Scalable Multi-Class Gaussian Process Classification using Expectation Propagation

Carlos Villacampa-Calvo and Daniel Hernández–Lobato
Computer Science Department
Universidad Autónoma de Madrid

http://dhnzl.org, daniel.hernandez@uam.es

Introduction to Multi-class Classification with GPs

Given $\mathbf{x}_i$ we want to make **predictions** about $y_i \in \{1, \ldots, C\}$, $C > 2$.

One can **assume** that (Kim & Ghahramani, 2006):

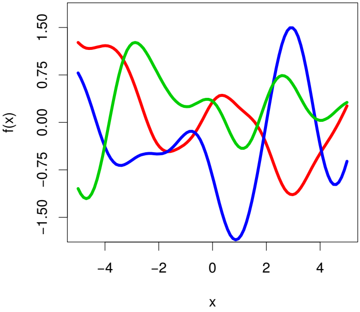
$$y_i = \arg\max_k \quad f^k(\mathbf{x}_i) \quad \text{for} \quad k \in \{1, \ldots, C\}$$

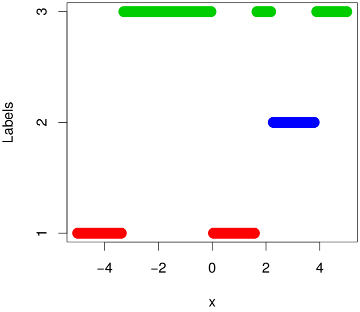
Introduction to Multi-class Classification with GPs

Given $\mathbf{x}_i$ we want to make **predictions** about $y_i \in \{1, \ldots, C\}$, $C > 2$.

One can **assume** that (Kim & Ghahramani, 2006):

$$y_i = \arg\max_k \quad f^k(\mathbf{x}_i) \quad \text{for} \quad k \in \{1, \ldots, C\}$$



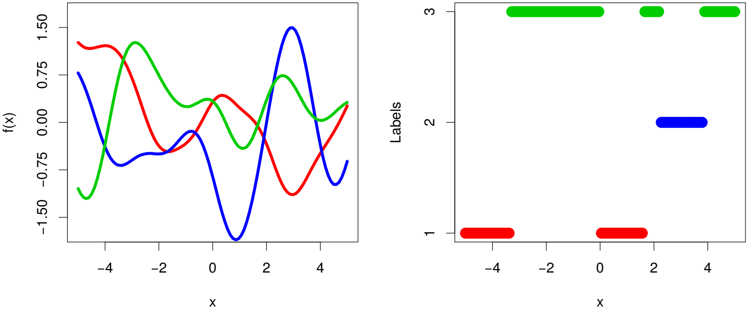


Introduction to Multi-class Classification with GPs

Given $\mathbf{x}_i$ we want to make **predictions** about $y_i \in \{1, \ldots, C\}$, $C > 2$.

One can **assume** that (Kim & Ghahramani, 2006):

$$y_i = \arg\max_k \quad f^k(\mathbf{x}_i) \quad \text{for} \quad k \in \{1, \ldots, C\}$$



Find $p(\mathbf{f}|\mathbf{y}) = p(\mathbf{y}|\mathbf{f})p(\mathbf{f})/p(\mathbf{y})$ **under** $p(\mathbf{f}^k) \sim \mathcal{GP}(0, k(\cdot, \cdot))$.

Challenges in Multi-class Classification with GPs

Binary classification has received **more attention** than multi-class!

Challenges in the **multi-class case**:

1. Approximate inference is more difficult.

Challenges in Multi-class Classification with GPs

Binary classification has received **more attention** than multi-class!

Challenges in the **multi-class case**:

1. Approximate inference is more difficult.
2. $C > 2$ latent functions instead of just one.

Challenges in Multi-class Classification with GPs

Binary classification has received **more attention** than multi-class!

Challenges in the **multi-class case**:

1. Approximate inference is more difficult.
2. $C > 2$ latent functions instead of just one.
3. Deal with more complicated likelihood factors.

Challenges in Multi-class Classification with GPs

Binary classification has received **more attention** than multi-class!

Challenges in the **multi-class case**:

1. Approximate inference is more difficult.
2. $C > 2$ latent functions instead of just one.
3. Deal with more complicated likelihood factors.
4. More expensive algorithms, computationally.

Challenges in Multi-class Classification with GPs

Binary classification has received **more attention** than multi-class!

Challenges in the **multi-class case**:

1. Approximate inference is more difficult.
2. $C > 2$ latent functions instead of just one.
3. Deal with more complicated likelihood factors.
4. More expensive algorithms, computationally.

Most techniques **do not scale** to large datasets: (Williams & Barber, 1998; Kim & Ghahramani, 2006; Girolami & Rogers, 2006; Chai, 2012; Riihimäki et al., 2013).

Challenges in Multi-class Classification with GPs

Binary classification has received **more attention** than multi-class!

Challenges in the **multi-class case**:

1. Approximate inference is more difficult.
2. $C > 2$ latent functions instead of just one.
3. Deal with more complicated likelihood factors.
4. More expensive algorithms, computationally.

Most techniques **do not scale** to large datasets: (Williams & Barber, 1998; Kim & Ghahramani, 2006; Girolami & Rogers, 2006; Chai, 2012; Riihimäki et al., 2013).

The best cost is $\mathcal{O}(CNM^2)$, if sparse priors are used.

Stochastic Variational Inference for Multi-class GPs

Hensman *et al.*, 2015, use a **robust likelihood** function:

$$p(y_i|\mathbf{f}_i) = (1-\epsilon)p_i + \frac{\epsilon}{C-1}(1-p_i) \quad \text{with} \quad p_i = \begin{cases} 1 & \text{if} \quad y_i = \arg\max_k \quad f^k(\mathbf{x}_i) \\ 0 & \text{otherwise} \end{cases}$$

Stochastic Variational Inference for Multi-class GPs

Hensman *et al.*, 2015, use a **robust likelihood** function:

$$p(y_i|\mathbf{f}_i) = (1-\epsilon)p_i + \frac{\epsilon}{C-1}(1-p_i) \quad \text{with} \quad p_i = \begin{cases} 1 & \text{if} \quad y_i = arg\max_k \quad f^k(\mathbf{x}_i) \\ 0 & \text{otherwise} \end{cases}$$

The **posterior approximation** is $q(\mathbf{f}) = \int p(\mathbf{f}|\bar{\mathbf{f}})q(\bar{\mathbf{f}})d\bar{\mathbf{f}}$

$$q(\bar{\mathbf{f}}) = \textstyle\prod_{k=1}^{C} \mathcal{N}(\bar{\mathbf{f}}^k|\boldsymbol{\mu}^k, \boldsymbol{\Sigma}^k)$$

$$\bar{\mathbf{f}}^k = (f^k(\bar{\mathbf{x}}_1^k), \ldots, f^k(\bar{\mathbf{x}}_M^k))^\mathsf{T} \qquad \bar{\mathbf{X}}^k = (\bar{\mathbf{x}}_1^k, \ldots, \bar{\mathbf{x}}_M^k)^\mathsf{T}$$

Stochastic Variational Inference for Multi-class GPs

Hensman *et al.*, 2015, use a **robust likelihood** function:

$$p(y_i|\mathbf{f}_i) = (1-\epsilon)p_i + \frac{\epsilon}{C-1}(1-p_i) \quad \text{with} \quad p_i = \begin{cases} 1 & \text{if} \quad y_i = arg\,\max\limits_k \quad f^k(\mathbf{x}_i) \\ 0 & \text{otherwise} \end{cases}$$

The **posterior approximation** is $q(\mathbf{f}) = \int p(\mathbf{f}|\bar{\mathbf{f}})q(\bar{\mathbf{f}})d\bar{\mathbf{f}}$

$$q(\bar{\mathbf{f}}) = \prod_{k=1}^{C} \mathcal{N}(\bar{\mathbf{f}}^k|\boldsymbol{\mu}^k, \boldsymbol{\Sigma}^k)$$

$$\bar{\mathbf{f}}^k = (f^k(\bar{\mathbf{x}}_1^k), \ldots, f^k(\bar{\mathbf{x}}_M^k))^\mathsf{T} \qquad \overline{\mathbf{X}}^k = (\bar{\mathbf{x}}_1^k, \ldots, \bar{\mathbf{x}}_M^k)^\mathsf{T}$$

The number of **latent variables** goes from CN to CM, with $M \ll N$.

