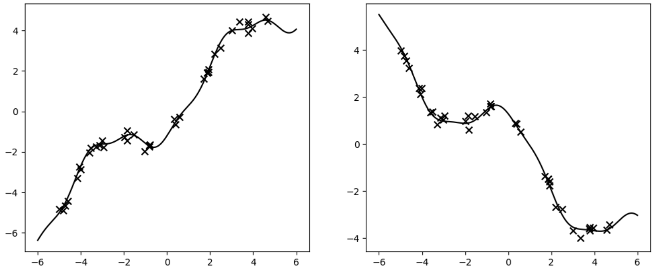


Figure: Two linearly correlated processes

Motivation: Multiple processes

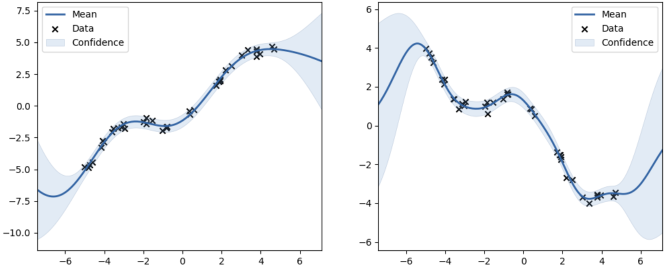


Figure: Two independent Gaussian process fits

$$f_1(\boldsymbol{x}) \sim \mathcal{GP}\left(0, k_1\left(\boldsymbol{x}, \boldsymbol{x}'\right)\right) \qquad f_2(\boldsymbol{x}) \sim \mathcal{GP}\left(0, k_2\left(\boldsymbol{x}, \boldsymbol{x}'\right)\right)$$

$$\mathrm{f}_1 \sim \mathcal{N}(0, \mathrm{K}_1) \qquad \mathrm{f}_2 \sim \mathcal{N}(0, \mathrm{K}_2)$$

Motivation: Multiple processes

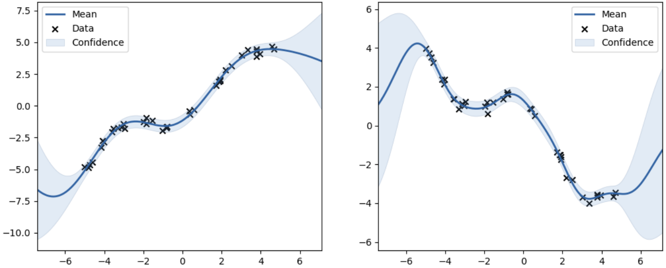


Figure: Two independent Gaussian process fits

$$\begin{bmatrix} f_1(\boldsymbol{x}) \\ f_2(\boldsymbol{x}) \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} K_1 & 0 \\ 0 & K_2 \end{bmatrix} \right)$$

Motivation: Multiple processes

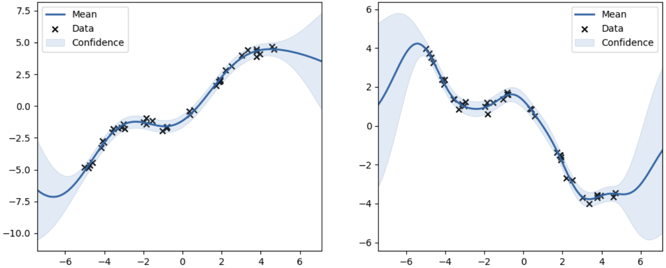


Figure: Two independent Gaussian process fits

$$\begin{bmatrix} f_1(\boldsymbol{x}) \\ f_2(\boldsymbol{x}) \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} K_1 & ? \\ ? & K_2 \end{bmatrix}\right)$$

Intrinsic Model of Coregionalisation (IMC)

General case

Sample S functions i.i.d. from the shared underlying process $\mathrm{u}^{(s)} \sim \mathcal{GP}\left(0, k\left(\boldsymbol{x}, \boldsymbol{x}'\right)\right)$.

Intrinsic Model of Coregionalisation (IMC)

General case

Sample S functions i.i.d. from the shared underlying process $\mathrm{u}^{(s)} \sim \mathcal{GP}\left(0, k\left(\boldsymbol{x}, \boldsymbol{x}'\right)\right)$.

