# Bayesian Alignments of Warped Multi-Output Gaussian Processes

**Markus Kaiser**
Siemens AG
Technical University of Munich
markus.kaiser@siemens.com

**Clemens Otte**
Siemens AG
clemens.otte@siemens.com

**Thomas Runkler**
Siemens AG
Technical University of Munich
thomas.runkler@siemens.com

**Carl Henrik Ek**
University of Bristol
carlhenrik.ek@bristol.ac.uk

## Abstract

We propose a novel Bayesian approach to modelling nonlinear alignments of time series based on latent shared information. We apply the method to the real-world problem of finding common structure in the sensor data of wind turbines introduced by the underlying latent and turbulent wind field. The proposed model allows for both arbitrary alignments of the inputs and non-parametric output warpings to transform the observations. This gives rise to multiple deep Gaussian process models connected via latent generating processes. We present an efficient variational approximation based on nested variational compression and show how the model can be used to extract shared information between dependent time series, recovering an interpretable functional decomposition of the learning problem. We show results for an artificial data set and real-world data of two wind turbines.

## 1 Introduction

Many real-world systems are inherently hierarchical and connected. Ideally, a machine learning method should model and recognize such dependencies. Take wind power production, which is one of the major providers for renewable energy today, as an example: To optimize the efficiency of a wind turbine the speed and pitch have to be controlled according to the local wind conditions (speed and direction). In a wind farm turbines are typically equipped with sensors for wind speed and direction. The goal is to use these sensor data to produce accurate estimates and forecasts of the wind conditions at every turbine in the farm. For the ideal case of a homogeneous and very slowly changing wind field, the wind conditions at each geometrical position in a wind farm can be estimated using the propagation times (time warps) computed from geometry, wind speed, and direction [21, 4, 18]. In the real world, however, wind fields are not homogeneous, exhibit global and local turbulences, and interfere with the turbines and the terrain inside and outside the farm and further, sensor faults may lead to data loss. This makes it extremely difficult to construct accurate analytical models of wind propagation in a farm. Also, standard approaches for extracting such information from data, e.g. generalized time warping [24], fail at this task because they rely on a high signal to noise ratio. Instead, we want to construct Bayesian nonlinear dynamic data based models for wind conditions and warpings which handle the stochastic nature of the system in a principled manner.

In this paper, we look at a generalization of this type of problem and propose a novel Bayesian approach to finding nonlinear alignments of time series based on latent shared information. We view the power production of different wind turbines as the outputs of a multi-output Gaussian process

Preprint. Work in progress.

(MO-GP) [1] which models the latent wind fronts. We embed this model in a hierarchy, adding a layer of non-linear alignments on top and a layer of non-linear warpings [19, 14] below which increases flexibility and encodes the original generative process. We show how the resulting model can be interpreted as a group of deep Gaussian processes with the added benefit of covariances between different outputs. The imposed structure is used to formulate prior knowledge in a principled manner, restrict the representational power to physically plausible models and recover the desired latent wind fronts and relative alignments. The presented model can be interpreted as a group of $D$ deep GPs all of which share one layer which is a MO-GP. This MO-GP acts as an interface to share information between the different GPs which are otherwise conditionally independent.

The paper has the following contributions: In Section 2, we propose a hierarchical, warped and aligned multi-output Gaussian process (AMO-GP). In Section 3, we present an efficient learning scheme via an approximation to the marginal likelihood which allows us to fully exploit the regularization provided by our structure, yielding highly interpretable results. We show these properties for an artificial data set and for real-world data of two wind turbines in Section 4.

## 2 Model Definition

We are interested in formulating shared priors over a set of functions $\{f_d\}_{d=1}^D$ using GPs, thereby directly parameterizing their interdependencies. In a traditional GP setting, multiple outputs are considered conditionally independent given the inputs, which significantly reduces the computational cost but also prevents the utilization of shared information. Such interdependencies can be formulated via *convolution processes (CPs)* as proposed by Boyle and Frean [5], a generalization of the *linear model of coregionalization (LMC)* [13, 7]. In the CP framework, the output functions are the result of a convolution of the latent processes $w_r$ with smoothing kernel functions $T_{d,r}$ for each output $f_d$, defined as

$$f_d(\boldsymbol{x}) = \sum_{r=1}^{R} \int T_{d,r}(\boldsymbol{x} - \boldsymbol{z}) \cdot w_r(\boldsymbol{z}) \, \mathrm{d}\boldsymbol{z}. \tag{1}$$

In this model, the convolutions of the latent processes generating the different outputs are all performed around the same point $\boldsymbol{x}$. We generalize this by allowing different *alignments* of the observations which depend on the position in the input space. This allows us to model the changing relative interaction times for the different latent wind fronts as described in the introduction. We also assume that the dependent functions $f_d$ are latent themselves and the data we observe is generated via independent noisy nonlinear transformations of their values. Every function $f_d$ is augmented with an alignment function $a_d$ and a warping $g_d$ on which we place independent GP priors.

For simplicity, we assume that the outputs are evaluated all at the same positions $\boldsymbol{X} = \{\boldsymbol{x_n}\}_{n=1}^N$. This can easily be generalized to different input sets for every output. In our application, the $\boldsymbol{x_n}$ are one-dimensional time indices. However, since the model can be generalized to multi-dimensional inputs, we do not restrict ourselves to the one-dimensional case. We note that in the multi-dimensional case, reasoning about priors on alignments can be challenging. We call the observations associated with the $d$-th function $\boldsymbol{y_d}$ and use the stacked vector $\boldsymbol{y} = (\boldsymbol{y_1}, \dots, \boldsymbol{y_D})$ to collect the data of all outputs. The final model is then given by

$$\boldsymbol{y_d} = g_d(f_d(a_d(\boldsymbol{X}))) + \boldsymbol{\epsilon_d}, \tag{2}$$

where $\boldsymbol{\epsilon_d} \sim \mathcal{N}(0, \sigma_{y,d}^2 \mathbf{I})$ is a noise term. The functions are applied element-wise. This encodes the generative process described above: For every turbine $\boldsymbol{y_d}$, observations at positions $\boldsymbol{X}$ are generated by first aligning to the latent wind fronts using $a_d$, applying the front in $f_d$, imposing turbine-specific components $g_d$ and adding noise in $\boldsymbol{\epsilon_d}$.

We assume independence between $a_d$ and $g_d$ across outputs and apply GP priors of the form $a_d \sim \mathcal{GP}(\mathrm{id}, k_{a,d})$ and $g_d \sim \mathcal{GP}(\mathrm{id}, k_{g,d})$. By setting the prior mean to the identity function $\mathrm{id}(x) = x$, the standard CP model is our default assumption. During learning, the model can choose the different $a_d$ and $g_d$ in a way to reveal the independent shared latent processes $\{w_r\}_{r=1}^R$ on which we also place GP priors $w_r \sim \mathcal{GP}(0, k_{u,r})$. Similar to Boyle and Frean [5], we assume the latent processes to be independent white noise processes by setting $\mathrm{cov}[w_r(\boldsymbol{z}), w_{r'}(\boldsymbol{z'})] = \delta_{rr'}\delta_{\boldsymbol{zz'}}$. Under this prior, the $f_d$ are also GPs with zero mean and $\mathrm{cov}[f_d(\boldsymbol{x}), f_{d'}(\boldsymbol{x'})] = \sum_{r=1}^R \int T_{d,r}(\boldsymbol{x} - \boldsymbol{z}) T_{d',r}(\boldsymbol{x'} - \boldsymbol{z}) \, \mathrm{d}\boldsymbol{z}$.

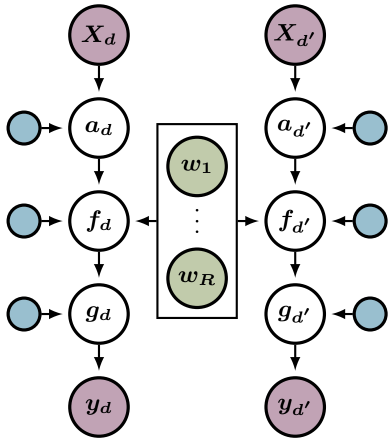


**Figure 1:** The graphical model of AMO-GP with variational parameters (blue). A CP, informed by $R$ latent processes, models shared information between multiple data sets with nonlinear alignments and warpings. This CP connects multiple deep GPs through a shared layer.

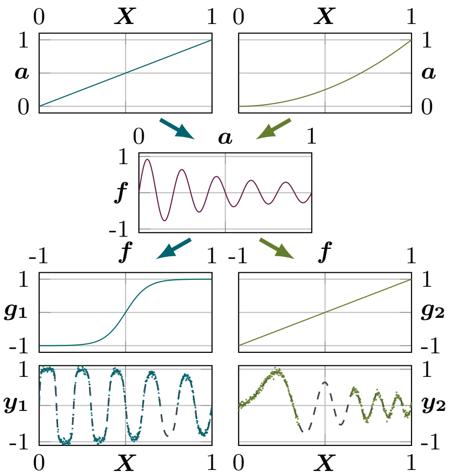


**Figure 2:** An artificial example of hierarchical composite data with multiple observations of shared latent information. This hierarchy generates two data sets using a dampened sine function which is never observed directly.

Using the squared exponential kernel for all $T_{d,r}$, the integral can be shown to have a closed form solution. With $\{\sigma_{d,r}, \boldsymbol{\ell_{d,r}}\}$ denoting the kernel hyper parameters associated with $T_{d,r}$, it is given by

$$\operatorname{cov}[f_d(\boldsymbol{x}), f_{d'}(\boldsymbol{x'})] = \sum_{r=1}^{R} \frac{(2\pi)^{\frac{K}{2}} \sigma_{d,r}\sigma_{d',r}}{\prod_{k=1}^{K} \hat{\ell}^{-1}_{d,d',r,k}} \exp\left( -\frac{1}{2} \sum_{k=1}^{K} \frac{(x_k - x'_k)^2}{\hat{\ell}^2_{d,d',r,k}} \right), \tag{3}$$

where $\boldsymbol{x}$ is $K$-dimensional and $\hat{\ell}_{d,d',r,k} = \sqrt{\ell^2_{d,r,k} + \ell^2_{d',r,k}}$.

## 3 Variational Approximation

Since exact inference in this model is intractable, we present a variational approximation to the model's marginal likelihood in this section. A detailed derivation of the variational bound can be found in Appendix A. Analogously to $\boldsymbol{y}$, we denote the random vectors which contain the function values of the respective functions and outputs as $\boldsymbol{a}$ and $\boldsymbol{f}$. The joint probability distribution of the data can then be written as

$$\begin{aligned} &\mathrm{p}(\boldsymbol{y}, \boldsymbol{f}, \boldsymbol{a} \mid \boldsymbol{X}) = \\ &\qquad \mathrm{p}(\boldsymbol{f} \mid \boldsymbol{a}) \prod_{d=1}^{D} \mathrm{p}(\boldsymbol{y_d} \mid \boldsymbol{f_d})\, \mathrm{p}(\boldsymbol{a_d} \mid \boldsymbol{X}), \end{aligned} \qquad \begin{aligned} \boldsymbol{a_d} \mid \boldsymbol{X} &\sim \mathcal{N}(\boldsymbol{X}, \boldsymbol{K_{a,d}} + \sigma^2_{a,d}\mathbf{I}), \\ \boldsymbol{f} \mid \boldsymbol{a} &\sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{K_f} + \sigma^2_{f}\mathbf{I}), \\ \boldsymbol{y_d} \mid \boldsymbol{f_d} &\sim \mathcal{N}(\boldsymbol{f_d}, \boldsymbol{K_{g,d}} + \sigma^2_{y,d}\mathbf{I}). \end{aligned} \tag{4}$$

Here, we use $\boldsymbol{K}$ to refer to the Gram matrices corresponding to the respective GPs. All but the convolution processes factorize over the different levels of the model as well as the different outputs.

