
Overlapping Mixtures of Gaussian Processes for the Data Association Problem

Miguel Lázaro Gredilla

Dept. of Signal Theory and Communications
Carlos III University of Madrid
Leganés, Spain
miguel@tsc.uc3m.es

Steven Van Vaerenbergh

Dept. of Communications Engineering
University of Cantabria
Santander, Spain
steven@gtas.dicom.unican.es

Neil Lawrence

Machine Learning and Optimization Research Group
University of Manchester
Manchester, UK
neill@cs.man.ac.uk

Abstract

In this work we introduce a mixture of GPs to address the data association problem, i.e. to label a group of observations according to the sources that generated them. Unlike several previously proposed GP mixtures, the novel mixture has the distinct characteristic of using no gating function to determine the association of samples and mixture components. Instead, all the GPs in the mixture are global and samples are clustered following “trajectories” across input space. We use a variational Bayesian algorithm to efficiently recover sample labels and use a KL-corrected bound to learn the hyperparameters. We show how multiobject tracking problems may be disambiguated and explore the characteristics of the model in more traditional regression settings.

1 Introduction

The data association problem arises in multi-target tracking scenarios. Given a set of observations that represent the positions of a number of moving sources, such as cars or airplanes, data association consists of inferring which observations originate from the same source [1, 2]. Typical multi-target tracking algorithms include joint Kalman filters [3] and joint particle filters [4], which operate online and usually make instant data association decisions based on nearest-neighbour criteria. Data association results can be significantly improved by postponing these decisions until enough information is available to exclude ambiguities [2], but the number of possible trajectories grows exponentially. We present an algorithm that is able to consider all available data points in batch form whilst avoiding the exponential growth in potential tracks. Furthermore, time instants do not need to be evenly spaced, nor contain observations from all sources.

Gaussian Processes (GPs) [5] are a powerful tool for Bayesian nonlinear regression. Gaussian process mixture models allow GPs to be applied to data where there are local non-stationarities or discontinuities [6, 7, 8, 9]. The components of the mixture model are GPs and the prior probability of any given component is typically provided by a gating function. The role of the gating function is to dictate which GP is a priori most likely to be responsible for the data in any given region of the input space. The gating network forces each component of the GP mixture to be localized. Our model is inspired by the data association problem. For any given location in input space there will be multiple targets, perhaps corresponding to multiple objects in a tracking system. We are interested in

constructing a GP mixture model that can associate each of these targets with separate components. When there is ambiguity, the posterior distribution of targets will reflect this. We therefore propose a simple mixture model in which each component is global in its scope. The assignment of the data to each GP is performed sample-wise, independently of input space localization. In other words no gating function is used. We call this model the Overlapping Mixture of GPs (OMGP).

The remainder of this paper is organised as follows: In Section 2 we provide a brief review of GPs in the regression setting, Section 3 describes the OMGP model and how to efficiently learn and use it to make inference. Hyperparameter selection is described in Section 4 and experiments on several data sets are provided in Section 5. We wrap up in Section 6 with a brief discussion.

2 Brief Review of Gaussian Processes

In recent years, Gaussian Processes (GPs) have attracted a lot of attention due to their nice analytical properties and their state-of-the-art performance in regression tasks (see [10]). In this section we provide a brief summary of the main results for GP regression, see [5] for further details.

Assume that a set of N multi-dimensional inputs and their corresponding scalar outputs, $\mathcal{D} \equiv \{\mathbf{x}_n, y_n\}_{n=1}^m$, are available. The regression task is, given a new input \mathbf{x}_* , to obtain the predictive distribution for the corresponding observation y_* based on \mathcal{D} .

The GP regression model assumes that the observations can be modelled as some noiseless latent function of the inputs plus independent noise $y = f(\mathbf{x}) + \varepsilon$, and then sets a zero-mean¹ GP prior on the latent function $f(\mathbf{x}) \sim \mathcal{GP}(0, k(\mathbf{x}, \mathbf{x}'))$ and a Gaussian prior on $\varepsilon \sim \mathcal{N}(0, \sigma^2)$ on the noise, where $k(\mathbf{x}, \mathbf{x}')$ is a covariance function and σ^2 is a hyperparameter that specifies the noise power.

The covariance function $k(\mathbf{x}, \mathbf{x}')$ specifies the degree of coupling between $y(\mathbf{x})$ and $y(\mathbf{x}')$ and encodes the properties of the GP such as power level, smoothness, etc. One of the best-known covariance functions is the anisotropic squared exponential. It has the form of an unnormalized Gaussian, $k(\mathbf{x}, \mathbf{x}') = \sigma_0^2 \exp(-\frac{1}{2}\mathbf{x}^\top \mathbf{\Lambda}^{-1} \mathbf{x})$ and depends on the signal power σ_0^2 and the length-scales $\mathbf{\Lambda}$, where $\mathbf{\Lambda}$ is a diagonal matrix containing one length-scale per input dimension. Each length-scale controls how fast the correlation between outputs decays as the separation along the corresponding input dimension grows. We will collectively refer to all kernel parameters as $\boldsymbol{\theta}$.

