Imperial College London

PROWLER.io™
the principled AI company

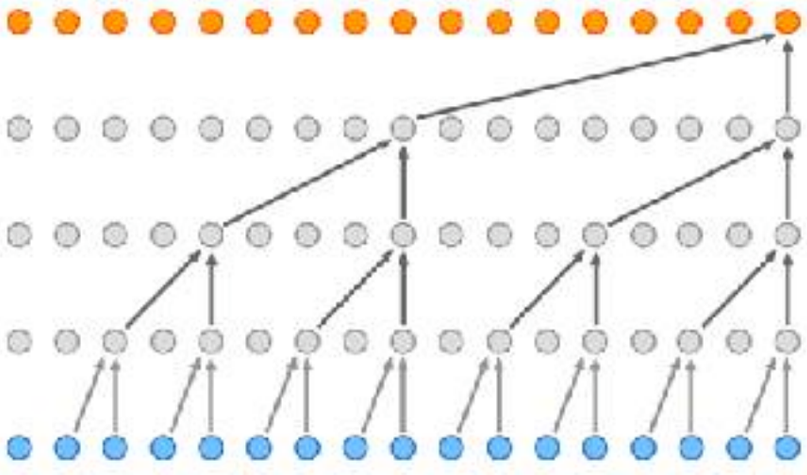
Deep Gaussian Processes with Importance-Weighted Variational Inference (and Latent Variables)

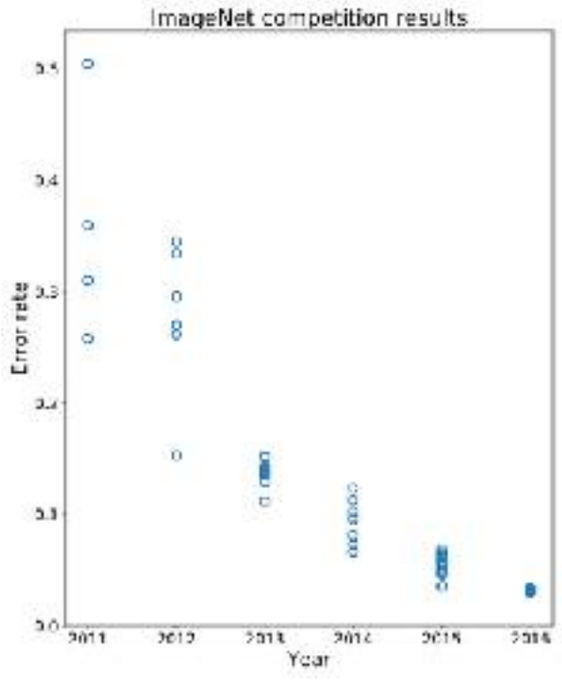
Hugh Salimbeni
10th October 2018

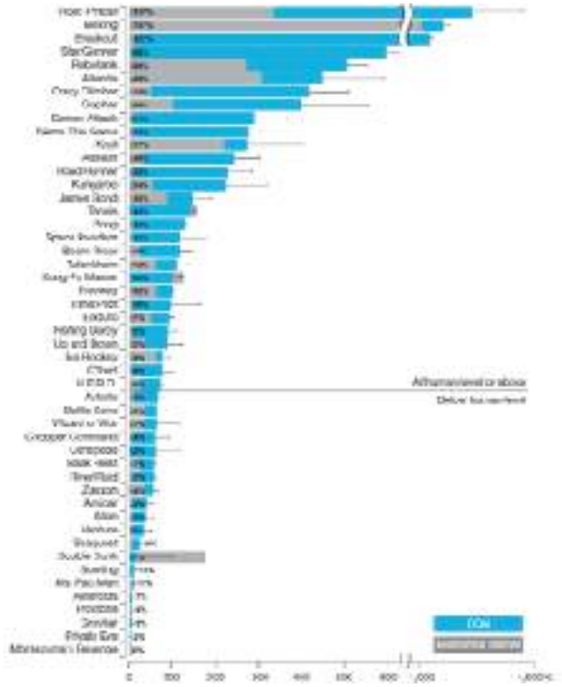
Outline:

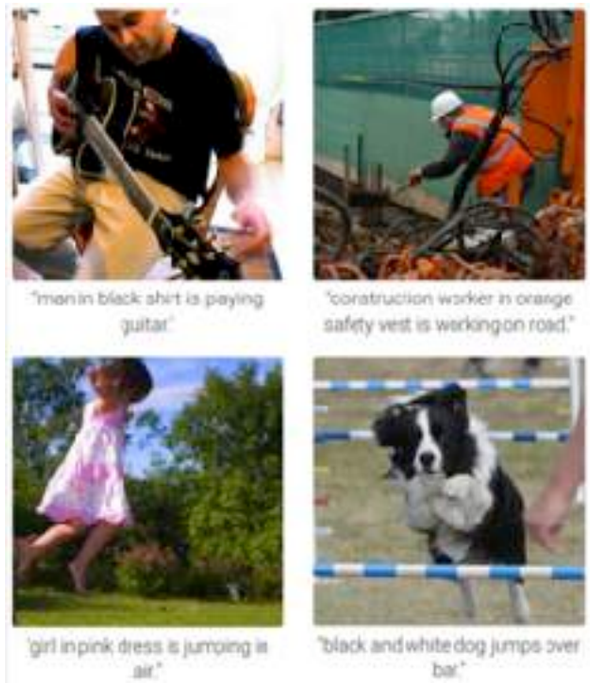
- Why we want a Deep GP (2 reasons)
- The Deep GP model
- Inference in the Deep GP
- Why latent variables are important
- Inference over latent variables
- Results

Why we might want a DGP (1)



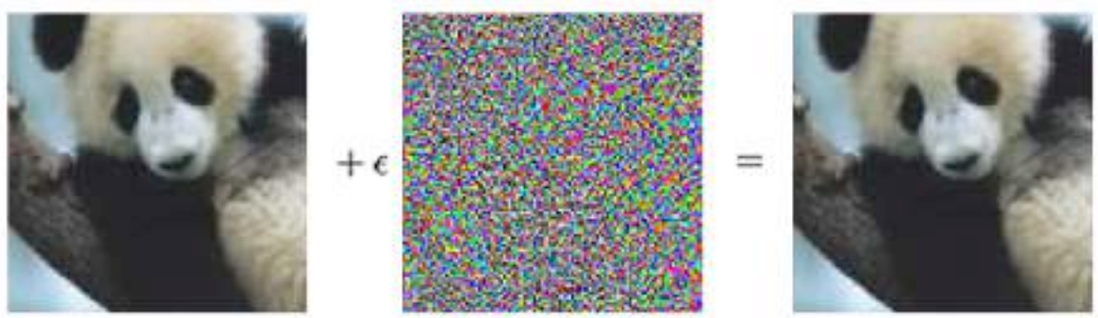






What's wrong with deep learning?

- Bag of tricks (often) necessary
- No (calibrated) uncertainty
- Black-box (sometimes) not acceptable
- Weakness to adversarial attacks



Ambition:

- Win at deep learning tasks using fully Bayesian methods
- Get accurate uncertainty, adversarial robustness, principled model training and model selection etc

Not quite there yet…

Fundamental trade-off?

David Silver [Deep Learning Indaba 2018]:

- “Trust in experience as the sole source of knowledge”
- “Learning from experience always wins in the long run”

He is (probably) right

But asymptotics aren’t (always) what we care about

A personal view:

Two extremal options:

- The success of deep learning is evidence that we have infinite data
- The success of deep learning is attributable to a magically effective inductive bias

The truth is likely to lie somewhere between

- To do well in modern deep learning tasks, Bayesians need the think about both

Why aren’t we there yet?

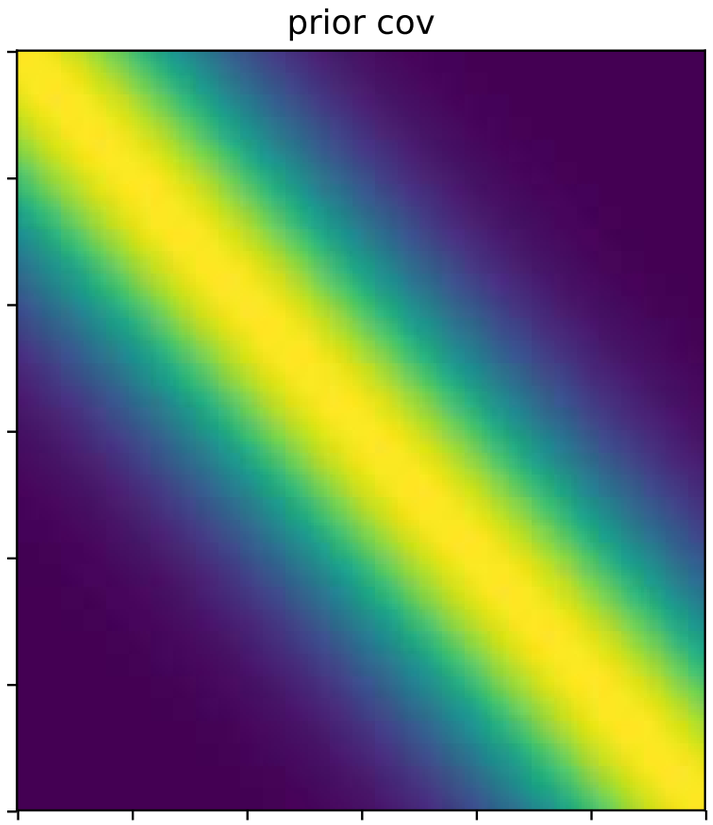
- Not sufficiently *scalable
- Insufficient understanding of probabilities in high dimensions

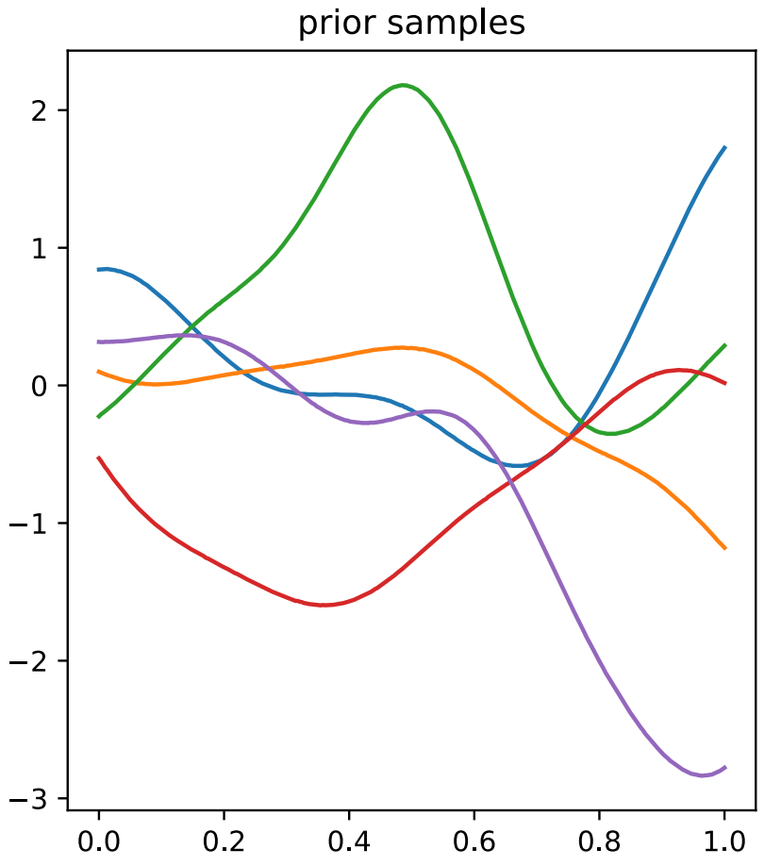
$$* \text{ scalability} = \frac{d(\text{performance})}{d(\text{resource})}$$

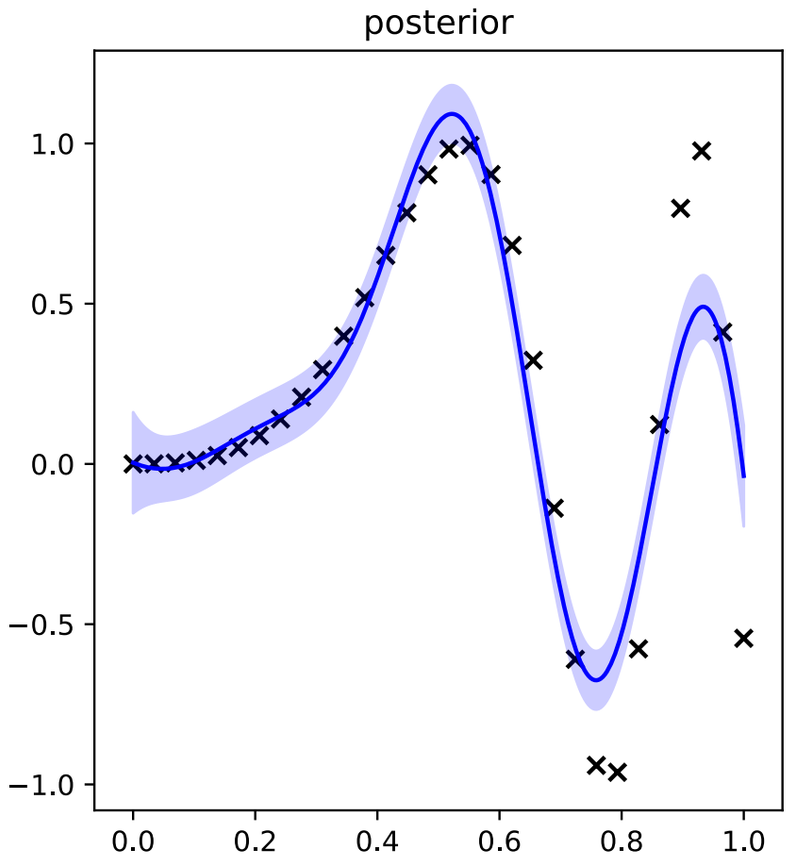
Outline:

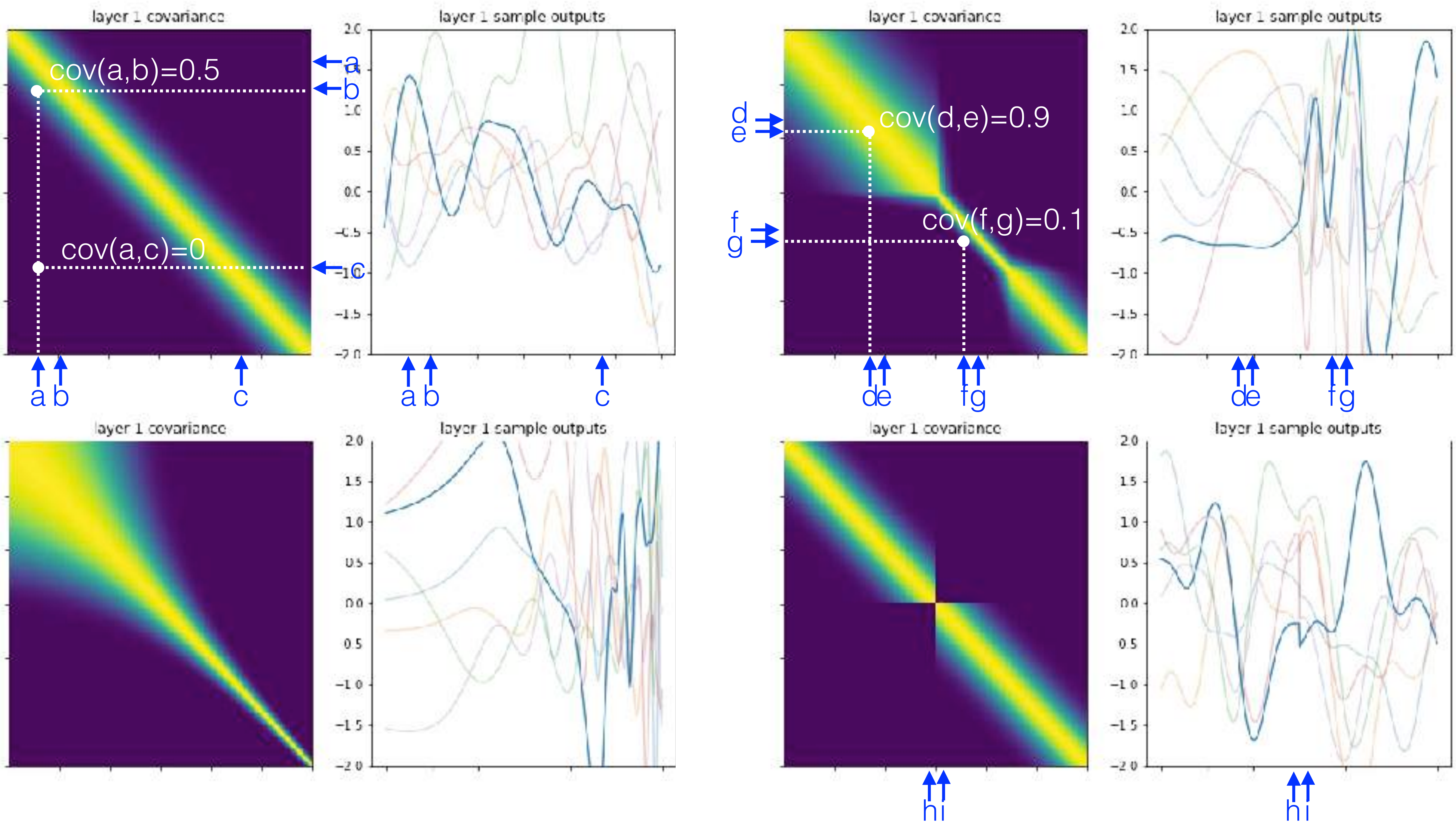
- Why we want a Deep GP (2 reasons)
- The Deep GP model
- Inference in the Deep GP
- Why latent variables are important
- Inference over latent variables
- Results

Why we might want a DGP (2)

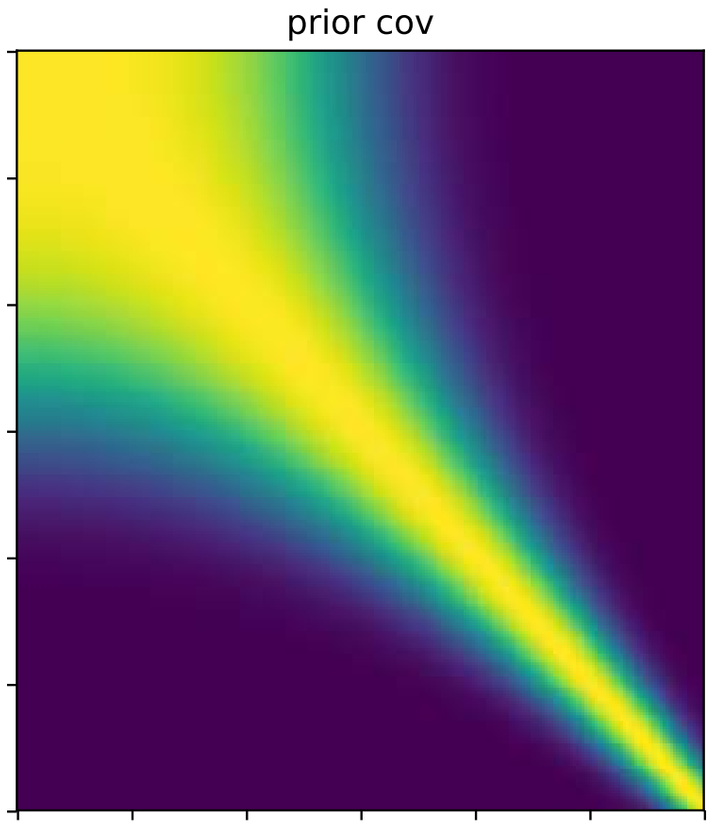


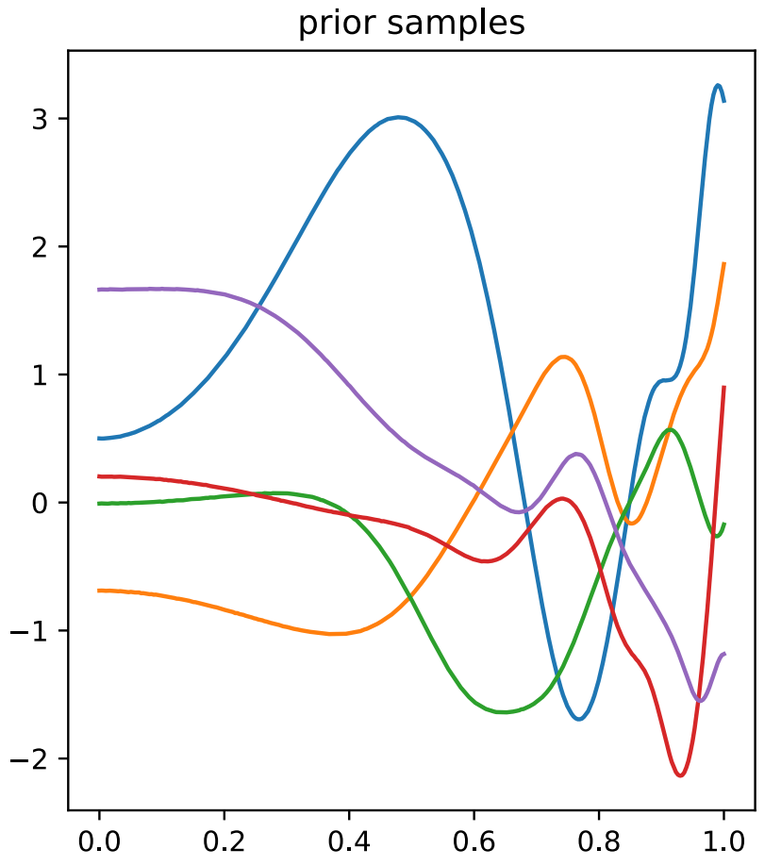


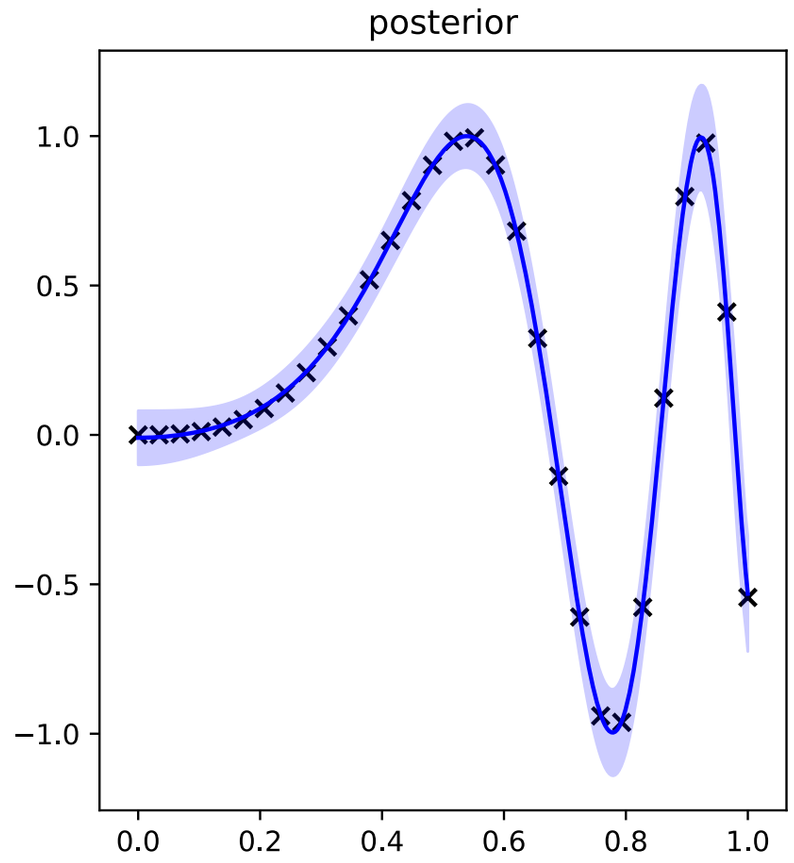



layer 1 covariance
layer 1 sample outputs
cov(a,b)=0.5
cov(a,c)=0
a
b
c
cov(d,e)=0.9
cov(f,g)=0.1
d
e
f
g
h
i

