Feature representation with Deep Gaussian processes

Andreas Damianou

Department of Neuro- and Computer Science, University of Sheffield, UK

Feature Extraction with Gaussian Processes Workshop, Sheffield, 18/09/2014

Outline

Part 1: A general view
- Deep modelling and deep GPs

Part 2: Structure in the latent space (priors for the features)
- Dynamics
- Autoencoders

Part 3: Deep Gaussian processes
- Bayesian regularization
- Inducing Points
- Structure: ARD and MRD (multi-view)
- Examples

Summary

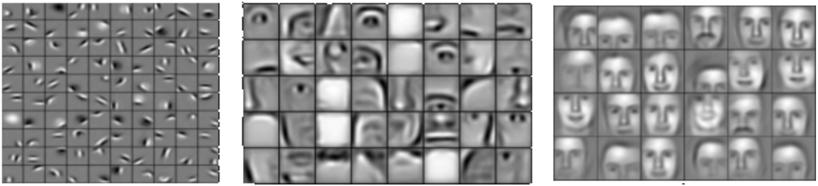
Outline

Feature representation + feature relationships

Feature representation + feature relationships

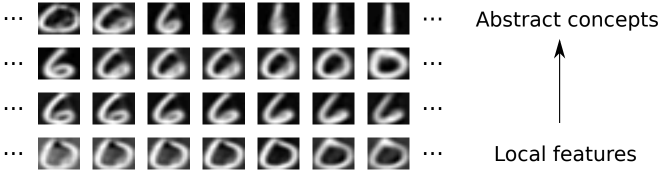
- Good data representations facilitate learning / inference.
- Deep structures facilitate learning associated with abstract information *(Bengio, 2009)*.

Hierarchy of features



[Lee et al. 09, “Convolutional DBNs for Scalable Unsupervised Learning of Hierarchical Representations”]

Hierarchy of features



[Damianou and Lawrence 2013, “Deep Gaussian Processes”]

