Unsupervised and Composite Gaussian Processes

Carl Henrik Ek - che29@cam.ac.uk

15th of September, 2021

http://carlhenrik.com

Today

15/09/2021

Learning Theory

- $\mathcal{F}$ space of functions
- $\mathcal{A}$ learning algorithm
- $\mathcal{S} = \{(x_1, y_1), \ldots, (x_N, y_N)\}$
- $\mathcal{S} \sim P(\mathcal{X} \times \mathcal{Y})$
- $\ell(\mathcal{A}_{\mathcal{F}}(\mathcal{S}), x, y)$ loss function

Statistical Learning

$$e(\mathcal{S}, \mathcal{A}, \mathcal{F}) = \mathbb{E}_{P(\{\mathcal{X},\mathcal{Y}\})}\left[\ell(\mathcal{A}_{\mathcal{F}}(\mathcal{S}), x, y)\right]$$

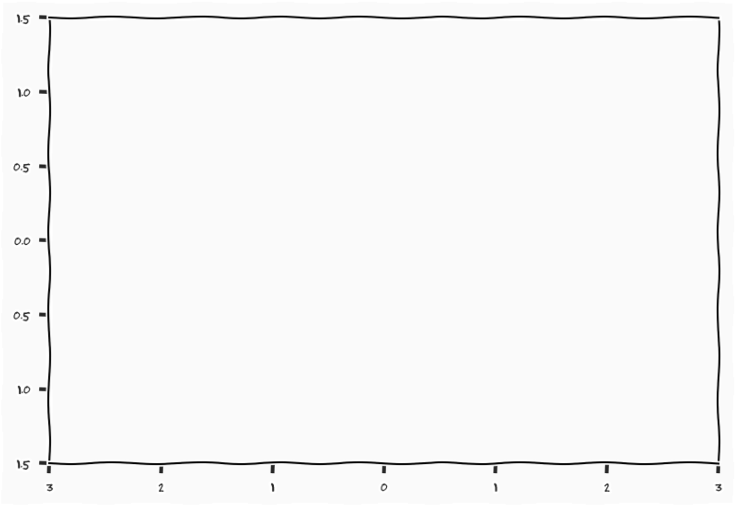
Statistical Learning

$$
\begin{aligned}
e(\mathcal{S}, \mathcal{A}, \mathcal{F}) &= \mathbb{E}_{P(\{\mathcal{X}, \mathcal{Y}\})}\left[\ell(\mathcal{A}_{\mathcal{F}}(\mathcal{S}), x, y)\right] \\
&\approx \frac{1}{M}\sum_{n=1}^{M} \ell(\mathcal{A}_{\mathcal{F}}(\mathcal{S}), x_n, y_n)
\end{aligned}
$$

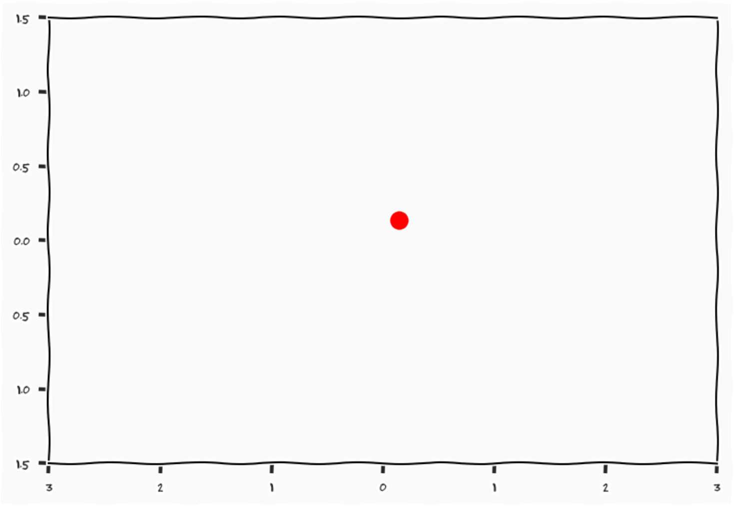
No Free Lunch

We can come up with a combination of $\{\mathcal{S}, \mathcal{A}, \mathcal{F}\}$ that makes $e(\mathcal{S}, \mathcal{A}, \mathcal{F})$ take an arbitary value

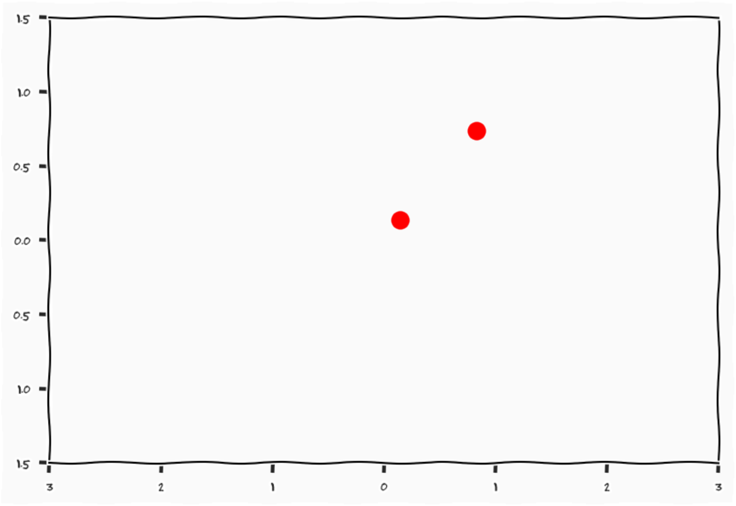
Example



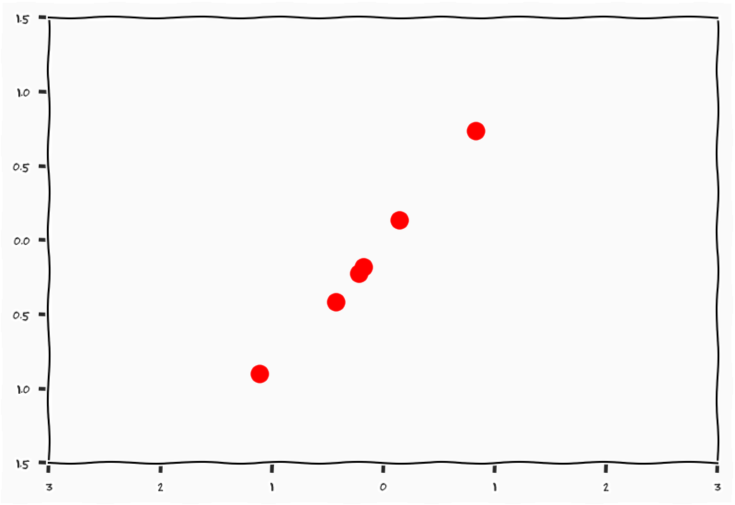
Example



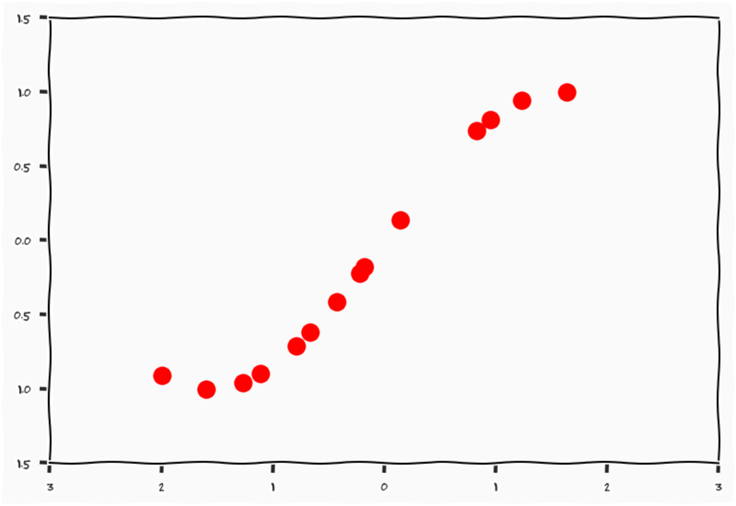
Example



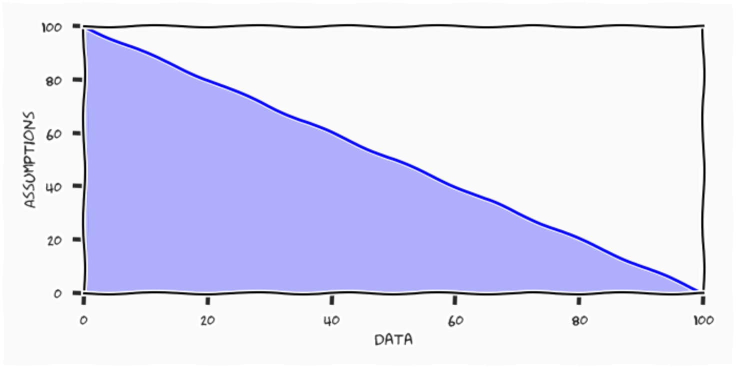
Example



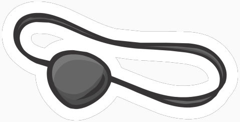
Example



Data and Knowledge



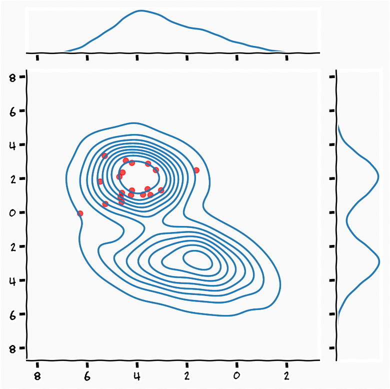
Assumptions: Algorithms



Statistical Learning

$$\mathcal{A}_{\mathcal{F}}(\mathcal{S})$$

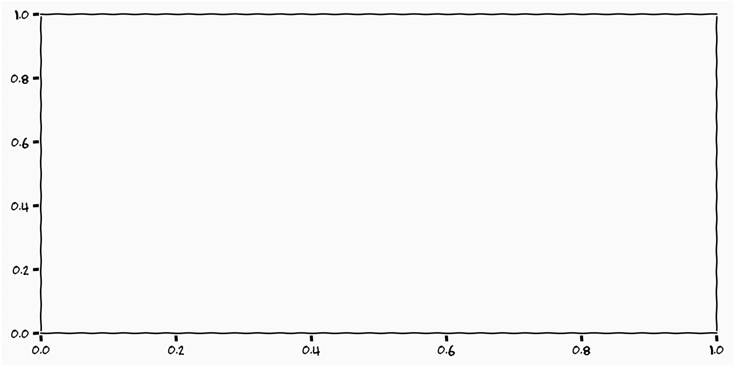
Assumptions: Biased Sample



Statistical Learning

$$\mathcal{A}_{\mathcal{F}}(\color{red}{\mathcal{S}})$$

Assumptions: Hypothesis space



Statistical Learning

$$\mathcal{A}_{\mathcal{F}}(\mathcal{S})$$

The No Free Lunch

- There seems to be a narrative that the more *flexible* a model is the better it is

The No Free Lunch

- There seems to be a narrative that the more *flexible* a model is the better it is
  - This is not true

The No Free Lunch

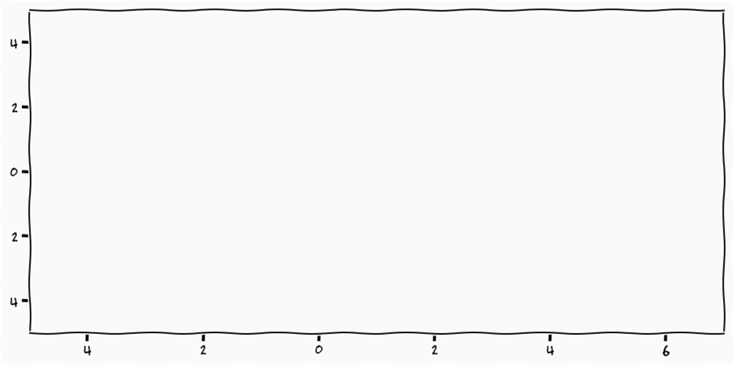
- There seems to be a narrative that the more *flexible* a model is the better it is
  - This is not true
- The best possible model has infinite support (nothing is excluded) but very focused mass

The No Free Lunch

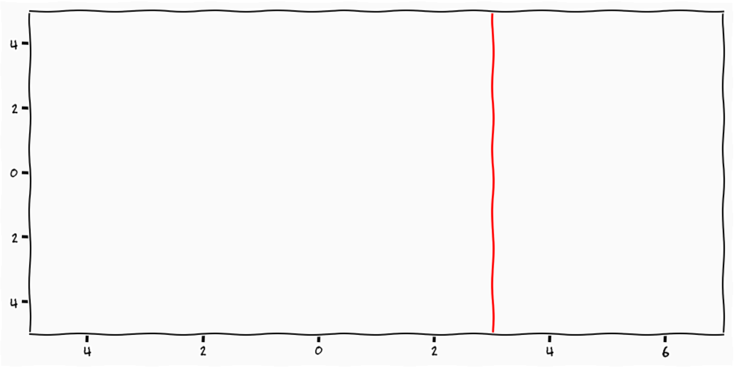
- There seems to be a narrative that the more *flexible* a model is the better it is
    - This is not true
- The best possible model has infinite support (nothing is excluded) but very focused mass
- *Your solution can only ever be interpreted in the light of your assumptions*

