Fitting difficult GP models

James Hensman

PROWLER.io

October 2017

Overview

Motivation

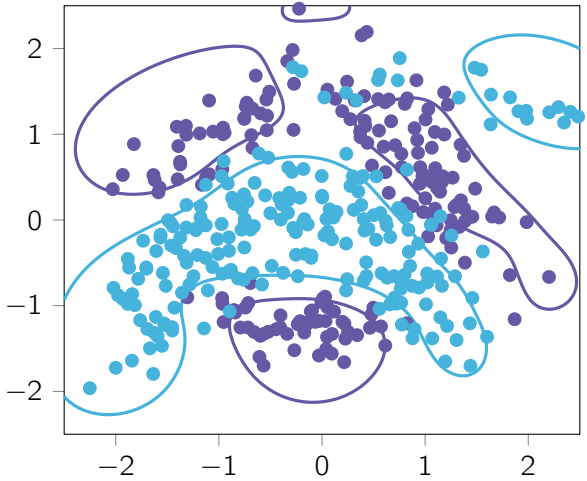
Variational Bayes

Variational Bayes with processes

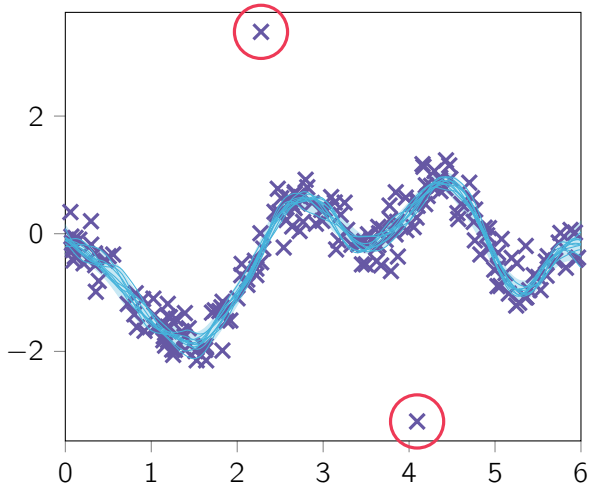
