Scalable Multi-Class Gaussian Process Classification via Data Augmentation

Théo Galy-Fajou[1], Florian Wenzel[1,2], Christian Donner[1] and Manfred Opper[1]

[1]TU Berlin, [2]TU Kaiserslautern

galy-fajou@tu-berlin.de

Technische Universität Berlin

TECHNISCHE UNIVERSITÄT KAISERSLAUTERN

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

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

$$p(y_i = k|\boldsymbol{f_i}) = \frac{\exp\left(f_i^k\right)}{\sum_{c=1}^C \exp\left(f_i^c\right)},$$

where $f_i^c = f^c(\boldsymbol{x_i})$.

Logistic-softmax likelihood

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

$$p(y_i = k|\boldsymbol{f_i}) = \frac{\sigma\left(f_i^k\right)}{\sum_{c=1}^C \sigma\left(f_i^c\right)}, \tag{1}$$

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 λ.

$$\begin{aligned} p(y_i = k|\boldsymbol{f_i}) =& \sigma\left(f_i^k\right) \int_0^\infty \exp\left(-\lambda^i \sum_{c=1}^C \sigma\left(f_i^c\right)\right) d\lambda^i \\ \Rightarrow p(y_i = k, \lambda^i|\boldsymbol{f_i}) =& \sigma\left(f_i^k\right) \prod_{c=1}^C \exp\left(-\lambda^i \sigma\left(f_i^c\right)\right) \end{aligned} \tag{2}$$

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:

$$\begin{aligned} p(y_i = k|\lambda^i, \boldsymbol{f_i}) =& \sigma\left(f_i^k\right) \prod_{c=1}^C \exp\left(-\lambda^i(1-\sigma\left(-f_i^c\right))\right) \\ =& \sigma\left(f_i^k\right) \prod_{c=1}^C \sum_{n_i^c=1}^\infty \exp(-\lambda^i) \left(\lambda^i \sigma\left(-f_i^c\right)^{n_c^i}/n_c^i!\right) \\ \Rightarrow p(y_i = k, \{n_c^i\}_{c=1}^C|\lambda^i, \boldsymbol{f_i}) =& \sigma\left(f_i^k\right) \prod_{c=1}^C \mathrm{Po}(n_c^i|\lambda^i) \sigma\left(-f_i^c\right)^{n_c^i} \end{aligned} \tag{3}$$

Pólya-Gamma Augmentation

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

$$\begin{aligned} p(y_i = k|\{n_c^i\}_{c=1}^C, \lambda^i, \boldsymbol{f_i}, \lambda^i) =& \sigma\left(f_i^k\right) \prod_{c=1}^C \sigma\left(-f_i^c\right)^{n_c^i} \\ \Rightarrow p(y_i = k|\{\omega_c^i\}_{c=0}^C, \{n_c^i\}_{c=1}^C, \lambda^i, \boldsymbol{f_i}) =& \end{aligned} \tag{4}$$

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

Inference

Mean Field Approximation

We approximate the full posterior by a variational distribution

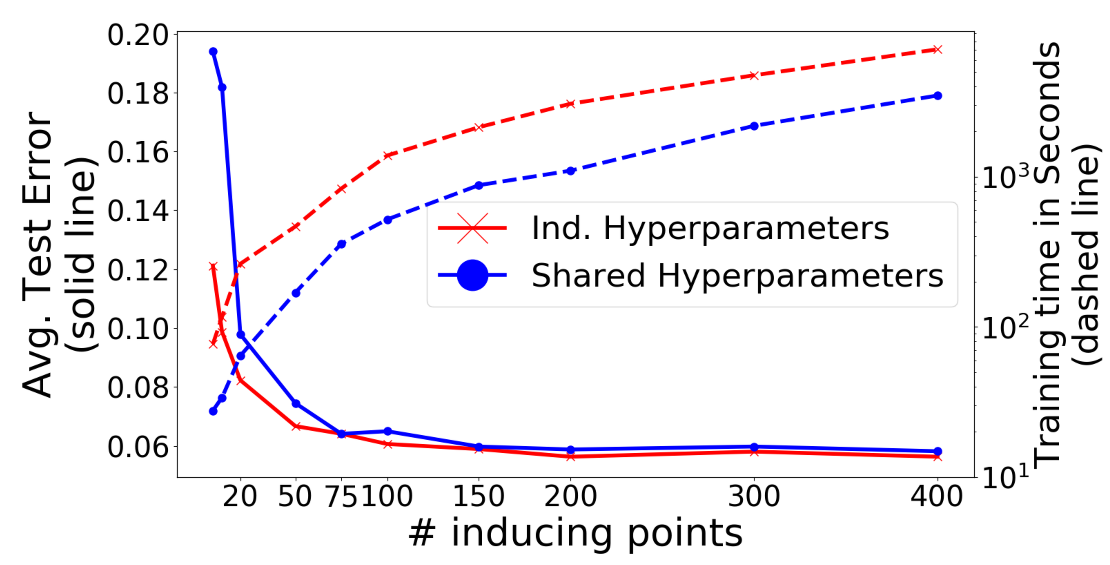
$$p(\boldsymbol{f}, \boldsymbol{\omega}, \boldsymbol{n}, \boldsymbol{\lambda}|\boldsymbol{y}) \approx \prod_{c=1}^C q(\boldsymbol{f_c}) \prod_{i=1}^N q(\lambda^i) q(n_c^i) q(\omega_c^i) \tag{5}$$