Stochastic Variational Inference for Multi-class GPs

Hensman *et al.*, 2015, use a **robust likelihood** function:

$$p(y_i|\mathbf{f}_i) = (1-\epsilon)p_i + \frac{\epsilon}{C-1}(1-p_i) \quad \text{with} \quad p_i = \begin{cases} 1 & \text{if} \quad y_i = arg\,\max_k \quad f^k(\mathbf{x}_i) \\ 0 & \text{otherwise} \end{cases}$$

The **posterior approximation** is $q(\mathbf{f}) = \int p(\mathbf{f}|\bar{\mathbf{f}})q(\bar{\mathbf{f}})d\bar{\mathbf{f}}$

$$q(\bar{\mathbf{f}}) = \prod_{k=1}^{C} \mathcal{N}(\bar{\mathbf{f}}^k|\boldsymbol{\mu}^k, \boldsymbol{\Sigma}^k)$$

$$\bar{\mathbf{f}}^k = (f^k(\bar{\mathbf{x}}_1^k), \ldots, f^k(\bar{\mathbf{x}}_M^k))^\mathsf{T} \qquad \overline{\mathbf{X}}^k = (\bar{\mathbf{x}}_1^k, \ldots, \bar{\mathbf{x}}_M^k)^\mathsf{T}$$

The number of **latent variables** goes from CN to CM, with $M \ll N$.

$$\mathcal{L}(q) = \sum_{i=1}^{N} \mathbb{E}_q \left[\log p(y_i|\mathbf{f}_i)\right] - \mathrm{KL}(q|p)$$

Stochastic Variational Inference for Multi-class GPs

Hensman *et al.*, 2015, use a **robust likelihood** function:

$$p(y_i|\mathbf{f}_i) = (1-\epsilon)p_i + \frac{\epsilon}{C-1}(1-p_i) \quad \text{with} \quad p_i = \begin{cases} 1 & \text{if} \quad y_i = arg\,\max_k \quad f^k(\mathbf{x}_i) \\ 0 & \text{otherwise} \end{cases}$$

The **posterior approximation** is $q(\mathbf{f}) = \int p(\mathbf{f}|\overline{\mathbf{f}})q(\overline{\mathbf{f}})d\overline{\mathbf{f}}$

$$q(\overline{\mathbf{f}}) = \textstyle\prod_{k=1}^{C} \mathcal{N}(\overline{\mathbf{f}}^k|\boldsymbol{\mu}^k, \boldsymbol{\Sigma}^k)$$

$$\overline{\mathbf{f}}^k = (f^k(\overline{\mathbf{x}}_1^k), \ldots, f^k(\overline{\mathbf{x}}_M^k))^\mathsf{T} \qquad \overline{\mathbf{X}}^k = (\overline{\mathbf{x}}_1^k, \ldots, \overline{\mathbf{x}}_M^k)^\mathsf{T}$$

The number of **latent variables** goes from CN to CM, with $M \ll N$.

$$\mathcal{L}(q) = \sum_{i=1}^{N} \mathbb{E}_q\left[\log p(y_i|\mathbf{f}_i)\right] - \mathrm{KL}(q|p)$$

The cost is $\mathcal{O}(CM^3)$ (uses **quadratures**)!

Stochastic Variational Inference for Multi-class GPs

Hensman *et al.*, 2015, use a **robust likelihood** function:

$$p(y_i|\mathbf{f}_i) = (1-\epsilon)p_i + \frac{\epsilon}{C-1}(1-p_i) \quad \text{with} \quad p_i = \begin{cases} 1 & \text{if} \quad y_i = arg\,\max_k \quad f^k(\mathbf{x}_i) \\ 0 & \text{otherwise} \end{cases}$$

The **posterior approximation** is $q(\mathbf{f}) = \int p(\mathbf{f}|\bar{\mathbf{f}})q(\bar{\mathbf{f}})d\bar{\mathbf{f}}$

$$q(\bar{\mathbf{f}}) = \prod_{k=1}^{C} \mathcal{N}(\bar{\mathbf{f}}^k|\boldsymbol{\mu}^k, \boldsymbol{\Sigma}^k)$$

$$\bar{\mathbf{f}}^k = (f^k(\bar{\mathbf{x}}_1^k), \ldots, f^k(\bar{\mathbf{x}}_M^k))^\mathsf{T} \qquad \bar{\mathbf{X}}^k = (\bar{\mathbf{x}}_1^k, \ldots, \bar{\mathbf{x}}_M^k)^\mathsf{T}$$

The number of **latent variables** goes from CN to CM, with $M \ll N$.

$$\mathcal{L}(q) = \sum_{i=1}^{N} \mathbb{E}_q \left[\log p(y_i|\mathbf{f}_i)\right] - \mathrm{KL}(q|p)$$

The cost is $\mathcal{O}(CM^3)$ (uses **quadratures**)! Can we do that with **EP**?

Expectation Propagation (EP)

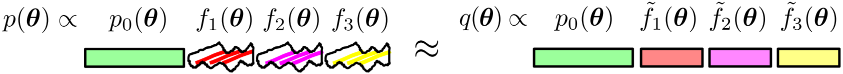
Let $\boldsymbol{\theta}$ summarize the latent variables of the model.

Approximates $\boxed{p(\boldsymbol{\theta}) \propto p_0(\boldsymbol{\theta}) \prod_{n=1}^{N} f_n(\boldsymbol{\theta})}$ with $\boxed{q(\boldsymbol{\theta}) \propto p_0(\boldsymbol{\theta}) \prod_{n=1}^{N} \tilde{f}_n(\boldsymbol{\theta})}$

Expectation Propagation (EP)

Let $\boldsymbol{\theta}$ summarize the latent variables of the model.

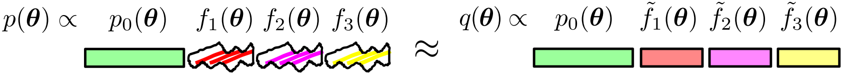
Approximates $p(\boldsymbol{\theta}) \propto p_0(\boldsymbol{\theta}) \prod_{n=1}^{N} f_n(\boldsymbol{\theta})$ with $q(\boldsymbol{\theta}) \propto p_0(\boldsymbol{\theta}) \prod_{n=1}^{N} \tilde{f}_n(\boldsymbol{\theta})$



Expectation Propagation (EP)

Let $\boldsymbol{\theta}$ summarize the latent variables of the model.

Approximates $p(\boldsymbol{\theta}) \propto p_0(\boldsymbol{\theta}) \prod_{n=1}^N f_n(\boldsymbol{\theta})$ with $q(\boldsymbol{\theta}) \propto p_0(\boldsymbol{\theta}) \prod_{n=1}^N \tilde{f}_n(\boldsymbol{\theta})$



The $\tilde{f}_n$ are tuned by minimizing the KL divergence

$$D_{\text{KL}}[p_n || q] \quad \text{for } n = 1, \ldots, N\,, \quad \text{where} \quad \begin{array}{rcl} p_n(\boldsymbol{\theta}) & \propto & f_n(\boldsymbol{\theta}) \prod_{j \neq n} \tilde{f}_j(\boldsymbol{\theta}) \\ q(\boldsymbol{\theta}) & \propto & \tilde{f}_n(\boldsymbol{\theta}) \prod_{j \neq n} \tilde{f}_j(\boldsymbol{\theta}) \end{array}.$$

Model Specification

We consider that $y_i = \arg\max_k \; f^k(\mathbf{x}_i)$, which gives the **likelihood**:

$$p(\mathbf{y}|\mathbf{f}) = \prod_{i=1}^{N} p(y_i|\mathbf{f}_i) = \prod_{i=1}^{N} \prod_{k \neq y_i} \Theta(f^{y_i}(\mathbf{x}_i) - f^k(\mathbf{x}_i))$$

Model Specification

We consider that $y_i = \arg\max_k \; f^k(\mathbf{x}_i)$, which gives the **likelihood**:

$$p(\mathbf{y}|\mathbf{f}) = \prod_{i=1}^N p(y_i|\mathbf{f}_i) = \prod_{i=1}^N \prod_{k \neq y_i} \Theta(f^{y_i}(\mathbf{x}_i) - f^k(\mathbf{x}_i))$$

The **posterior approximation** is also set to be $q(\mathbf{f}) = \int p(\mathbf{f}|\bar{\mathbf{f}})q(\bar{\mathbf{f}})d\bar{\mathbf{f}}$.

Model Specification

We consider that $y_i = \arg\max_k \; f^k(\mathbf{x}_i)$, which gives the **likelihood**:

$$p(\mathbf{y}|\mathbf{f}) = \prod_{i=1}^N p(y_i|\mathbf{f}_i) = \prod_{i=1}^N \prod_{k \neq y_i} \Theta(f^{y_i}(\mathbf{x}_i) - f^k(\mathbf{x}_i))$$

The **posterior approximation** is also set to be $q(\mathbf{f}) = \int p(\mathbf{f}|\bar{\mathbf{f}}) q(\bar{\mathbf{f}}) d\bar{\mathbf{f}}$.