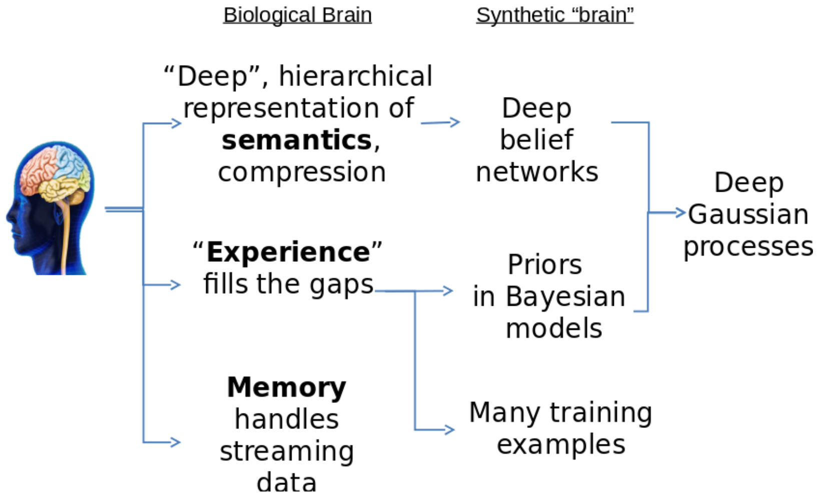
Biological Brain

“Deep”, hierarchical representation of **semantics**, compression

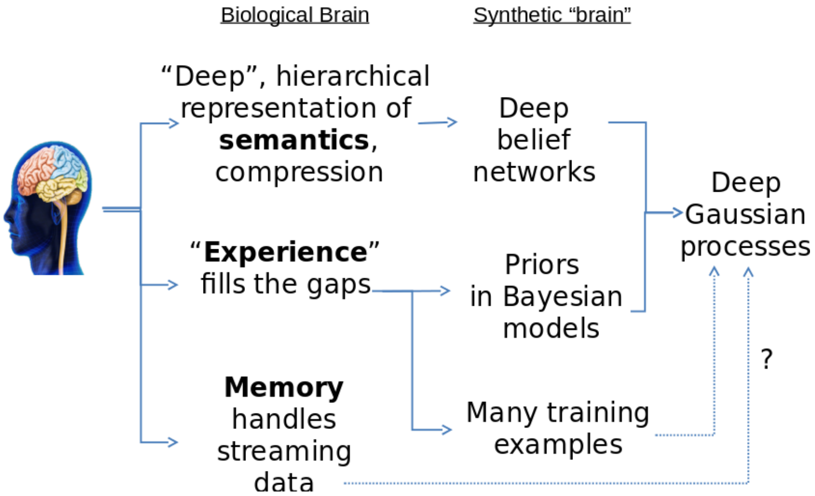
“**Experience**” fills the gaps

Memory handles streaming data

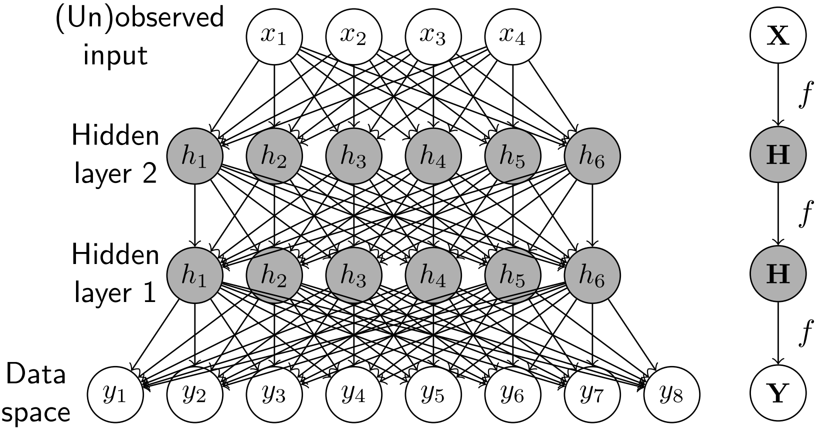
Biological Brain	Synthetic "brain"
"Deep", hierarchical representation of **semantics**, compression	Deep belief networks
"**Experience**" fills the gaps	Priors in Bayesian models
Memory handles streaming data	Many training examples



Biological Brain
Synthetic "brain"
"Deep", hierarchical representation of **semantics**, compression
"**Experience**" fills the gaps
Memory handles streaming data
Deep belief networks
Priors in Bayesian models
Many training examples
Deep Gaussian processes

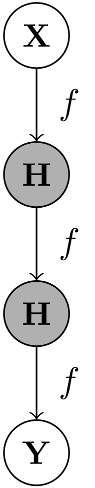

Biological Brain
Synthetic "brain"
"Deep", hierarchical representation of **semantics**, compression
"**Experience**" fills the gaps
Memory handles streaming data
Deep belief networks
Priors in Bayesian models
Many training examples
Deep Gaussian processes
?

Deep learning (directed graph)



$$\mathbf{Y} = f(f(\cdots f(\mathbf{X}))), \qquad \mathbf{H}_i = f_i(\mathbf{H}_{i-1})$$

Deep Gaussian processes - Big Picture



Deep GP:

- Directed graphical model
- Non-parametric, non-linear mappings f
- Mappings f marginalised out analytically
- Likelihood is a non-linear function of the inputs
- Continuous variables
- NOT a GP!

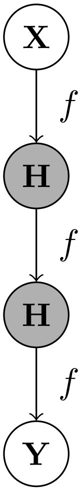
Challenges:

- Marginalise out $\mathbf{H}$
- No sampling: analytic approximation of objective

Solution:

- Variational approximation
- This also gives access to the *model evidence*

Deep Gaussian processes - Big Picture



Deep GP:

- Directed graphical model
- Non-parametric, non-linear mappings f
- Mappings f marginalised out analytically
- Likelihood is a non-linear function of the inputs
- Continuous variables
- NOT a GP!

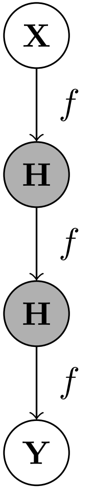
Challenges:

- Marginalise out $\mathbf{H}$
- No sampling: analytic approximation of objective

Solution:

- Variational approximation
- This also gives access to the *model evidence*

Deep Gaussian processes - Big Picture



Deep GP:

- Directed graphical model
- Non-parametric, non-linear mappings f
- Mappings f marginalised out analytically
- Likelihood is a non-linear function of the inputs
- Continuous variables
- NOT a GP!

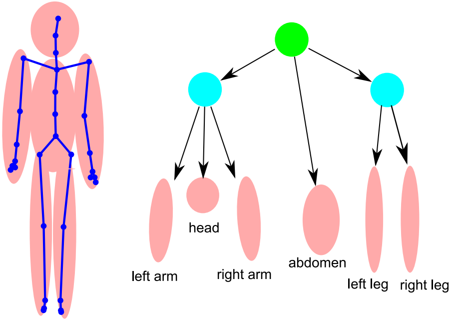
Challenges:

- Marginalise out $\mathbf{H}$
- No sampling: analytic approximation of objective

Solution:

- Variational approximation
- This also gives access to the *model evidence*

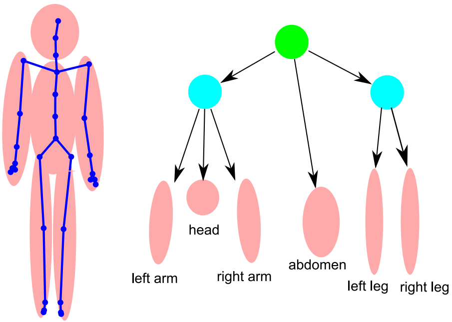
Hierarchical GP-LVM



- Hidden layers are not marginalised out.
- This leads to some difficulties.