The vector valued process is then defined as a weighted sum,

$$\boldsymbol{f}\left(\boldsymbol{x}\right) = \sum_{s=1}^{S} \boldsymbol{a}^{(s)} \mathrm{u}^{(s)}\left(\boldsymbol{x}\right).$$

where

$$\boldsymbol{f}\left(\boldsymbol{x}\right) = \left[f_1\left(\boldsymbol{x}\right), f_2\left(\boldsymbol{x}\right), \ldots, f_D\left(\boldsymbol{x}\right)\right]^T, \quad \boldsymbol{a}^{(s)} = \left[\mathrm{a}_1^{(s)}, \mathrm{a}_2^{(s)}, \ldots, \mathrm{a}_D^{(s)}\right]^T,$$

Sample S functions i.i.d. from the shared underlying process $\mathrm{u}^{(s)} \sim \mathcal{GP}\left(0, k\left(\boldsymbol{x}, \boldsymbol{x}'\right)\right)$.

The vector valued process is then defined as a weighted sum,

$$\boldsymbol{f}\left(\boldsymbol{x}\right) = \sum_{s=1}^{S} \boldsymbol{a}^{(s)} \mathrm{u}^{(s)}\left(\boldsymbol{x}\right).$$

where

$$\boldsymbol{f}\left(\boldsymbol{x}\right) = \left[f_1\left(\boldsymbol{x}\right), f_2\left(\boldsymbol{x}\right), \ldots, f_D\left(\boldsymbol{x}\right)\right]^T, \quad \boldsymbol{a}^{(s)} = \left[\mathrm{a}_1^{(s)}, \mathrm{a}_2^{(s)}, \ldots, \mathrm{a}_D^{(s)}\right]^T,$$

$$\begin{bmatrix} f_1\left(\boldsymbol{x}\right) \\ f_2\left(\boldsymbol{x}\right) \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} K_1 & ? \\ ? & K_2 \end{bmatrix}\right)$$

Intrinsic Model of Coregionalisation (IMC)

General case

$$\boxed{\text{cov}\left(\boldsymbol{f}\left(\boldsymbol{x}\right), \boldsymbol{f}\left(\boldsymbol{x}'\right)\right) = \mathbb{E}\left[\boldsymbol{f}\left(\boldsymbol{x}\right)\boldsymbol{f}\left(\boldsymbol{x}'\right)^T\right] - \mathbb{E}\left[\boldsymbol{f}\left(\boldsymbol{x}\right)\right]\mathbb{E}\left[\boldsymbol{f}\left(\boldsymbol{x}'\right)\right]^T}$$

$$\begin{aligned}
\boldsymbol{f}\left(\boldsymbol{x}\right) &= \boldsymbol{a}^{(1)}\mathrm{u}^{(1)}\left(\boldsymbol{x}\right) + \boldsymbol{a}^{(2)}\mathrm{u}^{(2)}\left(\boldsymbol{x}\right) + \dots \\
\text{cov}\left(\boldsymbol{f}\left(\boldsymbol{x}\right), \boldsymbol{f}\left(\boldsymbol{x}'\right)\right) &= \boldsymbol{a}^{(1)}\boldsymbol{a}^{(1)^T}\text{cov}\left(u^{(1)}\left(\boldsymbol{x}\right), u^{(1)}\left(\boldsymbol{x}'\right)\right) + \boldsymbol{a}^{(2)}\boldsymbol{a}^{(2)^T}\text{cov}\left(u^{(2)}\left(\boldsymbol{x}\right), u^{(2)}\left(\boldsymbol{x}'\right)\right) + \dots \\
&= \boldsymbol{a}^{(1)}\boldsymbol{a}^{(1)^T} k\left(\boldsymbol{x}, \boldsymbol{x}'\right) + \boldsymbol{a}^{(2)}\boldsymbol{a}^{(2)^T} k\left(\boldsymbol{x}, \boldsymbol{x}'\right) + \dots \\
&= \left[\boldsymbol{a}^{(1)}\boldsymbol{a}^{(1)^T} + \boldsymbol{a}^{(2)}\boldsymbol{a}^{(2)^T} + \dots\right] k\left(\boldsymbol{x}, \boldsymbol{x}'\right)
\end{aligned}$$