The **posterior** over $\bar{\mathbf{f}}$ is:

$$p(\bar{\mathbf{f}}|\mathbf{y}) = \frac{\int p(\mathbf{y}|\mathbf{f}) p(\mathbf{f}|\bar{\mathbf{f}}) d\mathbf{f} \, p(\bar{\mathbf{f}})}{p(\mathbf{y})} \approx \frac{[\prod_{i=1}^N \int p(y_i|\mathbf{f}_i) p(\mathbf{f}_i|\bar{\mathbf{f}}) d\mathbf{f}_i] p(\bar{\mathbf{f}})}{p(\mathbf{y})}$$

where we have used the FITC approximation $p(\mathbf{f}|\bar{\mathbf{f}}) \approx \prod_{i=1}^N p(\mathbf{f}_i|\bar{\mathbf{f}})$.

Model Specification

We consider that $y_i = \arg\max_k \quad f^k(\mathbf{x}_i)$, which gives the **likelihood**:

$$p(\mathbf{y}|\mathbf{f}) = \prod_{i=1}^N p(y_i|\mathbf{f}_i) = \prod_{i=1}^N \prod_{k \neq y_i} \Theta(f^{y_i}(\mathbf{x}_i) - f^k(\mathbf{x}_i))$$

The **posterior approximation** is also set to be $q(\mathbf{f}) = \int p(\mathbf{f}|\bar{\mathbf{f}})q(\bar{\mathbf{f}})d\bar{\mathbf{f}}$.

The **posterior** over $\bar{\mathbf{f}}$ is:

$$p(\bar{\mathbf{f}}|\mathbf{y}) = \frac{\int p(\mathbf{y}|\mathbf{f})p(\mathbf{f}|\bar{\mathbf{f}})d\mathbf{f}\,p(\bar{\mathbf{f}})}{p(\mathbf{y})} \approx \frac{[\prod_{i=1}^N \int p(y_i|\mathbf{f}_i)p(\mathbf{f}_i|\bar{\mathbf{f}})d\mathbf{f}_i]p(\bar{\mathbf{f}})}{p(\mathbf{y})}$$

where we have used the FITC approximation $p(\mathbf{f}|\bar{\mathbf{f}}) \approx \prod_{i=1}^N p(\mathbf{f}_i|\bar{\mathbf{f}})$.

The corresponding **likelihood factors** are:

$$\phi_i(\bar{\mathbf{f}}) = \int \left[\prod_{k \neq y_i} \Theta\left(f_i^{y_i} - f_i^k\right)\right] \prod_{k=1}^C p(f_i^k|\bar{\mathbf{f}}^k)d\mathbf{f}_i$$

Model Specification

We consider that $y_i = \arg\max_k \quad f^k(\mathbf{x}_i)$, which gives the **likelihood**:

$$p(\mathbf{y}|\mathbf{f}) = \prod_{i=1}^N p(y_i|\mathbf{f}_i) = \prod_{i=1}^N \prod_{k \neq y_i} \Theta(f^{y_i}(\mathbf{x}_i) - f^k(\mathbf{x}_i))$$

The **posterior approximation** is also set to be $q(\mathbf{f}) = \int p(\mathbf{f}|\overline{\mathbf{f}})q(\overline{\mathbf{f}})d\overline{\mathbf{f}}$.

The **posterior** over $\overline{\mathbf{f}}$ is:

$$p(\overline{\mathbf{f}}|\mathbf{y}) = \frac{\int p(\mathbf{y}|\mathbf{f})p(\mathbf{f}|\overline{\mathbf{f}})d\mathbf{f}\, p(\overline{\mathbf{f}})}{p(\mathbf{y})} \approx \frac{[\prod_{i=1}^N \int p(y_i|\mathbf{f}_i)p(\mathbf{f}_i|\overline{\mathbf{f}})d\mathbf{f}_i]p(\overline{\mathbf{f}})}{p(\mathbf{y})}$$

where we have used the FITC approximation $p(\mathbf{f}|\overline{\mathbf{f}}) \approx \prod_{i=1}^N p(\mathbf{f}_i|\overline{\mathbf{f}})$.

The corresponding **likelihood factors** are:

$$\phi_i(\overline{\mathbf{f}}) = \int \left[\prod_{k \neq y_i} \Theta\left(f_i^{y_i} - f_i^k\right)\right] \prod_{k=1}^C p(f_i^k|\overline{\mathbf{f}}^k)d\mathbf{f}_i$$

The integral is **intractable** and we cannot evaluate $\phi_i(\overline{\mathbf{f}})$ in closed form!

Approximate Likelihood Factors

It is possible to show that:

$$\phi_i(\bar{\mathbf{f}}) = p(f_i^{y_i} > f_i^1, \ldots, f_i^{y_i} > f_i^{y_i-1}, f_i^{y_i} > f_i^{y_i+1}, \ldots, f_i^{y_i} > f_i^C)$$

Approximate Likelihood Factors

It is possible to show that:

$$
\begin{aligned}
\phi_i(\bar{\mathbf{f}}) &= p(f_i^{y_i} > f_i^1, \ldots, f_i^{y_i} > f_i^{y_i-1}, f_i^{y_i} > f_i^{y_i+1}, \ldots, f_i^{y_i} > f_i^C) \\
&= p(f_i^{y_i} > f_i^1 | \ldots, f_i^{y_i} > f_i^{y_i-1}, f_i^{y_i} > f_i^{y_i+1}, \ldots, f_i^{y_i} > f_i^C) \times \\
&\quad p(f_i^{y_i} > f_i^2 | \ldots, f_i^{y_i} > f_i^{y_i-1}, f_i^{y_i} > f_i^{y_i+1}, \ldots, f_i^{y_i} > f_i^C) \times \cdots \\
&\quad \cdots \times p(f_i^{y_i} > f_i^{C-1} | f_i^{y_i} > f_i^C) \times p(f_i^{y_i} > f_i^C)
\end{aligned}
$$

Approximate Likelihood Factors

It is possible to show that:

$$
\begin{aligned}
\phi_i(\bar{\mathbf{f}}) &= p(f_i^{y_i} > f_i^1, \ldots, f_i^{y_i} > f_i^{y_i-1}, f_i^{y_i} > f_i^{y_i+1}, \ldots, f_i^{y_i} > f_i^C) \\
&= p(f_i^{y_i} > f_i^1 | \ldots, f_i^{y_i} > f_i^{y_i-1}, f_i^{y_i} > f_i^{y_i+1}, \ldots, f_i^{y_i} > f_i^C) \times \\
&\quad p(f_i^{y_i} > f_i^2 | \ldots, f_i^{y_i} > f_i^{y_i-1}, f_i^{y_i} > f_i^{y_i+1}, \ldots, f_i^{y_i} > f_i^C) \times \cdots \\
&\quad \cdots \times p(f_i^{y_i} > f_i^{C-1} | f_i^{y_i} > f_i^C) \times p(f_i^{y_i} > f_i^C) \\
&\approx \prod_{k \neq y_i} p(f_i^{y_i} > f_i^k) = \prod_{k \neq y_i} \Phi(\alpha_i^k)
\end{aligned}
$$

Approximate Likelihood Factors

It is possible to show that:

$$
\begin{aligned}
\phi_i(\bar{\mathbf{f}}) &= p(f_i^{y_i} > f_i^1, \ldots, f_i^{y_i} > f_i^{y_i-1}, f_i^{y_i} > f_i^{y_i+1}, \ldots, f_i^{y_i} > f_i^C) \\
&= p(f_i^{y_i} > f_i^1 | \ldots, f_i^{y_i} > f_i^{y_i-1}, f_i^{y_i} > f_i^{y_i+1}, \ldots, f_i^{y_i} > f_i^C) \times \\
&\quad p(f_i^{y_i} > f_i^2 | \ldots, f_i^{y_i} > f_i^{y_i-1}, f_i^{y_i} > f_i^{y_i+1}, \ldots, f_i^{y_i} > f_i^C) \times \cdots \\
&\quad \cdots \times p(f_i^{y_i} > f_i^{C-1} | f_i^{y_i} > f_i^C) \times p(f_i^{y_i} > f_i^C) \\
&\approx \prod_{k \neq y_i} p(f_i^{y_i} > f_i^k) = \prod_{k \neq y_i} \Phi(\alpha_i^k)
\end{aligned}
$$

where $\Phi(\cdot)$ is the cdf of a standard Gaussian and we have defined

$$
\alpha_i^k = (m_i^{y_i} - m_i^k) / \sqrt{v_i^{y_i} + v_i^k}
$$

with $m_i^{y_i}$, m_i^k, $v_i^{y_i}$ and v_i^k the mean and variances of $f_i^{y_i}$ and f_i^k.

