Gaussian Processes for Big Data

James Hensman

Overview

Motivation

Sparse Gaussian Processes

Stochastic Variational Inference

Examples

Overview

Motivation

Inference in a GP has the following demands:

$$\begin{aligned} \text{Complexity:} \quad & \mathcal{O}(n^3) \\ \text{Storage:} \quad & \mathcal{O}(n^2) \end{aligned}$$

Inference in a *sparse* GP has the following demands:

$$\begin{aligned} \text{Complexity:} \quad & \mathcal{O}(nm^2) \\ \text{Storage:} \quad & \mathcal{O}(nm) \end{aligned}$$

where we get to pick m!

Still not good enough!

Big Data

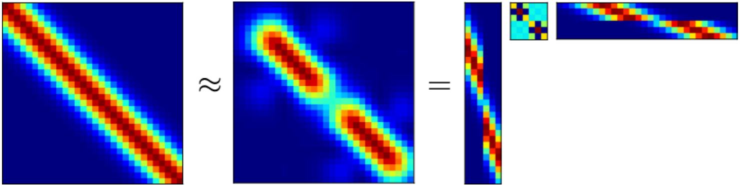
- In parametric models, stochastic optimisation is used.
- This allows for application to Big Data.

This work

- Show how to use Stochastic Variational Inference in GPs
- Stochastic optimisation scheme: each step requires $\mathcal{O}(m^3)$

Overview

Computational savings



$$\mathbf{K}_{nn} \approx \mathbf{Q}_{nn} = \mathbf{K}_{nm}\mathbf{K}_{mm}^{-1}\mathbf{K}_{mn}$$

Instead of inverting $\mathbf{K}_{nn}$, we make a low rank (or Nyström) approximation, and invert $\mathbf{K}_{mm}$ instead.

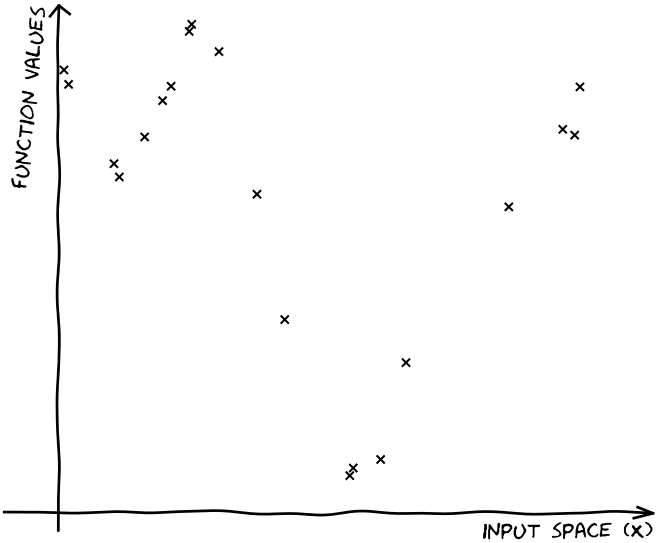
Information capture

Everything we want to do with a GP involves marginalising $\mathbf{f}$

- Predictions
- Marginal likelihood
- Estimating covariance parameters

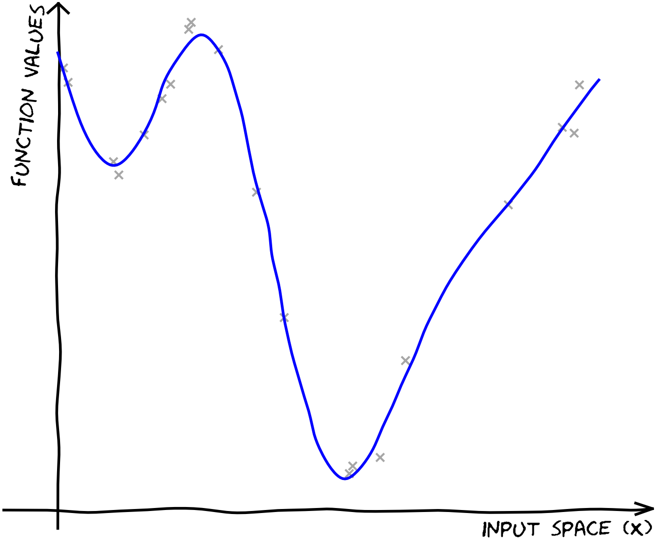
The posterior of $\mathbf{f}$ is the central object. This means inverting $\mathbf{K}_{nn}$.