Intrinsic Model of Coregionalisation (IMC)

General case

$$\boxed{\operatorname{cov}\left(\boldsymbol{f}\left(\boldsymbol{x}\right), \boldsymbol{f}\left(\boldsymbol{x}'\right)\right) = \mathbb{E}\left[\boldsymbol{f}\left(\boldsymbol{x}\right)\boldsymbol{f}\left(\boldsymbol{x}'\right)^T\right] - \mathbb{E}\left[\boldsymbol{f}\left(\boldsymbol{x}\right)\right]\mathbb{E}\left[\boldsymbol{f}\left(\boldsymbol{x}'\right)\right]^T}$$

$$\begin{aligned}
\boldsymbol{f}\left(\boldsymbol{x}\right) &= \boldsymbol{a}^{(1)}\mathrm{u}^{(1)}\left(\boldsymbol{x}\right) + \boldsymbol{a}^{(2)}\mathrm{u}^{(2)}\left(\boldsymbol{x}\right) + \dots \\
\operatorname{cov}\left(\boldsymbol{f}\left(\boldsymbol{x}\right), \boldsymbol{f}\left(\boldsymbol{x}'\right)\right) &= \boldsymbol{a}^{(1)}\boldsymbol{a}^{(1)^T}\operatorname{cov}\left(u^{(1)}\left(\boldsymbol{x}\right), u^{(1)}\left(\boldsymbol{x}'\right)\right) + \boldsymbol{a}^{(2)}\boldsymbol{a}^{(2)^T}\operatorname{cov}\left(u^{(2)}\left(\boldsymbol{x}\right), u^{(2)}\left(\boldsymbol{x}'\right)\right) + \dots \\
&= \boldsymbol{a}^{(1)}\boldsymbol{a}^{(1)^T}k\left(\boldsymbol{x}, \boldsymbol{x}'\right) + \boldsymbol{a}^{(2)}\boldsymbol{a}^{(2)^T}k\left(\boldsymbol{x}, \boldsymbol{x}'\right) + \dots \\
&= \underbrace{\left[\boldsymbol{a}^{(1)}\boldsymbol{a}^{(1)^T} + \boldsymbol{a}^{(2)}\boldsymbol{a}^{(2)^T} + \dots\right]}_{\hat{\boldsymbol{B}} \in \mathbb{R}^{D \times D}} k\left(\boldsymbol{x}, \boldsymbol{x}'\right)
\end{aligned}$$

Intrinsic Model of Coregionalisation (IMC)

General case

$$\boxed{\text{cov}\left(\boldsymbol{f}\left(\boldsymbol{x}\right), \boldsymbol{f}\left(\boldsymbol{x}'\right)\right) = \mathbb{E}\left[\boldsymbol{f}\left(\boldsymbol{x}\right)\boldsymbol{f}\left(\boldsymbol{x}'\right)^{T}\right] - \mathbb{E}\left[\boldsymbol{f}\left(\boldsymbol{x}\right)\right]\mathbb{E}\left[\boldsymbol{f}\left(\boldsymbol{x}'\right)\right]^{T}}$$

