Journal of Machine Learning Research 22 (2021) 1-51 Submitted 4/16; Revised 2/21; Published 4/21

Multi-view Learning as a Nonparametric Nonlinear Inter-Battery Factor Analysis

Andreas Damianou DAMIANOU@AMAZON.COM
Amazon, Cambridge, United Kingdom

Neil D. Lawrence NDL21@CAM.AC.UK
University of Cambridge, United Kingdom

Carl Henrik Ek CHE29@CAM.AC.UK
University of Cambridge, United Kingdom

Editor: Amos Storkey

Abstract

Factor analysis aims to determine latent factors, or traits, which summarize a given data set. *Inter-battery* factor analysis extends this notion to multiple views of the data. In this paper we show how a nonlinear, nonparametric version of these models can be recovered through the Gaussian process latent variable model. This gives us a flexible formalism for multi-view learning where the latent variables can be used both for exploratory purposes and for learning representations that enable efficient inference for ambiguous estimation tasks. Learning is performed in a Bayesian manner through the formulation of a variational compression scheme which gives a rigorous lower bound on the log likelihood. Our Bayesian framework provides strong regularization during training, allowing the structure of the latent space to be determined efficiently and automatically. We demonstrate this by producing the first (to our knowledge) published results of learning from dozens of views, even when data is scarce. We further show experimental results on several different types of multi-view data sets and for different kinds of tasks, including exploratory data analysis, generation, ambiguity modelling through latent priors and classification.

Keywords: representation learning, factor analysis, Gaussian processes, inter-battery factor analysis

1. Introduction

Learning representations from observed data is a central aspect of machine learning. In supervised learning we wish to retain the information in the input data which is relevant for performing the supervised task, while in the unsupervised learning scenario we seek to discover the underlying patterns and structures in the data to associate them with some meaning. In this paper the focus is on the unsupervised scenario which we will refer to as *representation learning.* A large body of work on representation learning considers approaches which build upon factor analysis. The general formulation of factor analysis is often attributed to Spearman (1904) from his work in experimental psychology and his theory of "general intelligence" as a *factor* to explain different human characteristics. However, the underlying ideas have a much longer history, stretching all the way back to the philosophers of ancient Greece. For a historical account of the philosophical ideas underlying factor analysis we would like to point the reader to the excellent work of Mulaik (1987).

A fundamental aspect of factor analysis is that the solution is unidentifiable. This means that there are infinitely many solutions which cannot be differentiated by the model. Even though all solutions might be completely equivalent from a statistical view point, from an experimental view point it is often important to be able to select a particular solution. This is because different solutions are often attributed with different "meanings". However, with many possible solutions it is not clear how to find a criterion justifying one explanation of the data over another. Disregarding this problem has been referred to as the *the inductivist fallacy* (Chomsky and Fodor, 1980). In other words, additional information capable of differentiating between the solutions needs to be included in order to completely solve the problem.

The most common approach to factor analysis is to associate the importance of each factor relatively to the amount of variance it explains in the data. This is known as principal component analysis (PCA) (Hotelling, 1933)[1]. Another approach is to choose a representation that minimizes distortions between the data and the factors, an approach known as classical multi-dimensional scaling (Mardia et al., 1979; Cox and Cox, 2008). A more recent approach, particularly relevant to our paper, is to relate the observed data and the discovered factors via a generative mapping for which we additionally include a preference for its functional form. Gaussian process latent variable models (GP-LVMs – Lawrence, 2005) achieve this in a probabilistic manner by employing Gaussian process priors. The benefit of taking this approach is that the representations are learned in a nonlinear, nonparametric fashion.

It is often the case that our data come from several different observation streams (also called "views" or "modalities") which arise from the same situation or phenomenon. Throughout this paper we will use the word "view" to mean two variables that are somehow aligned. One example of this could be the image frames of a video and the sound, the motion of the left and right arm of a person walking; or a blood and bonemarrow sample from the same patient. What we mean by "view" is on purpose very loose, simply referring to data that has been somehow aligned a-priori. The representation learning modelling approach has then to be extended to the *multi-view* scenario. In (Sharma et al., 2012) it is shown how several different factor analysis models can be extended to the multi-view setting. In particular, Canonical Correlation Analysis (CCA) is a linear model which, unlike PCA, learns representations based on the correlation between multiple views. On the other hand, inter-battery factor analysis (IBFA) (Tucker, 1958) considers a segmented (factorized) representation, meaning that it accounts for components to describe cross-view information (such as correlation) as well as within-view information (such as variance). This model, as many other factor analysis models (Hotelling, 1933, 1936; Spearman, 1904), was first proposed in experimental psychology. Much later it was rediscovered in machine learning

[1] Authors often allocate principal component analysis to Pearson (1901), but he was trying to resolve a different problem: that of a symmetric regression problem, and the model is different. Hotelling was inspired by factor analysis and his model is similar to those of Spearman. This may seem to be splitting hairs because *algorithmically* these models are fitted in the same manner, but the difference emerge when the models are nonlinearized, leading to different algorithms, so the interpretation of the model is important.

(Klami and Kaski, 2006; Ek et al., 2008a; Archambeau and Bach, 2008) under a probabilistic view point which, as noticed by Klami et al. (2013), ends up being very related to Browne's probabilistic interpretation (Browne, 1979) of IBFA.

In this work we focus on the aforementioned probabilistic generative model interpretation of IBFA. Our contribution is on achieving this in a nonlinear, nonparametric fashion by building upon the representation learning capabilities of the GP-LVM. By relaxing the assumption of linear mappings, our model manages to learn the complex relationships that exist in real, noisy data. Specifically, our model represents observed views $\mathbf{Y}^{(k)}$ through a common latent space $\mathbf{X}$. Separate Gaussian process mappings $f^{(k)}$ relate $\mathbf{X}$ to each view $\mathbf{Y}^{(k)}$. We introduce additional kernel parameters inside each $f^{(k)}$ which, in combination with our inference procedure, allow the model to "switch off" latent dimensions which are deemed unnecessary by each view independently. This provides an automatic factorization of the latent space. Therefore, we show how inter-battery factor analysis naturally manifests itself as a Gaussian process latent variable model. We further develop a Bayesian inference scheme and demonstrate that it is crucial in eliminating information redundancy in the latent space and, hence, in achieving efficient, automatic latent space factorization. We build on our previous work (Damianou et al., 2012) and further we reinterpret that model as inter-battery factor analysis and consider extensions to more than two views.

Although we do demonstrate downstream uses of our model in supervised learning (e.g. classification), our primary focus is to learn useful, interpretable representations from multiple aligned observation spaces, even when data is scarce. Therefore, we concentrate our efforts on latent space modeling and inference and, in particular, we strive to incorporate uncertainty in the probabilistic latent space. This is combined with our Gaussian process latent to observed space mapping in order to achieve nonlinear, multi-view representation learning with minimal assumptions about the data and by avoiding the risk of overfitting.

In summary, the main contributions of our paper are the following:

- a probabilistic, nonlinear, nonparametric formulation of inter-battery factor analysis,
- a variational framework for approximate, data-efficient Bayesian learning,
- a framework for introducing priors that encourage specific latent space configurations,
- a multi-view latent consolidation model that naturally extends beyond two views; to our knowledge, we present the first work which shows results of a data-efficient, factorized generative model with truly large number of views.

In the next section we will describe the relevant related work, placing the proposed model in context. We will then provide an introduction to Gaussian processes in general and Gaussian process latent variable models in specific. Section 4 represents the central part of the paper where we will describe and motivate the model we propose. We will then proceed in Section 5 to show experimental results of the proposed model comparing the approach to competing methods. Finally, Section 6 concludes the paper and outlines directions of future work.

2. Background

2.1 Multi-view learning

A large portion of the multi-view learning literature has been motivated CCA. In CCA the goal is to find two linear transformations, one for each view of the data, which align directions with significant correlation between the views. This model is often applied in a scenario where one wants to extract information about one view given the other. The same approach has been extended to learn transformations involving multiple views (Rupnik and Shawe-Taylor, 2010; Nicolaou et al., 2014). Traditionally, CCA assumes that the transformations are linear, something which constrains the applicability of the method. To that end, several nonlinear extensions have been suggested, for example through kernels (Kuss and Graepel, 2003). Using CCA in an unsupervised manner to find a joint representation of two data sets assumes that the maximally correlated directions are also directions of maximal variance. However, correlation as a measure can be rather misleading, as it does not depend on the actual statistical variance that is represented by the aligned directions. This means that the aligned subspace after projection might only represent a small portion of the data variance. In many application scenarios, correlation between views is manifested not only in the signal but also in the noise; this renders CCA-based approaches problematic, because they are prone to learning representations that amplify the noise.

One approach for overcoming the aforementioned limitation of CCA is to encode the views in such a manner that the representation encapsulates as much of the data variance as possible. This can be seen as a *joint dimensionality reduction problem* which is associated with the task of discovering low-dimensional manifolds given the observed views. Several past approaches have followed this idea, e.g. Ham et al. (2003, 2005, 2006). Each of the above three methods takes inspiration from the approaches previously developed for spectral dimensionality reduction from a single view (Tenenbaum et al., 2000; Yan et al., 2007; Roweis and Saul, 2000; Weinberger and Saul, 2006; Lawrence, 2010). These methods build constraints from local statistics in the data by aiming to find a manifold that preserves the *local* distances in the data set. However, computation of such distances is not always possible, for example when there is missing data. Further, noise in the data can result in "topological instability" (Balasubramanian, 2002), for example when two noisy points appear close together when in reality they should be embedded far apart.

By directly modelling the generative mapping this problem can be avoided. Bach and Jordan (2005) showed how CCA can be derived through a probabilistic latent variable model (this interpretation has roots in (Tucker, 1958)). Besides, Kalaitzis and Lawrence (2012) showed that CCA can also arise through formulation of a form of residual component analysis. Both models interpret the canonical correlates as directions induced in a latent variable model. In common with other generative approaches, rather than trying to map from the data to the latent space, they directly map from the latent to the data space by building a probabilistic model of the data.

Since multi-view learning is largely a representation learning task, it is natural to consider leveraging deep neural networks (DNNs) which are able to learn rich patterns from data. For example, a body of work (Andrew et al., 2013; Wang et al., 2015) combines the CCA objective with DNN feature extraction from two views; this approach inherits the CCA issues mentioned before and lacks a model for sample generation. Ainsworth et al.