EP Approximation of the Likelihood Factors

EP approximates each likelihood factor ϕ_i^k with a **Gaussian factor**:

$$\Phi(\alpha_i^k) = \phi_i^k(\bar{\mathbf{f}}) \approx \tilde{\phi}_i^k(\bar{\mathbf{f}}) = \tilde{s}_{i,k} \exp\left\{-\frac{1}{2}(\bar{\mathbf{f}}^{y_i})^\mathsf{T}\tilde{\mathbf{V}}_{i,k}^{y_i}\bar{\mathbf{f}}^{y_i} + (\bar{\mathbf{f}}^{y_i})^\mathsf{T}\tilde{\mathbf{m}}_{i,k}^{y_i}\right\} \times$$
$$\exp\left\{-\frac{1}{2}(\bar{\mathbf{f}}^{k})^\mathsf{T}\tilde{\mathbf{V}}_{i,k}^{k}\bar{\mathbf{f}}^{k} + (\bar{\mathbf{f}}^{k})^\mathsf{T}\tilde{\mathbf{m}}_{i,k}^{k}\right\}$$

EP Approximation of the Likelihood Factors

EP approximates each likelihood factor ϕ_i^k with a **Gaussian factor**:

$$\Phi(\alpha_i^k) = \phi_i^k(\bar{\mathbf{f}}) \approx \tilde{\phi}_i^k(\bar{\mathbf{f}}) = \tilde{s}_{i,k} \exp\left\{-\frac{1}{2}(\bar{\mathbf{f}}^{y_i})^\mathsf{T}\tilde{\mathbf{V}}_{i,k}^{y_i}\bar{\mathbf{f}}^{y_i} + (\bar{\mathbf{f}}^{y_i})^\mathsf{T}\tilde{\mathbf{m}}_{i,k}^{y_i}\right\} \times$$
$$\exp\left\{-\frac{1}{2}(\bar{\mathbf{f}}^{k})^\mathsf{T}\tilde{\mathbf{V}}_{i,k}^{k}\bar{\mathbf{f}}^{k} + (\bar{\mathbf{f}}^{k})^\mathsf{T}\tilde{\mathbf{m}}_{i,k}^{k}\right\}$$

$\tilde{\mathbf{V}}_{i,k}^{y_i}$ and $\tilde{\mathbf{V}}_{i,k}^{k}$ are **1-rank matrices**. Each $\tilde{\phi}_i^k$ only has $\mathcal{O}(M)$ parameters.

EP Approximation of the Likelihood Factors

EP approximates each likelihood factor ϕ_i^k with a **Gaussian factor**:

$$\Phi(\alpha_i^k) = \phi_i^k(\bar{\mathbf{f}}) \approx \tilde{\phi}_i^k(\bar{\mathbf{f}}) = \tilde{s}_{i,k} \exp\left\{ -\frac{1}{2} (\bar{\mathbf{f}}^{y_i})^\mathrm{T} \tilde{\mathbf{V}}_{i,k}^{y_i} \bar{\mathbf{f}}^{y_i} + (\bar{\mathbf{f}}^{y_i})^\mathrm{T} \tilde{\mathbf{m}}_{i,k}^{y_i} \right\} \times$$
$$\exp\left\{ -\frac{1}{2} (\bar{\mathbf{f}}^{k})^\mathrm{T} \tilde{\mathbf{V}}_{i,k}^{k} \bar{\mathbf{f}}^{k} + (\bar{\mathbf{f}}^{k})^\mathrm{T} \tilde{\mathbf{m}}_{i,k}^{k} \right\}$$

$\tilde{\mathbf{V}}_{i,k}^{y_i}$ and $\tilde{\mathbf{V}}_{i,k}^{k}$ are **1-rank matrices**. Each $\tilde{\phi}_i^k$ only has $\mathcal{O}(M)$ parameters.

The **posterior approximation** is:

$$q(\bar{\mathbf{f}}) = \frac{1}{Z_q} \prod_{i=1}^{N} \prod_{k \neq y_i} \tilde{\phi}_i^k(\bar{\mathbf{f}}) p(\bar{\mathbf{f}})$$

and Z_q approximates the **marginal likelihood** of the model.

Approx. Maximization of the Marginal Likelihood

Z_q is **maximized** w.r.t. $\boldsymbol{\xi}_k$ and $\overline{\mathbf{X}}^k$ to find good hyper-parameters.

Approx. Maximization of the Marginal Likelihood

Z_q is **maximized** w.r.t. $\boldsymbol{\xi}_k$ and $\overline{\mathbf{X}}^k$ to find good hyper-parameters.

If EP converges, the **gradient** of $\log Z_q$ is given by:

$$\frac{\partial \log Z_q}{\partial \xi_j^k} = \boldsymbol{\eta}^\mathsf{T} \frac{\partial \theta_{\text{prior}}}{\partial \xi_j^k} - \boldsymbol{\eta}_{\text{prior}}^\mathsf{T} \frac{\partial \theta_{\text{prior}}}{\partial \xi_j^k} + \sum_{i=1}^{N} \sum_{k \neq y_i} \frac{\partial \log Z_{i,k}}{\partial \xi_j^k}$$

where $Z_{i,k}$ is the normalization constant of $\phi_{i,k} q^{\setminus i,k}$ with $q^{\setminus i,k} \propto q / \tilde{\phi}_{i,k}$.

Approx. Maximization of the Marginal Likelihood

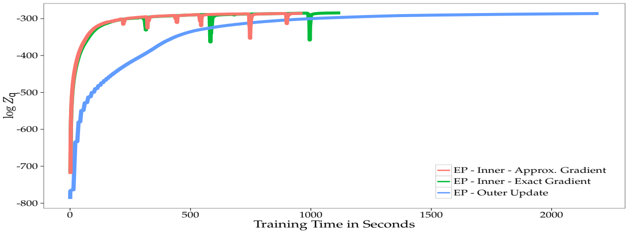
Z_q is **maximized** w.r.t. $\boldsymbol{\xi}_k$ and $\overline{\mathbf{X}}^k$ to find good hyper-parameters.

If EP converges, the **gradient** of $\log Z_q$ is given by:

$$\frac{\partial \log Z_q}{\partial \xi_j^k} = \boldsymbol{\eta}^\mathsf{T} \frac{\partial \theta_{\text{prior}}}{\partial \xi_j^k} - \boldsymbol{\eta}_{\text{prior}}^\mathsf{T} \frac{\partial \theta_{\text{prior}}}{\partial \xi_j^k} + \sum_{i=1}^N \sum_{k \neq y_i} \frac{\partial \log Z_{i,k}}{\partial \xi_j^k}$$

where $Z_{i,k}$ is the normalization constant of $\phi_{i,k} q^{\setminus i,k}$ with $q^{\setminus i,k} \propto q / \tilde{\phi}_{i,k}$.

Hernández-Lobato and Hernández-Lobato, 2016 show **convergence is not needed**.



Expectation Propagation using Mini-batches

Consider a **minibatch** of data $\mathcal{M}_b$:

Expectation Propagation using Mini-batches

Consider a **minibatch** of data $\mathcal{M}_b$:

1. Refine in parallel all approximate factors $\tilde{\phi}_{i,k}$ corresponding to $\mathcal{M}_b$.

Expectation Propagation using Mini-batches

Consider a **minibatch** of data $\mathcal{M}_b$:

1. Refine in parallel all approximate factors $\tilde{\phi}_{i,k}$ corresponding to $\mathcal{M}_b$.
2. Reconstruct the posterior approximation q.

Expectation Propagation using Mini-batches

Consider a **minibatch** of data $\mathcal{M}_b$:

1. Refine in parallel all approximate factors $\tilde{\phi}_{i,k}$ corresponding to $\mathcal{M}_b$.
2. Reconstruct the posterior approximation q.
3. Get a noisy estimate of the grad of $\log Z_q$ w.r.t to each ξ_j^k and $\overline{x}_{i,d}^k$.

Expectation Propagation using Mini-batches

Consider a **minibatch** of data $\mathcal{M}_b$:

1. Refine in parallel all approximate factors $\tilde{\phi}_{i,k}$ corresponding to $\mathcal{M}_b$.
2. Reconstruct the posterior approximation q.
3. Get a noisy estimate of the grad of $\log Z_q$ w.r.t to each ξ_j^k and $\overline{x}_{i,d}^k$.
4. Update all model hyper-parameters.