The joint distribution of the available observations (collected in \mathbf{y}) and some unknown output $y(\mathbf{x}_*)$ is a multivariate Gaussian distribution, with parameters specified by the covariance function:

$$\begin{bmatrix} \mathbf{y} \\ y_* \end{bmatrix} \sim \mathcal{N}\left(\mathbf{0}, \begin{bmatrix} \mathbf{K} + \sigma^2 \mathbf{I}_N & \mathbf{k}_* \\ \mathbf{k}_*^\top & k_{**} + \sigma^2 \end{bmatrix}\right), \quad (1)$$

where $[\mathbf{K}]_{nn'} = k(\mathbf{x}_n, \mathbf{x}_{n'})$, $[\mathbf{k}_*]_n = k(\mathbf{x}_n, \mathbf{x}_*)$ and $k_{**} = k(\mathbf{x}_*, \mathbf{x}_*)$. \mathbf{I}_N is used to denote the identity matrix of size N . The notation $[\mathbf{A}]_{nn'}$ refers to entry at row n , column n' of \mathbf{A} . Likewise, $[\mathbf{a}]_n$ is used to reference the n -th element of vector \mathbf{a} .

From (1) and conditioning on the observed training outputs we can obtain the predictive distribution

$$\begin{aligned} p_{\text{GP}}(y_* | \mathbf{x}_*, \mathcal{D}) &= \mathcal{N}(y_* | \mu_{\text{GP}*}, \sigma_{\text{GP}*}^2) \\ \mu_{\text{GP}*} &= \mathbf{k}_*^\top (\mathbf{K} + \sigma^2 \mathbf{I}_N)^{-1} \mathbf{y} \quad \sigma_{\text{GP}*}^2 = \sigma^2 + k_{**} - \mathbf{k}_*^\top (\mathbf{K} + \sigma^2 \mathbf{I}_N)^{-1} \mathbf{k}_*, \end{aligned} \quad (2)$$

which is computable in $\mathcal{O}(N^3)$ time (due to the inversion of the $N \times N$ matrix $\mathbf{K} + \sigma^2 \mathbf{I}_N$).

Hyperparameters $\{\boldsymbol{\theta}, \sigma\}$ are typically selected by maximizing the marginal likelihood (also called evidence) of the observations, which is

$$\log p(\mathbf{y} | \boldsymbol{\theta}, \sigma) = -\frac{1}{2} \mathbf{y}^\top (\mathbf{K} + \sigma^2 \mathbf{I}_N)^{-1} \mathbf{y} - \frac{1}{2} |\mathbf{K} + \sigma^2 \mathbf{I}_N| - \frac{N}{2} \log(2\pi). \quad (3)$$

If analytical derivatives of (3) are available, optimization can be carried out using gradient methods, with each gradient computation taking $\mathcal{O}(N^3)$ time.

¹To make his assumption hold, the sample mean $\{y(\mathbf{x}_n)\}_{n=1}^m$ is usually subtracted from data before proceeding further.

When dealing with multi-output functions, instead of a single set of observations \mathbf{y} , D sets are available, $\mathbf{y}_1 \dots \mathbf{y}_D$, each corresponding to a different output dimension. In this case we can assume independence across the outputs and perform the above procedure independently for each dimension.

3 Overlapping Mixtures of Gaussian Processes

Our overlapping mixture of Gaussian processes (OMGP) model assumes that there exist M different latent functions $\{f^{(m)}(\mathbf{x})\}_{m=1}^M$ (which we will call trajectories) and that each output is produced by evaluating one of these functions at the corresponding input and adding Gaussian noise. The association between samples and latent functions is determined by the $N \times M$ binary indicator matrix \mathbf{Z} : Entry $[\mathbf{Z}]_{nm}$ being non-zero specifies that n -th data point was generated using trajectory m . Of course, only one non-zero entry per row is allowed in \mathbf{Z} .

To model multi-dimensional trajectories (i.e., multiple outputs), D latent functions per trajectory can be used $\{f_d^{(m)}(\mathbf{x})\}_{m=1, d=1}^{M, D}$. Note that there is no need to extend \mathbf{Z} to specifically handle the multi-output case, since all the outputs corresponding to a single input are the same data point and must belong to the same trajectory.

For convenience we will collect all the outputs in a single matrix $\mathbf{Y} = [\mathbf{y}_1 \dots \mathbf{y}_D]$ and all the latent functions of trajectory m in a single matrix $\mathbf{F}^{(m)} = [\mathbf{f}_1^{(m)} \dots \mathbf{f}_D^{(m)}]$. We will refer to all the latent functions as $\{\mathbf{F}^{(m)}\}$.

