GAUSSIAN PROCESS NEURONS

Anonymous authors
Paper under double-blind review

ABSTRACT

We propose a method to learn stochastic activation functions for use in probabilistic neural networks. First, we develop a framework to embed stochastic activation functions based on Gaussian processes in probabilistic neural networks. Second, we analytically derive expressions for the propagation of means and covariances in such a network, thus allowing for an efficient implementation and training without the need for sampling. Third, we show how to apply variational Bayesian inference to regularize and efficiently train this model. The resulting model can deal with uncertain inputs and implicitly provides an estimate of the confidence of its predictions. Like a conventional neural network it can scale to datasets of arbitrary size and be extended with convolutional and recurrent connections, if desired.

1 INTRODUCTION

The popularity of deep learning and the implied race for better accuracy and performance has lead to new research of the fundamentals of neural networks. Finding an optimal architecture often focusses on a hyperparameter search over the network architecture, regularization parameters, and one of a few standard activation functions: tanh, ReLU (Glorot et al. (2011)), maxout (Goodfellow et al. (2013)) ... Focussing on the latter, looking into activation functions has only taken off since Nair & Hinton (2010) introduced the rectified linear unit (ReLU), which were shown to produce significantly better results on image recognition tasks Krizhevsky et al. (2012). Maas et al. (2013) then introduced the leaky ReLU, which has a very small, but non-zero, slope for negative values. He et al. (2015) proposed the parameterized ReLU, by making the slope of the negative part of the leaky ReLU adaptable. It was trained as an additional parameter for each neuron alongside the weights of the neural network using stochastic gradient descent. Thus, the activation function was not treated as a fixed hyper-parameter anymore but as adaptable to training data. While the parameterized ReLU only has one parameter, this was generalized in Agostinelli et al. (2014a) to piecewise linear activation functions that can have an arbitrary (but fixed) number of points where the function changes it slope. This can be interpreted as a different parameterization of a Maxout network (Goodfellow et al. (2013)), in which each neuron takes the maximum over a set of different linear combinations of its inputs.

Instead of having a fixed parameter for the negative slope of the ReLU, Xu et al. (2015) introduced stochasticity into the activation function by sampling the value for the slope with each training iteration from a fixed uniform distribution. Clevert et al. (2015) and Klambauer et al. (2017) replaced the negative part of ReLUs with a scaled exponential function and showed that, under certain conditions, this leads to automatic renormalization of the inputs to the following layer and thereby simplifies the training of the neural networks, leading to an improvement in accuracy in various tasks.

Nearly fully adaptable activation functions have been proposed by Eisenach et al. (2016). The authors use a Fourier basis expansion to represent the activation function; thus with enough coefficients any (periodic) activation function can be represented. The coefficients of this expansion are trained as network parameters using stochastic gradient descent or extensions thereof.

Promoting a more general approach, Agostinelli et al. (2014b) proposed to learn the activation functions alongside the layer weights. Their adaptive piecewise linear units consist of a sum of hinge-shaped functions with parameters to control the hinges and the slopes of the linear segments. However, by construction the derivative of these activation functions is not continuous at the joints between two linear segments, which often leads to non-optimal optimizer performance.

To our knowledge, previous research on learning activation functions took place in a fully deterministic setting, i.e. deterministic activation functions were parameterized and included in the optimization of a conventional neural network. Here instead, we explore the setting of probabilistic activation functions embedded in a graphical model of random variables resembling the structure of a neural network. We develop the theory of Gaussian-process neurons and subsequently derive a lower-bound approximation using variational inference, in order to develop a computationally efficient version of the Gaussian Process neuron.

Notation

To define the model we will need to slice matrices along rows and columns. Given a matrix X, we will write $X_{i\star}$ to select all elements of the i-th row and $X_{\star j}$ to select all elements of the j-th column.

2 Gaussian Processes

Gaussian Processes (GPs) are nonparametric models that provide flexible probabilistic approaches for function estimation. A Gaussian Process (Rasmussen & Williams, 2006) defines a distribution over a function $f(x) \sim \mathcal{GP}(m(x), k(x, x'))$ where m is called the mean function and k is the covariance function. For S inputs $\boldsymbol{x} \in \mathbb{R}^{S \times N}$ of N dimensions the corresponding function values $\boldsymbol{f}$ with $f_i \triangleq f(X_{i\star})$ follow a multivariate normal distribution[1]

$$\boldsymbol{f} \sim \mathcal{N}(\boldsymbol{m}, K(X, X))$$

with mean vector $m_i \triangleq m(X_{i\star})$ and $K(X, X)$ is the covariance matrix defined by $(K(X, X))_{ij} \triangleq k(X_{i\star}, X_{j\star})$. In this work we use the zero mean function $m(x) = 0$ and the squared exponential (SE) covariance function with scalar inputs,

$$k(x, x') = \exp\left(-\frac{(x - x')^2}{2\,\nu^2}\right) , \tag{1}$$

where ν is called the length-scale and determines how similar function values of nearby inputs are according to the GP distribution. Since this covariance function is infinitely differentiable, all function samples from a GP using it are smooth functions.