Bonus content: coping with image inputs

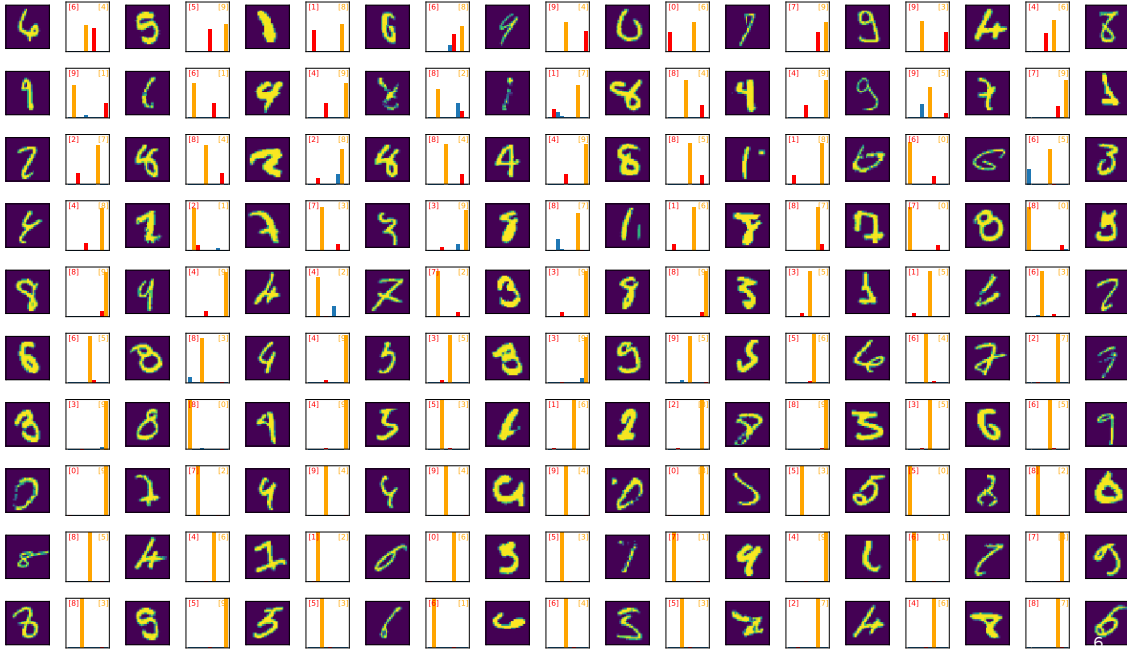
Motivation

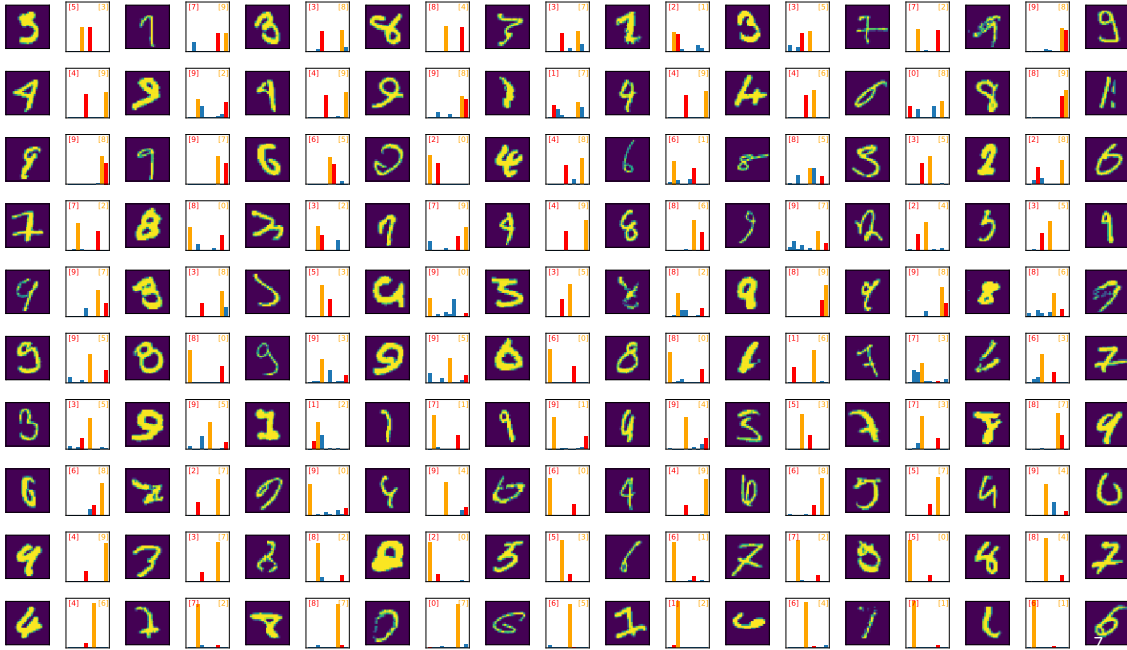
Non-Gaussian Gaussian processes



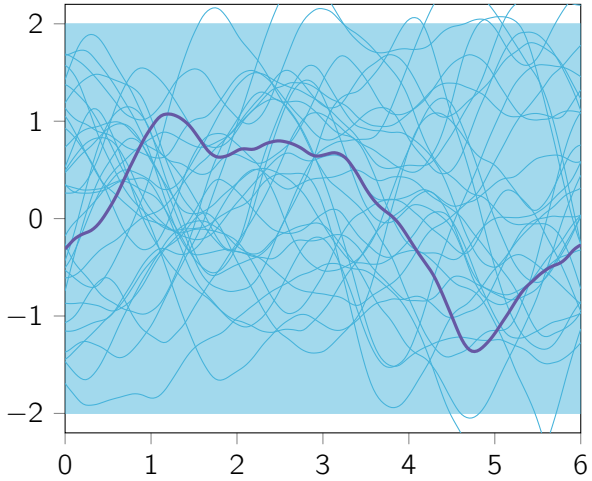
Non-Gaussian Gaussian processes

