

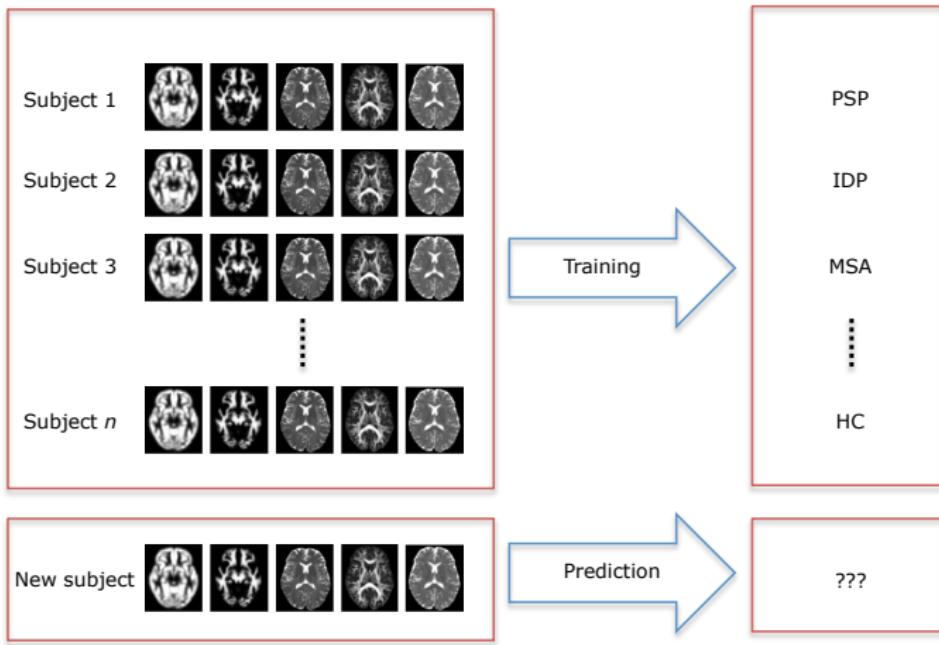
Unbiased computations for tractable and scalable learning of Gaussian processes

Maurizio Filippone

Department of Data Science, EURECOM
maurizio.filippone@eurecom.fr

April 13th, 2016

Motivating Application



HC - Healthy control

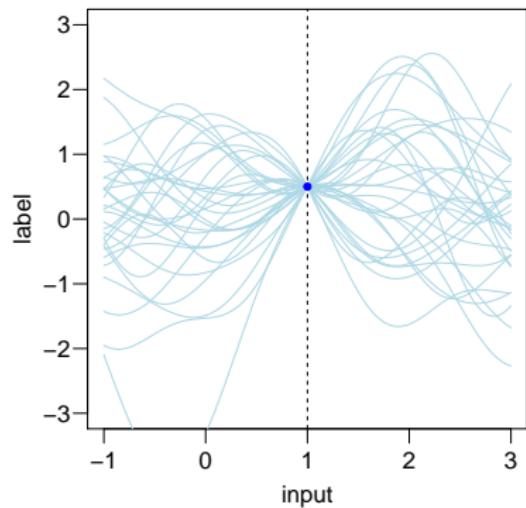
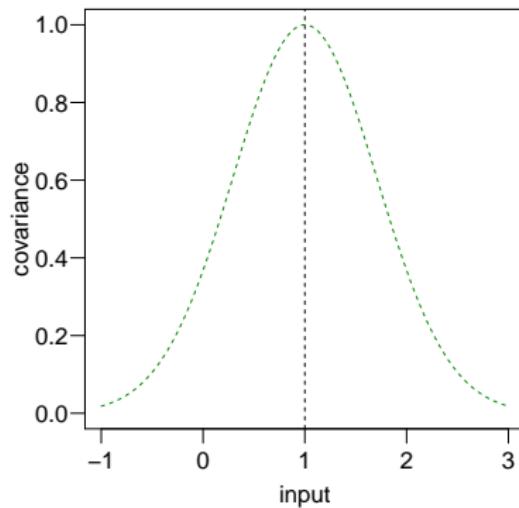
MSA - Multiple system atrophy

PSP - Progressive Supranuclear Palsy

IDP - Idiopathic Parkinson's disease

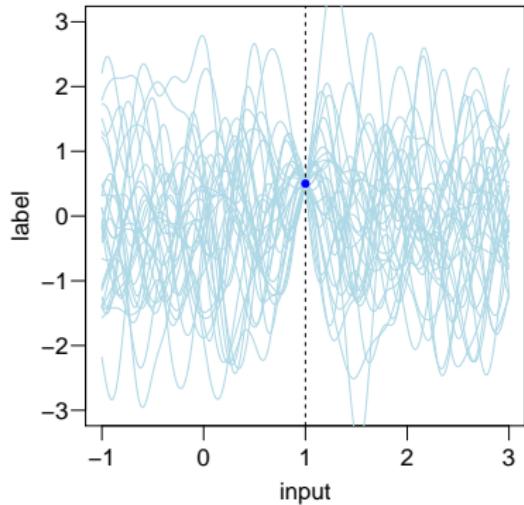
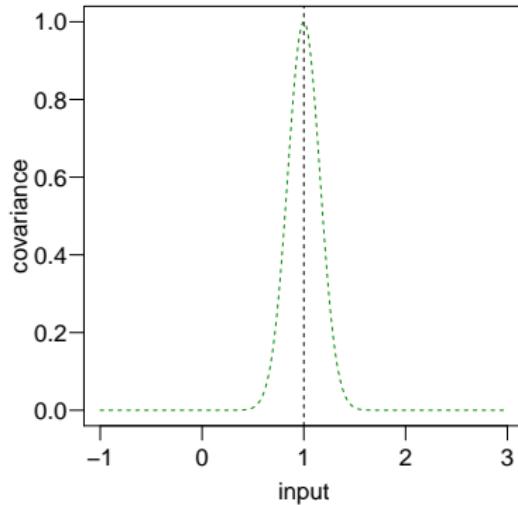
Gaussian Processes