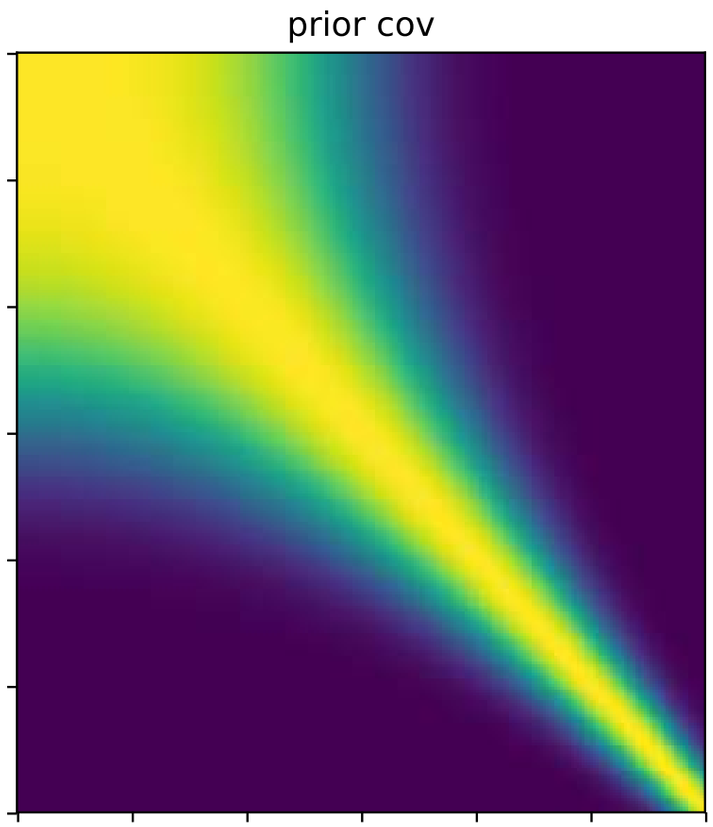
Good model

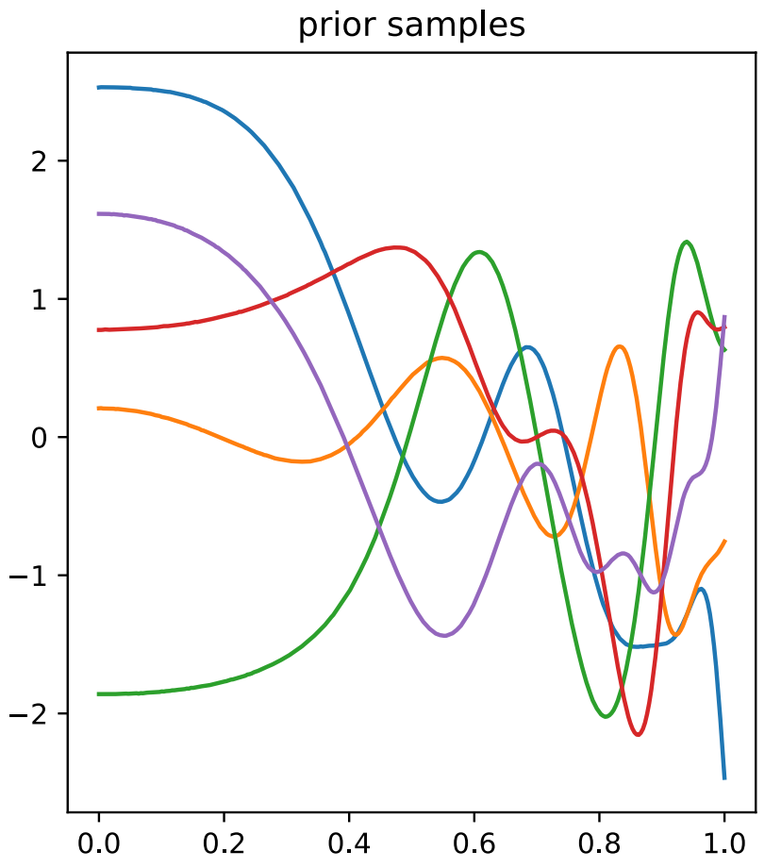

prior cov

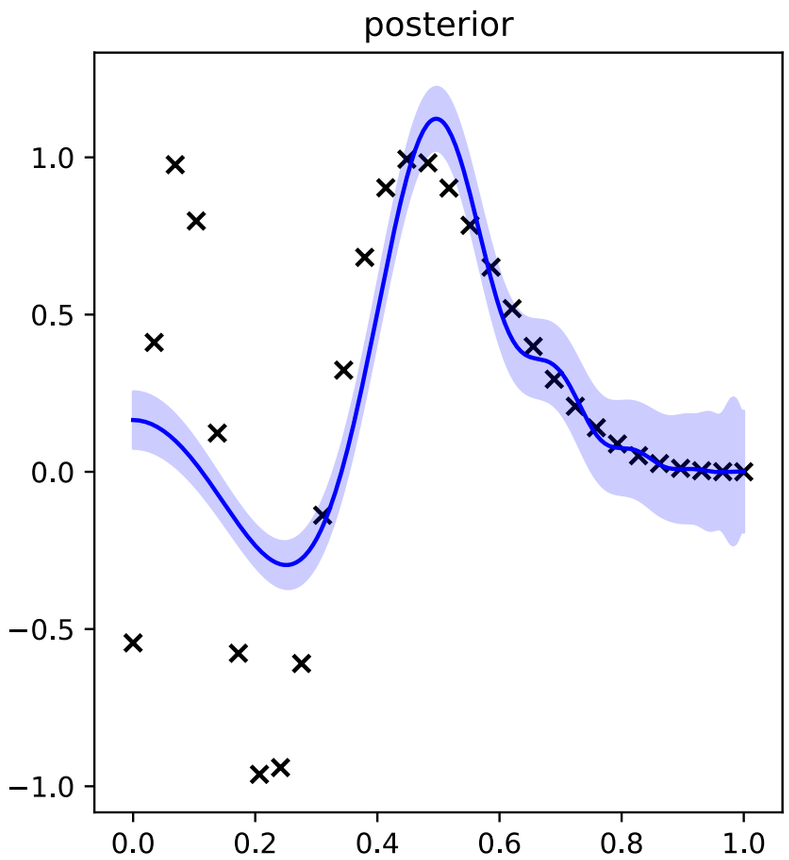

prior samples
3
2
1
0
−1
−2
0.0
0.2
0.4
0.6
0.8
1.0


posterior
1.0
0.5
0.0
−0.5
−1.0
0.0
0.2
0.4
0.6
0.8
1.0

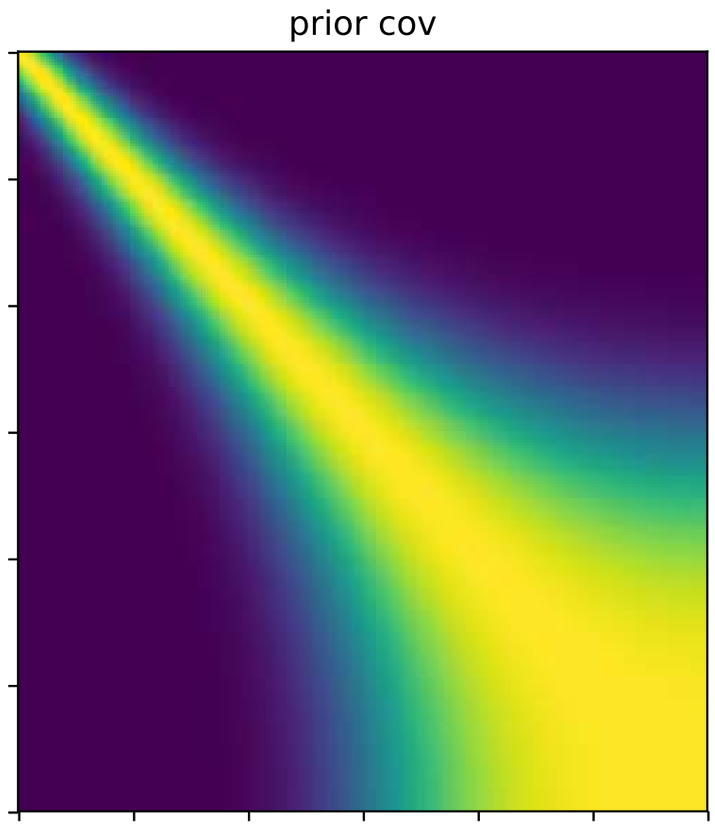
Bad model

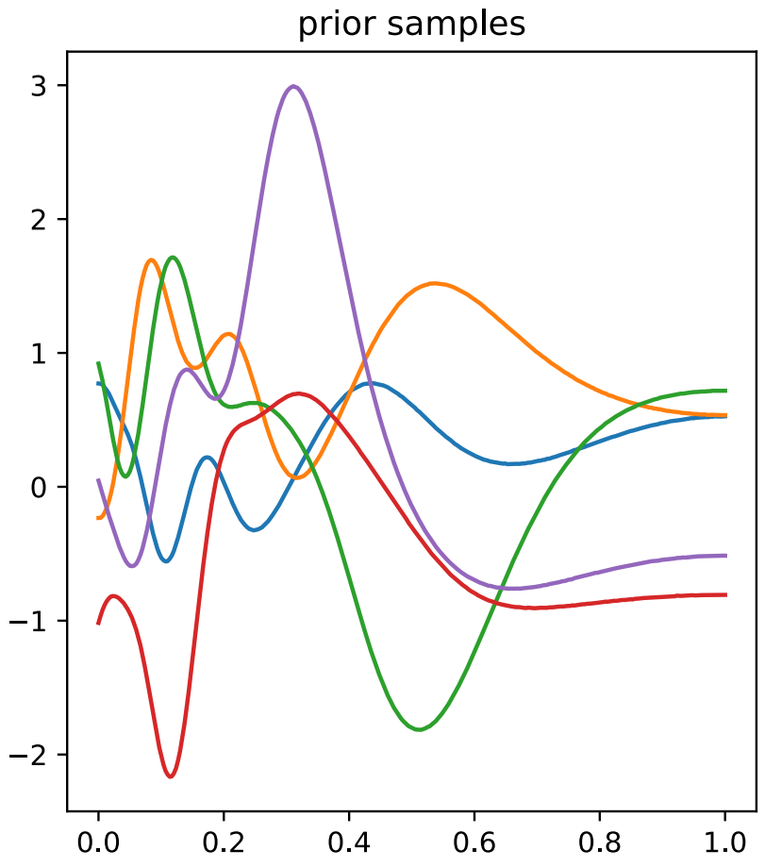

prior cov

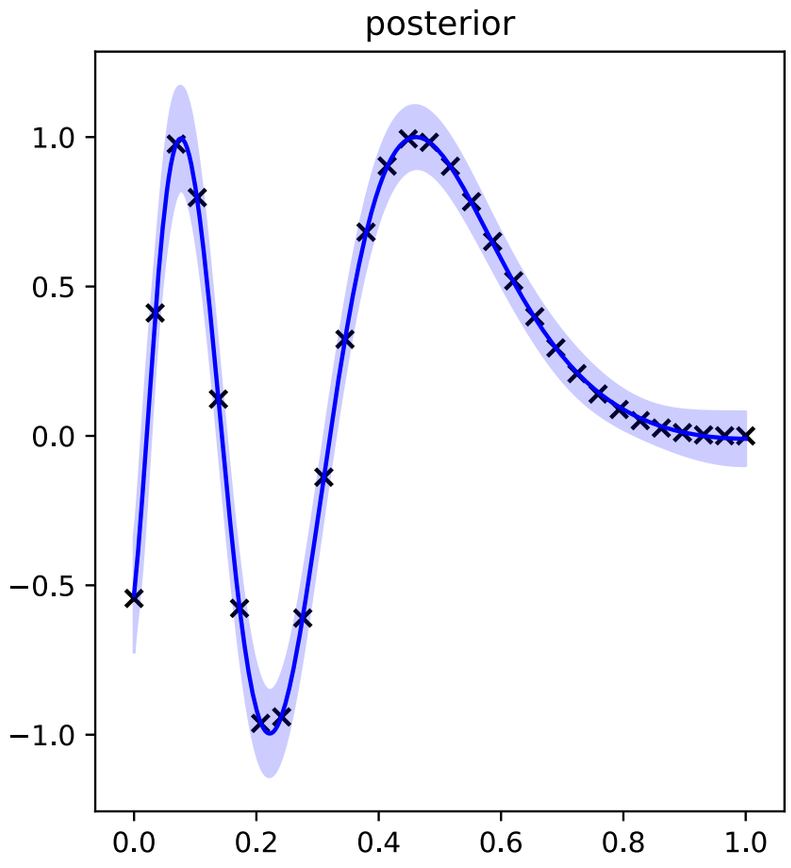

prior samples
2
1
0
−1
−2
0.0
0.2
0.4
0.6
0.8
1.0


posterior
1.0
0.5
0.0
−0.5
−1.0
0.0
0.2
0.4
0.6
0.8
1.0

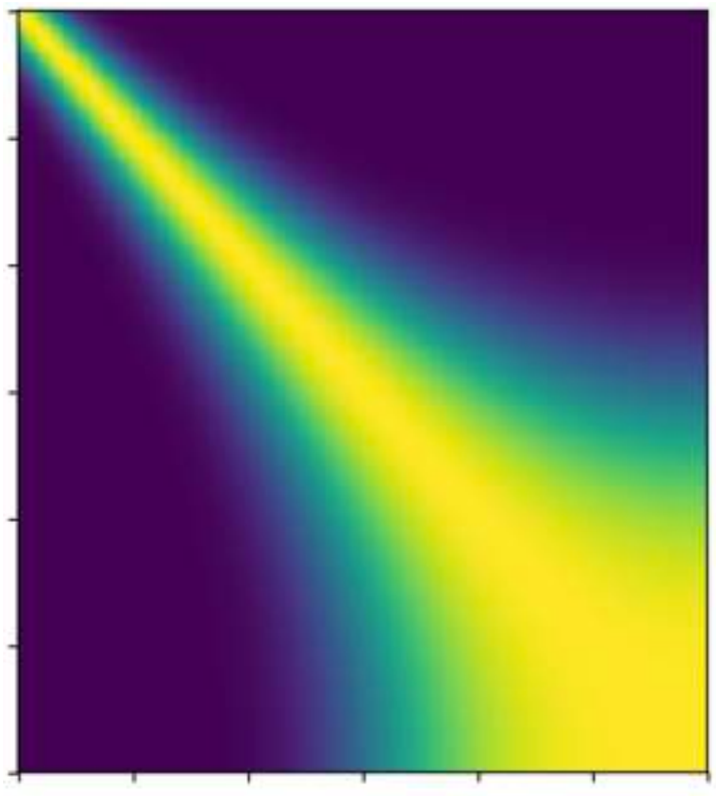
Good model


prior cov

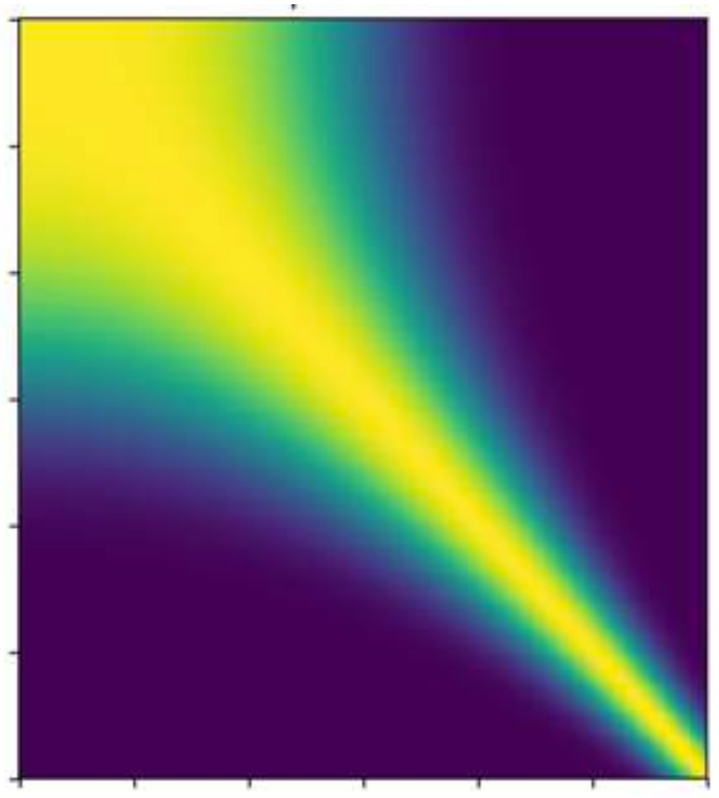

prior samples
3
2
1
0
−1
−2
0.0
0.2
0.4
0.6
0.8
1.0


posterior
1.0
0.5
0.0
−0.5
−1.0
0.0
0.2
0.4
0.6
0.8
1.0

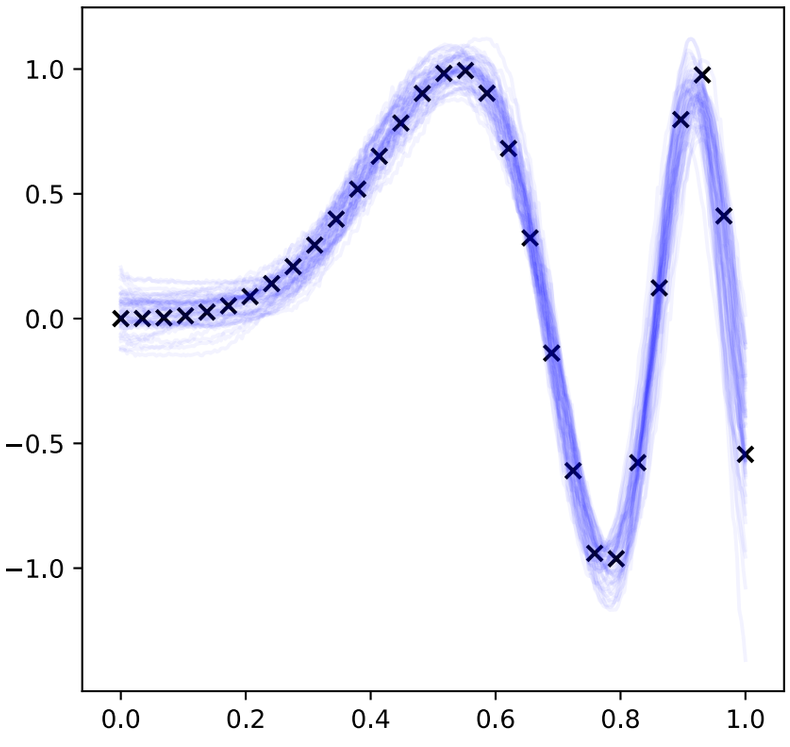
What if we want to consider both?

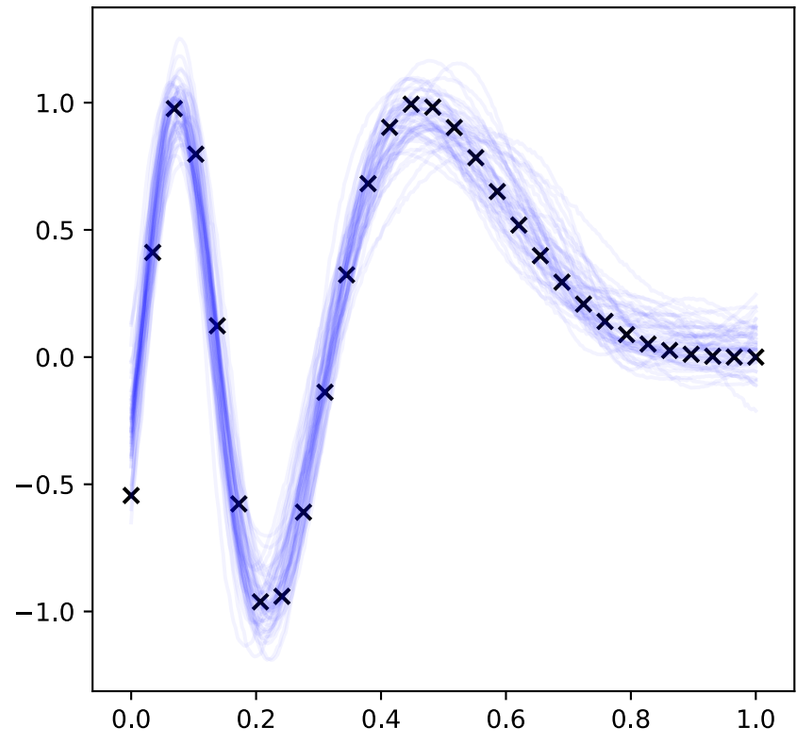


???