$\mathbf{X}, \mathbf{y}$



$\mathbf{X}, \mathbf{y}$

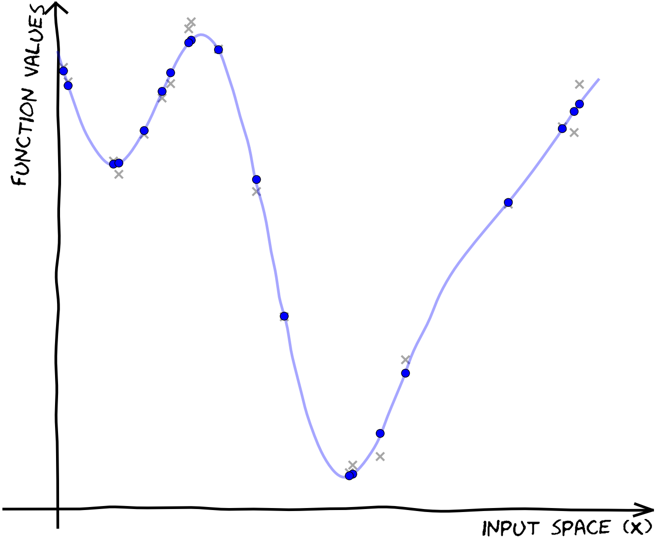
$f(\mathbf{x}) \sim \mathcal{GP}$



$\mathbf{X}, \mathbf{y}$

$f(\mathbf{x}) \sim \mathcal{GP}$

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

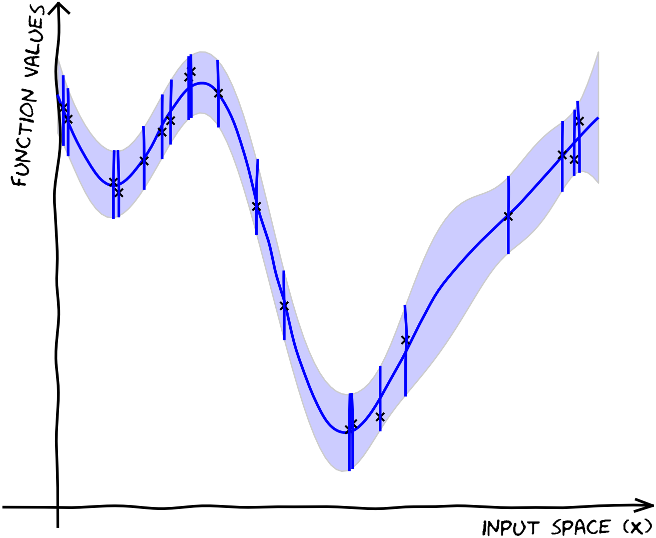


$\mathbf{X}, \mathbf{y}$

$f(\mathbf{x}) \sim \mathcal{GP}$

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

$p(\mathbf{f} | \mathbf{y}, \mathbf{X})$

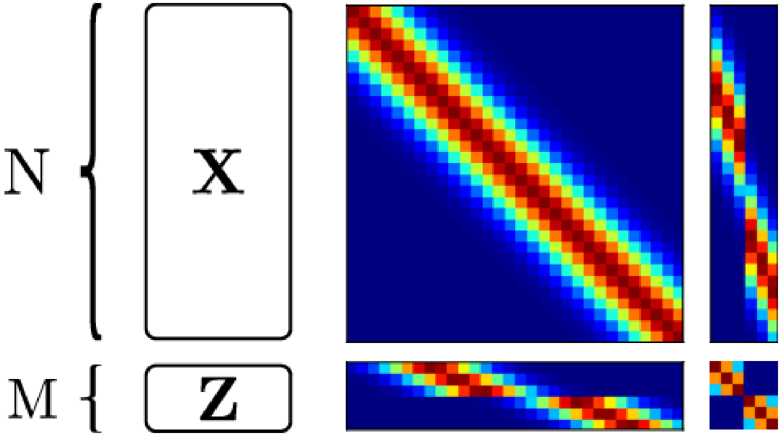


Introducing $\mathbf{u}$

Take and extra M points on the function, $\mathbf{u} = f(\mathbf{Z})$.

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

Introducing $\mathbf{u}$



Introducing $\mathbf{u}$

Take and extra M points on the function, $\mathbf{u} = f(\mathbf{Z})$.