- Gaussians with distance dependent covariance



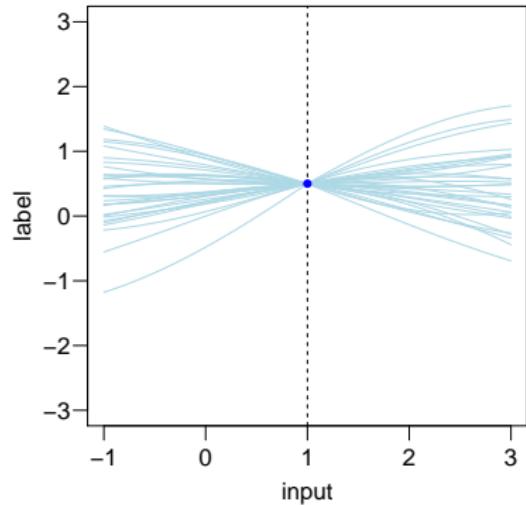
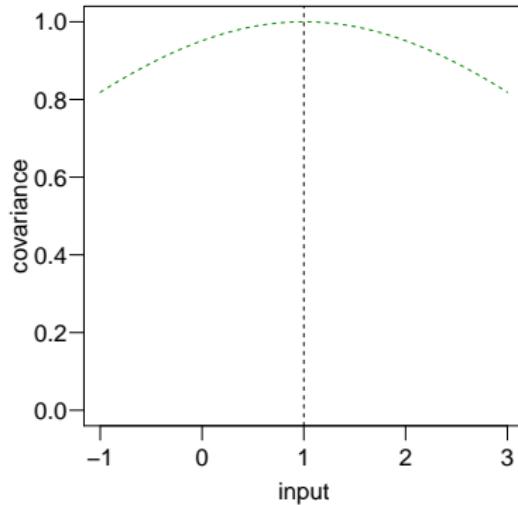
Gaussian Processes

- Gaussians with distance dependent covariance



Gaussian Processes

- Gaussians with distance dependent covariance



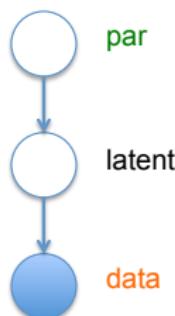
Gaussian Process Models

- Class of hierarchical models

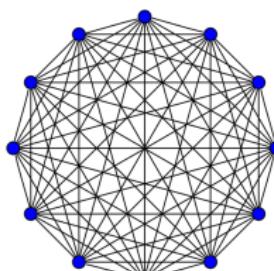
$$p(\mathbf{y}|\text{latent})$$

$$p(\text{latent}|\text{par})$$

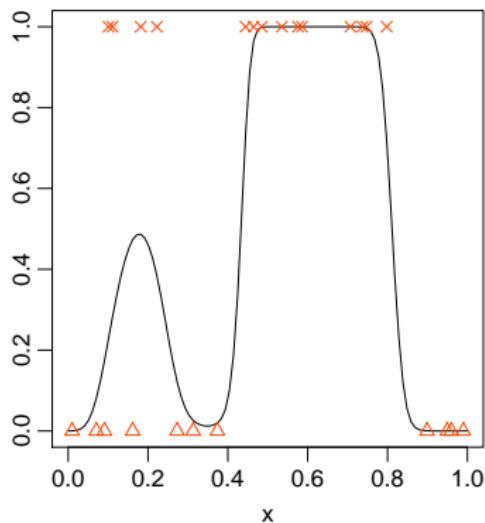
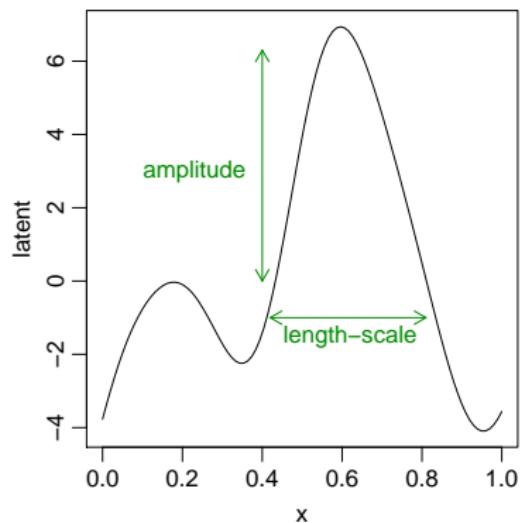
$$p(\text{par})$$



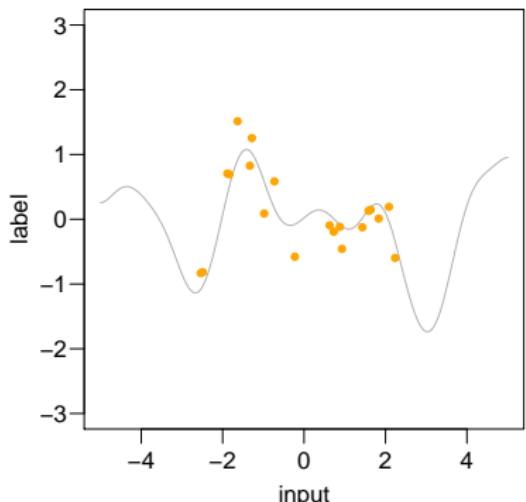
$p(\text{latent}|\text{par}) = \text{Gaussian Process}$



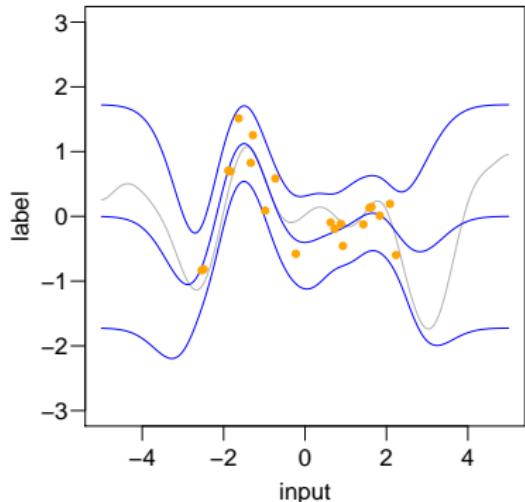
Gaussian Process Models - Classification example



Gaussian Processes - Regression example



$$K = \begin{matrix} & \star & \star \\ \star & \begin{matrix} \text{orange} & \text{blue} \\ \text{blue} & \text{orange} \end{matrix} & \begin{matrix} \text{orange} & \star \\ \text{blue} & \text{blue} \end{matrix} \\ & \star & \star \\ \star & \begin{matrix} \text{orange} & \text{blue} \\ \text{blue} & \text{orange} \end{matrix} & \begin{matrix} \text{orange} & \star \\ \text{blue} & \text{blue} \end{matrix} \\ & \star & \star \\ \star & \begin{matrix} \text{orange} & \text{blue} \\ \text{blue} & \text{orange} \end{matrix} & \begin{matrix} \text{orange} & \star \\ \text{blue} & \text{blue} \end{matrix} \end{matrix}$$



$$K = \begin{matrix} n & & n \\ & \text{orange} & \\ & \text{orange} & \end{matrix}$$