A Deep GP posterior





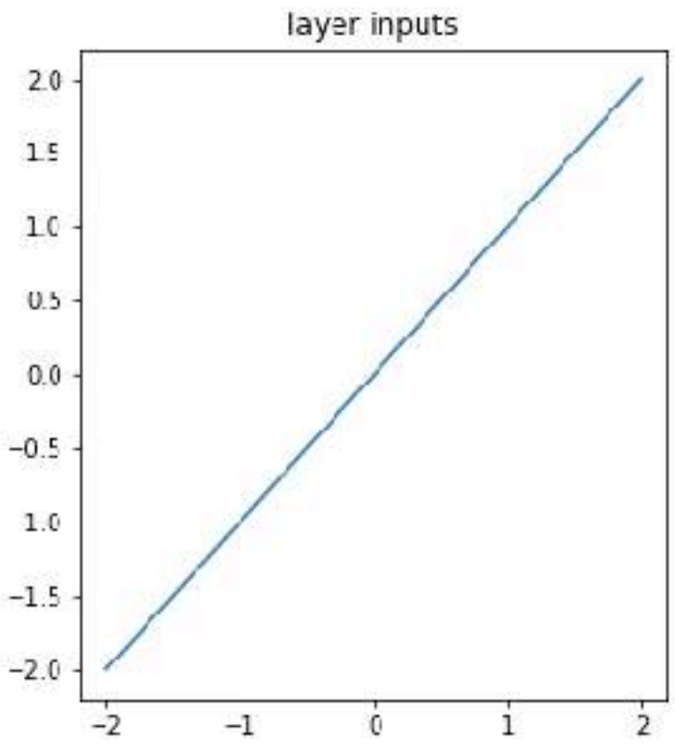
Ambition:

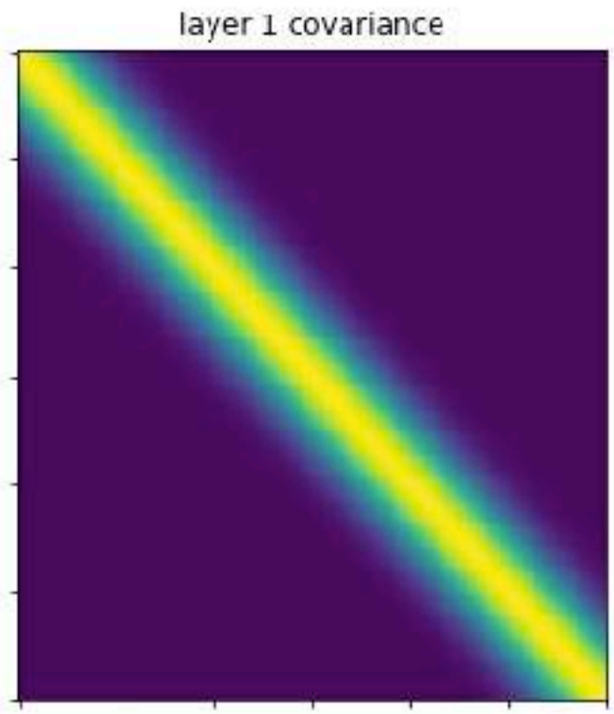
- Form covariances hierarchically
- Get ‘GP-like’ behaviour, but allow more flexibility in the prior

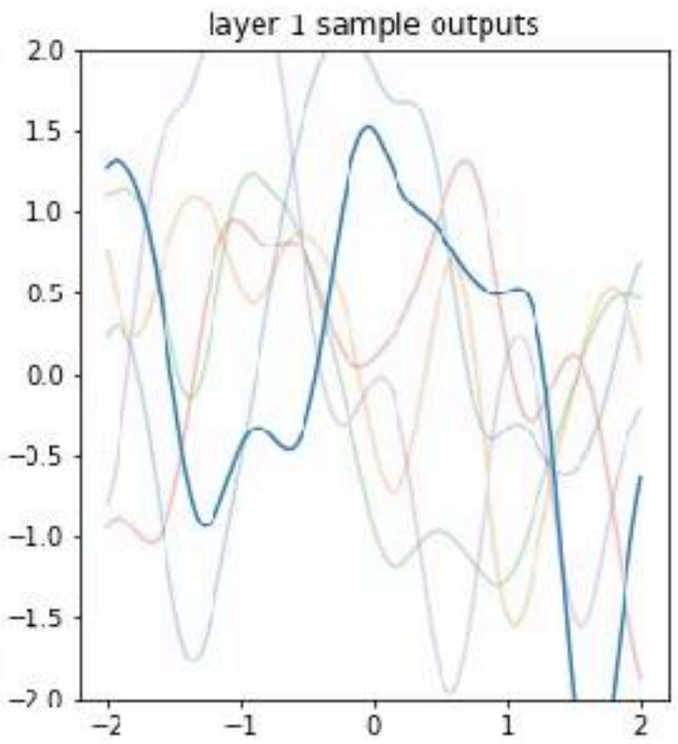
Outline:

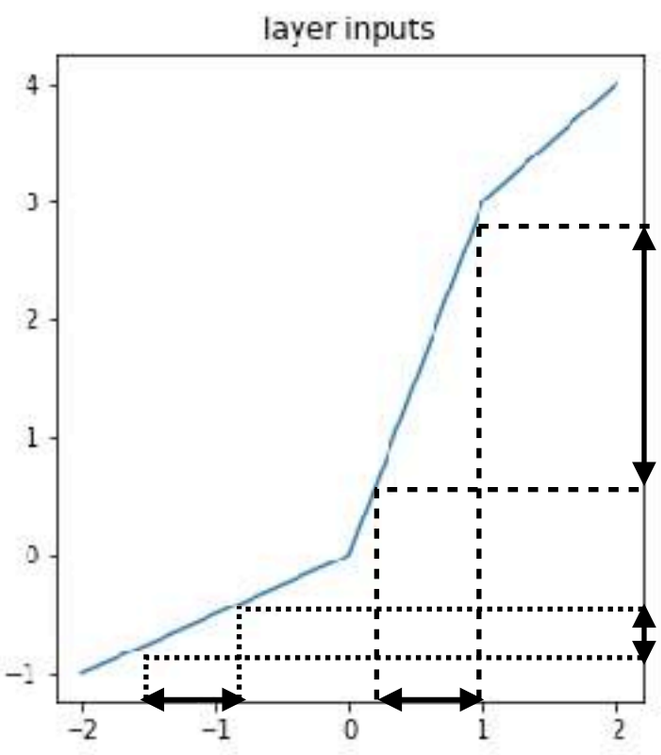
- Why we want a Deep GP (2 reasons)
- The Deep GP model
- Inference in the Deep GP
- Why latent variables are important
- Inference over latent variables
- Results

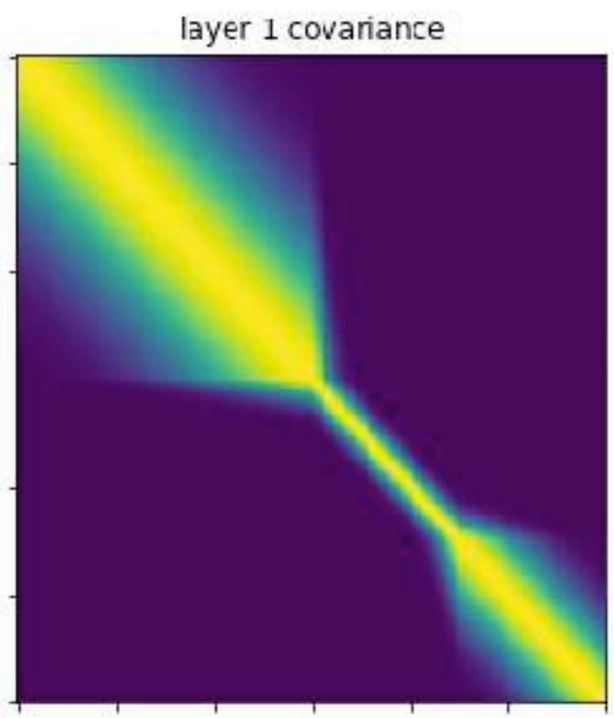
Key idea: form complex covariances with stationary kernels and input warping functions

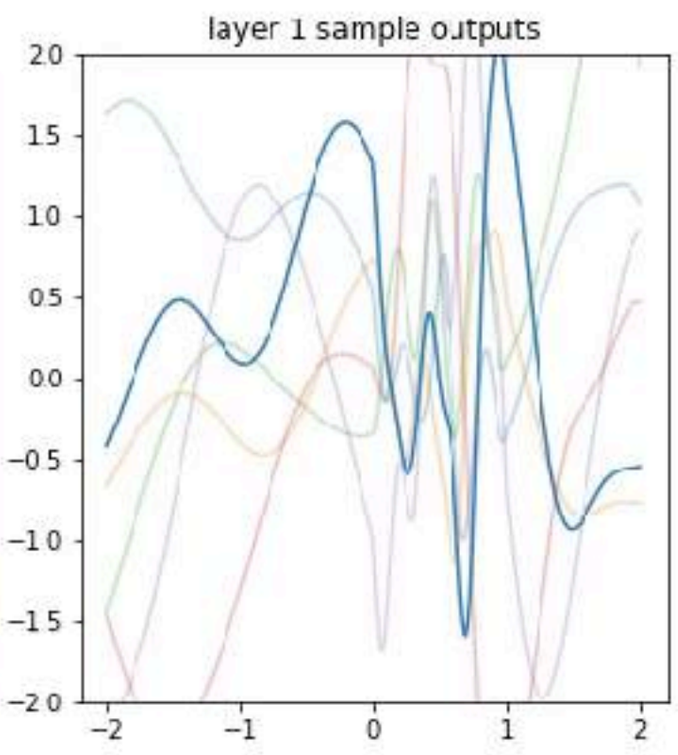




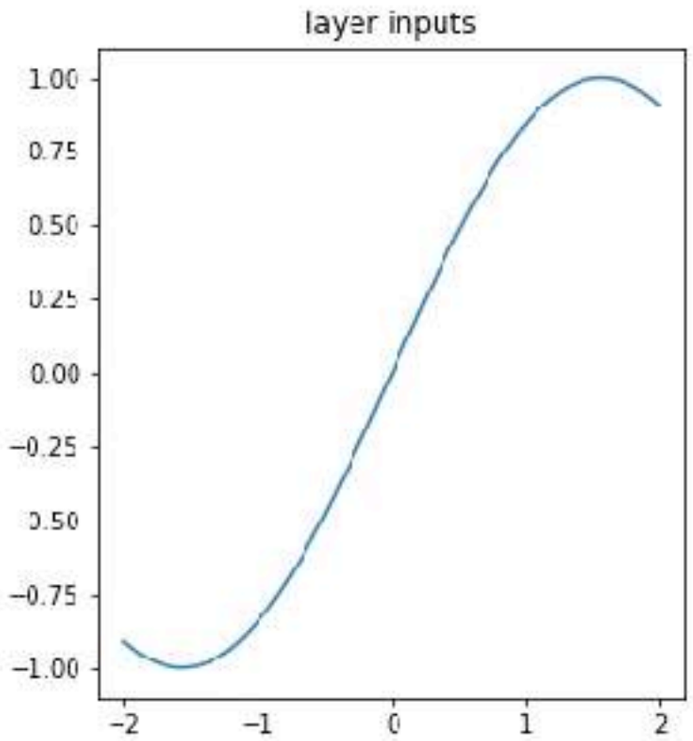


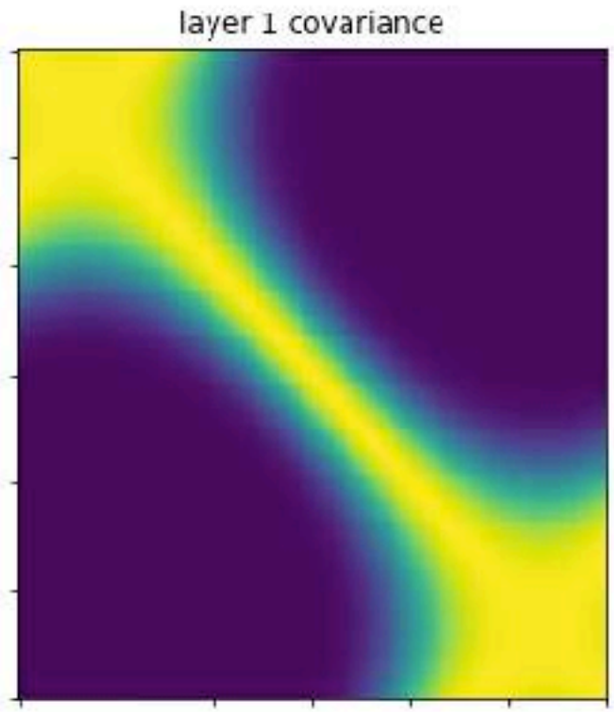


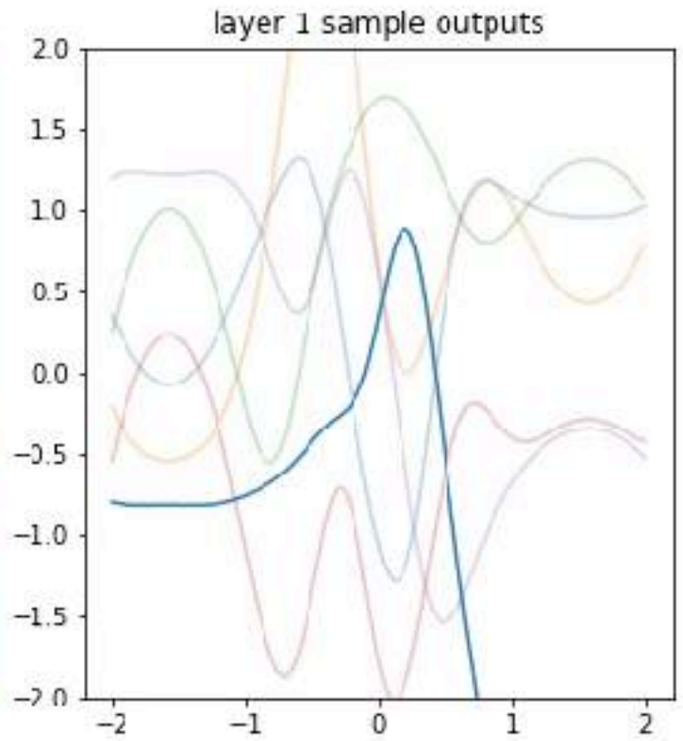


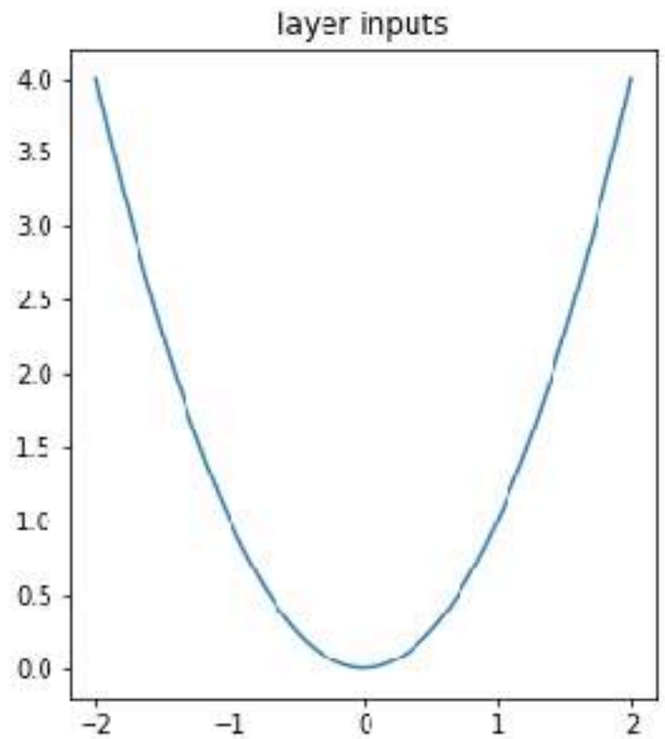


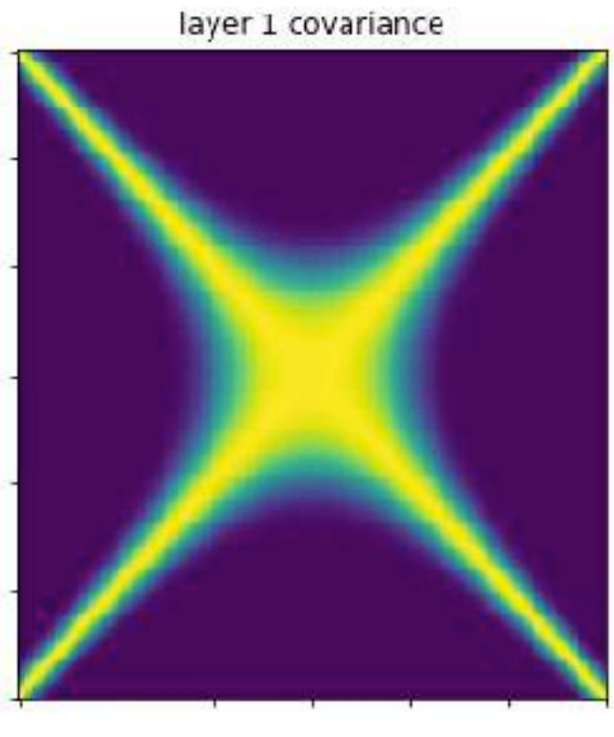
Examples

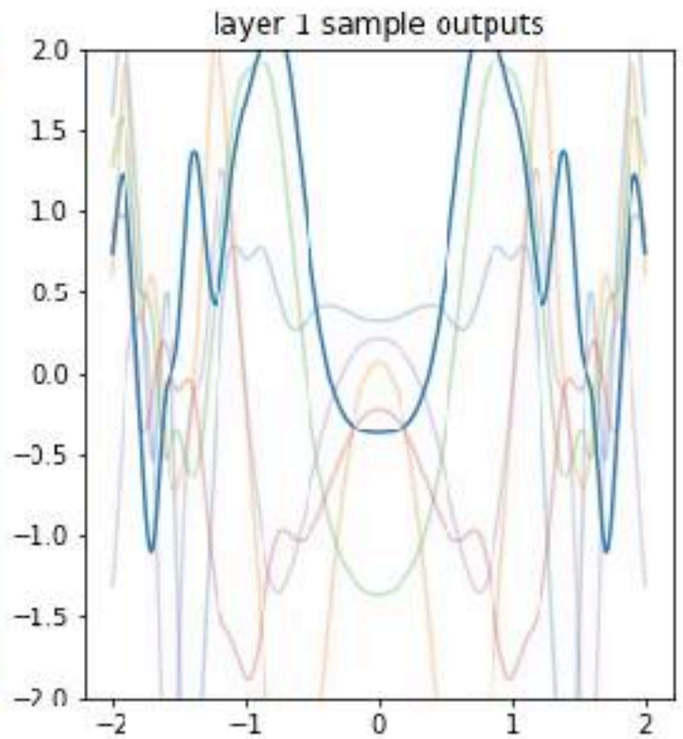

layer inputs
1.00
0.75
0.50
0.25
0.00
−0.25
0.50
−0.75
−1.00
−2
−1
0
1
2


layer 1 covariance


layer 1 sample outputs
2.0
1.5
1.0
0.5
0.0
−0.5
−1.0
−1.5
−2.0
−2
−1
0
1
2


layer inputs
4.0
3.5
3.0
2.5
2.0
1.5
1.0
0.5
0.0
−2
−1
0
1
2


layer 1 covariance


layer 1 sample outputs
2.0
1.5
1.0
0.5
0.0
−0.5
−1.0
−1.5
−2.0
−2
−1
0
1
2

To build a Deep GP:

$$y_n \sim \mathcal{N}(f(g(x_n)), \sigma^2)$$

$$f \sim \mathcal{GP}(m_1, k_1)$$

To build a Deep GP:

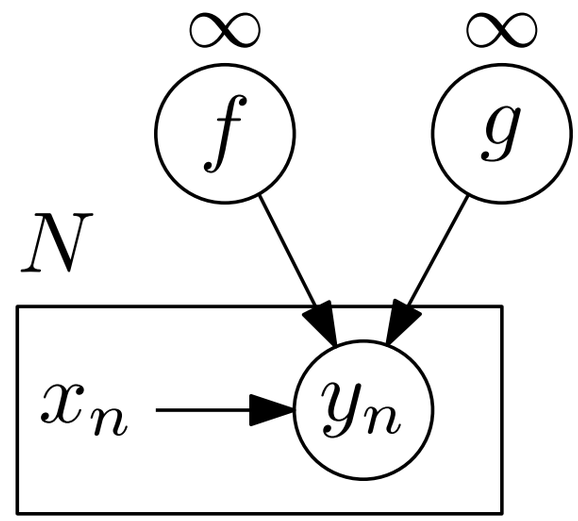
$$y_n \sim \mathcal{N}(f(g(x_n)), \sigma^2)$$

$$f \sim \mathcal{GP}(m_1, k_1)$$

$$g \sim \mathcal{GP}(m_2, k_2)$$

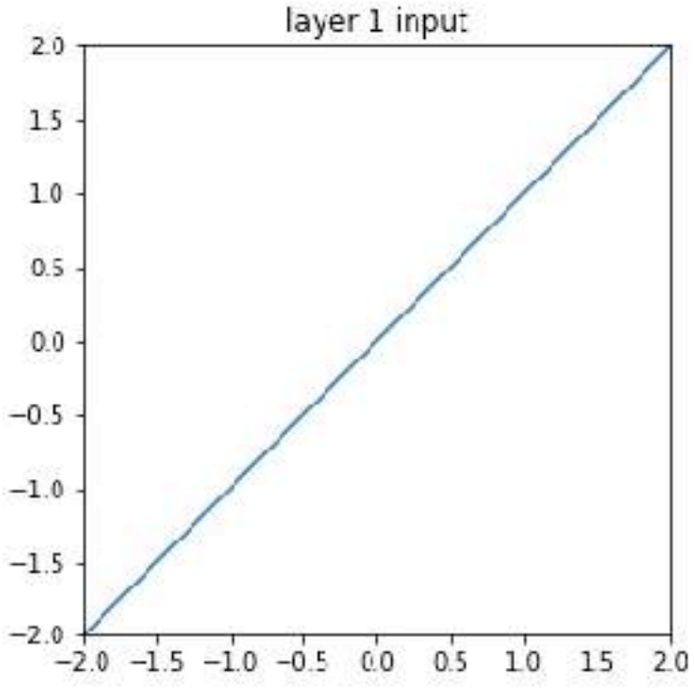
Model

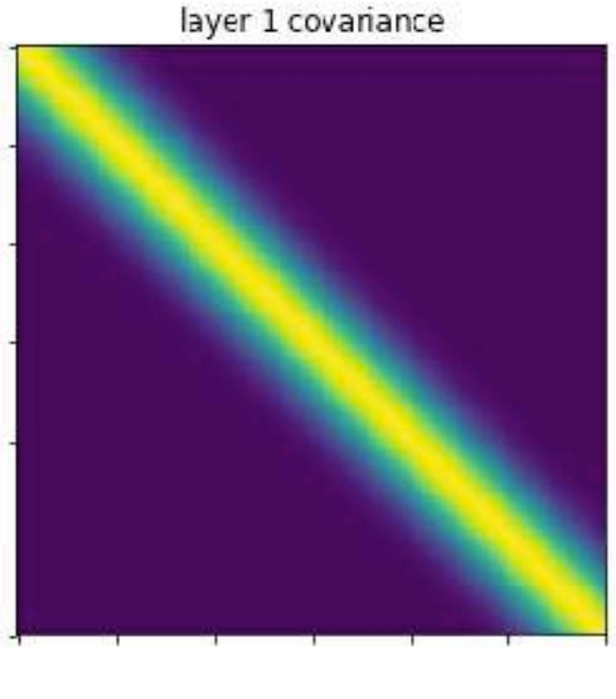
$$y_n \sim \mathcal{N}(f(g(x_n)), \sigma^2)$$
$$f \sim \mathcal{GP}(m_1, k_1)$$
$$g \sim \mathcal{GP}(m_2, k_2)$$