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

$$\begin{aligned}
p(\mathbf{y}\,|\,\mathbf{f}) &= \mathcal{N}\left(\mathbf{y}|\mathbf{f}, \sigma^2\mathbf{I}\right) \\
p(\mathbf{f}\,|\,\mathbf{u}) &= \mathcal{N}\left(\mathbf{f}|\mathbf{K}_{nm}\mathbf{K}_{mm1}\mathbf{u}, \widetilde{\mathbf{K}}\right) \\
p(\mathbf{u}) &= \mathcal{N}\left(\mathbf{u}|\mathbf{0}, \mathbf{K}_{mm}\right)
\end{aligned}$$

$\mathbf{X}, \mathbf{y}$

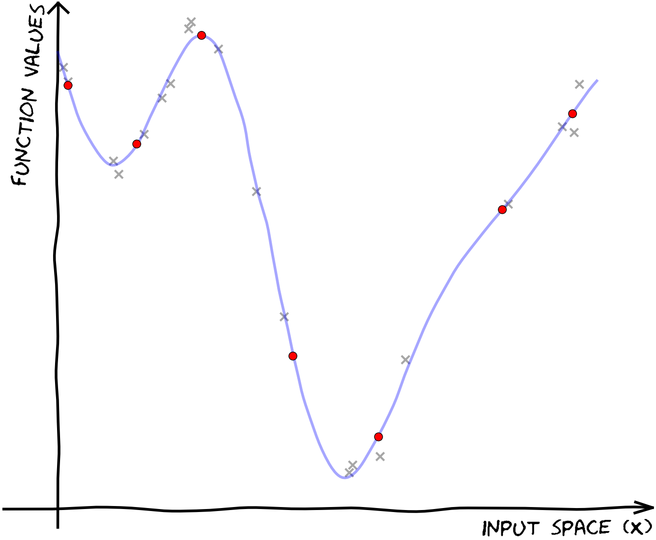
$f(\mathbf{x}) \sim \mathcal{GP}$

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

$p(\mathbf{f} \mid \mathbf{y}, \mathbf{X})$

$\mathbf{Z}, \mathbf{u}$

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



$\mathbf{X}, \mathbf{y}$

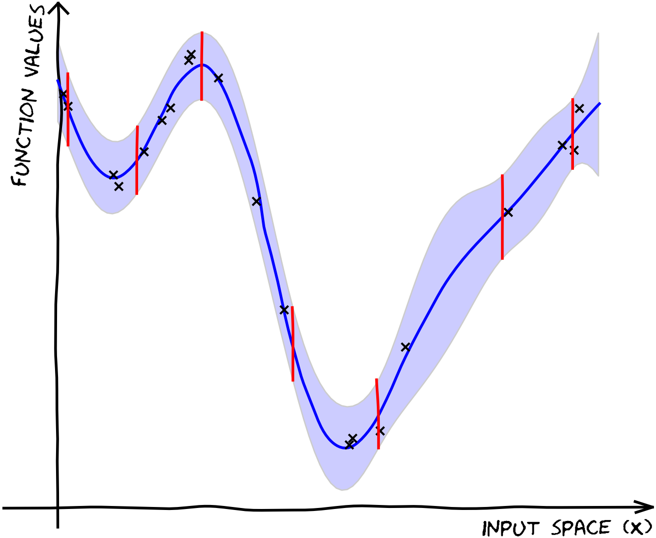
$f(\mathbf{x}) \sim \mathcal{GP}$

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

$p(\mathbf{f} \mid \mathbf{y}, \mathbf{X})$

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

$\widetilde{p}(\mathbf{u} \mid \mathbf{y}, \mathbf{X})$



The alternative posterior

Instead of doing

$$p(\mathbf{f}\,|\,\mathbf{y},\mathbf{X}) = \frac{p(\mathbf{y}\,|\,\mathbf{f})p(\mathbf{f}\,|\,\mathbf{X})}{\int p(\mathbf{y}\,|\,\mathbf{f})p(\mathbf{f}\,|\,\mathbf{X})\mathrm{d}\mathbf{f}}$$

We'll do

$$p(\mathbf{u}\,|\,\mathbf{y},\mathbf{Z}) = \frac{p(\mathbf{y}\,|\,\mathbf{u})p(\mathbf{u}\,|\,\mathbf{Z})}{\int p(\mathbf{y}\,|\,\mathbf{u})p(\mathbf{u}\,|\,\mathbf{Z})\mathrm{d}\mathbf{u}}$$