### 3.1 Variational Lower Bound

To approximate a single deep GP, that is a single string of GPs stacked on top of each other, Hensman and Lawrence [11] proposed nested variational compression in which every GP in the hierarchy is handled independently. In order to arrive at their lower bound they make two variational approximations. First, they consider a variational approximation $\mathrm{q}(\hat{\boldsymbol{a}}, \boldsymbol{u}) = \mathrm{p}(\hat{\boldsymbol{a}} \mid \boldsymbol{u})\, \mathrm{q}(\boldsymbol{u})$ to the true posterior of a single GP first introduced by Titsias [22]. In this approximation, the original model is augmented with *inducing variables* $\boldsymbol{u}$ together with their *inducing points* $\boldsymbol{Z}$ which are assumed to be latent observations of the same function and are thus jointly Gaussian with the observed data. In contrast to [22], the distribution $\mathrm{q}(\boldsymbol{u})$ is not chosen optimally but optimized using the closed form $\mathrm{q}(\boldsymbol{u}) \sim \mathcal{N}(\boldsymbol{u} \,|\, \boldsymbol{m}, \boldsymbol{S})$. This gives rise to the Scalable Variational GP presented in [10]. Second, in order to apply this variational bound for the individual GPs recursively, uncertainties have to be propagated through subsequent layers and inter-layer cross-dependencies are avoided using another variational approximation. The variational lower bound for the AMO-GP is given by

$$
\begin{aligned}
\log \mathrm{p}(\boldsymbol{y} \,|\, \boldsymbol{X}, \boldsymbol{Z}, \boldsymbol{u}) \geq{} & \sum_{d=1}^{D} \log \mathcal{N}\Big(\boldsymbol{y_d} \,\Big|\, \boldsymbol{\Psi_{g,d}} \boldsymbol{K}^{-1}_{\boldsymbol{u_{g,d}u_{g,d}}} \boldsymbol{m_{g,d}}, \sigma^2_{y,d}\mathbf{I}\Big) - \sum_{d=1}^{D} \frac{1}{2\sigma^2_{a,d}} \operatorname{tr}(\boldsymbol{\Sigma_{a,d}}) \\
& - \frac{1}{2\sigma^2_{f}} \Big(\psi_f - \operatorname{tr}\Big(\boldsymbol{\Phi_f} \boldsymbol{K}^{-1}_{\boldsymbol{u_f u_f}}\Big)\Big) - \sum_{d=1}^{D} \frac{1}{2\sigma^2_{y,d}} \Big(\psi_{g,d} - \operatorname{tr}\Big(\boldsymbol{\Phi_{g,d}} \boldsymbol{K}^{-1}_{\boldsymbol{u_{g,d}u_{g,d}}}\Big)\Big) \\
& - \sum_{d=1}^{D} \mathrm{KL}(\mathrm{q}(\boldsymbol{u_{a,d}}) \,\|\, \mathrm{p}(\boldsymbol{u_{a,d}})) - \mathrm{KL}(\mathrm{q}(\boldsymbol{u_f}) \,\|\, \mathrm{p}(\boldsymbol{u_f})) - \sum_{d=1}^{D} \mathrm{KL}(\mathrm{q}(\boldsymbol{u_{y,d}}) \,\|\, \mathrm{p}(\boldsymbol{u_{y,d}})) \\
& - \frac{1}{2\sigma^2_{f}} \operatorname{tr}\Big(\big(\boldsymbol{\Phi_f} - \boldsymbol{\Psi}^{\mathsf{T}}_{\boldsymbol{f}} \boldsymbol{\Psi_f}\big) \boldsymbol{K}^{-1}_{\boldsymbol{u_f u_f}} \big(\boldsymbol{m_f} \boldsymbol{m}^{\mathsf{T}}_{\boldsymbol{f}} + \boldsymbol{S_f}\big) \boldsymbol{K}^{-1}_{\boldsymbol{u_f u_f}}\Big) \\
& - \sum_{d=1}^{D} \frac{1}{2\sigma^2_{y,d}} \operatorname{tr}\Big(\big(\boldsymbol{\Phi_{g,d}} - \boldsymbol{\Psi}^{\mathsf{T}}_{\boldsymbol{g,d}} \boldsymbol{\Psi_{g,d}}\big) \boldsymbol{K}^{-1}_{\boldsymbol{u_{g,d}u_{g,d}}} \big(\boldsymbol{m_{g,d}} \boldsymbol{m}^{\mathsf{T}}_{\boldsymbol{g,d}} + \boldsymbol{S_{g,d}}\big) \boldsymbol{K}^{-1}_{\boldsymbol{u_{g,d}u_{g,d}}}\Big),
\end{aligned}
\tag{5}
$$

where KL denotes the KL-divergence. A detailed derivation can be found in Appendix A. The bound contains one Gaussian fit term per output dimension and a series of regularization terms for every GP in the hierarchy. The KL-divergences connect the variational approximations to the prior and the different trace terms regularize the variances of the different GPs (for a detailed discussion see [11]). This bound depends on the hyper parameters of the kernel and likelihood $\{\boldsymbol{\ell}, \boldsymbol{\sigma}\}$ and the variational parameters $\{\boldsymbol{Z_{l,d}}, \boldsymbol{m_{l,d}}, \boldsymbol{S_{l,d}} \,|\, l \in \{\boldsymbol{a}, \boldsymbol{f}, \boldsymbol{d}\}, d \in [D]\}$.

The bound can be calculated in $\mathcal{O}(NM^2)$ time and factorizes along the data points which enables stochastic optimization. Since every of the $N$ data points is associated with one of the $D$ outputs, the computational cost of the model is independent of $D$. Information is only shared between the different outputs using the inducing points in $\boldsymbol{f}$. As the different outputs share a common function, increasing $D$ allows us to reduce the number of variational parameters per output, because the shared function can still be represented completely.

A central component of this bound are expectations over kernel matrices, the three $\Psi$-statistics $\psi_f = \mathbb{E}_{\mathrm{q}(\boldsymbol{a})}[\operatorname{tr}(\boldsymbol{K_{ff}})]$, $\boldsymbol{\Psi_f} = \mathbb{E}_{\mathrm{q}(\boldsymbol{a})}[\boldsymbol{K_{fu}}]$ and $\boldsymbol{\Phi_f} = \mathbb{E}_{\mathrm{q}(\boldsymbol{a})}[\boldsymbol{K_{uf}} \boldsymbol{K_{fu}}]$. Closed form solutions for these statistics depend on the choice of kernel and are known for specific kernels, such as linear or RBF kernels, for example shown in [8]. In the following subsection we will give closed form solutions for these statistics required in the shared CP-layer of our model.

### 3.2 Convolution Kernel Expectations

The uncertainty about the first layer is captured by the variational distribution of the latent alignments $\boldsymbol{a}$ given by $\mathrm{q}(\boldsymbol{a}) \sim \mathcal{N}(\boldsymbol{\mu_a}, \boldsymbol{\Sigma_a})$. Every aligned point in $\boldsymbol{a}$ corresponds to one output of $\boldsymbol{f}$ and ultimately to one of the $\boldsymbol{y_d}$. Since the closed form of the multi output kernel depends on the choice of outputs, we will use the notation $\hat{f}(\boldsymbol{a_n})$ to denote $f_d(\boldsymbol{a_n})$ such that $\boldsymbol{a_n}$ is associated with output $d$.

For simplicity, we only consider one single latent process $w_r$. Since the latent processes are independent, the results can easily be generalized to multiple processes. Then, $\psi_f$ is given by

$$
\psi_f = \mathbb{E}_{\mathrm{q}(\boldsymbol{a})}[\operatorname{tr}(\boldsymbol{K_{ff}})] = \sum_{n=1}^{N} \hat{\sigma}^2_{nn}. \tag{6}
$$

Similar to the notation $\hat{f}(\cdot)$, we use the notation $\hat{\sigma}_{nn'}$ to mean the variance term associated with the covariance function $\operatorname{cov}[\hat{f}(\boldsymbol{a_n}), \hat{f}(\boldsymbol{a_{n'}})]$ as shown in (3). The expectation $\boldsymbol{\Psi_f} = \mathbb{E}_{\mathrm{q}(\boldsymbol{a})}[\boldsymbol{K_{fu}}]$

connecting the alignments and the pseudo inputs is given by

$$\begin{aligned}
\boldsymbol{\Psi_f} &= \mathbb{E}_{\mathrm{q}(\boldsymbol{a})}[\boldsymbol{K_{fu}}], \text{ with} \\
(\boldsymbol{\Psi_f})_{ni} &= \hat{\sigma}_{ni}^2 \sqrt{\frac{(\boldsymbol{\Sigma_a})_{nn}^{-1}}{\hat{\ell}_{ni} + (\boldsymbol{\Sigma_a})_{nn}^{-1}}} \exp\left(-\frac{1}{2}\frac{(\boldsymbol{\Sigma_a})_{nn}^{-1}\hat{\ell}_{ni}}{(\boldsymbol{\Sigma_a})_{nn}^{-1} + \hat{\ell}_{ni}}\left((\boldsymbol{\mu_a})_n - \boldsymbol{Z_i}\right)^2\right),
\end{aligned} \tag{7}$$

where $\hat{\ell}_{ni}$ is the combined length scale corresponding to the same kernel as $\hat{\sigma}_{ni}$. Lastly, $\boldsymbol{\Phi_f} = \mathbb{E}_{\mathrm{q}(\boldsymbol{a})}[\boldsymbol{K_{uf}K_{fu}}]$ connects alignments and pairs of pseudo inputs with the closed form

$$\begin{aligned}
\boldsymbol{\Phi_f} &= \mathbb{E}_{\mathrm{q}(\boldsymbol{a})}[\boldsymbol{K_{uf}K_{fu}}], \text{ with} \\
(\boldsymbol{\Phi_f})_{ij} &= \sum_{n=1}^{N} \hat{\sigma}_{ni}^2\hat{\sigma}_{nj}^2 \sqrt{\frac{(\boldsymbol{\Sigma_a})_{nn}^{-1}}{\hat{\ell}_{ni} + \hat{\ell}_{nj} + (\boldsymbol{\Sigma_a})_{nn}^{-1}}} \exp\left(-\frac{1}{2}\frac{\hat{\ell}_{ni}\hat{\ell}_{nj}}{\hat{\ell}_{ni} + \hat{\ell}_{nj}}(\boldsymbol{Z_i} - \boldsymbol{Z_j})^2\right. \\
&\qquad \left. - \frac{1}{2}\frac{(\boldsymbol{\Sigma_a})_{nn}^{-1}(\hat{\ell}_{ni} + \hat{\ell}_{nj})}{(\boldsymbol{\Sigma_a})_{nn}^{-1} + \hat{\ell}_{ni} + \hat{\ell}_{nj}}\left((\boldsymbol{\mu_a})_n - \frac{\hat{\ell}_{ni}\boldsymbol{Z_i} + \hat{\ell}_{nj}\boldsymbol{Z_j}}{\hat{\ell}_{ni} + \hat{\ell}_{nj}}\right)^2\right).
\end{aligned} \tag{8}$$

The $\Psi$-statistics factorize along the data and we only need to consider the diagonal entries of $\boldsymbol{\Sigma_a}$. If all the data belong to the same output, the $\Psi$-statistics of the squared exponential kernel can be recovered as a special case. This case is used for the output-specific warpings $\boldsymbol{g}$.

