

# Scalable Multi-Class Gaussian Process Classification via Data Augmentation

Théo Galy-Fajou<sup>1</sup>, Florian Wenzel<sup>1,2</sup>, Christian Donner<sup>1</sup> and Manfred Oppert<sup>1</sup>

<sup>1</sup>TU Berlin, <sup>2</sup>TU Kaiserslautern

galy-fajou@tu-berlin.de



## Main Points

- We propose a **new scalable multi-class Gaussian process classification** approach building on a novel **modified softmax likelihood function**.
- This likelihood allows for a latent variable augmentation that leads to a **conditionally conjugate model** and enables **efficient variational inference via block coordinate ascent updates**.
- **Sparse Gaussian Processes** with independent inducing points and kernel parameters.

## GP Multi-class problem

- $N$  data points  $\mathbf{X} = (\mathbf{x}_1, \dots, \mathbf{x}_N)$  with labels  $\mathbf{y} = (y_1, \dots, y_N)$ , where  $y_i \in \{1, \dots, C\}$  and  $C$  is the total number of classes.
- One latent GP prior for each class  $\mathbf{f} = (f^1, \dots, f^C)$ , where  $f^c \sim \text{GP}(0, k^c)$  and  $k^c$  is the corresponding kernel

One of the most used likelihood for multi-class problems is *softmax*:

$$p(y_i = k | \mathbf{f}_i) = \frac{\exp(f_i^k)}{\sum_{c=1}^C \exp(f_i^c)},$$

where  $f_i^c = f^c(\mathbf{x}_i)$ .

## Logistic-softmax likelihood

We introduce a modified version of the likelihood : **Logistic-softmax**

$$p(y_i = k | \mathbf{f}_i) = \frac{\sigma(f_i^k)}{\sum_{c=1}^C \sigma(f_i^c)}, \quad (1)$$

where  $\sigma(x) = (1 + \exp(-x))^{-1}$  is the logistic function.

Equivalent to softmax with  $h(f) = \log(\sigma(f))$ .

Reduces to the logistic likelihood when  $C = 2$  and setting the symmetry  $f^2 = -f^1$ .

## Augmentation procedure

This new likelihood allows us to reach a **full conditionally conjugate model** after a few augmentations:

### Augmentation 1 : Gamma Augmentation

We use the identity :  $\frac{1}{x} = \int_0^\infty \exp(-\lambda x) d\lambda$  on equation 1 and augment it with  $\lambda$ .

$$p(y_i = k | \mathbf{f}_i) = \sigma(f_i^k) \int_0^\infty \exp\left(-\lambda^i \sum_{c=1}^C \sigma(f_i^c)\right) d\lambda^i$$

$$\Rightarrow p(y_i = k, \lambda^i | \mathbf{f}_i) = \sigma(f_i^k) \prod_{c=1}^C \exp(-\lambda^i \sigma(f_i^c)) \quad (2)$$

### Augmentation 2 : Poisson Augmentation

We can now use the modification of the likelihood, since  $\sigma(x)$  is bounded and it leads to the property :  $\sigma(x) = 1 - \sigma(-x)$ . Additionally we use the definition of the exponential  $\exp(x) = \sum_{n=1}^\infty x^n / n!$

and augment our model:

$$p(y_i = k | \lambda^i, \mathbf{f}_i) = \sigma(f_i^k) \prod_{c=1}^C \exp(-\lambda^i (1 - \sigma(-f_i^c)))$$

$$= \sigma(f_i^k) \prod_{c=1}^C \sum_{n_c^i=1}^\infty \exp(-\lambda^i) \left( \lambda^i \sigma(-f_i^c)^{n_c^i} / n_c^i! \right)$$

$$\Rightarrow p(y_i = k, \{n_c^i\}_{c=1}^C | \lambda^i, \mathbf{f}_i) = \sigma(f_i^k) \prod_{c=1}^C \text{Po}(n_c^i | \lambda^i) \sigma(-f_i^c)^{n_c^i} \quad (3)$$