Multiclass classification with multiple sources

- Multiclass classification based on GPs

$p(\text{disease} = c | \text{sources})$ = unknown function

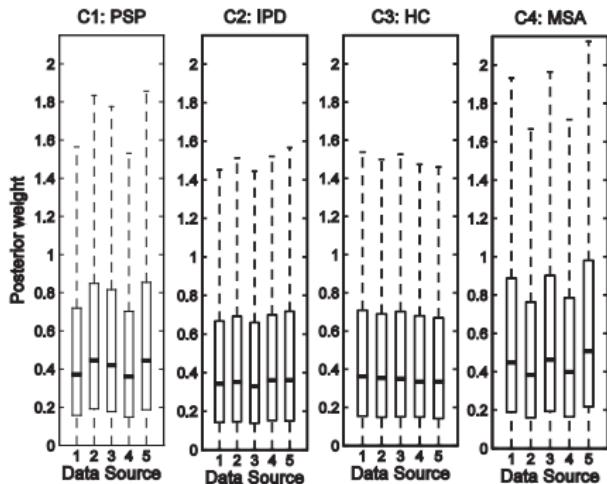
- unknown function modeled using GPs
- Covariance based on source-dependent covariances S_k

$$\sum_{k=1}^K w_{ck} S_k(\text{subject}_i, \text{subject}_j)$$

Filippone et al., AoAS, 2012

Parkinson syndromes data - multi source

Method	Accuracy
GP classifier	0.598
SimpleMKL	0.418

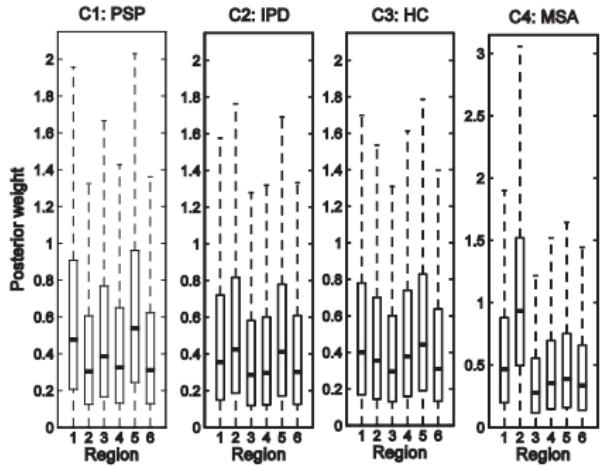


Analysis of brain regions

- ① brainstem
- ② bilateral cerebellum
- ③ bilateral caudate
- ④ bilateral middle occipital gyrus
- ⑤ bilateral putamen
- ⑥ all other regions

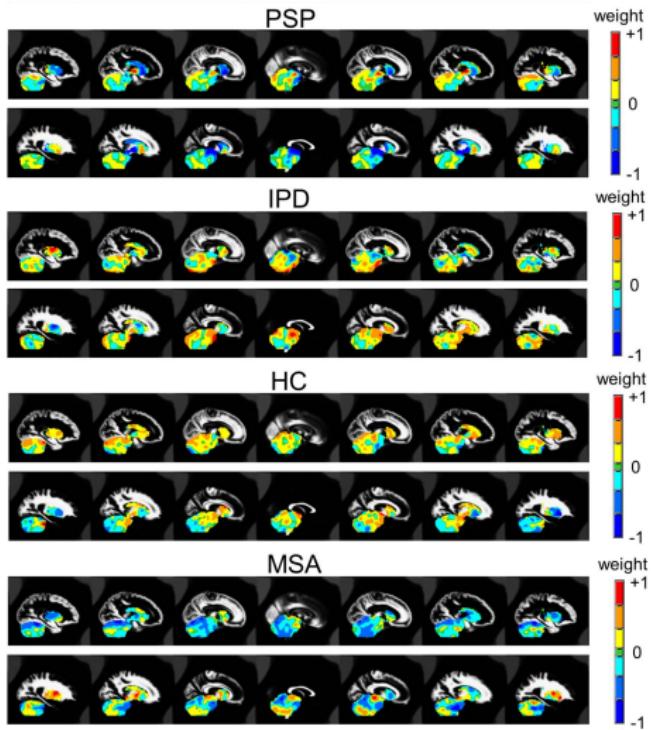
Multiclass classification with multiple regions