3 Gaussian Process Neurons

We will first describe the fundamental, non-parametric model, which will be approximated in the following sections for efficient training and inference. Let the input to the l-th layer of Gaussian Process neurons (GPNs) be denoted by $X^{l-1} \in \mathbb{R}^{S \times N_{l-1}}$ where S is the number of data points (samples) and N_{l-1} is the number of input dimensions. A layer $l \in \{1, \dots, L\}$ of N_l GPNs indexed by $n \in \{1, \dots N_l\}$ is defined by the joint probability

$$\mathrm{P}(X^l_{\star n}, F^l_{\star n} \,|\, X^{l-1}) = \mathrm{P}(F^l_{\star n} \,|\, X^{l-1})\, \mathrm{P}(X^l_{\star n} \,|\, F^l_{\star n}) \tag{2}$$

with the GP prior F^l conditioned on the layer inputs X^{l-1} multiplied with the weights W^l,

$$F^l_{\star n} \,|\, X^{l-1} \sim \mathcal{N}(\mathbf{0}, K^l(X^{l-1} W^l_{\star n}, X^{l-1} W^l_{\star n}))\,, \tag{3}$$

and an additive Gaussian noise distribution,

$$X^l_{\star n} \,|\, F^l_{\star n} \sim \mathcal{N}(F^l_{\star n}, (\sigma^l_n)^2 \mathbb{1})\,. \tag{4}$$

This corresponds to a probabilistic activation function $F^l_{sn} = f^l_n(X^{l-1}_{s\star}\, W^l_{\star n})$ with

$$f^l_n(z) \sim \mathcal{GP}(0, k(z, z'))\,. \tag{5}$$

This GP has *scalar* inputs and uses the standard squared exponential covariance function. Analogous to standard neural networks, GPN layers can be stacked to form a multi-layer feed-forward network. The joint probability of such a stack is

$$\mathrm{P}(X^{\{l\}}, F^{\{l\}}) = \mathrm{P}(X^0) \prod_{l=1}^{L} \mathrm{P}(X^l_{\star n}, F^l_{\star n} \,|\, X^{l-1})\,. \tag{6}$$

[1] We omit writing $P(\bullet)$.

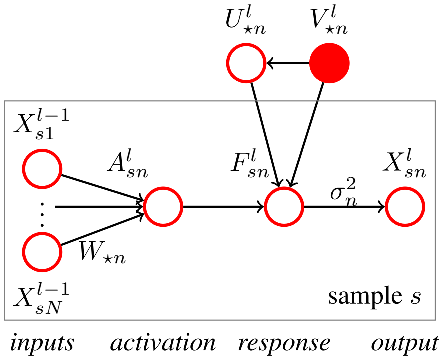

Figure 1: The auxiliary parametric representation of a GPN using virtual observation inducing points $V_\star$ and targets $U_\star$.

All input samples $X^0_{s\star}$, $s \in \{1, \dots, S\}$, are assumed to be normally distributed with known mean and covariance,

$$\mathrm{P}(X^0) = \prod_{s=1}^{S} \mathcal{N}(X^0_{s\star} \,|\, \mu^{\text{in}}_{s\star}, \Sigma^{\text{in}}_{s\star\star}) \,. \tag{7}$$

To obtain predictions $\mathrm{P}(X^L \,|\, X^0)$, all latent variables in eq. (6) would need to be marginalized out; unfortunately due to the occurrence of X^l in the covariance matrix in eq. (3) analytic integration is intractable.[2]

3.1 Auxiliary Parametric Representation

The path to obtain a tractable training objective is to temporarily parameterize the activation function $f^l_n(z)$ of each GPN using virtual observations (originally proposed by Quiñonero-Candela & Rasmussen (2005) for sparse approximations of GPs) of inputs and outputs of the function. These virtual observations are only introduced as an auxiliary device and will be marginalized out later. Each virtual observation r consists of a *scalar* inducing point V^l_{rn} and corresponding target U^l_{rn}. Under these assumptions on f^l_d, the GP prior $\mathrm{P}(F^l_{\star n} \,|\, X^{l-1})$ is replaced with

$$F^l_{\star d} \,|\, X^{l-1}, U^l_{\star n} \;\sim\; \mathcal{N}(\mu^{F^l}_{\star n}, \Sigma^{F^l}_{\star\star n}) \tag{8}$$

where the mean and variance are those obtained by using the virtual observations as "training" points for a GP regression evaluated at the "test" points $X^l W^l_{\star d}$,

$$\mu^{F^l}_{\star n} = K(X^{l-1} W^l_{\star n}, V^l_{\star n}) \, (\widetilde{K}^l_{n\star\star})^{-1} \, U^l_{\star n} \tag{9}$$

$$\Sigma^{F^l}_{\star\star d} = K(X^{l-1} W^l_{\star n}, X^{l-1} W^l_{\star n}) - K(X^{l-1} W^l_{\star n}, V^l_{\star n}) \, (\widetilde{K}^l_{n\star\star})^{-1} \, K(V^l_{\star n}, X^{l-1} W^l_{\star n}) \,. \tag{10}$$

where $\widetilde{K}^l_{nrt} = K(V^l_{rn}, V^l_{tn})$.