(2018) instead consider a generative structured sparsity model based on Variational Auto-Encoders (Kingma and Welling, 2013); however, their focus is on learning from grouped data rather than multi-view data (i.e. there is a single observation space with groups of outputs exhibiting structured correlations). In general, DNN-based models require at least one order of magnitude more multi-view training observations, compared to our model. Furthermore, learning multi-view representations based on DNNs is a significantly more challenging task compared to learning from a single observation space. Indeed, each additional view introduces a very large amount of new parameters. These parameters are learned in a non-Bayesian fashion and regularization tricks like dropout become less straightforward to apply, overall hindering the optimization process.

A typical assumption of spectral and generative multi-view modelling methods assumes that that all views share the same variations, as they leave no way of incorporating variance that is specific to one of the particular views. For many types of data this is a very crude assumption, as it means that all views share all generating parameters. Factorized latent variable models can tackle this issue by modeling separately the private and shared information contained in the different views[2]. The benefits of a factorized latent representation is best motivated through an example. Consider the task of trying to infer the full three-dimensional pose of a human body from its two-dimensional silhouette image, as illustrated in Figure 1. This is a very challenging task as it is not possible to differentiate between all different poses by looking only at the silhouette. Clearly, the same pose parameters can generate different silhouettes depending on the "shape" of the human. Similarly, a specific silhouette could have been generated by different poses, i.e. it is the pose-specific information which disambiguates the silhouette and not the shared information for pose/silhouette. Therefore, a shared latent space is needed to relate the two views and a private latent space is needed to disambiguate the prediction. A latent variable approach which does not consider private spaces will result in one of two undesirable scenarios: it will either completely avoid representing the non-shared variations or it will retain those variations but at the cost of reducing the quality of the estimate for the views that are not explained by them.

2.2 Factorized latent variable models

Let us consider two views $\mathbf{Y}^{(1)} \in \Re^{n \times p_1}$ and $\mathbf{Y}^{(2)} \in \Re^{n \times p_2}$ where there is an implicit correspondence (alignment) between each pair of instances belonging to the two views. Following the motivation of the previous paragraph, our goal is to learn a joint representation of the distribution over *both* sets of data. A factorized latent variable model aims at representing the variance from all views in a segmented latent space $\mathbf{X} = [\mathbf{X}^{(1)}, \mathbf{X}^{(2)}, \mathbf{X}^{(1,2)}]$, where $\mathbf{X}^{(1,2)}$ encodes the variations that exist in both views and $\mathbf{X}^{(1)}$ and $\mathbf{X}^{(2)}$ represent the variations that are unique to each respective view. Efficient segmentation of the available variation into shared and private latent variables is crucial. Compared to learning a separate model for each view (i.e. absence of shared information modelling), a factorized model provides a more efficient representation by re-using the shared variations to represent the shared signal of the views. In the other end of the spectrum, if we build a "factorized" model which fails to account for private factors we will end up with an over-parameterized latent representation which will be forced to relate irrelevant variations to each view.

[2] We refer to such models as "factorized" to emphasize that they constitute factor analyzers.

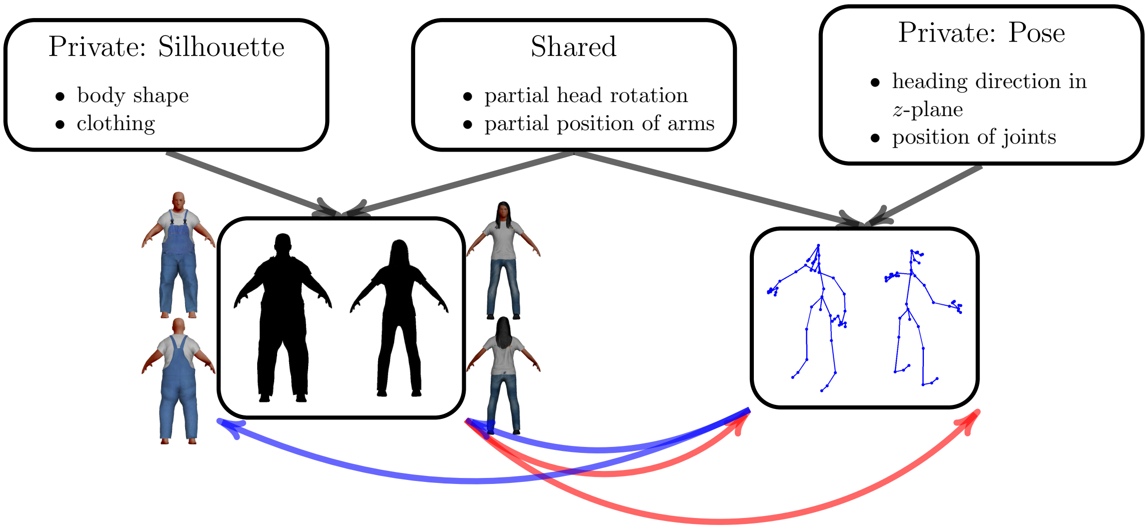


Figure 1: The above figure shows the benefit of a factorized latent representation using the example of 3D human pose and silhouette images. The blue and red arrows show the multi-modal relationship that exists in both "directions". By following the direction of the blue arrows it is not possible to determine information which is private to the silhouette, such as the shape of the body. Analogously, the silhouette for a person facing towards or away from the camera is the same, which means that the heading angle cannot be determined for this pose by following the red arrows direction.

One way of seeing factorized latent variable models is as extensions of the IBFA approach (see Klami et al., 2013), which we will now introduce more formally. Consider the case where we seek to relate two views $\mathbf{Y}^{(1)} = \{\mathbf{y}_{i,:}^{(1)}\}_{i=1}^{n}$ and $\mathbf{Y}^{(2)} = \{\mathbf{y}_{i,:}^{(2)}\}_{i=1}^{n}$ to and through a latent space $\mathbf{X} = \{\mathbf{x}_{i,:}\}_{i=1}^{n}$. To achieve this, the probabilistic IBFA approach considers a factorization of the latent variables (i.e. columns of $\mathbf{X}$ or, equivalently, dimensions of each latent instance $\mathbf{x}_{i,:}$) together with the definition of the following model:

$$\mathbf{y}_{i,:}^{(1)} = \mathbf{W}^{(1)}\mathbf{x}_{i,:}^{(1,2)} + \mathbf{U}^{(1)}\mathbf{x}_{i,:}^{(1)} + \boldsymbol{\epsilon}_{i,:}^{(1)}$$
$$\mathbf{y}_{i,:}^{(2)} = \mathbf{W}^{(2)}\mathbf{x}_{i,:}^{(1,2)} + \mathbf{U}^{(2)}\mathbf{x}_{i,:}^{(2)} + \boldsymbol{\epsilon}_{i,:}^{(2)},$$

where the latent variables $\mathbf{x}$ are assigned a standard normal distribution and the noise terms are also Gaussian with diagonal covariance matrices: $\boldsymbol{\epsilon}_{i,:}^{(1)} \sim \mathcal{N}\left(\mathbf{0}, \boldsymbol{\Sigma}^{(1)}\right)$ (and similarly for $\boldsymbol{\epsilon}_{i,:}^{(2)}$). The latent variables are factorized into those shared across views, $\mathbf{x}_{i,:}^{(1,2)}$, and those specific to each view, $\mathbf{x}_{i,:}^{(1)}, \mathbf{x}_{i,:}^{(2)}$. Separate factor loadings (namely $\mathbf{W}^{(k)}$ and $\mathbf{U}^{(k)}, k = 1, 2$) map the latent factors linearly to the observation space with the addition of noise. Notice that by setting $\mathbf{U}^{(1)}, \mathbf{U}^{(2)}$ to be matrices of zeros and allowing $\boldsymbol{\Sigma}^{(1)}, \boldsymbol{\Sigma}^{(2)}$ to be non-diagonal,

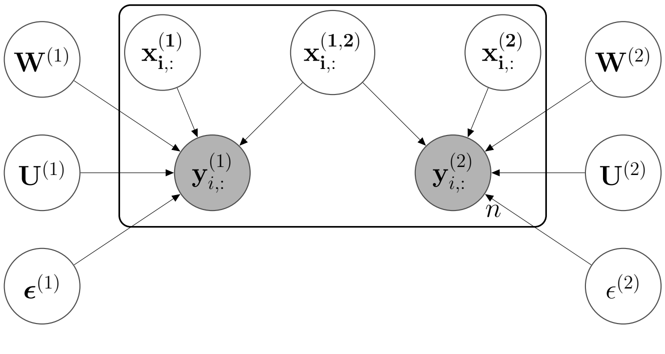

$\mathbf{W}^{(1)}$
$\mathbf{x}_{\mathbf{i},:}^{(\mathbf{1})}$
$\mathbf{x}_{\mathbf{i},:}^{(\mathbf{1},\mathbf{2})}$
$\mathbf{x}_{\mathbf{i},:}^{(\mathbf{2})}$
$\mathbf{W}^{(2)}$
$\mathbf{U}^{(1)}$
$\mathbf{y}_{i,:}^{(1)}$
$\mathbf{y}_{i,:}^{(2)}$
$\mathbf{U}^{(2)}$
n
$\boldsymbol{\epsilon}^{(1)}$
$\epsilon^{(2)}$

Figure 2: The graphical model for the inter-battery factor analysis model. The two observed sets of data, $\mathbf{Y}^{(1)}$ and $\mathbf{Y}^{(2)}$, are modelled using two latent variables $\mathbf{X}^{(1)}$ and $\mathbf{X}^{(2)}$ which describe the private variations, as well as a single latent variable $\mathbf{X}^{(1,2)}$ which describes the shared variations. The mappings from the latent to the observed data are linear and parameterized by $\mathbf{W}^{(1),(2)}$ and $\mathbf{U}^{(1),(2)}$ for the shared and the private spaces respectively.

we obtain a probabilistic version of CCA (Bach and Jordan, 2005), which does not make the factorization of the latent space explicit. Figure 2 illustrates this model graphically.