Given the above description, the likelihood of the OMGP model is

$$p(\mathbf{Y}|\{\mathbf{F}^{(m)}\}, \mathbf{Z}) = \prod_{n=1, m=1, d=1}^{N, M, D} \mathcal{N}([\mathbf{Y}]_{nd} | [\mathbf{F}^{(m)}]_{nd}, \sigma^2)^{[\mathbf{Z}]_{nm}}. \quad (4)$$

Following the standard Bayesian framework, we place priors on the unobserved latent variables

$$p(\mathbf{Z}) = \prod_{n=1, m=1}^{N, M} [\mathbf{\Pi}]_{nm}^{[\mathbf{Z}]_{nm}}, \quad p(\mathbf{F}^{(m)}|\mathbf{X}) = \prod_{m=1, d=1}^{M, D} \mathcal{N}(\mathbf{f}_d^{(m)} | \mathbf{0}, \mathbf{K}^{(m)}), \quad (5)$$

i.e., a multinomial distribution over the indicators (in which $\sum_{m=1}^M [\mathbf{\Pi}]_{nm} = 1 \forall n$) and independent GP priors over each latent function. We allow different covariance matrices for each trajectory. Though the multinomial distribution is specified here in its more general form, additional constraints are usually imposed, such as holding the prior probabilities constant for all data points. For the sake of clarity, we will omit the conditioning on the hyperparameters $\{\boldsymbol{\theta}, \mathbf{\Pi}, \sigma^2\}$, which can be assumed to be known for the moment.

The analytical computation of the posterior distribution $p(\mathbf{Z}, \{\mathbf{F}^{(m)}\}|\mathbf{X}, \mathbf{Y})$ is unfortunately intractable, so we will resort to approximate techniques.

3.1 Variational approximation

If the hyperparameters are known, it is possible to approximately compute the posterior using a variational approximation. We can use Jensen's inequality to construct a lower bound on the marginal likelihood as follows:

$$\begin{aligned} \log p(\mathbf{Y}|\mathbf{X}) &= \log \int p(\mathbf{Y}|\{\mathbf{F}^{(m)}\}, \mathbf{Z}) p(\mathbf{Z}) \prod_{m=1}^M p(\mathbf{F}^{(m)}|\mathbf{X}) d\{\mathbf{F}^{(m)}\} d\mathbf{Z} \\ &\geq \int q(\{\mathbf{F}^{(m)}\}, \mathbf{Z}) \log \frac{p(\mathbf{Y}|\{\mathbf{F}^{(m)}\}, \mathbf{Z}) p(\mathbf{Z}) \prod_{m=1}^M p(\mathbf{F}^{(m)}|\mathbf{X})}{q(\{\mathbf{F}^{(m)}\}, \mathbf{Z})} d\{\mathbf{F}^{(m)}\} d\mathbf{Z} = \mathcal{L}_{\text{VB}}. \end{aligned} \quad (6)$$

Here \mathcal{L}_{VB} is a lower bound on $\log p(\mathbf{Y}|\mathbf{X})$ for any *variational distribution* $q(\{\mathbf{F}^{(m)}\}, \mathbf{Z})$ and equality is attained if and only if $q(\{\mathbf{F}^{(m)}\}, \mathbf{Z}) = p(\mathbf{Z}, \{\mathbf{F}^{(m)}\}|\mathbf{X}, \mathbf{Y})$. Our objective is therefore to find a variational distribution that maximizes \mathcal{L}_{VB} , and thus becomes an approximation

to the true posterior. We will restrict our search to variational distributions that factorize as $q(\{\mathbf{F}^{(m)}\}, \mathbf{Z}) = q(\{\mathbf{F}^{(m)}\})q(\mathbf{Z})$.

If we assume that $q(\{\mathbf{F}^{(m)}\})$ is given (and therefore, also the marginals $q(\mathbf{f}_d^{(m)}) = \mathcal{N}(\mathbf{f}_d^{(m)} | \boldsymbol{\mu}_d^{(m)}, \boldsymbol{\Sigma}^{(m)})$ are available), it is possible to analytically maximize \mathcal{L}_{VB} with respect to $q(\mathbf{Z})$ by setting its derivative to zero and constraining it to be a probability density. The optimal $q(\mathbf{Z})$ is then:

$$q(\mathbf{Z}) = \prod_{n=1, m=1}^{N, M} [\hat{\boldsymbol{\Pi}}]_{nm}^{\mathbf{Z}_{nm}} \text{ with } [\hat{\boldsymbol{\Pi}}]_{nm} \propto [\boldsymbol{\Pi}]_{nm} \exp(a_{nm}) \quad (7)$$

$$\text{with } a_{nm} = \sum_{d=1}^D \left(-\frac{1}{2\sigma^2} \left(([\mathbf{y}_d]_n - [\boldsymbol{\mu}_d^{(m)}]_n)^2 + [\boldsymbol{\Sigma}^{(m)}]_{nn} \right) - \frac{1}{2} \log(2\pi\sigma^2) \right),$$

where we see that the (approximate) posterior distribution over the indicators $q(\mathbf{Z})$ factorizes for each sample.