### 3.3 Model Interpretation

The graphical model shown in Figure 1 illustrates that the presented model can be interpreted as a group of $D$ deep GPs all of which share one layer which is a CP. This CP acts as an interface to share information between the different GPs which are otherwise conditionally independent. This modelling-choice introduces a new quality to the model when compared to standard deep GPs with multiple output dimensions, since the latter are not able in principle to learn dependencies between the different outputs. Compared to standard multi-output GPs, the AMO-GP introduces more flexibility with respect to the shared information. CPs make strong assumptions about the relative alignments of the different outputs, that is, they assume constant time-offsets. The AMO-GP extends this by introducing a principled Bayesian treatment of general nonlinear alignments $a_d$ on which we can place informative priors derived from the problem at hand. Together with the warping layers $g_d$, our model can learn to share knowledge in an informative latent space learnt from the data.

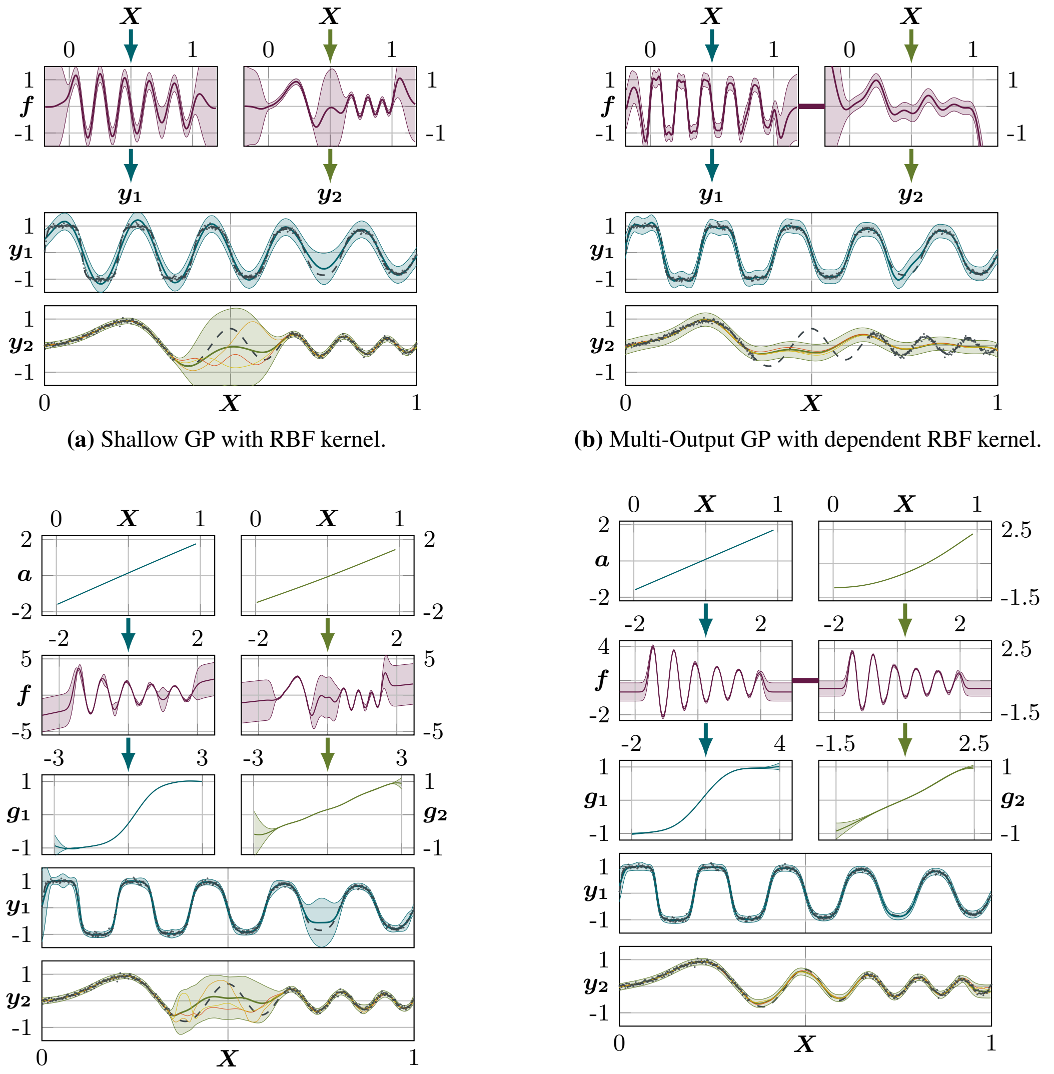
Alternatively, this model can be interpreted as a shared and warped latent variable model with a very specific prior: The indices $\boldsymbol{X}$ are part of the prior for the latent space $a_d(\boldsymbol{X})$ and specify a sense of order for the different data points $\boldsymbol{y}$ which is augmented with uncertainty by the alignment functions. Using this order, the convolution processes enforce the covariance structure for the different datapoints specified by the smoothing kernels.

In order to derive an inference scheme, we need the ability to propagate uncertainties about the correct alignments and latent shared information through subsequent layers. We adapted the approach of nested variational compression by Hensman and Lawrence [11], which is originally concerned with a single deep GP. The approximation is expanded to handle multiple GPs at once, yielding the bound in (5). The bound reflects the dependencies of the different outputs as the sharing of information between the different deep GPs is approximated through the shared inducing variables $\boldsymbol{u_{f,d}}$. Our main contribution for the inference scheme is the derivation of a closed-form solution for the $\Psi$-statistics of the convolution kernel in (6) to (8).

## 4 Experiments

In this section we show how to apply the AMO-GP to the task of finding common structure in time series observations. In this setting, we observe multiple time series $\mathcal{T}_d = (\boldsymbol{X_d}, \boldsymbol{y_d})$ and assume that there exist latent time series which determine the observations.

We will first apply the AMO-GP to an artificial data set in which we define a decomposed system of dependent time series by specifying a shared latent function generating the observations together



**(a)** Shallow GP with RBF kernel.

**(b)** Multi-Output GP with dependent RBF kernel.

**(c)** Deep GP with RBF kernels.

**(d)** AMO-GP with (dependent) RBF kernels.

**Figure 3:** A comparison of the AMO-GP with other GP models. The plots show mean predictions and a shaded area of two standard deviations. If available, the ground truth is displayed as a dashed line. Additional lines are noiseless samples drawn from the model. The shallow and deep GPs in Figures 3a and 3c model the data independently and revert back to the prior in $\boldsymbol{y_2}$. Because of the nonlinear alignment, a multi-output GP cannot model the data in Figure 3b. The AMO-GP in Figure 3d recovers the alignment and warping and shares information between the two outputs.

with relative alignments and warpings for the different time series. We will show that our model is able to recover this decomposition from the training data and compare the results to other approaches of modeling the data. Then we focus on a real world data set of a neighbouring pair of wind turbines in a wind farm, where the model is able to recover a representation of the latent prevailing wind condition and the relative timings of wind fronts at the two turbines.

### 4.1 Artificial data set

Our data set consists of two time series $\mathcal{T}_1$ and $\mathcal{T}_2$ generated by a dampened sine function. We choose the alignment of $\mathcal{T}_1$ and the warping of $\mathcal{T}_2$ to be the identity in order to prevent us from directly observing the latent function and apply a sigmoid warping to $\mathcal{T}_1$. The alignment of $\mathcal{T}_2$ is selected to

**Table 1:** Test-log-likelihoods for the models presented in Section 4.

| Experiment | Test set | GP | MO-GP | DGP | AMO-GP (Ours) |
|---|---|---|---|---|---|
| Artificial | $[0.7, 0.8] \subseteq \mathcal{T}_1$ | -0.12 | -0.053 | 0.025 | **1.54** |
| | $[0.35, 0.65] \subseteq \mathcal{T}_2$ | -0.19 | -5.66 | -0.30 | **0.72** |
| Wind | $[40, 45] \subseteq \mathcal{T}_2$ | -4.42 | -2.31 | -1.80 | **-1.43** |
| | $[65, 75] \subseteq \mathcal{T}_2$ | -7.26 | -0.73 | -1.93 | **-0.69** |

be a quadratic function. Figure 2 shows a visualization of this decomposed system of dependent time series. To obtain training data we uniformly sampled 500 points from the two time series and added Gaussian noise. We subsequently removed parts of the training sets to explore the generalization behaviour of our model, resulting in $|\mathcal{T}_1| = 450$ and $|\mathcal{T}_2| = 350$.

We use this setup to train our model using squared exponential kernels both in the conditionally independent GPs $\boldsymbol{a_d}$ and $\boldsymbol{g_d}$ and as smoothing kernels in $\boldsymbol{f}$. We can always choose one alignment and one warping to be the identity function in order to constrain the shared latent spaces $\boldsymbol{a}$ and $\boldsymbol{f}$ and provide a reference the other alignments and warpings will be relative to. Since we assume our artificial data simulates a physical system, we apply the prior knowledge that the alignment and warping processes have slower dynamics compared to the shared latent function which should capture most of the observed dynamics. To this end we applied priors to the $\boldsymbol{a_d}$ and $\boldsymbol{g_d}$ which prefer longer length scales and smaller variances compared to $\boldsymbol{f}$. Otherwise, the model could easily get stuck in local minima like choosing the upper two layers to be identity functions and model the time series independently in the $\boldsymbol{g_d}$. Additionally, our assumption of identity mean functions prevents pathological cases in which the complete model collapses to a constant function.

Figure 3d shows the AMO-GP's recovered function decomposition and joint predictions. The model successfully recovered a shared latent dampened sine function, a sigmoid warping for the first time series and an approximate quadratic alignment function for the second time series. In Figures 3a to 3c, we show the training results of a standard GP, a multi-output GP and a three-layer deep GP on the same data. For all of these models, we used RBF kernels and, in the case of the deep GP, applied priors similar to our model in order to avoid pathological cases. In Table 1 we report test log-likelihoods for the presented models, which illustrate the qualitative differences between the models. Because all models are non-parametric and converge well, repeating the experiments with different initializations leads to very similar likelihoods.

Both the standard GP and deep GP cannot learn dependencies between time series and revert back to the prior where no data is available. The deep GP has learned that two layers are enough to model the data and the resulting model is essentially a Bayesian warped GP which has identified the sigmoid warping for $\mathcal{T}_1$. Uncertainties in the deep GP are placed in the middle layer areas where no data are available for the respective time series, as sharing information between the two outputs is impossible. In contrast to the other two models, the multi-output GP can and must share information between the two time series. As discussed in Section 2 however, it is constrained to constant time-offsets and cannot model the nonlinear alignment in the data. Because of this, the model cannot recover the latent sine function and can only model one of the two outputs.

### 4.2 Pairs of wind turbines

This experiment is based on real data recorded from a pair of neighbouring wind turbines in a wind farm. The two time series $\mathcal{T}_1$ and $\mathcal{T}_2$ shown in gray in Figure 4 record the respective power generation of the two turbines over the course of one and a half hours, which was smoothed slightly using a rolling average over 60 seconds. There are 5400 data points for the first turbine (blue) and 4622 data points for the second turbine (green). We removed two intervals (drawn as dashed lines) from the second turbine's data set to inspect the behaviour of the model with missing data. This allows us to evaluate and compare the generative properties of our model in Figure 5.