Given enough inducing points that lie densely between the layer's activations $A^l_{\star n} = X^{l-1} W^l_{\star n}$, the shape of the activation function becomes predominantly determined by the corresponding targets of these inducing points. Consequently, the inter-sample correlation in eq. (8) becomes negligible, allowing us to further approximate this conditional by factorizing it over the samples; thus we have $\mathrm{P}(F^l_{\star n} \,|\, X^l, U^l_{\star n}) = \prod_s \mathrm{P}(F^l_{sn} \,|\, X^{l-1}_{s\star}, U^l_{\star n})$ with

$$F^l_{sn} \,|\, X^{l-1}_{s\star}, U^l_{\star n} \;\sim\; \mathcal{N}(\mu^{F^l}_{sn}, \Sigma^{F^l}_{ssn}) \,. \tag{11}$$

We now marginalize eq. (4) over $F^l_{\star n}$ and get $\mathrm{P}(X^l_{\star n} \,|\, X^{l-1}, U^l_{\star n}) = \prod_s \mathrm{P}(X^l_{sn} \,|\, X^{l-1}_{s\star}, U^l_{\star n})$ with

$$X^l_{sn} \,|\, X^{l-1}_{s\star}, U^l_{\star n} \;\sim\; \mathcal{N}(\mu^{F^l}_{sn}, \Sigma^{F^l}_{ssn} + (\sigma^l_n)^2) \,. \tag{12}$$

[2] The inverse of the covariance matrix appears in the PDF of a normal distribution. Thus, the dependency on X^l is highly non-linear and analytic calculation of the integral is not feasible.

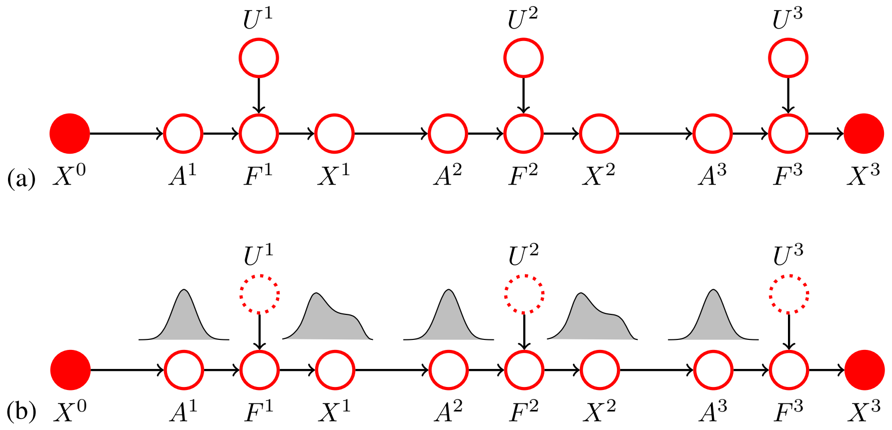

Figure 2: A GPN feed-forward network prior distribution (a) for three layers and the approximation of its posterior obtained using variational inference and the central limit theorem on the activations (b). Each node corresponds to all samples and GPN units within a layer. Dotted circles represent variational parameters.

Although we now have a distribution for X^l that is conditionally normal given the values of the previous layer, the marginals $\mathrm{P}(X^l_{\star n})$, $l \in \{1, \dots, L\}$, will, in general, not be normally distributed, because the input from the previous layer X^{l-1} appears non-linearly through the kernel function in the mean eq. (9).

By putting a GP prior on the distribution of the virtual observation targets U^l as shown in fig. 1.,

$$U^l_{\star n} \sim \mathcal{N}(\mathbf{0}, K(V^l_{\star n}, V^l_{\star n})) \,, \tag{13}$$

it can easily be verified that the marginal distribution of the response,

$$\mathrm{P}(F^l_{\star n} \,|\, X^l) = \int \mathrm{P}(F^l_{\star n} \,|\, X^l, U^l_{\star n}) \, \mathrm{P}(U^l_{\star n}) \, \mathrm{d}U^l_{\star n} = \mathcal{N}(F^l_{\star n} \,|\, 0, K(X^{l-1} W^l_{\star n}, X^{l-1} W^l_{\star n})) \,, \tag{14}$$

recovers the original, non-parametric GPN response distribution given by (3). The first use of this prior-restoring technique was presented in Titsias (2009) for finding inducing points of sparse GP regression using variational methods.

3.2 Approximate Bayesian Inference for GPN Networks

The joint distribution of a GPN feed-forward network is given by

$$\mathrm{P}(\{X\}_1^L, \{A\}_1^L, \{U\}_1^L, \{F\}_1^L \,|\, X^0) = \prod_{l=1}^{L} \mathrm{P}(A^l \,|\, X^{l-1}) \, \mathrm{P}(U^l) \, \mathrm{P}(F^l \,|\, A^l, U^l) \, \mathrm{P}(X^l \,|\, F^l) \,, \tag{15}$$

where to notation $\{\bullet\}_1^L$ should be read as $\{\bullet^1, \bullet^2, \dots, \bullet^L\}$. A graphical model corresponding to that distribution for three layers is shown in fig. 2a. Since exact marginalization over the latent variables is infeasible we apply the technique of variational inference Wainwright et al. (2008) to approximate the posterior of the model given some training data by a variational distribution Q.

