A APPENDIX

A.1 Reparametrization of the Pólya-Gamma variables

By applying the augmentation of the sigmoid (8) to the augmented likelihood (7), we obtain the Pólya-Gamma augmented likelihood

$$p(y_i = k|\boldsymbol{f}_i, \lambda_i, \boldsymbol{n}_i, \tilde{\omega}_i, \boldsymbol{\omega}_i) = \frac{1}{2}\exp\left(\frac{f_i^k}{2} - \frac{(f_i^k)^2}{2}\tilde{\omega}_i\right) \times \prod_{c=1}^{C} 2^{-n_i^c}\exp\left(-\frac{n_i^c f_i^c}{2} - \frac{(f_i^c)^2}{2}\omega_i^c\right), \quad (10)$$

where we impose the prior distributions

$$p(\tilde{\omega}_i) = \mathrm{PG}(1,0)$$
$$p(\boldsymbol{\omega}_i|\boldsymbol{n}_i) = \prod_c^C \mathrm{PG}(\omega_i^c|n_i^c, 0).$$

We simplify this expression by combining all terms corresponding to the index k. To this end, we use a one hot-encoding of $\boldsymbol{y} \in \{0,\dots,C\}^N$ as $\boldsymbol{y'} \in \{0,1\}^{C\times N}$,

$$y'^c_i = \begin{cases} 1 \text{ for } y_i = c \\ 0 \text{ otherwise.} \end{cases}.$$

Building on the identity $\omega_1 + \omega_2 = \omega_3$ with $\omega_1 \sim \mathrm{PG}(b_1, c)$, $\omega_2 \sim \mathrm{PG}(b_2, c)$ and $\omega_3 \sim \mathrm{PG}(b_1 + b_2, c)$, we rewrite equation (10) as

$$p(y_i = k|\boldsymbol{f}_i, \lambda_i, \boldsymbol{n}_i, \boldsymbol{\omega}_i) = \prod_{c=1}^{C} 2^{-(y'^c_i + n_i^c)}\exp\left(\frac{(y'^c_i - n_i^c)f_i^c}{2} - \frac{(f_i^c)^2}{2}\omega_i^c\right),$$

where the terms corresponding to $\tilde{\boldsymbol{\omega}}$ are now absorbed into the terms corresponding to $\boldsymbol{\omega}$.

A.2 Block coordinate ascent (CAVI) updates

The variational distribution is $q(\boldsymbol{u}, \boldsymbol{\lambda}, \boldsymbol{n}, \boldsymbol{\omega}) = q(\boldsymbol{u})q(\boldsymbol{\lambda})q(\boldsymbol{\omega}, \boldsymbol{n})$ and the factors are

$$q(\boldsymbol{u}) = \prod_c^C \mathcal{N}(\boldsymbol{u}^c|\boldsymbol{\mu}^c, \Sigma^c),\ q(\boldsymbol{\lambda}) = \prod_i \mathrm{Ga}(\lambda_i|\alpha_i, \beta_i),$$
$$q(\boldsymbol{\omega}, \boldsymbol{n}) = \prod_{i,c} \mathrm{PG}(\omega_i^c|y'^c_i + n_i^c, b_i^c)\mathrm{Po}(n_i^c|\gamma_i^c).$$

In the CAVI scheme (Hoffman et al., 2013) each factor is iteratively updated by the following equation. Suppose we want to update the variational distribution corresponding to the latent variable $\boldsymbol{\theta} \in \{\boldsymbol{u}, \boldsymbol{\lambda}, \boldsymbol{n}, \boldsymbol{\omega}\}$. Let $\overline{\boldsymbol{\theta}}$ be the set of the other latent variables, then $q^*(\boldsymbol{\theta})$ is updated by

$$q^*(\boldsymbol{\theta}) \propto \exp\left(\mathbb{E}_{q(\overline{\boldsymbol{\theta}})}\left[\log p(\theta \mid \overline{\boldsymbol{\theta}})\right]\right). \quad (11)$$

Using this equation gives the closed-form update for each variational parameter.