The power generated by a wind turbine is mainly dependent on the speed of the wind fronts interacting with the turbine. For system identification tasks concerned with the behaviour of multiple wind turbines, associating the observations on different turbines due to the same wind fronts is an important

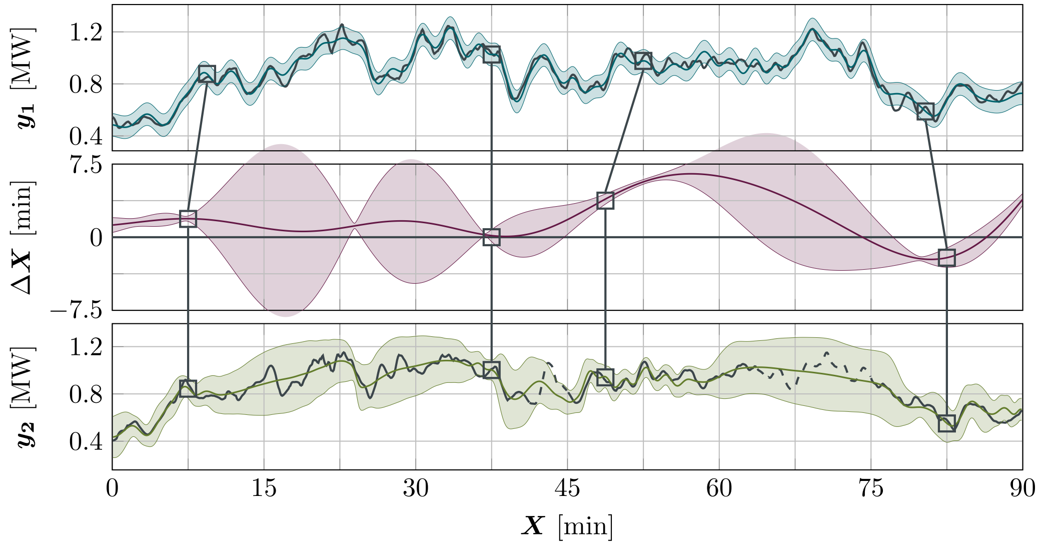
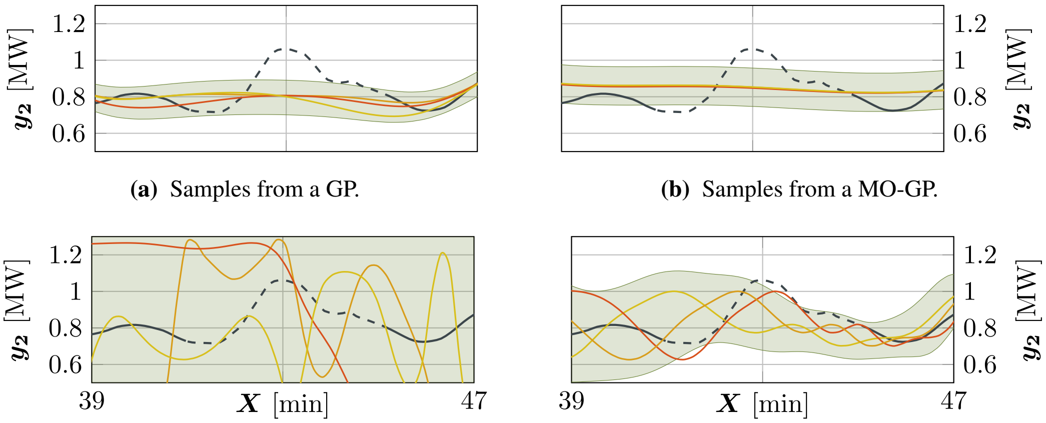


**Figure 4:** The joint posterior for two time series $y_1$ and $y_2$ of power production for a pair of wind turbines. The top and bottom plots show the two observed time series with training data and dashed missing data. The AMO-GP recovers an uncertain relative alignment of the two time series shown in the middle plot. High uncertainty about the alignment is placed in areas where multiple explanations are plausible due to the high amount of noise or missing data.



**(a)** Samples from a GP.

**(b)** Samples from a MO-GP.

**(c)** Samples from a DGP.

**(d)** Samples from the AMO-GP.

**Figure 5:** A comparison of noiseless samples drawn from a GP, a MO-GP, a DGP and the AMO-GP. The separation of uncertainties implied by the model structure of AMP-GP gives rise to an informative model. Since the uncertainty in the generative process is mainly placed in the relative alignment shown in Figure 4, all samples in Figure 5d resemble the underlying data in structure.

task. However it is usually not possible to directly measure these correspondences or wind propagation speeds between turbines, which means that there is no ground truth available. An additional problem is that the shared latent wind conditions are superimposed by turbine-specific local turbulences. Since these local effects are of comparable amplitude to short-term changes of wind speed, it is challenging to decide which parts of the signal to explain away as noise and which part to identify as the underlying shared process.

Our goal is the simultaneous learning of the uncertain alignment in time $\boldsymbol{a}$ and of the shared latent wind condition $\boldsymbol{f}$. Modelling the turbine-specific parts of the signals is not the objective, so they need to be explained by the Gaussian noise term. We use a squared exponential kernel as a prior for the alignment functions $\boldsymbol{a_d}$ and as smoothening kernels in $\boldsymbol{f}$. For the given data set we can assume the output warpings $\boldsymbol{g_d}$ to be linear functions because there is only one dimension, the power generation, which in this data set is of similar shape for both turbines. Again we encode a preference for alignments with slow dynamics with a prior on the length scales of $\boldsymbol{a_d}$. As the signal has turbine-specific autoregressive components, plausible alignments are not unique. To constrain the AMO-GP, we want it to prefer alignments close to the identity function which we chose as a prior mean function.

Figure 4 shows the joint model learned from the data in which $a_1$ is chosen to be the identity function. The possible alignments identified match the physical conditions of the wind farm. For the given turbines, time offsets of up to six minutes are plausible and for most wind conditions, the offset is expected to be close to zero. For areas where the alignment is quite certain however, the two time series are explained with comparable detail. The model is able to recover unambiguous associations well and successfully places high uncertainty on the alignment in areas where multiple explanations are plausible due to the noisy signal.

As expected, the uncertainty about the alignment also grows where data for the second time series is missing. This uncertainty is propagated through the shared function and results in higher predictive variances for the second time series. Because of the factorization in the model however, we can recover the uncertainties about the alignment and the shared latent function separately. Figure 5 compares samples drawn from our model with samples drawn from a GP, a MO-GP and a DGP. The GP reverts to their respective priors when data is missing, while the MO-GP does not handle short-term dynamics and smoothens the signal enough such that the nonlinear alignment can be approximated as constant. Samples drawn from a DGP model showcase the complexity of a DGP prior. Unconstrained composite GPs are hard to reason about and make the model very flexible in terms of representable functions. Since the model's evidence is very broad, the posterior is uninformed and inference is hard. Additionally, as discussed in Appendix B and [11], the nested variational compression bound tends to loosen with high uncertainties. AMO-GP shows richer structure: Due to the constraints imposed by the model, more robust inference leads to a more informed model. Samples show that it has learned that a maximum which is missing in the training data has to exist somewhere, but the uncertainty about the correct alignment due to the local turbulence means that different samples place the maximum at different locations in $\boldsymbol{X}$-direction.

## 5 Conclusion

We have proposed the warped and aligned multi-output Gaussian process (AMO-GP), in which MO-GPs are embedded in a hierarchy to find shared structure in latent spaces. We extended convolution processes [5] with conditionally independent Gaussian processes on both the input and output sides, giving rise to a highly structured deep GP model. This structure can be used to both regularize the model and encode expert knowledge about specific parts of the system. By applying nested variational compression [11] to inference in these models, we presented a variational lower bound which combines Bayesian treatment of all parts of the model with scalability via stochastic optimization.

We compared the model with GPs, deep GPs and multi-output GPs on an artificial data set and showed how the richer model-structure allows the AMO-GP to pick up on latent structure which the other approaches cannot model. We then applied the AMO-GP to real world data of two wind turbines and used the proposed hierarchy to model wind propagation in a wind farm and recover information about the latent non homogeneous wind field. With uncertainties decomposed along the hierarchy, our approach handles ambiguities introduced by the stochasticity of the wind in a principled manner. This indicates the AMO-GP is a good approach for these kinds of dynamical system, where multiple misaligned sensors measure the same latent effect.

## 6 Acknowledgement

The project this report is based on was supported with funds from the German Federal Ministry of Education and Research under project number 01IB15001. The sole responsibility for the reports contents lies with the authors.

## References

[1] Mauricio A. Alvarez, Lorenzo Rosasco, and Neil D. Lawrence. “Kernels for Vector-Valued Functions: a Review”. In: *arXiv:1106.6251 [cs, math, stat]* (June 2011). arXiv: 1106.6251.

[2] Mauricio A. Alvarez et al. “Efficient Multioutput Gaussian Processes through Variational Inducing Kernels.” In: *AISTATS*. Vol. 9. 2010, pp. 25–32.

[3] Mauricio Alvarez and Neil D. Lawrence. “Sparse convolved Gaussian processes for multi-output regression”. In: *Advances in neural information processing systems*. 2009, pp. 57–64.

[4] Eilyan Bitar and Pete Seiler. “Coordinated control of a wind turbine array for power maximization”. In: *American Control Conference (ACC), 2013*. IEEE, 2013, pp. 2898–2904.

[5] Phillip Boyle and Marcus R. Frean. “Dependent Gaussian Processes.” In: *NIPS*. Vol. 17. 2004, pp. 217–224.

[6] Phillip Boyle et al. *Multiple output gaussian process regression*. Tech. rep. 2005.

[7] Timothy C. Coburn. *Geostatistics for natural resources evaluation*. Taylor & Francis Group, 2000.

[8] Andreas C. Damianou and Neil D. Lawrence. “Deep Gaussian Processes”. In: *arXiv:1211.0358 [cs, math, stat]* (Nov. 2012). arXiv: 1211.0358.

[9] David Duvenaud et al. *Avoiding pathologies in very deep networks*. 2014.

[10] James Hensman, Nicolo Fusi, and Neil D. Lawrence. “Gaussian Processes for Big Data”. In: *arXiv:1309.6835 [cs, stat]* (Sept. 2013).

[11] James Hensman and Neil D. Lawrence. “Nested Variational Compression in Deep Gaussian Processes”. In: *arXiv:1412.1370 [stat]* (Dec. 2014). arXiv: 1412.1370.

[12] James Hensman, Alex Matthews, and Zoubin Ghahramani. “Scalable Variational Gaussian Process Classification”. In: *arXiv:1411.2005 [stat]* (Nov. 2014). arXiv: 1411.2005.

[13] Andre G. Journel and Ch J. Huijbregts. *Mining geostatistics*. Academic press, 1978.

[14] Miguel Lázaro-Gredilla. “Bayesian warped Gaussian processes”. In: *Advances in Neural Information Processing Systems*. 2012, pp. 1619–1627.

[15] Alexander G. de G. Matthews et al. “GPflow: A Gaussian process library using TensorFlow”. In: *Journal of Machine Learning Research* 18.40 (2017), pp. 1–6.

[16] Carl Edward Rasmussen and Christopher K I Williams. *Gaussian Processes for Machine Learning (Adaptive Computation and Machine Learning)*. The MIT Press, 2006.

[17] Hugh Salimbeni and Marc Deisenroth. “Doubly Stochastic Variational Inference for Deep Gaussian Processes”. In: *arXiv:1705.08933 [stat]* (May 2017). arXiv: 1705.08933.