[Lawrence and Moore, 2004]

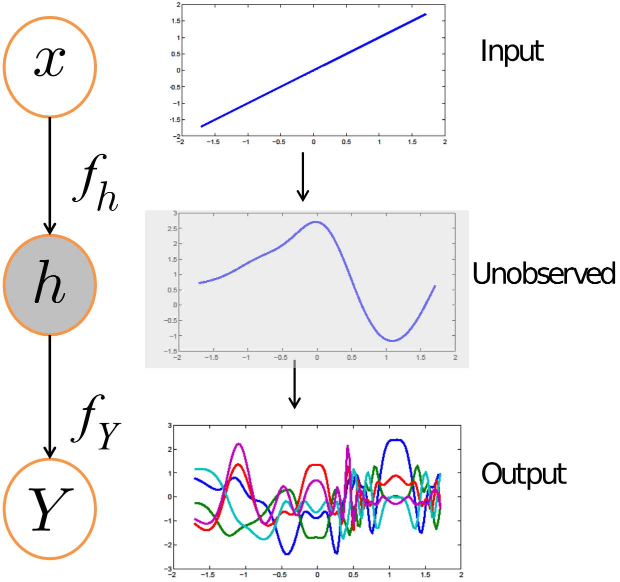
Hierarchical GP-LVM



- Hidden layers are not marginalised out.
- This leads to some difficulties.

[Lawrence and Moore, 2004]

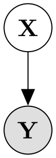
Sampling from a deep GP



Outline

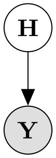
Dynamics

GP-LVM



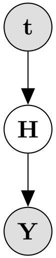
Dynamics

GP-LVM



Dynamics

GP-LVM



- If $\mathbf{Y}$ form is a multivariate time-series, then $\mathbf{H}$ also has to be one
- Place a temporal GP prior on the latent space:

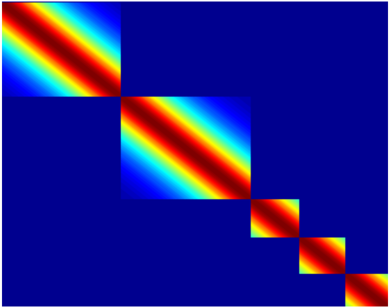
$$\mathbf{h} = h(t) = \mathcal{GP}(\mathbf{0}, k_h(t, t))$$

$$\mathbf{f} = f(h) = \mathcal{GP}(\mathbf{0}, k_f(h, h))$$

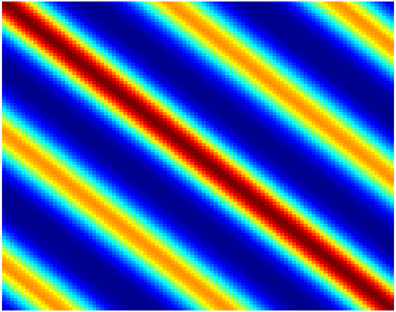
$$\mathbf{y} = f(h) + \epsilon$$

Dynamics

- Dynamics are encoded in the covariance matrix $\mathbf{K} = k(\mathbf{t}, \mathbf{t})$.
- We can consider special forms for $\mathbf{K}$.



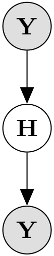
Model individual sequences



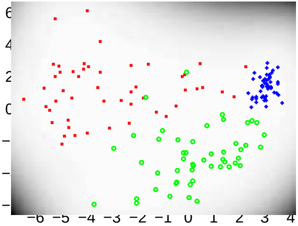
Model periodic data

- https://www.youtube.com/watch?v=i9TEoYxaBxQ (missa)
- https://www.youtube.com/watch?v=mUY1XHPnoCU (dog)
- https://www.youtube.com/watch?v=fHDWloJtgk8 (mocap)

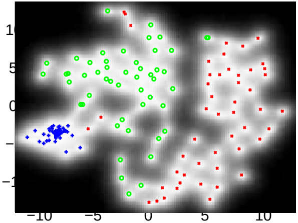
Autoencoder



GP-LVM:

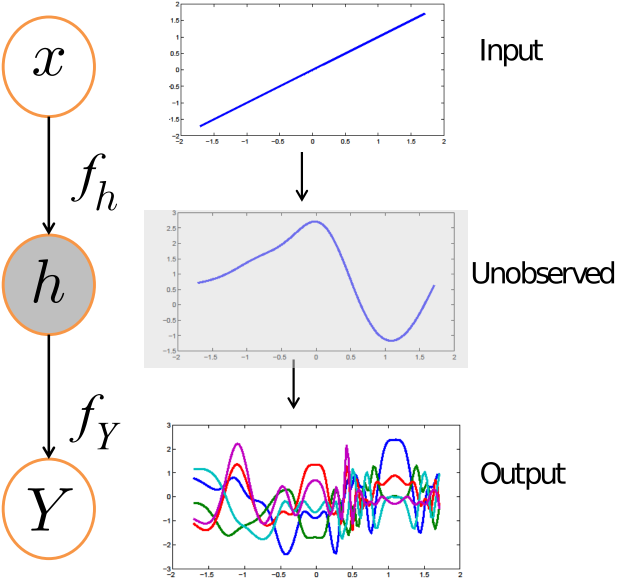


Non-parametric auto-encoder:

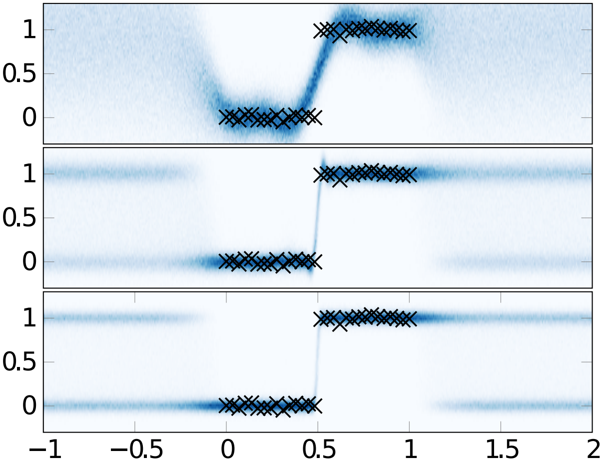


Outline

Sampling from a deep GP



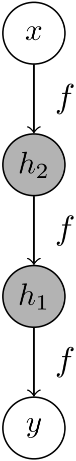
Modelling a step function - Posterior samples



- GP (top), 2 and 4 layer deep GP (middle, bottom).
- Posterior is bimodal.

Image by J. Hensman

MAP optimisation?



- Joint $= p(y|h_1)p(h_1|h_2)p(h_2|x)$
- MAP optimization is extremely problematic because:
  - Dimensionality of hs has to be decided a priori
  - Prone to overfitting, if h are treated as parameters
  - Deep structures are not supported by the model's objective but have to be forced [Lawrence & Moore '07]

Regularization solution: approximate Bayesian framework

- Analytic variational bound $\mathcal{F} \leq p(y|x)$
  - Extend the Variational Free Energy sparse GPs (Titsias 09) / Variational Compression tricks.
  - *Approximately* marginalise out h
- Automatic structure discovery (nodes, connections, layers)
  - Use the Automatic / Manifold Relevance Determination trick
- ...

Direct marginalisation of h is intractable (O_o)

- New objective: $p(y|x) = \int_{h_1} \left(p(y|h_1) \int_{h_2} p(h_1|h_2) p(h_2|x) \right)$
- $p(h_1|x) \quad = \int_{h_2, f_2} \quad p(h_1|f_2) p(f_2|h_2) \quad p(h_2|x)$
- $p(h_1|x, \tilde{h}_2) = \int_{h_2, f_2, u_2} p(h_1|f_2) p(f_2|u_2, h_2) p(u_2|\tilde{h}_2) p(h_2|x)$
- $\log p(h_1|x, \tilde{h}_2) \geq \int_{h_2, f_2, u_2} \mathcal{Q} \log \frac{p(h_1|f_2) \cancel{p(f_2|u_2, h_2)} p(u_2|\tilde{h}_2) p(h_2|x)}{\mathcal{Q} = \cancel{p(f_2|u_2, h_2)} q(u_2) q(h_2)}$
- $\log p(h_1|x, \tilde{h}_2) \geq \int_{h_2, f_2, u_2} \mathcal{Q} \log \frac{p(h_1|f_2) p(u_2|\tilde{h}_2) p(h_2|x)}{\mathcal{Q} = q(u_2) q(h_2)}$

$p(u_2|\tilde{h}_2)$ contains $k(\tilde{h}_2, \tilde{h}_2)^{-1}$

Direct marginalisation of h is intractable (O_o)

- New objective: $p(y|x) = \int_{h_1} \left(p(y|h_1) \int_{h_2} p(h_1|h_2) p(h_2|x) \right)$

Direct marginalisation of h is intractable (O_o)