The ideas of factorized latent structure for multi-view learning have been adopted in several works (Jia et al., 2010; Leen and Fyfe, 2008; Salzmann et al., 2010) for latent variable models and also applied to topic models (Jia et al., 2011; Virtanen et al., 2012; Zhang et al., 2013). In this paper we are interested in developing a probabilistic, nonparametric and nonlinear version of IBFA by building on the GP-LVM. Below we review the previous GP-LVM based approaches for multi-view learning, and emphasize that these approaches were non-factorized and non-Bayesian. We start with Figure 3(a) where we illustrate the single-view model, with the (non-factorized) extension to multiple views (Shon et al., 2006; Ek et al., 2007) in panel (b). In panel (c) we illustrate the approach of Ek et al. (2007) who incoroporated a back-constraint (Lawrence and Quiñonero-Candela, 2006) from one view to the latent space, in a first attempt to rectify the identifiability problem. The back-constraint from view $\mathbf{Y}^{(2)}$ implies a bijective relationship to which the generation of the remaining view, $\mathbf{Y}^{(1)}$ has to adapt through discarding the variations which are not present in $\mathbf{Y}^{(2)}$. The purpose of this was to force the latent space to ignore variations that did not exist in the non-constrained space. This approach should be considered as feature selection rather than consolidation of the views, as it directly uses one view as supervision to determine which variations are important in the other. Therefore, this approach could still not solve the problem of polluting the latent space with non-shared variations. This approach has also been used to extend to all views (Eleftheriadis et al., 2015, 2014). However this adds a severe constraint on the latent representation, as it needs to be a bijection from *all* the views, thereby hindering learning.

The application of the IBFA idea and the pioneering work of Klami and Kaski (2006) to GP-LVM based models comes from Ek et al. (2008a) and is graphically illustrated in Figure 3(d). In practice, learning the factorization of $\mathbf{X}$ within a GP-LVM based model is extremely challenging, due to the non-Bayesian treatment of $\mathbf{X}$ and the sensitivity of the

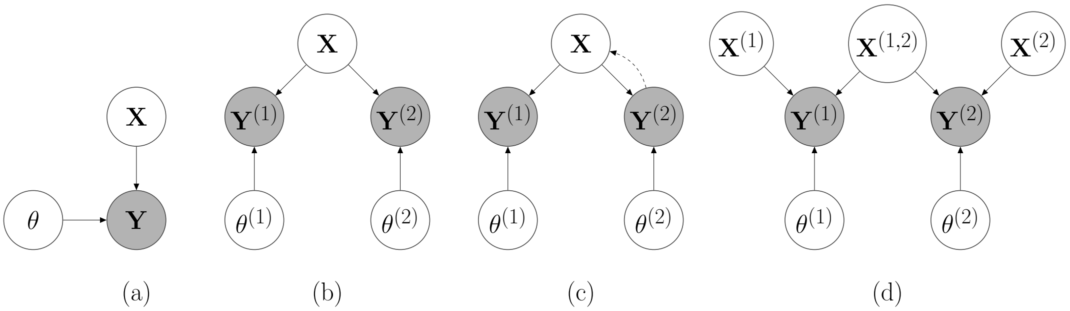


Figure 3: The development of the GP-LVM models from the initially proposed (Lawrence, 2005) (panel (a)) with a single observation and latent space through the first models adapted to a multi-view scenario with several observation spaces (Shon et al., 2006; Ek et al., 2007) (panels (b),(c)) to the first proposed model with a factorized latent space (Ek et al., 2008a) (panel (d)). Shaded nodes represent observed variables; $\boldsymbol{\theta}$ denotes the kernel parameters of the Gaussian process mappings; the dashed line represents a back-constraint.

model to initialization. This has lead to several different heuristics aimed at encouraging a specific separation of the variations. Ek et al. (2008a) proposed a spectral algorithm which, in combination with CCA, could factorize two views. In (Salzmann et al., 2010) a regularizer that encourages orthogonality between the subspaces was proposed, in order to reach a solution. Despite promising results, the method required a rather ad-hoc annealing scheme during learning. Further, orthogonality of sub-spaces as a constraint on a multi-view probabilistic model is an idea that lacks theoretical support. Indeed, the orthogonality of principal component analysis arises only as a constraint to resolve its rotational ambiguity, but when maximum likelihood solutions are sought, orthogonality cannot in general be shown to constitute a sensible constraint, even if it may provide a useful initialization. Jia et al. (2010) used a sparse linear model which reduces the complexity during model learning. The results of this *linear* method were better than the nonlinear method of Salzmann et al. (2010). This indicates that the latter model, even though it is representationally more powerful, is not always able to exploit this representational power due to difficulties in optimization.

In this paper we propose a probabilistic and nonlinear formulation of the IBFA model framed as a GP-LVM model. The proposed paper extends the shared multi-view approach proposed by Shon et al. (2006) using the factorized structure described by Ek et al. (2008b). The main contribution of the paper is a fully Bayesian treatment of the model which avoids reliance on heuristics, such as the regularizers proposed by Salzmann et al. (2010) for learning the factorization. Not only does this allows us to learn the structure of the factorization but also the dimensionality of each of the subspaces, which is a free parameter of the maximum likelihood. In contrast to (Klami et al., 2013) our model is nonlinear and fully nonparametric. The nonparametric nature is directly inherited from the use of a Gaussian process prior over the mappings between the latent space and the data space. Before we continue, we explain the notation used throughout the paper and, subsequently, we provide a more detailed description of Gaussian processes and their use in data consolidation and dimensionality reduction.

2.3 Notation

Throughout this paper, matrices are represented using boldface capital letters, vectors using boldface, lowercase letters and scalars using regular type face, lowercase letters. The indexing is as follows: given K views we will refer to the observed outputs in view k as $\mathbf{Y}^{(k)}$, and this collection of variables will be referred to as the "data view k". The individual points in a matrix are stored by rows, e.g. $\mathbf{Y}^{(k)} = [\mathbf{y}^{(k)}_{1,:}, \ldots, \mathbf{y}^{(k)}_{n,:}]^{\mathrm{T}}$. We will also denote dimensions (columns) of $\mathbf{Y}^{(k)}$ by $\mathbf{y}^{(k)}_{:,j}$ whereas $y^{(k)}_{i,j}$ denotes dimension j of the ith point. When we refer to test points the notation for rows, columns and single elements of a matrix becomes $\mathbf{y}^{(k)}_{i,*}$, $\mathbf{y}^{(k)}_{*,j}$ and $y^{(k)}_{*;i,j}$ respectively. Any matrix indexing will follow these conventions. $\mathbf{Y}^{\mathcal{A}}$ indicates a set of output data corresponding to a set of views $\mathcal{A} \subseteq (1, \ldots, K)$. As a short hand notation for referring to data from *all* views we will use $\mathcal{Y}$ i.e. $\mathcal{Y} \equiv \mathbf{Y}^{(1,\ldots,K)} = \{\mathbf{Y}^{(1)}, \mathbf{Y}^{(2)}, \ldots, \mathbf{Y}^{(K)}\}$. Latent variables are also called inputs because of the dependencies in the graphical model. The indexing $\mathbf{X}^{(k)}$ represents the latent space which is *private* for view k, that is, in the generative model the information in $\mathbf{X}^{(k)}$ is used to generate $\mathbf{Y}^{(k)}$ and no other view. Similarly, $\mathbf{X}^{(k,l)}$ represents the latent space which is *shared* between view k and l.

3. Gaussian Process Modeling

A Gaussian process (GP) is a stochastic process where each finite set of realizations follows a joint Gaussian distribution. The process is parametrized by its mean $\mu(\mathbf{x})$ and covariance function $k(\mathbf{x}, \mathbf{x}')$, where $\mathbf{x}, \mathbf{x}'$ denote instances taken from the input domain of the GP. A very common choice for the mean function is to constitute a constant vector of zeros, of appropriate size. The covariance function domain is an infinite set and is parametrized by $\boldsymbol{\theta}$, which is called a *hyperparameter set* with respect to the model. Due to the cardinality of the input domain, a GP can be used as a flexible prior over functions (Rasmussen and Williams, 2006) which allows for a fully Bayesian and analytic treatment over the space of functions.

More formally, consider the situation where we wish to model the relationship between two variables $\mathbf{X}$ and $\mathbf{Y}$ as a function f, i.e. $f : \mathbf{X} \to \mathbf{Y}$. Given corresponding multivariate instantiations $\mathbf{X} = [\mathbf{x}_{1,:}, \ldots, \mathbf{x}_{n,:}]^\top$ and $\mathbf{Y} = [\mathbf{y}_{1,:}, \ldots, \mathbf{y}_{n,:}]^\top$ where $\mathbf{x}_{i,:} \in \Re^q$ and $\mathbf{y}_{i,:} \in \Re^p$, we assume that each dimension of an instance $\mathbf{y}_{i,:}$ is generated by an independent function $f_{:,j}$ from input $\mathbf{f}_{i,:}$ with the addition of Gaussian noise $\epsilon_{i,j} \sim \mathcal{N}\left(0, \beta^{-1}\right)$, according to:

$$y_{i,j} = f_{:,j}(\mathbf{x}_{i,:}) + \epsilon_{i,j}. \tag{1}$$

All mapping functions are assigned a Gaussian process prior with common hyperparameters, that is, $f_{:,j} \sim \mathcal{GP}(\mathbf{0}, k(\mathbf{x}, \mathbf{x}'))$. The Gaussian process prior means that the set of all available function instantiations $\mathbf{f}_{:,j}$ are distributed as

$$p(\mathbf{f}_{:,j}|\mathbf{X}, \boldsymbol{\theta}) = \mathcal{N}(\mathbf{f}_{:,j}|\mathbf{0}, \mathbf{K}_{ff}), \tag{2}$$

where $\mathbf{K}_{ff} = k(\mathbf{X}, \mathbf{X})$ denotes the covariance matrix obtained after evaluating the covariance function k on all available instances $\mathbf{X}$. The choice for a particular covariance function can result from a model selection routine (e.g. cross-validation), or can be seen as an assumption during model design. A popular covariance function which encodes the assumption of smoothness in the input space is the infinitely differentiable exponentiated quadratic (EQ) one, also known as RBF:

$$k_{\mathrm{EQ}}(\mathbf{x}_{i,:}, \mathbf{x}_{r,:}) = \sigma_{\mathrm{EQ}}^2 \exp\left(-\frac{w}{2} \sum_{j=1}^{q} (x_{i,j} - x_{r,j})^2 \right), \tag{3}$$

where the kernel variance σ_{EQ}^2 and the so-called squared inverse length-scale w constitute the hyparparameter set $\boldsymbol{\theta}$.