The alternative posterior

Instead of doing

$$p(\mathbf{f}\,|\,\mathbf{y},\mathbf{X}) = \frac{p(\mathbf{y}\,|\,\mathbf{f})p(\mathbf{f}\,|\,\mathbf{X})}{\int p(\mathbf{y}\,|\,\mathbf{f})p(\mathbf{f}\,|\,\mathbf{X})\mathrm{d}\mathbf{f}}$$

We'll do

$$p(\mathbf{u}\,|\,\mathbf{y},\mathbf{Z}) = \frac{p(\mathbf{y}\,|\,\mathbf{u})p(\mathbf{u}\,|\,\mathbf{Z})}{\int p(\mathbf{y}\,|\,\mathbf{u})p(\mathbf{u}\,|\,\mathbf{Z})\mathrm{d}\mathbf{u}}$$

but $p(\mathbf{y}\,|\,\mathbf{u})$ involves inverting $\mathbf{K}_{nn}$

Variational marginalisation of $\mathbf{f}$

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

Variational marginalisation of $\mathbf{f}$

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

$$\ln p(\mathbf{y}\,|\,\mathbf{u}) = \ln p(\mathbf{y}\,|\,\mathbf{f}) + \ln \frac{p(\mathbf{f}\,|\,\mathbf{u})}{p(\mathbf{f}\,|\,\mathbf{y},\mathbf{u})}$$

Variational marginalisation of $\mathbf{f}$

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

$$\ln p(\mathbf{y}\,|\,\mathbf{u}) = \ln p(\mathbf{y}\,|\,\mathbf{f}) + \ln \frac{p(\mathbf{f}\,|\,\mathbf{u})}{p(\mathbf{f}\,|\,\mathbf{y},\mathbf{u})}$$

$$\ln p(\mathbf{y}\,|\,\mathbf{u}) = \mathbb{E}_{p(\mathbf{f}\,|\,\mathbf{u})}\Big[\ln p(\mathbf{y}\,|\,\mathbf{f})\Big] + \mathbb{E}_{p(\mathbf{f}\,|\,\mathbf{u})}\Big[\ln \frac{p(\mathbf{f}\,|\,\mathbf{u})}{p(\mathbf{f}\,|\,\mathbf{y},\mathbf{u})}\Big]$$

Variational marginalisation of $\mathbf{f}$

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

$$\ln p(\mathbf{y}\,|\,\mathbf{u}) = \ln p(\mathbf{y}\,|\,\mathbf{f}) + \ln \frac{p(\mathbf{f}\,|\,\mathbf{u})}{p(\mathbf{f}\,|\,\mathbf{y},\mathbf{u})}$$

$$\ln p(\mathbf{y}\,|\,\mathbf{u}) = \mathbb{E}_{p(\mathbf{f}\,|\,\mathbf{u})}\Big[\ln p(\mathbf{y}\,|\,\mathbf{f})\Big] + \mathbb{E}_{p(\mathbf{f}\,|\,\mathbf{u})}\Big[\ln \frac{p(\mathbf{f}\,|\,\mathbf{u})}{p(\mathbf{f}\,|\,\mathbf{y},\mathbf{u})}\Big]$$

$$\ln p(\mathbf{y}\,|\,\mathbf{u}) = \widetilde{p}(\mathbf{y}\,|\,\mathbf{u}) + \mathrm{KL}[p(\mathbf{f}|\mathbf{u})\|p(\mathbf{f}\,|\,\mathbf{y},\mathbf{u})]$$

No inversion of $\mathbf{K}_{nn}$ required

An approximate likelihood

$$\widetilde{p}(\mathbf{y} \mid \mathbf{u}) = \prod_{i=1}^{n} \mathcal{N}\left(\mathbf{y}_i | \mathbf{k}_{mn}^{\top} \mathbf{K}_{mm}^{-1} \mathbf{u}, \sigma^2\right) \exp\left\{-\frac{1}{2\sigma^2}\left(k_{nn} - \mathbf{k}_{mn}^{\top} \mathbf{K}_{mm}^{-1} \mathbf{k}_{mn}\right)\right\}$$

A straightforward likelihood approximation, and a penalty term

Now we can marginalise $\mathbf{u}$

$$\widetilde{p}(\mathbf{u} \mid \mathbf{y}, \mathbf{Z}) = \frac{\widetilde{p}(\mathbf{y} \mid \mathbf{u}) p(\mathbf{u} \mid \mathbf{Z})}{\int \widetilde{p}(\mathbf{y} \mid \mathbf{u}) p(\mathbf{u} \mid \mathbf{Z}) \mathrm{d}\mathbf{u}}$$

- Computing the (approximate) posterior costs $O(nm^2)$
- We also get a lower bound of the marginal likelihood
- This is the standard variational sparse GP [Titsias, 2009].