Gaussian Processes

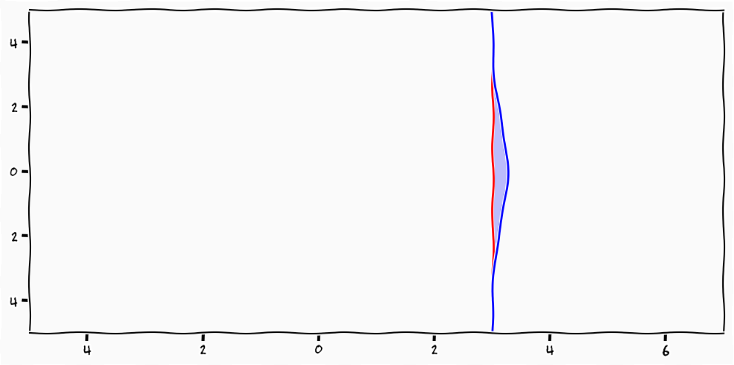
Gaussian Processes



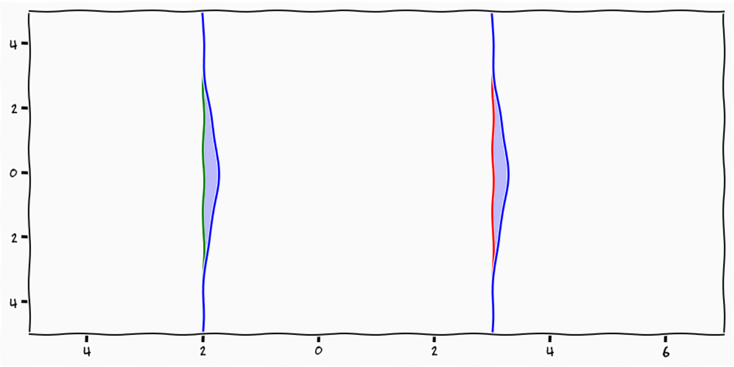
Gaussian Processes



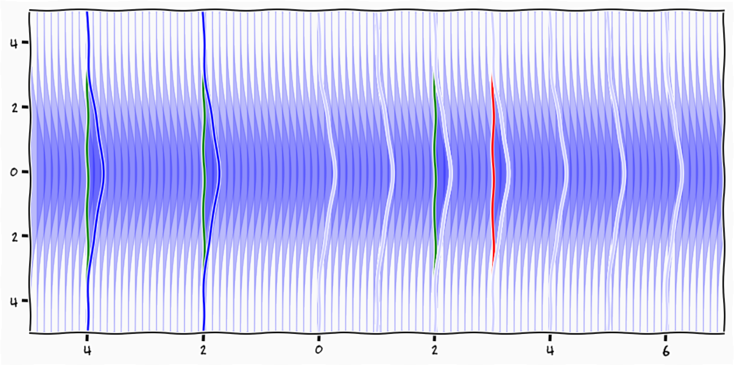
Gaussian Processes



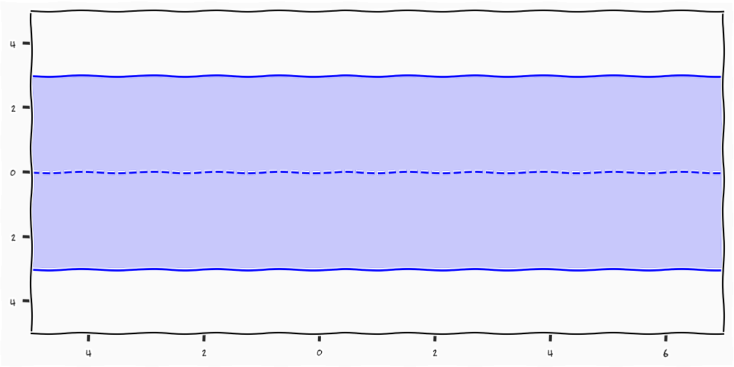
Gaussian Processes



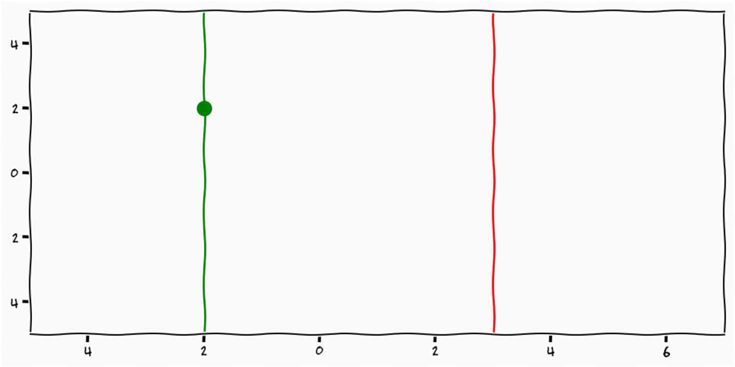
Gaussian Processes



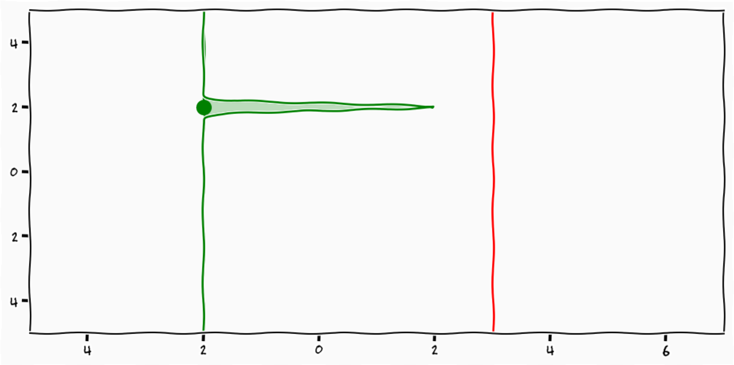
Gaussian Processes



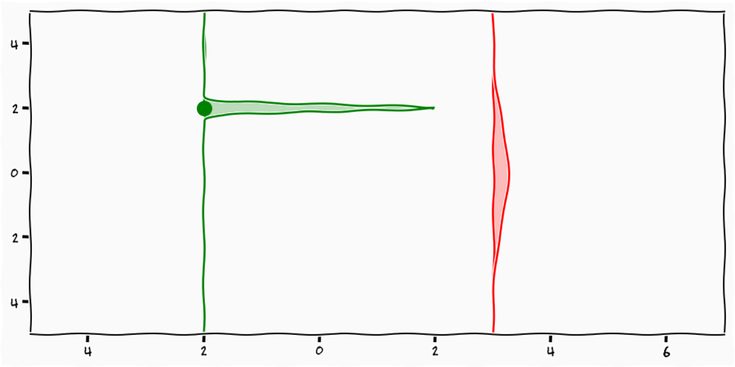
Gaussian Processes



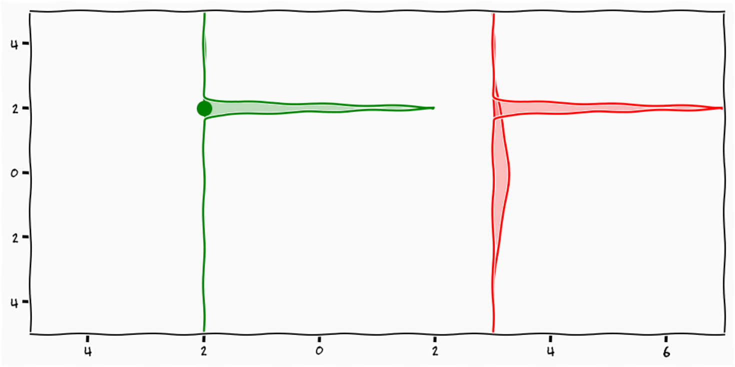
Gaussian Processes



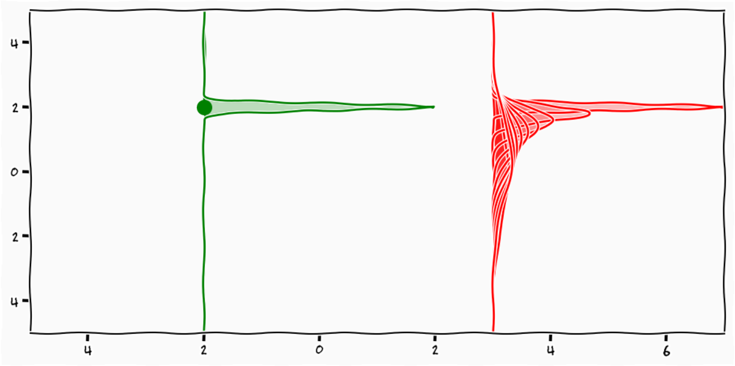
Gaussian Processes



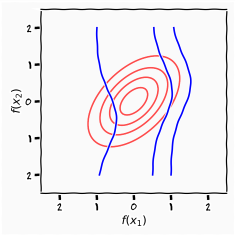
Gaussian Processes



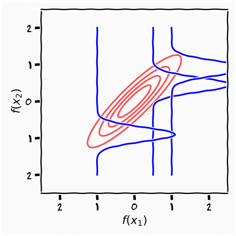
Gaussian Processes



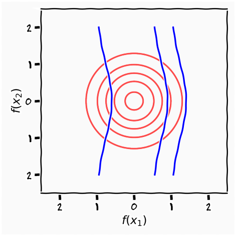
Conditional Gaussians



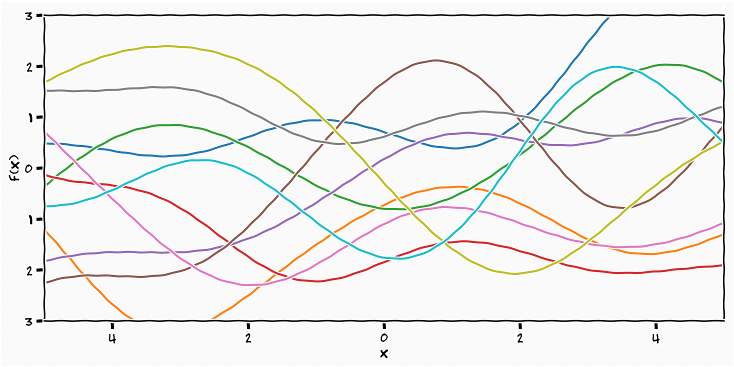
$$\mathsf{N}\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 & 0.5 \\ 0.5 & 1 \end{bmatrix}\right)$$



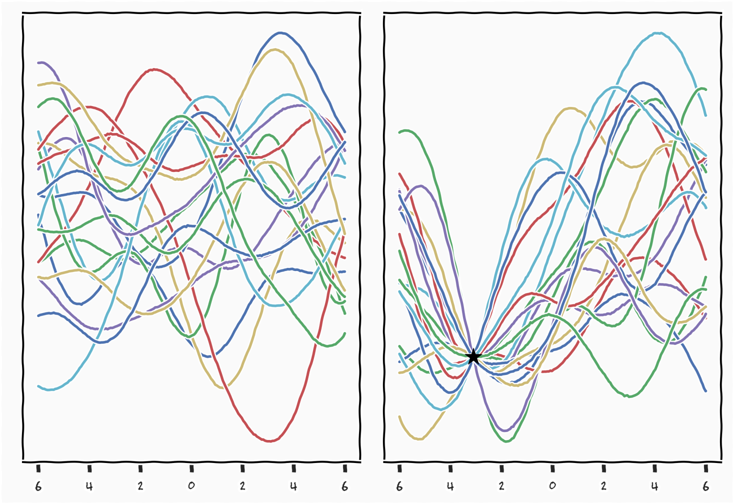
$$\mathsf{N}\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 & 0.9 \\ 0.9 & 1 \end{bmatrix}\right)$$



$$\mathsf{N}\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}\right)$$


3
2
1
0
1
2
3
f(x)
4
2
0
2
4
x

Gaussian Processes



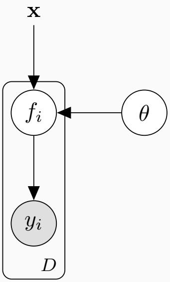
The Gaussian Identities

$$p(x_1, x_2) \qquad\qquad p(x_1) = \int p(x_1, x_2)\mathrm{d}x \quad p(x_1|x_2) = \frac{p(x_1, x_2)}{p(x_2)}$$