$$\begin{aligned}
\boldsymbol{f}\left(\boldsymbol{x}\right) &= \boldsymbol{a}^{(1)}\mathrm{u}^{(1)}\left(\boldsymbol{x}\right) + \boldsymbol{a}^{(2)}\mathrm{u}^{(2)}\left(\boldsymbol{x}\right) + \ldots \\
\text{cov}\left(\boldsymbol{f}\left(\boldsymbol{x}\right), \boldsymbol{f}\left(\boldsymbol{x}'\right)\right) &= \boldsymbol{a}^{(1)}\boldsymbol{a}^{(1)^{T}}\text{cov}\left(u^{(1)}\left(\boldsymbol{x}\right), u^{(1)}\left(\boldsymbol{x}'\right)\right) + \boldsymbol{a}^{(2)}\boldsymbol{a}^{(2)^{T}}\text{cov}\left(u^{(2)}\left(\boldsymbol{x}\right), u^{(2)}\left(\boldsymbol{x}'\right)\right) + \ldots \\
&= \boldsymbol{a}^{(1)}\boldsymbol{a}^{(1)^{T}} k\left(\boldsymbol{x}, \boldsymbol{x}'\right) + \boldsymbol{a}^{(2)}\boldsymbol{a}^{(2)^{T}} k\left(\boldsymbol{x}, \boldsymbol{x}'\right) + \ldots \\
&= \underbrace{\left[\boldsymbol{a}^{(1)}\boldsymbol{a}^{(1)^{T}} + \boldsymbol{a}^{(2)}\boldsymbol{a}^{(2)^{T}} + \ldots\right]}_{\hat{\boldsymbol{B}} \in \mathbb{R}^{D \times D}} k\left(\boldsymbol{x}, \boldsymbol{x}'\right)
\end{aligned}$$

$$\text{cov}\left(\boldsymbol{f}\left(\boldsymbol{x}\right), \boldsymbol{f}\left(\boldsymbol{x}'\right)\right) = \hat{\boldsymbol{B}} k\left(\boldsymbol{x}, \boldsymbol{x}'\right)$$

Intrinsic Model of Coregionalisation (IMC)

General case

$$\hat{\boldsymbol{B}} = \begin{bmatrix} b_{1,1} & b_{1,2} \\ b_{2,1} & b_{2,2} \end{bmatrix}, \qquad \text{if } D = 2$$

$$\begin{aligned} \begin{bmatrix} f_1(x) \\ f_2(x) \end{bmatrix} &\sim \mathcal{N}\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} b_{1,1}\mathrm{K} & b_{1,2}\mathrm{K} \\ b_{2,1}\mathrm{K} & b_{2,2}\mathrm{K} \end{bmatrix} \right) \\ &\sim \mathcal{N}\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \hat{\boldsymbol{B}} \otimes \mathrm{K} \right) \end{aligned}$$

Sample Q functions from separate processes $\mathrm{u}_q \sim \mathcal{GP}\left(0, k_q\left(\boldsymbol{x}, \boldsymbol{x}'\right)\right)$.

Semiparametric latent factor model (SLFM)

General case

Sample Q functions from separate processes $\mathrm{u}_q \sim \mathcal{GP}\left(0, k_q\left(\boldsymbol{x}, \boldsymbol{x}'\right)\right)$.

The vector valued process is then defined as a weighted sum,

$$\boldsymbol{f}\left(\boldsymbol{x}\right) = \sum_{q=1}^{Q} \boldsymbol{a}_q \mathrm{u}_q\left(\boldsymbol{x}\right).$$

where
$\boldsymbol{f}\left(\boldsymbol{x}\right) = \left[f_1\left(\boldsymbol{x}\right), f_2\left(\boldsymbol{x}\right), \ldots, f_D\left(\boldsymbol{x}\right)\right]^T, \quad \boldsymbol{a}_q = \left[\mathrm{a}_{q,1}, \mathrm{a}_{q,2}, \ldots, \mathrm{a}_{q,D}\right]^T,$

Semiparametric latent factor model (SLFM)

General case

Sample Q functions from separate processes $\mathrm{u}_q \sim \mathcal{GP}\left(0, k_q\left(\boldsymbol{x}, \boldsymbol{x}'\right)\right)$.

The vector valued process is then defined as a weighted sum,

$$\boldsymbol{f}\left(\boldsymbol{x}\right) = \sum_{q=1}^{Q} \boldsymbol{a}_q \mathrm{u}_q\left(\boldsymbol{x}\right).$$

where

$\boldsymbol{f}\left(\boldsymbol{x}\right) = \left[f_1\left(\boldsymbol{x}\right), f_2\left(\boldsymbol{x}\right), \ldots, f_D\left(\boldsymbol{x}\right)\right]^T, \quad \boldsymbol{a}_q = \left[\mathrm{a}_{q,1}, \mathrm{a}_{q,2}, \ldots, \mathrm{a}_{q,D}\right]^T,$

$$\begin{bmatrix} f_1\left(\boldsymbol{x}\right) \\ f_2\left(\boldsymbol{x}\right) \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} K_1 & ? \\ ? & K_2 \end{bmatrix}\right)$$