- New objective: $p(y|x) = \int_{h_1} \left(p(y|h_1) \int_{h_2} p(h_1|h_2) p(h_2|x) \right)$
- $p(h_1|x) \quad = \int_{h_2, f_2} \quad p(h_1|f_2) p(f_2|h_2) \quad p(h_2|x)$

Direct marginalisation of h is intractable (O_o)

- New objective: $p(y|x) = \int_{h_1} \left(p(y|h_1) \int_{h_2} p(h_1|h_2) p(h_2|x) \right)$
- $p(h_1|x) \quad = \int_{h_2, f_2} \quad p(h_1|f_2) p(f_2|h_2) \quad p(h_2|x)$

Direct marginalisation of h is intractable (O_o)

- New objective: $p(y|x) = \int_{h_1} \left(p(y|h_1) \int_{h_2} p(h_1|h_2) p(h_2|x) \right)$
- $p(h_1|x) \quad = \int_{h_2, f_2} p(h_1|f_2) \underbrace{p(f_2|h_2)}_{\substack{\text{contains} \\ (k(h_2,h_2))^{-1}}} p(h_2|x)$

Direct marginalisation of h is intractable (O_o)

- New objective: $p(y|x) = \int_{h_1} \left(p(y|h_1) \int_{h_2} p(h_1|h_2) p(h_2|x) \right)$
- $p(h_1|x) \quad = \int_{h_2, f_2} \; p(h_1|f_2) p(f_2|h_2) \; p(h_2|x)$
- $p(h_1|x, \tilde{h}_2) = \int_{h_2, f_2, u_2} p(h_1|f_2) p(f_2|u_2, h_2) p(u_2|\tilde{h}_2) p(h_2|x)$

Direct marginalisation of h is intractable (O_o)

- New objective: $p(y|x) = \int_{h_1} \left(p(y|h_1) \int_{h_2} p(h_1|h_2) p(h_2|x) \right)$
- $p(h_1|x) \quad = \int_{h_2, f_2} \; p(h_1|f_2) p(f_2|h_2) \; p(h_2|x)$
- $p(h_1|x, \tilde{h}_2) = \int_{h_2, f_2, u_2} p(h_1|f_2) p(f_2|u_2, h_2) p(u_2|\tilde{h}_2) p(h_2|x)$
- $\log p(h_1|x, \tilde{h}_2) \geq \int_{h_2, f_2, u_2} \mathcal{Q} \log \frac{p(h_1|f_2) p(f_2|u_2, h_2) p(u_2|\tilde{h}_2) p(h_2|x)}{\mathcal{Q}}$

Direct marginalisation of h is intractable (O_o)

- New objective: $p(y|x) = \int_{h_1} \left(p(y|h_1) \int_{h_2} p(h_1|h_2) p(h_2|x) \right)$
- $p(h_1|x) \quad = \int_{h_2, f_2} \quad p(h_1|f_2) p(f_2|h_2) \quad p(h_2|x)$
- $p(h_1|x, \tilde{h}_2) = \int_{h_2, f_2, u_2} p(h_1|f_2) p(f_2|u_2, h_2) p(u_2|\tilde{h}_2) p(h_2|x)$
- $\log p(h_1|x, \tilde{h}_2) \geq \int_{h_2, f_2, u_2} \mathcal{Q} \log \frac{p(h_1|f_2) \cancel{p(f_2|u_2, h_2)} p(u_2|\tilde{h}_2) p(h_2|x)}{\mathcal{Q} = \cancel{p(f_2|u_2, h_2)} q(u_2) q(h_2)}$

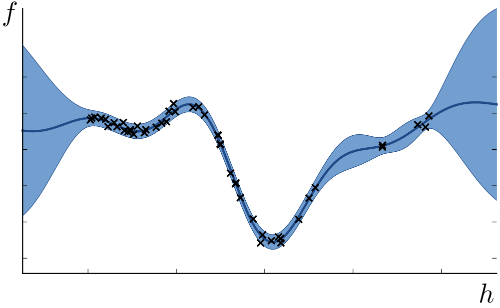
Direct marginalisation of h is intractable (O_o)

- New objective: $p(y|x) = \int_{h_1} \left(p(y|h_1) \int_{h_2} p(h_1|h_2) p(h_2|x) \right)$
- $p(h_1|x) \quad = \int_{h_2,f_2} \quad p(h_1|f_2) p(f_2|h_2) \quad p(h_2|x)$
- $p(h_1|x,\tilde{h}_2) = \int_{h_2,f_2,u_2} p(h_1|f_2) p(f_2|u_2,h_2) p(u_2|\tilde{h}_2) p(h_2|x)$
- $\log p(h_1|x,\tilde{h}_2) \geq \int_{h_2,f_2,u_2} \mathcal{Q} \log \frac{p(h_1|f_2)\cancel{p(f_2|u_2,h_2)}p(u_2|\tilde{h}_2)p(h_2|x)}{\mathcal{Q}=\cancel{p(f_2|u_2,h_2)}q(u_2)q(h_2)}$
- $\log p(h_1|x,\tilde{h}_2) \geq \int_{h_2,f_2,u_2} \mathcal{Q} \log \frac{p(h_1|f_2)p(u_2|\tilde{h}_2)p(h_2|x)}{\mathcal{Q}=q(u_2)q(h_2)}$