Sparsity

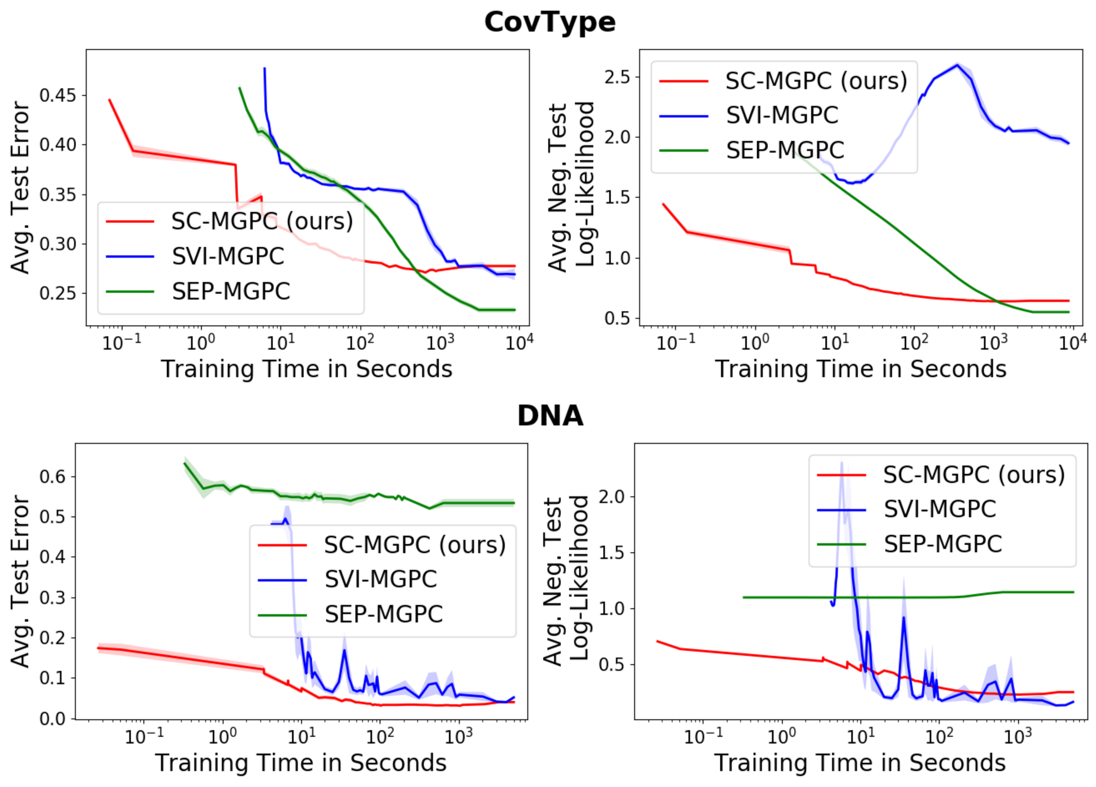
Augmentation with inducing points

Experiments

Independent vs Shared Prior among classes



Convergence



Forthcoming Research

- Improved hyperparameter optimization
- Class subsampling
-

References