Method	Accuracy
GP classifier	0.614
SimpleMKL	0.229



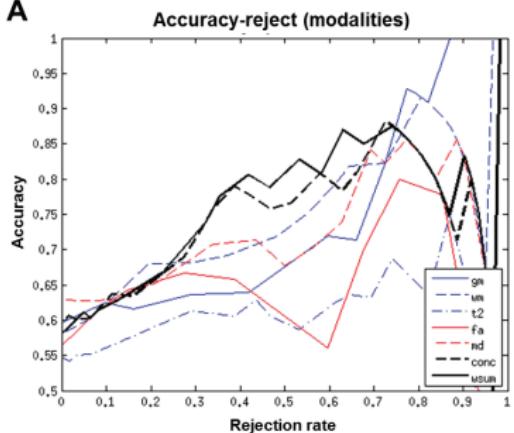
Filippone et al., AoAS, 2012

Brain Maps

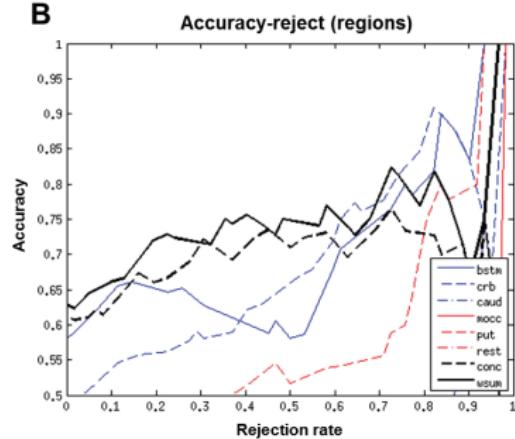


Reject Option

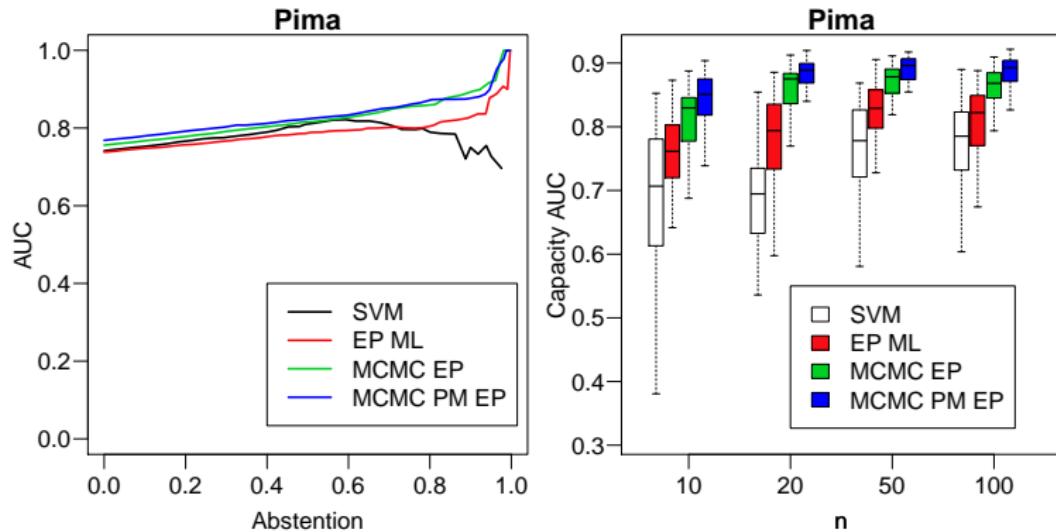
A



B

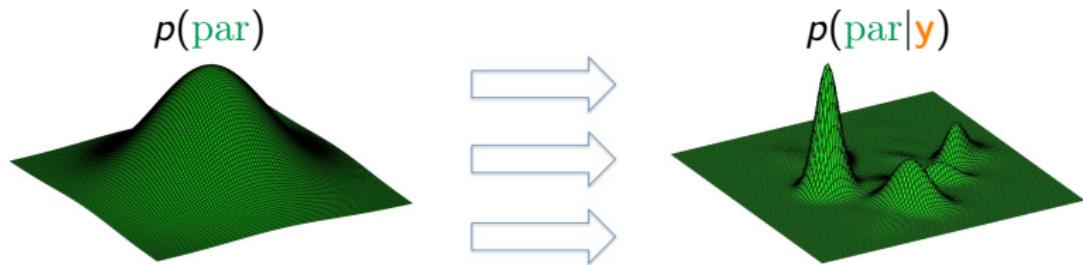


Importance of characterizing posterior over par



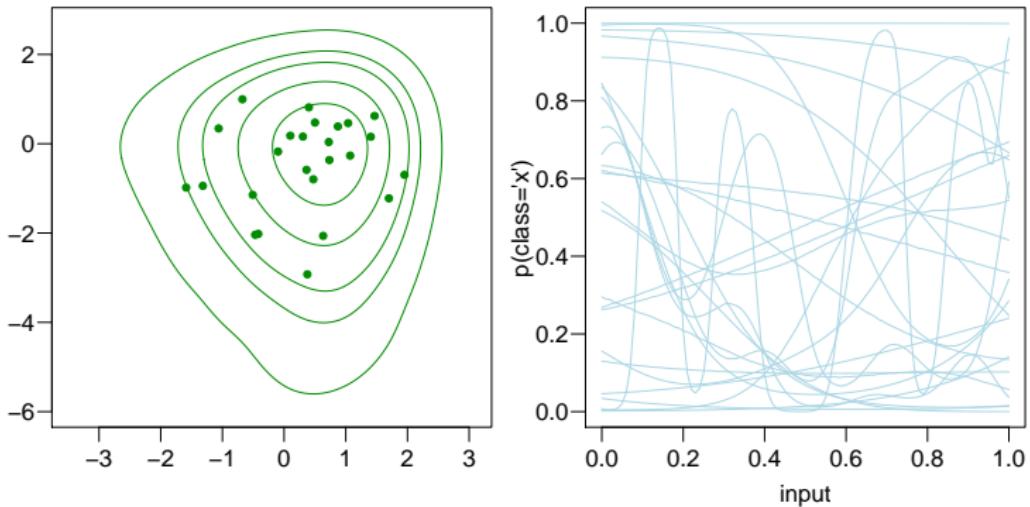
Bayesian Inference

- Inputs = X Labels = y
- $K = K(X, \text{par})$

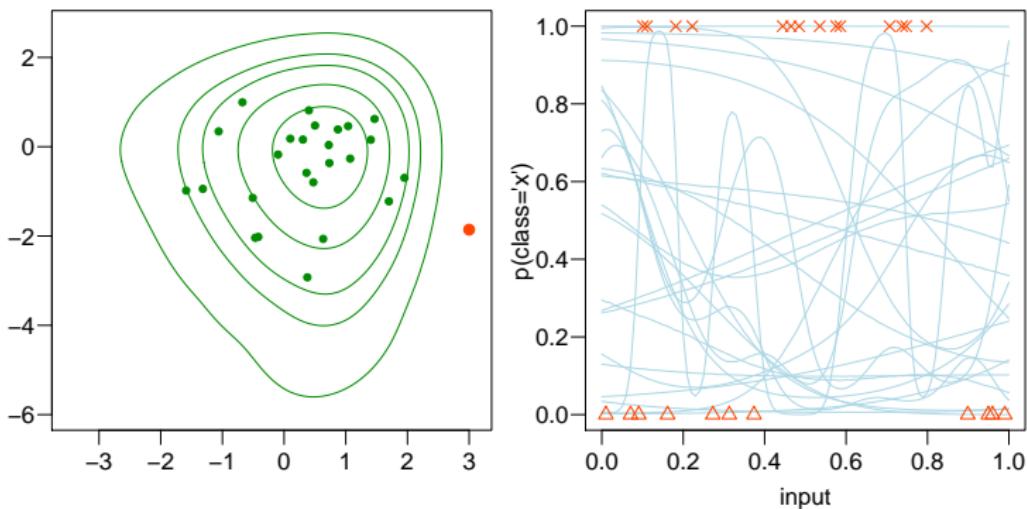


$$p(\text{par}|\mathbf{y}) = \frac{p(\mathbf{y}|\text{par})p(\text{par})}{\int p(\mathbf{y}|\text{par})p(\text{par})d\text{par}}$$