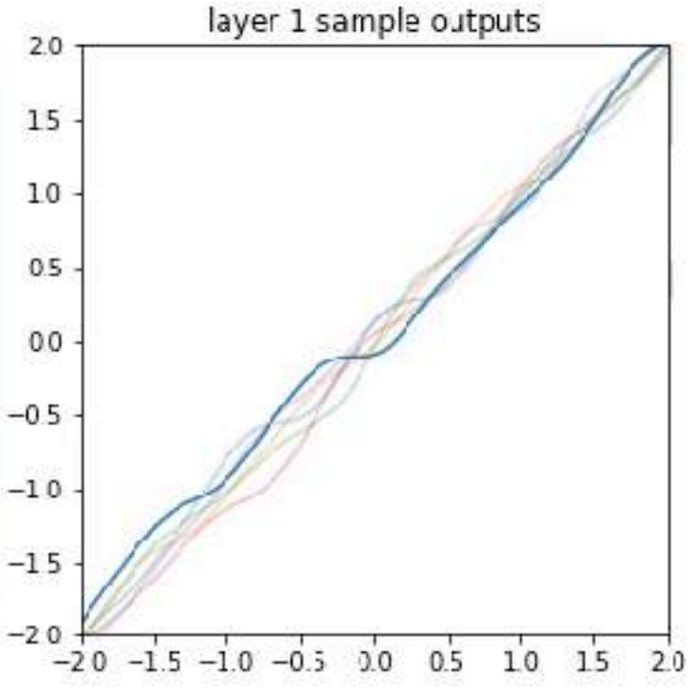


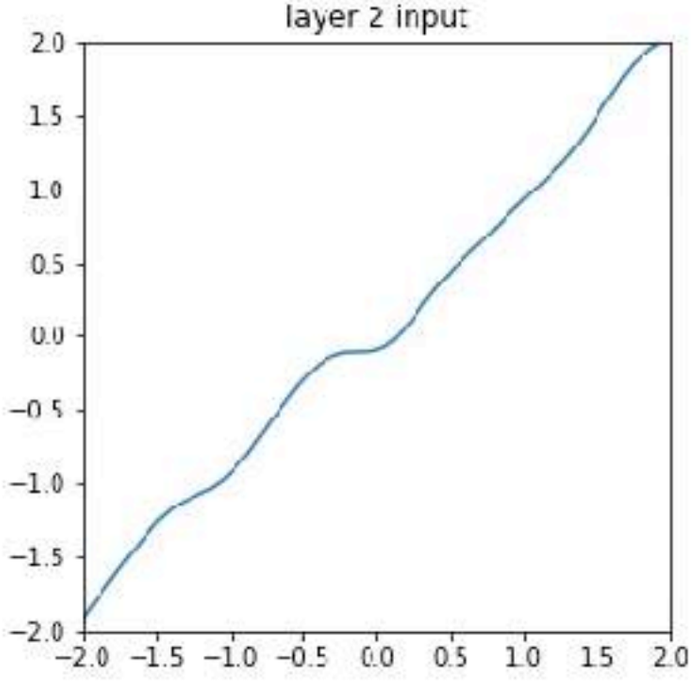
$$m_1(x) = x$$
$$m_2(x) = 0$$

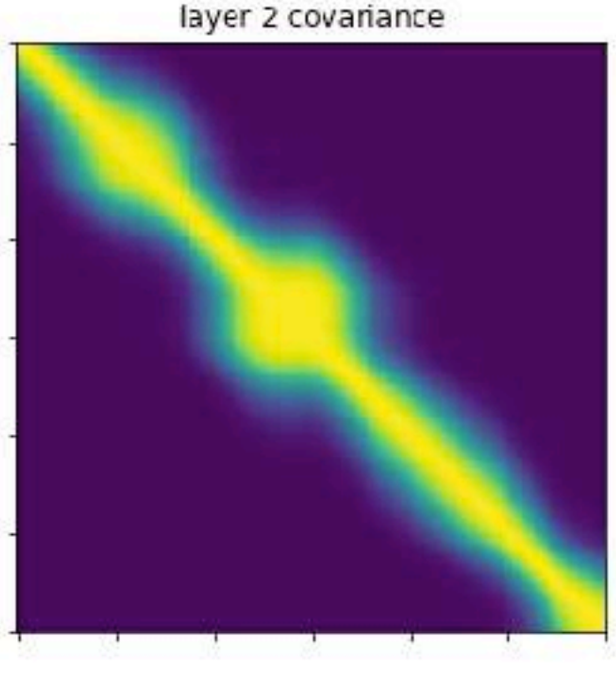
k_1, k_2 stationary RBF kernels

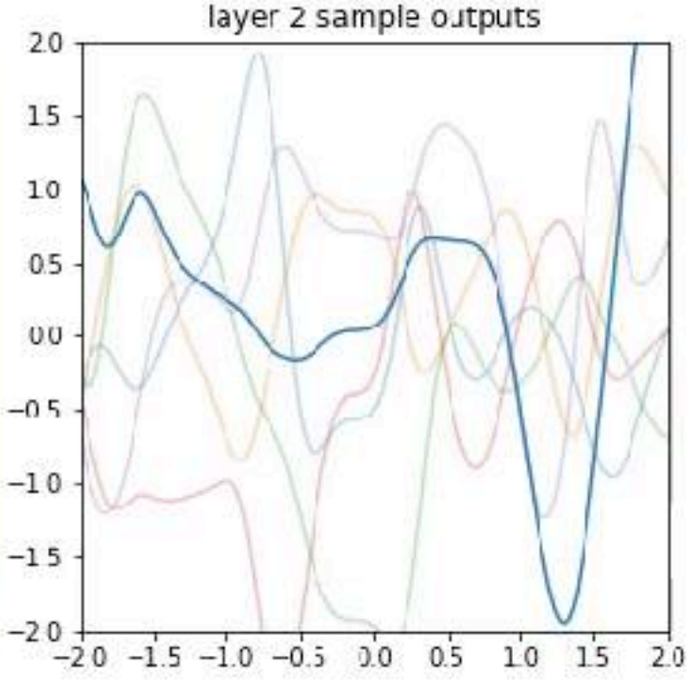

layer 1 input
2.0
1.5
1.0
0.5
0.0
−0.5
−1.0
−1.5
−2.0
−2.0 −1.5 −1.0 −0.5 0.0 0.5 1.0 1.5 2.0

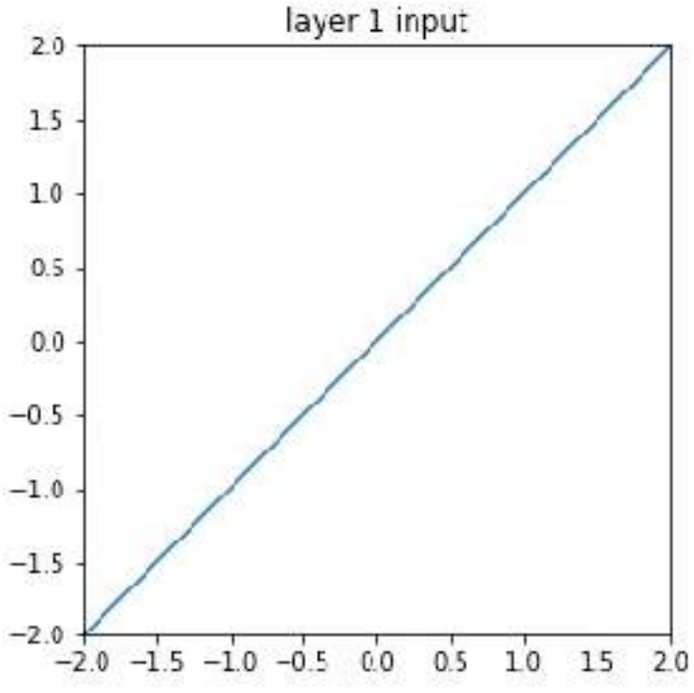

layer 1 covariance

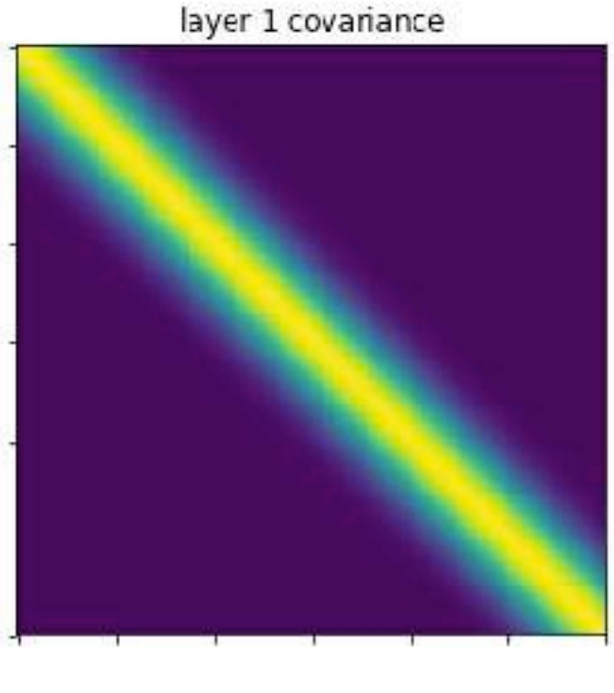

layer 1 sample outputs
2.0
1.5
1.0
0.5
0.0
−0.5
−1.0
−1.5
−2.0
−2.0 −1.5 −1.0 −0.5 0.0 0.5 1.0 1.5 2.0

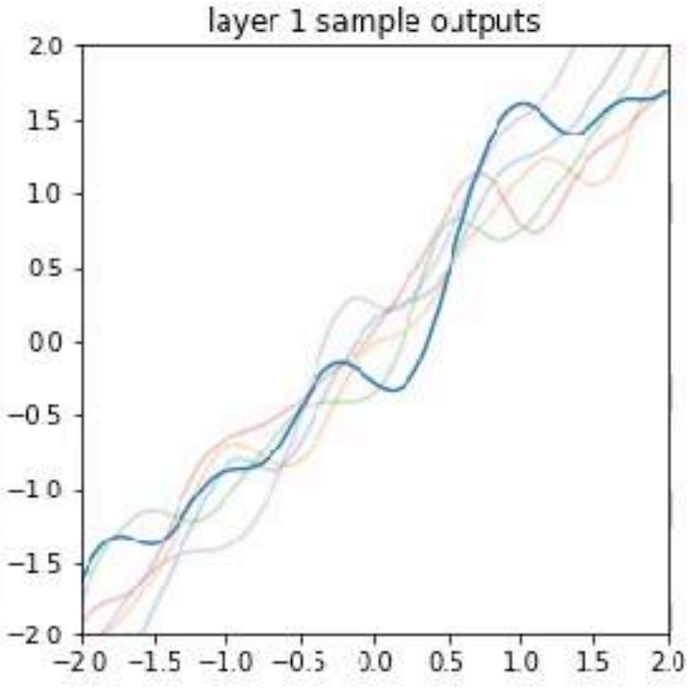

layer 2 input
2.0
1.5
1.0
0.5
0.0
−0.5
−1.0
−1.5
−2.0
−2.0 −1.5 −1.0 −0.5 0.0 0.5 1.0 1.5 2.0

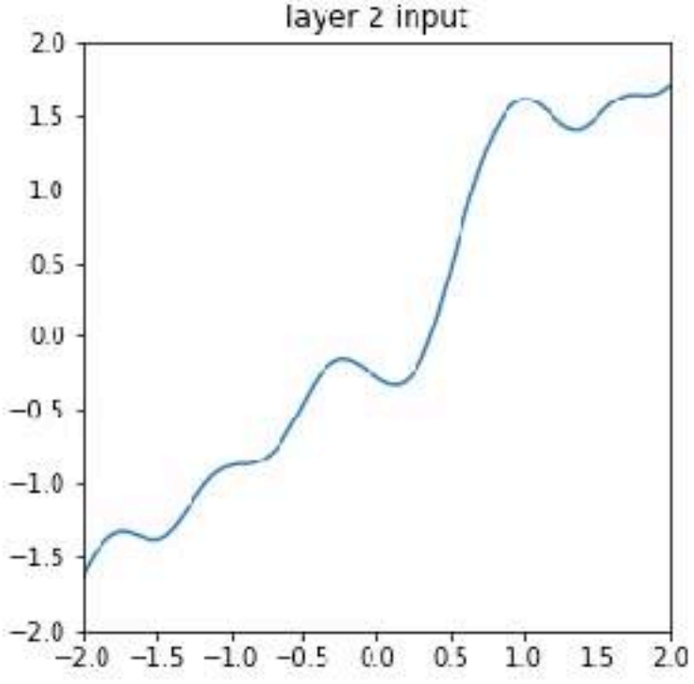

layer 2 covariance

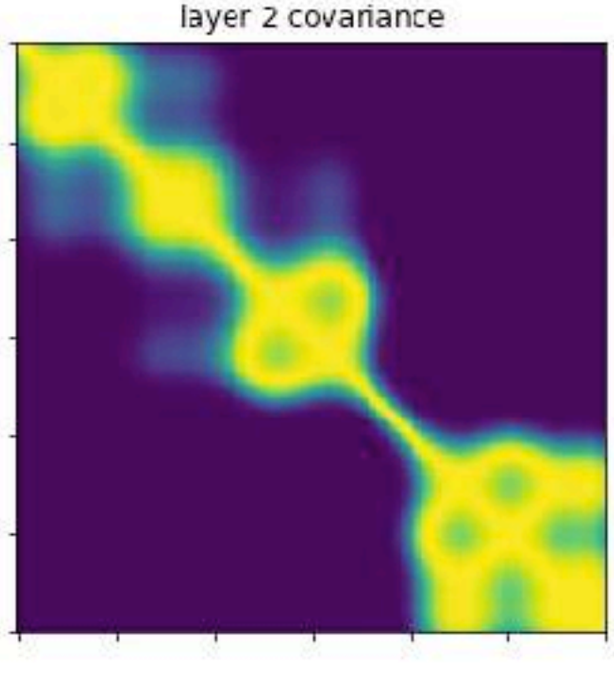

layer 2 sample outputs
2.0
1.5
1.0
0.5
0.0
−0.5
−1.0
−1.5
−2.0
−2.0 −1.5 −1.0 −0.5 0.0 0.5 1.0 1.5 2.0

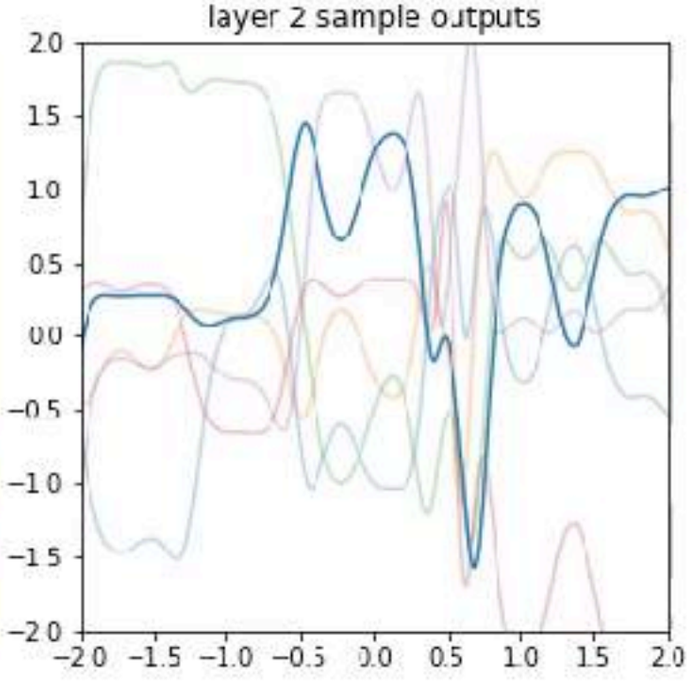

layer 1 input
2.0
1.5
1.0
0.5
0.0
−0.5
−1.0
−1.5
−2.0
−2.0 −1.5 −1.0 −0.5 0.0 0.5 1.0 1.5 2.0

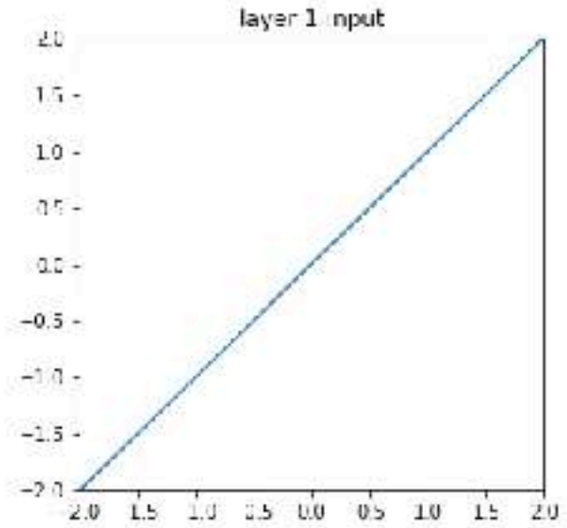

layer 1 covariance

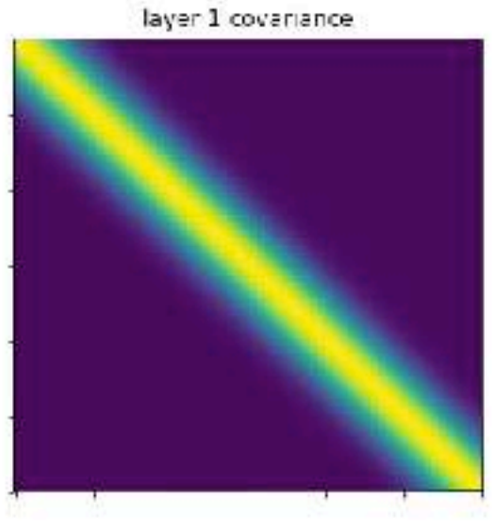

layer 1 sample outputs
2.0
1.5
1.0
0.5
0.0
−0.5
−1.0
−1.5
−2.0
−2.0 −1.5 −1.0 −0.5 0.0 0.5 1.0 1.5 2.0

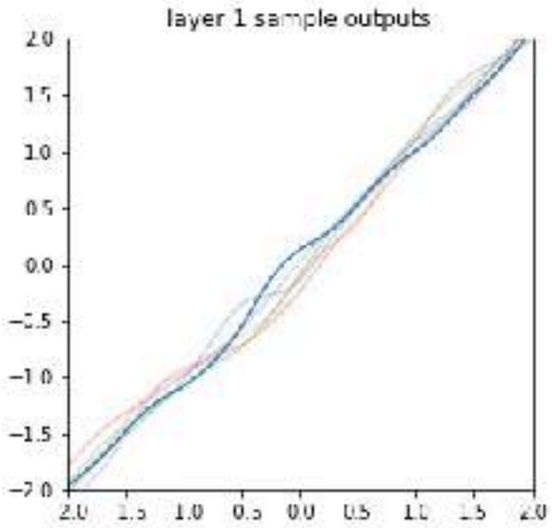

layer 2 input
2.0
1.5
1.0
0.5
0.0
−0.5
−1.0
−1.5
−2.0
−2.0 −1.5 −1.0 −0.5 0.0 0.5 1.0 1.5 2.0

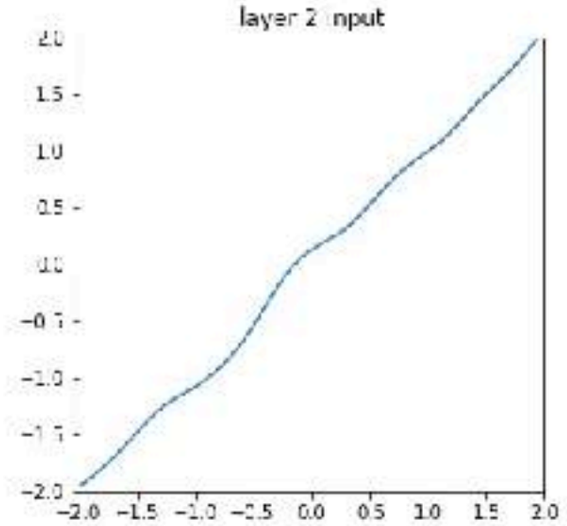

layer 2 covariance

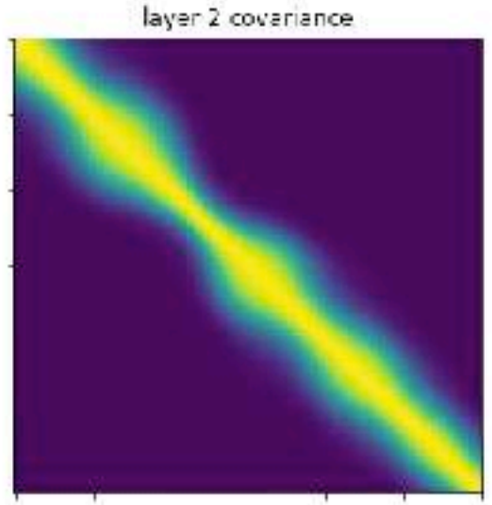

layer 2 sample outputs
2.0
1.5
1.0
0.5
0.0
−0.5
−1.0
−1.5
−2.0
−2.0 −1.5 −1.0 −0.5 0.0 0.5 1.0 1.5 2.0