[18] J. G. Schepers and S. P. Van der Pijl. “Improved modelling of wake aerodynamics and assessment of new farm control strategies”. In: *Journal of Physics: Conference Series*. Vol. 75. IOP Publishing, 2007, p. 012039.

[19] Edward Snelson, Carl Edward Rasmussen, and Zoubin Ghahramani. “Warped Gaussian Processes”. In: MIT Press, 2004, pp. 337–344.

[20] Jasper Snoek et al. “Input Warping for Bayesian Optimization of Non-stationary Functions”. In: *arXiv:1402.0929 [cs, stat]* (Feb. 2014). arXiv: 1402.0929.

[21] Maryam Soleimanzadeh and Rafael Wisniewski. “Controller design for a wind farm, considering both power and load aspects”. In: *Mechatronics* 21.4 (2011), pp. 720–727.

[22] Michalis K. Titsias. “Variational Learning of Inducing Variables in Sparse Gaussian Processes.” In: *AISTATS*. Vol. 5. 2009, pp. 567–574.

[23] Michalis K. Titsias and Neil D. Lawrence. “Bayesian Gaussian process latent variable model”. In: *International Conference on Artificial Intelligence and Statistics*. 2010, pp. 844–851.

[24] Feng Zhou and Fernando De la Torre. “Generalized time warping for multi-modal alignment of human motion”. In: *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*. IEEE, 2012, pp. 1282–1289.

# A Detailed Variational Approximation

In this section, we repeat the derivation of the variational approximation in more detail.

Since exact inference in this model is intractable, we discuss a variational approximation to the model's true marginal likelihood and posterior in this section. Analogously to $\boldsymbol{y}$, we denote the random vectors which contain the function values of the respective functions and outputs as $\boldsymbol{a}$ and $\boldsymbol{f}$. The joint probability distribution of the data can then be written as

$$\begin{aligned}
\mathrm{p}(\boldsymbol{y}, \boldsymbol{f}, \boldsymbol{a} \mid \boldsymbol{X}) &= \mathrm{p}(\boldsymbol{f} \mid \boldsymbol{a}) \prod_{d=1}^{D} \mathrm{p}(\boldsymbol{y_d} \mid \boldsymbol{f_d})\, \mathrm{p}(\boldsymbol{a_d} \mid \boldsymbol{X}), \\
\boldsymbol{a_d} \mid \boldsymbol{X} &\sim \mathcal{N}(\boldsymbol{X}, \boldsymbol{K_{a,d}} + \sigma_{a,d}^2 \mathbf{I}), \\
\boldsymbol{f} \mid \boldsymbol{a} &\sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{K_f} + \sigma_f^2 \mathbf{I}), \\
\boldsymbol{y_d} \mid \boldsymbol{f_d} &\sim \mathcal{N}(\boldsymbol{f_d}, \boldsymbol{K_{g,d}} + \sigma_{y,d}^2 \mathbf{I}).
\end{aligned} \tag{9}$$

Here, we use $\boldsymbol{K}$ to refer to the Gram matrix corresponding to the kernel of the respective GP. All but the CPs factorize over both the different levels of the model as well as the different outputs.

To approximate a single deep GP, Hensman and Lawrence [11] proposed nested variational compression in which every GP in the hierarchy is handled independently. While this forces a variational approximation of all intermediate outputs of the stacked processes, it has the appealing property that it allows optimization via stochastic gradient descent [10] and the variational approximation can after training be used independently of the original training data.

## A.1 Augmented Model

Nested variational compression focuses on augmenting a full GP model by introducing sets of *inducing variables* $\boldsymbol{u}$ with their *inducing inputs* $\boldsymbol{Z}$. Those variables are assumed to be latent observations of the same functions and are thus jointly Gaussian with the observed data.

It can be written using its marginals [22] as

$$\begin{aligned}
\mathrm{p}(\hat{\boldsymbol{a}}, \boldsymbol{u}) &= \mathcal{N}(\hat{\boldsymbol{a}} \mid \boldsymbol{\mu_a}, \boldsymbol{\Sigma_a})\, \mathcal{N}(\boldsymbol{u} \mid \boldsymbol{Z}, \boldsymbol{K_{uu}}), \text{ with} \\
\boldsymbol{\mu_a} &= \boldsymbol{X} + \boldsymbol{K_{au}} \boldsymbol{K_{uu}}^{-1} (\boldsymbol{u} - \boldsymbol{Z}), \\
\boldsymbol{\Sigma_a} &= \boldsymbol{K_{aa}} - \boldsymbol{K_{au}} \boldsymbol{K_{uu}}^{-1} \boldsymbol{K_{ua}},
\end{aligned} \tag{10}$$

where, after dropping some indices and explicit conditioning on $\boldsymbol{X}$ and $\boldsymbol{Z}$ for clarity, $\hat{\boldsymbol{a}}$ denotes the function values $a_d(\boldsymbol{X})$ without noise and we write the Gram matrices as $\boldsymbol{K_{au}} = k_{a,d}(\boldsymbol{X}, \boldsymbol{Z})$.

While the original model in (9) can be recovered exactly by marginalizing the inducing variables, considering a specific variational approximation of the joint $\mathrm{p}(\hat{\boldsymbol{a}}, \boldsymbol{u})$ gives rise to the desired lower bound in the next subsection. A central assumption of this approximation [22] is that given enough inducing variables at the correct location, they are a sufficient statistic for $\hat{\boldsymbol{a}}$, implying conditional independence of the entries of $\hat{\boldsymbol{a}}$ given $\boldsymbol{X}$ and $\boldsymbol{u}$. We introduce such inducing variables for every GP in the model, yielding the set $\{\boldsymbol{u_{a,d}}, \boldsymbol{u_{f,d}}, \boldsymbol{u_{g,d}}\}_{d=1}^{D}$ of inducing variables. Note that for the CP $f$, we introduce one set of inducing variables $\boldsymbol{u_{f,d}}$ per output $f_d$. These inducing variables play a crucial role in sharing information between the different outputs.

## A.2 Variational Lower Bound

To derive the desired variational lower bound for the log marginal likelihood of the complete model, multiple steps are necessary. First, we will consider the innermost GPs $a_d$ describing the alignment functions. We derive the Scalable Variational GP (SVGP), a lower bound for this model part which can be calculated efficiently and can be used for stochastic optimization, first introduced by Hensman, Fusi, and Lawrence [10]. In order to apply this bound recursively, we will both show how to propagate the uncertainty through the subsequent layers $f_d$ and $g_d$ and how to avoid the inter-layer cross-dependencies using another variational approximation as presented by Hensman and Lawrence [11]. While Hensman and Lawrence considered standard deep GP models, we will show how to apply their results to CPs.

**The First Layer** Since the inputs $\boldsymbol{X}$ are fully known, we do not need to propagate uncertainty through the GPs $a_d$. Instead, the uncertainty about the $\boldsymbol{a_d}$ comes from the uncertainty about the correct functions $a_d$ and is introduced by the processes themselves. To derive a lower bound on the marginal log likelihood of $\boldsymbol{a_d}$, we assume a variational distribution $\mathrm{q}(\boldsymbol{u_{a,d}}) \sim \mathcal{N}(\boldsymbol{m_{a,d}}, \boldsymbol{S_{a,d}})$ approximating $\mathrm{p}(\boldsymbol{u_{a,d}})$ and additionally assume that $\mathrm{q}(\boldsymbol{\hat{a}_d}, \boldsymbol{u_{a,d}}) = \mathrm{p}(\boldsymbol{\hat{a}_d} \mid \boldsymbol{u_{a,d}})\,\mathrm{q}(\boldsymbol{u_{a,d}})$. After dropping the indices again, using Jensen's inequality we get

$$\begin{aligned}
\log \mathrm{p}(\boldsymbol{a} \mid \boldsymbol{X}) &= \log \int \mathrm{p}(\boldsymbol{a} \mid \boldsymbol{u})\,\mathrm{p}(\boldsymbol{u})\,\mathrm{d}\boldsymbol{u} \\
&= \log \int \mathrm{q}(\boldsymbol{u}) \frac{\mathrm{p}(\boldsymbol{a} \mid \boldsymbol{u})\,\mathrm{p}(\boldsymbol{u})}{\mathrm{q}(\boldsymbol{u})}\,\mathrm{d}\boldsymbol{u} \\
&\geq \int \mathrm{q}(\boldsymbol{u}) \log \frac{\mathrm{p}(\boldsymbol{a} \mid \boldsymbol{u})\,\mathrm{p}(\boldsymbol{u})}{\mathrm{q}(\boldsymbol{u})}\,\mathrm{d}\boldsymbol{u} \\
&= \int \log \mathrm{p}(\boldsymbol{a} \mid \boldsymbol{u})\,\mathrm{q}(\boldsymbol{u})\,\mathrm{d}\boldsymbol{u} - \int \mathrm{q}(\boldsymbol{u}) \log \frac{\mathrm{q}(\boldsymbol{u})}{\mathrm{p}(\boldsymbol{u})}\,\mathrm{d}\boldsymbol{u} \\
&= \mathbb{E}_{\mathrm{q}(\boldsymbol{u})}[\log \mathrm{p}(\boldsymbol{a} \mid \boldsymbol{u})] - \mathrm{KL}(\mathrm{q}(\boldsymbol{u}) \parallel \mathrm{p}(\boldsymbol{u})),
\end{aligned} \tag{11}$$

where $\mathbb{E}_{\mathrm{q}(\boldsymbol{u})}[\,\cdot\,]$ denotes the expected value with respect to the distribution $\mathrm{q}(\boldsymbol{u})$ and $\mathrm{KL}(\,\cdot \parallel \cdot\,)$ denotes the KL divergence, which can be evaluated analytically.

To bound the required expectation, we use Jensen's inequality again together with (10) which gives

$$\begin{aligned}
\log \mathrm{p}(\boldsymbol{a} \mid \boldsymbol{u}) &= \log \int \mathrm{p}(\boldsymbol{a} \mid \boldsymbol{\hat{a}})\,\mathrm{p}(\boldsymbol{\hat{a}} \mid \boldsymbol{u})\,\mathrm{d}\boldsymbol{\hat{a}} \\
&= \log \int \mathcal{N}(\boldsymbol{a} \mid \boldsymbol{\hat{a}}, \sigma_a^2 \mathbf{I})\,\mathcal{N}(\boldsymbol{\hat{a}} \mid \boldsymbol{\mu_a}, \boldsymbol{\Sigma_a})\,\mathrm{d}\boldsymbol{\hat{a}} \\
&\geq \int \log \mathcal{N}(\boldsymbol{a} \mid \boldsymbol{\hat{a}}, \sigma_a^2 \mathbf{I})\,\mathcal{N}(\boldsymbol{\hat{a}} \mid \boldsymbol{\mu_a}, \boldsymbol{\Sigma_a})\,\mathrm{d}\boldsymbol{\hat{a}} \\
&= \log \mathcal{N}(\boldsymbol{a} \mid \boldsymbol{\mu_a}, \sigma_a^2 \mathbf{I}) - \frac{1}{2\sigma_a^2}\,\mathrm{tr}(\boldsymbol{\Sigma_a}).
\end{aligned} \tag{12}$$