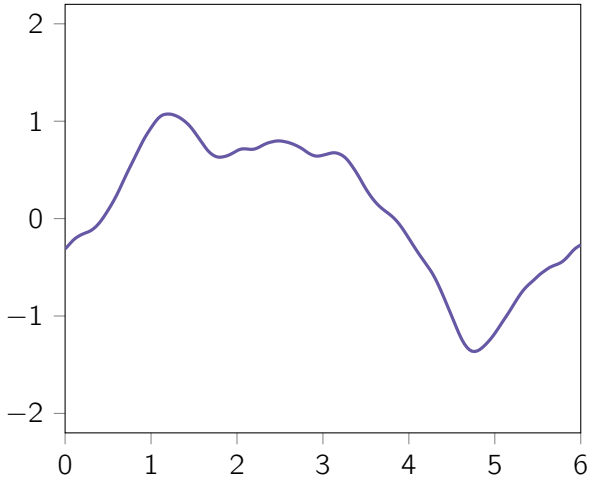


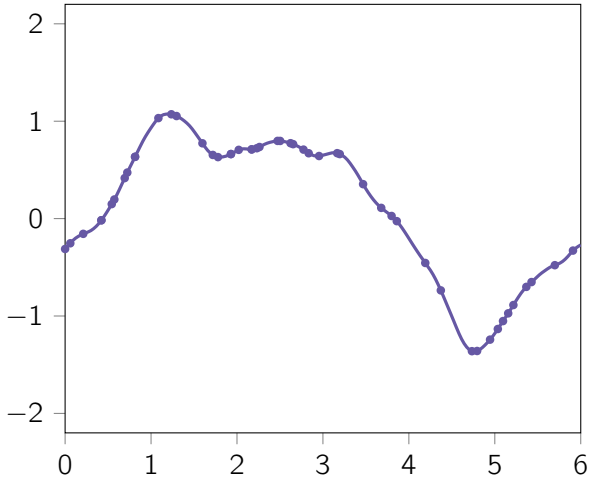


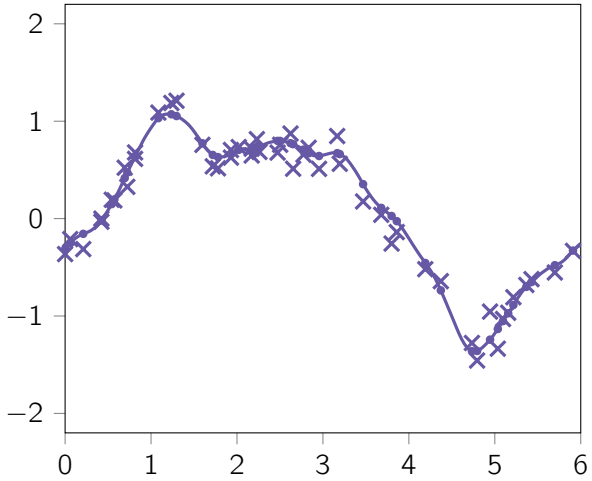


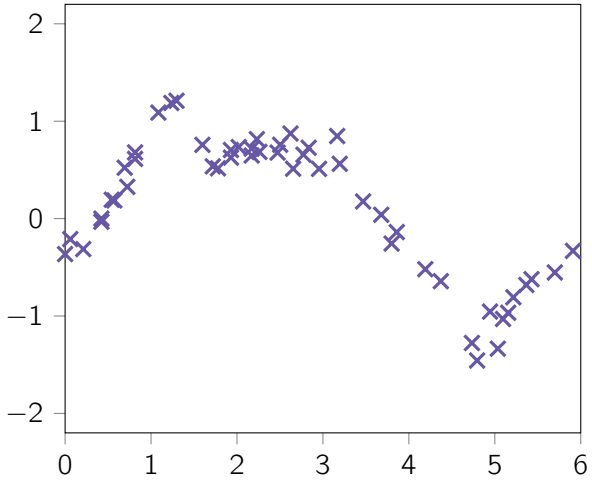
How are data generated in Gausian process regression?