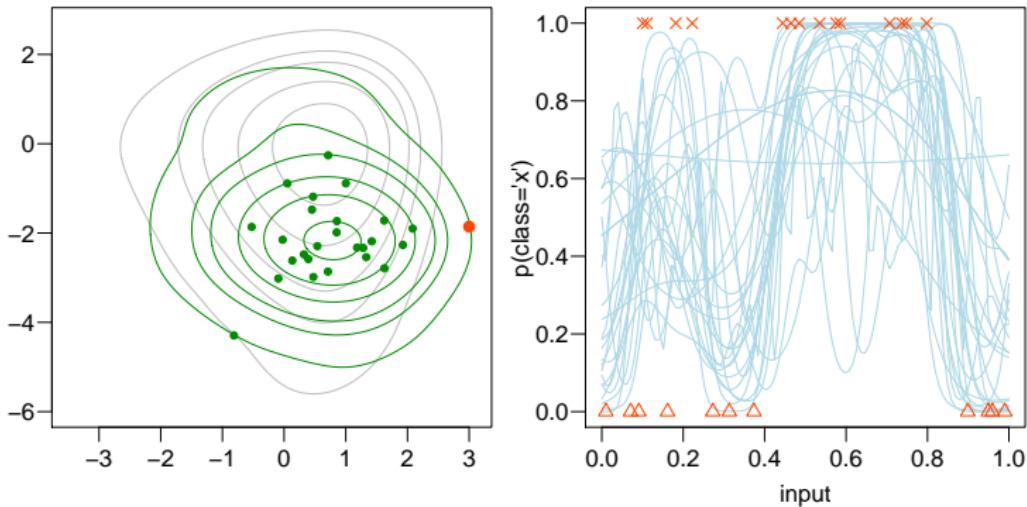
Bayesian Inference - Prior



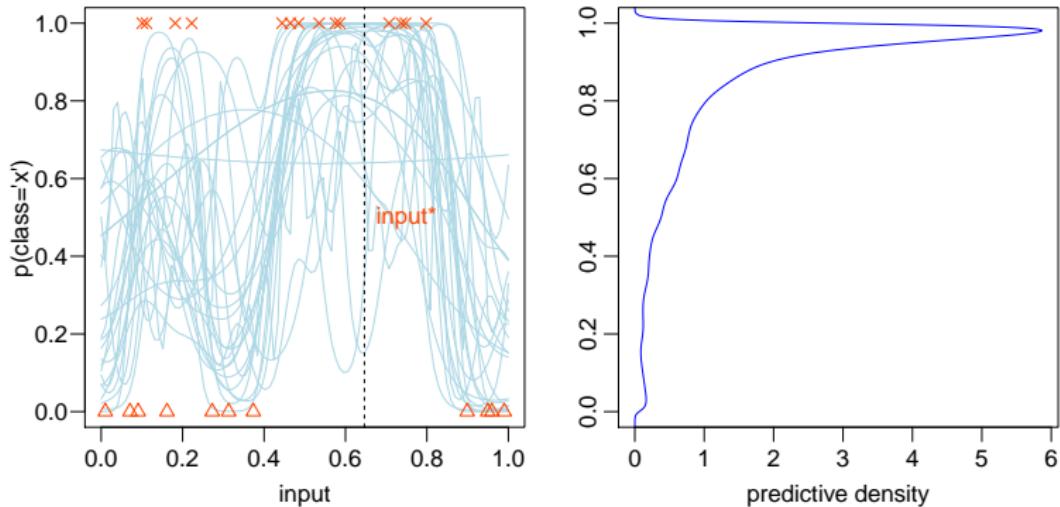
Bayesian Inference - Data



Bayesian Inference - Posterior



Bayesian Inference and Predictions



Bayesian Inference and Predictions

- Predictions for new data

$$p(\mathbf{y}_* | \mathbf{y}) = \int p(\mathbf{y}_* | \text{par}) p(\text{par} | \mathbf{y}) d\text{par}$$

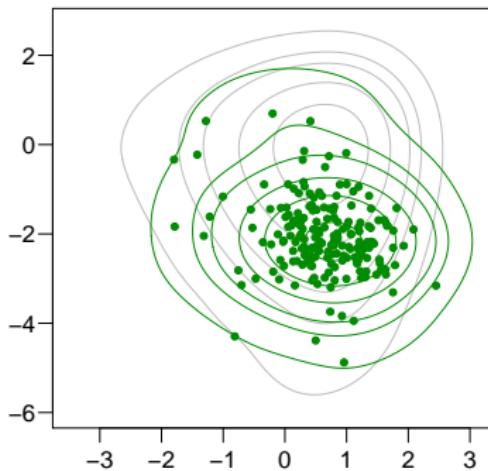
- Monte Carlo integration:

$$\int p(\mathbf{y}_* | \text{par}) p(\text{par} | \mathbf{y}) d\text{par} \simeq \frac{1}{N} \sum_{i=1}^N p(\mathbf{y}_* | \text{par}^{(i)})$$

with $\text{par}^{(i)}$ drawn from $p(\text{par} | \mathbf{y})$

Bayesian Inference and Predictions

- Draw samples according to the posterior density



- Bayesian inference

$$p(\text{par}|\text{y}) = \frac{p(\text{y}|\text{par})p(\text{par})}{\int p(\text{data}|\text{par})p(\text{par})d\text{par}}$$

- Random walk sampler - accept a proposal with probability

$$\min \left(1, \frac{p(\text{par}'|\text{y})}{p(\text{par}|\text{y})} \right)$$

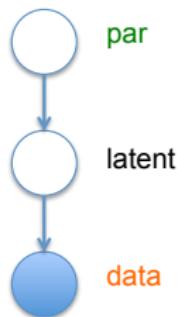
Markov chain Monte Carlo - Random walk example

Acceptance probability : $\min \left(1, \frac{p(\mathbf{y}|\text{par}') p(\text{par}')}{p(\mathbf{y}|\text{par}) p(\text{par})} \right)$

Metropolis et al., *JoCP*, 1953 - Hastings, *Biometrika*, 1970

Challenges

- $p(\mathbf{y}|\text{par})$ might be expensive to compute
- $p(\mathbf{y}|\text{par})$ might not even be computable!



- Marginal likelihood

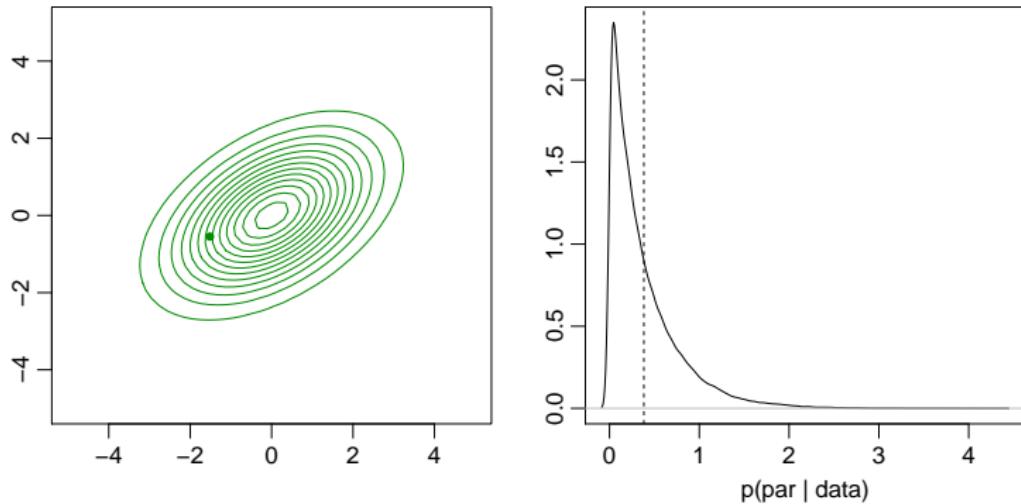
$$p(\mathbf{y}|\text{par}) = \int p(\mathbf{y}|\text{latent})p(\text{latent}|\text{par})d\text{latent}$$

can only be computed if $p(\mathbf{y}|\text{latent})$ is Gaussian

- What if $p(\mathbf{y}|\text{latent})$ is **not** Gaussian?

