Identifiable Mixtures of Sparse Variational Gaussian Process Experts

Aidan Scannell
University of Bristol
aidan.scannell@bristol.ac.uk

Carl Henrik Ek
University of Cambridge
che29@cam.ac.uk

Arthur Richards
University of Bristol
arthur.richards@bristol.ac.uk

***Abstract*—Mixture models are inherently unidentifiable as different combinations of component distributions and mixture weights can generate the same distributions over the observations. We propose a scalable Mixture of Experts model where both the experts and gating functions are modelled using Gaussian processes. Importantly, this balanced treatment of the experts and the gating network introduces an interplay between the different parts of the model. This can be used to constrain the set of admissible functions reducing the identifiability issues normally associated with mixture models. The model resembles the original Mixture of Gaussian Process Experts method with a GP-based gating network. However, we derive a variational inference scheme that allows for stochastic updates enabling the model to be used in a more scalable fashion.**

I. Introduction

Given an input $\mathbf{x}_n$ and an output y_n, mixture models model a mixture of distributions over the output $p(y_n|\mathbf{x}_n) = \sum_{k=1}^{K} \Pr(\alpha_n = k) p(y_n|\alpha_n = k, \mathbf{x}_n)$. The predictive distribution $p(y_n|\mathbf{x}_n)$ consists of K mixture components $p(y_n|\alpha_n = k, \mathbf{x}_n)$ that are weighted according to the mixing probabilities $\Pr(\alpha_n = k)$. Mixture models are inherently unidentifiable as different combinations of mixture components and mixing probabilities can generate the same distributions over the output. The mixture of experts (ME) model is an extension where the mixing probabilities depend on the input variable $\Pr(\alpha_n = k \mid \mathbf{x}_n)$ [3]. These are referred to as gating functions and collectively the gating network. The individual component densities $p(y_n \mid \alpha_n = k, \mathbf{x}_n)$ are then referred to as experts, as at different regions in the input space, different components are responsible for predicting. Given a set of observations $\mathcal{D} = \{(\mathbf{x}_n, y_n)\}_{n=1}^{N}$ with inputs $\mathbf{X} \in \mathbb{R}^{N \times D}$ and outputs $\mathbf{y} \in \mathbb{R}^{N}$, the ME marginal likelihood is given by,

$$p(\mathbf{y} \mid \mathbf{X}) = \prod_{n=1}^{N} \sum_{k=1}^{K} \underbrace{\Pr(\alpha_n = k \mid \mathbf{x}_n)}_{\text{Mixing Probability}} \underbrace{p(y_n \mid \alpha_n = k, \mathbf{x}_n)}_{\text{Expert}} \quad (1)$$

where $\alpha_n \in \{1, \dots, K\}$ is the expert indicator variable assigning the n^{th} observation to an expert.

Modelling the experts as Gaussian processes (GPs) gives rise to a class of powerful models known as Mixtures of Gaussian Process Experts (MoGPE). Under the standard Gaussian likelihood model, each expert is given by,

$$y_n = f_k(\mathbf{x}_n) + \epsilon_k, \quad \epsilon_k \sim \mathcal{N}(0, \sigma_k^2) \quad (2)$$

$$p(y_n \mid \alpha_n = k, f_k(\mathbf{x}_n)) = \mathcal{N}\left(y_n \mid f_k(\mathbf{x}_n), \sigma_k^2\right) \quad (3)$$

where f_k and σ_k^2 represent the latent function and the noise variance associated with the k^{th} expert. Placing independent GP priors on each of the expert's latent functions,

$$p\left(f_k(\mathbf{X})\right) = \mathcal{N}\left(f_k(\mathbf{X}) \mid \mu_k(\mathbf{X}), k_k(\mathbf{X}, \mathbf{X})\right) \quad (4)$$

where $\mu_k(\cdot)$ and $k_k(\cdot, \cdot)$ represent the mean and covariance functions associated with the k^{th} expert respectively, leads to each expert resembling a standard GP regression model. Note that the dependence on the inputs $\mathbf{X}$ and hyperparameters θ_k has been dropped for notational conciseness.

The gating network can be seen as a handle for encoding prior knowledge that can be used to constrain the set of admissible functions. This can improve identifiability and lead to learned representations that better reflect our understanding of the system. The simplest case being reordering the experts.

In the remainder of this paper we will formulate the MoGPE model with a GP-based gating network. After observing its poor computational complexity, we will augment the probability space with a set of psuedo-training observations (a.k.a inducing points) and use them to derive a variational lower bound, that can be optimised with stochastic gradient methods.