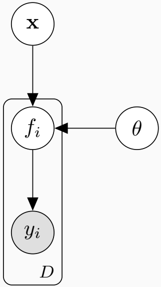
Gaussian Identities

Unsupervised Gaussian Processes

Unsupervised Learning

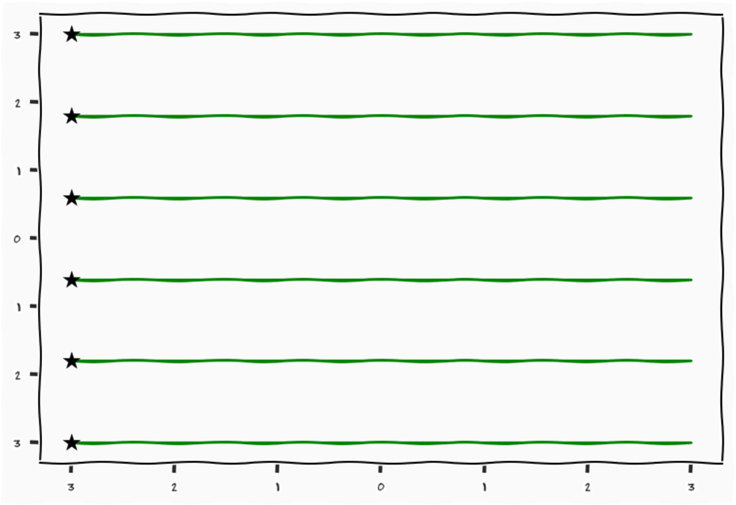


$p(y|x)$

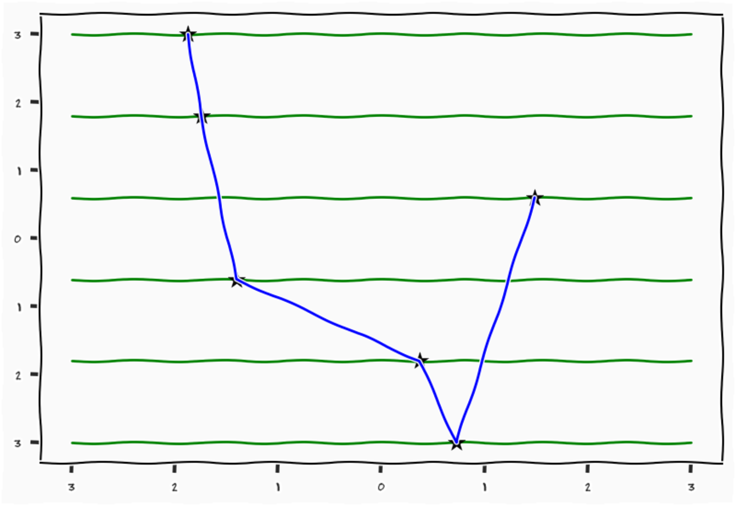


$p(y)$

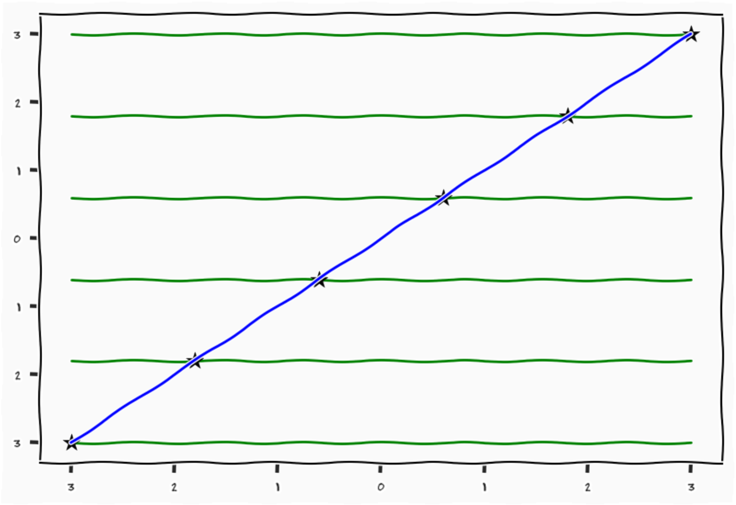
Unsupervised Learning



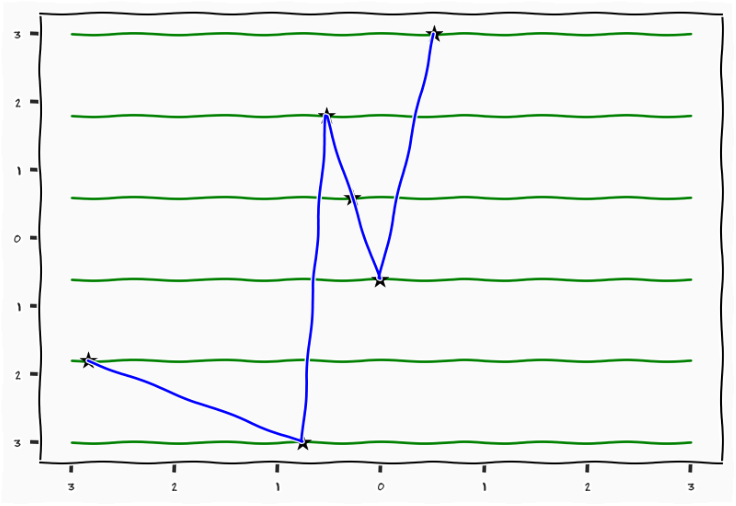
Unsupervised Learning



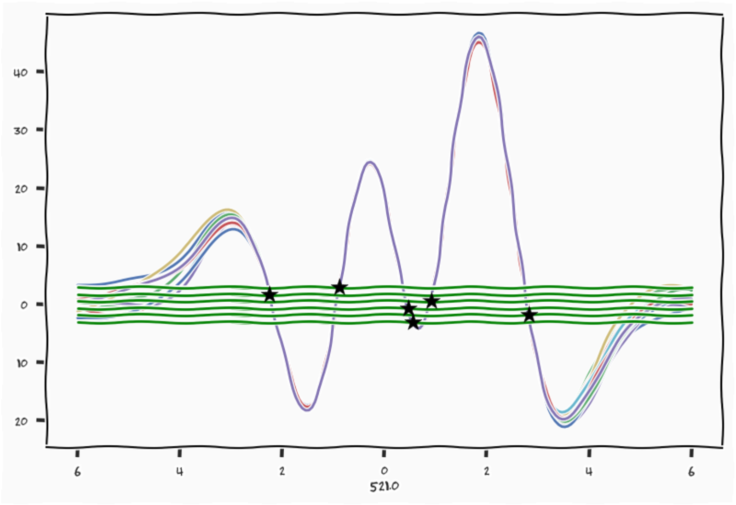
Unsupervised Learning



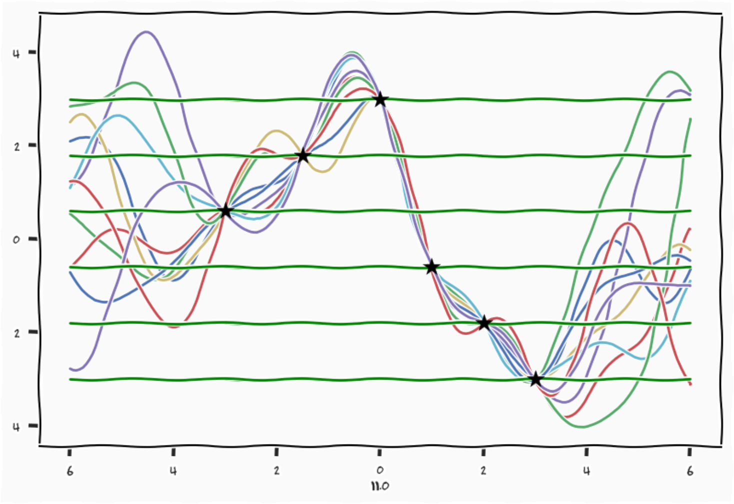
Unsupervised Learning



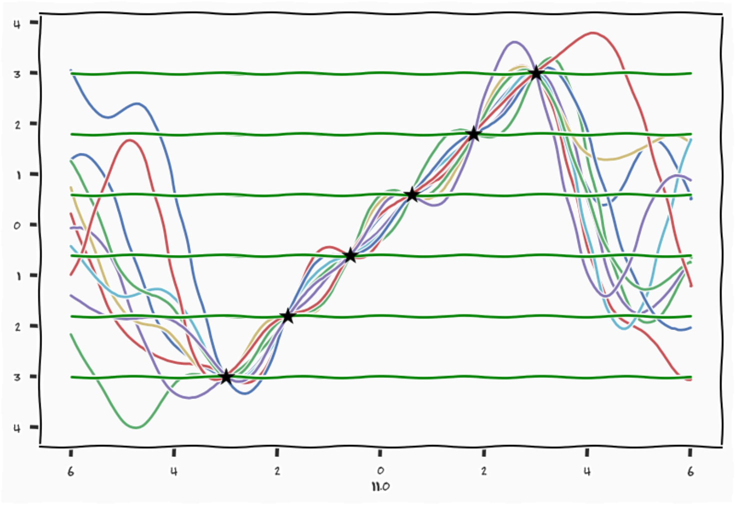
Unsupervised Learning

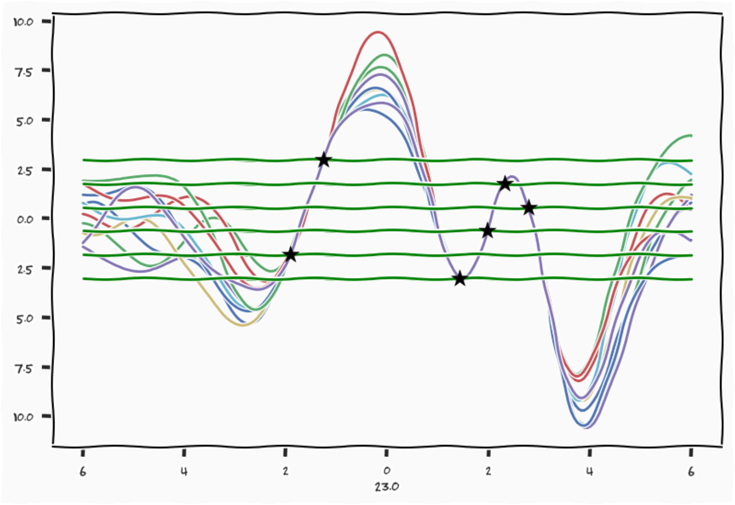


Unsupervised Learning

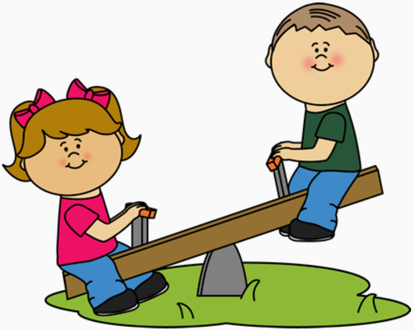


Unsupervised Learning




10.0
7.5
5.0
2.5
0.0
2.5
5.0
7.5
10.0
6
4
2
0
2
4
6
23.0

Priors



Priors

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

1. Priors that makes sense

   p(f) describes our belief/assumptions and defines our notion of complexity in the function

   p(x) expresses our belief/assumptions and defines our notion of complexity in the latent space

2. Now lets churn the handle

Relationship between x and data

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

- GP prior

$$p(f|x) \sim \mathcal{N}(0, K) \propto e^{-\frac{1}{2}(f^{\mathrm{T}}K^{-1}f)}$$

$$K_{ij} = e^{-(x_i - x_j)^{\mathrm{T}} M^{\mathrm{T}} M (x_i - x_j)}$$

Relationship between x and data

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

- GP prior

$$p(f|x) \sim \mathcal{N}(0, K) \propto e^{-\frac{1}{2}(f^{\mathrm{T}}K^{-1}f)}$$

$$K_{ij} = e^{-(x_i - x_j)^{\mathrm{T}}M^{\mathrm{T}}M(x_i - x_j)}$$

- Likelihood

$$p(y|f) \sim N(y|f, \beta) \propto e^{-\frac{1}{2\beta}\mathrm{tr}(y-f)^{\mathrm{T}}(y-f)}$$

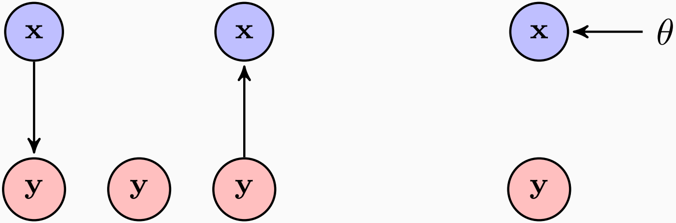
Laplace Integration

"Nature laughs at the difficulties of integrations"
– Simon Laplace

Approximate Inference

$$p(y) = \int p(y \mid x)p(x)\mathrm{d}x$$

Lower Bound



$$p(y) = \int_x p(y|x)p(x) = \frac{p(y|x)p(x)}{p(x|y)}$$

$$q_\theta(x) \approx p(x|y)$$

Variational Bayes

$$p(y)$$

Variational Bayes

$$\log p(y)$$

Variational Bayes

$$\log\, p(y) = \log\, p(y) + \int \log\, \frac{p(x|y)}{p(x|y)} \mathrm{d}x$$

Variational Bayes

$$\begin{aligned}\log\, p(y) &= \log\, p(y) + \int \log\, \frac{p(x|y)}{p(x|y)} \mathrm{d}x \\ &= \int q(x) \log\, p(y) \mathrm{d}x + \int q(x) \log\, \frac{p(x|y)}{p(x|y)} \mathrm{d}x\end{aligned}$$

Variational Bayes

$$
\begin{aligned}
\log\, p(y) &= \log\, p(y) + \int \log\, \frac{p(x|y)}{p(x|y)}\mathrm{d}x \\
&= \int q(x)\log\, p(y)\mathrm{d}x + \int q(x)\log\, \frac{p(x|y)}{p(x|y)}\mathrm{d}x \\
&= \int q(x)\log\, \frac{p(x|y)p(y)}{p(x|y)}\mathrm{d}x
\end{aligned}
$$