“Noisy” Markov chain Monte Carlo

$$\mathbb{E} \{\tilde{p}(\mathbf{y}|\text{par})\} = p(\mathbf{y}|\text{par})$$



Andrieu and Roberts, AoS, 2009

“Noisy” Markov chain Monte Carlo

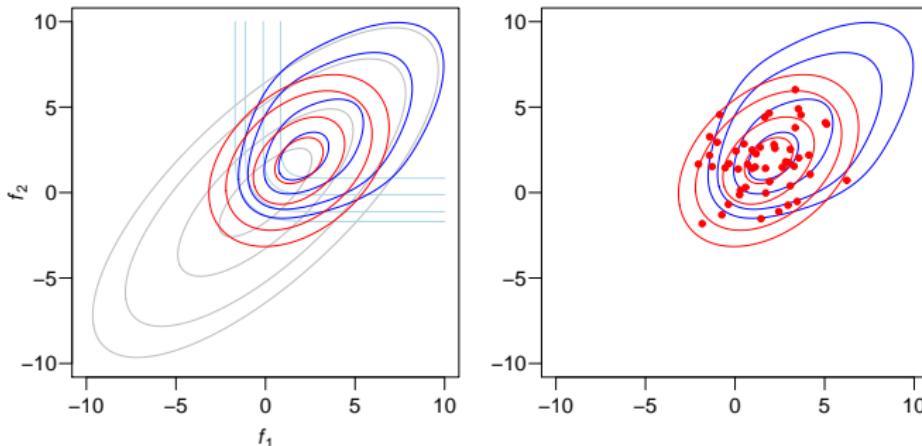
Acceptance probability : $\min \left(1, \frac{\tilde{p}(\mathbf{y}|\text{par}') p(\text{par}')}{\tilde{p}(\mathbf{y}|\text{par}) p(\text{par})} \right)$

Andrieu and Roberts, AoS, 2009

Importance Sampling estimator

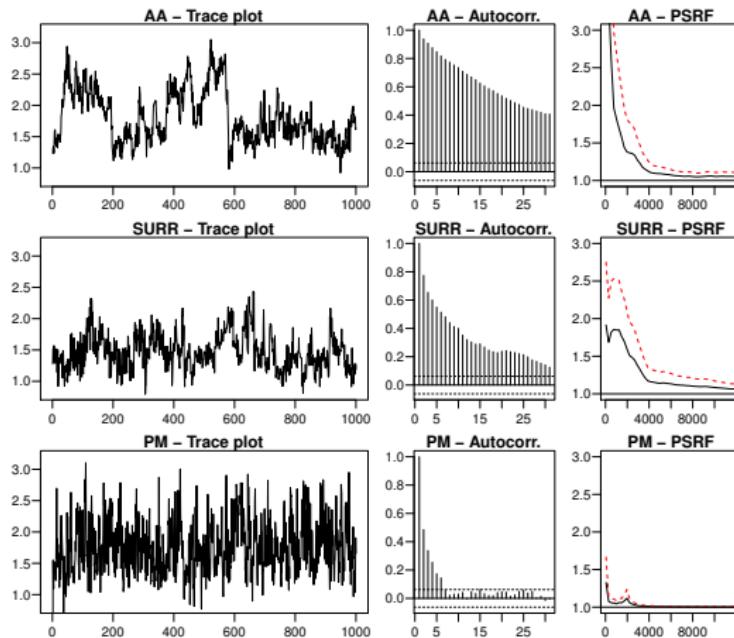
- Approximate posterior over latent variables using $q(\text{latent})$
- Then

$$\tilde{p}(\mathbf{y}|\text{par}) = \frac{1}{N} \sum_{i=1}^N \frac{p(\mathbf{y}|\text{latent}^{(i)}) p(\text{latent}^{(i)}|\text{par})}{q(\text{latent}^{(i)})}$$



Convergence speed and efficiency

Abalone data set (two classes) $n = 2835$ - inference of length-scale



Marginal likelihood in GP models

- Marginal likelihood

$$p(\mathbf{y}|\text{par}) = \int p(\mathbf{y}|\text{latent})p(\text{latent}|\text{par})d\text{latent}$$

can only be computed if $p(\mathbf{y}|\text{latent})$ is Gaussian

- ... even then

$$\log[p(\mathbf{y}|\text{par})] = -\frac{1}{2} \log |K| - \frac{1}{2} \mathbf{y}^T K^{-1} \mathbf{y} + \text{const.}$$

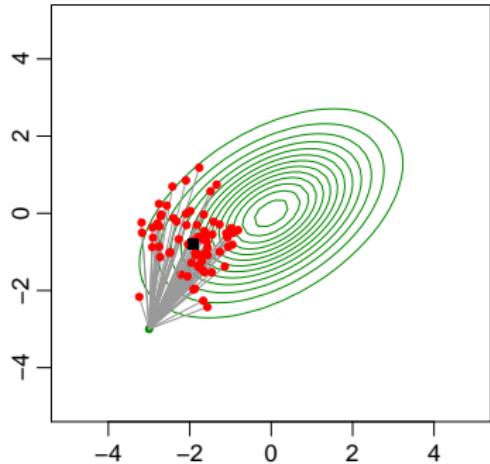
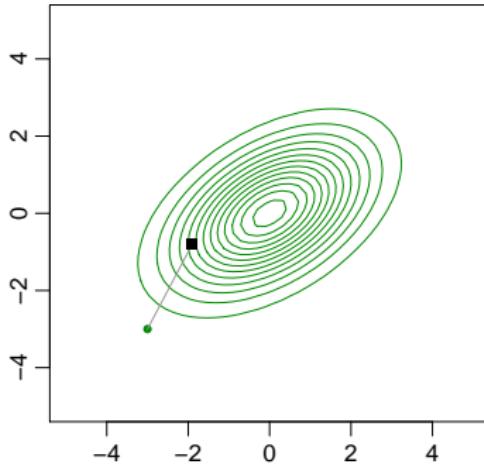
where $K = K(\mathbf{X}, \text{par})$ is a $n \times n$ dense matrix!

Gradient ascent

$$\text{par}' = \text{par} + \frac{\alpha}{2} \nabla_{\text{par}} \log[p(\mathbf{y}|\text{par})p(\text{par})]$$

Stochastic Gradient ascent

$$E \left\{ \widetilde{\nabla_{\text{par}}} \log[p(\mathbf{y}|\text{par})] \right\} = \nabla_{\text{par}} \log[p(\mathbf{y}|\text{par})]$$



Robbins and Monro, AoMS, 1951

Stochastic Gradient ascent

$$\text{par}' = \text{par} + \frac{\alpha_t}{2} \widetilde{\nabla_{\text{par}}} \log[p(\mathbf{y}|\text{par})p(\text{par})] \quad \alpha_t \rightarrow 0$$

Robbins and Monro, AoMS, 1951

Stochastic Gradient Langevin Dynamics (SGLD) algorithm

$$\text{par}' = \text{par} + \frac{\alpha_t}{2} \widetilde{\nabla_{\text{par}}} \log[p(\mathbf{y}|\text{par})p(\text{par})] + \eta_t \quad \eta_t \sim \mathcal{N}(0, \alpha_t)$$

Stochastic Gradients in GP regression

- Marginal likelihood

$$\log[p(\mathbf{y}|\text{par})] = -\frac{1}{2} \log |K| - \frac{1}{2} \mathbf{y}^T K^{-1} \mathbf{y} + \text{const.}$$

- Derivatives wrt par

$$\frac{\partial \log[p(\mathbf{y}|\text{par})]}{\partial \text{par}_i} = -\frac{1}{2} \text{Tr} \left(K^{-1} \frac{\partial K}{\partial \text{par}_i} \right) + \frac{1}{2} \mathbf{y}^T K^{-1} \frac{\partial K}{\partial \text{par}_i} K^{-1} \mathbf{y}$$