Semiparametric latent factor model (SLFM)

General case

$$\boxed{\operatorname{cov}\left(\boldsymbol{f}\left(\boldsymbol{x}\right), \boldsymbol{f}\left(\boldsymbol{x}'\right)\right) = \mathbb{E}\left[\boldsymbol{f}\left(\boldsymbol{x}\right)\boldsymbol{f}\left(\boldsymbol{x}'\right)^T\right] - \mathbb{E}\left[\boldsymbol{f}\left(\boldsymbol{x}\right)\right]\mathbb{E}\left[\boldsymbol{f}\left(\boldsymbol{x}'\right)\right]^T}$$

$$\begin{aligned}
\boldsymbol{f}\left(\boldsymbol{x}\right) &= \boldsymbol{a}_1 \mathrm{u}_1\left(\boldsymbol{x}\right) + \boldsymbol{a}_2 \mathrm{u}_2\left(\boldsymbol{x}\right) + \dots \\
\operatorname{cov}\left(\boldsymbol{f}\left(\boldsymbol{x}\right), \boldsymbol{f}\left(\boldsymbol{x}'\right)\right) &= \boldsymbol{a}_1\boldsymbol{a}_1^T \operatorname{cov}\left(u_1\left(\boldsymbol{x}\right), u_1\left(\boldsymbol{x}'\right)\right) + \boldsymbol{a}_2\boldsymbol{a}_2^T \operatorname{cov}\left(u_2\left(\boldsymbol{x}\right), u_2\left(\boldsymbol{x}'\right)\right) + \dots \\
&= \underbrace{\boldsymbol{a}_1\boldsymbol{a}_1^T}_{\tilde{\boldsymbol{B}}_1 \in \mathbb{R}^{D\times D}} k_1\left(\boldsymbol{x}, \boldsymbol{x}'\right) + \underbrace{\boldsymbol{a}_2\boldsymbol{a}_2^T}_{\tilde{\boldsymbol{B}}_2 \in \mathbb{R}^{D\times D}} k_2\left(\boldsymbol{x}, \boldsymbol{x}'\right) + \dots
\end{aligned}$$

Semiparametric latent factor model (SLFM)

General case

$$\boxed{\operatorname{cov}\left(\boldsymbol{f}\left(\boldsymbol{x}\right), \boldsymbol{f}\left(\boldsymbol{x}'\right)\right) = \mathbb{E}\left[\boldsymbol{f}\left(\boldsymbol{x}\right)\boldsymbol{f}\left(\boldsymbol{x}'\right)^{T}\right] - \mathbb{E}\left[\boldsymbol{f}\left(\boldsymbol{x}\right)\right]\mathbb{E}\left[\boldsymbol{f}\left(\boldsymbol{x}'\right)\right]^{T}}$$

$$\begin{aligned}
\boldsymbol{f}\left(\boldsymbol{x}\right) &= \boldsymbol{a}_1 \mathrm{u}_1\left(\boldsymbol{x}\right) + \boldsymbol{a}_2 \mathrm{u}_2\left(\boldsymbol{x}\right) + \dots \\
\operatorname{cov}\left(\boldsymbol{f}\left(\boldsymbol{x}\right), \boldsymbol{f}\left(\boldsymbol{x}'\right)\right) &= \boldsymbol{a}_1\boldsymbol{a}_1^T \operatorname{cov}\left(u_1\left(\boldsymbol{x}\right), u_1\left(\boldsymbol{x}'\right)\right) + \boldsymbol{a}_2\boldsymbol{a}_2^T \operatorname{cov}\left(u_2\left(\boldsymbol{x}\right), u_2\left(\boldsymbol{x}'\right)\right) + \dots \\
&= \underbrace{\boldsymbol{a}_1\boldsymbol{a}_1^T}_{\tilde{\boldsymbol{B}}_1 \in \mathbb{R}^{D\times D}} k_1\left(\boldsymbol{x}, \boldsymbol{x}'\right) + \underbrace{\boldsymbol{a}_2\boldsymbol{a}_2^T}_{\tilde{\boldsymbol{B}}_2 \in \mathbb{R}^{D\times D}} k_2\left(\boldsymbol{x}, \boldsymbol{x}'\right) + \dots
\end{aligned}$$