We apply this bound to the expectation to get

$$\mathbb{E}_{\mathrm{q}(\boldsymbol{u})}[\log \mathrm{p}(\boldsymbol{a} \mid \boldsymbol{u})] \geq \mathbb{E}_{\mathrm{q}(\boldsymbol{u})}[\log \mathcal{N}(\boldsymbol{a} \mid \boldsymbol{\mu_a}, \sigma_a^2 \mathbf{I})] - \frac{1}{2\sigma_a^2}\,\mathrm{tr}(\boldsymbol{\Sigma_a}), \text{ with} \tag{13}$$

$$\begin{aligned}
\mathbb{E}_{\mathrm{q}(\boldsymbol{u})}[\log \mathcal{N}(\boldsymbol{a} \mid \boldsymbol{\mu_a}, \sigma_a^2 \mathbf{I})] &= \log \mathcal{N}(\boldsymbol{a} \mid \boldsymbol{K_{au}} \boldsymbol{K_{uu}^{-1}} \boldsymbol{m}, \sigma_a^2 \mathbf{I}) \\
&\quad + \frac{1}{2\sigma_a^2}\,\mathrm{tr}\big(\boldsymbol{K_{au}} \boldsymbol{K_{uu}^{-1}} \boldsymbol{S} \boldsymbol{K_{uu}^{-1}} \boldsymbol{K_{ua}}\big).
\end{aligned} \tag{14}$$

Resubstituting this result into (11) yields the final bound

$$\begin{aligned}
\log \mathrm{p}(\boldsymbol{a} \mid \boldsymbol{X}) \geq{}& \log \mathcal{N}(\boldsymbol{a} \mid \boldsymbol{K_{au}} \boldsymbol{K_{uu}^{-1}} \boldsymbol{m}, \sigma_a^2 \mathbf{I}) - \mathrm{KL}(\mathrm{q}(\boldsymbol{u}) \parallel \mathrm{p}(\boldsymbol{u})) \\
&- \frac{1}{2\sigma_a^2}\,\mathrm{tr}(\boldsymbol{\Sigma_a}) - \frac{1}{2\sigma_a^2}\,\mathrm{tr}\big(\boldsymbol{K_{au}} \boldsymbol{K_{uu}^{-1}} \boldsymbol{S} \boldsymbol{K_{uu}^{-1}} \boldsymbol{K_{ua}}\big).
\end{aligned} \tag{15}$$

This bound, which depends on the hyper parameters of the kernel and likelihood $\{\boldsymbol{\theta}, \sigma_a\}$ and the variational parameters $\{\boldsymbol{Z}, \boldsymbol{m}, \boldsymbol{S}\}$, can be calculated in $\mathcal{O}(NM^2)$ time. It factorizes along the data points which enables stochastic optimization.

In order to obtain a bound on the full model, we apply the same techniques to the other processes. Since the alignment processes $a_d$ are assumed to be independent, we have $\log \mathrm{p}(\boldsymbol{a_1}, \dots, \boldsymbol{a_D} \mid \boldsymbol{X}) = \sum_{d=1}^{D} \log \mathrm{p}(\boldsymbol{a_d} \mid \boldsymbol{X})$, where every term can be approximated using the bound in (15). However, for all subsequent layers, the bound is not directly applicable, since the inputs are no longer known but instead are given by the outputs of the previous process. It is therefore necessary to propagate their uncertainty and also handle the interdependencies between the layers introduced by the latent function values $\boldsymbol{a}$, $\boldsymbol{f}$ and $\boldsymbol{g}$.

**The Second and Third Layer** Our next goal is to derive a bound on the outputs of the second layer

$$\log \mathrm{p}(\boldsymbol{f} \mid \boldsymbol{u_f}) = \log \int \mathrm{p}(\boldsymbol{f}, \boldsymbol{a}, \boldsymbol{u_a} \mid \boldsymbol{u_f}) \,\mathrm{d}\boldsymbol{a} \,\mathrm{d}\boldsymbol{u_a}, \tag{16}$$

that is, an expression in which the uncertainty about the different $\boldsymbol{a_d}$ and the cross-layer dependencies on the $\boldsymbol{u_{a,d}}$ are both marginalized. While on the first layer, the different $\boldsymbol{a_d}$ are conditionally independent, the second layer explicitly models the cross-covariances between the different outputs via convolutions over the shared latent processes $w_r$. We will therefore need to handle all of the different $\boldsymbol{f_d}$, together denoted as $\boldsymbol{f}$, at the same time.

We start by considering the relevant terms from (9) and apply (12) to marginalize $\boldsymbol{a}$ in

$$\begin{aligned}
\log \mathrm{p}(\boldsymbol{f} \mid \boldsymbol{u_f}, \boldsymbol{u_a}) &= \log \int \mathrm{p}(\boldsymbol{f}, \boldsymbol{a} \mid \boldsymbol{u_f}, \boldsymbol{u_a}) \,\mathrm{d}\boldsymbol{a} \\
&\geq \log \int \tilde{\mathrm{p}}(\boldsymbol{f} \mid \boldsymbol{u_f}, \boldsymbol{a}) \tilde{\mathrm{p}}(\boldsymbol{a} \mid \boldsymbol{u_a}) \cdot \exp\left( -\frac{1}{2\sigma_a^2} \operatorname{tr}(\boldsymbol{\Sigma_a}) - \frac{1}{2\sigma_f^2} \operatorname{tr}(\boldsymbol{\Sigma_f}) \right) \mathrm{d}\boldsymbol{a} \\
&\geq \mathbb{E}_{\tilde{\mathrm{p}}(\boldsymbol{a} \mid \boldsymbol{u_a})}[\log \tilde{\mathrm{p}}(\boldsymbol{f} \mid \boldsymbol{u_f}, \boldsymbol{a})] - \mathbb{E}_{\tilde{\mathrm{p}}(\boldsymbol{a} \mid \boldsymbol{u_a})}\left[ \frac{1}{2\sigma_f^2} \operatorname{tr}(\boldsymbol{\Sigma_f}) \right] - \frac{1}{2\sigma_a^2} \operatorname{tr}(\boldsymbol{\Sigma_a}),
\end{aligned} \tag{17}$$

where we write $\tilde{\mathrm{p}}(\boldsymbol{a} \mid \boldsymbol{u_a}) = \mathcal{N}\big(\boldsymbol{a} \,\big|\, \boldsymbol{\mu_a}, \sigma_a^2 \mathbf{I}\big)$ to incorporate the Gaussian noise in the latent space. Due to our assumption that $\boldsymbol{u_a}$ is a sufficient statistic for $\boldsymbol{a}$ we choose

$$\begin{aligned}
\mathrm{q}(\boldsymbol{a} \mid \boldsymbol{u_a}) &= \tilde{\mathrm{p}}(\boldsymbol{a} \mid \boldsymbol{u_a}), \text{ and} \\
\mathrm{q}(\boldsymbol{a}) &= \int \tilde{\mathrm{p}}(\boldsymbol{a} \mid \boldsymbol{u_a}) \,\mathrm{q}(\boldsymbol{u_a}) \,\mathrm{d}\boldsymbol{u_a},
\end{aligned} \tag{18}$$

and use another variational approximation to marginalize $\boldsymbol{u_a}$. This yields

$$\begin{aligned}
\log \mathrm{p}(\boldsymbol{f} \mid \boldsymbol{u_f}) &= \log \int \mathrm{p}(\boldsymbol{f}, \boldsymbol{u_a} \mid \boldsymbol{u_f}) \,\mathrm{d}\boldsymbol{u_a} \\
&= \log \int \mathrm{p}(\boldsymbol{f} \mid \boldsymbol{u_f}, \boldsymbol{u_a}) \,\mathrm{p}(\boldsymbol{u_a}) \,\mathrm{d}\boldsymbol{u_a} \\
&\geq \int \mathrm{q}(\boldsymbol{u_a}) \log \frac{\mathrm{p}(\boldsymbol{f} \mid \boldsymbol{u_f}, \boldsymbol{u_a}) \,\mathrm{p}(\boldsymbol{u_a})}{\mathrm{q}(\boldsymbol{u_a})} \,\mathrm{d}\boldsymbol{u_a} \\
&= \mathbb{E}_{\mathrm{q}(\boldsymbol{u_a})}[\log \mathrm{p}(\boldsymbol{f} \mid \boldsymbol{u_a}, \boldsymbol{u_f})] - \mathrm{KL}(\mathrm{q}(\boldsymbol{u_a}) \,\|\, \mathrm{p}(\boldsymbol{u_a})) \\
&\geq \mathbb{E}_{\mathrm{q}(\boldsymbol{u_a})}\big[\mathbb{E}_{\tilde{\mathrm{p}}(\boldsymbol{a} \mid \boldsymbol{u_a})}[\log \tilde{\mathrm{p}}(\boldsymbol{f} \mid \boldsymbol{u_f}, \boldsymbol{a})]\big] - \mathrm{KL}(\mathrm{q}(\boldsymbol{u_a}) \,\|\, \mathrm{p}(\boldsymbol{u_a})) \\
&\quad - \frac{1}{2\sigma_a^2} \operatorname{tr}(\boldsymbol{\Sigma_a}) - \mathbb{E}_{\mathrm{q}(\boldsymbol{u_a})}\left[ \mathbb{E}_{\tilde{\mathrm{p}}(\boldsymbol{a} \mid \boldsymbol{u_a})}\left[ \frac{1}{2\sigma_f^2} \operatorname{tr}(\boldsymbol{\Sigma_f}) \right] \right] \\
&\geq \mathbb{E}_{\mathrm{q}(\boldsymbol{a})}[\log \tilde{\mathrm{p}}(\boldsymbol{f} \mid \boldsymbol{u_f}, \boldsymbol{a})], - \mathrm{KL}(\mathrm{q}(\boldsymbol{u_a}) \,\|\, \mathrm{p}(\boldsymbol{u_a})) \\
&\quad - \frac{1}{2\sigma_a^2} \operatorname{tr}(\boldsymbol{\Sigma_a}) - \frac{1}{2\sigma_f^2} \mathbb{E}_{\mathrm{q}(\boldsymbol{a})}[\operatorname{tr}(\boldsymbol{\Sigma_f})],
\end{aligned} \tag{19}$$

where we apply Fubini's theorem to exchange the order of integration in the expected values. The expectations with respect to $\mathrm{q}(\boldsymbol{a})$ involve expectations of kernel matrices, also called $\Psi$-statistics, in the same way as in [8] and are given by

$$\begin{aligned}
\psi_f &= \mathbb{E}_{\mathrm{q}(\boldsymbol{a})}[\operatorname{tr}(\boldsymbol{K_{ff}})], \\
\boldsymbol{\Psi_f} &= \mathbb{E}_{\mathrm{q}(\boldsymbol{a})}[\boldsymbol{K_{fu}}], \\
\boldsymbol{\Phi_f} &= \mathbb{E}_{\mathrm{q}(\boldsymbol{a})}[\boldsymbol{K_{uf}} \boldsymbol{K_{fu}}].
\end{aligned} \tag{20}$$