$p(u_2|\tilde{h}_2)$ contains $k(\tilde{h}_2,\tilde{h}_2)^{-1}$

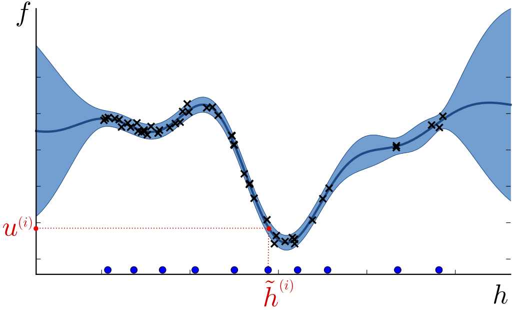
Inducing points: sparseness, tractability and Big Data

h_1	$\mathbf{f}_1$
h_2	$\mathbf{f}_2$
$\dots$	$\dots$
h_{30}	$\mathbf{f}_{30}$
h_{31}	$\mathbf{f}_{31}$
$\dots$	$\dots$
h_N	$\mathbf{f}_N$



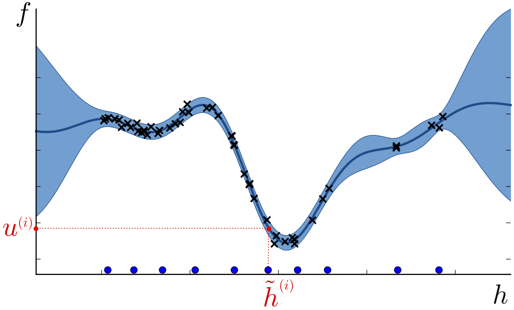
Inducing points: sparseness, tractability and Big Data

h_1	$\mathbf{f}_1$
h_2	$\mathbf{f}_2$
$\dots$	$\dots$
h_{30}	$\mathbf{f}_{30}$
$\tilde{h}^{(i)}$	$u^{(i)}$
h_{31}	$\mathbf{f}_{31}$
$\dots$	$\dots$
h_N	$\mathbf{f}_N$



Inducing points: sparseness, tractability and Big Data

h_1	$\mathbf{f}_1$
h_2	$\mathbf{f}_2$
$\dots$	$\dots$
h_{30}	$\mathbf{f}_{30}$
$\tilde{h}^{(i)}$	$u^{(i)}$
h_{31}	$\mathbf{f}_{31}$
$\dots$	$\dots$
h_N	$\mathbf{f}_N$



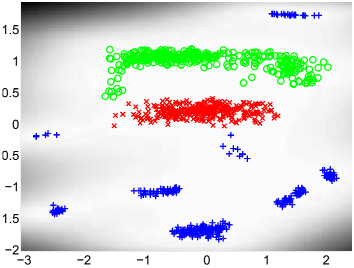
- Inducing points originally introduced for faster **(sparse) GPs**
- Our manipulation allows to **compress information** from the inputs of every layer (var. compression)
- This induces **tractability**
- Viewing them as **global variables**
  $\Rightarrow$ extension to **Big Data** [Hensman et al., UAI 2013]

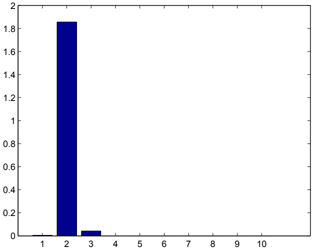
Automatic dimensionality detection

- Achieved by employing *automatic relevance determination (ARD)* priors for the mapping f.
- $f \sim \mathcal{GP}(\mathbf{0}, k_f)$ with:

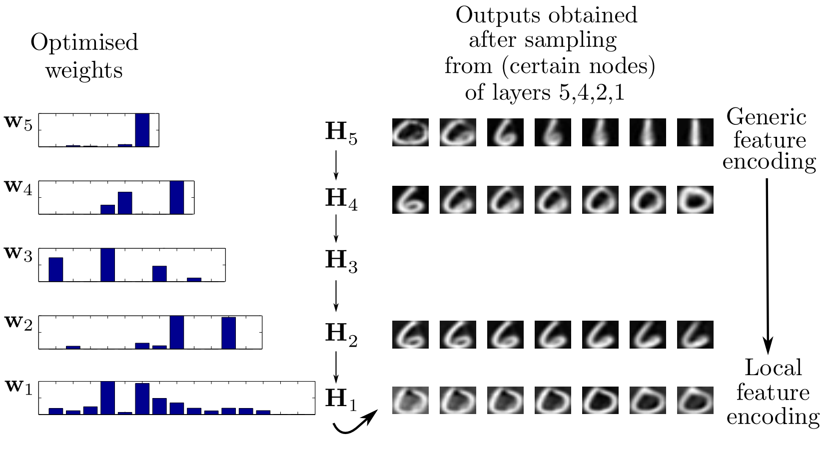
$$k_f(\mathbf{x}_i, \mathbf{x}_j) = \sigma^2 \exp\left(-\frac{1}{2}\sum_{q=1}^{Q} w_q (x_{i,q} - x_{j,q})^2\right)$$

- Example:





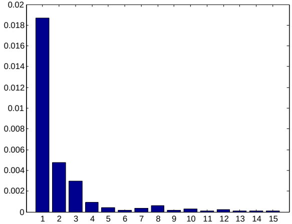
Deep GP: MNIST example



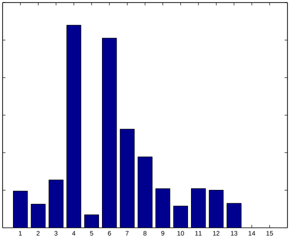
▸ http://staffwww.dcs.sheffield.ac.uk/people/A.Damianou/research/index.html#DeepGPs

MNIST: The first layer

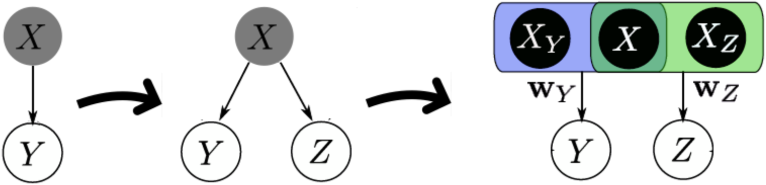
1 layer GP-LVM:



5 layer deep GP (showing 1st layer):

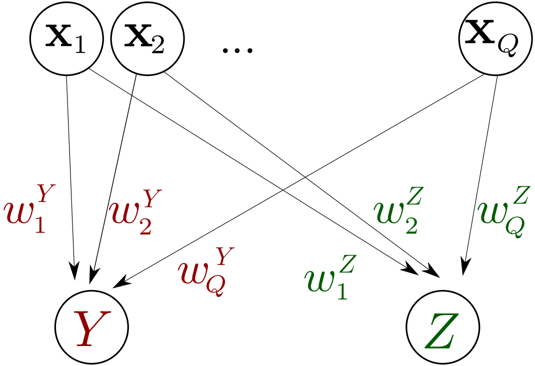


Manifold Relevance Determination

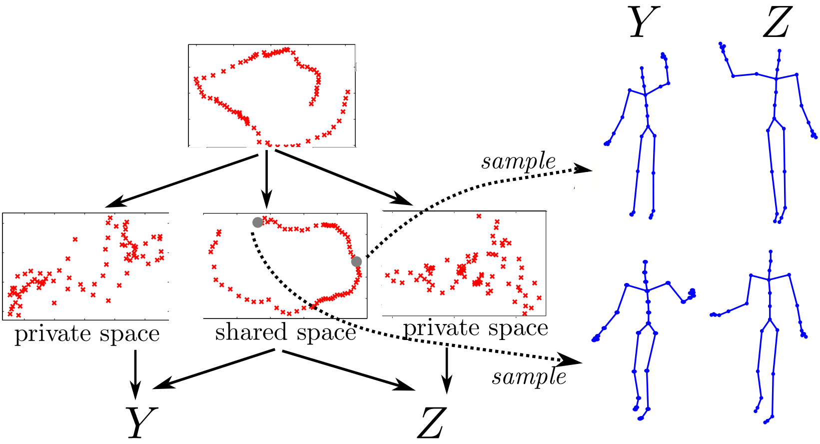


- Observations come into two different *views*: Y and Z.
- The latent space is segmented into parts private to Y, private to Z and shared between Y and Z.
- Used for data consolidation and discovering commonalities.