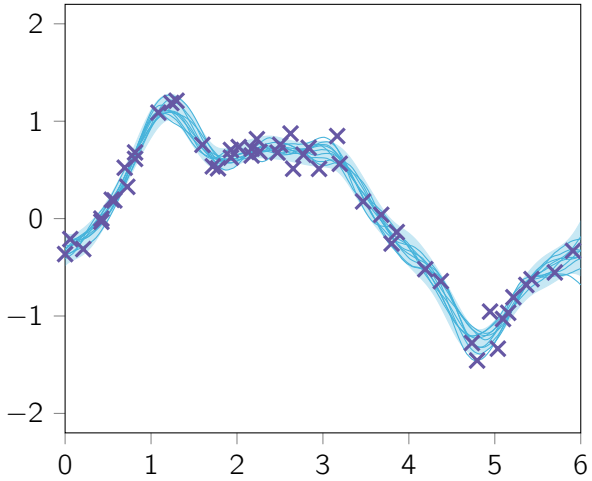

2
1
0
−1
−2
0
1
2
3
4
5
6



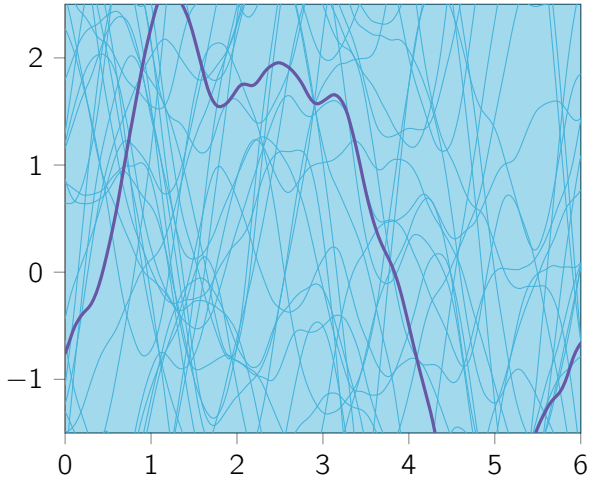

2
1
0
−1
−2
0
1
2
3
4
5
6

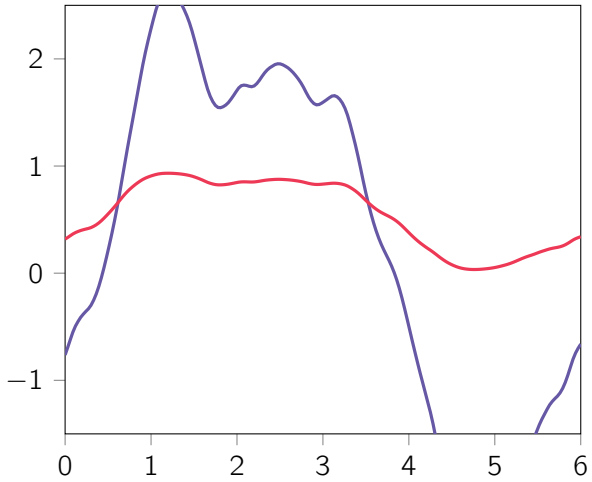

2
1
0
−1
−2
0
1
2
3
4
5
6

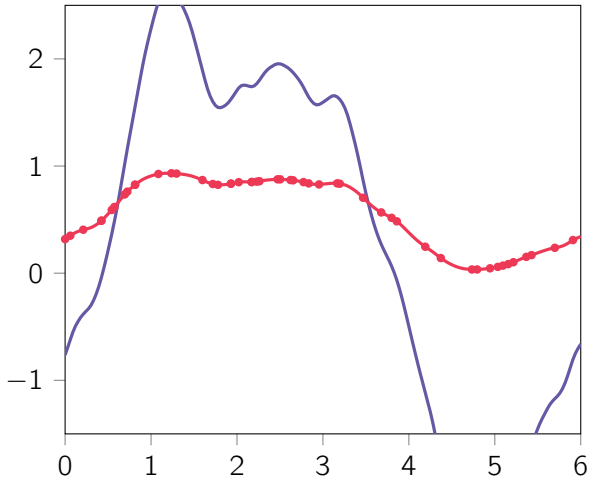

2
1
0
−1
−2
0
1
2
3
4
5
6

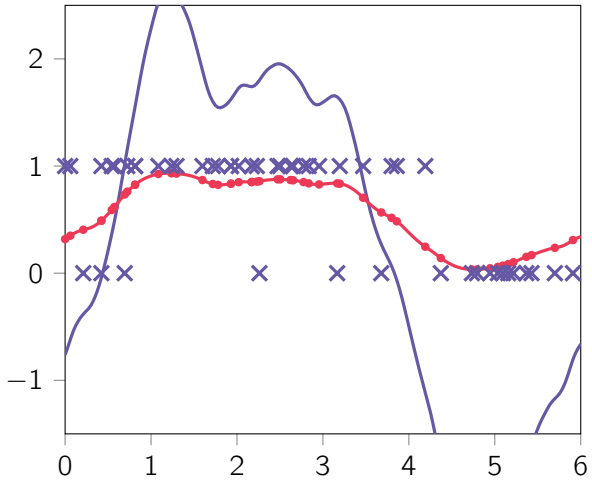

2
1
0
−1
−2
0
1
2
3
4
5
6

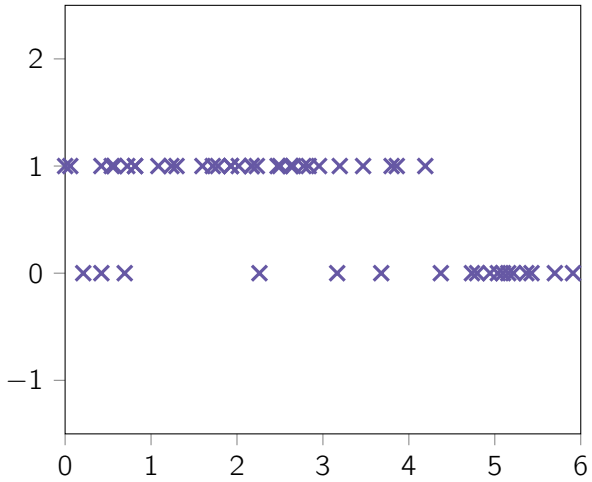
How are data generated in Gausian process classification?