Analogously, assuming $q(\mathbf{Z})$ as known, it is possible to analytically obtain the distribution over the latent functions that maximizes \mathcal{L}_{VB} . For the OMGP model, this distribution factorizes both over trajectories and dimensions, and is given by

$$q(\mathbf{f}_d^{(m)}) = \mathcal{N}(\mathbf{f}_d^{(m)} | \boldsymbol{\mu}_d^{(m)}, \boldsymbol{\Sigma}^{(m)}) \text{ with } \boldsymbol{\Sigma}^{(m)} = (\mathbf{K}^{-1(m)} + \mathbf{B}^{(m)})^{-1} \text{ and } \boldsymbol{\mu}_d^{(m)} = \boldsymbol{\Sigma}^{(m)} \mathbf{B}^{(m)} \mathbf{y}_d^{(m)} \quad (8)$$

where $\mathbf{B}^{(m)}$ is a diagonal matrix with elements $[\hat{\boldsymbol{\Pi}}]_{1m}/\sigma^2 \dots [\hat{\boldsymbol{\Pi}}]_{Nm}/\sigma^2$.

It is now possible to initialize $q(\mathbf{Z})$ and $q(\mathbf{f}_d^{(m)})$ from their prior distributions and iterate updates (7) and (8) to obtain increasingly refined approximation to the posterior. Since both steps are optimal with respect to the distribution that they compute, they are guaranteed to increase \mathcal{L}_{VB} , and therefore the algorithm is guaranteed to converge to a local maximum.

Monotonous convergence can be monitored by computing \mathcal{L}_{VB} after each update. \mathcal{L}_{VB} can be expressed as

$$\mathcal{L}_{\text{VB}} = \left\langle \log p(\mathbf{Y} | \{\mathbf{F}^{(m)}\}, \mathbf{Z}) \right\rangle_{q(\{\mathbf{F}^{(m)}\}, \mathbf{Z})} - \text{KL}(q(\{\mathbf{F}^{(m)}\}) || p(\{\mathbf{F}^{(m)}\})) - \text{KL}(q(\mathbf{Z}) || p(\mathbf{Z}))$$

where the first term is given by

$$\left\langle \log p(\mathbf{Y} | \{\mathbf{F}^{(m)}\}, \mathbf{Z}) \right\rangle_{q(\{\mathbf{F}^{(m)}\}, \mathbf{Z})} = \sum_{n, m}^{N, M} [\hat{\boldsymbol{\Pi}}]_{nm} a_{nm},$$

and the two remaining terms are the Kullback-Leibler (KL) divergences from the approximate posterior to the prior, which are straightforward to compute.

Update (7) takes only $\mathcal{O}(NM)$ computation time, whereas (8) takes $\mathcal{O}(MN^3)$ time, due to the M matrix inversions. The model as presented here has therefore the same limitations as conventional GPs regarding the size of the data sets that it can be applied to. However, when the posterior probability of some indicator $[\hat{\boldsymbol{\Pi}}]_{nm}$ is close to zero, sample n no longer affects trajectory m and can be dropped in its computation, thus reducing the cost. Furthermore, it is possible to use sparse GPs² to reduce this cost³ to $\mathcal{O}(MN)$ time by making use of the matrix inversion lemma.

3.2 Predictive distributions

The OMGP model can be used for a variety of tasks. In the data association problem (i.e., trajectory clustering) the task at hand is to cluster observations into trajectories, which can be achieved by assigning each observation to the trajectory that more likely generated it, i.e., to assign label $m^* = \arg \max_m [\hat{\boldsymbol{\Pi}}]_{nm}$ to the n -th observation, so no further computations are necessary. For other tasks,

²Such as the standard FITC approximation, described in [11] or the variational approach introduced in [12].

³Obviously, the cost also depends on the quality of the approximation by a constant factor. If the FITC approximation with r pseudo-inputs (or other rank- r approximation) is used, the computational complexity could be expressed as $\mathcal{O}(MNr^2)$.

however, it can be necessary to obtain predictive distributions over the output space at new locations. Under the variational approximation, this predictive distributions can be computed analytically.

The predictive distribution in the output dimension d corresponding to a new test input location \mathbf{x}_* can be expressed as

$$\begin{aligned} p(y_{*d}|\mathbf{x}_*, \mathbf{X}, \mathbf{Y}) &= \sum_{m=1}^M [\mathbf{\Pi}]_{*m} \int p(y_{*d}|\mathbf{f}_d^{(m)}, \mathbf{x}_*, \mathbf{X}) p(\mathbf{f}_d^{(m)}|\mathbf{X}, \mathbf{Y}) d\mathbf{f}_d^{(m)} \\ &\approx \sum_{m=1}^M [\mathbf{\Pi}]_{*m} \int p(y_{*d}|\mathbf{f}_d^{(m)}, \mathbf{x}_*, \mathbf{X}) q(\mathbf{f}_d^{(m)}|\mathbf{X}, \mathbf{Y}) d\mathbf{f}_d^{(m)} = \sum_{m=1}^M [\mathbf{\Pi}]_{*m} \mathcal{N}(y_{*d}|\mu_{*d}^{(m)}, \sigma_{*d}^{2(m)}) \end{aligned}$$

with

$\mu_{*d}^{(m)} = \mathbf{k}_*^{\top(m)} (\mathbf{K}^{(m)} + \mathbf{B}^{-1(m)})^{-1} \mathbf{y}_d$, $\sigma_{*d}^{2(m)} = \sigma^2 + k_{**} - \mathbf{k}_*^{\top(m)} (\mathbf{K}^{(m)} + \mathbf{B}^{-1(m)})^{-1} \mathbf{k}_*^{(m)}$, i.e., a Gaussian mixture under the approximate posterior. The mixing factors $[\mathbf{\Pi}]_{*m}$ are the prior probabilities of each component, one of the given hyperparameters of the model, and typically constant for all inputs.