$$\operatorname{cov}\left(\boldsymbol{f}\left(\boldsymbol{x}\right), \boldsymbol{f}\left(\boldsymbol{x}'\right)\right) = \tilde{\boldsymbol{B}}_1 k_1\left(\boldsymbol{x}, \boldsymbol{x}'\right) + \tilde{\boldsymbol{B}}_2 k_2\left(\boldsymbol{x}, \boldsymbol{x}'\right) + \dots$$

$$\begin{bmatrix} f_1(x) \\ f_2(x) \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \sum_{q=1}^{Q} \tilde{\boldsymbol{B}}_q \otimes \mathrm{K}_q\right), \qquad \text{with } D = 2$$

Linear Model of Coregionalisation (LMC)

General case

Sample S_q functions from Q separate processes $\mathrm{u}_q^{(s)} \sim \mathcal{GP}\left(0, k_q\left(\boldsymbol{x}, \boldsymbol{x}'\right)\right)$.

Sample S_q functions from Q separate processes $\mathrm{u}_q^{(s)} \sim \mathcal{GP}\left(0, k_q\left(\boldsymbol{x}, \boldsymbol{x}'\right)\right)$.

The vector valued process is then defined as a weighted sum,

$$\mathrm{f}\left(\boldsymbol{x}\right) = \sum_{q=1}^{Q} \sum_{s=1}^{S} \boldsymbol{a}_q^{(s)} \mathrm{u}_q^{(s)}\left(\boldsymbol{x}\right).$$

where

$$\boldsymbol{f}\left(\boldsymbol{x}\right) = \left[f_1\left(\boldsymbol{x}\right), f_2\left(\boldsymbol{x}\right), \ldots, f_D\left(\boldsymbol{x}\right)\right]^T, \quad \boldsymbol{a}_q^{(s)} = \left[a_{q,1}^{(s)}, a_{q,2}^{(s)}, \ldots, a_{q,D}^{(s)}\right]^T$$

Linear Model of Coregionalisation (LMC)

General case

$$\boxed{\operatorname{cov}\left(\boldsymbol{f}\left(\boldsymbol{x}\right), \boldsymbol{f}\left(\boldsymbol{x}'\right)\right) = \mathbb{E}\left[\boldsymbol{f}\left(\boldsymbol{x}\right)\boldsymbol{f}\left(\boldsymbol{x}'\right)^T\right] - \mathbb{E}\left[\boldsymbol{f}\left(\boldsymbol{x}\right)\right]\mathbb{E}\left[\boldsymbol{f}\left(\boldsymbol{x}'\right)\right]^T}$$

$$\begin{aligned}\boldsymbol{f}\left(\boldsymbol{x}\right) = & \overbrace{\left[\boldsymbol{a}_1^{(1)}\mathrm{u}_1^{(1)}\left(\boldsymbol{x}\right) + \boldsymbol{a}_1^{(2)}\mathrm{u}_1^{(2)}\left(\boldsymbol{x}\right) + \dots\right]}^{\text{Intrinsic Model of Coregionalization}} \\ & + \left[\boldsymbol{a}_2^{(1)}\mathrm{u}_2^{(1)}\left(\boldsymbol{x}\right) + \boldsymbol{a}_2^{(2)}\mathrm{u}_2^{(2)}\left(\boldsymbol{x}\right) + \dots\right] + \dots\end{aligned}$$

$$\begin{aligned}\operatorname{cov}\left(\boldsymbol{f}\left(\boldsymbol{x}\right), \boldsymbol{f}\left(\boldsymbol{x}'\right)\right) = & \overbrace{\left[\boldsymbol{a}_1^{(1)}\boldsymbol{a}_1^{(1)T} + \boldsymbol{a}_1^{(2)}\boldsymbol{a}_1^{(2)T}\right]}^{\boldsymbol{B}_1 \in \mathbb{R}^{D\times D}} k_1\left(\boldsymbol{x}, \boldsymbol{x}'\right) \\ & + \underbrace{\left[\boldsymbol{a}_2^{(1)}\boldsymbol{a}_2^{(1)T} + \boldsymbol{a}_2^{(2)}\boldsymbol{a}_2^{(2)T}\right]}_{\boldsymbol{B}_2 \in \mathbb{R}^{D\times D}} k_2\left(\boldsymbol{x}, \boldsymbol{x}'\right) + \dots\end{aligned}$$