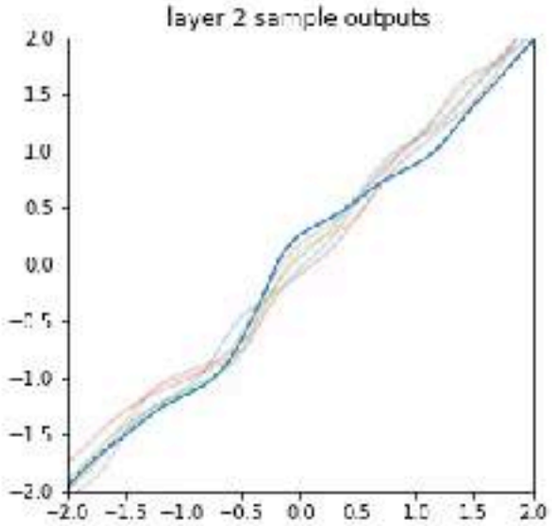

layer 1 input

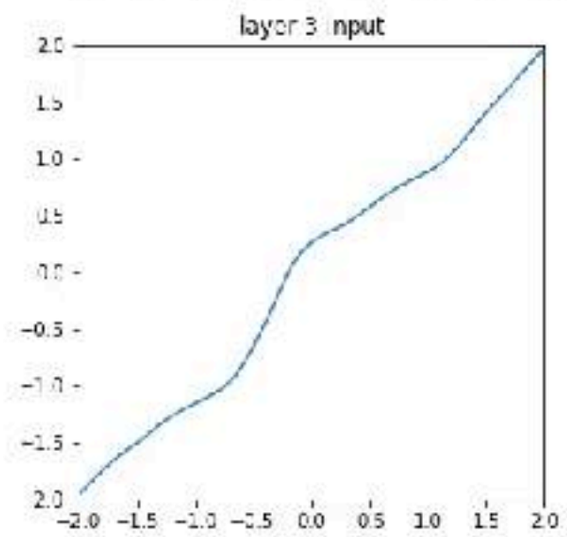

layer 1 covariance

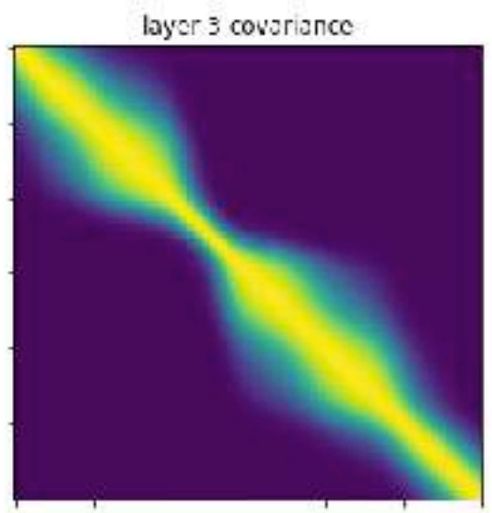

layer 1 sample outputs

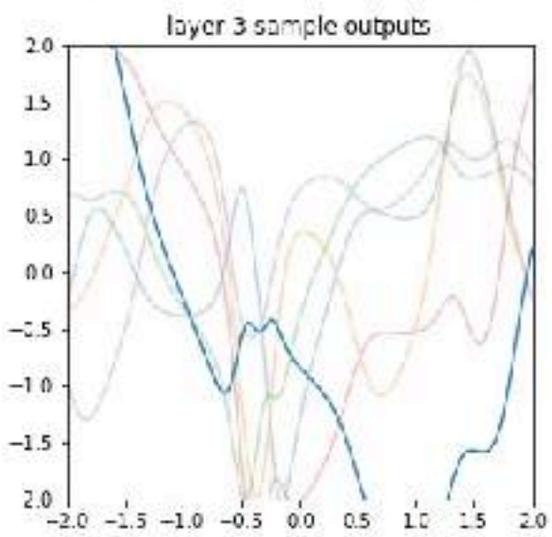

layer 2 input

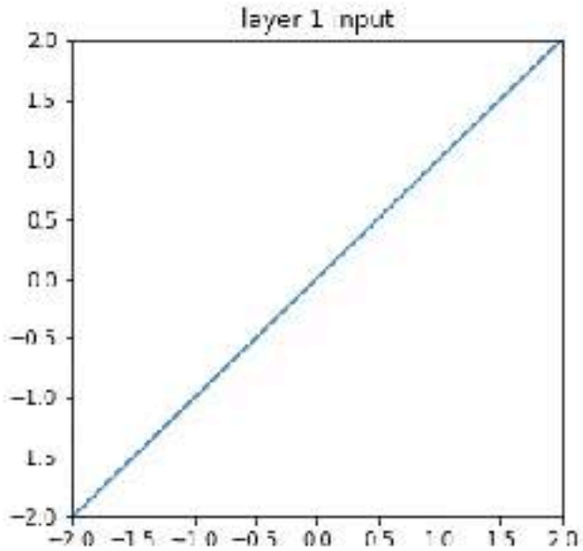

layer 2 covariance

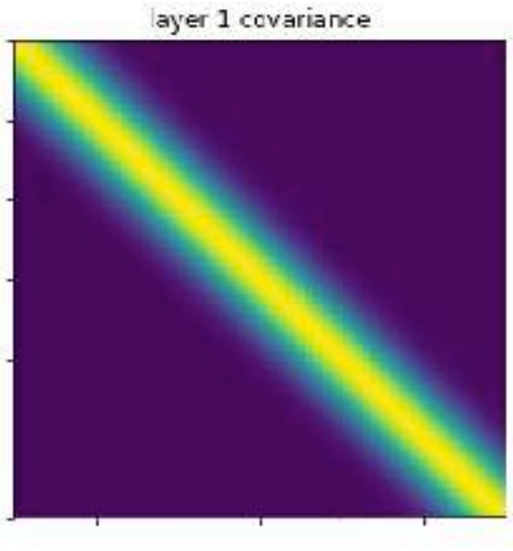

layer 2 sample outputs

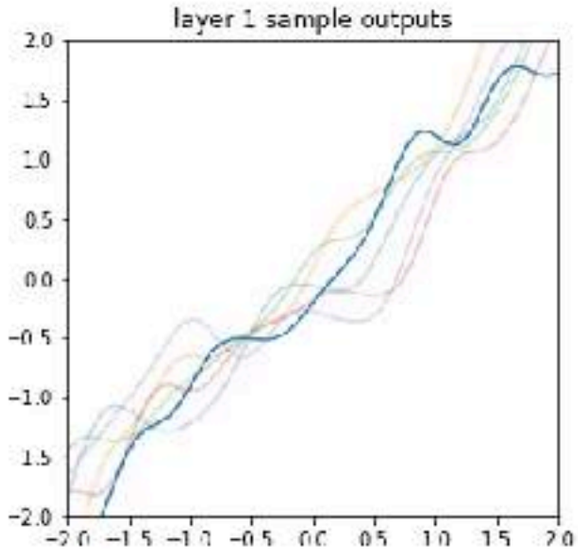

layer 3 input

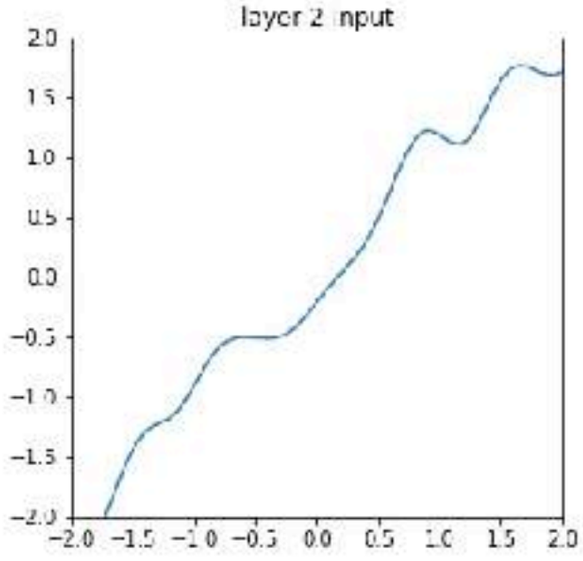

layer 3 covariance

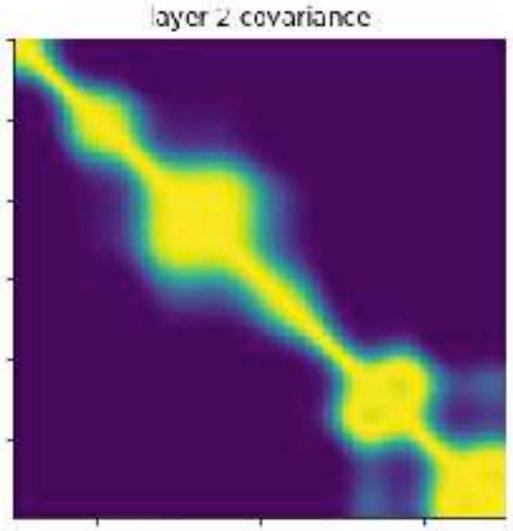

layer 3 sample outputs

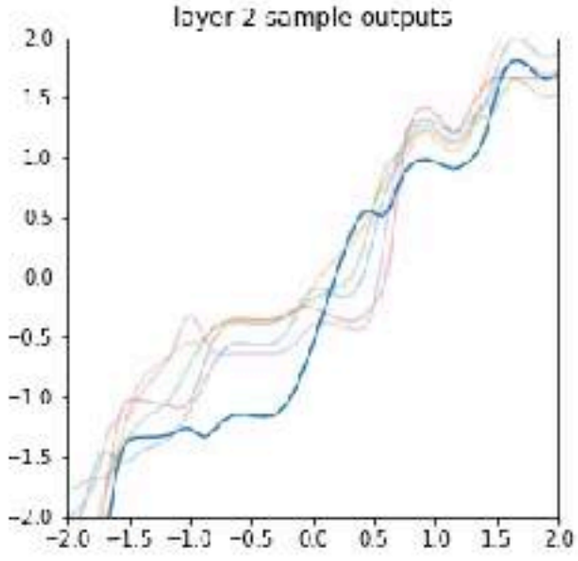

layer 1 input

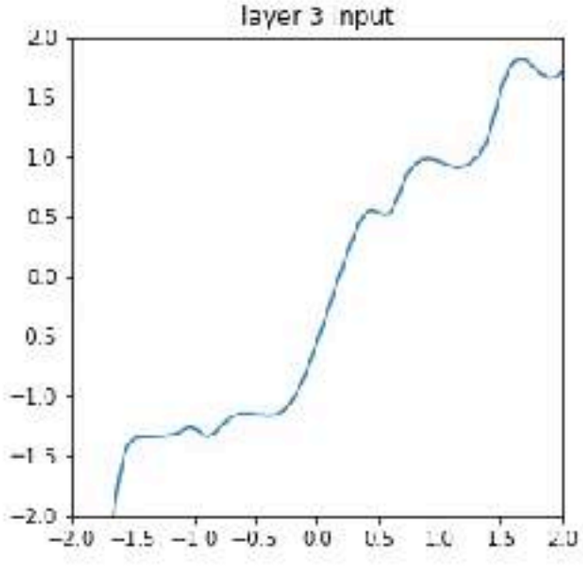

layer 1 covariance

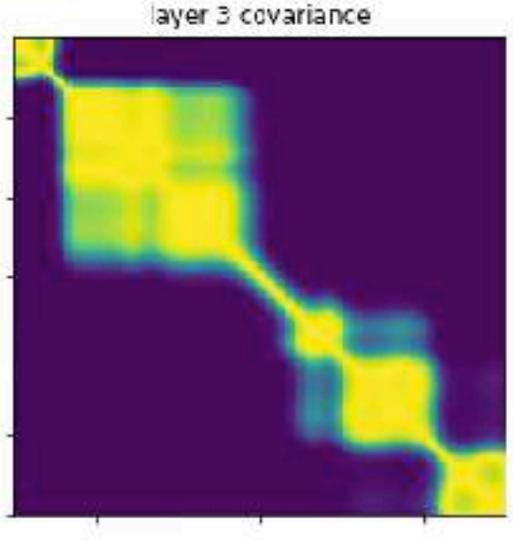

layer 1 sample outputs

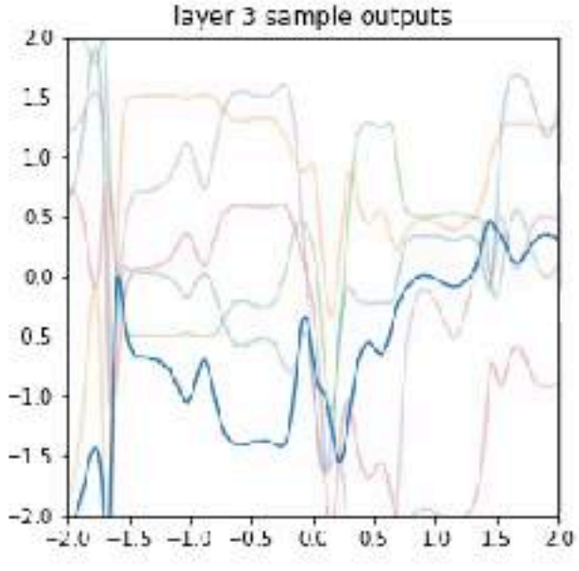

layer 2 input

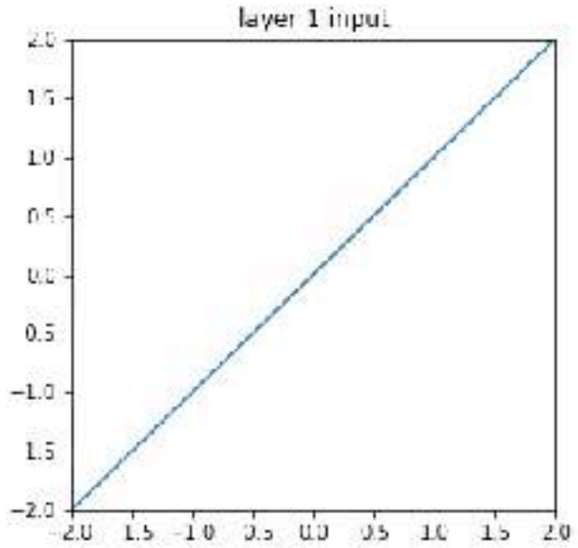

layer 2 covariance

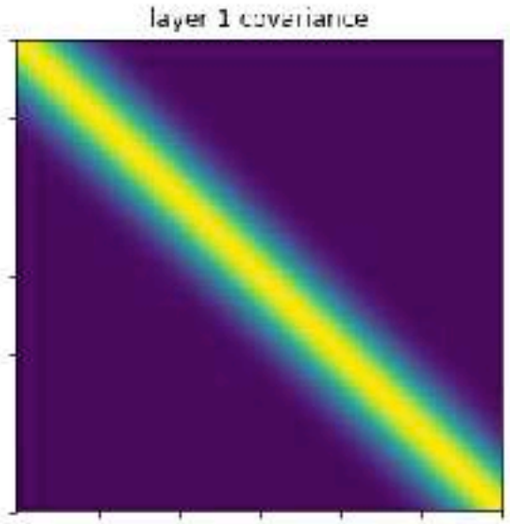

layer 2 sample outputs

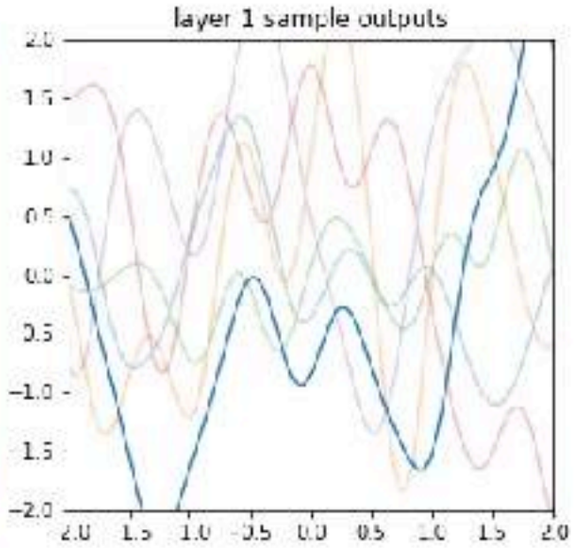

layer 3 input

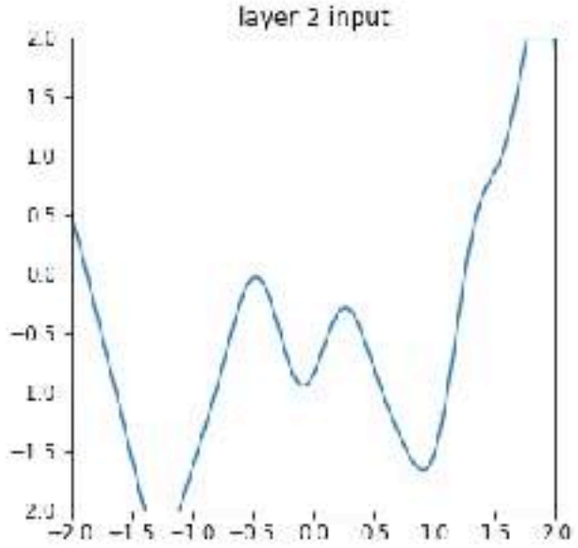

layer 3 covariance

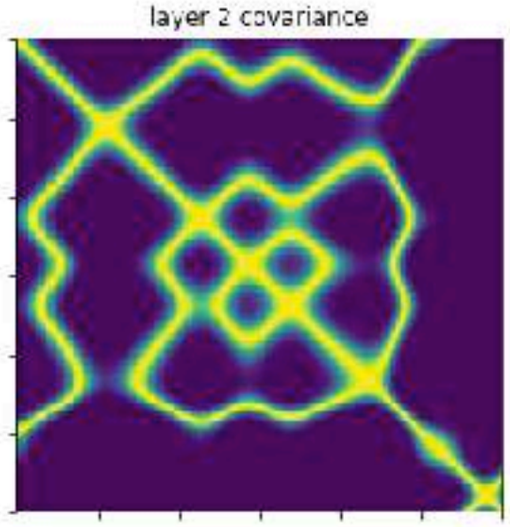

layer 3 sample outputs

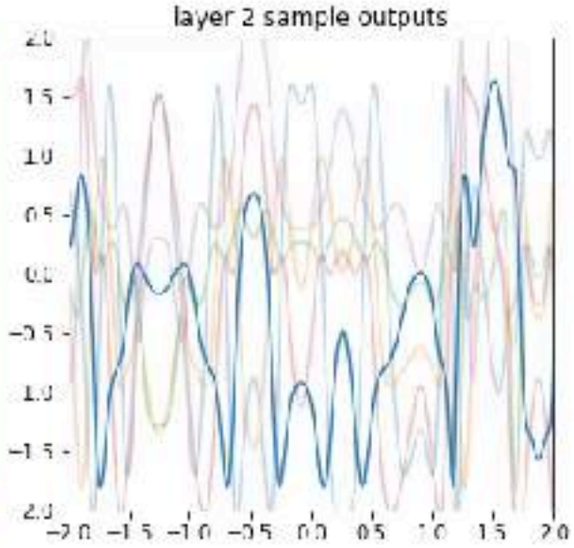

layer 1 input

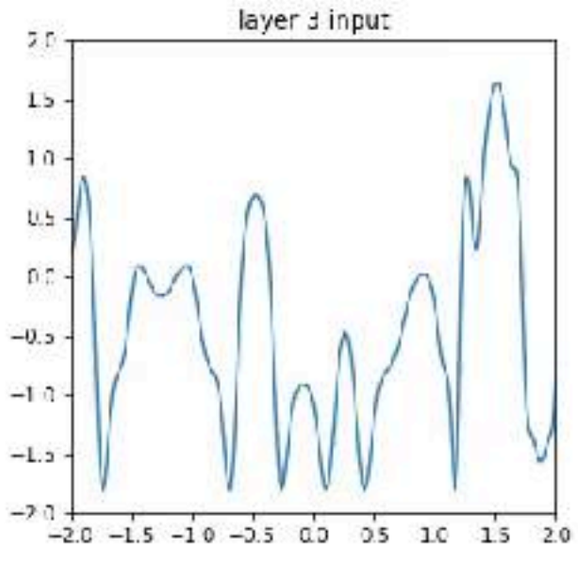

layer 1 covariance

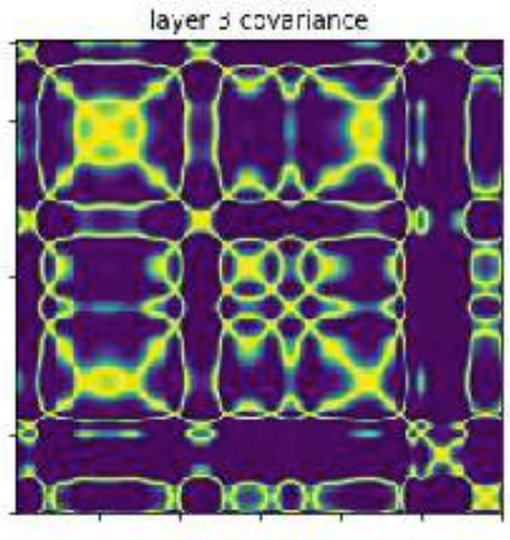

layer 1 sample outputs

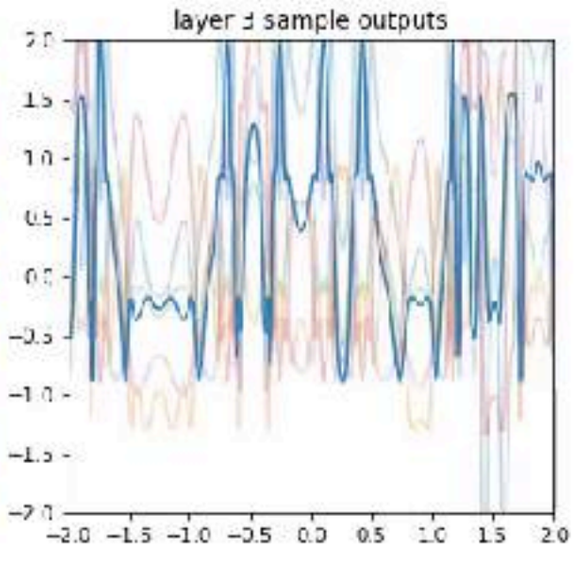

layer 2 input


layer 2 covariance


layer 2 sample outputs


layer 3 input


layer 3 covariance


layer 3 sample outputs

Outline:

- Why we want a Deep GP (2 reasons)
- The Deep GP model
- Inference in the Deep GP
- Why latent variables are important
- Inference over latent variables
- Results

Variational Inference

Fundamental identity for variational inference:

$$\log p(y) = \underbrace{\mathbb{E}_{q(z)} \log \frac{p(y|z)p(z)}{q(z)}}_{\text{VI objective ('ELBO')}} + \mathrm{KL}(q(z)||p(z|y))$$

Fixed

Maximize

Minimize

Model:

$$y_n \sim \mathcal{N}(f(g(x_n)), \sigma^2)$$
$$f \sim \mathcal{GP}(m_1, k_1)$$
$$g \sim \mathcal{GP}(m_2, k_2)$$

The VI identity in our case:

$$\log p(y) = \underbrace{\mathbb{E}_{q(f,g)} \log \frac{p(y|f,g)p(f)p(g)}{q(f,g)}}_{\text{VI objective ('ELBO')}} + \mathrm{KL}(q(f,g)||p(f,g|y))$$

$$\text{ELBO} = \mathbb{E}_{q(f,g)} \log \frac{p(y|f,g)p(f)p(g)}{q(f,g)}$$

Assumption 1 of 3

$$q(f,g) = q(f)q(g)$$

$$\text{ELBO} = \mathbb{E}_{q(f)q(g)} \log \frac{\left[\prod_n p(y_n|x_n,f,g)\right] p(f)p(g)}{q(f)q(g)}$$

Data terms

KL terms

$$\text{ELBO} = \sum_n \mathbb{E}_{q(f)q(g)} \log p(y_n|x_n,f,g) - \text{KL}(q(f)||p(f)) - \text{KL}(q(g)||p(g))$$

The KL terms:

$$\boxed{\mathrm{KL}(q(f)||p(f))} = -\mathbb{E}_{q(f)} \log \frac{p(f)}{q(f)}$$

$$\mathrm{KL}(q(f)||p(f)) = -\mathbb{E}_{q(f)} \log \frac{p(f|\tilde{f})p(\tilde{f})}{q(f|\tilde{f})q(\tilde{f})}$$

Finite set of inducing points $\tilde{f} = \{f(\tilde{x}_i)\}_{i=1}^{M}$

Assumption 2 of 3

$$q(f) = p(f|\tilde{f})q(\tilde{f})$$

$$\begin{aligned}
\mathrm{KL}(q(f)||p(f)) &= -\mathbb{E}_{q(f)} \log \frac{p(f|\tilde{f})p(\tilde{f})}{p(f|\tilde{f})q(\tilde{f})} \\
&= -\mathbb{E}_{q(\tilde{f})} \log \frac{p(f)}{q(\tilde{f})} \\
&= \mathrm{KL}(q(\tilde{f})||p(\tilde{f}))
\end{aligned}$$

$$p(\tilde{f}) = \mathcal{N}(0, \tilde{K})$$

$$q(\tilde{f}) = \mathcal{N}(\tilde{m}, \tilde{S})$$

Assumption 3 of 3

Assumption 2 of 3

$$q(f) = p(f|\tilde{f})q(\tilde{f})$$

Assumption 3 of 3

$$q(\tilde{f}) = \mathcal{N}(\tilde{m}, \tilde{S})$$

It follows that: $q(g) = \mathcal{GP}(\mu, \Sigma)$

With:

$$\mu(x) = \mathbf{k}(x)^\top \tilde{\mathbf{K}}^{-1} \tilde{\mathbf{m}}$$

$$\Sigma(x, x') = k(x, x') - \mathbf{k}(x)^\top \tilde{\mathbf{K}}^{-1} \left(\tilde{\mathbf{K}} - \tilde{\mathbf{S}}\right) \tilde{\mathbf{K}}^{-1} \mathbf{k}(x')$$

(Temporary matrix notation)

NB:

$$q(g(x)) = \mathcal{N}(\mu(x), \Sigma(x, x)))$$

The data terms:

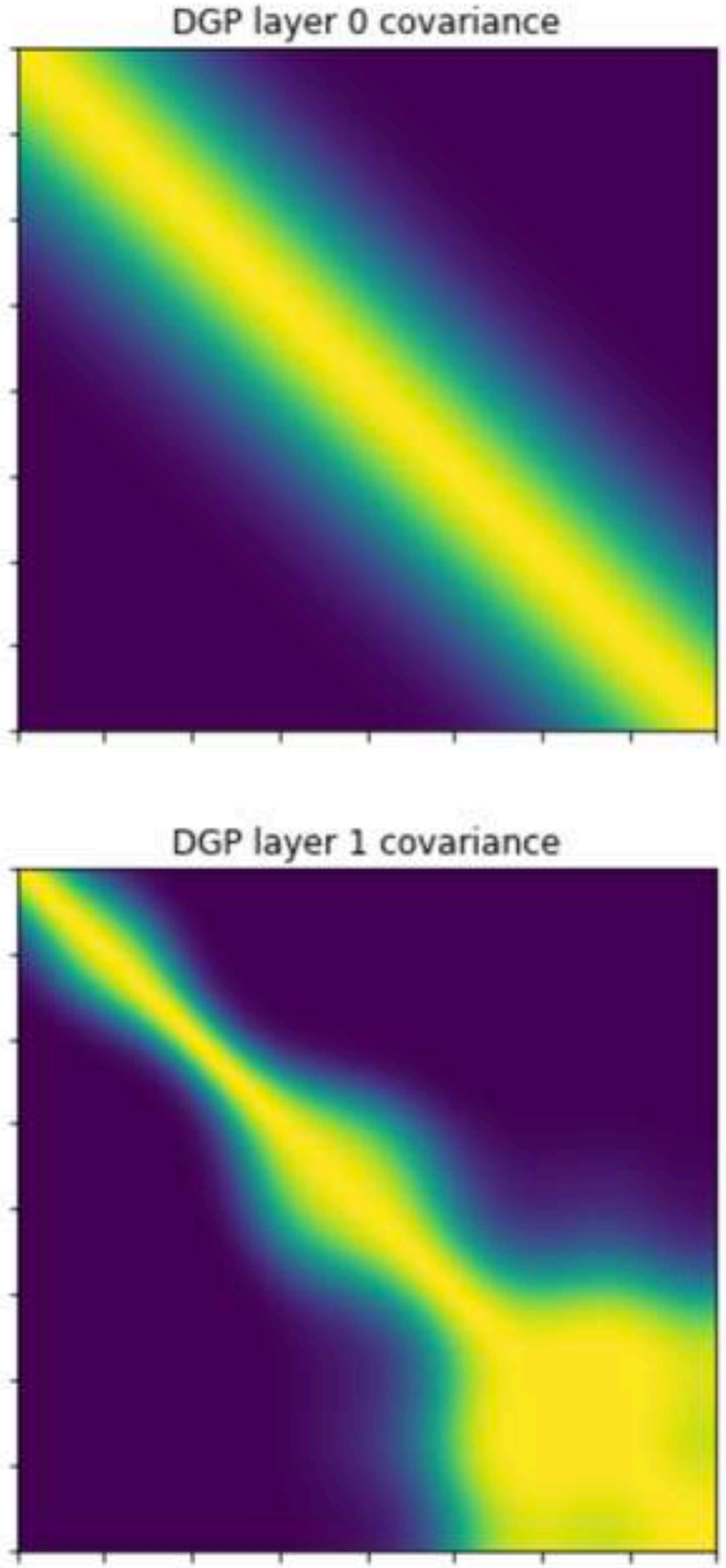
$$\text{ELBO} = \sum_n \mathbb{E}_{q(f)q(g)} \log p(y_n|x_n, f, g) - \text{KL}(q(f)||p(f)) - \text{KL}(q(g)||p(g))$$