II. Gaussian Process Gating Network

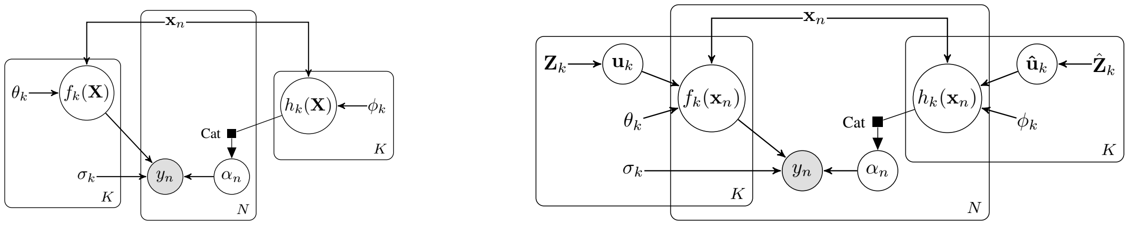
Motivated by identifiablility, this work adopts a GP-based gating network as seen in the original MoGPE [6] model. The gating network resembles a Gaussian process classification model, i.e. it places independent GP priors on each of the gating functions and normalises their output to obtain a Categorical distribution over the mode indicator variable. The probabilities of this Categorical distribution $\Pr(\alpha = k \mid \mathbf{h}(\cdot))$ are obtained by evaluating K latent gating functions $\mathbf{h}(\cdot) = \{h_k(\cdot)\}_{k=1}^{K}$ and normalising their output. In the general case, the gating network uses the softmax function,

$$\Pr\left(\alpha = k \mid \mathbf{h}(\cdot)\right) = \text{softmax}_k\left(\mathbf{h}(\cdot)\right) = \frac{\exp\left(h_k(\cdot)\right)}{\sum_{k=1}^{K} \exp\left(h_k(\cdot)\right)}.$$

Each gating function $h_k(\cdot)$ describes how its corresponding expert's mixing probability varies over the input space. We then place independent GP priors on each gating function, giving the distribution over all gating functions as,

$$p\left(\mathbf{h}(\mathbf{X})\right) = \prod_{k=1}^{K} p(h_k(\mathbf{X})) = \prod_{k=1}^{K} \mathcal{N}\left(h_k(\mathbf{X}) \mid \hat{\mu}_k(\mathbf{X}), \hat{k}_k(\mathbf{X}, \mathbf{X})\right)$$

Aidan Scannell is a PhD student at the EPSRC Centre for Doctoral Training in Future Autonomous and Robotic Systems (FARSCOPE CDT).



(a) Joint probability model

(b) Augmented probability model

Fig. 1. Graphical models showing a) the joint probability model and b) the approximate joint probability model after augmenting the probability space with pseudo observations. The K latent gating functions h_k are evaluated and normalised to obtain the mixing probabilities $\Pr(\alpha_n = k \mid \mathbf{x}_n)$. The expert indicator variable α_n is then sampled from a Categorical distribution governed by these probabilities. The indicated expert's latent function f_k and Gaussian likelihood $\sigma^{(k)}$ are then evaluated to generate the output y_n.

where $\hat{\mu}_k(\cdot)$ and $\hat{k}_k(\cdot,\cdot)$ are the mean and covariance functions associated with the k^{th} gating function. With this formulation, the marginal likelihood can be rewritten as,

$$p(\mathbf{y} \mid \mathbf{X}) = \sum_{k=1}^{K} \Bigg(\underbrace{\mathbb{E}_{p(\mathbf{h}(\mathbf{X}))}\left[\Pr\left(\{\alpha_n = k\}_{k=1}^{K} \mid \mathbf{h}(\mathbf{X})\right)\right]}_{\text{Mixing Probabilities}} \underbrace{\mathbb{E}_{p(f_k(\mathbf{X}))}\left[p\left(\mathbf{y} \mid f_k\left(\mathbf{X}\right)\right)\right]}_{\text{GP Expert}} \Bigg) \tag{5}$$

which is the original MoGPE model proposed by [6]. Observe that the GP priors have removed the factorisation over data which is present in the ME marginal likelihood (Eq. 1). Fig. 1a shows the associated graphical model.

III. Inference

The marginal likelihood in Eq. 5 is extremely expensive to evaluate $\mathcal{O}(KN^4)$ and is also intractable due to the marginalisation of $\mathbf{h}$ in the gating network. This work focuses on inducing point methods [4], where the latent variables are augmented with inducing input-output pairs known as inducing "inputs" $\mathbf{Z}$, and inducing "variables" $\mathbf{u} = f(\mathbf{Z})$. Following the approach by [5], we introduce a set of M inducing points for each of the experts $p(\mathbf{u}_k \mid \mathbf{Z}_k) = \mathcal{N}(\mathbf{u}_k \mid \mu_k(\mathbf{Z}_k), k_k(\mathbf{Z}_k, \mathbf{Z}_k))$ and each of the gating functions $p(\hat{\mathbf{u}}_k \mid \hat{\mathbf{Z}}_k) = \mathcal{N}(\hat{\mathbf{u}}_k \mid \hat{\mu}_k(\hat{\mathbf{Z}}_k), \hat{k}_k(\hat{\mathbf{Z}}_k, \hat{\mathbf{Z}}_k))$. However, instead of collapsing these inducing variables, we follow [1, 2] and explicitly represent them as variational distributions,