Expectation Propagation using Mini-batches

Consider a **minibatch** of data $\mathcal{M}_b$:

1. Refine in parallel all approximate factors $\tilde{\phi}_{i,k}$ corresponding to $\mathcal{M}_b$.
2. Reconstruct the posterior approximation q.
3. Get a noisy estimate of the grad of $\log Z_q$ w.r.t to each ξ_j^k and $\overline{x}_{i,d}^k$.
4. Update all model hyper-parameters.
5. Reconstruct the posterior approximation q.

Expectation Propagation using Mini-batches

Consider a **minibatch** of data $\mathcal{M}_b$:

1. Refine in parallel all approximate factors $\tilde{\phi}_{i,k}$ corresponding to $\mathcal{M}_b$.
2. Reconstruct the posterior approximation q.
3. Get a noisy estimate of the grad of $\log Z_q$ w.r.t to each ξ_j^k and $\overline{x}_{i,d}^k$.
4. Update all model hyper-parameters.
5. Reconstruct the posterior approximation q.

If $|\mathcal{M}_b| < M$ the **cost** is $\mathcal{O}(CM^3)$. **Memory** cost is $\mathcal{O}(NCM)$.

Stochastic Expectation Propagation

Li et al., 2015 suggest to store in memory only the **product** of the $\tilde{\phi}_i^k$:

$$\tilde{\phi} = \prod_{i=1}^{N} \prod_{k \neq y_i} \tilde{\phi}_i^k$$

Stochastic Expectation Propagation

Li et al., 2015 suggest to store in memory only the **product** of the $\tilde{\phi}_i^k$:

$$\tilde{\phi} = \prod_{i=1}^{N} \prod_{k \neq y_i} \tilde{\phi}_i^k$$

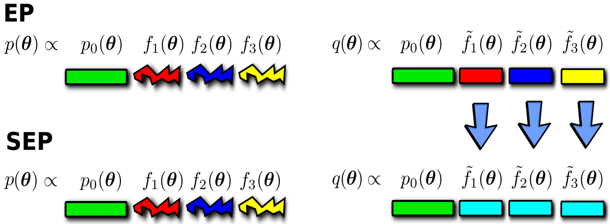
The **cavity distribution** is computed as $q^{\setminus i,k} \propto q / \tilde{\phi}^{\frac{1}{N_{\text{factors}}}}$.

Stochastic Expectation Propagation

Li et al., 2015 suggest to store in memory only the **product** of the $\tilde{\phi}_i^k$:

$$\tilde{\phi} = \prod_{i=1}^{N} \prod_{k \neq y_i} \tilde{\phi}_i^k$$

The **cavity distribution** is computed as $q^{\setminus i,k} \propto q / \tilde{\phi}^{\frac{1}{N_{\text{factors}}}}$.

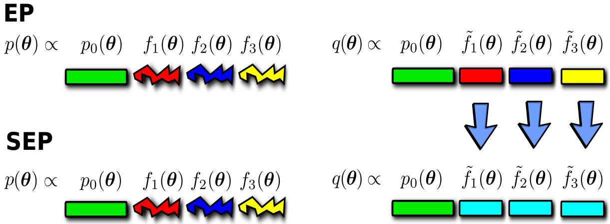


Stochastic Expectation Propagation

Li et al., 2015 suggest to store in memory only the **product** of the $\tilde{\phi}_i^k$:

$$\tilde{\phi} = \prod_{i=1}^{N} \prod_{k \neq y_i} \tilde{\phi}_i^k$$

The **cavity distribution** is computed as $q^{\setminus i,k} \propto q / \tilde{\phi}^{\frac{1}{N_{\text{factors}}}}$.



The **memory cost is reduced** to $\mathcal{O}(CM^2)$.

Baseline Method: Generalized FITC Approximation

- The **same** likelihood as the proposed method (Kim & Ghahramani, 2006).

Baseline Method: Generalized FITC Approximation

- The **same** likelihood as the proposed method (Kim & Ghahramani, 2006).

- **Original GFITC formulation** (Naish-Guzman & Hoden, 2008).

Baseline Method: Generalized FITC Approximation

- The **same** likelihood as the proposed method (Kim & Ghahramani, 2006).
- **Original GFITC formulation** (Naish-Guzman & Hoden, 2008).
- **Key difference**: The latent variables corresponding to the inducing points $\bar{\mathbf{f}}$ are **marginalized out** to obtain an approximate prior:

$$p(\mathbf{f}) = \int p(\mathbf{f}|\bar{\mathbf{f}})p(\bar{\mathbf{f}})d\bar{\mathbf{f}} \approx \prod_{k=1}^{C} \mathcal{N}\left(\mathbf{f}^k|\mathbf{0}, \mathbf{Q}_{NN}^k - \text{diag}\left(\mathbf{K}_{NN}^k - \mathbf{Q}_{NN}^k\right)\right)$$

Baseline Method: Generalized FITC Approximation

- The **same** likelihood as the proposed method (Kim & Ghahramani, 2006).

- **Original GFITC formulation** (Naish-Guzman & Hoden, 2008).

- **Key difference**: The latent variables corresponding to the inducing points $\bar{\mathbf{f}}$ are **marginalized out** to obtain an approximate prior:

$$p(\mathbf{f}) = \int p(\mathbf{f}|\bar{\mathbf{f}})p(\bar{\mathbf{f}})d\bar{\mathbf{f}} \approx \prod_{k=1}^{C} \mathcal{N}\left(\mathbf{f}^k|\mathbf{0}, \mathbf{Q}_{NN}^k - \text{diag}\left(\mathbf{K}_{NN}^k - \mathbf{Q}_{NN}^k\right)\right)$$

- Training **costs** $\mathcal{O}(CNM^2)$. **Does not allow** for scalable training!

UCI Repository datasets

Initial comparison on **small datasets** and **batch training**.

Dataset	#Instances	#Attributes	#Classes
Glass	214	9	6
New-thyroid	215	5	3
Satellite	6435	36	6
Svmguide2	391	20	3
Vehicle	846	18	4
Vowel	540	10	6
Waveform	1000	21	3
Wine	178	13	3

UCI Repository (test error)

	Problem	GFITC	EP	SEP	VI
M = 5%	Glass	**0.23** ± **0.02**	0.31 ± 0.02	0.31 ± 0.02	0.35 ± 0.02
	New-thyroid	**0.02** ± **0.01**	0.04 ± 0.01	0.02 ± 0.01	0.03 ± 0.01
	Satellite	0.12 ± 0.01	**0.11** ± **0.01**	0.12 ± 0.01	0.12 ± 0.01
	Svmguide2	0.2 ± 0.01	0.2 ± 0.01	0.2 ± 0.02	**0.19** ± **0.01**
	Vehicle	0.17 ± 0.01	0.17 ± 0.01	**0.16** ± **0.01**	0.17 ± 0.01
	Vowel	**0.05** ± **0.01**	0.09 ± 0.01	0.09 ± 0.01	0.06 ± 0.01
	Waveform	0.17 ± 0.01	**0.15** ± **0.01**	0.16 ± 0.01	0.17 ± 0.01
	Wine	0.03 ± 0.01	**0.03** ± **0.01**	0.03 ± 0.01	0.04 ± 0.01
	Avg. Rank	**2.24** ± **0.07**	2.33 ± 0.07	2.61 ± 0.06	2.82 ± 0.08
	Avg. Time	131 ± 3.11	53.8 ± 0.19	**48.5** ± **0.97**	157 ± 0.59
M = 10%	Glass	**0.2** ± **0.01**	0.29 ± 0.02	0.3 ± 0.02	0.35 ± 0.02
	New-thyroid	0.03 ± 0.01	**0.02** ± **0.01**	0.03 ± 0.01	0.03 ± 0.01
	Satellite	0.11 ± 0.01	**0.11** ± **0.01**	0.12 ± 0.01	0.12 ± 0.01
	Svmguide2	0.19 ± 0.02	0.2 ± 0.02	0.2 ± 0.02	**0.17** ± **0.02**
	Vehicle	0.17 ± 0.01	0.16 ± 0.01	0.16 ± 0.01	**0.15** ± **0.01**
	Vowel	**0.03** ± **0.01**	0.05 ± 0.01	0.06 ± 0.01	0.06 ± 0.01
	Waveform	0.17 ± 0.01	**0.16** ± **0.01**	0.16 ± 0.01	0.18 ± 0.01
	Wine	0.04 ± 0.01	**0.02** ± **0.01**	0.03 ± 0.01	0.03 ± 0.01
	Avg. Rank	2.4 ± 0.08	**2.21** ± **0.07**	2.62 ± 0.06	2.76 ± 0.08
	Avg. Time	264 ± 6.91	102 ± 0.64	**96.6** ± **1.99**	179 ± 0.78
M = 20%	Glass	**0.2** ± **0.02**	0.28 ± 0.02	0.28 ± 0.02	0.36 ± 0.02
	New-thyroid	0.03 ± 0.01	0.02 ± 0.01	**0.02** ± **0.01**	0.03 ± 0.01
	Satellite	0.11 ± 0.01	**0.11** ± **0.01**	0.12 ± 0.01	0.11 ± 0.01
	Svmguide2	0.2 ± 0.01	0.19 ± 0.01	0.2 ± 0.02	**0.19** ± **0.02**
	Vehicle	0.17 ± 0.01	0.16 ± 0.01	0.16 ± 0.01	**0.15** ± **0.01**
	Vowel	**0.03** ± **0.01**	0.03 ± 0.01	0.05 ± 0.01	0.03 ± 0.01
	Waveform	0.17 ± 0.01	**0.16** ± **0.01**	0.17 ± 0.01	0.18 ± 0.01
	Wine	0.04 ± 0.01	**0.01** ± **0.01**	0.03 ± 0.01	0.03 ± 0.01
	Avg. Rank	2.48 ± 0.08	**2.06** ± **0.07**	2.69 ± 0.07	2.77 ± 0.08
	Avg. Time	683 ± 17.3	228 ± 0.78	**216** ± **2.88**	248 ± 0.66