Stochastic Gradients in GP regression

- Stochastic estimate of the trace

$$\text{Tr} \left(K^{-1} \frac{\partial K}{\partial \text{par}_i} \right) = \text{Tr} \left(K^{-1} \frac{\partial K}{\partial \text{par}_i} E[\mathbf{r}\mathbf{r}^T] \right) = E \left[\mathbf{r}^T K^{-1} \frac{\partial K}{\partial \text{par}_i} \mathbf{r} \right]$$

with $E[\mathbf{r}\mathbf{r}^T] = I$

- For example r_j drawn from $\{-1, 1\}$ with $p = 1/2$

Stochastic Gradients in GP regression

- Stochastic estimate of the trace

$$\text{Tr} \left(K^{-1} \frac{\partial K}{\partial \text{par}_i} \right) = \text{Tr} \left(K^{-1} \frac{\partial K}{\partial \text{par}_i} E[\mathbf{r}\mathbf{r}^T] \right) = E \left[\mathbf{r}^T K^{-1} \frac{\partial K}{\partial \text{par}_i} \mathbf{r} \right]$$

with $E[\mathbf{r}\mathbf{r}^T] = I$

- For example r_j drawn from $\{-1, 1\}$ with $p = 1/2$
- Stochastic gradient

$$-\frac{1}{2N_r} \sum_{i=1}^{N_r} \mathbf{r}^{(i)T} K^{-1} \frac{\partial K}{\partial \text{par}_i} \mathbf{r}^{(i)} + \frac{1}{2} \mathbf{y}^T K^{-1} \frac{\partial K}{\partial \text{par}_i} K^{-1} \mathbf{y}$$

- Linear systems only!

Solving linear systems

- Linear systems:

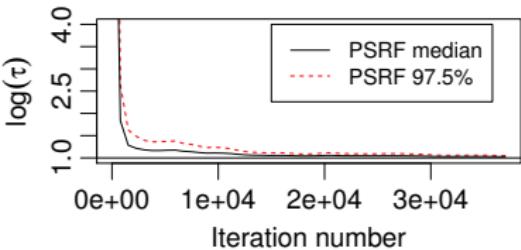
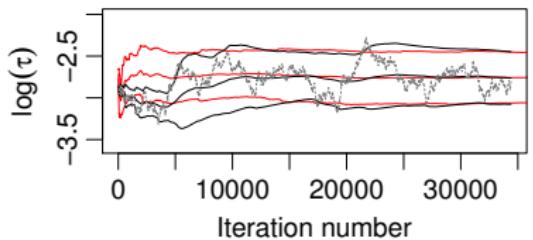
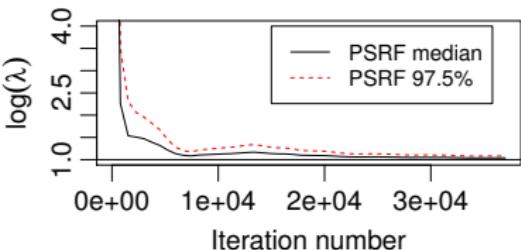
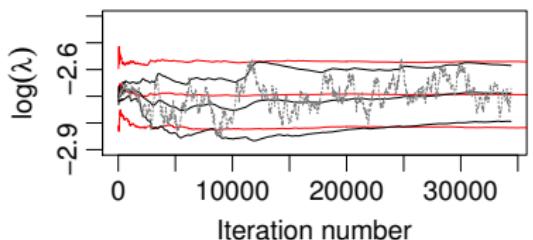
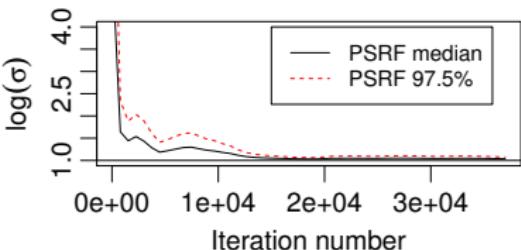
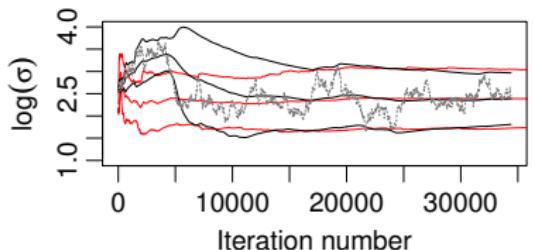
$$Ks = b$$

- Can be solved using conjugate gradient:

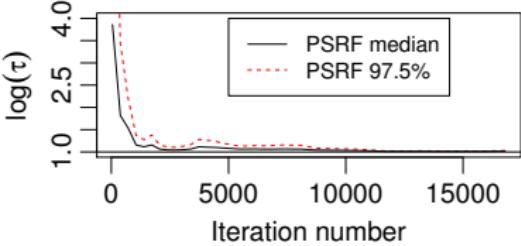
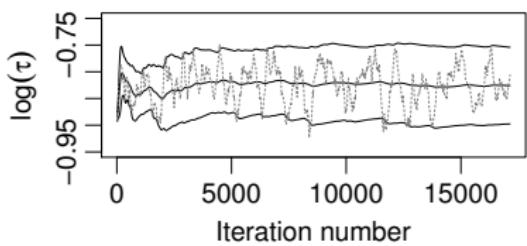
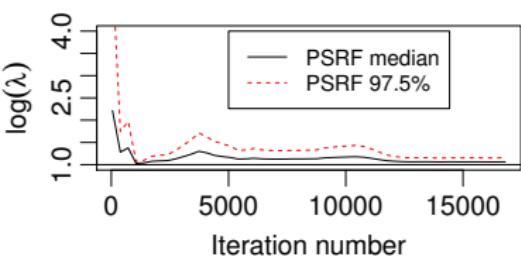
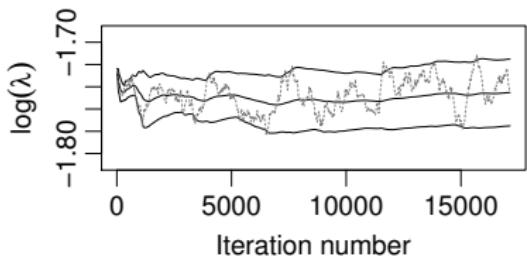
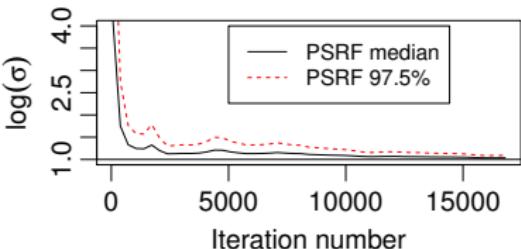
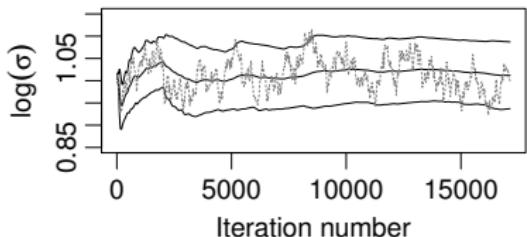
$$s = \arg \min_x \left(\frac{1}{2} x^T K x - x^T b \right)$$