MRD weights



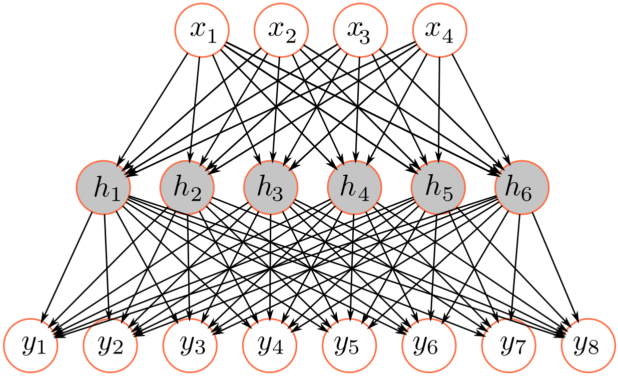
Deep GPs: Another multi-view example



Automatic structure discovery

Tools:

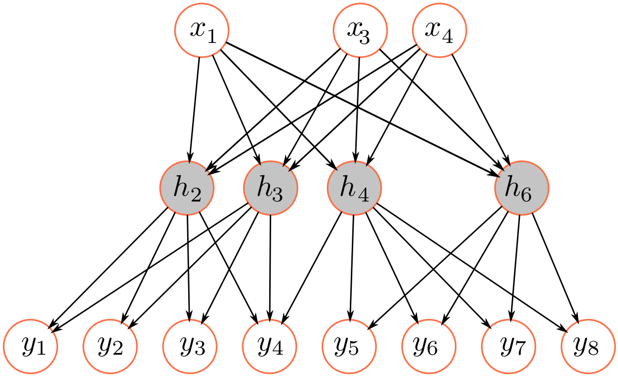
- ARD: Eliminate uncessary nodes/connections
- MRD: Conditional independencies
- Approximating evidence: Number of layers (?)



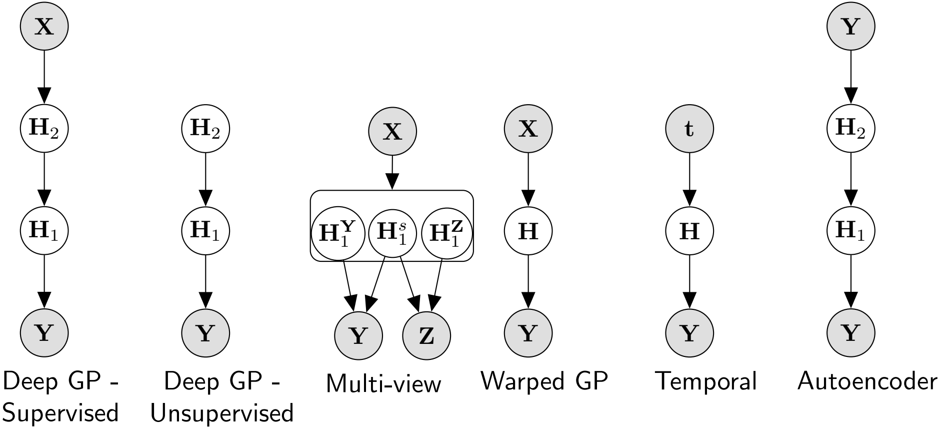
Automatic structure discovery

Tools:

- ARD: Eliminate uncessary nodes/connections
- MRD: Conditional independencies
- Approximating evidence: Number of layers (?)



Deep GP variants



Summary

- A deep GP is not a GP.
- Sampling is straight-forward. Regularization and training needs to be worked out.
- The solution is a special treatment of auxiliary variables.
- Many variants: multi-view, temporal, autoencoders ...
- Future: how to make it scalable?
- Future: how does it compare to / complement more traditional deep models?

Thanks

Thanks to Neil Lawrence, James Hensman, Michalis Titsias, Carl Henrik Ek.

References:

- N. D. Lawrence (2006) "The Gaussian process latent variable model" Technical Report no CS-06-03, The University of Sheffield, Department of Computer Science
- N. D. Lawrence (2006) "Probabilistic dimensional reduction with the Gaussian process latent variable model" (talk)
- C. E. Rasmussen (2008), "Learning with Gaussian Processes", Max Planck Institute for Biological Cybernetics, Published: Feb. 5, 2008 (Videolectures.net)
- Carl Edward Rasmussen and Christopher K. I. Williams. Gaussian Processes for Machine Learning. MIT Press, Cambridge, MA, 2006. ISBN 026218253X.
- M. K. Titsias (2009), "Variational learning of inducing variables in sparse Gaussian processes", AISTATS 2009
- A. C. Damianou, M. K. Titsias and N. D. Lawrence (2011), "Variational Gaussian process dynamical systems", NIPS 2011
- A. C. Damianou, C. H. Ek, M. K. Titsias and N. D. Lawrence (2012), "Manifold Relevance Determination", ICML 2012
- A. C. Damianou and N. D. Lawrence (2013), "Deep Gaussian processes", AISTATS 2013
- J. Hensman (2013), "Gaussian processes for Big Data", UAI 2013