Variational Bayes

$$
\begin{aligned}
\log\, p(y) &= \log\, p(y) + \int \log\, \frac{p(x|y)}{p(x|y)}\mathrm{d}x \\
&= \int q(x)\log\, p(y)\mathrm{d}x + \int q(x)\log\, \frac{p(x|y)}{p(x|y)}\mathrm{d}x \\
&= \int q(x)\log\, \frac{p(x|y)p(y)}{p(x|y)}\mathrm{d}x = \int q(x)\log\, \frac{p(x,y)}{p(x|y)}\mathrm{d}x
\end{aligned}
$$

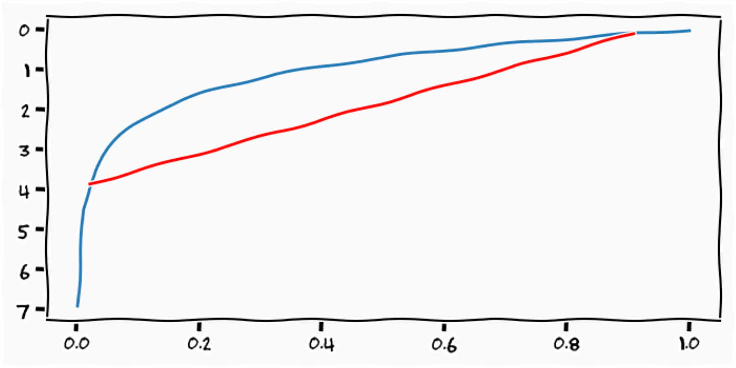
Variational Bayes

$$
\begin{aligned}
\log\, p(y) &= \log\, p(y) + \int \log\, \frac{p(x|y)}{p(x|y)} \mathrm{d}x \\
&= \int q(x) \log\, p(y) \mathrm{d}x + \int q(x) \log\, \frac{p(x|y)}{p(x|y)} \mathrm{d}x \\
&= \int q(x) \log\, \frac{p(x|y)p(y)}{p(x|y)} \mathrm{d}x = \int q(x) \log\, \frac{p(x,y)}{p(x|y)} \mathrm{d}x \\
&= \int q(x) \log\, \frac{q(x)}{q(x)} \mathrm{d}x + \int q(x) \log\, p(x,y) \mathrm{d}x + \int q(x)\ \log\, \frac{1}{p(x|y)} \mathrm{d}x
\end{aligned}
$$

Variational Bayes

$$
\begin{aligned}
\log p(y) &= \log p(y) + \int \log \frac{p(x|y)}{p(x|y)} \mathrm{d}x \\
&= \int q(x) \log p(y) \mathrm{d}x + \int q(x) \log \frac{p(x|y)}{p(x|y)} \mathrm{d}x \\
&= \int q(x) \log \frac{p(x|y)p(y)}{p(x|y)} \mathrm{d}x = \int q(x) \log \frac{p(x,y)}{p(x|y)} \mathrm{d}x \\
&= \int q(x) \log \frac{q(x)}{q(x)} \mathrm{d}x + \int q(x) \log p(x,y) \mathrm{d}x + \int q(x) \log \frac{1}{p(x|y)} \mathrm{d}x \\
&= {\color{blue}\int q(x) \log \frac{1}{q(x)} \mathrm{d}x + \int q(x) \log p(x,y) \mathrm{d}x} + {\color{red}\int q(x) \log \frac{q(x)}{p(x|y)} \mathrm{d}x}
\end{aligned}
$$

Jensen Inequality



$$\int \log(x)p(x)\mathrm{d}x \geq \log\left(\int xp(x)\mathrm{d}x\right)$$

moving the log outside the the integral is a lower-bound on the integral

The "posterior" term

$$\int q(x)\,\log\,\frac{q(x)}{p(x|y)}\mathrm{d}x$$

The "posterior" term

$$\int q(x)\,\log\,\frac{q(x)}{p(x|y)}\mathrm{d}x = -\int q(x)\,\log\,\frac{p(x|y)}{q(x)}\mathrm{d}x$$

The "posterior" term

$$\begin{aligned}\color{red}{\int q(x)\,\log\frac{q(x)}{p(x|y)}\mathrm{d}x} &= -\int q(x)\,\log\frac{p(x|y)}{q(x)}\mathrm{d}x\\ &\geq -\log\int p(x|y)\mathrm{d}x\\ &= -\log 1 = 0\end{aligned}$$

The "posterior" term

$$\int q(x) \, \log \, \frac{q(x)}{p(x|y)} \mathrm{d}x$$

The "posterior" term

$$\int q(x)\,\log\frac{q(x)}{p(x|y)}\mathrm{d}x = \{\text{Lets assume that } q(x) = p(x|y)\}$$

The "posterior" term

$$\int q(x) \log \frac{q(x)}{p(x|y)} \mathrm{d}x = \{\text{Lets assume that } q(x) = p(x|y)\}$$

$$= \int p(x|y) \log \underbrace{\frac{p(x|y)}{p(x|y)}}_{=1} \mathrm{d}x$$

The "posterior" term

$$
\begin{aligned}
\int q(x) \log \frac{q(x)}{p(x|y)} \mathrm{d}x &= \{\text{Lets assume that } q(x) = p(x|y)\} \\
&= \int p(x|y) \log \underbrace{\frac{p(x|y)}{p(x|y)}}_{=1} \mathrm{d}x \\
&= 0
\end{aligned}
$$

Kullback-Leibler Divergence

$$KL(q(x)||p(x|y)) = \int q(x) \log \frac{q(x)}{p(x|y)} \mathrm{d}x$$

- Measure of divergence between distributions
- Not a metric (not symmetric)
- $KL(q(x)||p(x|y)) = 0 \Leftrightarrow q(x) = p(x|y)$
- $KL(q(x)||p(x|y)) \geq 0$

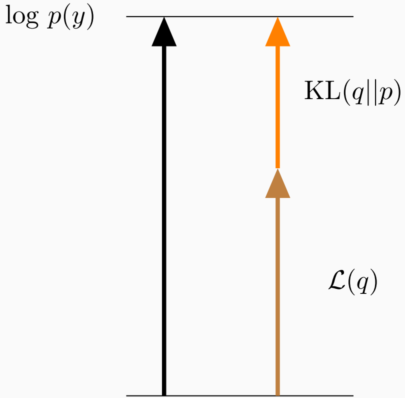
Variational Bayes

$$\log p(y) = \int q(x) \log \frac{1}{q(x)} \mathrm{d}x + \int q(x) \log p(x, y) \mathrm{d}x + \int q(x) \log \frac{q(x)}{p(x|y)} \mathrm{d}x$$

$$\geq -\int q(x) \log q(x) \mathrm{d}x + \int q(x) \log p(x, y) \mathrm{d}x$$

- The Evidence Lower BOnd
- Tight if $q(x) = p(x|y)$

Deterministic Approximation



ELBO

$$\log p(y) \geq -\int q(x)\log q(x)\mathrm{d}x + \int q(x)\log p(x,y)\mathrm{d}x$$
$$= \mathbb{E}_{q(x)}\left[\log p(x,y)\right] - H(q(x)) = \mathcal{L}(q(x))$$

- if we maximise the ELBO we,
  - find an approximate posterior
  - lower bound the marginal likelihood
- *maximising* $p(y)$ is learning
- finding $q(x) \approx p(x|y)$ is prediction

How to choose Q?

$$\mathcal{L}(q(x)) = \mathbb{E}_{q(x)}\left[\log p(x,y)\right] - H(q(x))$$

- We have to be able to compute an expectation over the joint distribution
- The second term should be trivial

Lower Bound[1]

$$\mathcal{L} = \int_x q(x) \log \left(\frac{p(y, f, x)}{q(x)} \right)$$

[1] Damianou, 2015

Lower Bound[1]

$$
\begin{aligned}
\mathcal{L} &= \int_x q(x) \log \left(\frac{p(y, f, x)}{q(x)} \right) \\
&= \int_x q(x) \log \left(\frac{p(y \mid f) p(f \mid x) p(x))}{q(x)} \right)
\end{aligned}
$$

[1] Damianou, 2015

Lower Bound[1]

$$
\begin{aligned}
\mathcal{L} &= \int_x q(x) \log \left(\frac{p(y, f, x)}{q(x)} \right) \\
&= \int_x q(x) \log \left(\frac{p(y \mid f) p(f \mid x) p(x))}{q(x)} \right) \\
&= \int_x q(x) \log p(y \mid f) p(f \mid x) - \int_x q(x) \log \frac{q(x)}{p(x)}
\end{aligned}
$$

[1] Damianou, 2015

Lower Bound[1]

$$
\begin{aligned}
\mathcal{L} &= \int_x q(x) \log \left(\frac{p(y, f, x)}{q(x)} \right) \\
&= \int_x q(x) \log \left(\frac{p(y \mid f) p(f \mid x) p(x))}{q(x)} \right) \\
&= \int_x q(x) \log p(y \mid f) p(f \mid x) - \int_x q(x) \log \frac{q(x)}{p(x)} \\
&= \tilde{\mathcal{L}} - \mathrm{KL}(q(x) \parallel p(x))
\end{aligned}
$$

[1] Damianou, 2015

Lower Bound

$$\tilde{\mathcal{L}} = \int q(x) \log p(y|f)p(f|x)\mathrm{d}f\mathrm{d}x$$

- Has not eliviate the problem at all, x still needs to go through f to reach the data
- Idea of sparse approximations[2]

[2] Candela et al., 2005

Lower Bound [3]

$$p(f, u \mid x, z)$$

- Add another set of samples from the same prior
- Conditional distribution

[3] Titsias et al., 2010

Lower Bound [3]

$$p(f, u \mid x, z) = p(f \mid u, x, z)p(u \mid z)$$

- Add another set of samples from the same prior
- Conditional distribution

[3] Titsias et al., 2010

Lower Bound [3]

$$
\begin{aligned}
p(f, u \mid x, z) &= p(f \mid u, x, z)p(u \mid z) \\
&= \mathcal{N}(f \mid K_{fu}K_{uu}^{-1}u, K_{ff} - K_{fu}K_{uu}^{-1}K_{uf})\mathcal{N}(u \mid \mathbf{0}, K_{uu})
\end{aligned}
$$

- Add another set of samples from the same prior
- Conditional distribution

[3] Titsias et al., 2010

Lower Bound

$$p(y, f, u, x \mid z) = p(y \mid f)p(f \mid u, x)p(u \mid z)p(x)$$

- we have done nothing to the model, just project an additional set of marginals from the GP
- *however* we will now interpret u and z not as random variables but variational parameters
- i.e. the variational distribution $q(\cdot)$ is parametrised by these

Lower Bound

- Variational distributions are approximations to intractable posteriors,

$$
\begin{aligned}
q(u) &\approx p(u \mid y, x, z, f) \\
q(f) &\approx p(f \mid u, x, z, y) \\
q(x) &\approx p(x \mid y)
\end{aligned}
$$

Lower Bound

- Variational distributions are approximations to intractable posteriors,

$$q(u) \approx p(u \mid y, x, z, f)$$
$$q(f) \approx p(f \mid u, x, z, y)$$
$$q(x) \approx p(x \mid y)$$

- Bound is tight if u completely represents f i.e. u is sufficient statistics for f

$$q(f) \approx p(f \mid u, x, z, y) = p(f \mid u, x, z)$$

Lower Bound

$$\tilde{\mathcal{L}} = \int_{x,f,u} q(f)q(u)q(x)\log\frac{p(y,f,y \mid x,z)}{q(f)q(u)}$$

Lower Bound

$$\tilde{\mathcal{L}} = \int_{x,f,u} q(f)q(u)q(x) \log \frac{p(y,f,y \mid x,z)}{q(f)q(u)}$$
$$= \int_{x,f,u} q(f)q(u)q(x) \log \frac{p(y \mid f)p(f \mid u,x,z)p(u \mid z)}{q(f)q(u)}$$

- Assume that u is sufficient statistics of f

$$q(f) = p(f \mid u,x,z)$$

Lower Bound

$$\tilde{\mathcal{L}} = \int_{x,f,u} q(f)q(u)q(x) \log \frac{p(y \mid f)p(f \mid u, x, z)p(u \mid z)}{q(f)q(u)}$$

Lower Bound

$$\begin{aligned}
\tilde{\mathcal{L}} &= \int_{x,f,u} q(f)q(u)q(x) \log \frac{p(y \mid f)p(f \mid u,x,z)p(u \mid z)}{q(f)q(u)} \\
&= \int_{x,f,u} p(f \mid u,x,z)q(u)q(x) \log \frac{p(y \mid f)p(f \mid u,x,z)p(u \mid z)}{p(f \mid u,x,z)q(u)}
\end{aligned}$$

Lower Bound

$$\begin{aligned}\tilde{\mathcal{L}} &= \int_{x,f,u} q(f)q(u)q(x)\log\frac{p(y \mid f)p(f \mid u,x,z)p(u \mid z)}{q(f)q(u)} \\ &= \int_{x,f,u} p(f \mid u,x,z)q(u)q(x)\log\frac{p(y \mid f)\textcolor{red}{p(f \mid u,x,z)}p(u \mid z)}{\textcolor{red}{p(f \mid u,x,z)}q(u)}\end{aligned}$$