$$\overline{f_i^c} = \sqrt{\mathbb{E}_{q(f^c)}\left[(f_i^c)^2\right]} = \sqrt{\widetilde{K}_{ii}^c + \kappa_i^c \Sigma^c \kappa_i^{c\top} + (\kappa_i^c \boldsymbol{\mu}^c)^\top \kappa_i^c \boldsymbol{\mu}^c}$$

$$\gamma_i^c = \frac{\exp\left(\psi(\alpha_i)\right)\exp\left(-\frac{\kappa_i^c \boldsymbol{\mu}^c}{2}\right)}{\beta_i \cosh\left(\frac{\overline{f_i^c}}{2}\right)} \quad (12)$$

$$\alpha_i = 1 + \sum_{c=1}^{C}\gamma_i^c, \quad \beta_i = C \quad (13)$$

$$b_i^c = \overline{f_i^c}, \quad (14)$$

$$\theta_i^c = \mathbb{E}_{q(\omega_i^c, n_i^c)}\left[\omega_i^c\right] = \frac{y'^c_i + \gamma_i^c}{2b_i^c}\tanh\frac{b_i^c}{2}$$

$$\boldsymbol{\mu}^c = \frac{1}{2}(\Sigma^c)^{-1}\kappa^{c\top}\left(\boldsymbol{y'}^c - \boldsymbol{\gamma}^c\right) \quad (15)$$

$$\Sigma^c = \left(\kappa^{c\top}\mathrm{diag}\left(\boldsymbol{\theta}^c\right)\kappa^c + (K_{mm}^c)^{-1}\right)^{-1}, \quad (16)$$

where $\psi(.)$ is the digamma function. When $\kappa\mu \ll 0$, equation (12) easily overflows. One can solve this problem by approximating $\exp(-0.5\kappa\mu)/\cosh(0.5\bar{f})$ with $\sigma(\kappa\mu)$ by neglecting the variance terms $\widetilde{K} + \kappa\Sigma\kappa^\top$ in $\bar{f}$.

Equation (12) and (13) shows a direct interdependence between α_i and γ_i^c. We use inner loop of alternating between updating both variables until convergence to solve the problem. We find that 5 iterations in the inner loop are enough.

Finally, if class subsampling (the extreme classification version of our algorithm Alg. 2) is used, α_i is approximated by

$$\alpha_i = 1 + \frac{C}{|\mathcal{K}|}\sum_{c\in\mathcal{K}}\gamma_i^c, \quad (17)$$

where C is the number of classes and $|\mathcal{K}|$ is the number of sub-sampled classes.

A.3 Subsampling the classes (extreme classification version)

The extreme classification version of our algorithm is presented in Alg. 2. In each iteration we only consider a minibatch of the classes $\mathcal{B} \subset \{1,\dots,C\}$ and the variational parameters b_i^c, α_i^c, μ^c, Σ^c (lines 13, 11, 18, 19 in Alg. 1) are only updated for $i \in \mathcal{B}$. The updates that are global w.r.t. the classes, i.e. λ_i and the hyperparameters

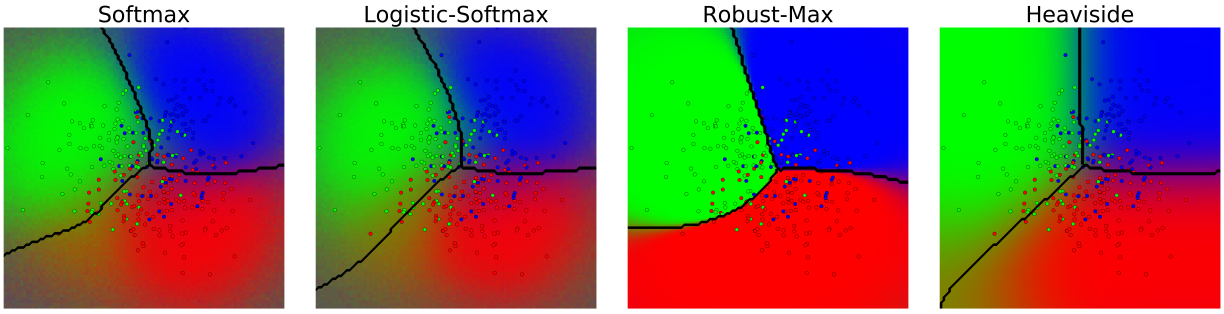

Figure 8: RGB representation of the predictive likelihood for a toy dataset as described in section 5.1 with variance $\sigma^2 = 0.5$. Each class is attributed a color channel (Red, Green, Blue) and predictive likelihoods are mapped into RGB values.

h (lines 11, 22) are now replaced by stochastic gradient updates.