Further, the assumption about Gaussian noise in the generative model of equation (1) means that the outputs are distributed according to

$$p(\mathbf{y}_{:,j} | \mathbf{f}_{:,j}, \beta) = \mathcal{N}(\mathbf{y}_{:,j} | \mathbf{f}_{:,j}, \beta^{-1}\mathbf{I}).$$

Then, the marginal likelihood can be written in closed form as

$$p(\mathbf{y}_{:,j} | \mathbf{X}, \boldsymbol{\theta}) = \int_{\mathbf{f}_{:,j}} p(\mathbf{y}_{:,j} | \mathbf{f}_{:,j}) p(\mathbf{f}_{:,j} | \mathbf{X}, \boldsymbol{\theta}),$$

where we dropped conditioning on β from the expression, for clarity. Both of the distributions appearing in the integral are Gaussian and, due to the self-conjugacy of the Gaussian distribution, the marginal likelihood is also a Gaussian. Traditionally, the parameters $\boldsymbol{\theta}$ of the GP and the noise precision, β, are learned together by maximizing the above marginal likelihood.

3.1 Gaussian Processes Latent Variable Models

The Gaussian process latent variable model (GP-LVM Lawrence, 2005) is an unsupervised learning framework for using GP priors for dimensionality reduction. Given a set of high dimensional data $\mathbf{Y} \in \Re^{n \times p}$, the aim is to explain them through a set of low dimensional variables $\mathbf{X} \in \Re^{n \times q}$. The setting is similar to that of standard GP regression but now only the outputs $\mathbf{Y}$ are observed, whereas the inputs $\mathbf{X}$ are considered to be *latent*. Each dimension of the observations, $\mathbf{y}_{:,j}$, is assumed to be generated by the same latent input variable $\mathbf{X}$ via a GP mapping, as in equation (1). To better understand how dimensionality reduction is achieved by the GP-LVM, we write here the full generative model with all appropriate factorizations:

$$\begin{aligned} p(\mathbf{Y} | \mathbf{X}, \boldsymbol{\theta}) &= \int_{\mathbf{F}} p(\mathbf{Y} | \mathbf{F}) p(\mathbf{F} | \mathbf{X}, \boldsymbol{\theta}) \\ &= \int_{\mathbf{F}} \prod_{j=1}^{p} \prod_{i=1}^{n} p(y_{i,j} | f_{i,j}) \prod_{j=1}^{p} p(\mathbf{f}_{:,j} | \mathbf{X}, \boldsymbol{\theta}) \\ &= \prod_{j=1}^{p} \mathcal{N}\left(\mathbf{y}_{:,j} | \mathbf{0}, \mathbf{K}_{ff} + \beta^{-1}\mathbf{I} \right). \end{aligned} \tag{4}$$

The above equation is exactly the same as for the GP regression case, but now the optimization procedure also includes the latent variables $\mathbf{X}$. By choosing the dimensionality of $\mathbf{X}$ to be much smaller than that of the observed data, the GP provides sufficient regularization such that both the parameters $\boldsymbol{\theta}$ and the latent locations $\mathbf{X}$ can be found through maximum likelihood. Due to the flexible nature of the GP-LVM a large range of extensions have been suggested. In particular, by adding priors and seeking a MAP solution to the latent variable $\mathbf{X}$ different structures of latent representations can be found such that are topologically constrained (Urtasun et al., 2008), consistent with a dynamic assumption (Wang et al., 2008) or constrained by class information (Urtasun and Darrell, 2007), to name just a few.

3.2 Shared Gaussian Process Latent Variable Models

The focus of this paper is to model K different views $\mathbf{Y}^{(k)}$ within the same model. Shon et al. (2006) proposed a GP-LVM where two sets of observations, $\mathbf{Y}^{(1)}$ and $\mathbf{Y}^{(2)}$, are assumed to be generated from the same latent variable $\mathbf{X}$ by two sets of independent GP mappings, each with a separate covariance function. To generalize this beyond two views we introduce further notation for the collections of instantiated random variables for all views, i.e. we denote the observed views as $\mathcal{Y} = \{\mathbf{Y}^{(k)}\}_{k=1}^{K}$ and their associated noiseless versions as $\mathcal{F} = \{\mathbf{F}^{(k)}\}_{k=1}^{K}$. The latter depend on separate GP mappings which are parameterized respectively by $\boldsymbol{\Theta} = \{\boldsymbol{\theta}^{(k)}\}_{k=1}^{K}$. By denoting the jth dimension of all data points in the kth observation space as $\mathbf{y}_{:,j}^{(k)}$, the marginal distribution of equation (4) is now generalized to,

$$\begin{aligned} p(\mathcal{Y}|\mathbf{X}, \boldsymbol{\Theta}, \{\beta^{(k)}\}_{k=1}^{K}) &= \int_{\mathcal{F}} p(\mathcal{Y}|\mathcal{F}, \{\beta^{(k)})\}_{k=1}^{K}) p(\mathcal{F}|\mathbf{X}, \boldsymbol{\Theta}) \\ &= \int_{\mathcal{F}} \prod_{k=1}^{K} p(\mathbf{Y}^{(k)}|\mathbf{F}^{(k)}, \beta^{(k)}) p(\mathbf{F}^{(k)}|\mathbf{X}, \boldsymbol{\theta}^{(k)}) \\ &= \prod_{k=1}^{K} \prod_{j=1}^{p} p(\mathbf{y}_{:,j}^{(k)}|\mathbf{X}, \boldsymbol{\theta}^{(k)}, \beta^{(k)}), \end{aligned} \tag{5}$$

where we exceptionally included all parameters in the expressions to clearly show the dependencies.

From the above explained generative model, referred to as shared GP-LVM, different variants emerge depending on the regularization and factorization imposed for the latent space. These variants were reviewed in Section 2.1 and summarized in Figure 3. Having just described the shared GP-LVM in detail, we will now explain the issues that have caused major practical problems in past shared GP-LVM approaches. Specifically, the straightforward adoption of the GP-LVM to the multi-view scenario by the shared GP-LVM implies certain non-obvious assumptions. Being a generative model, the GP-LVM objective reflects how well *all* the variations in all views are summarized (modelled) in the latent space. This means that if a variation is confined to only one view this might have a detrimental effect on the generation of the other views, as the latent space will be reluctant to encode this. The approach of Ek et al. (2008a) has proven to be the more robust in terms of avoiding the negative effect of having private variations polluting the model. Recall that this model introduced a factorized latent space where the shared and private subspaces are completely segmented: a single shared space $\mathbf{X}^{(s)}$ represents the variations that co-existed in all K views and one private space $\mathbf{X}^{(k)}$ associated with each view represents the variations unique to that specific view. The learning of the factorized latent space model is performed in the same manner as for the standard GP-LVM model, i.e. by maximum likelihood through gradient based methods. The non-convexity of the objective function requires a good initialization of the latent locations for such approach to reach a good solution. For a model with one single latent space the suggested approach in (Lawrence, 2005) was to initialize the latent space with the solution to an analogous spectral nonlinear algorithm such as isomap (Tenenbaum et al., 2000), locally linear embeddings (Roweis and Saul, 2000) or maximum variance unfolding (Weinberger et al., 2004). Alternatively, a linear generative approach such as probabilistic principal component analysis (Tipping and Bishop, 1999) can be used. However, for the aforementioned factorized model no such analogous approach exists and, instead, an algorithm or heuristic has to be developed for initialization. Ek et al. (2008a) used CCA to initialize the shared representation and a constrained version of PCA, referred to as non-consolidating component analysis (NCCA), to initialize the private space. However, as stated in the introduction, the objective of CCA is to maximize correlation, which is quite different objective from that of a shared GP-LVM which aims to reconstruct the data. Another approach is to alter the model by adding regularizers (Salzmann et al., 2010). However, for that method to avoid local minima a rather involved annealing scheme was required which made the model difficult to train.

The fundamental problem of all past shared GP-LVM approaches was that the dimensions of the latent space were divided into either shared or private; this hard separation rendered challenging the “transfer” of variations (information) between the different spaces during optimization. Furthermore, due to intractability issues the previous approaches were unable to communicate uncertainty throughout all nodes of the model, something which resulted in poorly regularized models.

In this paper we aim at expressing the shared GP-LVM as a IBFA model and, importantly, embedding it in our derived fully Bayesian framework. Our model then enjoys the benefits of the popular IBFA formulation while using powerful probabilistic and nonlinear mappings with simultaneous regularization. In particular, the Bayesian formalism solves many of the regularization problems mentioned in the previous paragraphs in two ways: firstly, it allows for a relaxation of the structural factorization of the model from a hard and discrete representation, where each latent variable is exclusively associated with either a private or a shared space, to a smooth and continuous one. Secondly, the Bayesian formulation that we will present allows for robust model training without having to rely on ad-hoc regularizers; instead, a principled variational learning scheme is used for automatic and simultaneous estimation of both, the dimensionality and the structure of the latent representation. Notice that this Bayesian framework also provides better handling of the uncertainty through an approximation to the full posterior of the latent points given the data, something which is desirable for preventing overfitting or further extending the model in more complicated scenarios, for example deep architectures (Damianou and Lawrence, 2013; Damianou, 2015). Further, as we will show, it allows for including priors on the latent space such as dynamic models (Damianou et al., 2011).