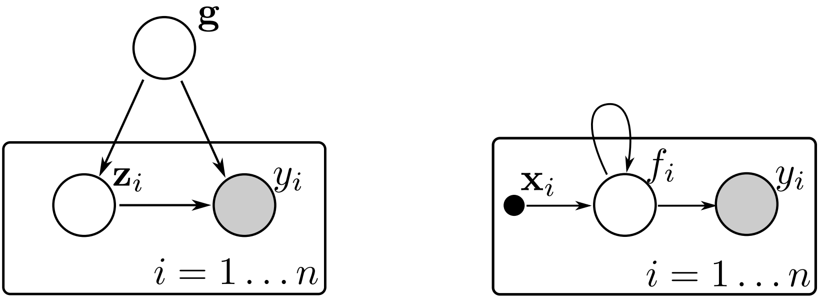
Overview

Variational Bayes

- Approximate the true posterior distribution with a simpler one.
- Usually assume factorisation in the approximation
- Iterative 'update' procedure (like EM)
- Can be seen as a coordinate-wise steepest ascent method

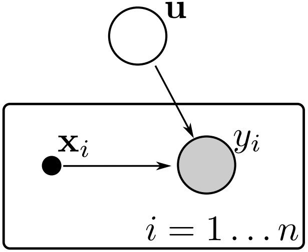
Stochastic Variational Inference

- Combine the ideas of stochastic optimisation with Variational inference
- example: apply Latent Dirichlet allocation to project Gutenberg
- Can apply variational techniques to Big Data
- How could this work in GPs?



Maintain the factorisation!

- The variational marginalisation of $\mathbf{f}$ introduced factorisation across the datapoints (conditioned on $\mathbf{u}$)
- Marginalising $\mathbf{u}$ re-introdcuced dependencies between the data
- Solution: a variational treatment of $\mathbf{u}$



$$\log p(\mathbf{y} \,|\, \mathbf{X}) \geq \left\langle \mathcal{L}_1 + \log p(\mathbf{u}) - \log q(\mathbf{u}) \right\rangle_{q(\mathbf{u})} \triangleq \mathcal{L}_3. \tag{1}$$

$$\begin{aligned} \mathcal{L}_3 = \sum_{i=1}^{n} \Big\{ & \log \mathcal{N}\left(y_i | \mathbf{k}_{mn}^{\top} \mathbf{K}_{mm}^{-1} \mathbf{m}, \beta^{-1}\right) \\ & - \frac{1}{2} \beta \widetilde{k}_{i,i} - \frac{1}{2} \mathrm{tr}\left(\mathbf{S} \mathbf{\Lambda}_i\right) \Big\} \\ & - \mathrm{KL}\left(q(\mathbf{u}) \,\|\, p(\mathbf{u})\right) \end{aligned} \tag{2}$$

Optimisation

The variational objective $\mathcal{L}_3$ is a function of

- the parameters of the covariance function
- the parameters of $q(\mathbf{u})$
- the inducing inputs, $\mathbf{Z}$

Strategy: set $\mathbf{Z}$. Take the data in small minibatches, take stochastic gradient steps in the covariance function parameters, stochastic *natural* gradient steps in the parameters of $q(\mathbf{u})$.