Note the correspondence of these predictive equations with the standard predictions for GP regression (2). The only difference is in the noise component which is scaled for each sample according to $[\hat{\mathbf{\Pi}}]_{nm}^{-1}$. I. e., as the posterior probability of a sample belonging to the current trajectory (sometimes known as responsibility) decays, the amount of noise associated to that sample is proportionally grown, thus reducing its effect on the posterior process.

4 Model selection

So far we have assumed that all the hyperparameters of the model are known. However, in practice, some procedure to select them is needed. The most straightforward way of achieving this would be to select them so as to maximize \mathcal{L}_{VB} , interleaving this procedure with updates (7) and (8). However, when the quality of this bound is sensitive to changes of the model hyperparameters results in very slow convergence. A solution to this problem is described in [13] where the advantages of maximizing an alternative, tighter bound on the likelihood are shown.

The improved bound proposed in [13] is still a lower bound on the likelihood but it can be proved to also be a an upper bound on the standard variational bound \mathcal{L}_{VB} . As shown in [13], if we subtract \mathcal{L}_{VB} from the improved bound, the result takes on the form of a KL-divergence. This fact can be used both to show that it upper-bounds \mathcal{L}_{VB} (since KL-divergences are always positive) and to name the new bound, which is referred to as the KL-corrected variational bound.

The KL-corrected bound for the OMGP model arises when the term $p(\mathbf{Y}|\{\mathbf{F}^{(m)}\}, \mathbf{Z})p(\mathbf{Z})$ from the true marginal likelihood (6) is replaced with $q(\mathbf{Z}) \log \frac{p(\mathbf{Y}|\{\mathbf{F}^{(m)}\}, \mathbf{Z})p(\mathbf{Z})}{q(\mathbf{Z})}$, which, after integrating over \mathbf{Z} and according to Jensen's inequality, constitutes a lower bound for any distribution $q(\mathbf{Z})$:

$$\begin{aligned} \log p(\mathbf{Y}|\mathbf{X}) &\geq \log \int \prod_{m=1}^M p(\mathbf{F}^{(m)}|\mathbf{X}) e^{\int q(\mathbf{Z}) \log \frac{p(\mathbf{Y}|\{\mathbf{F}^{(m)}\}, \mathbf{Z})p(\mathbf{Z})}{q(\mathbf{Z})} d\mathbf{Z}} d\{\mathbf{F}^{(m)}\} = \mathcal{L}_{\text{CorrVB}} \\ &= \sum_{m=1, d=1}^{M, D} \log \mathcal{N}(\mathbf{y}_d^{(m)}|\mathbf{0}, \mathbf{K}^{(m)} + \mathbf{B}^{-1(m)}) - \text{KL}(q(\mathbf{Z})||p(\mathbf{Z})) + \frac{D}{2} \sum_{n=1, m=1}^{N, M} \log \frac{(2\pi\sigma^2)^{1-[\hat{\mathbf{\Pi}}]_{nm}}}{[\hat{\mathbf{\Pi}}]_{nm}}. \end{aligned}$$

The KL-corrected lower bound $\mathcal{L}_{\text{CorrVB}}$ can be computed analytically and has the advantage with respect to \mathcal{L}_{VB} , of depending only on $q(\mathbf{Z})$ (and not $q(\{\mathbf{F}^{(m)}\})$), since it is possible to integrate $\prod_{m=1}^M p(\mathbf{F}^{(m)}|\mathbf{X})$ out analytically. While $\mathcal{L}_{\text{CorrVB}}$ provides an improved bound, which remains more stable across different hyperparameter selections, it does not provide a simple way to compute the approximate posterior $q(\mathbf{Z})$, so the iterative minimization of \mathcal{L}_{VB} presented in the previous section is still used to obtain it. Direct minimization of this bound is another option that we will pursue in further work.

To perform model selection in the OMGP, we interleave the iterations described in the previous section (which are run until convergence) with the optimization of the hyperparameters (which is achieved by minimizing $\mathcal{L}_{\text{CorrVB}}(\boldsymbol{\theta}, \mathbf{\Pi}, \sigma^2)$ using conjugate gradient descent).

5 Experiments

5.1 Data association tasks

We first apply OMGP to perform data association on a toy data set. The targets perform circular motions, one clockwise and one counterclockwise. The position of both targets (plus Gaussian noise) at known time instants is available, without knowledge of which position corresponds to which target, see Fig. 1(a). Both trajectories are circles with the same center and radius, hence the sources cross each other twice per revolution. As shown in Fig. 1(b), OMGP is capable of successfully identifying the unknown trajectories.