These $\Psi$-statistics can be computed analytically for multiple kernels, including the squared exponential kernel. In Appendix A.3 we show closed-form solutions for these $\Psi$-statistics for the implicit kernel defined in the CP layer. To obtain the final formulation of the desired bound for $\log \mathrm{p}(\boldsymbol{f} \mid \boldsymbol{u_f})$

we substitute (20) into (19) and get the analytically tractable bound

$$
\begin{aligned}
\log \mathrm{p}(\boldsymbol{f} \,|\, \boldsymbol{u_f}) \geq{} & \log \mathcal{N}\Big(\boldsymbol{f} \,\Big|\, \boldsymbol{\Psi_f} \boldsymbol{K}_{\boldsymbol{u_f u_f}}^{-1} \boldsymbol{m_f}, \sigma_f^2 \mathbf{I}\Big) - \mathrm{KL}(\mathrm{q}(\boldsymbol{u_a}) \,\|\, \mathrm{p}(\boldsymbol{u_a})) - \frac{1}{2\sigma_a^2} \operatorname{tr}(\boldsymbol{\Sigma_a}) \\
& - \frac{1}{2\sigma_f^2} \left(\psi_f - \operatorname{tr}\Big(\boldsymbol{\Psi_f} \boldsymbol{K}_{\boldsymbol{u_f u_f}}^{-1}\Big)\right) \\
& - \frac{1}{2\sigma_f^2} \operatorname{tr}\left(\left(\boldsymbol{\Phi_f} - \boldsymbol{\Psi}_{\boldsymbol{f}}^{\mathsf{T}} \boldsymbol{\Psi_f}\right) \boldsymbol{K}_{\boldsymbol{u_f u_f}}^{-1} \left(\boldsymbol{m_f} \boldsymbol{m}_{\boldsymbol{f}}^{\mathsf{T}} + \boldsymbol{S_f}\right) \boldsymbol{K}_{\boldsymbol{u_f u_f}}^{-1}\right)
\end{aligned}
\tag{21}
$$

The uncertainties in the first layer have been propagated variationally to the second layer. Besides the regularization terms, $\boldsymbol{f} \mid \boldsymbol{u_f}$ is a Gaussian distribution. Because of their cross dependencies, the different outputs $\boldsymbol{f_d}$ are considered in a common bound and do not factorize along dimensions. The third layer warpings $\boldsymbol{g_d}$ however are conditionally independent given $\boldsymbol{f}$ and can therefore be considered separately. In order to derive a bound for $\log \mathrm{p}(\boldsymbol{y} \,|\, \boldsymbol{u_g})$ we apply the same steps as described above, resulting in the final bound, which factorizes along the data, allowing for stochastic optimization methods:

$$
\begin{aligned}
\log \mathrm{p}(\boldsymbol{y} \,|\, \boldsymbol{X}) \geq{} & \sum_{d=1}^{D} \log \mathcal{N}\Big(\boldsymbol{y_d} \,\Big|\, \boldsymbol{\Psi_{g,d}} \boldsymbol{K}_{\boldsymbol{u_{g,d} u_{g,d}}}^{-1} \boldsymbol{m_{g,d}}, \sigma_{y,d}^2 \mathbf{I}\Big) - \sum_{d=1}^{D} \frac{1}{2\sigma_{a,d}^2} \operatorname{tr}(\boldsymbol{\Sigma_{a,d}}) \\
& - \frac{1}{2\sigma_f^2} \left(\psi_f - \operatorname{tr}\Big(\boldsymbol{\Phi_f} \boldsymbol{K}_{\boldsymbol{u_f u_f}}^{-1}\Big)\right) - \sum_{d=1}^{D} \frac{1}{2\sigma_{y,d}^2} \left(\psi_{g,d} - \operatorname{tr}\Big(\boldsymbol{\Phi_{g,d}} \boldsymbol{K}_{\boldsymbol{u_{g,d} u_{g,d}}}^{-1}\Big)\right) \\
& - \sum_{d=1}^{D} \mathrm{KL}(\mathrm{q}(\boldsymbol{u_{a,d}}) \,\|\, \mathrm{p}(\boldsymbol{u_{a,d}})) - \mathrm{KL}(\mathrm{q}(\boldsymbol{u_f}) \,\|\, \mathrm{p}(\boldsymbol{u_f})) - \sum_{d=1}^{D} \mathrm{KL}(\mathrm{q}(\boldsymbol{u_{y,d}}) \,\|\, \mathrm{p}(\boldsymbol{u_{y,d}})) \\
& - \frac{1}{2\sigma_f^2} \operatorname{tr}\left(\left(\boldsymbol{\Phi_f} - \boldsymbol{\Psi}_{\boldsymbol{f}}^{\mathsf{T}} \boldsymbol{\Psi_f}\right) \boldsymbol{K}_{\boldsymbol{u_f u_f}}^{-1} \left(\boldsymbol{m_f} \boldsymbol{m}_{\boldsymbol{f}}^{\mathsf{T}} + \boldsymbol{S_f}\right) \boldsymbol{K}_{\boldsymbol{u_f u_f}}^{-1}\right) \\
& - \sum_{d=1}^{D} \frac{1}{2\sigma_{y,d}^2} \operatorname{tr}\left(\left(\boldsymbol{\Phi_{g,d}} - \boldsymbol{\Psi}_{\boldsymbol{g,d}}^{\mathsf{T}} \boldsymbol{\Psi_{g,d}}\right) \boldsymbol{K}_{\boldsymbol{u_{g,d} u_{g,d}}}^{-1} \left(\boldsymbol{m_{g,d}} \boldsymbol{m}_{\boldsymbol{g,d}}^{\mathsf{T}} + \boldsymbol{S_{g,d}}\right) \boldsymbol{K}_{\boldsymbol{u_{g,d} u_{g,d}}}^{-1}\right)
\end{aligned}
\tag{22}
$$

### A.3 Convolution Kernel Expectations

In Section 2 we assumed the latent processes $w_r$ to be white noise processes and the smoothing kernel functions $T_{d,r}$ to be squared exponential kernels, leading to an explicit closed form formulation for the covariance between outputs shown in (3). In this section, we derive the $\Psi$-statistics for this generalized squared exponential kernel needed to evaluate (22).

The uncertainty about the first layer is captured by the variational distribution of the latent alignments $\boldsymbol{a}$ given by $\mathrm{q}(\boldsymbol{a}) \sim \mathcal{N}(\boldsymbol{\mu_a}, \boldsymbol{\Sigma_a})$, with $\boldsymbol{a} = (\boldsymbol{a_1}, \dots, \boldsymbol{a_d})$. Every aligned point in $\boldsymbol{a}$ corresponds to one output of $\boldsymbol{f}$ and ultimately to one of the $\boldsymbol{y_d}$. Since the closed form of the multi output kernel depends on the choice of outputs, we will use the notation $\hat{f}(\boldsymbol{a_n})$ to denote $f_d(\boldsymbol{a_n})$ such that $\boldsymbol{a_n}$ is associated with output $d$.

For notational simplicity, we only consider the case of one single latent process $w_r$. Since the latent processes are independent, the results can easily be generalized to multiple processes. Then, $\psi_f$ is given by

$$
\begin{aligned}
\psi_f &= \mathbb{E}_{\mathrm{q}(\boldsymbol{a})}[\operatorname{tr}(\boldsymbol{K_{ff}})] \\
&= \sum_{n=1}^{N} \mathbb{E}_{\mathrm{q}(\boldsymbol{a_n})}\Big[\operatorname{cov}\Big[\hat{f}(\boldsymbol{a_n}), \hat{f}(\boldsymbol{a_n})\Big]\Big] \\
&= \sum_{n=1}^{N} \int \operatorname{cov}\Big[\hat{f}(\boldsymbol{a_n}), \hat{f}(\boldsymbol{a_n})\Big] \,\mathrm{q}(\boldsymbol{a_n}) \,\mathrm{d}\boldsymbol{a_n} \\
&= \sum_{n=1}^{N} \hat{\sigma}_{nn}^2.
\end{aligned}
\tag{23}
$$

Similar to the notation $\hat{f}(\cdot)$, we use the notation $\hat{\sigma}_{nn'}$ to mean the variance term associated with the covariance function $\operatorname{cov}[\hat{f}(\boldsymbol{a_n}), \hat{f}(\boldsymbol{a_{n'}})]$. The expectation $\boldsymbol{\Psi_f} = \mathbb{E}_{\mathrm{q}(\boldsymbol{a})}[\boldsymbol{K_{fu}}]$ connecting the alignments and the pseudo inputs is given by

$$\begin{aligned}
\boldsymbol{\Psi_f} &= \mathbb{E}_{\mathrm{q}(\boldsymbol{a})}[\boldsymbol{K_{fu}}], \text{ with} \\
(\boldsymbol{\Psi_f})_{ni} &= \int \operatorname{cov}\Big[\hat{f}(\boldsymbol{a_n}), \hat{f}(\boldsymbol{Z_i})\Big] \, \mathrm{q}(\boldsymbol{a_n}) \, \mathrm{d}\boldsymbol{a_n} \\
&= \hat{\sigma}_{ni}^2 \sqrt{\frac{(\boldsymbol{\Sigma_a})_{nn}^{-1}}{\hat{\ell}_{ni} + (\boldsymbol{\Sigma_a})_{nn}^{-1}}} \cdot \exp\left(-\frac{1}{2}\frac{(\boldsymbol{\Sigma_a})_{nn}^{-1}\hat{\ell}_{ni}}{(\boldsymbol{\Sigma_a})_{nn}^{-1} + \hat{\ell}_{ni}}\left((\boldsymbol{\mu_a})_n - \boldsymbol{Z_i}\right)^2\right)
\end{aligned} \tag{24}$$

where $\hat{\ell}_{ni}$ is the combined length scale corresponding to the same kernel as $\hat{\sigma}_{ni}$. Lastly, $\boldsymbol{\Phi_f} = \mathbb{E}_{\mathrm{q}(\boldsymbol{a})}[\boldsymbol{K_{uf}}\boldsymbol{K_{fu}}]$ connects alignments and pairs of pseudo inputs with the closed form

$$\begin{aligned}
\boldsymbol{\Phi_f} &= \mathbb{E}_{\mathrm{q}(\boldsymbol{a})}[\boldsymbol{K_{uf}}\boldsymbol{K_{fu}}], \text{ with} \\
(\boldsymbol{\Phi_f})_{ij} &= \sum_{n=1}^{N} \int \operatorname{cov}\Big[\hat{f}(\boldsymbol{a_n}), \hat{f}(\boldsymbol{Z_i})\Big] \cdot \operatorname{cov}\Big[\hat{f}(\boldsymbol{a_n}), \hat{f}(\boldsymbol{Z_j})\Big] \, \mathrm{q}(\boldsymbol{a_n}) \, \mathrm{d}\boldsymbol{a_n} \\
&= \sum_{n=1}^{N} \hat{\sigma}_{ni}^2 \hat{\sigma}_{nj}^2 \sqrt{\frac{(\boldsymbol{\Sigma_a})_{nn}^{-1}}{\hat{\ell}_{ni} + \hat{\ell}_{nj} + (\boldsymbol{\Sigma_a})_{nn}^{-1}}} \cdot \exp\left(-\frac{1}{2}\frac{\hat{\ell}_{ni}\hat{\ell}_{nj}}{\hat{\ell}_{ni} + \hat{\ell}_{nj}}(\boldsymbol{Z_i} - \boldsymbol{Z_j})^2\right. \\
&\qquad \left. - \frac{1}{2}\frac{(\boldsymbol{\Sigma_a})_{nn}^{-1}(\hat{\ell}_{ni} + \hat{\ell}_{nj})}{(\boldsymbol{\Sigma_a})_{nn}^{-1} + \hat{\ell}_{ni} + \hat{\ell}_{nj}} \cdot \left((\boldsymbol{\mu_a})_n - \frac{\hat{\ell}_{ni}\boldsymbol{Z_i} + \hat{\ell}_{nj}\boldsymbol{Z_j}}{\hat{\ell}_{ni} + \hat{\ell}_{nj}}\right)^2\right).
\end{aligned} \tag{25}$$

Note that the $\Psi$-statistics factorize along the data and we only need to consider the diagonal entries of $\boldsymbol{\Sigma_a}$. If all the data belong to the same output, the $\Psi$-statistics of the standard squared exponential kernel can be recovered as a special case. It is used to propagate the uncertainties through the output-specific warpings $\boldsymbol{g}$.