## Pólya-Gamma Augmentation

After managing to get rid of all exponential and sum terms we can now apply the Pólya-Gamma augmentation  $\sigma^n(x) = 2^{-n} \int_0^\infty \exp(\frac{nx}{2} - \frac{\omega x^2}{2}) \text{PG}(\omega | n, 0) d\omega$

$$p(y_i = k | \{n_c^i\}_{c=1}^C, \lambda^i, \mathbf{f}_i, \lambda^i) = \sigma(f_i^k) \prod_{c=1}^C \sigma(-f_i^c)^{n_c^i}$$

$$\Rightarrow p(y_i = k | \{\omega_c^i\}_{c=0}^C, \{n_c^i\}_{c=1}^C, \lambda^i, \mathbf{f}_i) = \quad (4)$$

$$\frac{1}{2} \exp\left(\frac{f_i^c}{2} - \frac{(f_i^c)^2}{2} \omega_0^i\right) \prod_{c=1}^C 2^{-n_c^i} \exp\left(-\frac{n_c^i f_i^c}{2} - \frac{(f_i^c)^2}{2} \omega_c^i\right)$$

## Inference

### Mean Field Approximation

We approximate the full posterior by a variational distribution

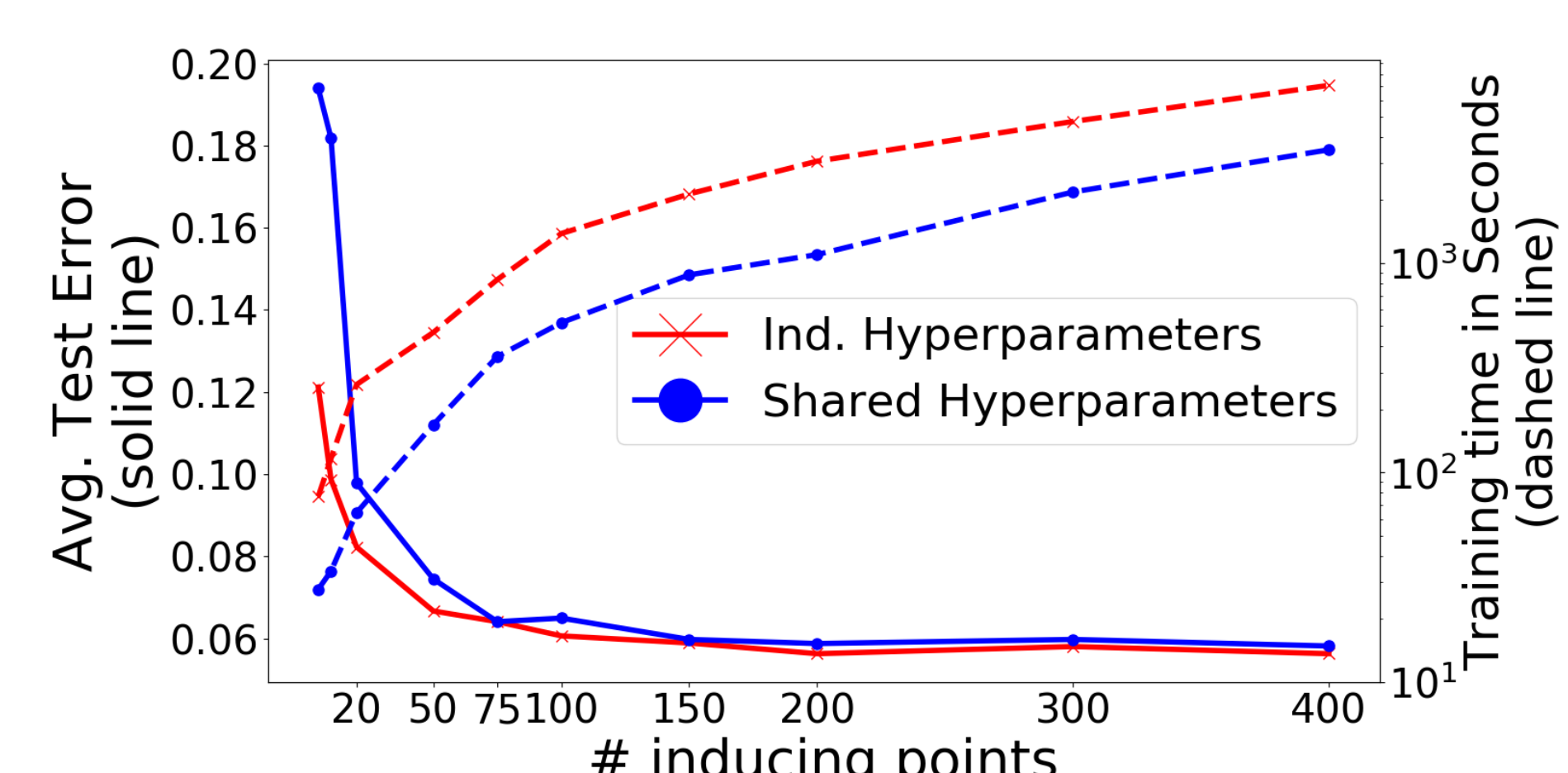
$$p(\mathbf{f}, \boldsymbol{\omega}, \mathbf{n}, \boldsymbol{\lambda} | \mathbf{y}) \approx \prod_{c=1}^C q(\mathbf{f}_c) \prod_{i=1}^N q(\lambda^i) q(n_c^i) q(\omega_c^i) \quad (5)$$