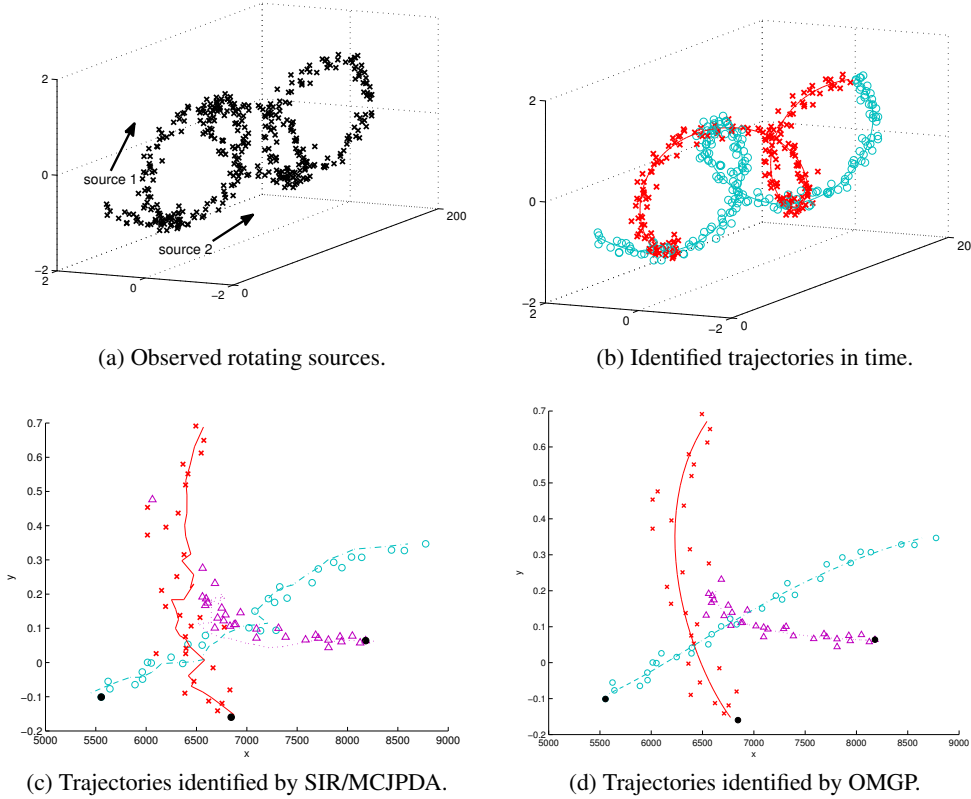


Figure 1: Top row: Observations and data association solution for two sources that move in opposite circles. Bottom row: Missile-to-air data association problem with three sources. The starting point of each source is marked with a black dot.

Next, we consider a missile-to-air tracking scenario as described in [4]. The motion dynamics of this scenario are defined by the following state-space equations:

$$\mathbf{s}_{t+1} = \begin{bmatrix} \mathbf{I}_3 & T\mathbf{I}_3 \\ \mathbf{O}_3 & \mathbf{I}_3 \end{bmatrix} \mathbf{s}_t + \begin{bmatrix} \frac{T^2}{2}\mathbf{I}_3 \\ T\mathbf{I}_3 \end{bmatrix} \mathbf{v}_t; \quad \mathbf{r}_t = h(\mathbf{s}_t) = \begin{bmatrix} \sqrt{X_t^2 + Y_t^2 + Z_t^2} \\ \arctan(\frac{Y_t}{X_t}) \\ \arctan(\frac{-Z_t}{\sqrt{X_t^2 + Y_t^2}}) \end{bmatrix} + \mathbf{e}_t.$$

In this model, the state vector $\mathbf{s}_t = [X_t, Y_t, Z_t, V_{x,t}, V_{y,t}, V_{z,t}]$ contains the source position and velocity components, \mathbf{r}_t contains the observed measurements, T is the sampling interval, and \mathbf{I}_3 and \mathbf{O}_3 represent the 3×3 unity matrix and null matrix, respectively. The process noise \mathbf{v}_t and measurement noise \mathbf{e}_t are assumed Gaussian, $\mathbf{v}_t \in N(0, \mathbf{Q})$ and $\mathbf{e}_t \in N(0, \mathbf{R})$. For more details refer to [4]. The problem posed in [4] consists in tracking two sources and estimating their unknown state vector, given their correct initial states $\mathbf{s}_0^1 = [6500, -1000, 2000, -50, 100, 0]$ and $\mathbf{s}_0^2 = [5050, -450, 2000, 100, 50, 0]$. We consider a more complex scenario by adding a third source, with initial state $\mathbf{s}_0^3 = [8000, 500, 2000, -100, 0, 0]$. We then apply OMGP and the SIR/MCJPDA

filter from [4] to perform data association for the observations. The SIR/MCJPCA filter consists of a set of joint particle filters that perform tracking of multiple sources, combined with a joint probability data association (JPDA) technique which provides instantaneous data association. The number of particles used in this experiment is 25000. To evaluate the performance of both algorithms, we measure the number of observations that are assigned to the wrong trajectories, out of a total of 90 observations.