Algorithm 2 Conjugate multi-class Gaussian process classification with class subsampling

Input: data $\boldsymbol{X}$,$\boldsymbol{y}$, minibatch size $|\mathcal{S}|$and $|\mathcal{B}|$
Output: variational posterior GPs $p(u^c|\mu^c, \Sigma^c)$
Set the learning rate schedules ρ_t, ρ_t^h appropriately
Initialize all variational parameters and hyperparameters
Select M inducing points locations (e.g. kMeans)
for iteration $t = 1, 2, \ldots$ **do**
    `# Sample minibatch:`
    Sample a minibatch of the data $\mathcal{S} \subset \{1, \ldots, N\}$
    Sample a set of labels $\mathcal{K} \subset \{1, \ldots, C\}$
    `# Local variational updates`
    for $i \in \mathcal{S}$ **do**
        Update $(\alpha_i, \gamma_i^c)_{c \in \mathcal{K}}$ (Eq. 12,17)
        for $c \in \mathcal{K}$ **do**
            Update b_i^c (Eq. 14)
        end for
    end for
    `# Global variational GP updates`
    for $c \in \mathcal{K}$ **do**
        $\mu^c \leftarrow (1 - \rho_t)\mu^c + \rho_t \hat{\mu}^c$ (Eq. 15)
        $\Sigma^c \leftarrow (1 - \rho_t)\Sigma^c + \rho_t \hat{\Sigma}^c$ (Eq. 16)
    end for
    `# Hyperparameter updates`
    Gradient step $h \leftarrow h + \rho_t^h \nabla_h \mathcal{L}$
end for

A.4 Visualization of the different likelihoods

To get a better intuition of the behavior of each likelihood, we visualize the prediction function of each method as a contour plot using the toy dataset from section 5.1. To visualize the predictive likelihood, we map the predictive values of each class to a RGB color channel (where each class corresponds to one color and mixing of colors indicates a contribution of multiple classes). A highly saturated color corresponds to a high confidence in the class prediction, while mixed colors indicate zones of transition between classes and lower confidence. The results are shown in Figure 8 for a toy dataset consisting of 500 points generated from a mixture of Gaussians with variance $\sigma^2 = 0.5$. As expected, the robust-max likelihood leads to extremely sharp decision boundaries and high confidences for all regions (even for the overlapping regions). The other likelihoods lead to better calibration resulting in soft boundaries and less confident predictions in the overlapping regions.

A.5 Convexity of the negative ELBO

In the following we prove that the negative ELBO ($-\mathcal{L}$) of our augmented model is convex in the global variational parameters μ^c and Σ^c. To prove this statement, we write the negative ELBO in terms of μ^c and Σ^c,

$$-\mathcal{L}(\mu^c, \Sigma^c) \overset{c}{=} \frac{1}{2}\left[\sum_{i=1}^{N}(y_i'^c - \gamma_i^c)\mu_i^c - \theta_i^c\left((\mu_i^c)^2 + \Sigma_{ii}^c\right)\right] \frac{1}{2}\left[\mu^{c\top} K^{-1}\mu^c + \mathrm{tr}(K^{-1}\Sigma^c) - \log|\Sigma^c|\right].$$

Differentiating twice in μ^c gives $\mathrm{diag}(\theta^c) + K^{-1}$ which is positive definite since $\theta_i^c > 0$ for all i and by definition of K. Therefore, the negative ELBO is convex in μ^c for all c.

Differentiating twice in Σ^c gives $(\Sigma^c)^{-1} \otimes (\Sigma^c)^{-1}$, where $\otimes$ is the Kroenecker product. This is again positive definite since $(\Sigma^c)^{-1}$ is positive definite and the Kroenecker product preserves positive definiteness. Therefore, the negative ELBO is also convex in Σ^c for all c.