Natural Gradients

$$\widetilde{\mathbf{g}}(\boldsymbol{\theta}) = G(\boldsymbol{\theta})^{-1}\frac{\partial \mathcal{L}_3}{\partial \boldsymbol{\theta}} = \frac{\partial \mathcal{L}_3}{\partial \boldsymbol{\eta}}.$$

$$\begin{aligned}
\boldsymbol{\theta}_{2(t+1)} &= -\frac{1}{2}\mathbf{S}1_{(t+1)} \\
&= -\frac{1}{2}\mathbf{S}1_{(t)} + \ell\left(-\frac{1}{2}\boldsymbol{\Lambda} + \frac{1}{2}\mathbf{S}1_{(t)}\right), \\
\boldsymbol{\theta}_{1(t+1)} &= \mathbf{S}1_{(t+1)}\mathbf{m}_{(t+1)} \\
&= \mathbf{S}1_{(t)}\mathbf{m}_{(t)} + \ell\left(\beta\mathbf{K}_{mm}1\mathbf{K}_{mn}\mathbf{y} - \mathbf{S}1_{(t)}\mathbf{m}_{(t)}\right),
\end{aligned}$$

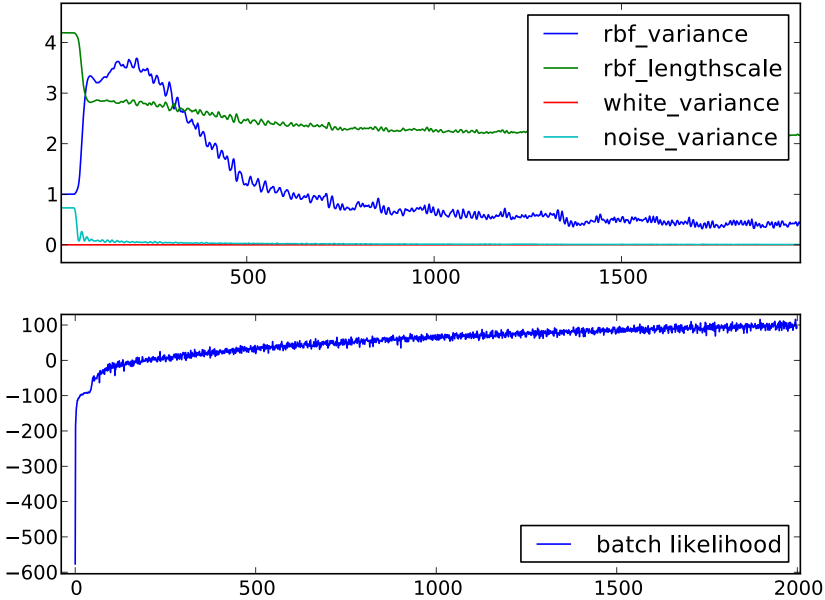
Overview

Motivation

Sparse Gaussian Processes

Stochastic Variational Inference

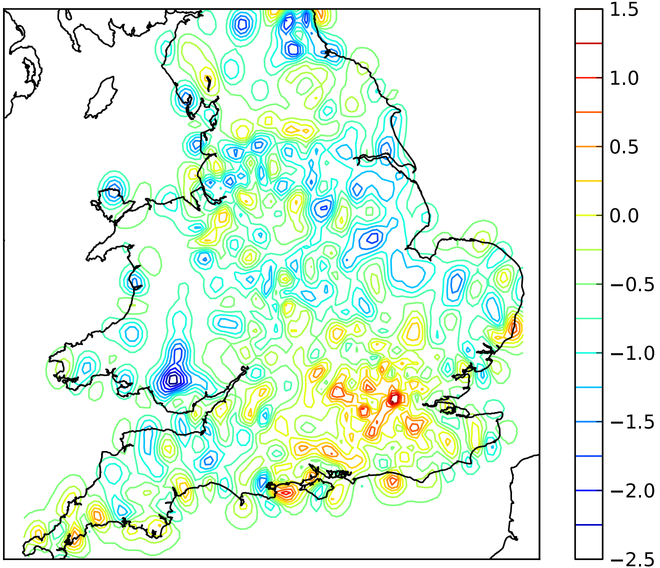
Examples

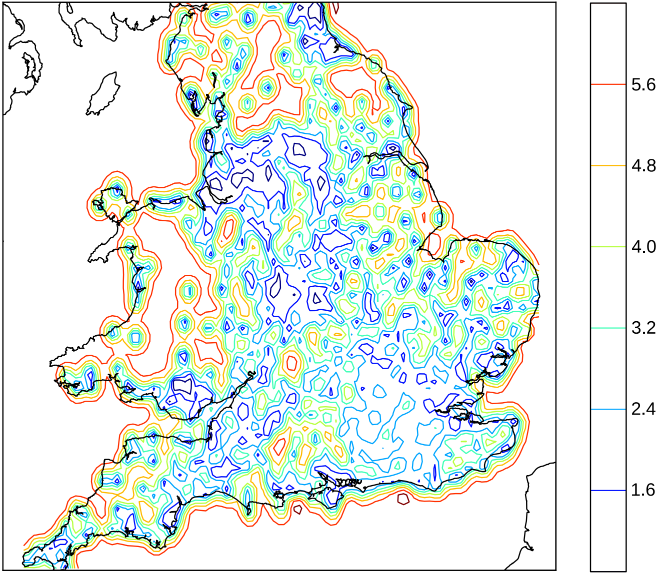


rbf_variance
rbf_lengthscale
white_variance
noise_variance
batch likelihood

UK apartment prices

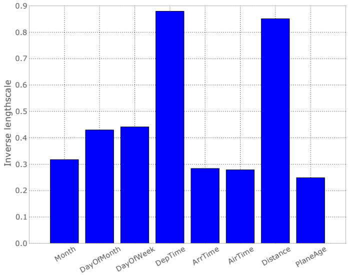
- Monthly price paid data for February to October 2012 (England and Wales)
- from `http://data.gov.uk/dataset/land-registry-monthly-price-paid-data/`
- 75,000 entries