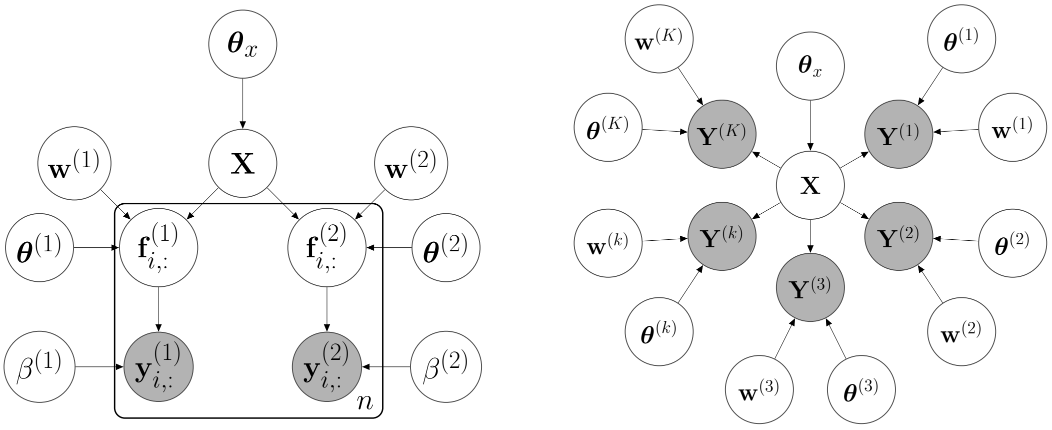


Figure 4: Manifold relevance determination (MRD). Left: the graphical model for two views, depicting the dependencies of all variables before integrating out the latent mapping. Each observation space $\mathbf{Y}^{(k)} \in \mathbb{R}^{p_k}$ depends on a common latent space $\mathbf{X} \in \mathbb{R}^{q}$. A weight vector $\mathbf{w}^{(k)} \in \mathbb{R}^{q}$ associates the relative relevance of each dimension in the latent space for generating the view. $\boldsymbol{\theta}^{(k)}$ denotes a set of additional kernel parameters that control the prior over the generative mapping. Right: extension to K views, shown after marginalizing the latent mappings and dropping dependence on the noise precision $\beta^{(k)}$. Despite the apparent similarity with the graphical model of Figure 3(b) which depicts a fully shared latent space, here the role of $\mathbf{X}$ is completely different; $\mathbf{X}$ is marginalized out and, together with the additional weight parameters, operates in a Bayesian factorized model.

4. Bayesian Nonparametric Nonlinear IBFA

A central motivating idea behind the model we propose is to relax the “hard” (discrete) factorization into shared and private latent subspaces and instead use learn a “soft” (continuous) factorization. This is achieved by allowing the different views to choose the importance, if any, of each latent dimension from a *single* latent space, while employing a continuous scale to measure the level of importance. This selection is done jointly with learning a posterior distribution over the latent space and as we will show this will let the shared and private dimensions to naturally emerge during optimization. We refer to this idea as *manifold relevance determination (MRD)*.

A subtle but important difference wth past models is that, in MRD, there is only a single latent variable $\mathbf{X}$, from where we can still extract a private latent space, e.g. $\mathbf{X}^{(k)}$, or a shared space, e.g. $\mathbf{X}^{(k,l)}$, by chosing different sets of dimensions from $\mathbf{X}$. Contrast this with shared GP-LVM approaches which explicitly used different random variables to represent the private and shared space, thereby assuming a static structure a priori. Using MRD, the model is allowed to (and indeed often does) completely allocate a latent dimension to private or shared spaces, but may also choose to endow a shared latent dimension with more or less relevance for a particular view. Importantly, this factorization is learned from data by maximizing a variational lower bound on the model evidence, rather than through construction of bespoken regularizers to achieve a similar effect. MRD can be seen as a natural generalization of the traditional approach to manifold learning; we still assume the existence of a low-dimensional representation encoding the underlying phenomenon, but the variance contained in an observation space does not necessarily need to be governed by the full manifold or by a submanifold that has been selected in an ad-hoc way.

4.1 Latent Factorization as Feature Selection

MRD circumvents the challenges associated with the shared GP-LVM by relaxing the discrete factorization of the latent space to obtain a continuous one. To achieve this, we rephrase the factorized latent variable modeling in terms of a feature selection problem. Two ingredients are necessary for solving this feature selection problem: firstly, a means of incorporating continuous scale parameters factorizing the latent space and, secondly, a Bayesian training framework by which the scale parameters are determined, exploiting the principle of Bayesian shrinkage (Tipping and Bishop, 1999). We elaborate on these two ingredients below.

To start with, we adopt the automatic relevance determination (ARD) idea (Neal, 1996) which is a popular way of performing automatic feature selection (Neal, 1996) in Bayesian settings. In the Gaussian process framework, ARD can be implemented by introducing a weight parameter w_j which is associated by way of multiplication with the corresponding dimension $\mathbf{x}_{:,j}$ of the input space. The weight w_j is then learned such that it scales $\mathbf{x}_{:,j}$ according to the importance of the $j-$th dimension for predicting the output. The introduction of the ARD vector $\mathbf{w}$ to the exponentiated quadratic (EQ) covariance function of equation (3), leads to the expression of the EQ-ARD covariance function:

$$k_{\mathrm{EQ}}(\mathbf{x}_{i,:}, \mathbf{x}_{r,:}) = \sigma^2_{\mathrm{EQ}} \exp\left(-\frac{1}{2}\sum_{j=1}^{q} w_j (x_{i,j} - x_{r,j})^2\right).$$

In this equation the weight w_j encodes the relative relevance of dimension q in determining the co-variance between the ith and the rth data point. A similar effect is achieved by introducing ARD to some other covariance function, such as the linear one. From the perspective of a shared GP-LVM, the above observation motivates the idea of using a *single* latent space while considering an independent ARD covariance function for each view. Using a different weight set $\mathbf{w}^{(k)}$ for each view k will allow the model to automatically infer the responsibility of each latent dimension for generating points in each view, as can be seen in Figure 4. We collectively denote the set of additional kernel hyperparameters as $\mathbf{W} = \{\mathbf{w}^{(k)}\}_{k=1}^{K}$, and refer to them as “weights” (rather than squared inverse length-scales) to highlight their distinct role in our model.

While incorporating the ARD idea in the shared GP-LVM setting might seem straightforward, it is important to note that the addition of $\mathbf{W}$ alone cannot have the same shrinkage/feature selection effect as observed in standard GPs. This is because, in the shared GP-LVM case, the inputs to the covariance function are latent, and treating both $\mathbf{W}$ and $\mathbf{X}$ as parameters can lead to severe overfitting, as has been demonstrated by Damianou et al. (2016). Therefore, in contrast to the traditional shared GP-LVM approaches, in MRD we aim for a fully Bayesian training framework where the latent space is treated as a distribution, so that the effects of Bayesian shrinkage can be realized (see e.g. Tipping, 2000; Bishop, 1999). The Bayesian nonparametric approach induces an intricate interplay between latent locations and ARD weights through the marginal likelihood optimization objective. This allows the Bayesian training procedure to "switch off" unnecessary latent dimensions through the corresponding ARD weight, without having to include explicit weight regularizers. Unfortunately, treating the latent space as a distribution is intractable in our case because our nonlinear IBFA approach needs to employ nonlinear covariance functions, through which distributions on the latent space cannot be propagated analytically. Standard variational approaches also fail to cope with this intractability. In this work, we solve this issue by adopting recent work on variational propagation of uncertainty in Gaussian processes (Titsias and Lawrence, 2010; Damianou et al., 2016). The resulting framework is explained in the next section.

4.2 Bayesian Training

The traditional maximum likelihood training of GPs is challenging, due to the existence of multiple local minima. As described in the previous section, learning the factorized model poses an even greater challenge and incorporating additional parameters by introducing ARD kernels is only going to make training harder. To solve the latent factorization learning and ARD parameter determination at the same time we wish to use the data evidence as an optimization objective within a Bayesian framework. Specifically, the evidence is obtained after placing a prior distribution on the latent space, $\mathbf{X}$, and marginalizing it out. Then, equation (5) is transformed into:

$$p(\mathcal{Y}|\mathbf{W},\boldsymbol{\Theta},\boldsymbol{\theta}_x) = \int_{\mathcal{F}} \left(p(\mathcal{Y}|\mathcal{F}) \int_{\mathbf{X}} p(\mathcal{F}|\mathbf{X},\mathbf{W},\boldsymbol{\Theta}) p(\mathbf{X}|\boldsymbol{\theta}_x) \right), \tag{6}$$

where the expressions with calligraphic notation for all views are expanded into a product as in equation (5). A simple choice for the latent space prior is a standard normal, $p(\mathbf{X}) \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$, but in Section 4.3 we will investigate more structured choices. We are also allowed to define further priors on the parameters $\mathbf{W}$, $\boldsymbol{\Theta}$ and $\boldsymbol{\theta}_x$. For the remainder of this paper we will, for clarity, drop the dependency on these parameters from our expressions. The Bayesian training procedure allows for an automatic Occam's razor during which the dimensions of each weight vector $\mathbf{w}^{(k)}$ are switched off if needed; this would be impossible if $\mathbf{X}$ was not marginalized, since the likelihood would typically increase by allowing a larger latent space. However, the above integral is intractable, since the factors in $p(\mathcal{F}|\mathbf{X}) = \prod_{k=1}^{K} p(\mathbf{F}^{(k)}|\mathbf{X})$ contain $\mathbf{X}$ nonlinearly. Even standard variational approaches fail (Damianou, 2015). Such methods attempt to find a variational distribution $\mathcal{Q}(\mathcal{F},\mathbf{X})$ which best approximates the true posterior of the integrands through minimizing the KL divergence,

$$\mathrm{KL}\left(\mathcal{Q}(\mathcal{F},\mathbf{X}) \,\|\, p(\mathcal{F},\mathbf{X}|\mathcal{Y})\right) = \log p(\mathcal{Y}) - \mathcal{L}(\mathcal{Q}(\mathcal{F},\mathbf{X})), \tag{7}$$

$$\text{where} \quad \mathcal{L}(\mathcal{Q}(\mathcal{F},\mathbf{X})) = \int_{\mathcal{F},\mathbf{X}} \mathcal{Q}(\mathcal{F},\mathbf{X}) \log \frac{p(\mathcal{Y}|\mathcal{F}) p(\mathcal{F}|\mathbf{X}) p(\mathbf{X})}{\mathcal{Q}(\mathcal{F},\mathbf{X})}. \tag{8}$$

By rearranging the terms in equation (7) we see that minimizing the KL-divergence between the true and the approximate posterior is equivalent to maximizing the lower bound $\mathcal{L}(\mathcal{Q}(\mathcal{F},\mathbf{X}))$ on the model evidence. Indeed, since the KL term is non-negative, we can drop it to form the inequality

$$\log p(\mathcal{Y}) \geq \mathcal{L}(\mathcal{Q}(\mathcal{F},\mathbf{X})) \tag{9}$$

revealing an alternative optimization objective: maximizing the functional $\mathcal{L}$ with respect to the variational distribution $\mathcal{Q}$ and the (hyper)parameters is equivalent to maximizing the model evidence with respect to the model (hyper)parameters[3]. Having made this observation, we will henceforth further simplify our notation by writing $\mathcal{L}(\mathcal{F}, \mathbf{X})$ instead of $\mathcal{L}(\mathcal{Q}(\mathcal{F}, \mathbf{X}))$.