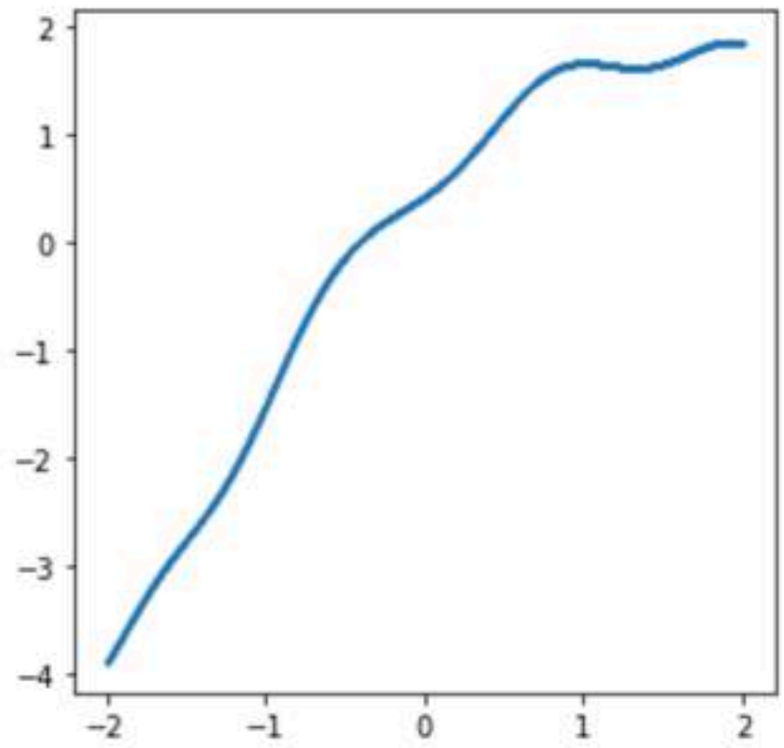
$$\begin{aligned}
\mathcal{L}_n =& \mathbb{E}_{q(f)q(g)} \log p(y_n|x_n, f, g) \\
=& \mathbb{E}_{q(f)q(g)} \log p(y_n|f(g(x_n))) \\
=& \mathbb{E}_{q(f)p(\epsilon)} \log p(y_n|f(z)), \quad z = \mu(x_n) + \epsilon\sqrt{\Sigma(x_n, x_n)}, \quad \epsilon \sim \mathcal{N}(0, 1)
\end{aligned}$$

Outline:

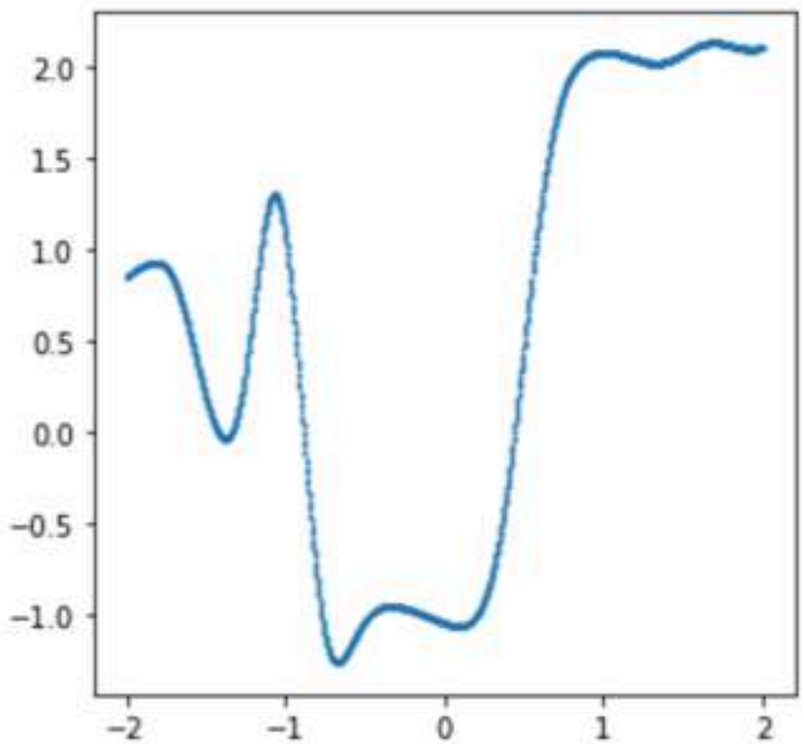
- Why we want a Deep GP (2 reasons)
- The Deep GP model
- Inference in the Deep GP
- Why latent variables are important
- Inference over latent variables
- Results

Noise-free


DGP layer 0 covariance


2
1
0
−1
−2
−3
−4
−2
−1
0
1
2

DGP layer 1 covariance

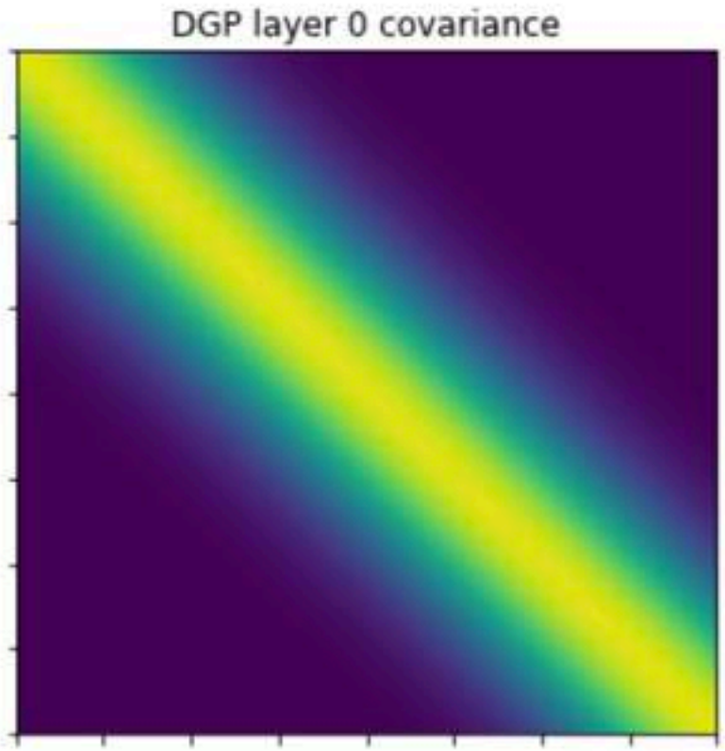

2.0
1.5
1.0
0.5
0.0
−0.5
−1.0
−2
−1
0
1
2

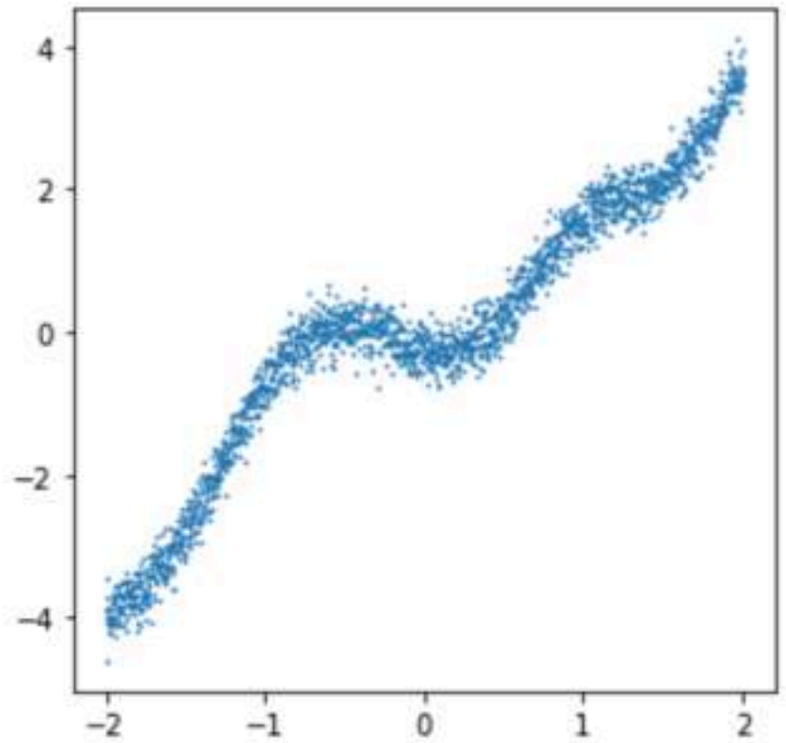
What's wrong with this?

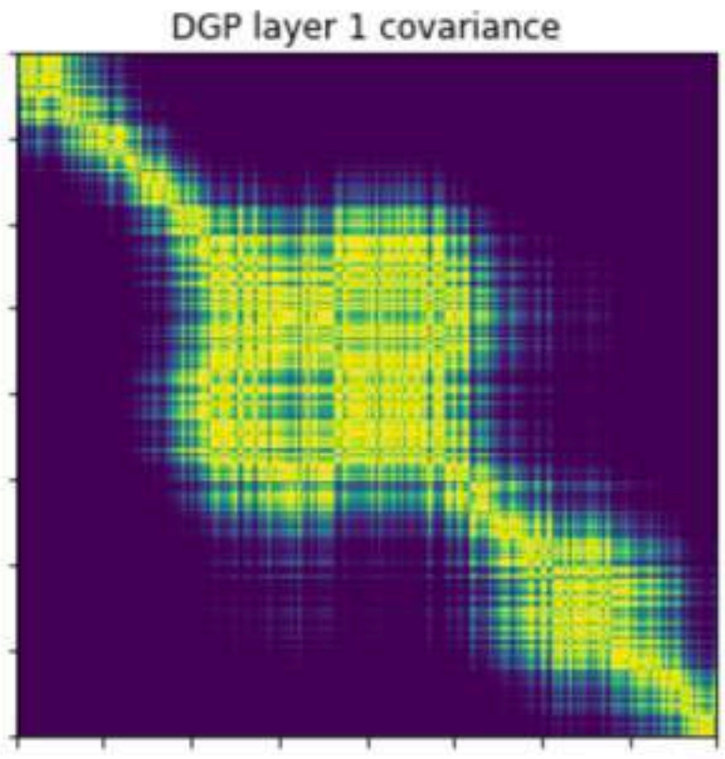
'Epistemic uncertainty' - uncertainty from lack of data
'Aleatoric uncertainty' - uncertainty from from inherent randomness

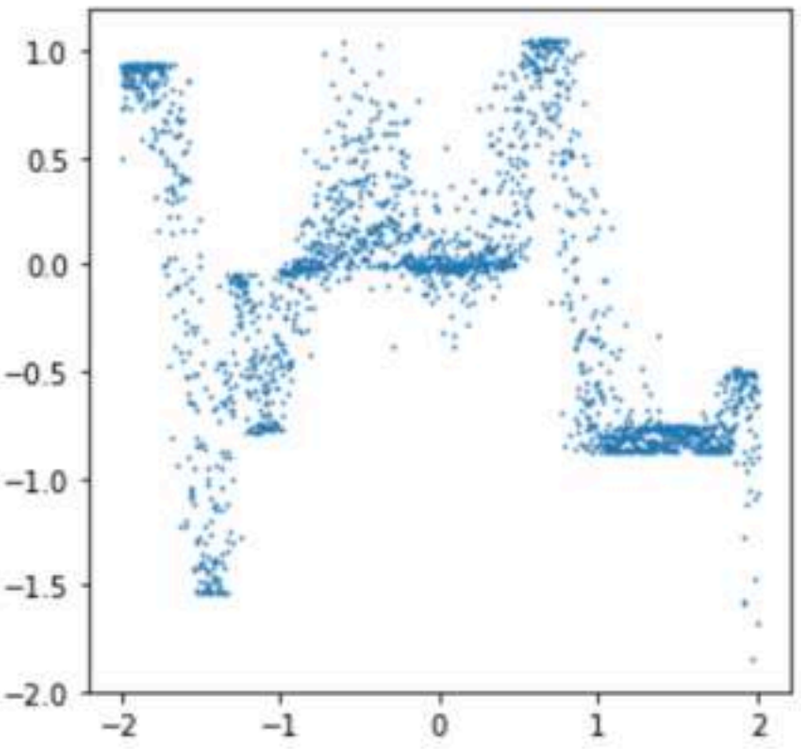
- GPs only model epistemic uncertainty, or marginal Gaussian aleatoric uncertainty for noisy kernels (noise = k(x, x) - k(x, x') for limit x->x')
- In noise-free case, we rely on epistemic uncertainty to get non-Gaussian marginals
- Noisy variables cannot be represented by our posterior, so the ELBO always favours the noise-free model

Additive noise







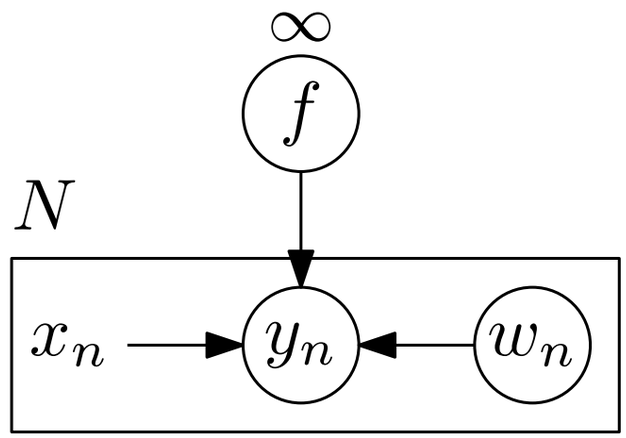


What's wrong with this?

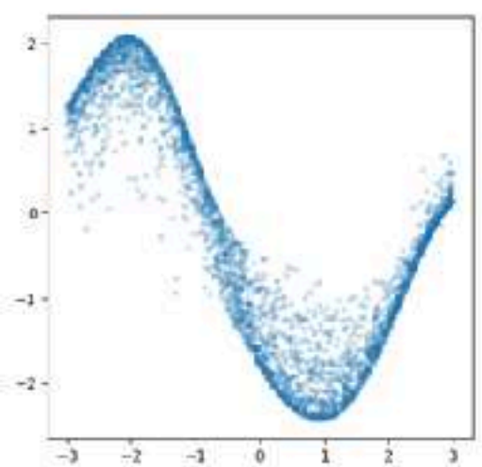
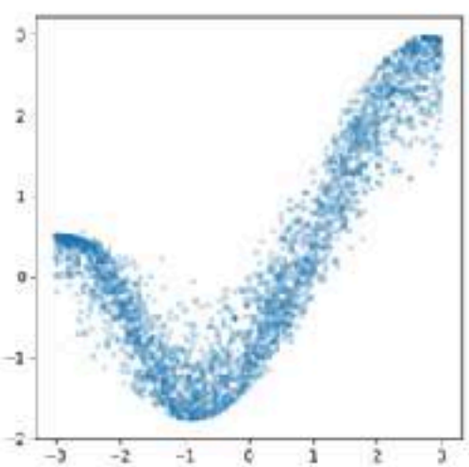
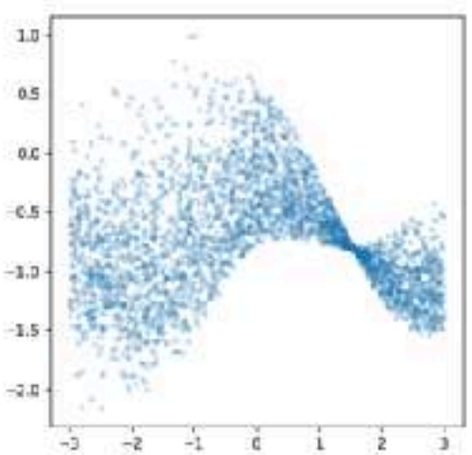
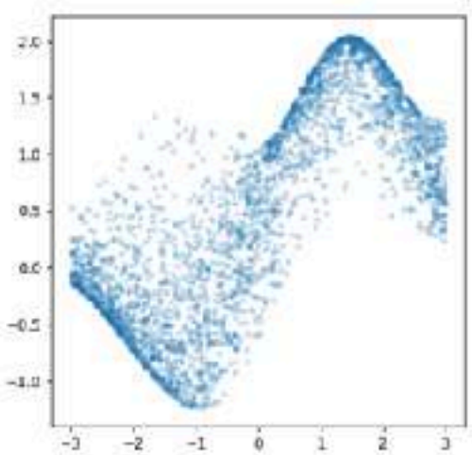
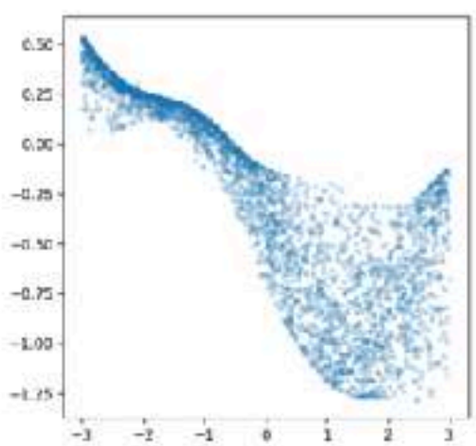
- Inference is difficult (cannot use inducing points)
- Modelling assumptions not clear (what does the noise mean?)
- Not easy to vary the dimensionality and strength of the noise

Single layer GP with ‘latent variables’

‘Latent variable’ = white noise GP

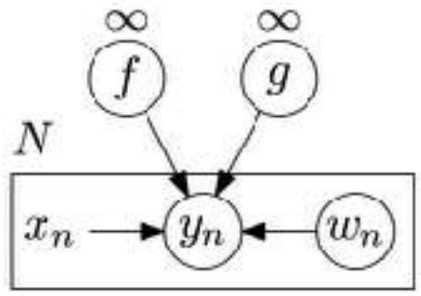


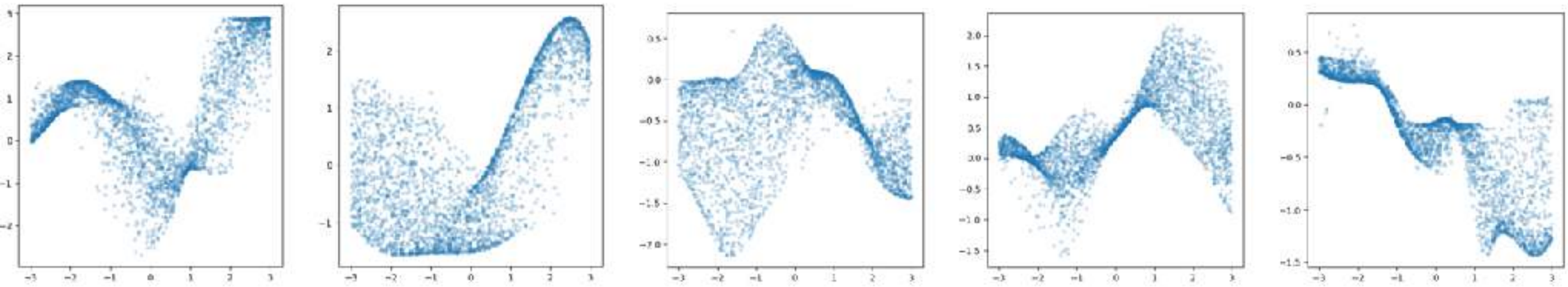
$$y_n = \mathcal{N}(f([x, w_n]), \sigma^2)$$

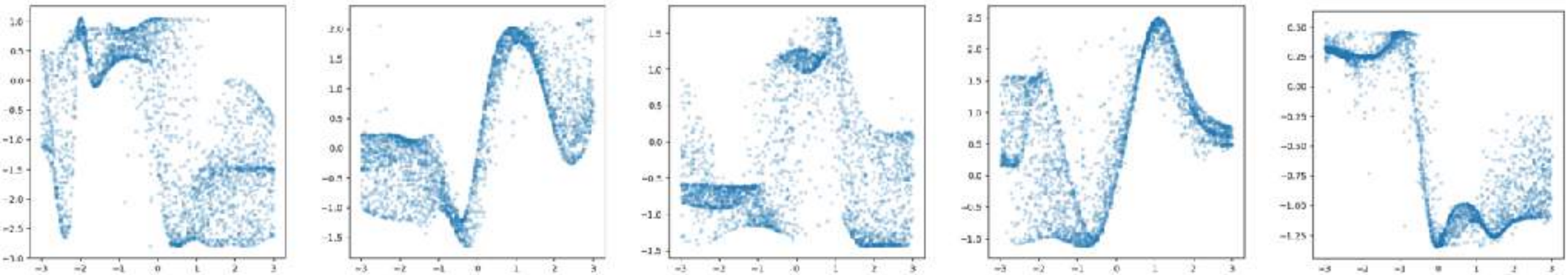
Going deeper:

$$y_n = \mathcal{N}(f(g([x, w_n])), \sigma^2)$$



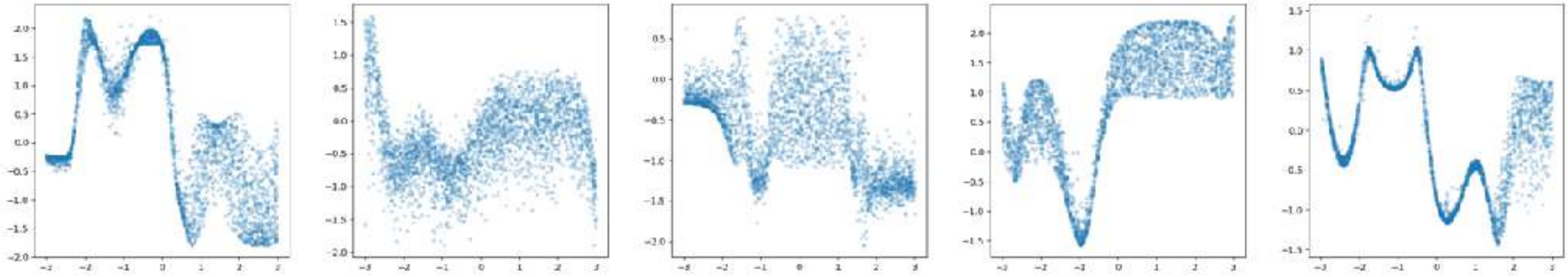


$$y_n = \mathcal{N}(f(g(h([x, w_n]))), \sigma^2)$$



Latent variables in different places:

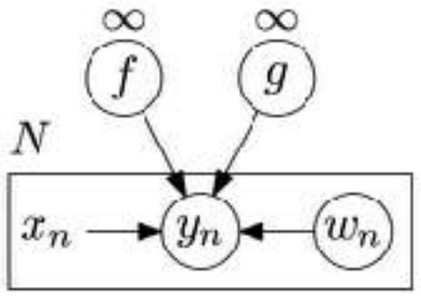
$$y_n = \mathcal{N}(f([g(h(x)), w_n]), \sigma^2)$$



Outline:

- Why we want a Deep GP (2 reasons)
- The Deep GP model
- Inference in the Deep GP
- Why latent variables are important
- Inference over latent variables
- Results

Inference with latent variables



$$q(w_n) = \mathcal{N}(a_n, b_n)$$

- Mean field for the latent variables
- This is reasonable as they are a priori independent

$$p(y) = \mathop{\mathbb{E}}_{f,g,w}\left[p(y|f,g,w)\frac{p(f)p(g)p(w)}{q(f)q(g)q(w)}\right]$$

$$\log p(y) \geq \sum_n (A_n - \mathrm{KL}_{w_n}) - \mathrm{KL}_f - \mathrm{KL}_g$$

$$A_n = \mathbb{E}_{f,g,w_n} \log p(y_n|f,g,w_n)$$

- We use variational inference or importance weighted variational inference for the latent variables
- Subtle modification to use the final layer analytic results

$$p(y) = \mathop{\mathbb{E}}_{f,g,w} \frac{1}{K}\sum_{k=1}^{K} p(y|f,g,w^{(k)})\frac{p(w^{(k)})}{q(w^{(k)})}\frac{p(f)p(g)}{q(f)q(g)}$$

$$\log p(y) \geq \sum_{n=1}^{N} B_n - \mathrm{KL}_f - \mathrm{KL}_g$$

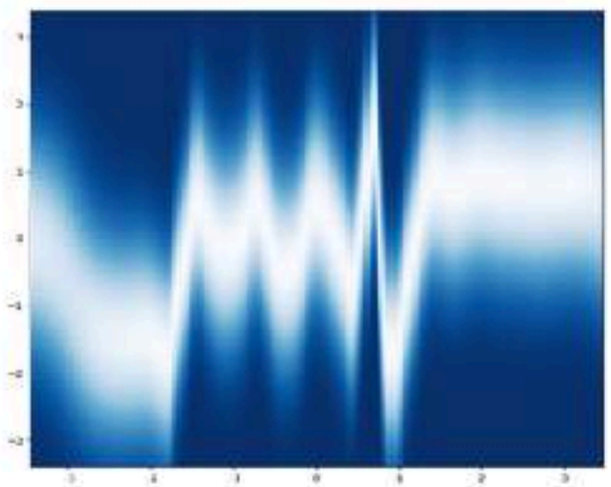
$$B_n = \mathop{\mathbb{E}}_{f,g,w_n} \log \frac{1}{K}\sum_k p(y_n|f,g,w_n^{(k)})\frac{p(w_n^{(k)})}{q(w_n^{(k)})}$$

Outline:

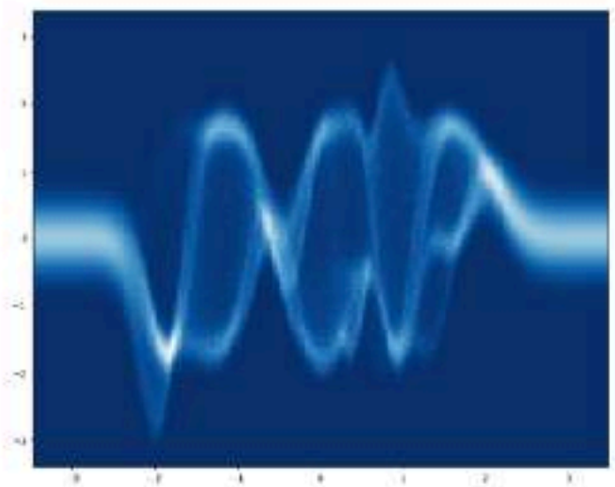
- Why we want a Deep GP (2 reasons)
- The Deep GP model
- Inference in the Deep GP
- Why latent variables are important
- Inference over latent variables
- Results

1D demo:

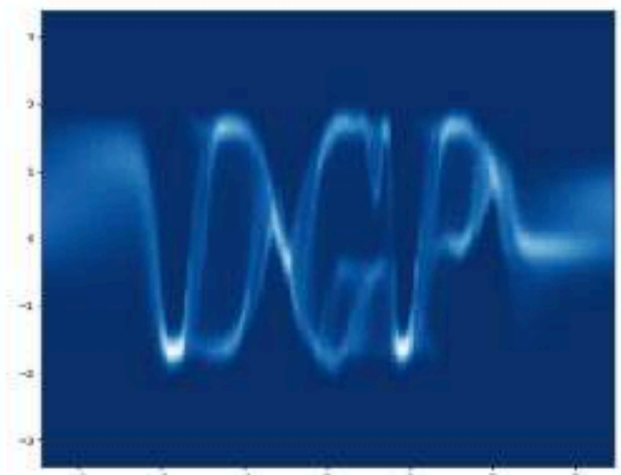
(a) GP



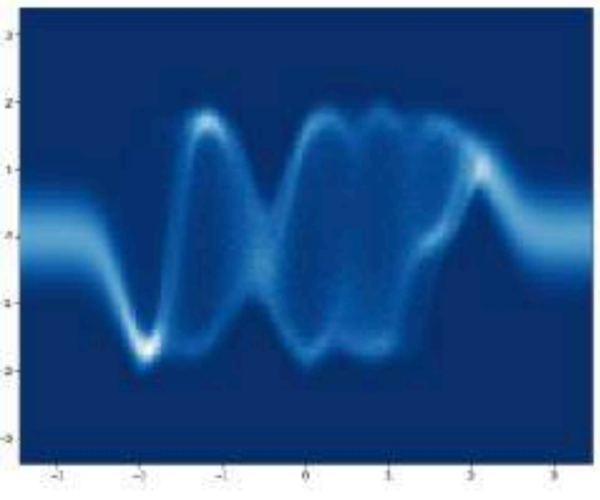
(b) GP-GP



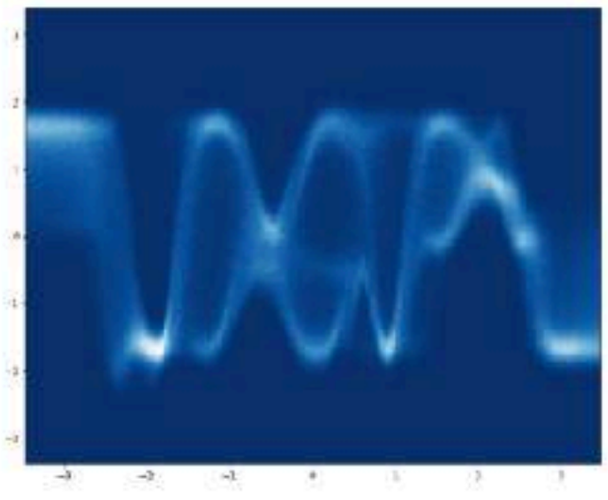
(c) LV-GP



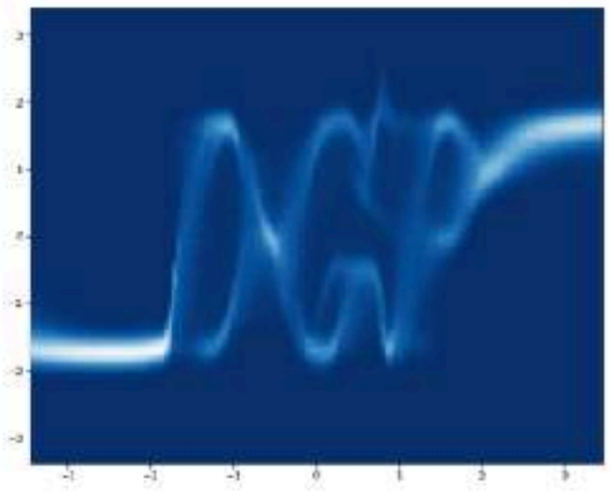
(d) LV-GP-GP



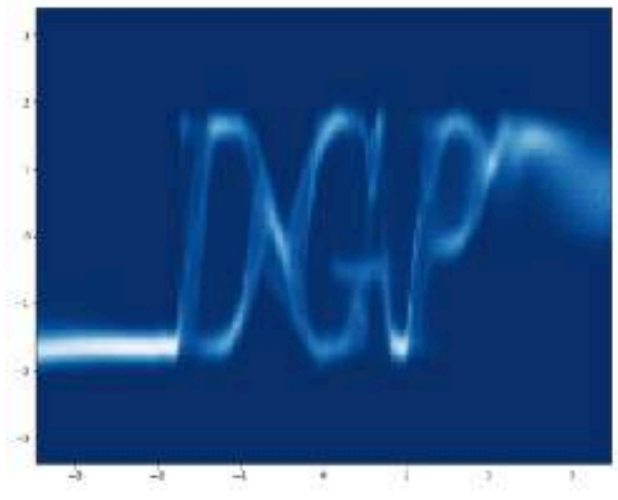
(a) LV-GP, VI



(b) LV-GP-GP, VI



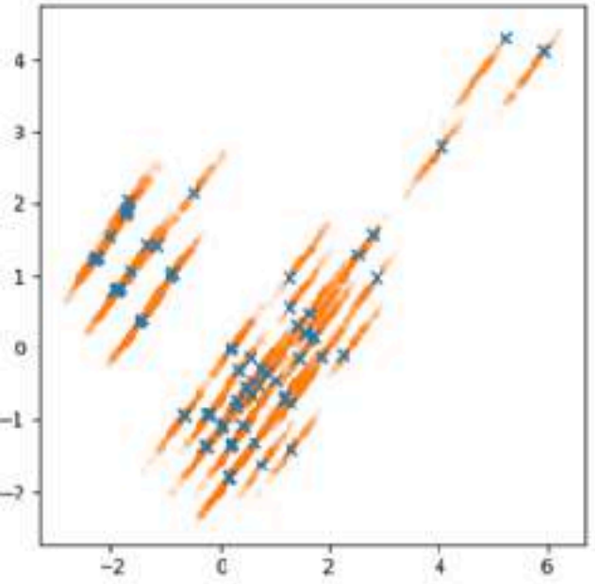
(c) LV-GP-GP-GP, VI



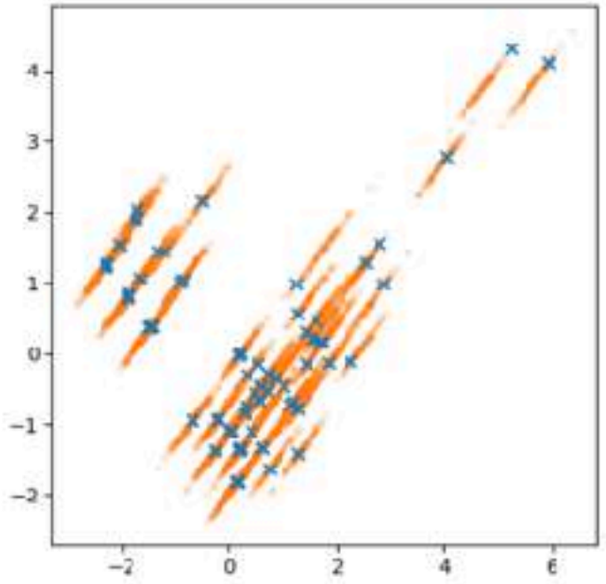
(d) LV-GP-GP-GP, IW

Model architecture			GP	GP-GP	GP-GP-GP	LV-GP		LV-GP-GP		LV-GP-GP-GP	
Importance weighted?			-	-	-	×	✓	×	✓	×	✓
dataset	N	D									
fertility	100	9	-1.56 (0.13)	-1.40 (0.13)	-1.33 (0.04)	-1.27 (0.06)	-1.27 (0.06)	-1.13 (0.04)	-1.21 (0.02)	-1.13 (0.04)	-1.19 (0.02)
concreteslump	103	7	1.59 (0.08)	1.87 (0.09)	1.94 (0.09)	1.59 (0.08)	1.58 (0.08)	2.16 (0.16)	1.86 (0.08)	2.08 (0.16)	1.92 (0.09)
autos	159	25	-0.32 (0.05)	-2.66 (0.18)	-3.16 (1.03)	-0.32 (0.05)	-0.33 (0.05)	-2.15 (0.81)	-3.22 (0.86)	-1.18 (0.24)	-1.16 (0.23)
servo	167	4	-0.16 (0.10)	-0.12 (0.09)	-0.05 (0.08)	-0.03 (0.09)	-0.02 (0.09)	-0.05 (0.09)	0.01 (0.09)	-0.05 (0.13)	0.05 (0.14)
breastcancer	194	33	-1.35 (0.06)	-12.28 (1.77)	-23.16 (4.73)	-1.31 (0.05)	-1.31 (0.05)	-1.53 (0.18)	-1.69 (0.23)	-1.47 (0.18)	-1.64 (0.22)
machine	209	7	-0.56 (0.05)	-0.44 (0.04)	-0.47 (0.05)	-0.55 (0.05)	-0.55 (0.05)	-0.48 (0.05)	-0.46 (0.04)	-0.50 (0.06)	-0.51 (0.05)
yacht	308	6	2.26 (0.13)	2.59 (0.12)	2.59 (0.10)	2.20 (0.14)	2.49 (0.16)	2.58 (0.09)	2.75 (0.08)	2.65 (0.10)	2.74 (0.11)
autompg	392	7	-0.34 (0.08)	-0.35 (0.06)	-0.39 (0.10)	-0.27 (0.08)	-0.27 (0.08)	-0.30 (0.07)	-0.24 (0.05)	-0.29 (0.07)	-0.24 (0.04)
boston	506	13	-0.12 (0.04)	-0.10 (0.08)	-0.12 (0.11)	-0.12 (0.03)	-0.12 (0.03)	-0.04 (0.08)	-0.07 (0.06)	-0.14 (0.17)	-0.10 (0.08)
forest	517	12	-1.37 (0.01)	-1.44 (0.02)	-1.59 (0.09)	-0.50 (0.08)	-0.43 (0.09)	0.10 (0.12)	0.13 (0.11)	0.10 (0.11)	0.08 (0.09)
stock	536	11	-0.20 (0.01)	-0.19 (0.04)	-0.26 (0.05)	-0.20 (0.01)	-0.20 (0.01)	-0.19 (0.04)	-0.17 (0.01)	-0.18 (0.05)	-0.19 (0.04)
pendulum	630	9	-0.43 (0.04)	-0.06 (0.06)	0.05 (0.07)	-0.55 (0.03)	-0.55 (0.03)	-0.24 (0.07)	-0.22 (0.06)	-0.15 (0.05)	-0.25 (0.06)
energy	768	8	1.56 (0.05)	1.58 (0.07)	1.59 (0.07)	1.53 (0.05)	1.54 (0.06)	1.59 (0.06)	1.59 (0.05)	1.63 (0.09)	1.59 (0.08)
concrete	1030	8	-0.41 (0.01)	-0.38 (0.02)	-0.27 (0.04)	-0.37 (0.01)	-0.37 (0.01)	-0.37 (0.04)	-0.32 (0.01)	-0.33 (0.03)	-0.27 (0.01)
solar	1066	10	-1.34 (0.07)	-1.31 (0.08)	-1.34 (0.08)	1.35 (0.05)	1.53 (0.04)	2.28 (0.13)	2.30 (0.05)	2.03 (0.30)	2.01 (0.23)
airfoil	1503	5	-0.34 (0.05)	0.05 (0.05)	0.13 (0.02)	-0.41 (0.03)	-0.39 (0.03)	-0.01 (0.03)	0.01 (0.03)	0.01 (0.05)	0.06 (0.04)
winered	1599	11	-1.14 (0.04)	-1.15 (0.04)	-1.15 (0.05)	-1.13 (0.04)	-1.13 (0.04)	1.13 (0.23)	1.52 (0.17)	0.47 (0.42)	2.00 (0.18)
gas	2565	128	0.69 (0.03)	1.02 (0.01)	1.00 (0.02)	0.98 (0.15)	1.02 (0.15)	1.36 (0.02)	1.42 (0.03)	1.35 (0.04)	1.44 (0.01)
skillcraft	3338	19	-0.97 (0.01)	-0.98 (0.02)	-0.98 (0.02)	-0.95 (0.01)	-0.95 (0.01)	-0.95 (0.01)	-0.94 (0.02)	-0.95 (0.01)	-0.94 (0.02)
sml	4137	26	1.04 (0.01)	1.39 (0.01)	1.42 (0.02)	1.03 (0.01)	1.03 (0.01)	1.35 (0.01)	1.33 (0.01)	1.41 (0.01)	1.41 (0.01)
winewhite	4898	11	-1.14 (0.00)	-1.13 (0.01)	-1.13 (0.01)	-1.14 (0.01)	-1.14 (0.01)	-1.18 (0.01)	-1.13 (0.01)	-1.19 (0.01)	-1.12 (0.01)
parkinsons	5875	20	0.48 (0.01)	0.84 (0.01)	0.88 (0.10)	0.32 (0.05)	0.47 (0.03)	0.25 (0.06)	0.57 (0.05)	0.64 (0.05)	0.60 (0.06)
kin8nm	8192	8	-0.35 (0.01)	-0.07 (0.01)	0.02 (0.01)	-0.42 (0.01)	-0.40 (0.01)	-0.09 (0.01)	-0.05 (0.01)	-0.02 (0.01)	-0.01 (0.01)
power	9568	4	0.01 (0.02)	0.04 (0.02)	0.04 (0.02)	0.04 (0.01)	0.07 (0.02)	0.08 (0.01)	0.12 (0.01)	0.07 (0.02)	0.14 (0.01)
naval	11934	14	3.84 (0.09)	3.41 (0.03)	3.51 (0.02)	3.78 (0.07)	3.79 (0.08)	3.33 (0.04)	3.38 (0.04)	3.48 (0.02)	3.41 (0.02)
pol	15000	26	0.17 (0.01)	1.24 (0.03)	1.47 (0.01)	0.10 (0.02)	0.18 (0.03)	1.36 (0.03)	1.65 (0.02)	1.61 (0.06)	1.90 (0.02)
elevators	16599	18	-0.45 (0.00)	-0.35 (0.01)	-0.35 (0.01)	-0.41 (0.01)	-0.40 (0.01)	-0.31 (0.00)	-0.29 (0.00)	-0.29 (0.01)	-0.28 (0.00)
bike	17379	17	0.66 (0.02)	1.85 (0.02)	2.36 (0.05)	1.29 (0.01)	1.35 (0.01)	2.74 (0.06)	2.77 (0.04)	3.17 (0.08)	3.07 (0.10)
kin40k	40000	8	-0.27 (0.01)	0.47 (0.00)	1.00 (0.02)	-0.41 (0.01)	-0.33 (0.00)	0.35 (0.01)	0.42 (0.00)	0.75 (0.08)	0.86 (0.03)
protein	45730	9	-1.13 (0.00)	-1.08 (0.00)	-1.06 (0.00)	-0.85 (0.01)	-0.79 (0.01)	-0.73 (0.01)	-0.67 (0.00)	-0.71 (0.01)	-0.64 (0.01)
tamielectric	45781	3	-1.42 (0.00)	-1.42 (0.00)	-1.42 (0.00)	-1.28 (0.00)	-1.27 (0.00)	-1.29 (0.00)	-1.28 (0.00)	-1.28 (0.00)	-1.27 (0.00)
keggdirected	48827	20	0.98 (0.02)	1.03 (0.02)	1.03 (0.02)	0.97 (0.02)	1.37 (0.03)	1.13 (0.09)	1.71 (0.04)	1.09 (0.04)	1.68 (0.04)
slice	53500	385	0.35 (0.01)	1.09 (0.02)	1.61 (0.01)	0.41 (0.02)	0.44 (0.01)	1.08 (0.01)	1.13 (0.01)	1.58 (0.01)	1.57 (0.02)
keggundirected	63608	27	0.69 (0.00)	0.72 (0.00)	0.73 (0.01)	1.72 (0.02)	1.76 (0.02)	2.27 (0.27)	3.23 (0.02)	2.18 (0.17)	3.25 (0.05)
3droad	434874	3	-1.03 (0.00)	-0.93 (0.00)	-0.87 (0.01)	-0.96 (0.00)	-0.90 (0.01)	-0.94 (0.03)	-0.77 (0.01)	-0.92 (0.02)	-0.75 (0.00)
song	515345	90	-1.21 (0.00)	-1.18 (0.00)	-1.18 (0.00)	-1.19 (0.00)	-1.18 (0.00)	-1.15 (0.00)	-1.14 (0.00)	-1.14 (0.00)	-1.12 (0.00)
buzz	583250	77	-0.26 (0.00)	-0.03 (0.00)	-0.01 (0.00)	-0.27 (0.01)	-0.27 (0.01)	0.01 (0.00)	0.02 (0.01)	0.03 (0.00)	0.05 (0.00)
nytaxi	1420068	8	-0.78 (0.00)	-0.69 (0.00)	-0.68 (0.00)	-0.67 (0.01)	-0.61 (0.01)	-0.50 (0.00)	-0.43 (0.00)	-0.49 (0.00)	-0.42 (0.00)
houseelectric	2049280	11	1.29 (0.01)	1.51 (0.00)	1.51 (0.00)	1.35 (0.04)	1.43 (0.01)	1.50 (0.00)	1.80 (0.00)	1.51 (0.00)	1.79 (0.01)
Median			-0.28	-0.09	-0.09	-0.29	-0.27	-0.00	-0.02	0.02	0.05
Median difference from GP			0	0.08	0.10	0.02	0.05	0.20	0.25	0.22	0.29
Mean			-0.09 (0.19)	-0.23 (0.39)	-0.46 (0.69)	0.07 (0.19)	0.11 (0.19)	0.32 (0.22)	0.38 (0.24)	0.38 (0.22)	0.49 (0.23)
Mean difference from GP			0(0)	-0.14 (0.31)	-0.37 (0.63)	0.15 (0.08)	0.20 (0.08)	0.41 (0.15)	0.47 (0.17)	0.47 (0.13)	0.57 (0.15)
Average rank			2.67 (0.35)	4.29 (0.37)	5.04 (0.45)	3.32 (0.36)	4.30 (0.36)	5.26 (0.34)	6.62 (0.32)	6.21 (0.32)	7.29 (0.33)