Lower Bound

$$\begin{aligned}\tilde{\mathcal{L}} &= \int_{x,f,u} q(f)q(u)q(x)\log\frac{p(y \mid f)p(f \mid u,x,z)p(u \mid z)}{q(f)q(u)} \\ &= \int_{x,f,u} p(f \mid u,x,z)q(u)q(x)\log\frac{p(y \mid f)\textcolor{red}{p(f \mid u,x,z)}p(u \mid z)}{\textcolor{red}{p(f \mid u,x,z)}q(u)} \\ &= \int_{x,f,u} p(f \mid u,x,z)q(u)q(x)\log\frac{p(y \mid f)p(u \mid z)}{q(u)}\end{aligned}$$

Lower Bound

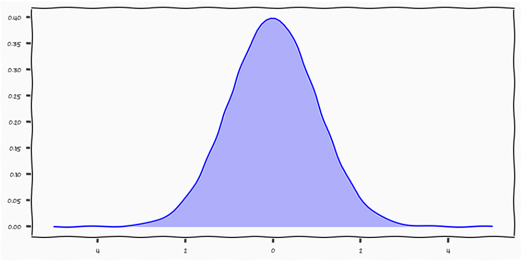
$$
\begin{aligned}
\tilde{\mathcal{L}} &= \int_{x,f,u} q(f)q(u)q(x)\log\frac{p(y \mid f)p(f \mid u,x,z)p(u \mid z)}{q(f)q(u)} \\
&= \int_{x,f,u} p(f \mid u,x,z)q(u)q(x)\log\frac{p(y \mid f){\color{red}p(f \mid u,x,z)}p(u \mid z)}{{\color{red}p(f \mid u,x,z)}q(u)} \\
&= \int_{x,f,u} p(f \mid u,x,z)q(u)q(x)\log\frac{p(y \mid f)p(u \mid z)}{q(u)} \\
&= \mathbb{E}_{p(f \mid u,x,z)}[p(y \mid f)] - \mathrm{KL}(q(u) \parallel p(u \mid z))
\end{aligned}
$$

Lower Bound

$$\mathcal{L} = \mathbb{E}_{p(f|u,x,z)}[p(y \mid f)] - \mathrm{KL}(q(u) \parallel p(u \mid z)) - \mathrm{KL}(q(x) \parallel p(x))$$

- Expectation tractable (for some co-variances)
- Allows us to place priors and not "regularisers" over the latent representation
- Stochastic inference Hensman et al., 2013
- Importantly $p(x)$ only appears in $\mathrm{KL}(\cdot \parallel \cdot)$ term!

Latent Space Priors



$$p(x) \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I})$$

Automatic Relevance Determination

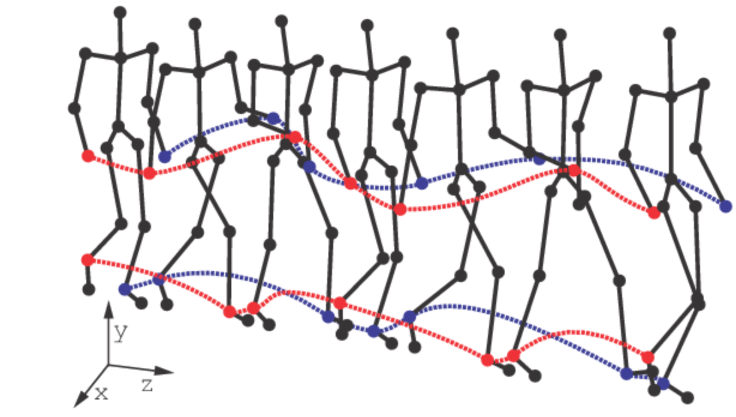
$$k(\mathbf{x}_i, \mathbf{x}_j) = \sigma e^{-\sum_d^D \alpha_d \cdot (x_{i,d} - x_{j,d})^2}$$

GPy

Code

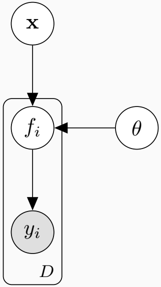
[]python RBF(...,ARD=True) Matern32(...,ARD=True)

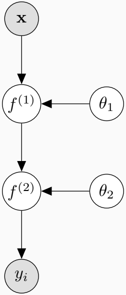
Dynamic Prior

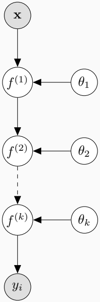


$$p(x \mid t) = \mathcal{N}(\mu_t, K_t)$$

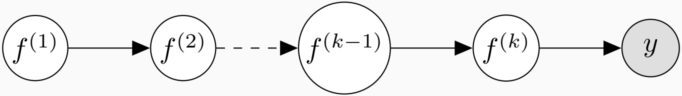
Composite Gaussian Processes







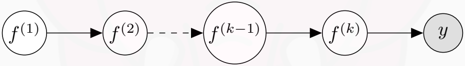
Composite Gaussian Processes [4]

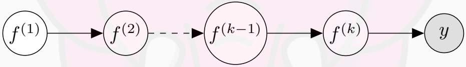


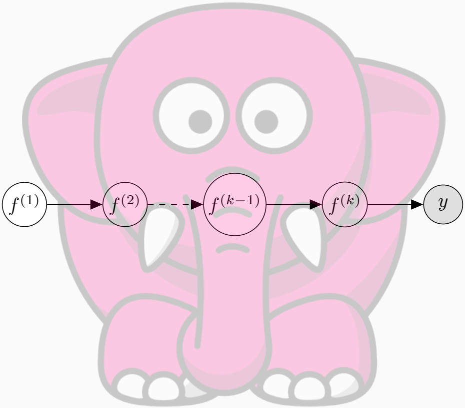
$$y = f^{(k)}(f^{(k-1)}(\cdots f^{(2)}(f^{(1)}(x))))$$

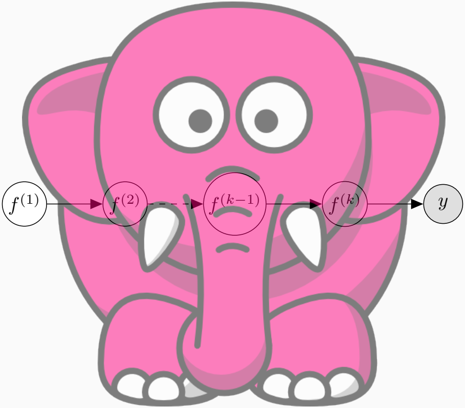
[4] Damianou et al., 2013

Composite Models

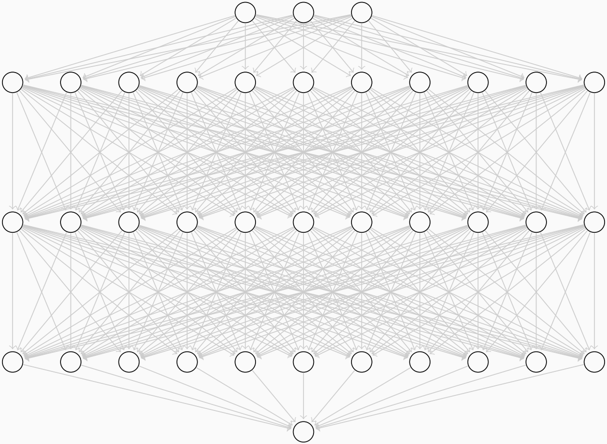

$f^{(1)}$
$f^{(2)}$
$f^{(k-1)}$
$f^{(k)}$
y


$f^{(1)}$
$f^{(2)}$
$f^{(k-1)}$
$f^{(k)}$
y


$f^{(1)}$
$f^{(2)}$
$f^{(k-1)}$
$f^{(k)}$
y


$f^{(1)}$
$f^{(2)}$
$f^{(k-1)}$
$f^{(k)}$
y

Neural Networks



$\boldsymbol{x} \in \mathbb{R}^d$

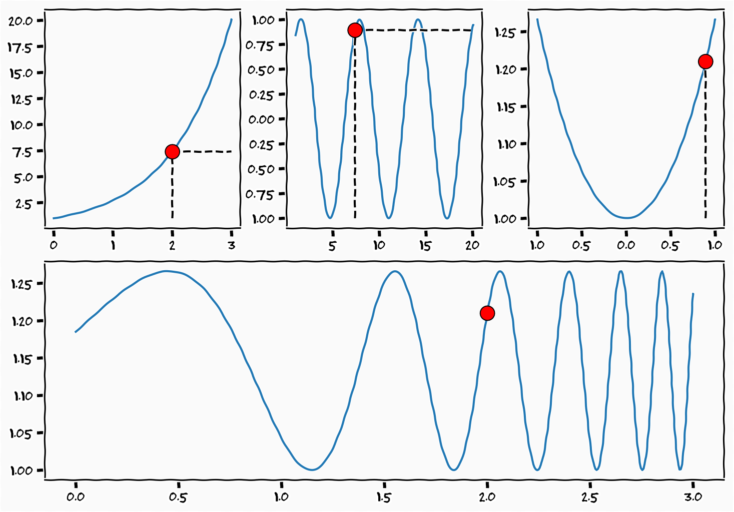
$\boldsymbol{z}_1 = \sigma(\boldsymbol{W}_1 \boldsymbol{x})$

$\boldsymbol{z}_2 = \sigma((\boldsymbol{W}_2 \boldsymbol{V}_1^\top)\boldsymbol{h}_1)$

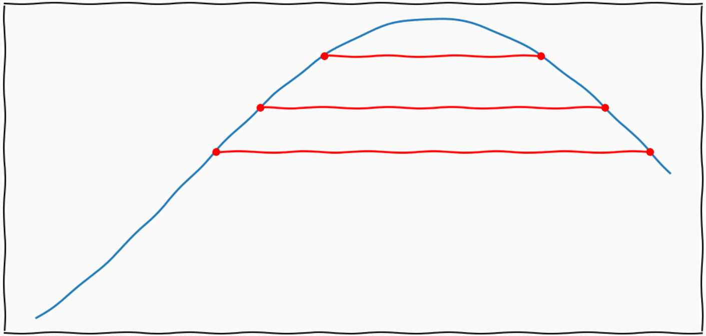
$\boldsymbol{z}_3 = \sigma((\boldsymbol{W}_3 \boldsymbol{V}_2^\top)\boldsymbol{h}_2)$

$\boldsymbol{y} = \boldsymbol{V}_3^\top \boldsymbol{z}_3$

What is a composite function?



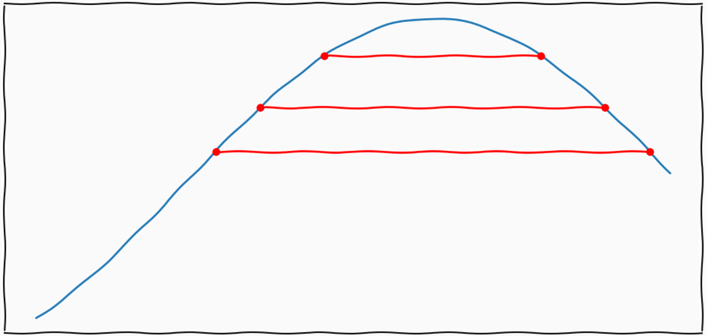
What Does Compositions Do?



$$\mathsf{Im}(f)[\mathcal{X}] = \{f(x) \mid x \in \mathcal{X}\}$$

$$\mathsf{Kern}(f)[\mathcal{X}] = \{(x, x') \mid f(x) = f(x'), \;\; (x, x') \in \mathcal{X} \times \mathcal{X}\}$$

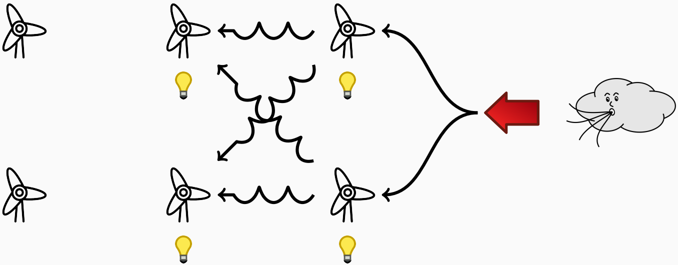
What Does Compositions Do?



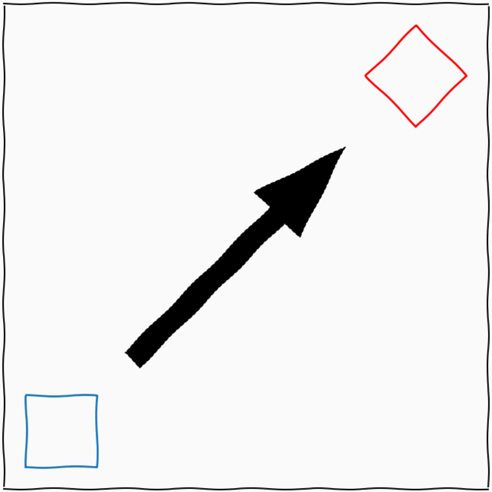
$$\text{Kern}(f_1) \subseteq \text{Kern}(f_{k-1} \circ \ldots \circ f_2 \circ f_1) \subseteq \text{Kern}(f_k \circ f_{k-1} \circ \ldots \circ f_2 \circ f_1)$$

$$\text{Im}(f_k \circ f_{k-1} \circ \ldots \circ f_2 \circ f_1) \subseteq \text{Im}(f_k \circ f_{k-1} \circ \ldots \circ f_2) \subseteq \ldots \subseteq \text{Im}(f_k)$$

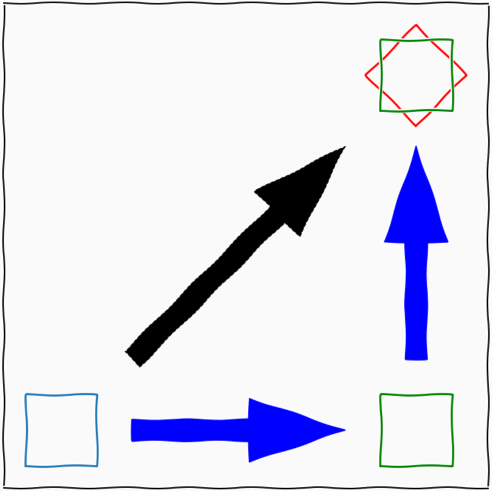
Why do we want composite functions?