However, the above standard variational approach is problematic, since it still requires integration of $\mathbf{X}$ through the challenging term $p(\mathcal{F}|\mathbf{X})$ which still appears when we expand the numerator of $\mathcal{L}(\mathcal{F}, \mathbf{X})$. To circumvent this problem we follow Titsias (2009); Titsias and Lawrence (2010) and apply the "data augmentation" principle, where we augment the probability space with auxiliary pseudo-inputs $\mathcal{Z} = \{\mathbf{Z}^{(k)}\}_{k=1}^K$ corresponding to output values $\mathcal{U} = \{\mathbf{U}^{(k)}\}_{k=1}^K$ drawn from the same priors as $\mathcal{F}$, that is, $p(\mathbf{U}^{(k)}|\mathbf{Z}^{(k)})$ is a distribution with the same form as its corresponding $p(\mathbf{F}^{(k)}|\mathbf{X})$. Here we consider the general case where a separate set of inducing inputs is used per modality, but another approach would be to use the same $\mathbf{Z}$ for all k. However, keeping a different $\mathbf{Z}$ per view is a more natural choice, since $\mathbf{Z}^{(k)}$ is related to $\mathbf{U}^{(k)}$ which, in turn, is related to $\mathbf{f}^{(k)}$. Since the model's construction uses separate GP mappings, the optimal solution for all $\mathbf{Z}^{(k)}$ is to not be identical.

With the addition of inducing variables, the probability of the GP prior shown in equation (2) now takes the following augmented expression for each view and dimension:

$$\begin{aligned} p(\mathbf{f}_{:,j}^{(k)}, \mathbf{u}_{:,j}^{(k)}|\mathbf{X}, \mathbf{Z}^{(k)}) &= p(\mathbf{f}_{:,j}^{(k)}|\mathbf{u}_{:,j}^{(k)}, \mathbf{X}, \mathbf{Z}^{(k)}) p(\mathbf{u}_{:,j}^{(k)}|\mathbf{Z}^{(k)}) \\ &= \mathcal{N}\left(\mathbf{f}_{:,j}^{(k)}|\mathbf{K}_{fu}^{(k)}(\mathbf{K}_{uu}^{(k)})^{-1}\mathbf{u}_{:,j}^{(k)}, \mathbf{K}_{ff}^{(k)} - \mathbf{K}_{fu}^{(k)}(\mathbf{K}_{uu}^{(k)})^{-1}\mathbf{K}_{uf}^{(k)}\right) \mathcal{N}\left(\mathbf{u}_{:,j}^{(k)}|\mathbf{0}, \mathbf{K}_{uu}^{(k)}\right), \end{aligned}$$

where $\mathbf{K}_{uu}^{(k)} = k^{(k)}(\mathbf{Z}^{(k)}, \mathbf{Z}^{(k)})$ denotes the covariance matrix obtained by evaluating the covariance function of view k on the auxiliary inputs, $\mathbf{K}_{uf}^{(k)} = k^{(k)}(\mathbf{Z}^{(k)}, \mathbf{X})$ and $\mathbf{K}_{fu}^{(k)} = k^{(k)}(\mathbf{X}, \mathbf{Z}^{(k)})$. As can be seen, the auxiliary variables are used to *compress* the latent function signal into a representation which relies on a low-rank covariance matrix (Quiñonero Candela and Rasmussen, 2005).

After expanding the probability space with auxiliary variables, equations (7) and (8) now become

$$\begin{aligned} &\mathrm{KL}\left(\mathcal{Q}(\mathcal{F}, \mathbf{X}, \mathcal{U}) \,\|\, p(\mathcal{F}, \mathbf{X}, \mathcal{U}|\mathcal{Y}, \mathcal{Z})\right) = \log p(\mathcal{Y}) - \mathcal{L}(\mathcal{F}, \mathbf{X}, \mathcal{U}, \mathcal{Z}) \\ &\text{where } \ \mathcal{L}(\mathcal{F}, \mathbf{X}, \mathcal{U}, \mathcal{Z}) = \int_{\mathcal{F}, \mathbf{X}, \mathcal{U}} \mathcal{Q}(\mathcal{F}, \mathbf{X}, \mathcal{U}) \log \frac{p(\mathcal{Y}|\mathcal{F}) p(\mathcal{F}|\mathcal{U}, \mathbf{X}, \mathcal{Z}) p(\mathcal{U}|\mathcal{Z}) p(\mathbf{X})}{\mathcal{Q}(\mathcal{F}, \mathbf{X}, \mathcal{U})}, \end{aligned} \tag{10}$$

where the numerator inside the logarithm shows the augmented joint distribution. The above integral is still intractable, since $p(\mathcal{F}|\mathcal{U}, \mathbf{X}, \mathcal{Z})$ again contains $\mathbf{X}$ nonlinearly. However, we are now able to remove this term from the log. after first compressing it in a way that its contribution is manifested through the set of auxiliary variables. To achieve this in a principled way, we follow a special mean-field methodology by defining a variational

[3] Recall that the model (hyper)parameters appear in the model evidence of equation (6) and similarly in equation (8) for $\mathcal{L}$, but were dropped for clarity from the expressions.

distribution which factorizes as,

$$\begin{aligned}\mathcal{Q}(\mathcal{F}, \mathbf{X}, \mathcal{U}) &= p(\mathcal{F}|\mathcal{U}, \mathbf{X}, \mathcal{Z})q(\mathcal{U})q(\mathbf{X}) \qquad (11)\\ &= q(\mathbf{X}) \prod_k \underbrace{\prod_{j=1}^{p_k} p(\mathbf{f}_{:,j}^{(k)}|\mathbf{u}_{:,j}^{(k)}, \mathbf{X}, \mathbf{Z}^{(k)})q(\mathbf{u}_{:,j}^{(k)})}_{\mathcal{Q}^{(k)}} .\end{aligned}$$

While the forms of each individual factor will be defined later on, for the moment we notice that the above factorization causes the collection of challenging terms $p(\mathcal{F}|\mathcal{U}, \mathbf{X}, \mathcal{Z})$ to cancel out inside the logarithm of $\mathcal{L}(\mathcal{F}, \mathbf{X}, \mathcal{U}, \mathcal{Z})$ in equation (10), leaving us with a tractable expression for the bound. By keeping a variational distribution for $q(\mathcal{U})$, the framework encourages compression of the latent function signal into the auxiliary variables (Titsias, 2009; Hensman and Lawrence, 2014). After replacing equation (11) into the bound (10) and cancelling the aforementioned terms inside the log, we expand the resulting expression by separating the integrals, so that the variational bound becomes:

$$\mathcal{L}(\mathcal{F}, \mathbf{X}, \mathcal{U}, \mathcal{Z}) = \mathbb{E}_{\mathcal{Q}}[\log p(\mathcal{Y}|\mathcal{F})] - \mathrm{KL}\left(q(\mathcal{U}) \,\|\, p(\mathcal{U}|\mathcal{Z})\right) - \mathrm{KL}\left(q(\mathbf{X}) \,\|\, p(\mathbf{X})\right), \qquad (12)$$

where $\mathbb{E}_{\mathcal{Q}}[p(\cdot)]$ is the expectation of $p(\cdot)$ under $\mathcal{Q} = \mathcal{Q}(\mathcal{F}, \mathbf{X}, \mathcal{U})$. As per equation (9), the expression (12) constitutes our final objective to be maximized. The obtained form also reveals the availability of the posterior marginal $q(\mathbf{X})$, constituting an approximation to the true latent space posterior $p(\mathbf{X}|\mathcal{Y})$. This approximate posterior appears in the term $-\mathrm{KL}\left(q(\mathbf{X}) \,\|\, p(\mathbf{X})\right)$ of our variational objective. This term acts as a Bayesian regularizer, penalizing unnecessarily complex posteriors.

We have so far left unspecified the forms of the variational distributions appearing in $\mathcal{Q}$, in equation (11). We choose these forms so that the variational objective (12) remains tractable. Specifically, we choose $q(\mathbf{X})$ to be a factorized distribution:

$$q(\mathbf{X}) = \prod_{i=1}^{n} \mathcal{N}(\mathbf{x}_{i,:}|\boldsymbol{\mu}_{i,:}, \mathbf{S}_{i,:}), \qquad (13)$$

where the parameters of each Gaussian are to be learned through the variational optimization framework. Notice that the dimensions of each $\mathbf{x}_{i,:}$ are uncorrelated, so that $\mathbf{S}_{i,:}$ is a diagonal matrix. As for $q(\mathcal{U}) = q(\{\mathbf{U}^{(k)}\}_{k=1}^{K})$, one option which maintains tractability of equation (12) is to allow each $q(\mathbf{U}^{(k)})$ to be a Gaussian distribution the parameters of which need to be learned. Another option is to follow (Titsias and Lawrence, 2010) and replace each $q(\mathbf{U}^{(k)})$ with its optimal form, found by differentiating the objective (12) with respect to $q(\mathbf{U}^{(k)})$, setting the expression to zero and solving for $q(\mathbf{U}^{(k)})$. The latter approach is taken in our work. Specifically, the optimal distribution for each $q(\mathbf{U}^{(k)})$ depends on all the terms which interact with $q(\mathbf{U}^{(k)})$ in the objective, that is, $\mathbb{E}_{\mathcal{Q}^{(k)}}[p(\mathbf{Y}^{(k)}|\mathbf{F}^{(k)})]$, $p(\mathbf{U}^{(k)}|\mathbf{Z}^{(k)})$ and $q(\mathbf{X})$. Further details for this derivation are given in Appendix A. The result of the above procedure, referred to as optimally "collapsing" the variational distribution $q(\mathcal{U})$, is a variational lower bound which does no longer depend on $\mathcal{U}$ and is tighter than the previous bound:

$$\mathcal{L}(\mathcal{F}, \mathbf{X}, \mathcal{Z}) = -\mathrm{KL}\left(q(\mathbf{X}) \,\|\, p(\mathbf{X})\right) + \sum_k \mathcal{L}^{(k)} \geq \mathcal{L}(\mathcal{F}, \mathcal{U}, \mathbf{X}, \mathcal{Z}), \qquad (14)$$

where the term $\mathcal{L}^{(k)}$ is the collapsed version of the terms

$$\mathbb{E}_{\mathcal{Q}}[\log p(\mathbf{Y}^{(k)}|\mathbf{F}^{(k)})] - \text{KL}\left(q(\mathbf{U}^{(k)}) \parallel p(\mathbf{U}^{(k)}|\mathbf{Z}^{(k)})\right)$$

appearing when we expand equation (12).