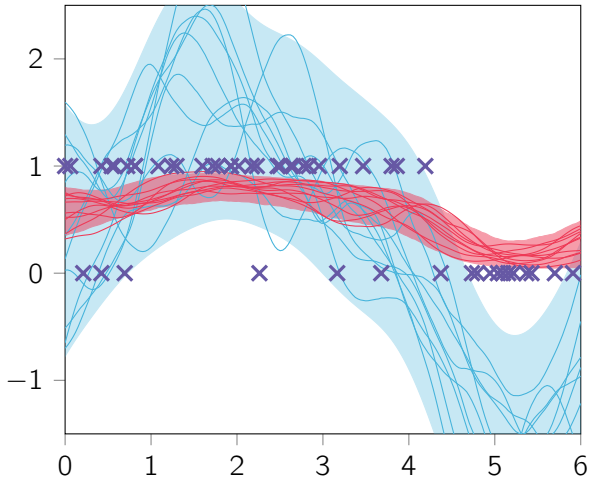

2
1
0
−1
0
1
2
3
4
5
6


2
1
0
−1
0
1
2
3
4
5
6


2
1
0
−1
0
1
2
3
4
5
6




2
1
0
−1
0
1
2
3
4
5
6


2
1
0
−1
0
1
2
3
4
5
6

Big Gaussian processes

A long-standing problem with GPs has been that they scale cubically with the number of data.

Recall

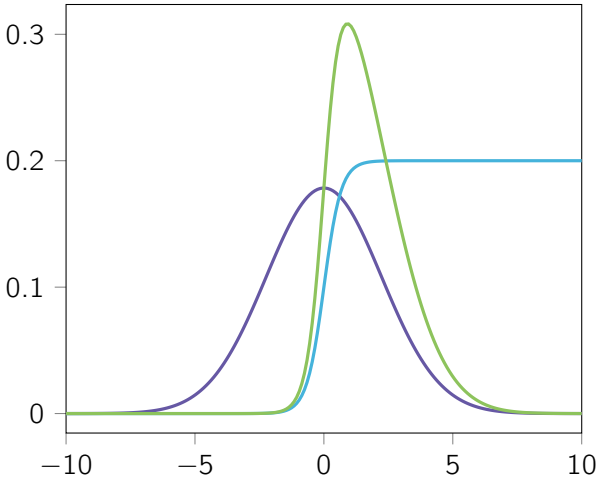
$$\underbrace{p(f(\cdot))}_{\text{GP prior}} = \underbrace{p(f(\cdot)\,|\,\mathbf{f})}_{\text{GP conditional}} \quad \underbrace{p(\mathbf{f})}_{\text{Gaussian dist}}$$

Variational Bayes

Variational Bayes

- In Gaussian process models, we want to do Bayesian inference on functions
- We're going to have to make approximations!
- VB is our method of choice for approximation.

$$p(x \mid y) = p(y \mid x)p(x)/p(y)$$



Worked example

$$
\begin{aligned}
p(x) &= \mathcal{N}(x \mid 0, \sigma_p^2) \\
p(y \mid x) &= \phi(x) \\
p(x \mid y) &= ? \\
p(y) &= ? \\
q(x) &= \mathcal{N}(x \mid m, \sigma_q^2)
\end{aligned}
$$

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

$$\mathrm{KL}[q(x)||p(x \mid y)] = \int q(x)[\log q(x) - \log p(x \mid y)]\, \mathrm{d}x$$

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

$$\mathrm{KL}[q(x)||p(x \mid y)] = \int q(x)[\log q(x) - \log p(x \mid y)]\, \mathrm{d}x$$