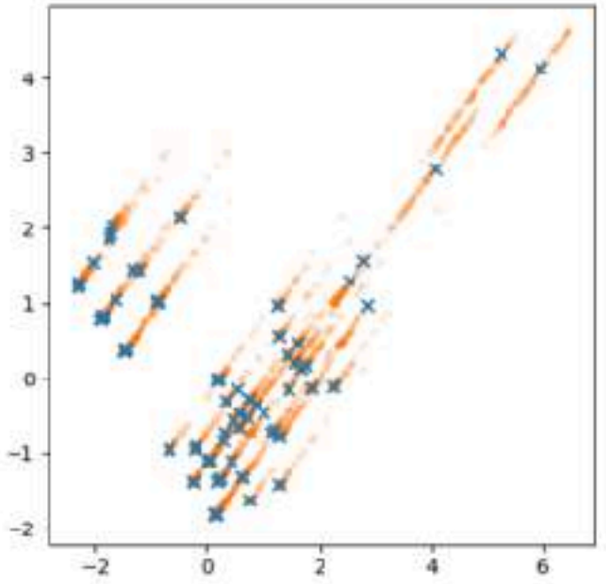
Model architecture			GP	GP-GP	GP-GP-GP	LV-GP		LV-GP-GP		LV-GP-GP-GP	
Importance weighted?			-	-	-	×	✓	×	✓	×	✓
dataset	N	D	Test log likelihoods (standard errors)								
solar	1066	10	-1.34 (0.07)	-1.31 (0.08)	-1.34 (0.08)	1.35 (0.05)	1.53 (0.04)	2.28 (0.13)	2.30 (0.05)	2.03 (0.30)	2.01 (0.23)
winered	1599	11	-1.14 (0.04)	-1.15 (0.04)	-1.15 (0.05)	-1.13 (0.04)	-1.13 (0.04)	1.13 (0.23)	1.52 (0.17)	0.47 (0.42)	2.00 (0.18)
bike	17379	17	0.66 (0.02)	1.85 (0.02)	2.36 (0.05)	1.29 (0.01)	1.35 (0.01)	2.74 (0.06)	2.77 (0.04)	3.17 (0.08)	3.07 (0.10)
...			*... an additional 36 rows can be found in the supplementary material...*								
Median			-0.28	-0.09	-0.09	-0.29	-0.27	-0.00	-0.02	0.02	0.05
Median difference from GP			0	0.08	0.10	0.02	0.05	0.20	0.25	0.22	0.29
Mean			-0.09 (0.19)	-0.23 (0.39)	-0.46 (0.69)	0.07 (0.19)	0.11 (0.19)	0.32 (0.22)	0.38 (0.24)	0.38 (0.22)	0.49 (0.23)
Mean difference from GP			0(0)	-0.14 (0.31)	-0.37 (0.63)	0.15 (0.08)	0.20 (0.08)	0.41 (0.15)	0.47 (0.17)	0.47 (0.13)	0.57 (0.15)
Average ranks			2.67 (0.35)	4.29 (0.37)	5.04 (0.45)	3.32 (0.36)	4.30 (0.36)	5.26 (0.34)	6.62 (0.32)	6.21 (0.32)	7.29 (0.33)



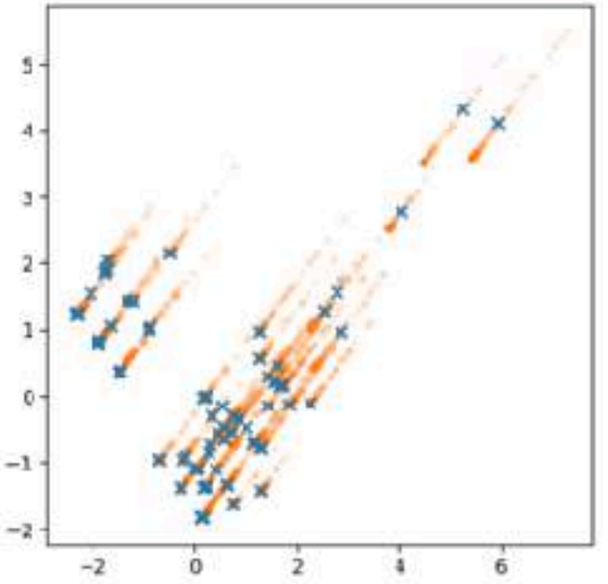
(a) solar GP



(b) solar GP-GP

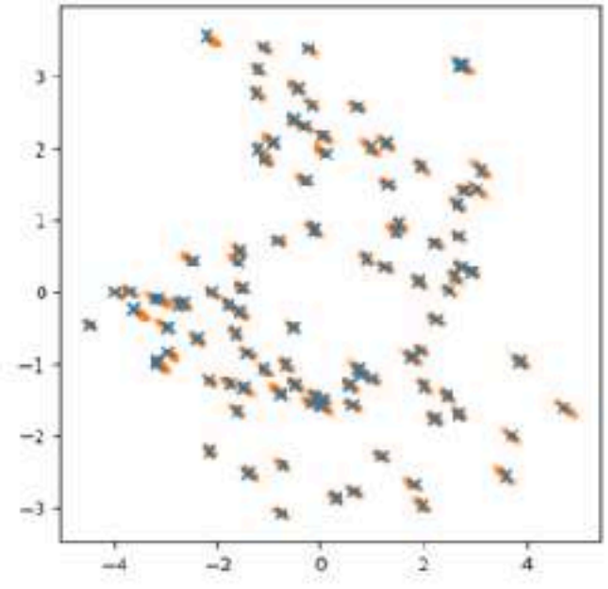


(c) solar LV-GP

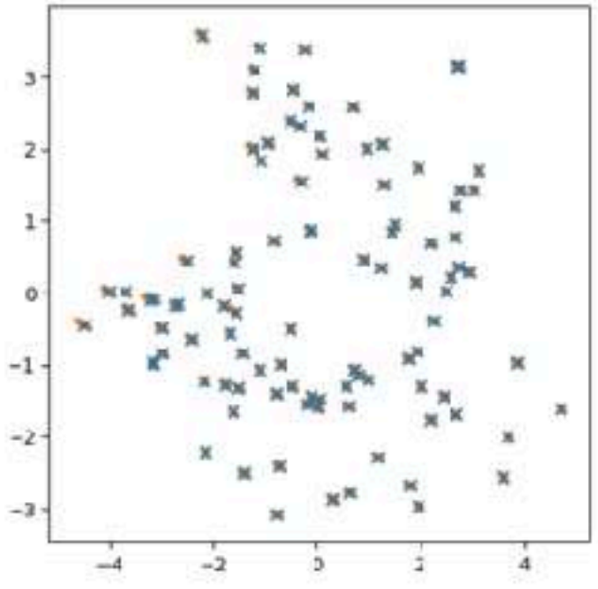


(d) solar LV-GP-GP

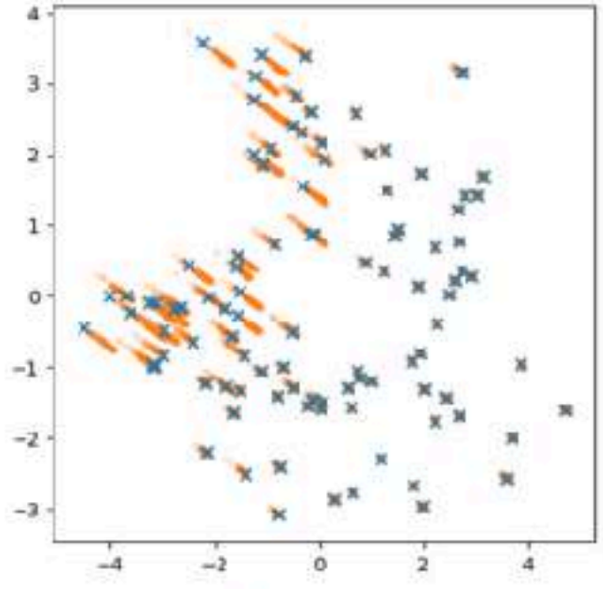
Model architecture			GP	GP-GP	GP-GP-GP	LV-GP		LV-GP-GP		LV-GP-GP-GP	
Importance weighted?			-	-	-	×	✓	×	✓	×	✓
dataset	N	D	Test log likelihoods (standard errors)								
solar	1066	10	-1.34 (0.07)	-1.31 (0.08)	-1.34 (0.08)	1.35 (0.05)	1.53 (0.04)	2.28 (0.13)	2.30 (0.05)	2.03 (0.30)	2.01 (0.23)
winered	1599	11	-1.14 (0.04)	-1.15 (0.04)	-1.15 (0.05)	-1.13 (0.04)	-1.13 (0.04)	1.13 (0.23)	1.52 (0.17)	0.47 (0.42)	2.00 (0.18)
bike	17379	17	0.66 (0.02)	1.85 (0.02)	2.36 (0.05)	1.29 (0.01)	1.35 (0.01)	2.74 (0.06)	2.77 (0.04)	3.17 (0.08)	3.07 (0.10)
...			*... an additional 36 rows can be found in the supplementary material...*								
Median			-0.28	-0.09	-0.09	-0.29	-0.27	-0.00	-0.02	0.02	0.05
Median difference from GP			0	0.08	0.10	0.02	0.05	0.20	0.25	0.22	0.29
Mean			-0.09 (0.19)	-0.23 (0.39)	-0.46 (0.69)	0.07 (0.19)	0.11 (0.19)	0.32 (0.22)	0.38 (0.24)	0.38 (0.22)	0.49 (0.23)
Mean difference from GP			0(0)	-0.14 (0.31)	-0.37 (0.63)	0.15 (0.08)	0.20 (0.08)	0.41 (0.15)	0.47 (0.17)	0.47 (0.13)	0.57 (0.15)
Average ranks			2.67 (0.35)	4.29 (0.37)	5.04 (0.45)	3.32 (0.36)	4.30 (0.36)	5.26 (0.34)	6.62 (0.32)	6.21 (0.32)	7.29 (0.33)



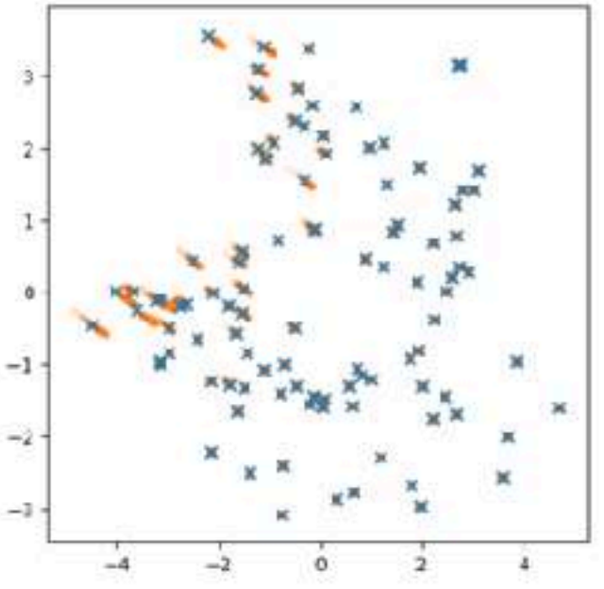
(e) bike GP



(f) bike GP-GP



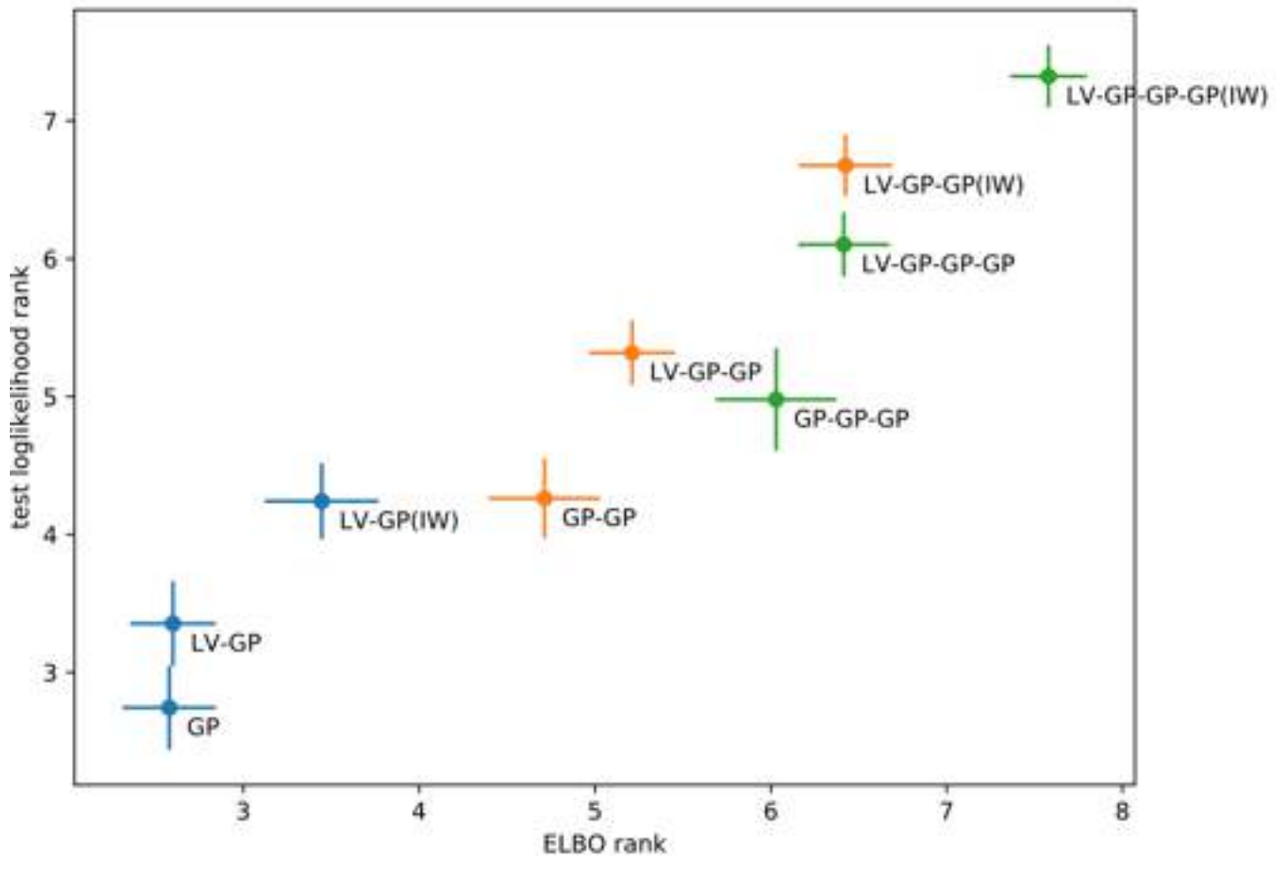
(g) bike LV-GP



(h) bike LV-GP-GP

Model architecture			GP	GP-GP	GP-GP-GP	LV-GP		LV-GP-GP		LV-GP-GP-GP	
Importance weighted?			-	-	-	×	✓	×	✓	×	✓
dataset	N	D	Test log likelihoods (standard errors)								
solar	1066	10	-1.34 (0.07)	-1.31 (0.08)	-1.34 (0.08)	1.35 (0.05)	1.53 (0.04)	2.28 (0.13)	2.30 (0.05)	2.03 (0.30)	2.01 (0.23)
winered	1599	11	-1.14 (0.04)	-1.15 (0.04)	-1.15 (0.05)	-1.13 (0.04)	-1.13 (0.04)	1.13 (0.23)	1.52 (0.17)	0.47 (0.42)	2.00 (0.18)
bike	17379	17	0.66 (0.02)	1.85 (0.02)	2.36 (0.05)	1.29 (0.01)	1.35 (0.01)	2.74 (0.06)	2.77 (0.04)	3.17 (0.08)	3.07 (0.10)
...			*... an additional 36 rows can be found in the supplementary material...*								
Median			-0.28	-0.09	-0.09	-0.29	-0.27	-0.00	-0.02	0.02	0.05
Median difference from GP			0	0.08	0.10	0.02	0.05	0.20	0.25	0.22	0.29
Mean			-0.09 (0.19)	-0.23 (0.39)	-0.46 (0.69)	0.07 (0.19)	0.11 (0.19)	0.32 (0.22)	0.38 (0.24)	0.38 (0.22)	0.49 (0.23)
Mean difference from GP			0(0)	-0.14 (0.31)	-0.37 (0.63)	0.15 (0.08)	0.20 (0.08)	0.41 (0.15)	0.47 (0.17)	0.47 (0.13)	0.57 (0.15)
Average ranks			2.67 (0.35)	4.29 (0.37)	5.04 (0.45)	3.32 (0.36)	4.30 (0.36)	5.26 (0.34)	6.62 (0.32)	6.21 (0.32)	7.29 (0.33)

ELBO vs test log-likelihood



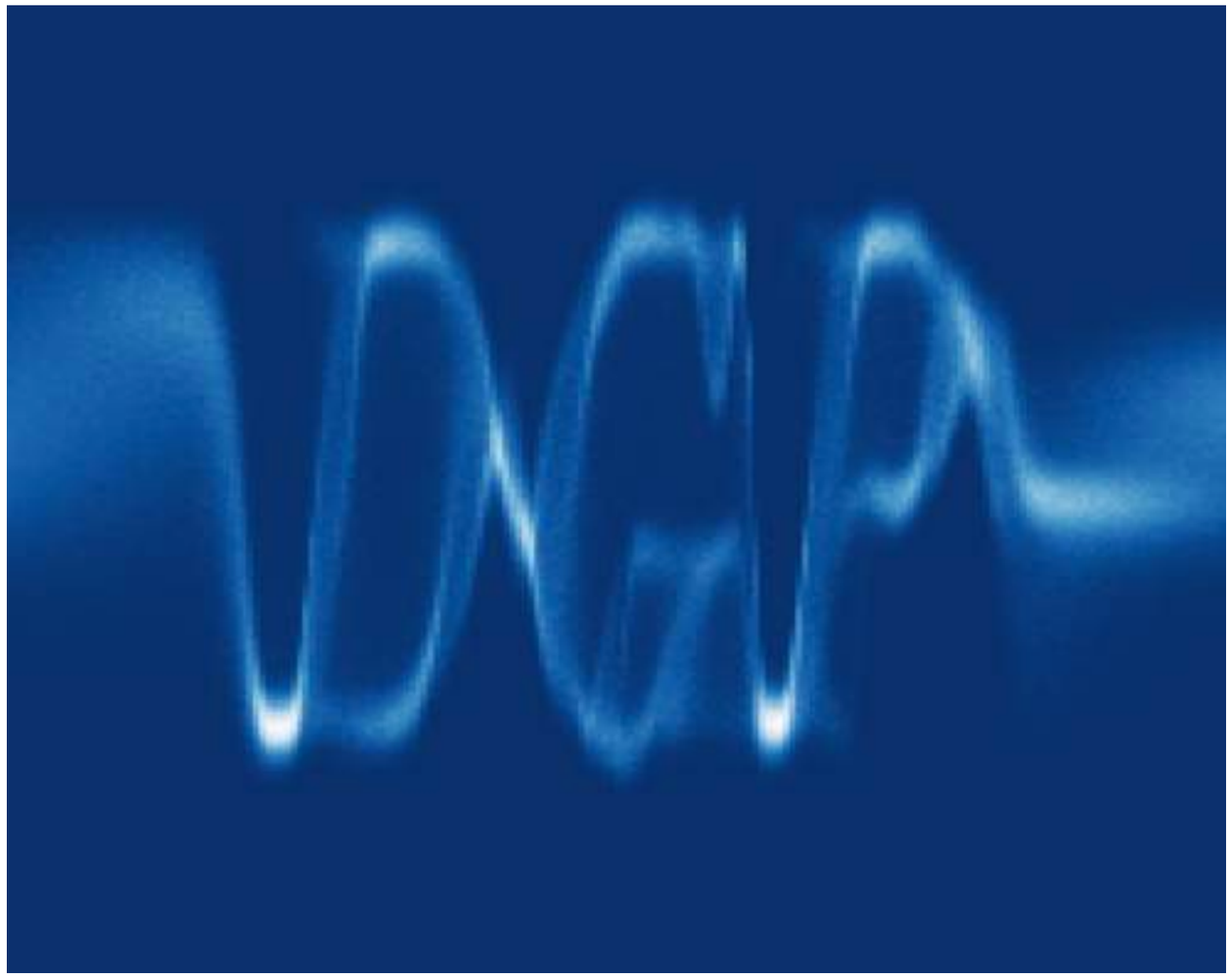
Summary:

- Deep GP gives a more useful prior than GP
- Need latent variables to get non-Gaussian marginals
- Variational inference appears to be effective in the noise-free case, and importance-weighted variational inference in the latent variable case
- Real data supports the hypothesis that both depth and latent variables are useful in practice

Further work:

- We haven’t broken into Deep Learning territory (yet)
- We’ve been thinking about scalability the wrong way
- We need more parameters in our variational distribution
- We need more specialised structures (e.g., convolutions)

Thanks for listening



Variational Inference

$$p(y) = \mathop{\mathbb{E}}_{f,g,w}\left[p(y|f,g,w)\frac{p(f)p(g)p(w)}{q(f)q(g)q(w)}\right]$$

$$\log p(y) \geq \sum_n (A_n - \mathrm{KL}_{w_n}) - \mathrm{KL}_f - \mathrm{KL}_g$$

$$A_n = \mathbb{E}_{f,g,w_n} \log p(y_n|f,g,w_n)$$

$$w_n = a_n + \epsilon_1\sqrt{b_n}$$

$$g([x_n,w_n]) = \mu_2([x_n,w_n]) + \epsilon_2\sqrt{k_2([x_n,w_n],[x_n,w_n])}$$

$$\epsilon_1,\epsilon_2 \sim N(0,1)$$

Naive importance weighting

$$p(y) = \mathop{\mathbb{E}}_{f,g,w} \frac{1}{K} \sum_{k=1}^{K} p(y|f,g,w^{(k)}) \frac{p(w^{(k)})}{q(w^{(k)})} \frac{p(f)p(g)}{q(f)q(g)}$$

$$\log p(y) \geq \sum_{n=1}^{N} B_n - \mathrm{KL}_f - \mathrm{KL}_g$$

$$B_n = \mathop{\mathbb{E}}_{f,g,w_n} \log \frac{1}{K} \sum_{k} p(y_n|f,g,w_n^{(k)}) \frac{p(w_n^{(k)})}{q(w_n^{(k)})}$$

Better importance weighting

$$p(y) = \mathop{\mathbb{E}}_{g,w} p(y|g,w)\frac{p(w)p(g)}{q(w)q(g)}$$

$$\log p(y|g,w) \geq \sum_n L_n(g,w_n) - \mathrm{KL}_f$$

$$L_n(g,w_n) = \mathbb{E}_f \log p(y_n|f,g,w_n)$$

$$p(y|g,w) \geq \exp\left[\sum_n L_n(g,w_n) - \mathrm{KL}_f\right]$$

$$p(y) \geq \mathop{\mathbb{E}}_{g,w} \exp\left[\sum_n L_n(g,w_n) - \mathrm{KL}_f\right]\frac{p(w)p(g)}{q(w)q(g)}$$

$$\log p(y) \geq \sum_n \mathbb{E}_g \underbrace{\log \mathbb{E}_w \frac{e^{L_n(g,w_n)}p(w_n)}{q(w_n)}}_{T_n(g)} - \mathrm{KL}_f - \mathrm{KL}_g$$

$$\log p(y) \geq \sum_n \mathbb{E}_g \underbrace{\log \mathbb{E}_w \frac{e^{L_n(g,w_n)} p(w_n)}{q(w_n)}}_{T_n(g)} - \mathrm{KL}_f - \mathrm{KL}_g$$

$$T_n(g) = \log \mathbb{E}_{w_n} \frac{1}{K} \sum_k \frac{e^{L_n(g,w_n^{(k)})} p(w_n^{(k)})}{q(w_n^{(k)})}$$

$$\sum_n \mathop{\mathbb{E}}_{g,w_n} \log \frac{1}{K} \sum_k \frac{e^{L_n(g,w_n^{(k)})} p(w_n^{(k)})}{q(w_n^{(k)})} - \mathrm{KL}_f - \mathrm{KL}_g$$

$$\log p(y) \geq \sum_{n=1}^{N} B_n - \mathrm{KL}_f - \mathrm{KL}_g$$

$$B_n = \mathop{\mathbb{E}}_{f,g,w_n} \log \frac{1}{K} \sum_k p(y_n | f, g, w_n^{(k)}) \frac{p(w_n^{(k)})}{q(w_n^{(k)})}$$