The information about the activation functions learned from the training data is mediated via the virtual observation targets U^l, thus their variational posterior must be adaptable in order to store that information. Hence, we choose a normal distribution factorized over the GPN units within a layer with free mean and covariance for the approximative posterior of U^l,

$$\mathrm{Q}(U^l) = \prod_{n=1}^{N_l} \mathrm{Q}(U^l_{\star n}) \,, \qquad \mathrm{Q}(U^l_{\star n}) = \mathcal{N}(U^l_{\star n} \,|\, \widehat{\mu}^{U^l}_{\star n}, \widehat{\Sigma}^{U^l}_{\star\star n}) \,. \tag{16}$$

This allows the inducing targets of a GPN to be correlated, but the covariance matrix can be constrained to be diagonal, if it is desired to reduce the number of variational parameters. We keep the rest of the model distribution unchanged from the prior; thus the overall approximating posterior is

$$\mathrm{Q}(\{U\}_1^L, \{X\}_1^{L-1}, \{A\}_1^L, \{F\}_1^L) = \prod_{l=1}^{L} \mathrm{P}(A^l \,|\, X^{l-1})\, \mathrm{Q}(U^l)\, \mathrm{P}(F^l \,|\, A^l, U^l)\, \mathrm{P}(X^l \,|\, F^l)\,. \tag{17}$$

Estimating the variational parameters $\widehat{\mu}^{U^l}$ and $\widehat{\Sigma}^{U^l}$ requires maximizing the evidence lower bound (ELBO) given by

$$\begin{aligned} Ł = -\int \cdots \int \mathrm{Q}(\{U\}_1^L, \{X\}_1^{L-1}, \{A\}_1^L, \{F\}_1^L) \,\log \frac{\mathrm{Q}(\{U\}_1^L, \{X\}_1^{L-1}, \{A\}_1^L, \{F\}_1^L)}{\mathrm{P}(\{U\}_1^L, \{X\}_1^L, \{A\}_1^L, \{F\}_1^L \,|\, X^0)} \,\cdot \\ \mathrm{d}\{U\}_1^L \,\mathrm{d}\{X\}_1^L \,\mathrm{d}\{A\}_1^L \,\mathrm{d}\{F\}_1^L\,. \end{aligned} \tag{18}$$

Substituting the distributions into this equation results in $Ł = -Ł_{\mathrm{reg}} + Ł_{\mathrm{pred}}$ with

$$Ł_{\mathrm{reg}} = \sum_{l=1}^{L} \int \mathrm{Q}(U^l) \,\log \frac{\mathrm{Q}(U^l)}{\mathrm{P}(U^l)} \,\mathrm{d}U^l\,, \tag{19}$$

$$Ł_{\mathrm{pred}} = \int \mathrm{Q}(F^L) \,\log \mathrm{P}(X^L \,|\, F^L) \,\mathrm{d}F^L\,. \tag{20}$$

The term $Ł_{\mathrm{reg}}$ can be identified as the sum of the KL-divergences between the GP prior on the virtual observation targets and their approximative posterior $\mathrm{Q}(U^l)$. Since this term enters $Ł$ with a negative sign, its purpose is to keep the approximative posterior close to the prior; thus it can be understood as a regularization term. Evaluating using the formula for the KL-divergence between two normal distributions gives

$$\begin{aligned} Ł_{\mathrm{reg}} &= \sum_{l=1}^{L} \mathrm{KL}\big(\mathrm{Q}(U^l) \,||\, \mathrm{P}(U^l)\big) = \sum_{l=1}^{L} \sum_{n=1}^{N_l} \mathrm{KL}\big(\mathrm{Q}(U^l_{\star n}) \,||\, \mathrm{P}(U^l_{\star n})\big) \qquad (21) \\ &\propto \frac{1}{2} \sum_{l=1}^{L} \sum_{n=1}^{N_l} \left(\mathrm{tr}\Big(K(V^l_{\star n}, V^l_{\star n})^{-1}\, \widehat{\Sigma}^{U^l}_{\star\star n} \Big) + (\widehat{\mu}^{U^l}_{\star n})^T K(V^l_{\star n}, V^l_{\star n})^{-1} \widehat{\mu}^{U^l}_{\star n} + \log \frac{\big|K(V^l_{\star n}, V^l_{\star n})\big|}{\Big|\widehat{\Sigma}^{U^l}_{\star\star n}\Big|} \right)\,. \end{aligned}$$

The term $Ł_{\mathrm{pred}}$ cannot be evaluated yet because the exact marginal $\mathrm{Q}(F^L)$ is still intractable.

3.2.1 Central Limit Distribution of Activations

Due to the central limit theorem the activation A^l (weighted sum of inputs) of each GPN will converge to a normal distribution, *if* the number of incoming connections is sufficiently large (≥ 50) *and* the weights W^l have a sufficiently random distribution. For standard feed forward neural networks Wang & Manning (2013) experimentally showed that even *after* training the weights are sufficiently random for this assumption to hold. Hence we postulate that the same is true for GPNs and assume that the *marginal* distributions $\mathrm{Q}(A^l)$, $l \in \{1, \dots, L\}$, can be written as

$$\mathrm{Q}(A^l) = \prod_{s=1}^{S} \mathrm{Q}(A^l_{s\star})\,, \qquad \mathrm{Q}(A^l_{s\star}) \approx \mathcal{N}\left(A^l_{s\star} \,\middle|\, \mu^{A^l}_{s\star}, \Sigma^{A^l}_{s\star\star} \right)\,. \tag{22}$$

A graphical model corresponding to this approximate posterior is shown in fig. 2b. This allows the moments of $\mathrm{Q}(A^l)$ to be calculated exactly and propagated analytically from layer to layer. For this purpose we need to evaluate the conditional distributions

$$\mathrm{Q}(X^l \,|\, X^{l-1}) = \int \mathrm{Q}(F^l \,|\, A^l = W^l X^{l-1})\, \mathrm{P}(X^l \,|\, F^l) \,\mathrm{d}F^l\,, \tag{23}$$

$$\mathrm{Q}(F^l \,|\, A^l) = \prod_{n=1}^{N_l} \int \mathrm{Q}(U^l_{\star n})\, \mathrm{P}(F^l_{\star n} \,|\, A^l_{\star n}, U^l_{\star n}) \,\mathrm{d}U^l_{\star n}\,. \tag{24}$$