## Sparsity

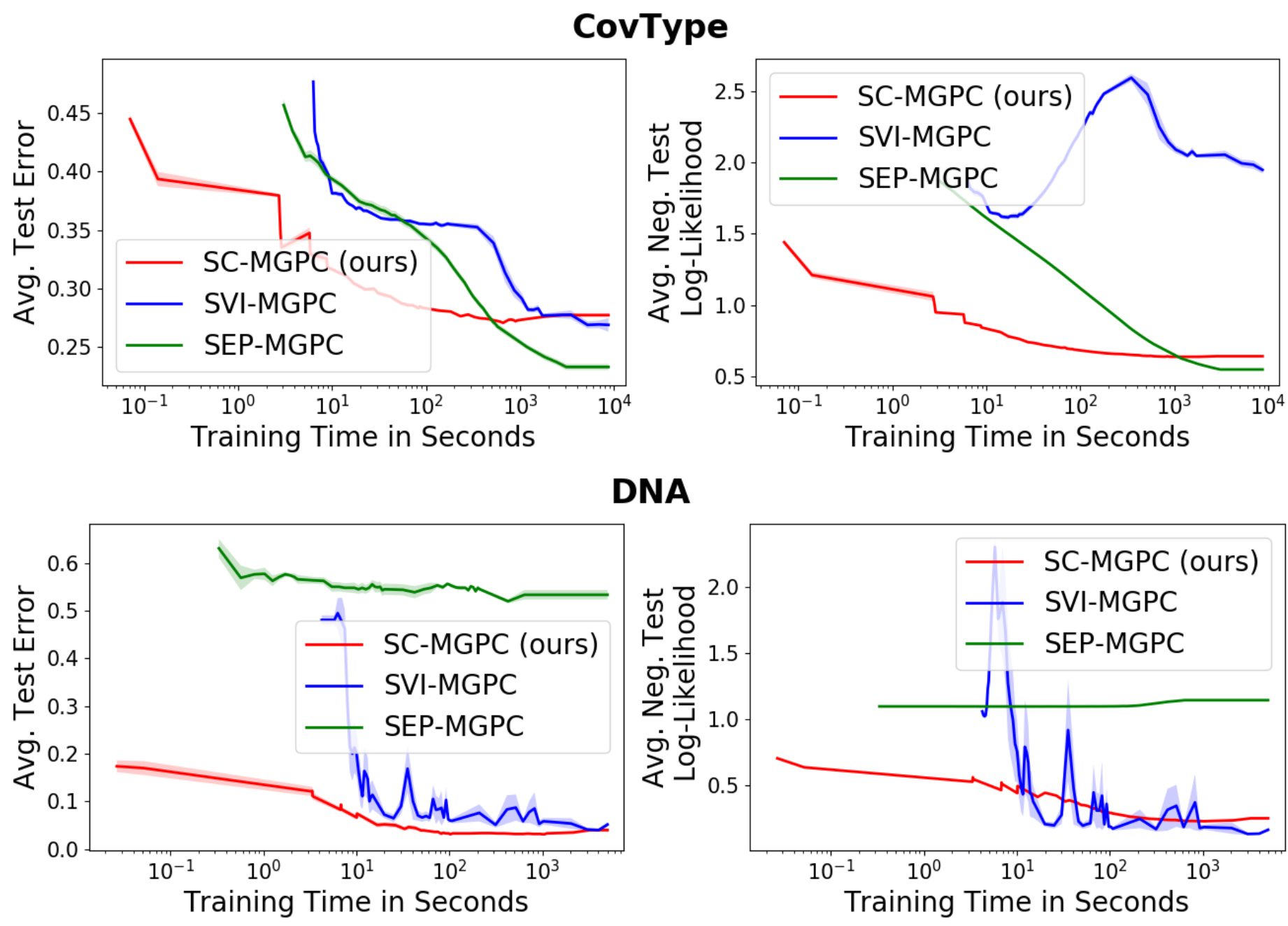
Augmentation with inducing points

## Experiments

### Independent vs Shared Prior among classes



## Convergence



## Forthcoming Research

- Improved hyperparameter optimization
- Class subsampling
- 

## References