UCI Repository (test error)

	Problem	GFITC	EP	SEP	VI
M = 5%	Glass	**0.23 ± 0.02**	0.31 ± 0.02	0.31 ± 0.02	0.35 ± 0.02
	New-thyroid	**0.02 ± 0.01**	0.04 ± 0.01	0.02 ± 0.01	0.03 ± 0.01
	Satellite	0.12 ± 0.01	**0.11 ± 0.01**	0.12 ± 0.01	0.12 ± 0.01
	Svmguide2	0.2 ± 0.01	0.2 ± 0.01	0.2 ± 0.02	**0.19 ± 0.01**
	Vehicle	0.17 ± 0.01	0.17 ± 0.01	**0.16 ± 0.01**	0.17 ± 0.01
	Vowel	**0.05 ± 0.01**	0.09 ± 0.01	0.09 ± 0.01	0.06 ± 0.01
	Waveform	0.17 ± 0.01	**0.15 ± 0.01**	0.16 ± 0.01	0.17 ± 0.01
	Wine	0.03 ± 0.01	**0.03 ± 0.01**	0.03 ± 0.01	0.04 ± 0.01
	Avg. Rank	**2.24 ± 0.07**	2.33 ± 0.07	2.61 ± 0.06	2.82 ± 0.08
	Avg. Time	131 ± 3.11	53.8 ± 0.19	**48.5 ± 0.97**	157 ± 0.59
M = 10%	Glass	**0.2 ± 0.01**	0.29 ± 0.02	0.3 ± 0.02	0.35 ± 0.02
	New-thyroid	0.03 ± 0.01	**0.02 ± 0.01**	0.03 ± 0.01	0.03 ± 0.01
	Satellite	0.11 ± 0.01	**0.11 ± 0.01**	0.12 ± 0.01	0.12 ± 0.01
	Svmguide2	0.19 ± 0.02	0.2 ± 0.02	0.2 ± 0.02	**0.17 ± 0.02**
	Vehicle	0.17 ± 0.01	0.16 ± 0.01	0.16 ± 0.01	**0.15 ± 0.01**
	Vowel	**0.03 ± 0.01**	0.05 ± 0.01	0.06 ± 0.01	0.06 ± 0.01
	Waveform	0.17 ± 0.01	**0.16 ± 0.01**	0.16 ± 0.01	0.18 ± 0.01
	Wine	0.04 ± 0.01	**0.02 ± 0.01**	0.03 ± 0.01	0.03 ± 0.01
	Avg. Rank	2.4 ± 0.08	**2.21 ± 0.07**	2.62 ± 0.06	2.76 ± 0.08
	Avg. Time	264 ± 6.91	102 ± 0.64	**96.6 ± 1.99**	179 ± 0.78
M = 20%	Glass	**0.2 ± 0.02**	0.28 ± 0.02	0.28 ± 0.02	0.36 ± 0.02
	New-thyroid	0.03 ± 0.01	0.02 ± 0.01	**0.02 ± 0.01**	0.03 ± 0.01
	Satellite	0.11 ± 0.01	**0.11 ± 0.01**	0.12 ± 0.01	0.11 ± 0.01
	Svmguide2	0.2 ± 0.01	0.19 ± 0.01	0.2 ± 0.02	**0.19 ± 0.02**
	Vehicle	0.17 ± 0.01	0.16 ± 0.01	0.16 ± 0.01	**0.15 ± 0.01**
	Vowel	**0.03 ± 0.01**	0.03 ± 0.01	0.05 ± 0.01	0.03 ± 0.01
	Waveform	0.17 ± 0.01	**0.16 ± 0.01**	0.17 ± 0.01	0.18 ± 0.01
	Wine	0.04 ± 0.01	**0.01 ± 0.01**	0.03 ± 0.01	0.03 ± 0.01
	Avg. Rank	2.48 ± 0.08	**2.06 ± 0.07**	2.69 ± 0.07	2.77 ± 0.08
	Avg. Time	683 ± 17.3	228 ± 0.78	**216 ± 2.88**	248 ± 0.66

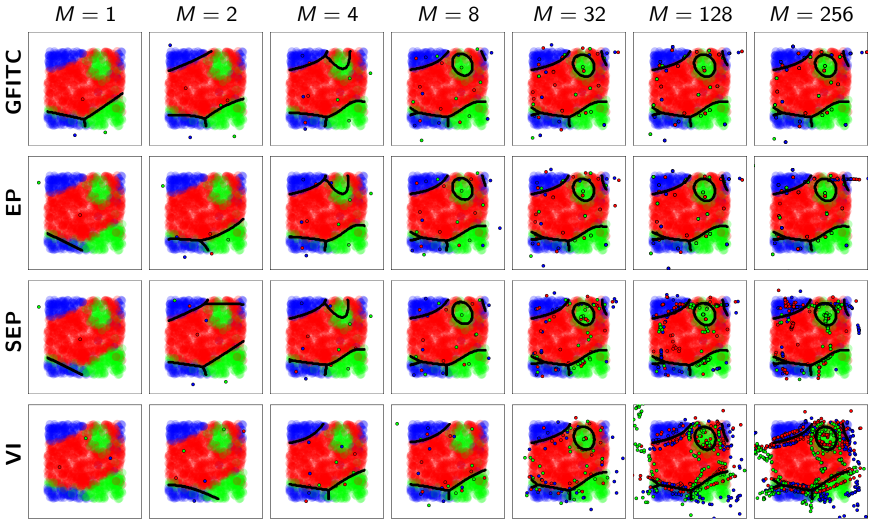
UCI Repository (negative test log-likelihood)