Since $\mathrm{Q}(F^l \,|\, A^l)$ is the conditional of a GP with normally distributed observations, the joint distribution $\mathrm{Q}(F^l_{\star n}, U^l_{\star n} \,|\, A^l_{\star n}) = \mathrm{Q}(U^l_{\star n})\, \mathrm{P}(F^l_{\star n} \,|\, A^l_{\star n}, U^l_{\star n})$ must itself be normal,

$$\mathrm{Q}(F^l_{\star n}, U^l_{\star n} \,|\, A^l_{\star n}) = \mathcal{N}\left(\begin{bmatrix} F^l_{\star n} \\ U^l_{\star n} \end{bmatrix} \,\middle|\, \begin{bmatrix} \widehat{\mu}^{F^l}_{\star n} \\ \widehat{\mu}^{U^l}_{\star n} \end{bmatrix}, \begin{bmatrix} \widehat{\Sigma}^{F^l}_{\star\star n} & \widetilde{\Sigma}_{FU} \\ \widetilde{\Sigma}_{UF} & \widehat{\Sigma}^{U^l}_{\star\star n} \end{bmatrix} \right)\,, \tag{25}$$

and we can find the values for the unknown parameters $\widehat{\mu}^{F^l}_{\star n}$, $\widehat{\Sigma}^{F^l}_{\star\star n}$ (and $\widetilde{\Sigma}_{FU} = \widetilde{\Sigma}^T_{UF}$) by equating the moments of its conditional distribution $\mathrm{Q}(F^l_{\star n} \,|\, U^l_{\star n}, A^l_{\star n})$ with $\mathrm{P}(F^l_{\star n} \,|\, U^l_{\star n}, A^l_{\star n})$. Thus by solving the resulting equations we obtain for eq. (24),

$$\mathrm{Q}(F^l \,|\, A^l) = \prod_{n=1}^{N^l} \mathcal{N}(F^l_{\star n} \,|\, \widehat{\mu}^{F^l}_{\star n}, \widehat{\Sigma}^{F^l}_{\star\star n})\,. \tag{26}$$

where

$$\widehat{\mu}^{F^l}_{\star n} = K(A^l_{\star n}, V^l_{\star n})\, K(V^l_{\star n}, V^l_{\star n})^{-1}\, \widehat{\mu}^{U^l}_{\star n} \tag{27}$$

$$\widehat{\Sigma}^{F^l}_{\star\star n} = K(A^l_{\star n}, A^l_{\star n}) - K(A^l_{\star n}, V^l_{\star n})\, \widehat{K}^{U^l}_{\star\star n}\, K(V^l_{\star n}, A^l_{\star n}) \tag{28}$$

with

$$\widehat{K}^{U^l}_{\star\star n} \triangleq K(V^l_{\star n}, V^l_{\star n})^{-1} - K(V^l_{\star n}, V^l_{\star n})^{-1}\, \widehat{\Sigma}^{U^l}_{\star\star n}\, K(V^l_{\star n}, V^l_{\star n})^{-1}\,. \tag{29}$$

For deterministic observations, that is $\widehat{\Sigma}^{U^l}_{\star\star n} = 0$, we obtain $\widehat{K}^{U^l}_{\star\star n} = K(V^l_{\star n}, V^l_{\star n})^{-1}$ and thus recover the standard GP regression distribution as expected. If U^l follows its prior, that is $\widehat{\mu}^{U^l}_{\star n} = \mathbf{0}$ and $\widehat{\Sigma}^{U^l}_{\star\star n} = K(V^l_{\star n}, V^l_{\star n})$, we obtain $\widehat{K}^{U^l}_{\star\star n} = 0$ and thus recover the GP prior on F^l. In that case the virtual observations behave as if they were not present.

Having $\mathrm{Q}(F^l \,|\, A^l)$ immediately allows us to evaluate eq. (23) since $\mathrm{P}(X^l \,|\, F^l)$ just provides additive Gaussian noise; thus we obtain

$$\mathrm{Q}(X^l \,|\, X^{l-1}) = \prod_{n=1}^{N^l} \mathcal{N}(F^l_{\star n} \,|\, \widehat{\mu}^{F^l}_{\star n}, \widehat{\Sigma}^{F^l}_{\star\star n} + (\sigma^l_n)^2 \mathbb{1})\,. \tag{30}$$

Returning to Ł_{pred} from (20) and writing it as

$$\text{Ł}_{\mathrm{pred}} = \iiint \mathrm{Q}(A^L)\, \mathrm{Q}(U^L)\, \mathrm{P}(F^L \,|\, A^L, U^L)\, \log \mathrm{P}(X^L \,|\, F^L)\, \mathrm{d}A^L\, \mathrm{d}U^L\, \mathrm{d}F^L\,. \tag{31}$$

shows that we first need to obtain the distribution $\mathrm{Q}(A^L)$. This is done by iteratively calculating the marginals $\mathrm{Q}(A^l)$ for $l \in \{1, \dots, L\}$.