Linear Model of Coregionalisation (LMC)

General case

$$\boxed{\operatorname{cov}\left(\boldsymbol{f}\left(\boldsymbol{x}\right),\boldsymbol{f}\left(\boldsymbol{x}'\right)\right)=\mathbb{E}\left[\boldsymbol{f}\left(\boldsymbol{x}\right)\boldsymbol{f}\left(\boldsymbol{x}'\right)^{T}\right]-\mathbb{E}\left[\boldsymbol{f}\left(\boldsymbol{x}\right)\right]\mathbb{E}\left[\boldsymbol{f}\left(\boldsymbol{x}'\right)\right]^{T}}$$

$$\begin{aligned}\operatorname{cov}\left(\boldsymbol{f}\left(\boldsymbol{x}\right),\boldsymbol{f}\left(\boldsymbol{x}'\right)\right)=&\overbrace{\left[\boldsymbol{a}_1^{(1)}\boldsymbol{a}_1^{(1)T}+\boldsymbol{a}_1^{(2)}\boldsymbol{a}_1^{(2)T}\right]}^{\boldsymbol{B}_1\in\mathbb{R}^{D\times D}}k_1\left(\boldsymbol{x},\boldsymbol{x}'\right)\\&+\underbrace{\left[\boldsymbol{a}_2^{(1)}\boldsymbol{a}_2^{(1)T}+\boldsymbol{a}_2^{(2)}\boldsymbol{a}_2^{(2)T}\right]}_{\boldsymbol{B}_2\in\mathbb{R}^{D\times D}}k_2\left(\boldsymbol{x},\boldsymbol{x}'\right)+\dots\end{aligned}$$

$$\operatorname{cov}\left(\boldsymbol{f}\left(\boldsymbol{x}\right),\boldsymbol{f}\left(\boldsymbol{x}'\right)\right)=\boldsymbol{B}_1k_1\left(\boldsymbol{x},\boldsymbol{x}'\right)+\boldsymbol{B}_2k_2\left(\boldsymbol{x},\boldsymbol{x}'\right)+\dots$$

$$\begin{bmatrix}f_1(x)\\f_2(x)\end{bmatrix}\sim\mathcal{N}\left(\begin{bmatrix}0\\0\end{bmatrix},\sum_{q=1}^{Q}\boldsymbol{B}_q\otimes\mathrm{K}_q\right),\qquad\text{with } D=2$$

Linear Model of Coregionalisation (LMC)

Example

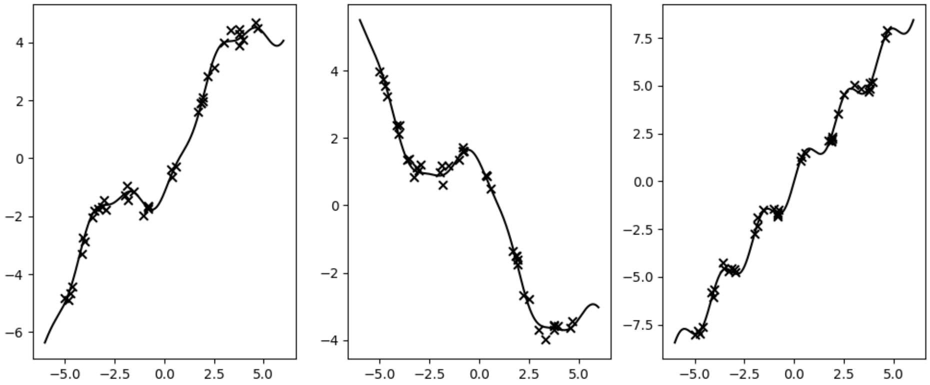


Figure: A third process

Linear Model of Coregionalisation (LMC)