$$q(\mathbf{f}_n) := \prod_{k=1}^{K} q\left(f_k(\mathbf{x}_n)\right) = \prod_{k=1}^{K} \int p\left(f_k(\mathbf{x}_n) \mid \mathbf{u}_k\right) q\left(\mathbf{u}_k\right) \mathrm{d}\mathbf{u}_k$$
$$q(\mathbf{h}_n) := \prod_{k=1}^{K} q\left(h_k(\mathbf{x}_n)\right) = \prod_{k=1}^{K} \int p\left(h_k(\mathbf{x}_n) \mid \hat{\mathbf{u}}_k\right) q\left(\hat{\mathbf{u}}_k\right) \mathrm{d}\hat{\mathbf{u}}_k,$$

and use them to lower bound the marginal likelihood. Note that $q(\mathbf{u}_k)$ and $q(\hat{\mathbf{u}}_k)$ are given by $q(\mathbf{u}_k) = \mathcal{N}(\mathbf{u}_k \mid \mathbf{m}_k, \mathbf{S}_k)$ and $q(\hat{\mathbf{u}}_k) = \mathcal{N}\left(\hat{\mathbf{u}}_k \mid \hat{\mathbf{m}}_k, \hat{\mathbf{S}}_k\right)$, and $\{\mathbf{m}_k, \hat{\mathbf{m}}_k, \mathbf{S}_k \hat{\mathbf{S}}_k\}_{k=1}^{K}$ are treated as variational parameters. Given our variational posterior, we lower bound the marginal likelihood,

$$\mathcal{L} = \sum_{n=1}^{N} \mathbb{E}_{q(\mathbf{f}_n, \mathbf{h}_n)} \left[\log \sum_{k=1}^{K} \Pr(\alpha_n = k \mid \mathbf{h}_n) p(y_n \mid \alpha_n = k, f_k(\mathbf{x}_n)) \right] - \sum_{k=1}^{K} \mathrm{KL}\left[q(\mathbf{u}_k) \mid\mid p(\mathbf{u}_k | \mathbf{Z}_k)\right] - \sum_{k=1}^{K} \mathrm{KL}\left[q(\hat{\mathbf{u}}_k) \mid\mid p(\hat{\mathbf{u}}_k | \hat{\mathbf{Z}}_k)\right] \tag{6}$$

This bound meets the necessary conditions to perform stochastic gradient methods on $\{q(\mathbf{u}_k), q(\hat{\mathbf{u}}_k)\}_{k=1}^{K}$ as the sum of N terms corresponds to input-output pairs. The inducing inputs $\{\mathbf{Z}_k, \hat{\mathbf{Z}}_k\}_{k=1}^{K}$, kernel hyperparameters $\{\theta_k, \phi_k\}_{k=1}^{K}$ and noise variances $\{\sigma_k\}_{k=1}^{K}$ are treated as variational hyperparameters and are optimised using stochastic gradient descent alongside the variational parameters. The expectation in Eq. 6 is intractable so we approximate it by drawing single samples from $q(\mathbf{f}_n)$ and $q(\mathbf{h}_n)$.

Note that our augmented model captures the dependencies in the joint distribution of the data through the inducing variables, but as $M \ll N$ these have a much lower computational burden. Given a batch of N_b observations our bound has complexity $\mathcal{O}\left(N_b K M^3\right)$.

IV. Conclusion

This paper presents a novel variational inference scheme that improves the scalability of the Mixture of Gaussian Process Experts model with a GP-based gating network. The GP-based gating network can be used to constrain the set of admissible functions through the placement of informative GP priors on the gating functions. This aids the inherent identifiability issues associated with mixture models. Our variational lower bound principally handles uncertainty and provides scalability as it can be optimised with stochastic gradient methods. The proposed lower bound provides a coupling between the optimisation of the experts and the gating network by efficiently marginalising the expert indicator variable.

References

[1] J. Hensman et al. "Gaussian Processes for Big Data". In: *Proceedings of the 29th Conference on Uncertainty in Artificial Intelligence*. Uncertainty in Artificial Intelligence. Vol. 29. 2013, pp. 282–290.

[2] J. Hensman et al. "Scalable Variational Gaussian Process Classification". In: *Artificial Intelligence and Statistics*. Artificial Intelligence and Statistics. PMLR, Feb. 21, 2015, pp. 351–360.

[3] R. A. Jacobs et al. "Adaptive Mixtures of Local Experts". In: *Neural Computation* 3.1 (Mar. 1, 1991), pp. 79–87.

[4] E. Snelson and Z. Ghahramani. "Sparse Gaussian Processes Using Pseudo-Inputs". In: *Proceedings of the 18th International Conference on Neural Information Processing Systems*. 2005.

[5] M. Titsias. "Variational Learning of Inducing Variables in Sparse Gaussian Processes". In: *Artificial Intelligence and Statistics*. Artificial Intelligence and Statistics. PMLR, Apr. 15, 2009, pp. 567–574.

[6] V. Tresp. "Mixtures of Gaussian Processes". In: *Advances in Neural Information Processing Systems*. Vol. 13. 2000, pp. 654–660.