3.2.2 Propagation of Uncertainty

For $l \geq 1$ the marginal distribution of the activations is

$$\mathrm{Q}(A^{l+1}) = \iiint \mathrm{Q}(A^l)\, \mathrm{Q}(F^l \,|\, A^l)\, \mathrm{P}(X^l \,|\, F^l)\, \mathrm{P}(A^{l+1} \,|\, X^l)\, \mathrm{d}A^l\, \mathrm{d}F^l\, \mathrm{d}X^l \tag{32}$$

where $\mathrm{Q}(F^l \,|\, A^l)$ is given by (26). We first evaluate the mean and covariance of the marginal $\mathrm{Q}(F^l) = \int \mathrm{Q}(A^l)\, \mathrm{Q}(F^l \,|\, A^l)\, \mathrm{d}A^l$. For the marginal mean of the response $\widetilde{\mu}^{F^l}_{sn} \triangleq \mathrm{E}_{\mathrm{Q}(F^l)}\left[F^l_{sn}\right] = \mathrm{E}_{\mathrm{Q}(A^l_{s\star})}\left[\widehat{\mu}^{F^l}_{sn}\right]$ we obtain

$$\widetilde{\mu}^{F^l}_{sn} = \mathrm{E}_{\mathrm{Q}(A^l_{s\star})}\left[K(A^l_{sn}, V^l_{\star n})\right]^T K(V^l_{\star n}, V^l_{\star n})^{-1}\, \widehat{\mu}^{U^l}_{\star n} = (\psi^l_{s\star n})^T\, K(V^l_{\star n}, V^l_{\star n})^{-1}\, \widehat{\mu}^{U^l}_{\star n} \tag{33}$$

with

$$\psi^l_{stn} \triangleq \mathrm{E}_{\mathrm{Q}(A^l_{sn})}[K(A^l_{sn}, V^l_{tn})] = \sqrt{\frac{\nu^2}{\nu^2 + \Sigma^{A^l}_{snn}}}\, \exp\left(-\frac{(\mu^{A^l}_{sn} - V^l_{tn})^2}{2(\nu^2 + \Sigma^{A^l}_{snn})}\right)\,,$$

which was calculated by expressing the squared exponential kernel as a normal PDF and applying the product formula for Gaussian PDFs (Bromiley, 2003). For the marginal covariances of the response $\widetilde{\Sigma}^{F^l}_{s\star\star}$ we obtain by applying the law of total expectation

$$\widetilde{\Sigma}^{F^l}_{snn'} \triangleq \mathrm{Cov}(F^l_{sn}, F^l_{sn'}) = \mathrm{E}_{\mathrm{Q}(A^l_{s\star})}\left[\mathrm{E}_{\mathrm{Q}(F^l_{sn}, F^l_{sn'} \,|\, A^l)}[F^l_{sn}\, F^l_{sn'}]\right] - \widetilde{\mu}^{F^l}_{sn}\, \widetilde{\mu}^{F^l}_{sn'} \tag{34}$$

For the elements representing the variance, i.e. the diagonal $n = n'$, this becomes

$$\begin{aligned}\widetilde{\Sigma}^{F^l}_{snn} &= \mathrm{E}_{\mathrm{Q}(A^l_{s\star})}\left[\widehat{\Sigma}^{F^l}_{ssn} + (\widehat{\mu}^{F^l}_{sn})^2\right] - (\widetilde{\mu}^{F^l}_{sn})^2 \\ &= 1 - \mathrm{Tr}\Big([\widehat{K}^{U^l}_{\star\star n} - \beta^l_{\star n}(\beta^l_{\star n})^T]\, \Omega^l_{s\star\star n}\Big) - \mathrm{Tr}\big(\psi^l_{s\star n}(\psi^l_{s\star n})^T\, \beta^l_{\star n}(\beta^l_{\star n})^T\big)\end{aligned} \tag{35}$$

with $\beta^l_{\star n} \triangleq K(V^l_{\star n}, V^l_{\star n})^{-1}\, \widehat{\mu}^{U^l}_{\star n}$ and, using the same method as above,

$$\begin{aligned}\Omega^l_{strn} &\triangleq \mathrm{E}_{\mathrm{Q}(A^l_{s\star})}[K(A^l_{sn}, V^l_{tn})\, K(A^l_{sn}, V^l_{rn})] \\ &= \sqrt{\frac{\nu^2}{\nu^2 + 2\Sigma^{A^l}_{snn}}} \exp\left(-\frac{\left(\mu^{A^l}_{sn} - \frac{V^l_{tn}+V^l_{rn}}{2}\right)^2}{\nu^2 + 2\Sigma^{A^l}_{snn}} - \frac{(V^l_{tn} - V^l_{rn})^2}{4\nu^2} \right) . \end{aligned} \tag{36}$$