The trajectories obtained by each method can be found in Fig. 1 (bottom row), along with the predicted measurements. The SIR/MCJPCA filter erroneously assigns 16 observations to different trajectories, compared to 1 wrongly assigned observation for OMGP. The SIR/MCJPCA filter clearly shows more errors when the observations are close, since it performs data association in each time instant and lacks knowledge of future data. The solution of OMGP, on the other hand, is based on a global evaluation of the entire trajectories. Finally, notice that the SIR/MCJPCA filter has complete knowledge of the used state-space model and the initial state vectors \mathbf{x}_0^i in this implementation, while OMGP is completely blind in this regard.

5.2 Regression tasks

We now consider application of the model in more standard regression tasks. In particular we consider tasks where the target density is multimodal perhaps because the data comes from multiple sources.

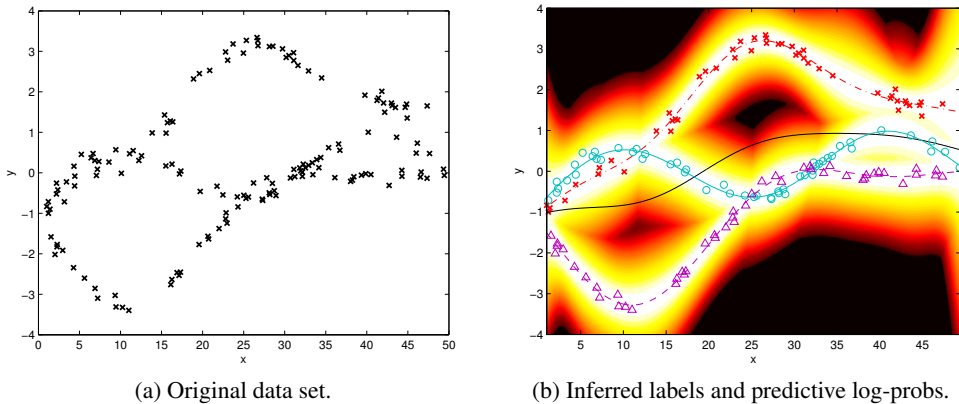


Figure 2: Posterior log-probability of the OMGP model and label inference.

Multilevel regression Consider the data set from Fig. 2(a), which corresponds to observations from three independent functions. A normal GP would fail to produce valid multimodal outputs and previously proposed mixtures of GPs would restrict the component GPs to local parts of the space. OMGP can properly label each observation according to the generating function and provide multimodal predictive distributions, as depicted in Fig. 2(b).

Fig. 2 can also be interpreted as measurements of the position of three particles moving along one dimension, of which snapshots are taken at irregular time intervals (horizontal axis). Each snapshot introduces noise in the position measurement and does not necessarily capture the position of all the particles. In this case OMGP could be used to predict the position of any particle at any given point in time, as well as to properly label the samples in each snapshot.

Robust regression Since each GP in the mixture can use a different covariance function, it is possible to use a GP to capture unrelated outliers and another one to interpolate the main function. This is easily achieved by a mixture of two GPs, one with the ARD-SE covariance function and another with $k(x, x') = b^2 \delta(x, x')$, i.e., white noise. We consider the problem of regression in a noisy sinc in which some outliers have been introduced in Fig. 3 (top row). Observe how OMGP a) identifies the outliers and b) ignores them, resulting in much better predictive means and variances.