- Iterative update $s = s_0 + \delta_1 + \dots + \delta_T$
- Requires only Kv multiplications! $O(n^2)$ time
- No need to store K ! $O(n)$ space

Comparison with MCMC - Concrete dataset - $n \approx 1K$

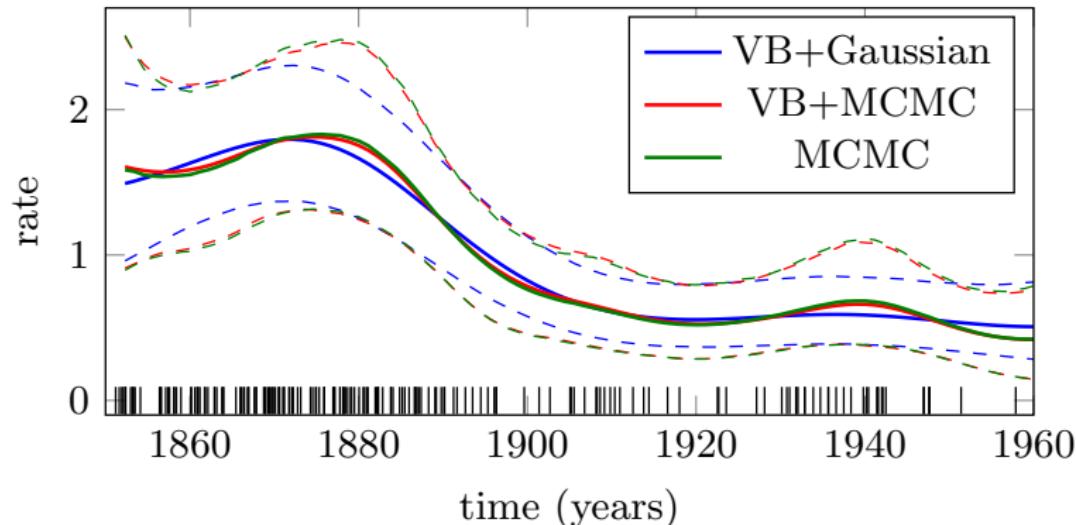


Larger n - Census dataset - $n \approx 23K$



Marrying Sparse Variational and MCMC learning for GPs

- Coal mining disaster data
- Analysis with 30 inducing points



- MNIST data
- Three inducing points before optimizing their position



Three inducing points before optimizing their position

- ... and after

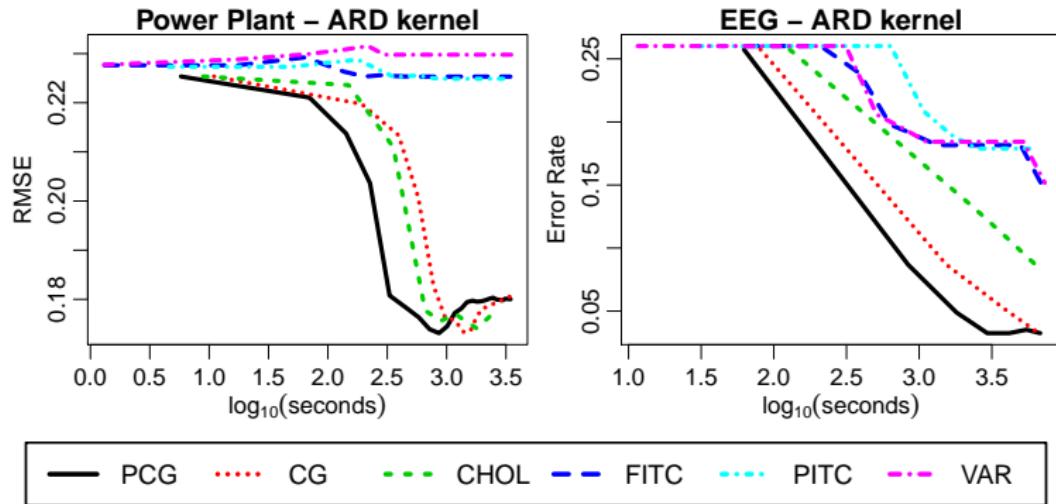


- Overall accuracy 98.04% with 500 inducing points

Hensman, Matthews, Filippone, Ghahramani, *NIPS*, 2015

Preconditioning Kernel Matrices

- Stochastic Gradient Optimization



Cutajar, Osborne, Cunningham, Filippone, ICML, 2016

Conclusions and ongoing work

- GPs form the basis of flexible and interpretable nonparametric statistical models

Conclusions and ongoing work

- GPs form the basis of flexible and interpretable nonparametric statistical models
- GP learning is generally computationally intensive

Conclusions and ongoing work

- GPs form the basis of flexible and interpretable nonparametric statistical models
- GP learning is generally computationally intensive
- Introducing approximations can severely affect quantification of uncertainty and reduce accuracy

Conclusions and ongoing work

- GPs form the basis of flexible and interpretable nonparametric statistical models
- GP learning is generally computationally intensive
- Introducing approximations can severely affect quantification of uncertainty and reduce accuracy
- Approximate **unbiased** computations offer practical and scalable ways to carry out **exact** GP learning

Acknowledgments & References



Andre Marquand
Radboud



Guido Sanguinetti
Edinburgh



James Hensman
Lancaster



Mark Girolami
Warwick



Alessandro Vinciarelli
Glasgow



Dirk Husmeier
Glasgow

- [1] K. Cutajar, M. A. Osborne, J. P. Cunningham, and M. Filippone. Preconditioning kernel matrices, *ICML 2016*.
- [2] J. Hensman, A. G. de G. Matthews, M. Filippone, and Z. Ghahramani. MCMC for variationally sparse Gaussian processes, *NIPS 2015*.
- [3] M. Filippone and R. Engler. Enabling scalable stochastic gradient-based inference for Gaussian processes by employing the Unbiased LInear System SolvEr (ULISSE), *ICML 2015*.
- [4] M. Filippone and M. Girolami. Pseudo-Marginal Bayesian inference for Gaussian processes, *IEEE T-PAMI, 2014*.
- [5] M. Filippone. Bayesian inference for Gaussian process classifiers with annealing and pseudo-marginal MCMC, *ICPR 2014*.
- [6] M. Filippone et al. Probabilistic prediction of neurological disorders with a statistical assessment of neuroimaging data modalities. *Annals of Applied Statistics, 2012*.
- [7] A. F. Marquand et al. Automated, high accuracy classification of Parkinsonian disorders: a pattern recognition approach. *PLoS ONE, 2013*.
- [8] M. Filippone et al. A comparative evaluation of stochastic-based inference methods for Gaussian process models. *Machine Learning, 2013*.