For off-diagonal elements, $n \neq n'$, we observe that F^l_{sn} and $F^l_{sn'}$ are *conditionally independent* given A^l because the activation functions of GPNs n and n' are represented by two different GPs. Hence we have

$$\widetilde{\Sigma}^{F^l}_{snn'} = \mathrm{E}_{\mathrm{Q}(A^l_{s\star})}\left[\widehat{\mu}^{F^l}_{sn}\, \widehat{\mu}^{F^l}_{sn'}\right] - \widetilde{\mu}^{F^l}_{sn}\, \widetilde{\mu}^{F^l}_{sn'} = (\beta^l_{\star n})^T\, \Lambda^l_{s\star\star nn'}\, \beta^l_{\star n'} - \widetilde{\mu}^{F^l}_{sn}\, \widetilde{\mu}^{F^l}_{sn'} \,. \tag{37}$$

where

$$\Lambda^l_{srtnn'} \triangleq \mathrm{E}_{\mathrm{Q}(A^l_{s\star})}\left[K(A^l_{sn}, V^l_{tn})\, K(A^l_{sn'}, V^l_{rn'})\right] = \frac{\nu^2\, \exp(\mathscr{A}^l_{srtnn'} / \mathscr{B}^l_{snn'})}{\sqrt{[\nu^2 + \Sigma^{A^l}_{snn}]\, [\nu^2 + \Sigma^{A^l}_{sn'n'}] - (\Sigma^{A^l}_{nn'})^2}}$$

with

$$\begin{aligned}\mathscr{A}^l_{srtnn'} &\triangleq (V^l_{rn'} - \mu^{A^l}_{sn'})^2\, [\nu^2 + \Sigma^{A^l}_{snn}] + (V^l_{tn} - \mu^{A^l}_{sn})^2\, [\nu^2 + \Sigma^{A^l}_{sn'n'}] + \\ &\quad 2\, (V^l_{tn} - \mu^{A^l}_{sn})\, (\mu^{A^l}_{sn'} - V^l_{rn'})\, \Sigma^{A^l}_{snn'} \\ \mathscr{B}^l_{snn'} &\triangleq 2\left([\nu^2 + \Sigma^{A^l}_{snn}]\, [\nu^2 + \Sigma^{A^l}_{sn'n'}] - (\Sigma^{A^l}_{sn'n})^2 \right) .\end{aligned}$$

This concludes the calculation of the moments of F^l. We can now state how the activation distribution propagates from a layer to the next. The marginal distribution of A^{l+1} is given by

$$\mathrm{Q}(A^{l+1}) = \mathcal{N}(A^{l+1}_{s\star} \mid \widetilde{\mu}^{A^{l+1}}_{s\star}, \widetilde{\Sigma}^{A^{l+1}}_{s\star\star}) \tag{38}$$

with

$$\widetilde{\mu}^{A^{l+1}}_{s\star} = W^{l+1}\, \widetilde{\mu}^{F^l}_{s\star} \tag{39}$$

$$\widetilde{\Sigma}^{A^{l+1}}_{s\star\star} = W^{l+1}\, \left(\widetilde{\Sigma}^{F^l}_{s\star\star} + \operatorname{diag}(\sigma^l)^2 \right)\, (W^{l+1})^T \,. \tag{40}$$

Thus $\mathrm{Q}(A^L)$ in (31) can be calculated by iterating the application of eqs. (33), (35), (37), (39) and (40) over the layers l. To save computational power only the variances can be propagated by assuming that $\widetilde{\Sigma}^{F^l}_{s\star\star}$ is diagonal and therefore ignoring (37).

Now Ł_{pred} can be identified as the expected log-probability of the observations under the marginal distribution $Q(F^L)$ and thus we can expand it as follows,

$$\begin{aligned}\text{Ł}_{\text{pred}} &= \mathrm{E}_{\mathrm{Q}(F^L)}\left[\log \mathrm{P}(X^L \mid F^L)\right] \propto \mathrm{E}_{\mathrm{Q}(F^L)}\left[-S \sum_{n=1}^{N_l} \log \sigma^L_n - \frac{1}{2} \sum_{s=1}^{S} \sum_{n=1}^{N_L} \frac{\left(X^L_{sn} - F^L_{sn}\right)^2}{(\sigma^L_n)^2} \right] \\ &= -S \sum_{n=1}^{N_L} \log \sigma^L_n - \frac{1}{2} \sum_{s=1}^{S} \sum_{n=1}^{N_L} \frac{(X^L_{sn})^2 - 2\, X^L_{sn}\, \mathrm{E}_{\mathrm{Q}(F^L)}\left[F^L_{sn}\right] + \mathrm{E}_{\mathrm{Q}(F^L)}\left[(F^L_{sn})^2\right]}{(\sigma^L_n)^2} \end{aligned} \tag{41}$$

where S is the number of training samples and X^L are the training targets. The distribution $Q(F^L)$ is of arbitrary form, but only its first and second moments are required to evaluate Ł_{pred}. For the first moment we obtain $\mathrm{E}_{\mathrm{Q}(F^L)}\left[F^l_{sn}\right] = \widetilde{\mu}^{F^L}_{sn}$ with $\widetilde{\mu}^{F^L}_{sn}$ given by (33) and the second moment evaluates to

$$\begin{aligned}\mathrm{E}_{\mathrm{Q}(F^L)}\left[(F^L_{sn})^2\right] &= \mathrm{Var}_{\mathrm{Q}(F^L)}\left(F^L_{sn}\right) + \mathrm{E}_{\mathrm{Q}(F^L)}\left[F^L_{sn}\right]^2 = \widetilde{\Sigma}^{F^L}_{snn} + \left(\widetilde{\mu}^{F^L}_{snn}\right)^2 \\ &= 1 - \operatorname{tr}\left[\left(\widehat{K}^{U^L}_{\star\star n} - \beta^L_{\star n} (\beta^L_{\star n})^T\right) \Omega^L_{s\star\star n}\right] , \end{aligned} \tag{42}$$

with $\beta^L_{\star n} \triangleq K(V^L_{\star n}, V^L_{\star n})^{-1}\, \widehat{\mu}^{U^L}_{\star n}$ and Ω^L from (36).