To summarize, the optimization procedure uses equation (14) as an objective and it uses a gradient based method to jointly optimize the following parameters:

- *Variational parameters*: the $2(n \times q)$ parameters $\{\boldsymbol{\mu}_{i,:}, \mathbf{S}_{i,:}\}_{i=1}^{n}$ of $q(\mathbf{X})$; the $m \times p_k$ matrix of inducing inputs $\mathbf{Z}^{(k)}$ for each view k.
- *Model parameters*: the Gaussian noise variances $\{\beta^{(k)}\}_{k=1}^{K}$
- *Hyperparameters*: the kernel parameters $\boldsymbol{\Theta}$ and the kernel relevance weights $\mathbf{W}$.

Notice that, in contrast to a non-Bayesian GP-LVM model, the choice of q is not a crucial hyperparameter. Indeed, non-Bayesian GP-LVMs lack the Bayesian automatic capacity control mechanism and tend to use all the dimensions of $\mathbf{X}$. Therefore, the practitioner-chosen parameter q affects the properties of the generated representation. On the other hand, MRD is a Bayesian model whose built-in Occam's razor will find an "effective" number of dimensions $q_{\text{effective}} \leq q$, i.e. the number of latent dimensions that are not "switched-off" by the ARD weights. The concept of effective dimensionality is first discussed by Bishop (1999) for Bayesian PCA, which is a linear, single-view special case of MRD.

In principle, $q_{\text{effective}}$ can be found correctly irrespective of our choice for q. In practice, there are two issues that one has to keep in mind. Firstly, if q is extremely small it might harm the capacity of the model. Secondly and conversely, if q is too big it might cause convergence issues (and it will make optimization slower). For our experiments with two views we typically use 10 to 15 dimensions (or we set $q = d$ if $d < 10$ where d is the sum of the dimensionalities of all output views). This is only a rule of thumb coming from our experience of working with both the Bayesian GP-LVM and MRD. We have never seen any of the two models (when using non-linear kernels) select an effective dimension of more than 15 per view, which is attributed to the powerful non-linear mappings and the relatively small data sets that are used with these models. Therefore, we have found that setting q to around 15 is a safe choice. A more thorough approach is to perform experiments on a validation set with a sparse grid of candidates for q, i.e. $[5, 10, 15, 20]$.

4.3 Factor Constraints Through Priors

In the previous section we showed that the MRD framework can approximately integrate out the latent space $\mathbf{X}$ and maximize the logarithm of the evidence $p(\mathcal{Y})$. In addition to the previously discussed benefits of MRD as an inference engine for nonlinear, nonparametric IBFA, notice that MRD also allows principled incorporation of additional priors over the latent space. Such priors express our preference for specific properties of IBFA's factors and provide disambiguation at test time (when different factors compete to explain test data). The ability to incorporate priors in a principled way stems from the Bayesian nature of MRD, as can be seen in equation (6). As an important example of this, we will now describe how a latent prior that respects the temporal dynamics of the data can be incorporated into the MRD model.

Many types of data have an inherent dynamical structure. When learning a latent representation of such data there are several benefits to encourage the representation to respect this dynamic. An example is when we wish to synthesize novel data by either inter- or extrapolating a sequence. For the standard GP-LVM model, an autoregressive (Markovian) prior was suggested by Wang et al. (2008), while Damianou et al. (2011) proposed a regressive (temporal) prior for the fully Bayesian model. The autoregressive structure proposed by Wang et al. was introduced to the shared GP-LVM by Ek et al. (2008a, 2007), where the dynamics were exploited to disambiguate sequences of human motion in order to perform human pose estimation. In the experimental section of the paper we will reproduce the experiments performed by Ek et al., this time using the MRD approach and the regressive dynamics framework.

Here we follow Damianou et al. (2011) and use a temporal, Gaussian process prior to model the dynamics. Using a nonparametric prior constitutes a flexible solution, in line with the Bayesian formulation of MRD. Given the work of Damianou et al. (2011), it is straightforward to include such prior in the MRD framework. Nevertheless, we will reiterate the corresponding derivations here as this is instructive about how other kinds of latent space priors can be developed.

We start by reformulating the latent space as q independent latent *functions* $\{\mathbf{x}_{:,j}\}_{j=1}^{q}$. To encode the sequential structure of each $n-$dimensional latent function, we introduce correlation through the $n-$dimensional vector of time-stamps $\mathbf{t} = [t_1, \dots, t_n]^\top, t_i \in \Re$, which we assume are given together with our outputs $\mathcal{Y}$. Each element t_i represents the time at which the $i-$th collection of corresponding view-instances $\{\mathbf{y}_{i,:}^{(k)}\}_{k=1}^{K}$ was observed. This kind of latent coupling reflects the temporal correlation between instances of each view $\mathbf{Y}^{(k)}$, and we assume that the same dynamics are respected by all views. Then, we have:

$$x_{:,j}(t) \sim \mathcal{GP}(0, k_x(t', t'')),$$

where k_x is the temporal covariance function. The joint probability density of the model which is augmented with inducing points and time-stamps takes the form,

$$p(\mathcal{Y}, \mathcal{F}, \mathcal{U}, \mathbf{X}|\mathcal{Z}, \mathbf{t}) = p(\mathcal{Y}|\mathcal{F})p(\mathcal{F}|\mathcal{U}, \mathbf{X})p(\mathcal{U}|\mathcal{Z})p(\mathbf{X}|\mathbf{t}),$$

where $p(\mathbf{X}|\mathbf{t})$ is a product of q independent Gaussian distributions (due to the GP priors employed for the latent space):

$$p(\mathbf{X}|\mathbf{t}) = \prod_{j=1}^{q} \mathcal{N}\left(\mathbf{x}_{:,j}|\mathbf{0}, \mathbf{K}_t\right), \quad \text{where:} \quad \mathbf{K}_t = k_x(\mathbf{t}, \mathbf{t}),$$

i.e. $\mathbf{K}_t$ is a covariance matrix constructed through k_x with training time-stamps as inputs. The derivations for the variational framework of this temporal model follow as before. Specifically, from equation (12) we see that $p(\mathbf{X})$ only appears in the last KL term; the rest of the terms do not get affected by the dynamics directly, but only indirectly through $q(\mathbf{X})$. Therefore, the variational lower bound for the dynamical MRD is:

$$\mathcal{L}(\mathcal{F}, \mathbf{X}, \mathcal{U}, \mathcal{Z}) = \mathbb{E}_{\mathcal{Q}}[\log p(\mathcal{Y}|\mathcal{F})] - \text{KL}\left(q(\mathcal{U}) \parallel p(\mathcal{U}|\mathcal{Z})\right) - \text{KL}\left(q(\mathbf{X}) \parallel p(\mathbf{X}|\mathbf{t})\right).$$

Although the above form is very similar to the non-dynamical variational bound of equation (12), here the framework employs a point-wise latent space coupling which is reflected in the approximate posterior by choosing a coupled form for $q(\mathbf{X})$. To maintain tractability we choose $q(\mathbf{X})$ to be a Gaussian distribution, as before. However, in contrast to equation (13), we now factorize the variational distribution according to dimensions, following Damianou et al. (2011). In this way, the latent points remain a posteriori coupled through the dynamics, that is

$$q(\mathbf{X}) = \prod_{j=1}^{q} \mathcal{N}(\mathbf{x}_{:,j} | \boldsymbol{\mu}_{:,j}, \mathbf{S}_{:,j}),$$

where $\mathbf{S}_{:,j}$ is a full, $n \times n$ covariance matrix (in contrast to the diagonal matrix assumed in equation (13)). Notice that we now have q, $n \times n$ parameters to learn for all $\{\mathbf{S}_{:,j}\}_{j=1}^{q}$, however reparameterization tricks to reduce this number exist (Opper and Archambeau, 2009; Damianou et al., 2011).

With this dynamical approach, we are also allowed to learn the structure of multiple statistically independent sequences which, nevertheless, share the same dynamical structure (e.g. multiple strands of different people walking). Each sequence is represented by a collection of multi-view instances and corresponding time-stamps. Sequence learning through MRD is achieved by learning a common latent space for all latent timeseries while, at the same time, ignoring correlations between latent points which correspond to outputs of different sequences. To add this functionality to our framework, we just need to define the temporal covariance matrix $\mathbf{K}_t$ to have a block-diagonal structure by setting $k_x(t, t') = 0$ if t and t' belong to different sequences. In the experimental section of the paper we will evaluate our model in this setting.