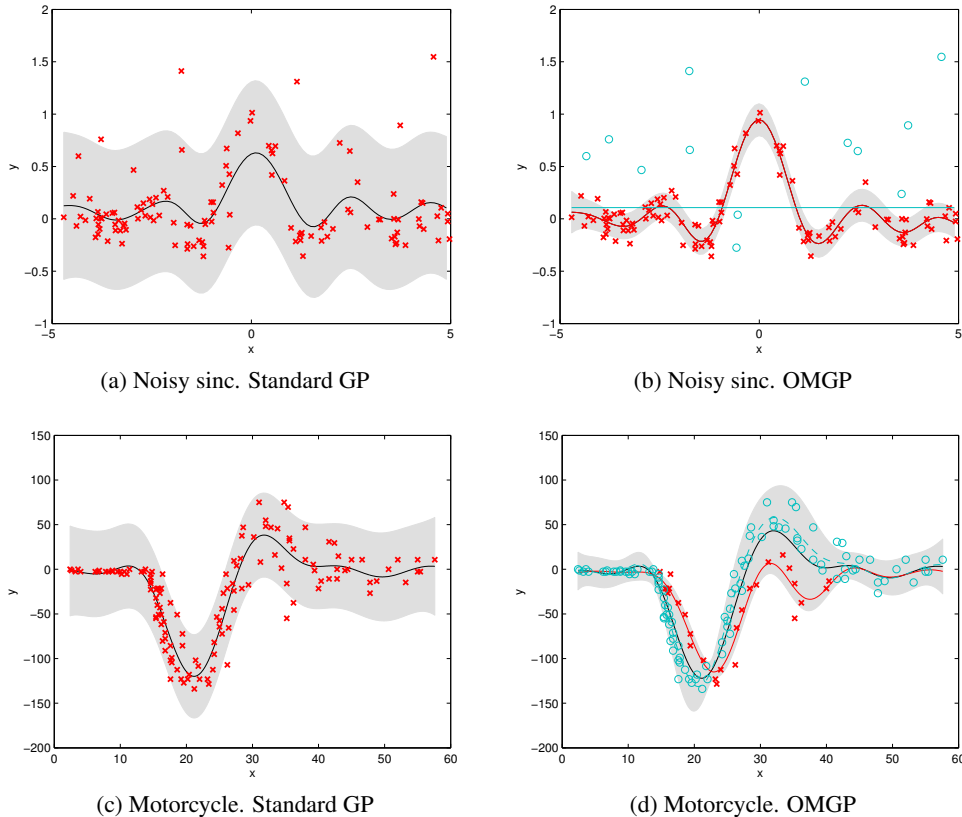


Figure 3: Predictive means and variances for two different data sets. The shaded area denotes ± 2 standard deviations around the mean. Top row: Noisy sinc with outliers. (a) Standard GP and (b) OMGP with a noise-only component. (Only the predictive mean and variance of the signal component is depicted, which includes noise σ^2). Bottom row: Silverman’s motorcycle data set.

Heteroscedastic behavior Finally, Fig. 3 (bottom row) shows the results of running a GP and OMGP on the motorcycle data set from [14]. Two components have been identified, which might or might not correspond to two actual physical mechanisms alternatively producing observations. The predictive variances show improved behaviour with respect to the standard GP.

6 Discussion and future work

In this work we have introduced a novel GP mixture model inspired by multi-target tracking problems. The new model has the important difference with respect to previous approaches of using global mixture components and assigning samples to components by relying on their value in output space, instead of input space (as it is done when gating functions are used).

A simple and efficient algorithm for inference relying on the variational Bayesian framework has been provided. The model can be applied in practice due to the use of an improved, KL-corrected variational bound to learn the hyperparameters. Direct optimization of this bound both to obtain an approximate posterior and to learn the hyperparameters will be considered in a further work.

The OMGP model offers promising results when tracking moving targets, as has been illustrated experimentally in Section 5 and compares favourably with established methods in the field. Also, through imaginative application of the model using different covariance functions we were able to adapt the approach to robust regression and heteroscedastic noise.

Naive implementation of GPs limits their applicability to only a few thousand data samples. However, recent advances in sparse approximations (e.g. [11, 12]) greatly should enable our approach to be applied to much larger data sets.

References

- [1] Y. Bar-Shalom. *Tracking and data association*. Academic Press Professional, Inc. San Diego, CA, USA, 1987.
- [2] I. J. Cox. A review of statistical data association techniques for motion correspondence. *International Journal of Computer Vision*, 10(1):53–66, 1993.
- [3] D. Reid. An algorithm for tracking multiple targets. *Automatic Control, IEEE Transactions on*, 24(6):843–854, dec 1979.
- [4] R. Karlsson and F. Gustafsson. Monte Carlo data association for multiple target tracking. *IEEE International Seminar on Target Tracking: Algorithms and Applications*, 1:13, 2001.
- [5] C. E. Rasmussen and C. K. I. Williams. *Gaussian Processes for Machine Learning*. MIT Press, 2006.
- [6] V. Tresp. A Bayesian committee machine. *Neural Computation*, 12:2719–2741, 2000.
- [7] C. E. Rasmussen and Z. Ghahramani. Infinite mixtures of Gaussian process experts. In *Advances in Neural Information Processing Systems 14*, pages 881–888. MIT Press, 2002.
- [8] E. Meeds and S. Osindero. An alternative infinite mixture of Gaussian process experts. In *Advances in Neural Information Processing Systems 18*, pages 883–890. MIT Press, 2006.
- [9] C. Yuan and C. Neubauer. Variational mixture of Gaussian process experts. In *Advances in Neural Information Processing Systems 21*, pages 1897–1904. 2009.
- [10] C. E. Rasmussen. *Evaluation of Gaussian Processes and other Methods for Non-linear Regression*. PhD thesis, University of Toronto, 1996.
- [11] E. Snelson and Z. Ghahramani. Sparse Gaussian processes using pseudo-inputs. In *Advances in Neural Information Processing Systems 18*, pages 1259–1266. MIT Press, 2006.
- [12] M. K. Titsias. Variational learning of inducing variables in sparse Gaussian processes. In *Proceedings of the 12th International Workshop on AI Stats*, pages 567–574, 2009.
- [13] N. J. King and N. D. Lawrence. Fast variational inference for Gaussian process models through KL-correction. In *ECML*, pages 270–281. Lecture Notes in Computer Science, Berlin, 2006.
- [14] B. W. Silverman. Some aspects of the spline smoothing approach to non-parametric regression curve fitting. *Journal of the Royal Statistical Society*, 47:1–52, 1985.