This concludes the calculation of all terms of the variational lower bound (18). The resulting objective is a fully deterministic function of the parameters. Training of the model is performed by maximizing Ł w.r.t. to the variational parameters $\widehat{\mu}^{U^l}$, $\widehat{\Sigma}^{U^l}$ and the model parameters σ^l, W^l and V^l. This can be performed using any gradient-descent based algorithm. The necessary derivatives are not derived here and it is assumed that this can be performed automatically using symbolic or automatic differentiation in an appropriate framework.

3.3 Computational and Model Complexity

The activation function are represented using $2R$ variational parameters per GPN, where R is the number of inducing points and targets. It can be shown that $R = 10$ linearly spaced inducing points are enough to represent the most commonly used activation functions (sigmoid, tanh, soft ReLU) with very high accuracy. The number of required parameters can be reduced by sharing the same activation function within groups of neurons or even across whole layers of neurons. If the inducing points V^l are fixed (for example by equally distributing them in the interval $[-1, 1]$), the kernel matrices $K(V^l, V^l)$ and their inverses can be precomputed since they are constant. The number of parameters and the computational complexity of propagating the means and covariances only depend on R and are therefore *independent* of the number of training samples. Thus, like a conventional neural network, a GPN network can inherently be trained on datasets of unlimited size.

4 Conclusion

We have presented a non-parametric model based on GPs for learning of activation functions in a multi-layer neural network. We then successively applied variational to make fully Bayesian inference feasible and efficient while keeping its probabilistic nature and providing not only best guess predictions but also confidence estimations in our predictions. Although we employ GPs, our parametric approximation allows our model to scale to datasets of unlimited size like conventional neural networks do.

We have validated networks of Gaussian Process Neurons in a set of experiments, the details of which we submit in a subsequent publication. In those experiments, our model shows to be significantly less prone to overfitting than a traditional feed-forward network of same size, despite having more parameters.

References

Forest Agostinelli, Matthew Hoffman, Peter Sadowski, and Pierre Baldi. Learning activation functions to improve deep neural networks. *arXiv preprint arXiv:1412.6830*, 2014a.

Forest Agostinelli, Matthew D. Hoffman, Peter J. Sadowski, and Pierre Baldi. Learning activation functions to improve deep neural networks. *CoRR*, abs/1412.6830, 2014b.

Paul Bromiley. Products and convolutions of Gaussian probability density functions. *Tina-Vision Memo*, 3, 2003.

Djork-Arné Clevert, Thomas Unterthiner, and Sepp Hochreiter. Fast and accurate deep network learning by exponential linear units (elus). *arXiv preprint arXiv:1511.07289*, 2015.

Carson Eisenach, Zhaoran Wang, and Han Liu. Nonparametrically learning activation functions in deep neural nets. 2016.

Xavier Glorot, Antoine Bordes, and Yoshua Bengio. Deep sparse rectifier neural networks. In *Proceedings of the 14th International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2011.

I. J. Goodfellow, D. Warde-Farley, M. Mirza, A. Courville, and Y. Bengio. Maxout Networks. *ArXiv e-prints*, February 2013.

Ian J Goodfellow, David Warde-Farley, Mehdi Mirza, Aaron Courville, and Yoshua Bengio. Maxout networks. *arXiv preprint arXiv:1302.4389*, 2013.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *Proceedings of the IEEE international conference on computer vision*, pp. 1026–1034, 2015.

Günter Klambauer, Thomas Unterthiner, Andreas Mayr, and Sepp Hochreiter. Self-normalizing neural networks. *arXiv preprint arXiv:1706.02515*, 2017.

Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger (eds.), *Advances in Neural Information Processing Systems 25*, pp. 1097–1105. Curran Associates, Inc., 2012.

Andrew L Maas, Awni Y Hannun, and Andrew Y Ng. Rectifier nonlinearities improve neural network acoustic models. In *Proc. ICML*, volume 30, 2013.

Vinod Nair and Geoffrey E. Hinton. Rectified linear units improve restricted boltzmann machines. In *Proceedings of the 27th International Conference on International Conference on Machine Learning*, ICML'10, pp. 807–814, USA, 2010. Omnipress. ISBN 978-1-60558-907-7.

Joaquin Quiñonero-Candela and Carl Edward Rasmussen. A unifying view of sparse approximate Gaussian process regression. *Journal of Machine Learning Research*, 6(Dec):1939–1959, 2005.

Carl Edward Rasmussen and Christopher K. I. Williams. *Gaussian Processes for Machine Learning*. The MIT Press, 2006.

Michalis K Titsias. Variational learning of inducing variables in sparse gaussian processes. In *International Conference on Artificial Intelligence and Statistics*, pp. 567–574, 2009.

Martin J Wainwright, Michael I Jordan, et al. Graphical models, exponential families, and variational inference. *Foundations and Trends® in Machine Learning*, 1(1–2):1–305, 2008.

Sida I Wang and Christopher D Manning. Fast dropout training. In *ICML (2)*, pp. 118–126, 2013.

Bing Xu, Naiyan Wang, Tianqi Chen, and Mu Li. Empirical evaluation of rectified activations in convolutional network. *arXiv preprint arXiv:1505.00853*, 2015.