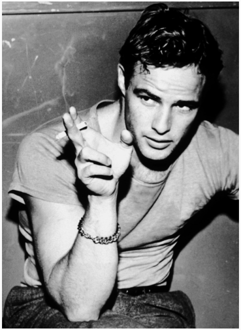
Why do we want composite functions?



Why do we want composite functions?

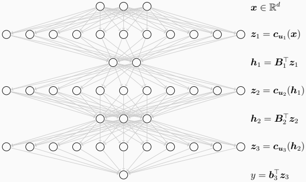


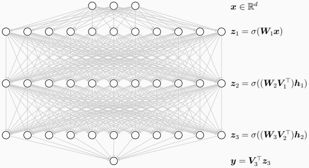
Because we want to hang out with the cool kids



Deep Learning is a bit like smoking, you know that its wrong but you do it anyway because you want to look cool.
– Fantomens Djungelordspråk

DGP vs DNN Neal, 1996





Posterior

- Approximate Posterior

$$\begin{aligned} p(f \mid u, x, z, y) \approx q(f) &= p(f \mid u, x, z) \\ &= \mathcal{N}(f \mid K_{fu}K_{uu}^{-1}u, K_{ff} - K_{fu}K_{uu}^{-1}K_{uf}) \end{aligned}$$

- Linear Mapping

$$\begin{aligned} \mathbb{E}[f(x)] &= K_{fu}K_{uu}^{-1}u = b^T c_u(x) \\ c_u(x) &= k(x, u) \end{aligned}$$

Prediction

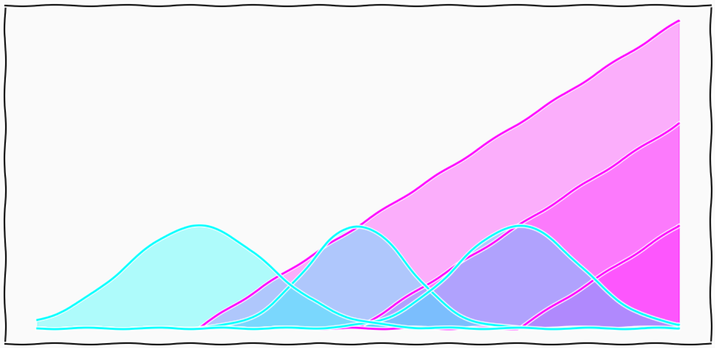
Composite GP predictive mean

$$\mathbb{E}[f_{DGP}(x)] = B_L^{\mathrm{T}} c_{u_L}(\cdots B_2^{\mathrm{T}} c_{u_2}(B_1^{\mathrm{T}}(x)))$$

Neural Network forward pass

$$f_{NN}(x) = V_L^{\mathrm{T}} \sigma(W_L \cdots V_2^{\mathrm{T}} \sigma(W_2 V_1^{\mathrm{T}} \sigma(W_1 x)))$$

"Activations"



DNNs as point estimates for DGPs[5]

$$c_u(\cdot) \sim \sigma(W\cdot)$$

- Define an equivalence between activation functions and co-variance
- Interdomain Gaussian Processes Lázaro-Gredilla et al., 2009

[5] Dutordoir et al., 2021a.

Same Same

Learning

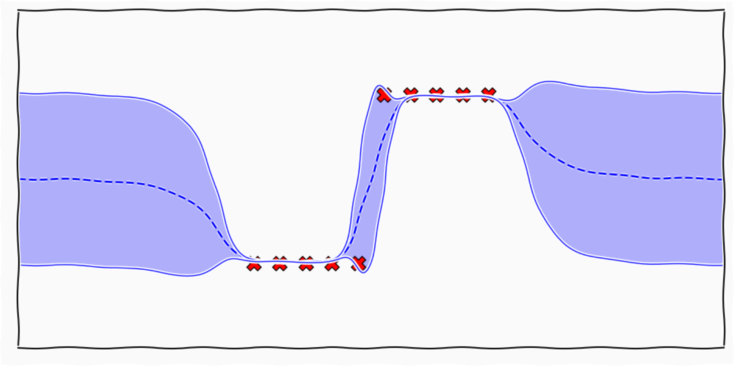
- Gaussian process

$$\operatorname*{argmax}_{\theta} \underbrace{\int p(y \mid f_L) p(f_L \mid f_{L-1}) \cdots p(f_2 \mid f_1) p(f_1) \mathrm{d} f_{L,L-1,\ldots,2,1}}_{p_\theta(y)}$$