Example

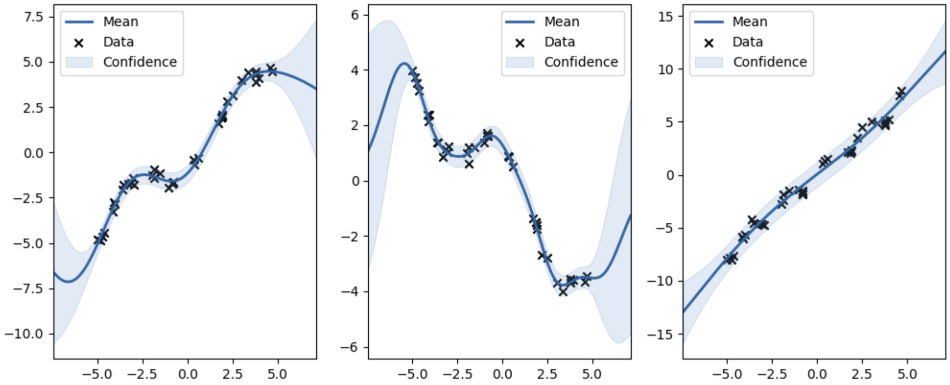


Figure: Independent Gaussian process fits

Linear Model of Coregionalisation (LMC)

Example

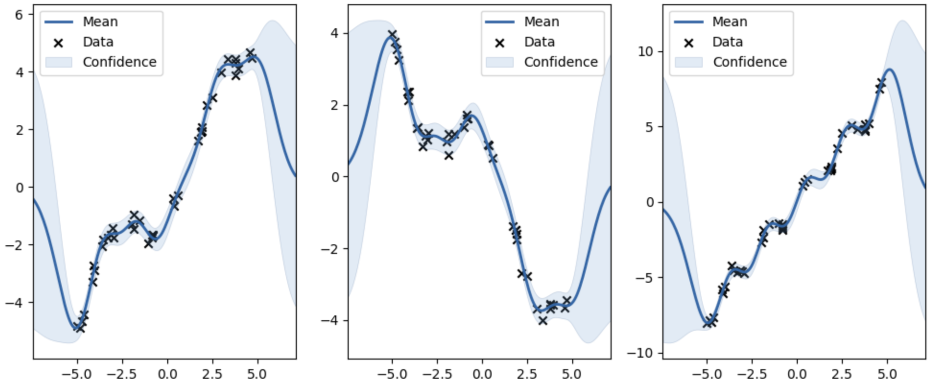


Figure: Linear Model of Coregionalisation fit

Next time

Next lecture on kernel learning

- Matérn kernel
- Multiple kernel learning
- Non-stationary RBF kernel
- Spectral kernels

Bibliography I

Sumon Ahmed, Magnus Rattray, and Alexis Boukouvalas.
Grandprix: scaling up the bayesian gplvm for single-cell data.
Bioinformatics, 35(1):47–54, 2019.

Andreas Damianou.
Deep Gaussian processes and variational propagation of uncertainty.
PhD thesis, University of Sheffield, 2015.

Carl Henrik Ek and PHTND Lawrence.
Shared Gaussian process latent variable models.
PhD thesis, Citeseer, 2009.

Neil Lawrence.
Probabilistic non-linear principal component analysis with gaussian process latent variable models.
Journal of machine learning research, 6(Nov):1783–1816, 2005.

Bibliography II

Neil D Lawrence and Joaquin Quiñonero-Candela.
Local distance preservation in the gp-lvm through back constraints.
In *Proceedings of the 23rd international conference on Machine learning*, pages 513–520, 2006.

Michalis Titsias.
Variational learning of inducing variables in sparse gaussian processes.
In *Artificial Intelligence and Statistics*, pages 567–574, 2009.

Michalis Titsias and Neil D Lawrence.
Bayesian gaussian process latent variable model.
In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, pages 844–851. JMLR Workshop and Conference Proceedings, 2010.

Katsu Yamane, Yuka Ariki, and Jessica Hodgins.
Animating non-humanoid characters with human motion data.
In *Proceedings of the 2010 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, pages 169–178, 2010.