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

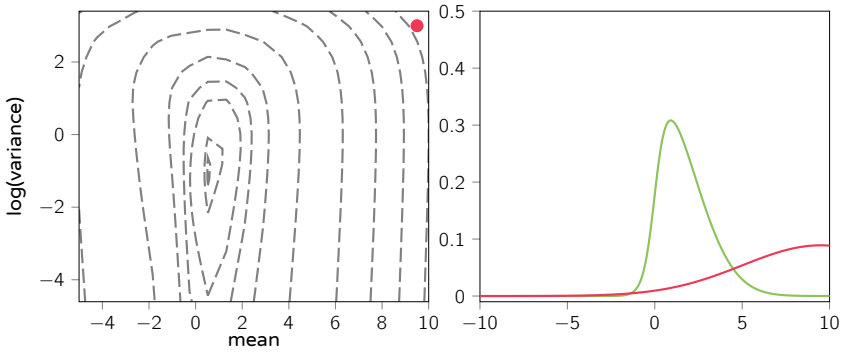
$$\mathrm{KL}[q(x)||p(x \mid y)] = \int q(x)[\log q(x) - \log p(y \mid x) - \log p(x)]\, \mathrm{d}x + \log p(y)$$

$$\mathrm{KL}[q(x)||p(x \mid y)] = \int q(x)[\log q(x) - \log p(x \mid y)]\, \mathrm{d}x$$

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

$$\mathrm{KL}[q(x)||p(x \mid y)] = \int q(x)[\log q(x) - \log p(y \mid x) - \log p(x)]\, \mathrm{d}x + \log p(y)$$

$$\mathrm{KL}[q(x)||p(x \mid y)] = -\mathrm{ELBO}(q(x)) + p(y)$$


log(variance)
2
0
−2
−4
−4
−2
0
2
4
6
8
10
mean
0.5
0.4
0.3
0.2
0.1
0
−10
−5
0
5
10

Variational Bayes with processes

Not so different from our toy example…

$$
\begin{aligned}
p(x) &= \mathcal{N}(x \mid 0, \sigma_p^2) && \rightarrow && p(f(\cdot)) &= \mathcal{GP}\big(0,\ k(\cdot,\cdot)\big) \\
p(y \mid x) &= \phi(x) && \rightarrow && p(\mathbf{y} \mid f(\cdot)) &= \prod_{i=1}^{n} p(y_i \mid f(x_i)) \\
p(x \mid y) &= ? && \rightarrow && p(f(\cdot) \mid \mathbf{y}) &= ? \\
p(y) &= ? && \rightarrow && p(\mathbf{y}) &= ? \\
q(x) &= \mathcal{N}(x \mid m, \sigma_q^2) && \rightarrow && q_{\mathsf{m},\mathsf{L},\mathsf{Z}}(f(\cdot)) &= \mathcal{GP}\big(\mu(\cdot),\ v(\cdot,\cdot)\big)
\end{aligned}
$$

The form of $q(f(\cdot))$

Recall the GP conditional equations:

$$p(f(\cdot)) = p(f(\cdot)\,|\,\mathbf{f})p(\mathbf{f})$$

with

$$p(f(\cdot)\,|\,\mathbf{f}) = \mathcal{GP}\big(k(\cdot, X)\mathbf{K}^{-1}\mathbf{f},\, k(\cdot,\cdot) - k(\cdot, X)\mathbf{K}^{-1}k(X,\cdot)\big)$$

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

where

$$\mathbf{f} = [f(x_i)]_{i=1}^{N}\,.$$

The form of $q(f(\cdot))$

Now choose some points **Z** so

$$p(f(\cdot)) = p(f(\cdot)\,|\,\mathbf{u})p(\mathbf{u})$$

with

$$p(f(\cdot)\,|\,\mathbf{u}) = \mathcal{GP}\big(k(\cdot,\mathbf{Z})\mathbf{K}^{-1}\mathbf{u},\, k(\cdot,\cdot) - k(\cdot,\mathbf{Z})\mathbf{K}^{-1}k(\mathbf{Z},\cdot)\big)$$

$$p(\mathbf{u}) = \mathcal{N}\big(0,\mathbf{K}\big)$$

where

$$\mathbf{u} = [f(x_i)]_{i=1}^{M}\,.$$

Now let

$$q(f(\cdot) \mid \mathbf{u}) = p(f(\cdot) \mid \mathbf{u})$$

$$q(\mathbf{u}) = \mathcal{N}(\mathbf{u} \mid \mathbf{m}, \mathbf{L}\mathbf{L}^\top)$$