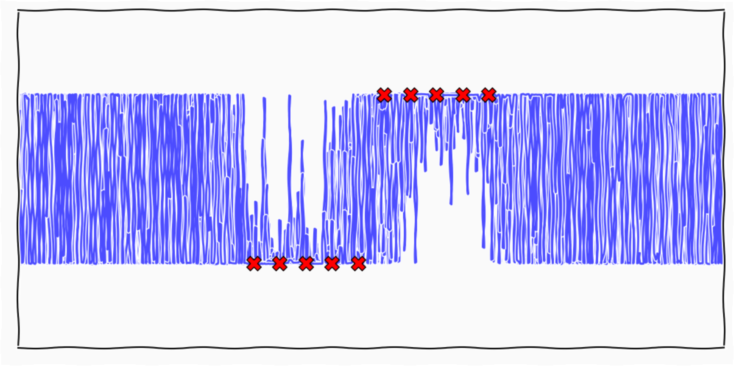
- Neural Network

$$\operatorname*{arxmax}_{W,V,\theta} \; \ell(W, V, \theta)$$

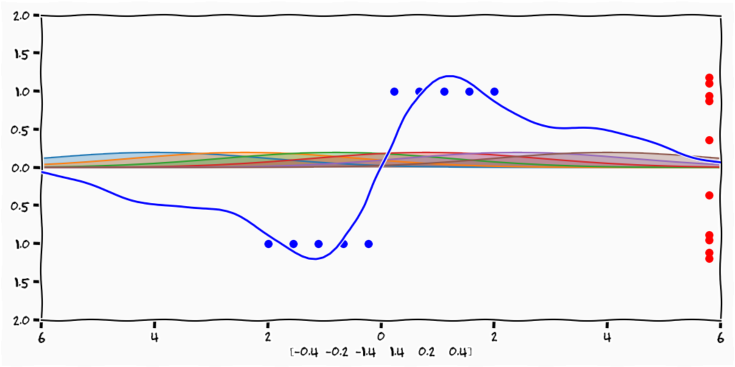
Composite GP Step



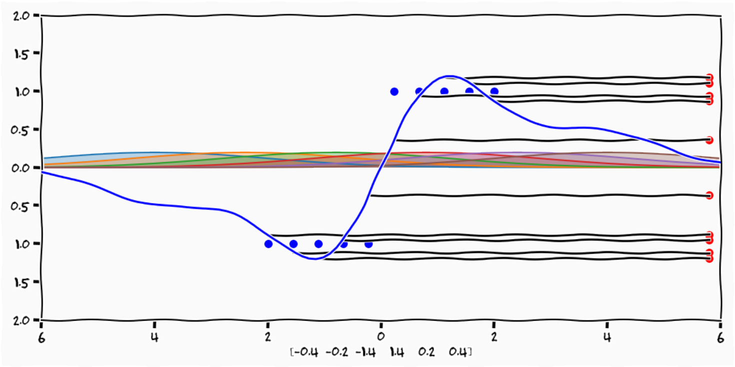
Composite GP Step



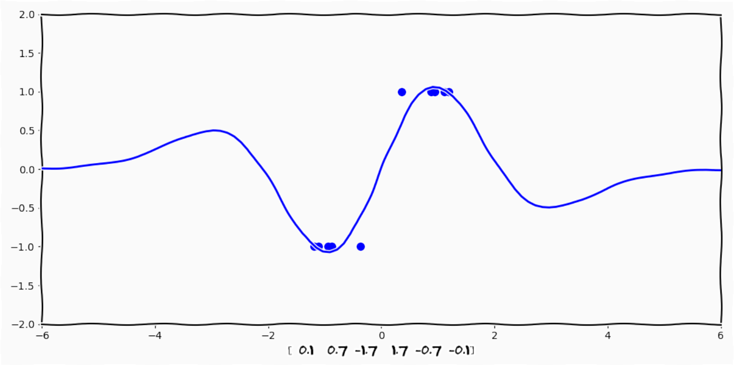
Composite Functions



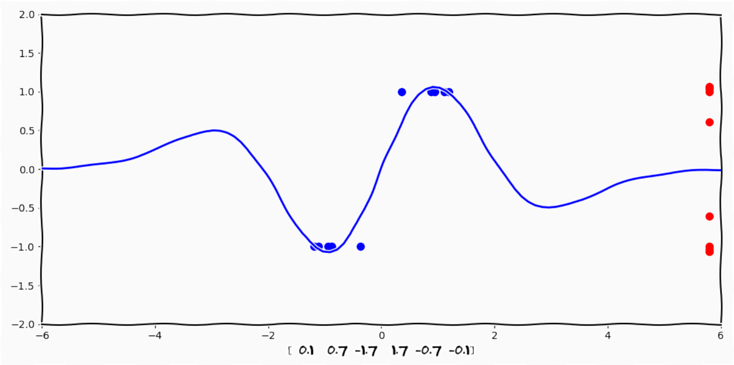
Composite Functions



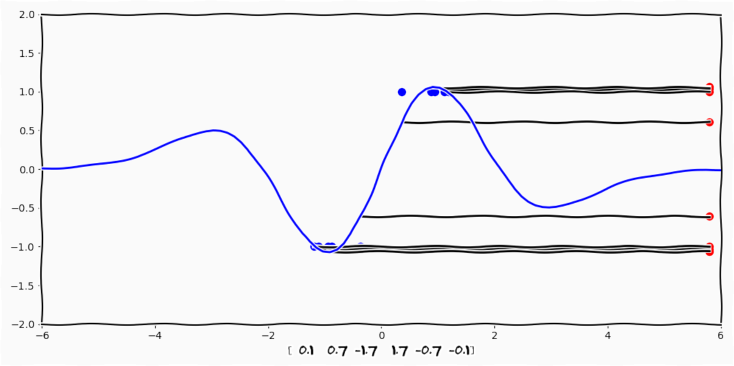
Composite Functions



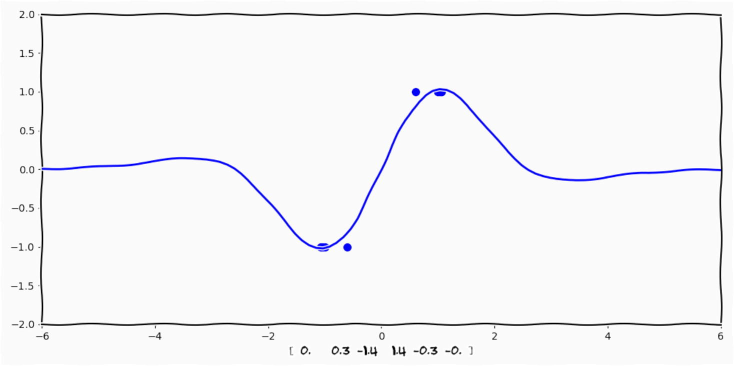
Composite Functions



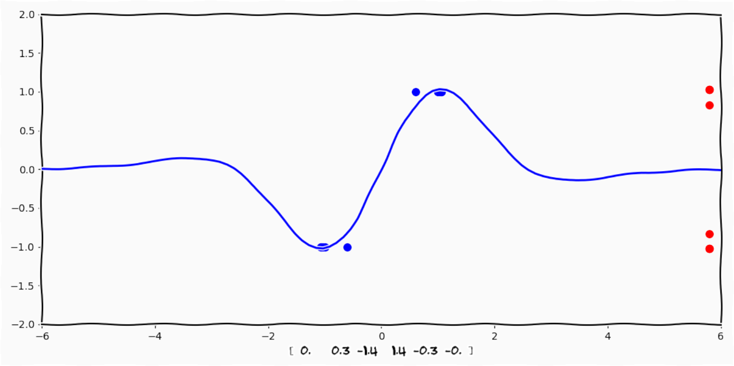
Composite Functions



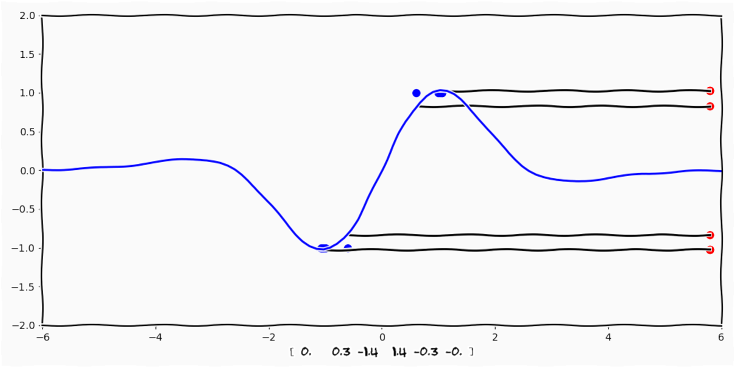
Composite Functions



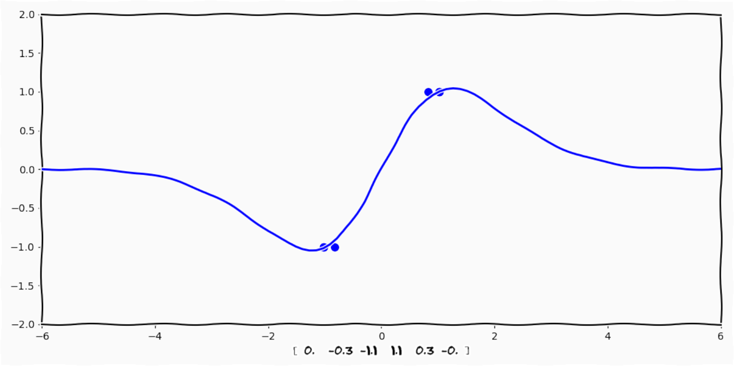
Composite Functions



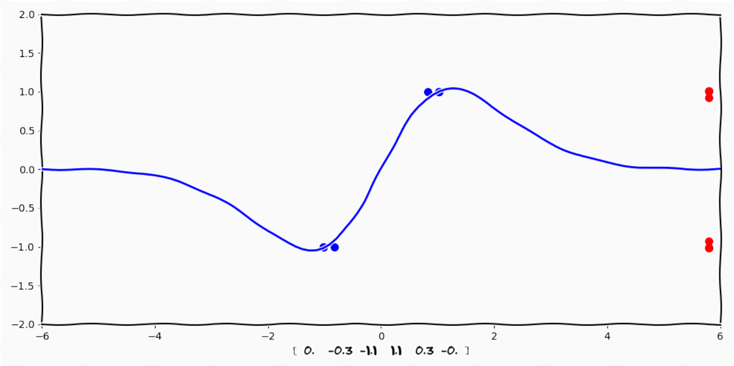
Composite Functions



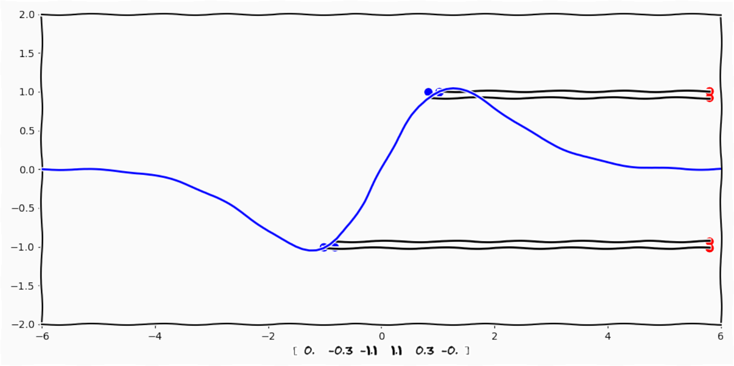
Composite Functions



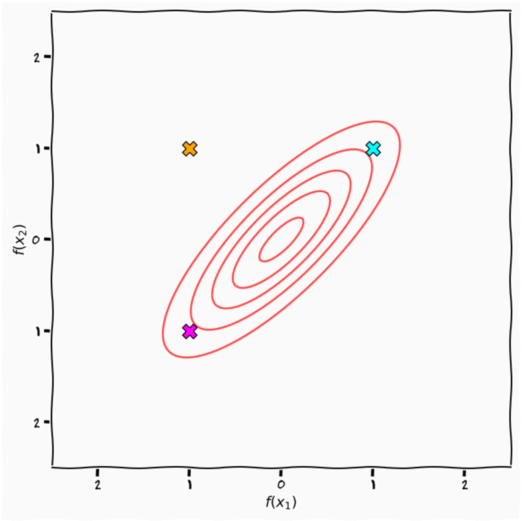
Composite Functions



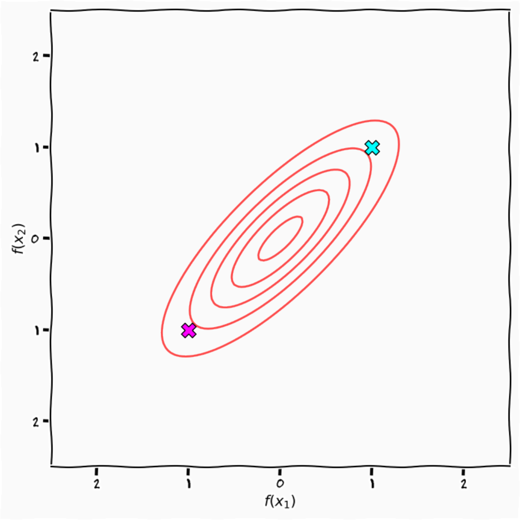
Composite Functions



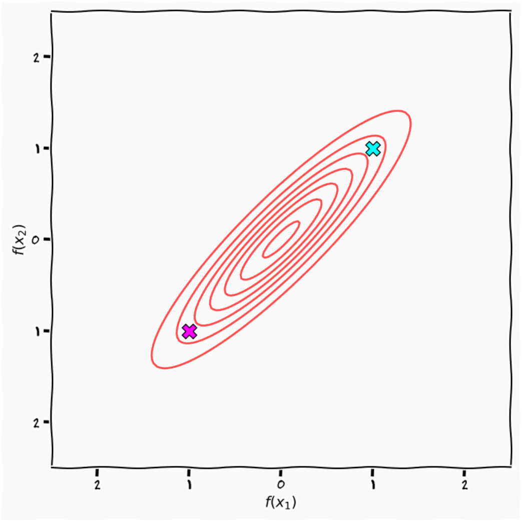
Composite Gaussian Processes



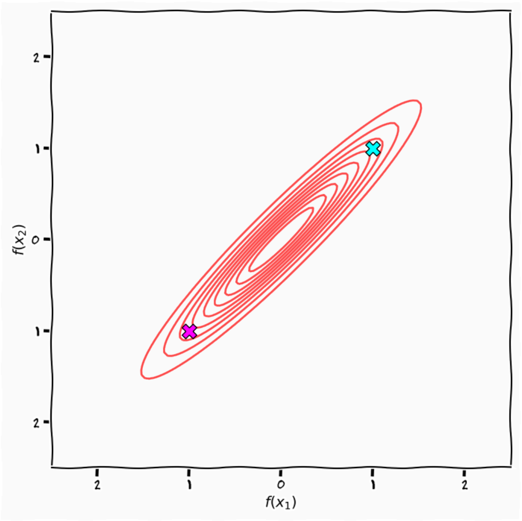
Composite Gaussian Processes

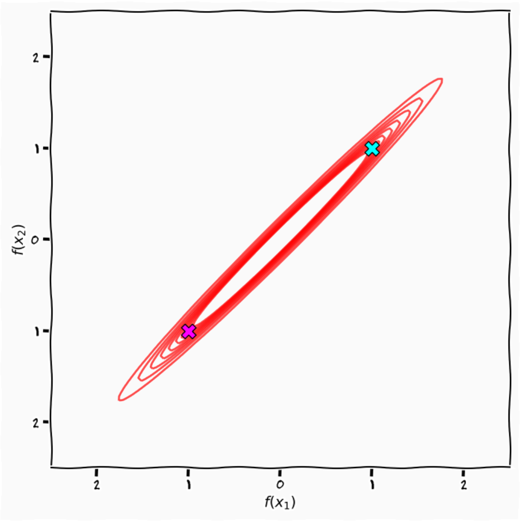


Composite Gaussian Processes

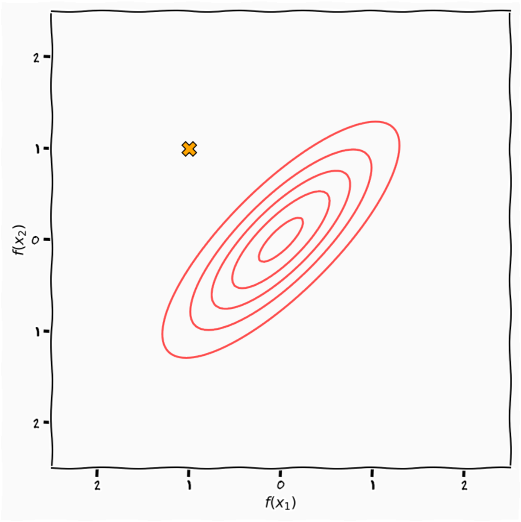


Composite Gaussian Processes

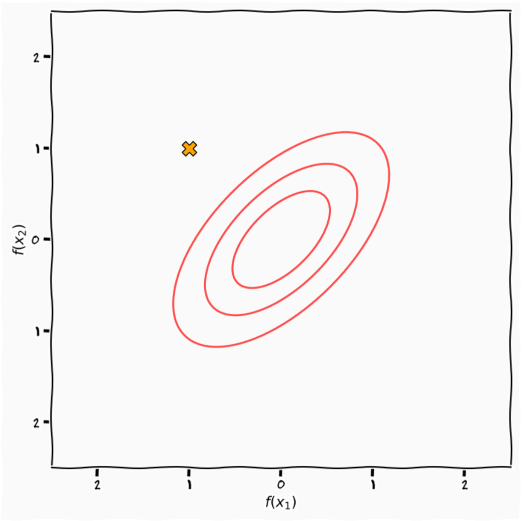



f(x_2)
2
1
0
1
2
2
1
0
1
2
f(x_1)

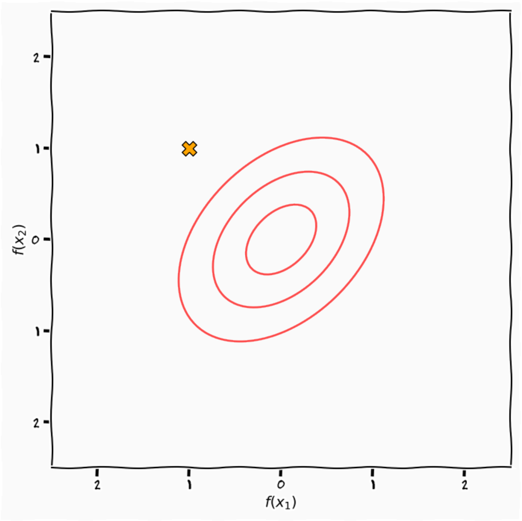
Composite Gaussian Processes



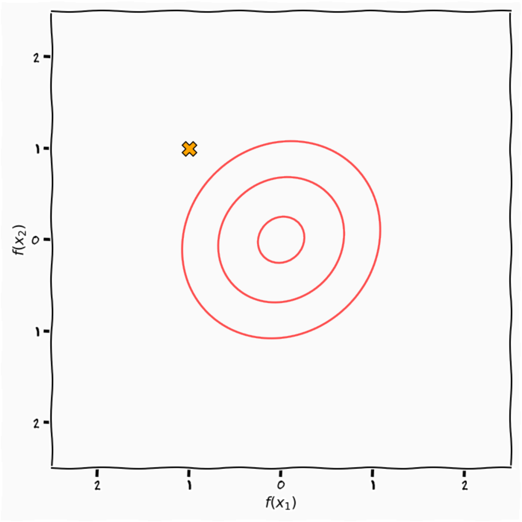
Composite Gaussian Processes



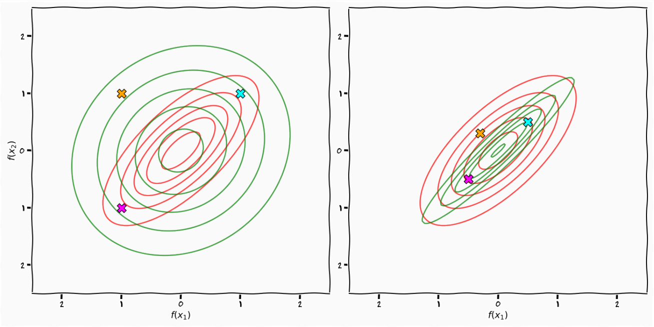
Composite Gaussian Processes



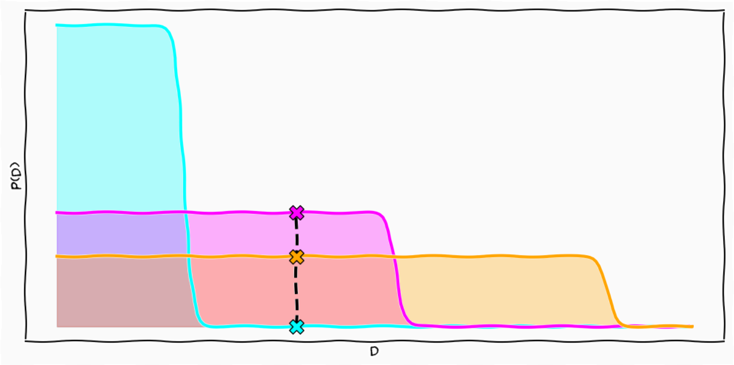
Composite Gaussian Processes



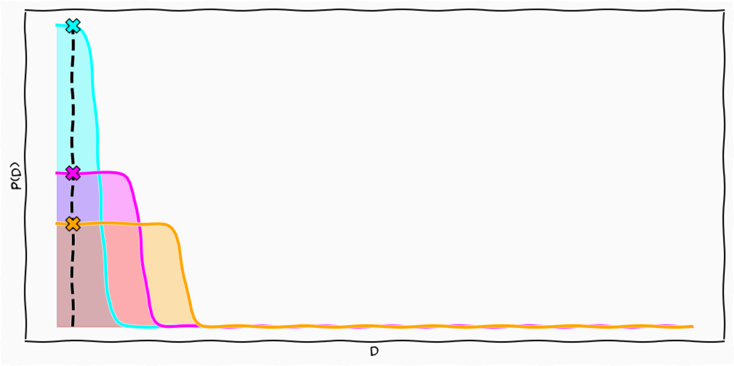
Composite Gaussian Processes



Learning



Learning



Is this useful?