### A.4 Approximative Predictions

Using the variational lower bound in (5), our model can be fitted to data, resulting in appropriate choices of the kernel hyper parameters and variational parameters. Now assume we want to predict approximate function values $\boldsymbol{g_{d,\star}}$ for previously unseen points $\boldsymbol{X_{d,\star}}$ associated with output $d$, which are given by $\boldsymbol{g_{d,\star}} = g_d(f_d(a_d(\boldsymbol{X_{d,\star}})))$.

Because of the conditional independence assumptions in the model, other outputs $d' \neq d$ only have to be considered in the shared layer $\boldsymbol{f}$. In this shared layer, the belief about the different outputs and the shared information and is captured by the variational distribution $\mathrm{q}(\boldsymbol{u_f})$. Given $\mathrm{q}(\boldsymbol{u_f})$, the different outputs are conditionally independent of one another and thus, predictions for a single dimension in our model are equivalent to predictions in a single deep GP with nested variational compression as presented by Hensman and Lawrence [11].

## B Joint models for wind experiment

In the following, we show plots with joint predictions for the models discussed in Section 4.2. Similar to Section 4.1, we trained a standard GP in Figure 6, a multi-output GP in Figure 7, a deep GP in Figure 8 and our model in Figure 9. All models were trained until convergence and multiple runs result in very similar models. For all models we used RBF kernels or dependent RBF kernels where applicable.

Each plot shows the data in gray and two mean predictions and uncertainty bands. The first violet uncertainty band is the result of the variational approximation of the respective model. The second green or blue posterior is obtained via sampling. For both the GP and MO-GP, we used the SVGP approximation [12] and since the models are shallow, the approximation is almost exact.

Figure 8 showcases the difficulty of training a deep GP model and the shortcomings of the nested variational compression. The violet variational approximation is used for training and approximates the data comparatively well. As discussed above, the deep GP cannot share information, so the test sets cannot be predicted. However, as discussed in more detail in [12], the approximation tends to underestimate uncertainties when propagating them through the different layers and because of this, uncertainties obtained via sampling tend to vary considerably more. Because during model selection sample performance does not matter, the true posterior can be (and in this case is) considerably different.

Our approach in principle has the same problem as the deep GP. However, because of the strong interpretability of the different parts of the hierarchy, uncertainties within the model are never placed arbitrarily and because of this, the variational posteriors and true posteriors look much more similar. They tend to disagree in places when there is high uncertainty about the alignment.

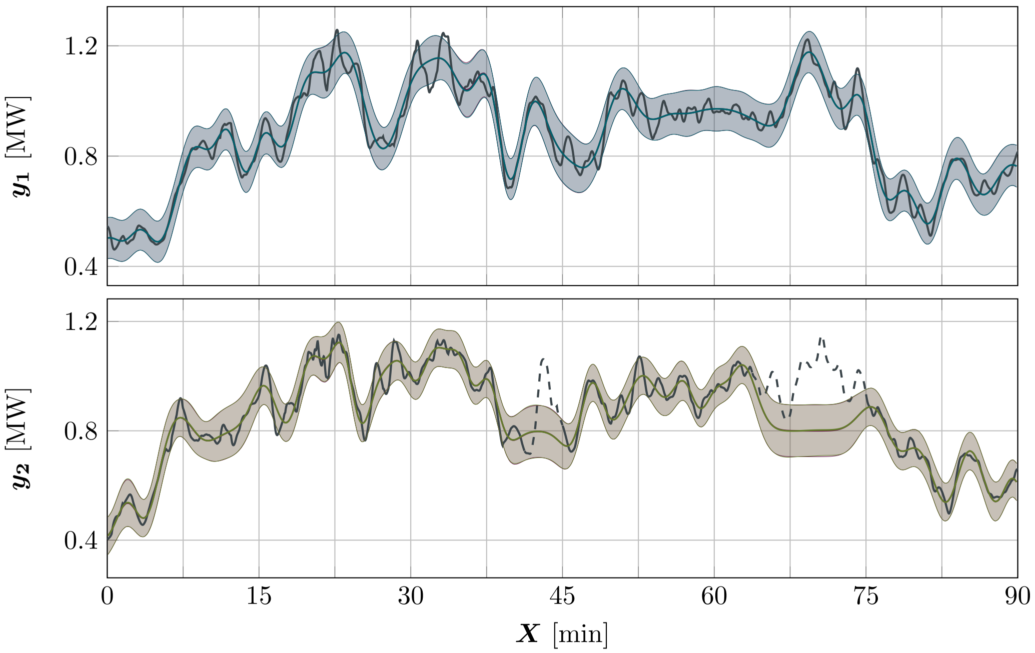


**Figure 6:** GP

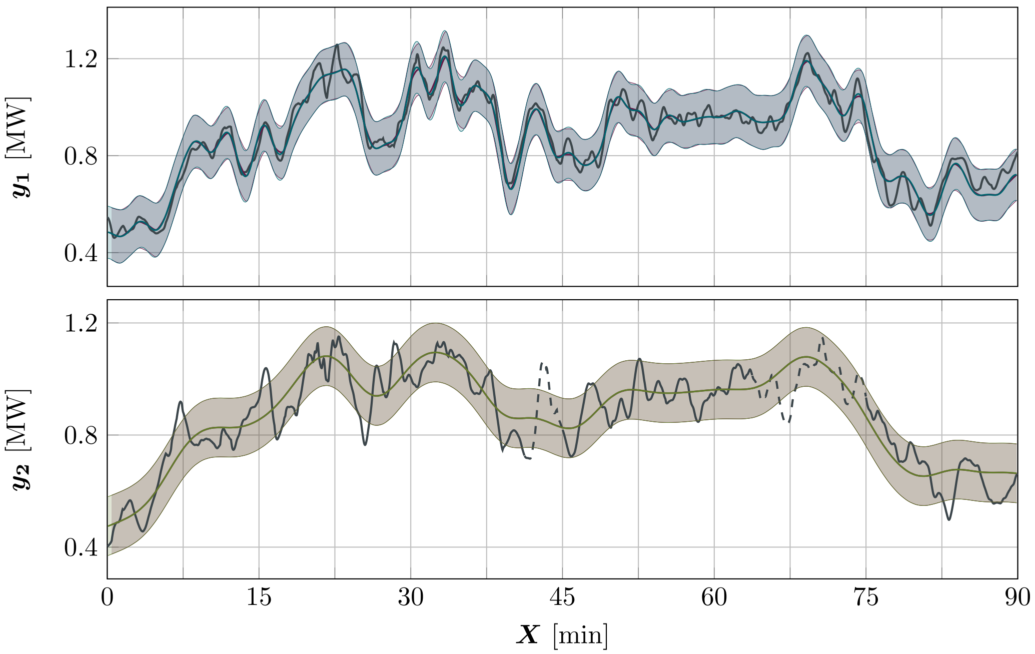


**Figure 7:** MO-GP

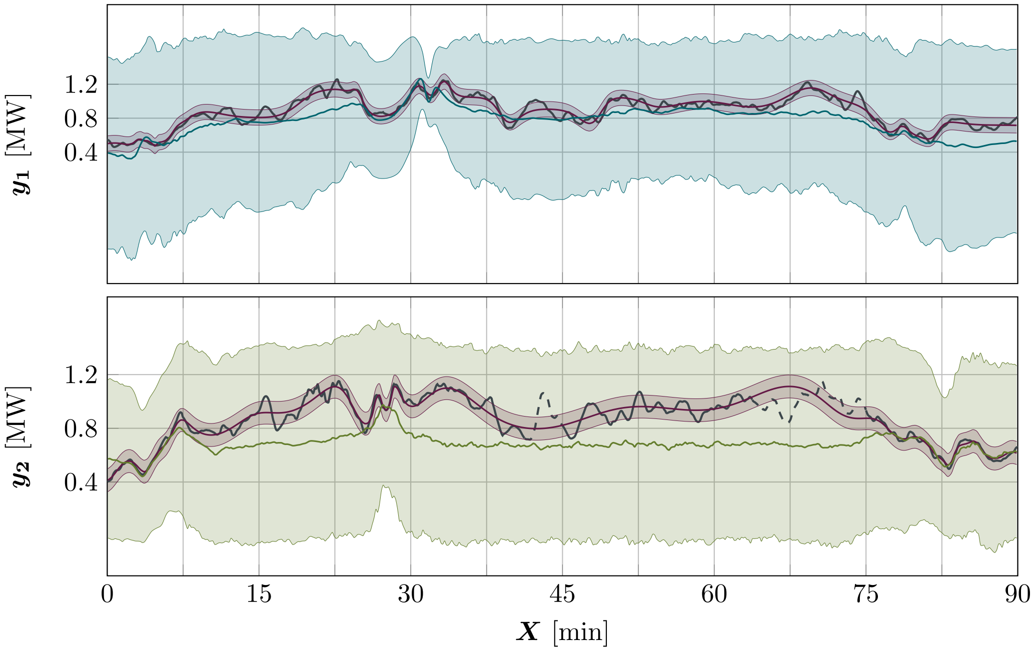


**Figure 8:** DGP

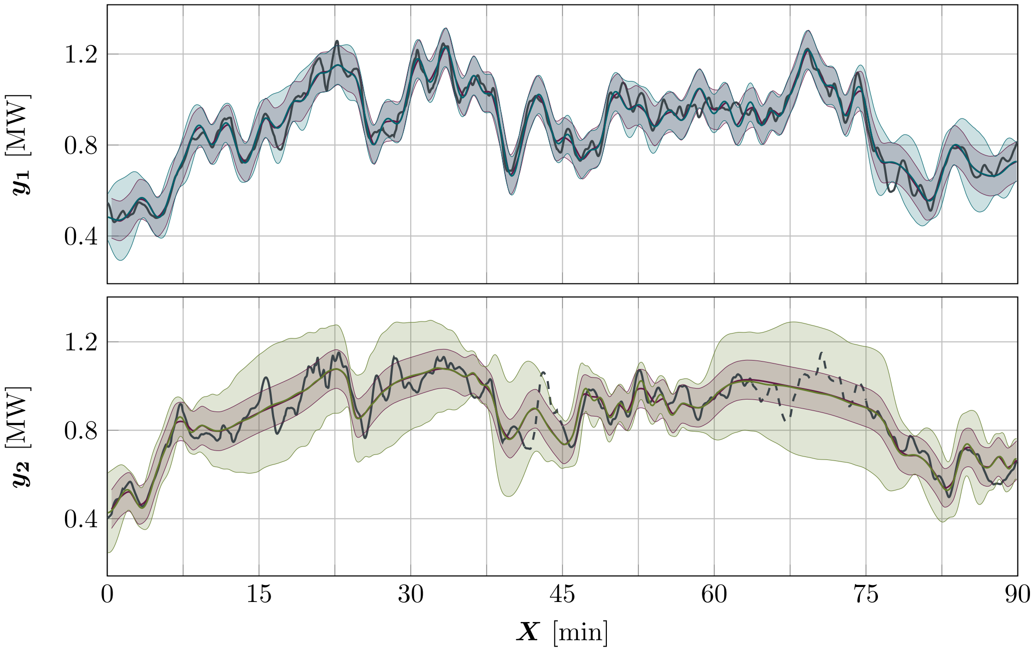


**Figure 9:** AMO-GP (Ours)