Now let

$$q(f(\cdot)\,|\,\mathbf{u}) = p(f(\cdot)\,|\,\mathbf{u})$$
$$q(\mathbf{u}) = \mathcal{N}(\mathbf{u}\,|\,\mathbf{m}, \mathbf{L}\mathbf{L}^\top)$$

so that

$$q(f(\cdot)) = \mathcal{GP}\big(k(\cdot, \mathbf{Z})\mathbf{K}^{-1}\mathbf{m},\, k(\cdot,\cdot) - k(\cdot, \mathbf{Z})\mathbf{K}^{-1}[\mathbf{K} - \mathbf{L}\mathbf{L}^\top]\mathbf{K}^{-1}k(\mathbf{Z}, \cdot)\big)$$

Now let

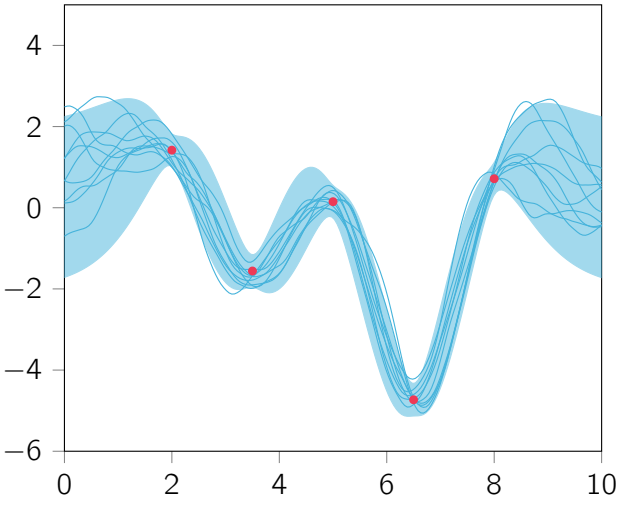
$$q(f(\cdot)\,|\,\mathbf{u}) = p(f(\cdot)\,|\,\mathbf{u})$$
$$q(\mathbf{u}) = \mathcal{N}(\mathbf{u}\,|\,\mathbf{m}, \mathbf{L}\mathbf{L}^\top)$$

so that

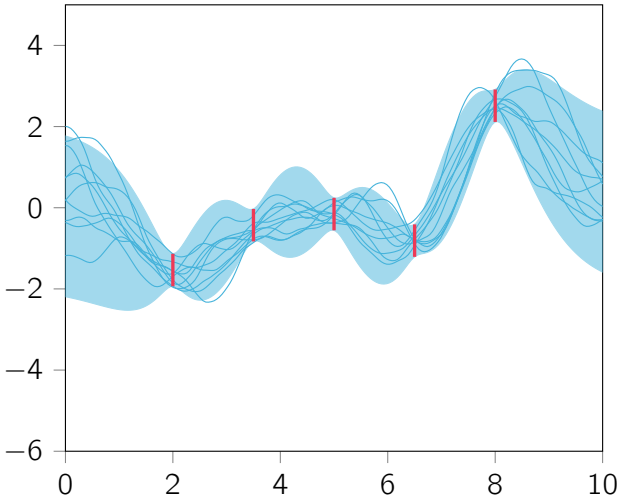
$$q(f(\cdot)) = \mathcal{GP}\big(k(\cdot, \mathbf{Z})\mathbf{K}^{-1}\mathbf{m},\, k(\cdot,\cdot) - k(\cdot, \mathbf{Z})\mathbf{K}^{-1}[\mathbf{K} - \mathbf{L}\mathbf{L}^\top]\mathbf{K}^{-1}k(\mathbf{Z}, \cdot)\big)$$

now: optimize $\mathrm{KL}[q(f(\cdot))||p(f(\cdot)\,|\,\mathbf{Y}]$ w.r.t. $\mathbf{Z}$, $\mathbf{m}$, $\mathbf{L}$.

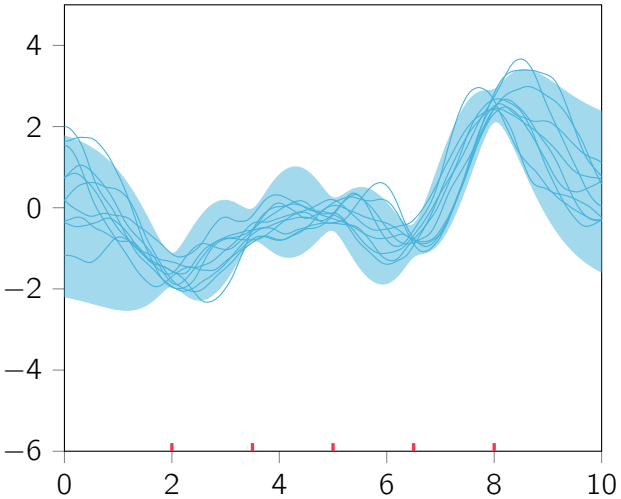
What happens when we adjust **m**?