"A theory that explains everything, explains nothing"
– Karl Popper The Logic of Scientific Discovery

Approximate Inference

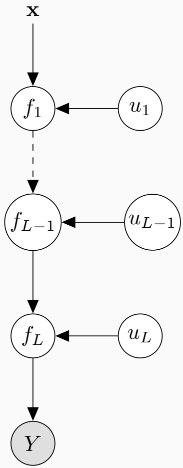
- Sufficient statistics

$$q(\mathbf{F})q(\mathbf{U})q(\mathbf{X}) = p(\mathbf{F}|\mathbf{Y},\mathbf{U},\mathbf{X},\mathbf{Z})q(\mathbf{U})q(\mathbf{X})$$

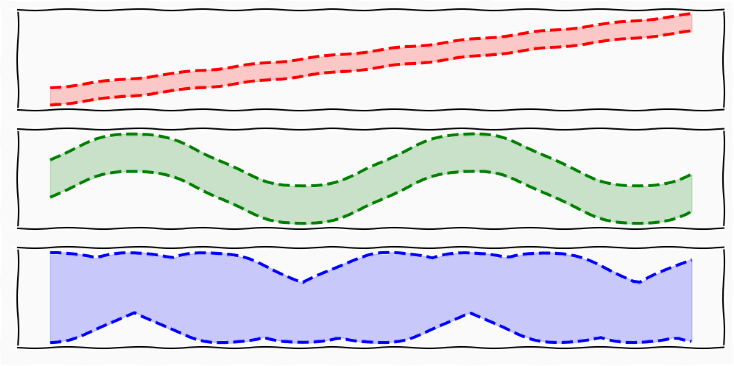
$$= p(\mathbf{F}|\mathbf{U},\mathbf{X},\mathbf{Z})q(\mathbf{U})q(\mathbf{X})$$

- Mean-Field

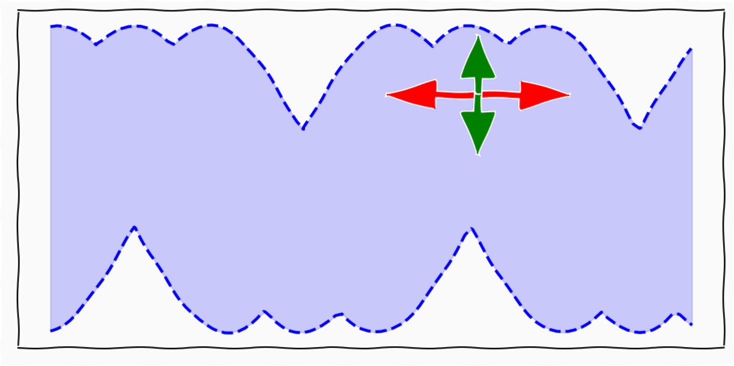
$$q(\mathbf{U}) = \prod_{i}^{L} q(\mathbf{U}_i)$$



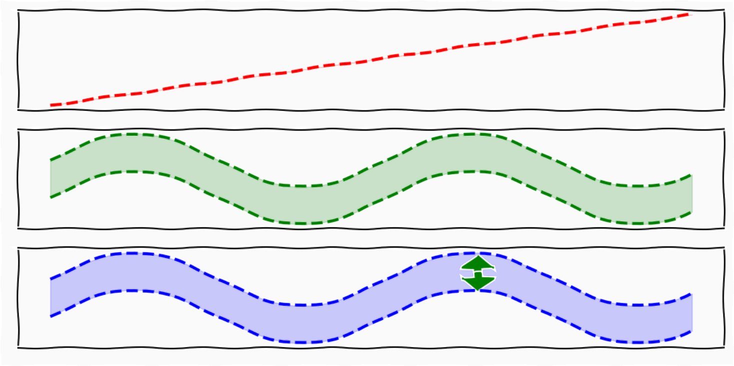
Composite Uncertainty



Composite Uncertainty



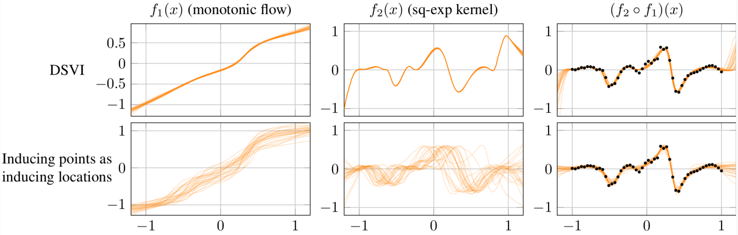
The Effect of Independence



Motivation

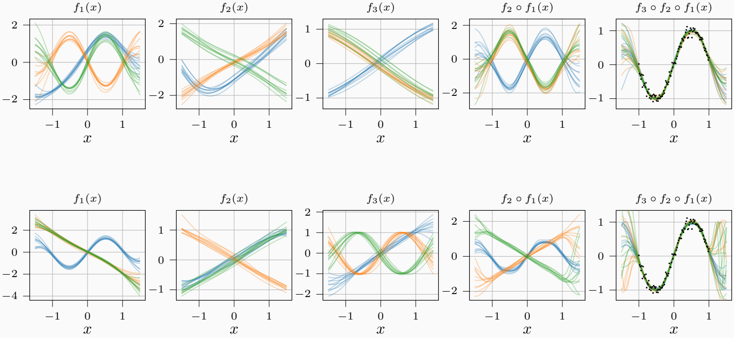
Often when people talk about the limitations of variational inference, they really mean the limitations of mean-field.
– Danilo J. Rezende (on twitter)

Results[6]



[6]Ustyuzhaninov et al., 2020.

"Multi-Modality"[7]



[7] Ustyuzhaninov et al., 2020.

Software

Code

[]python Initialise a 4-layer model consisting of NN layers and GP layers model = Sequential ([Dense (...) , Convolution (...) , GPLayer (...) , GPLayer (...)]) model.compile(loss=LikelihoodLoss(Gaussian ()), optimizer ="Adam") Fitting callbacks = [ReduceLROnPlateau (), TensorBoard (), ModelCheckpoint ()] model.fit(X, Y, callbacks =callbacks) Evaluating model.predict(X)

GPFlux Dutordoir et al., 2021b

Summary

Summary

- Unsupervised learning[8] is very hard.

[8] I would argue that there is no such thing

Summary

- Unsupervised learning[8] is very hard.
    - *Its actually not, its really really easy.*

[8] I would argue that there is no such thing

Summary

- Unsupervised learning[8] is very hard.
  - *Its actually not, its really really easy.*
- Relevant assumptions needed to learn anything useful

[8] I would argue that there is no such thing

Summary

- Unsupervised learning[8] is very hard.
    - *Its actually not, its really really easy.*
- Relevant assumptions needed to learn anything useful
- Strong assumptions needed to learn anything from "sensible" amounts of data

[8] I would argue that there is no such thing

Summary

- Unsupervised learning[8] is very hard.
    - *Its actually not, its really really easy.*
- Relevant assumptions needed to learn anything useful
- Strong assumptions needed to learn anything from "sensible" amounts of data
- Stochastic processes such as GPs provide strong, interpretative assumptions that aligns well to our intuitions allowing us to make relevant assumptions

[8] I would argue that there is no such thing

Summary II

- Composite functions cannot model more things

Summary II

- Composite functions cannot model more things
- However, they can easily warp the input space to model less things

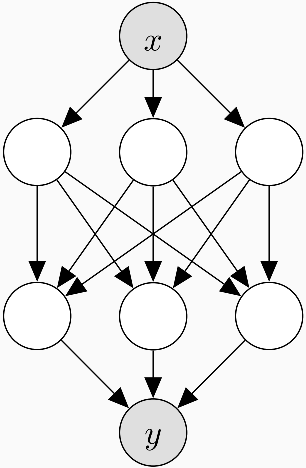
Summary II

- Composite functions cannot model more things
- However, they can easily warp the input space to model less things
- This leads to high requirements on data

"Bayesian Neural Networks"

$$y = f(x, \mathbf{W})$$

$$w \sim \mathcal{N}(0, I)$$



Bayesian Superiority

Thoughts

- Compositions are good parametrisations for learning parameters[9]

[9] Neural Networks (Maybe) Evolved to Make Adam The Best Optimizer

Thoughts

- Compositions are good parametrisations for learning parameters[9]
- Adding probabilities to regularise the learning makes sense

[9] Neural Networks (Maybe) Evolved to Make Adam The Best Optimizer

Thoughts

- Compositions are good parametrisations for learning parameters[9]
- Adding probabilities to regularise the learning makes sense
- But the posterior can only be interpreted in light of the prior

[9] Neural Networks (Maybe) Evolved to Make Adam The Best Optimizer

Thoughts

- Compositions are good parametrisations for learning parameters[9]
- Adding probabilities to regularise the learning makes sense
- But the posterior can only be interpreted in light of the prior
- And uncertainties are composite themselves

[9] Neural Networks (Maybe) Evolved to Make Adam The Best Optimizer

Thoughts

- Can you ever defend a composite model if your knowledge is not composite?

Thoughts

- Can you ever defend a composite model if your knowledge is not composite?
    - $k(f(x), f(x'))$, $k([x, z], [x, z'])$

Thoughts

- Can you ever defend a composite model if your knowledge is not composite?
    - $k(f(x), f(x'))$, $k([x, z], [x, z'])$
- Current "frameworks" doesn't allow for compartmentalisations

Thoughts

- Can you ever defend a composite model if your knowledge is not composite?
  - $k(f(x), f(x'))$, $k([x,z],[x,z'])$
- Current "frameworks" doesn't allow for compartmentalisations
  - what is a composite probability?

Thoughts

- Can you ever defend a composite model if your knowledge is not composite?
  - $k(f(x), f(x'))$, $k([x, z], [x, z'])$
- Current "frameworks" doesn't allow for compartmentalisations
  - what is a composite probability?
  - what is a composite function prior?

eof

Reference

References

Candela, Joaquin Quiñonero and Carl Edward Rasmussen (2005). "A Unifying View of Sparse Approximate Gaussian Process Regression". In: *Journal of Machine Learning Research* 6, pp. 1939–1959.

Damianou, Andreas C (Feb. 2015). "Deep Gaussian Processes and Variational Propagation of Uncertainty". PhD thesis. University of Sheffield.

- Damianou, Andreas C and Neil D Lawrence (2013). "Deep Gaussian Processes". In: *International Conference on Airtificial Inteligence and Statistical Learning*, pp. 207–215.
- Dutordoir, Vincent et al. (2021a). "Deep Neural Networks As Point Estimates for Deep Gaussian Processes". In: *CoRR*.
- Dutordoir, Vincent et al. (2021b). "Gpflux: a Library for Deep Gaussian Processes". In: *CoRR*.
- Hensman, James, N Fusi, and Neil D Lawrence (2013). "Gaussian Processes for Big Data". In: *Uncertainty in Artificial Intelligence*.

Lázaro-Gredilla, Miguel and Aníbal Figueiras-Vidal (2009). "Inter-domain Gaussian Processes for Sparse Inference using Inducing Features". In: *Advances in Neural Information Processing Systems*. Ed. by Y. Bengio et al. Vol. 22. Curran Associates, Inc.

Neal, Radford M (1996). *Bayesian Learning for Neural Networks*. Vol. 8. New York: Springer-Verlag.

Popper, K.R. (1959). *The Logic of Scientific Discovery*. ISSR library. Routledge.

Titsias, Michalis and Neil D Lawrence (2010). "Bayesian Gaussian Process Latent Variable Model". In: *International Conference on Airtificial Inteligence and Statistical Learning*, pp. 844–851.

Ustyuzhaninov, Ivan et al. (2020). "Compositional uncertainty in deep Gaussian processes". In: *Proceedings of the Thirty-Sixth Conference on Uncertainty in Artificial Intelligence, UAI 2020, virtual online, August 3-6, 2020*. Ed. by Ryan P. Adams and Vibhav Gogate. Vol. 124. Proceedings of Machine Learning Research. AUAI Press, pp. 480–489.