- Cross referenced against a postcode database to get lattitude and longitude
- Regressed the normalised logarithm of the apartment prices

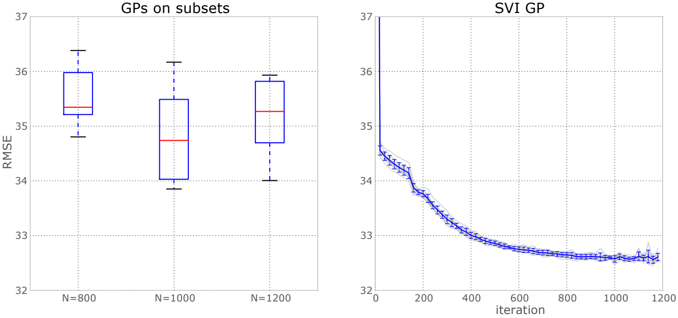

1.5
1.0
0.5
0.0
−0.5
−1.0
−1.5
−2.0
−2.5


5.6
4.8
4.0
3.2
2.4
1.6

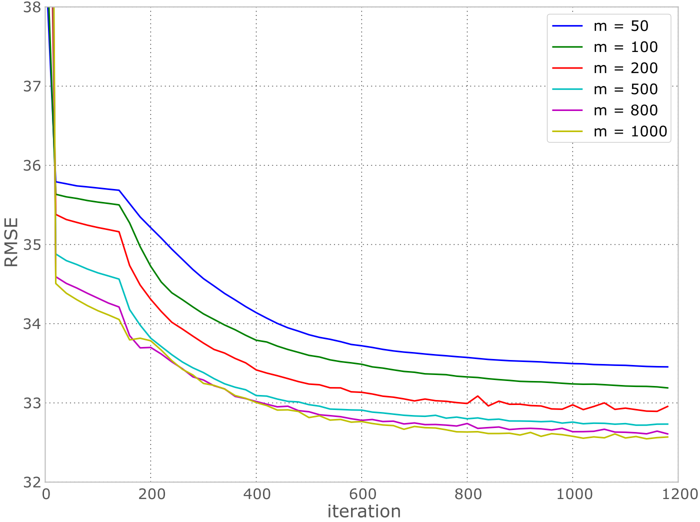
Airline data

- Flight delays for every commercial flight in the USA from January to April 2008.
- Average delay was 30 minutes.
- We randomly selected 800,000 datapoints (we have limited memory!)
- 700,000 train, 100,000 test





GPs on subsets
RMSE
37
36
35
34
33
32
N=800
N=1000
N=1200
SVI GP
0
200
400
600
800
1000
1200
iteration



m = 50
m = 100
m = 200
m = 500
m = 800
m = 1000
RMSE
iteration
38
37
36
35
34
33
32
0
200
400
600
800
1000
1200

Download the code!

`github.com/SheffieldML/GPy`

Cite our paper!

Hensman, Fusi and Lawrence,
Gaussian Processes for Big Data
Proceedings of UAI 2013

Michalis K. Titsias. Variational learning of inducing variables in sparse Gaussian processes. In David van Dyk and Max Welling, editors, *Proceedings of the Twelfth International Workshop on Artificial Intelligence and Statistics*, volume 5, pages 567–574, Clearwater Beach, FL, 16-18 April 2009. JMLR W&CP 5.