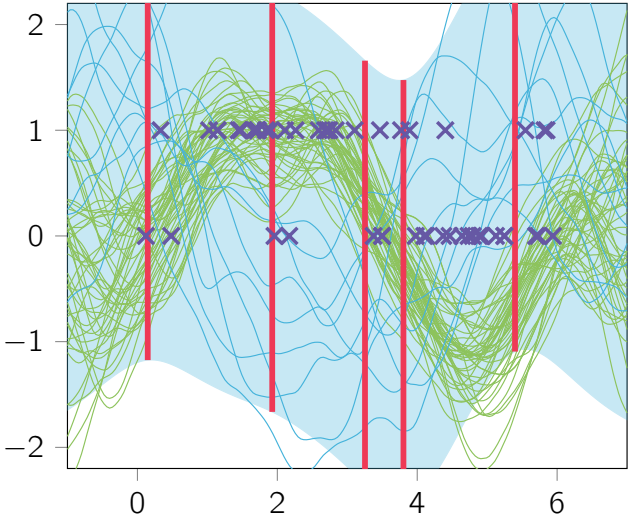


What happens when we adjust **L**?



What happens when we adjust **Z**?





The ELBO for GP models

$$\text{ELBO} = \int q(f(\cdot)) \left[\log p(\mathbf{y} \mid f(\cdot)) + \log \frac{p(f(\cdot))}{q(f(\cdot))} \right] \mathrm{d}f(\cdot)$$

- We can compute the term $p(f(\cdot))/q(f(\cdot))$ because of our careful construction of q.
- We can compute the integral of $p(\mathbf{y} \mid f(\cdot))$ since we only need the marginals of $q(f(\cdot))$.
- We can minibatch if needed!

Questions

?

- What about parameters of the kernel?
- How many **Z** should I pick?
- How reasonable is the Gaussian assumption?
- What if the likelihood does not factorize?
- What if the likelihood is Gaussian?

Bonus content: coping with image inputs

Convolutional Gaussian Processes

Mark van der Wilk
Department of Engineering
University of Cambridge, UK
mv310@cam.ac.uk

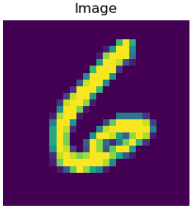
Carl Edward Rasmussen
Department of Engineering
University of Cambridge, UK
cer54@cam.ac.uk

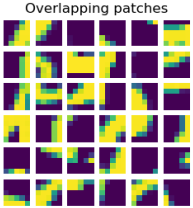
James Hensman
prowler.io
Cambridge, UK
james@prowler.io

Abstract

We present a practical way of introducing convolutional structure into Gaussian processes, making them more suited to high-dimensional inputs like images. The main contribution of our work is the construction of an inter-domain inducing point approximation that is well-tailored to the convolutional kernel. This allows us to gain the generalisation benefit of a convolutional kernel, together with fast but accurate posterior inference. We investigate several variations of the convolutional kernel, and apply it to MNIST and CIFAR-10, where we obtain significant improvements over existing Gaussian process models. We also show how the marginal likelihood can be used to find an optimal weighting between convolutional and RBF kernels to further improve performance. This illustration of the usefulness of the marginal likelihood may help automate discovering architectures in larger models.

Convolutional Gaussian Processes

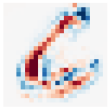


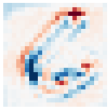


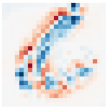
$$f(\mathbf{x}) = \sum_p w_p g\left(\mathbf{x}^{[p]}\right)$$

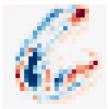
$$k(\mathbf{x}, \mathbf{x}') = \sum_p \sum_{p'} k_g\left(\mathbf{x}^{[p]}, \mathbf{x}'^{[p']}\right)$$

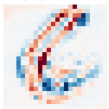
$$k(\mathbf{x}, \mathbf{z}') = \sum_p k_g\left(\mathbf{x}^{[p]}, \mathbf{z}\right)$$

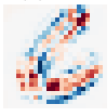


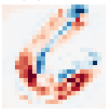


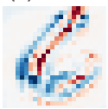













PROWLER.io
AI for Autonomous
Decision Making