	Problem	GFITC	EP	SEP	VI
M = 5%	Glass	**0.61** ± **0.05**	0.78 ± 0.06	0.77 ± 0.07	2.45 ± 0.14
	New-thyroid	**0.06** ± **0.01**	0.11 ± 0.03	0.06 ± 0.01	0.09 ± 0.02
	Satellite	0.33 ± 0.01	**0.31** ± **0.01**	0.33 ± 0.01	0.61 ± 0.01
	Svmguide2	**0.63** ± **0.06**	0.63 ± 0.06	0.67 ± 0.06	1.03 ± 0.08
	Vehicle	**0.32** ± **0.01**	0.34 ± 0.02	0.34 ± 0.02	0.76 ± 0.05
	Vowel	**0.16** ± **0.01**	0.25 ± 0.01	0.25 ± 0.01	0.41 ± 0.05
	Waveform	0.42 ± 0.01	**0.36** ± **0.01**	0.39 ± 0.01	0.89 ± 0.02
	Wine	0.08 ± 0.02	**0.07** ± **0.01**	0.08 ± 0.01	0.08 ± 0.02
	Avg. Rank	**1.92** ± **0.07**	2.09 ± 0.07	2.46 ± 0.06	3.52 ± 0.08
	Avg. Time	131 ± 3.11	53.8 ± 0.19	**48.5** ± **0.97**	157 ± 0.59
M = 10%	Glass	**0.58** ± **0.05**	0.74 ± 0.06	0.79 ± 0.07	2.18 ± 0.14
	New-thyroid	0.07 ± 0.01	0.06 ± 0.01	0.06 ± 0.01	**0.05** ± **0.01**
	Satellite	0.34 ± 0.01	**0.30** ± **0.01**	0.34 ± 0.01	0.58 ± 0.01
	Svmguide2	**0.67** ± **0.05**	0.67 ± 0.05	0.74 ± 0.07	0.90 ± 0.10
	Vehicle	**0.33** ± **0.01**	0.33 ± 0.02	0.34 ± 0.02	0.72 ± 0.04
	Vowel	**0.14** ± **0.01**	0.19 ± 0.01	0.19 ± 0.01	0.30 ± 0.04
	Waveform	0.42 ± 0.01	**0.36** ± **0.01**	0.41 ± 0.01	0.85 ± 0.01
	Wine	0.07 ± 0.01	**0.06** ± **0.01**	0.07 ± 0.01	0.07 ± 0.01
	Avg. Rank	2.11 ± 0.08	**2.01** ± **0.08**	2.58 ± 0.07	3.31 ± 0.1
	Avg. Time	264 ± 6.91	102 ± 0.64	**96.6** ± **1.99**	179 ± 0.78
M = 20%	Glass	**0.6** ± **0.07**	0.75 ± 0.06	0.81 ± 0.07	2.30 ± 0.15
	New-thyroid	0.07 ± 0.01	0.06 ± 0.01	**0.05** ± **0.01**	0.05 ± 0.01
	Satellite	0.34 ± 0.01	**0.30** ± **0.01**	0.36 ± 0.01	0.53 ± 0.01
	Svmguide2	0.67 ± 0.05	**0.65** ± **0.06**	0.74 ± 0.07	0.94 ± 0.08
	Vehicle	0.33 ± 0.01	**0.33** ± **0.02**	0.34 ± 0.02	0.63 ± 0.04
	Vowel	**0.12** ± **0.01**	0.16 ± 0.01	0.18 ± 0.01	0.15 ± 0.03
	Waveform	0.43 ± 0.01	**0.37** ± **0.01**	0.45 ± 0.01	0.80 ± 0.01
	Wine	0.07 ± 0.01	**0.05** ± **0.01**	0.06 ± 0.01	0.06 ± 0.02
	Avg. Rank	2.17 ± 0.07	**1.91** ± **0.07**	2.68 ± 0.06	3.23 ± 0.1
	Avg. Time	683 ± 17.3	228 ± 0.78	**216** ± **2.88**	248 ± 0.66

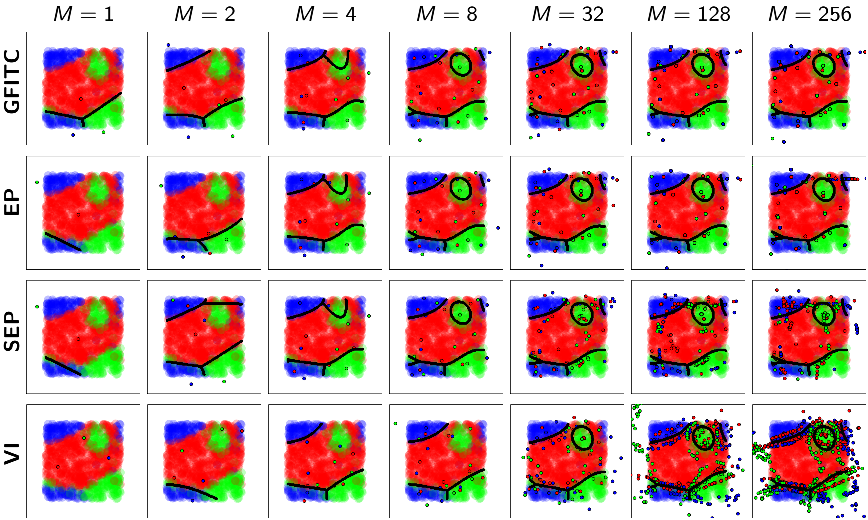
UCI Repository (negative test log-likelihood)

	Problem	GFITC	EP	SEP	VI
M = 5%	Glass	**0.61 ± 0.05**	0.78 ± 0.06	0.77 ± 0.07	2.45 ± 0.14
	New-thyroid	**0.06 ± 0.01**	0.11 ± 0.03	0.06 ± 0.01	0.09 ± 0.02
	Satellite	0.33 ± 0.01	**0.31 ± 0.01**	0.33 ± 0.01	0.61 ± 0.01
	Svmguide2	**0.63 ± 0.06**	0.63 ± 0.06	0.67 ± 0.06	1.03 ± 0.08
	Vehicle	**0.32 ± 0.01**	0.34 ± 0.02	0.34 ± 0.02	0.76 ± 0.05
	Vowel	**0.16 ± 0.01**	0.25 ± 0.01	0.25 ± 0.01	0.41 ± 0.05
	Waveform	0.42 ± 0.01	**0.36 ± 0.01**	0.39 ± 0.01	0.89 ± 0.02
	Wine	0.08 ± 0.02	**0.07 ± 0.01**	0.08 ± 0.01	0.08 ± 0.02
	Avg. Rank	**1.92 ± 0.07**	2.09 ± 0.07	2.46 ± 0.06	3.52 ± 0.08
	Avg. Time	131 ± 3.11	53.8 ± 0.19	**48.5 ± 0.97**	157 ± 0.59
M = 10%	Glass	**0.58 ± 0.05**	0.74 ± 0.06	0.79 ± 0.07	2.18 ± 0.14
	New-thyroid	0.07 ± 0.01	0.06 ± 0.01	0.06 ± 0.01	**0.05 ± 0.01**
	Satellite	0.34 ± 0.01	**0.30 ± 0.01**	0.34 ± 0.01	0.58 ± 0.01
	Svmguide2	**0.67 ± 0.05**	0.67 ± 0.05	0.74 ± 0.07	0.90 ± 0.10
	Vehicle	**0.33 ± 0.01**	0.33 ± 0.02	0.34 ± 0.02	0.72 ± 0.04
	Vowel	**0.14 ± 0.01**	0.19 ± 0.01	0.19 ± 0.01	0.30 ± 0.04
	Waveform	0.42 ± 0.01	**0.36 ± 0.01**	0.41 ± 0.01	0.85 ± 0.01
	Wine	0.07 ± 0.01	**0.06 ± 0.01**	0.07 ± 0.01	0.07 ± 0.01
	Avg. Rank	2.11 ± 0.08	**2.01 ± 0.08**	2.58 ± 0.07	3.31 ± 0.1
	Avg. Time	264 ± 6.91	102 ± 0.64	**96.6 ± 1.99**	179 ± 0.78
M = 20%	Glass	**0.6 ± 0.07**	0.75 ± 0.06	0.81 ± 0.07	2.30 ± 0.15
	New-thyroid	0.07 ± 0.01	0.06 ± 0.01	**0.05 ± 0.01**	0.05 ± 0.01
	Satellite	0.34 ± 0.01	**0.30 ± 0.01**	0.36 ± 0.01	0.53 ± 0.01
	Svmguide2	0.67 ± 0.05	**0.65 ± 0.06**	0.74 ± 0.07	0.94 ± 0.08
	Vehicle	0.33 ± 0.01	**0.33 ± 0.02**	0.34 ± 0.02	0.63 ± 0.04
	Vowel	**0.12 ± 0.01**	0.16 ± 0.01	0.18 ± 0.01	0.15 ± 0.03
	Waveform	0.43 ± 0.01	**0.37 ± 0.01**	0.45 ± 0.01	0.80 ± 0.01
	Wine	0.07 ± 0.01	**0.05 ± 0.01**	0.06 ± 0.01	0.06 ± 0.02
	Avg. Rank	2.17 ± 0.07	**1.91 ± 0.07**	2.68 ± 0.06	3.23 ± 0.1
	Avg. Time	683 ± 17.3	228 ± 0.78	**216 ± 2.88**	248 ± 0.66

Inducing Point Placement Analysis

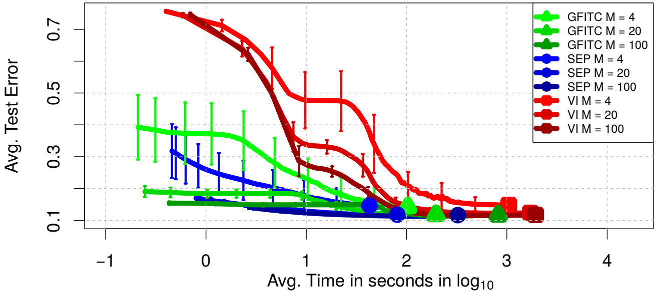


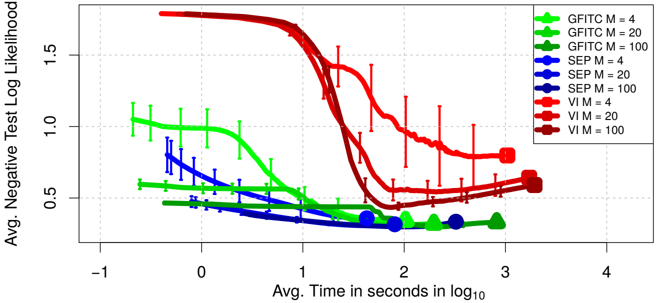
Inducing Point Placement Analysis



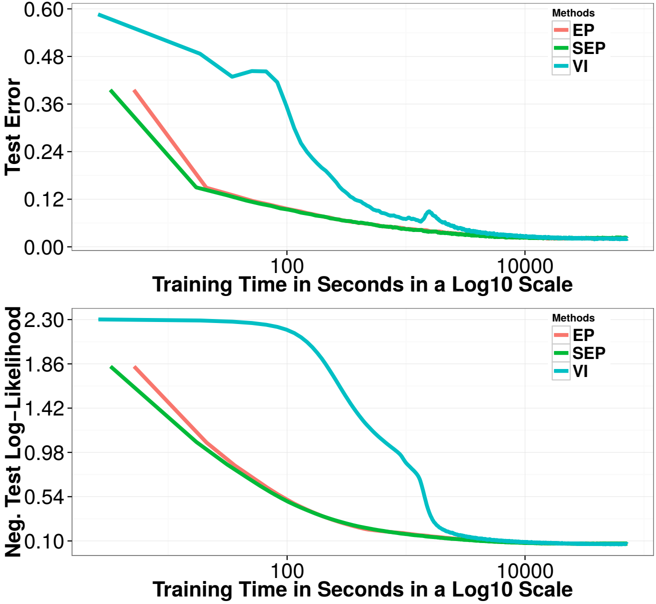
EP based methods perform **inducing point pruning** (Bauer et al., 2016)!

Performance in Terms of Time (Satellite Dataset)





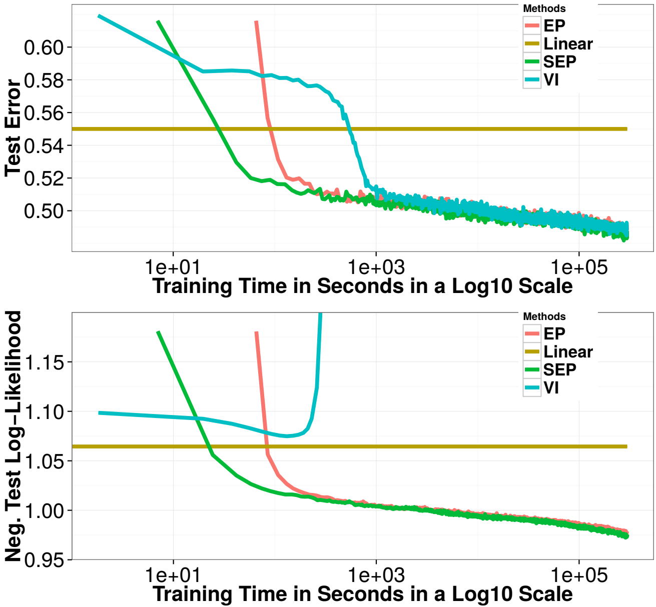
Minibatch Training: MNIST Dataset $M = 200$



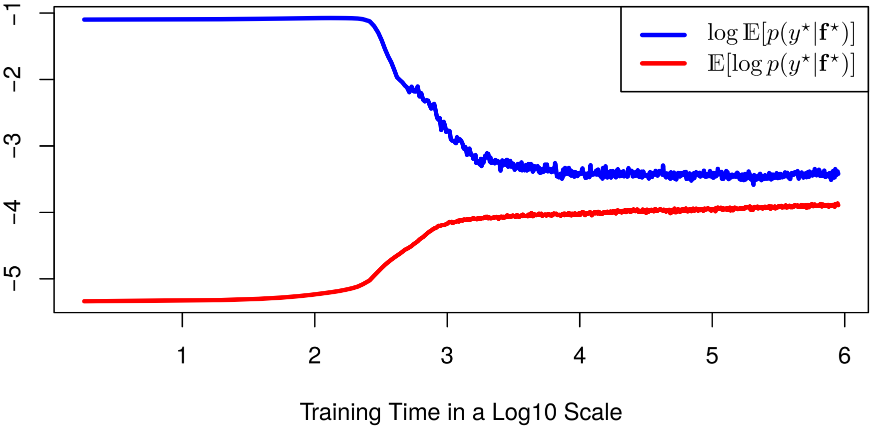
Minibatch Training: MNIST Dataset $M = 200$

Method	Test Error in %	Neg. Test Log-Likelihood
EP	2.10	0.0735
SEP	2.08	0.0725
VI	2.02	0.0682

Minibatch Training: Airline-delays $M = 200$



Minibatch Training: Airline-delays $M = 200$



Conclusions

- EP method for **multi-class** classification using GPs.

Conclusions

- EP method for **multi-class** classification using GPs.
- **Efficient** training and memory usage with cost $\mathcal{O}(CM^3)$.

Conclusions

- EP method for **multi-class** classification using GPs.
- **Efficient** training and memory usage with cost $\mathcal{O}(CM^3)$.
- Extensive **experimental comparison** with related methods.

Conclusions

- EP method for **multi-class** classification using GPs.
- **Efficient** training and memory usage with cost $\mathcal{O}(CM^3)$.
- Extensive **experimental comparison** with related methods.
- SEP is slightly **faster** than VI and is **quadrature free**.

Conclusions

- EP method for **multi-class** classification using GPs.
- **Efficient** training and memory usage with cost $\mathcal{O}(CM^3)$.
- Extensive **experimental comparison** with related methods.
- SEP is slightly **faster** than VI and is **quadrature free**.
- EP methods carry out inducing point **pruning**.

Conclusions

- EP method for **multi-class** classification using GPs.
- **Efficient** training and memory usage with cost $\mathcal{O}(CM^3)$.
- Extensive **experimental comparison** with related methods.
- SEP is slightly **faster** than VI and is **quadrature free**.
- EP methods carry out inducing point **pruning**.
- VI sometimes gives **bad test log-likelihoods**.

Conclusions

- EP method for **multi-class** classification using GPs.
- **Efficient** training and memory usage with cost $\mathcal{O}(CM^3)$.
- Extensive **experimental comparison** with related methods.
- SEP is slightly **faster** than VI and is **quadrature free**.
- EP methods carry out inducing point **pruning**.
- VI sometimes gives **bad test log-likelihoods**.

Thank you for your attention!

References

- Bauer, M., van der Wilk, M., and Rasmussen, C. E. Understanding probabilistic sparse Gaussian process approximations. NIPS 29, pp. 1533-1541. 2016.
- Chai, K. M. A. Variational multinomial logit Gaussian process. JMLR, 13:1745-1808, 2012.
- Girolami, M. and Rogers, S. Variational Bayesian multinomial probit regression with Gaussian process priors. Neural Computation, 18:1790-1817, 2006.
- Hensman, J., Matthews, A. G., Filippone, M., and Ghahramani, Z. MCMC for variationally sparse Gaussian processes. NIPS 28, pp. 1648-1656. 2015.
- Hernández-Lobato, D. and Hernández-Lobato, J. M. Scal- able Gaussian process classification via expectation propagation. AISTATS, pp. 168-176, 2016.
- Kim, H.-C. and Ghahramani, Z. Bayesian Gaussian process classification with the EM-EP algorithm. IEEE PAMI, 28, 1948-1959, 2006.
- Li, Y., Hernandez-Lobato, J. M., and Turner, R. E. Stochas- tic expectation propagation. NIPS 28, pp. 2323-2331. 2015.
- Naish-Guzman, A. and Holden, S. The generalized FITC approximation. NIPS 20, pp. 1057-1064. 2008.
- Riihimäki, J., Jylänki, P., and Vehtari, A. Nested expectation propagation for Gaussian process classification with a multinomial probit likelihood. JMLR, 14, 75-109, 2013.
- Snelson, E. and Ghahramani, Z. Sparse Gaussian processes using pseudo-inputs. NIPS 18, pp. 1257-1264, 2006.
- Williams, C. K. I. and Barber, D. Bayesian classification with Gaussian processes. IEEE PAMI, 20,1342-1351, 1998.