4.4 Latent Space Post-hoc Analysis

As will be discussed in the next section, the training and inference within MRD can be realized within the framework of "soft" latent space factorization. However, in certain cases we might need an explicit "hard" segmentation for posterior tasks. For example, consider the case where we wish to use the shared latent space discovered from very high-dimensional views as features for a subsequent classification task. In this case, we will need to firstly decide on which latent dimensions we consider to be shared and which private. Another example is the scenario where we simply need to perform high-level reasoning about correlations discovered between the different views. Therefore, we wish to split the common latent space $\mathbf{X}$ into subspaces $[\mathbf{X}^{\mathcal{A},\mathcal{B}}, \mathbf{X}^{\mathcal{A}}, \mathbf{X}^{\mathcal{B}}, \mathbf{X}^{\setminus\{\mathcal{A},\mathcal{B}\}}]$, where $\mathbf{X}^{\mathcal{A},\mathcal{B}}$ denotes the subspace shared for view-sets[4] $\mathcal{A}$ and $\mathcal{B}$; $\mathbf{X}^{\mathcal{A}}$ denotes the subspace which is private for $\mathcal{A}$ (and analogously for $\mathbf{X}^{\mathcal{B}}$); $\mathbf{X}^{\setminus\{\mathcal{A},\mathcal{B}\}}$ denotes a subspace which is irrelevant for both $\mathcal{A}$ and $\mathcal{B}$. To achieve this segmentation, we compare the relative value of the ARD weights for each view and per dimension of $\mathbf{X}$. To have a comparison criterion which is consistent across views, we can first normalize all weight sets $\mathbf{w}^{(k)} = [w_1^{(k)}, \cdots, w_q^{(k)}], \forall k \in \{\mathcal{A}, \mathcal{B}\}$ to be in the same range. Without loss of generality let us assume that for each k we create a weight set $\tilde{\mathbf{w}}^{(k)}$ which is a normalized version of $\mathbf{w}^{(k)}$ such that the maximum element per weight-vector is 1. Then, dimension j is deemed as shared between views r, l if $\tilde{w}_{:,j}^{(r)}$ and $\tilde{w}_{:,j}^{(l)}$

[4] Latent space segmentation for multiple view-sets follows trivially in the same way.

are larger than a small threshold value ε, since $\tilde{w}_{:,j}^{(r)}$ is related to the amount of variance in view r that is explained by latent dimension j. We will explain this concept mathematically below, but first let us introduce the notation $\mathbf{X}_{:,\mathcal{J}}, \mathcal{J} \subseteq \{1, 2, \cdots, q\}$ to mean a collection of columns (dimensions) of $\mathbf{X}$, that is, a subspace of the common latent space. Then, the segmentation of the latent space is defined as follows:

$$\begin{aligned}
&\forall j \in \{1, \ldots, q\},\ a \in \mathcal{A},\ b \in \mathcal{B} : \\
\mathbf{X}^{\mathcal{A},\mathcal{B}} &= \mathbf{X}_{:,\mathcal{J}}, \text{ for } \mathcal{J} = \{j | \tilde{w}_j^{(a)} \geq \varepsilon,\ \tilde{w}_j^{(b)} \geq \varepsilon\}, \\
\mathbf{X}^{\mathcal{A}} &= \mathbf{X}_{:,\mathcal{J}}, \text{ for } \mathcal{J} = \{j | \tilde{w}_j^{(a)} \geq \varepsilon,\ \tilde{w}_j^{(b)} < \varepsilon\}, \\
\mathbf{X}^{\mathcal{B}} &= \mathbf{X}_{:,\mathcal{J}}, \text{ for } \mathcal{J} = \{j | \tilde{w}_j^{(a)} < \varepsilon,\ \tilde{w}_j^{(b)} \geq \varepsilon\}, \\
\mathbf{X}^{\setminus\{\mathcal{A},\mathcal{B}\}} &= \mathbf{X}_{:,\mathcal{J}}, \text{ for } \mathcal{J} = \{j | \tilde{w}_j^{(a)} < \varepsilon,\ \tilde{w}_j^{(b)} < \varepsilon\}.
\end{aligned}$$

It is worth emphasizing that, in practice, we found that during optimization the model is capable of performing the above truncation automatically. Specifically, the strong Bayesian regularization drives weights for unnecessary (per view) dimensions to values which are practically zero, considering machine precision limits, giving a clear separation with regards to relevant dimensions. Therefore, we can safely set the threshold ε to a small number without the need to represent it as a parameter. In our experiments we use $\varepsilon = 10^{-3}$.

4.5 Predictions

A central motivation behind the IBFA model is that it provides a natural latent structure for efficient and intuitive inference, especially in scenarios where the estimation task is ambiguous. We will now proceed to describe how we can infer outputs in a subset of views, denoted by $\mathcal{A}$, given information about outputs in a different subset of views, denoted by $\mathcal{B}$. The MRD framework renders this task easy because, even if the views live in different, incomparable spaces, they are linked through a common latent space. View-specific weights automatically define latent subspaces by which subsets of views are related.

Our task at test time is to generate a new (or infer a training) set of outputs $\mathbf{Y}_*^{\mathcal{B}}$ given a set of (potentially partially) observed test points $\mathbf{Y}_*^{\mathcal{A}}$. The inference proceeds by predicting (the distribution over) the set of latent points $\mathbf{X}_* \in \mathbb{R}^{n_* \times q}$ which is most likely to have generated $\mathbf{Y}_*^{\mathcal{A}}$. For this, we use the approximate posterior $q(\mathbf{X}, \mathbf{X}_*) \approx p(\mathbf{X}, \mathbf{X}_* | \mathbf{Y}^{\mathcal{A}}, \mathbf{Y}_*^{\mathcal{A}})$. This approximate posterior is found by optimizing a variational lower bound on the marginal likelihood $p(\mathbf{Y}^{\mathcal{A}}, \mathbf{Y}_*^{\mathcal{A}})$. This bound has analogous form with the training objective function of equation (12). That is, to find $q(\mathbf{X}, \mathbf{X}_*)$ we use equation (12) but now the data is the augmented set $(\mathbf{Y}^{\mathcal{A}}, \mathbf{Y}_*^{\mathcal{A}})$ and the latent space is $(\mathbf{X}, \mathbf{X}_*)$. We assume that $q(\mathbf{X}, \mathbf{X}_*) = q(\mathbf{X})q(\mathbf{X}_*)$ and since $q(\mathbf{X})$ is already obtained from the training phase, the expression for the augmented variational bound can be decomposed to statistics that have been computed at training time (and are held fixed during test time) and to statistics which are inferred during test time (Titsias and Lawrence, 2010; Damianou et al., 2016), see Appendix B for details. It is also worth noting that the test posterior is fully factorized, i.e. $q(\mathbf{X}_*) = \prod_{i=1}^{n_*} q(\mathbf{x}_{i,*})$, so that the test computations can be made in parallel. After finding $q(\mathbf{X}, \mathbf{X}_*)$, we can then find a distribution of the outputs $\mathbf{Y}_*^{\mathcal{B}}$ by taking the expectation of the likelihood $p(\mathbf{Y}^{\mathcal{B}} | \mathbf{X})$

under the marginal $q(\mathbf{X}_*)$:

$$\begin{aligned}
p(\mathbf{Y}_*^{\mathcal{B}}) &\approx \int_{\mathbf{F}_*^{\mathcal{B}},\mathbf{U},\mathbf{X}_*} p(\mathbf{Y}_*^{\mathcal{B}}|\mathbf{F}_*^{\mathcal{B}}) p(\mathbf{F}_*^{\mathcal{B}}|\mathbf{U},\mathbf{X}_*) q(\mathbf{U}) q(\mathbf{X}_*) \\
&= \int_{\mathbf{X}_*} p(\mathbf{Y}_*^{\mathcal{B}}|\mathbf{X}_*) q(\mathbf{X}_*). \qquad (15)
\end{aligned}$$

The above expectation takes the form of Gaussian process prediction with uncertain inputs and is intractable. However, the predictive distribution's moments can be computed analytically following the methodology which is outlined in detail by Girard et al. (2003); Titsias and Lawrence (2010); Damianou et al. (2016). The resulting expressions are given in Appendix C. Notice that to compute this expectation we used the posterior $q(\mathbf{X}_*)$ optimized for the test data of the observed views, $\mathbf{Y}^{\mathcal{A}}$, while we re-used the posterior $q(\mathbf{U})$ estimated during training time. This is because, in contrast to the latent input variables, the auxiliary variables are used as *global* variables.

The above described way of finding the posterior $q(\mathbf{X}_*)$ based on a subset of views $\mathbf{Y}_*^{\mathcal{A}}$ makes full use of the "soft" latent space factorization which is unique to MRD: the posterior $q(\mathbf{X}_*) \approx p(\mathbf{X}_*|\mathbf{Y}_*^{\mathcal{A}})$ is found, based on which the missing instances for view-set $\mathcal{B}$ are predicted without having to decide on a "hard" latent space segmentation. Indeed, the relevance weights $\mathbf{w}^{\mathcal{A}}$ and $\mathbf{w}^{\mathcal{B}}$ are involved in the above inference and weight all outcomes automatically and consistently.

The inference procedure described above means that the latent dimensions that are (emerging as) private for $\mathbf{Y}_*^{\mathcal{B}}$ are taken directly from the prior, since the views in $\mathbf{Y}^{\mathcal{A}}$ can tell us nothing about the private information in $\mathbf{Y}^{\mathcal{B}}$. Although this works well in practice, another approach is to "force" the private latent space $\mathbf{X}^{\mathcal{B}}$ to take values that are closer to those found in the training set. For example, when the modelled data is images and the predictive task is generation of new images, we might prefer to sacrifice predictive accuracy for obtaining sharper images. For this scenario, a simple heuristic is suggested: after optimizing $q(\mathbf{X}_*)$ based on $\mathbf{Y}_*^{\mathcal{A}}$ as was described above, we perform a nearest neighbour search to find the training latent points $\tilde{\mathbf{X}}$ which are closest to the mean of $q(\mathbf{X}_*)$ in the projection to the dimensions that are shared between the views in $\mathbf{Y}^{\mathcal{A}}$ and in $\mathbf{Y}^{\mathcal{B}}$. For every test marginal $q(\mathbf{x}_*)$, we then use the nearest training point $\tilde{\mathbf{x}}$ to fill (replace) the dimensions of the mean for $q(\mathbf{x}_*)$ which have been informed only through the prior, i.e. the dimensions corresponding to $\mathbf{X}^{\mathcal{B}}$. This results in a hybrid test distribution $q(\tilde{\mathbf{X}}_*)$ which is predicted from view $\mathcal{A}$ through posterior inference but also takes training information from view $\mathcal{B}$ through the heuristic. $\mathbf{Y}_*^{\mathcal{B}}$ can then be predicted as explained in the previous paragraph, i.e. by computing the expectation of the likelihood $p(\mathbf{Y}^{\mathcal{B}}|\mathbf{X})$ under $q(\tilde{\mathbf{X}}_*)$. This procedure, summarized in Algorithm 1, is used for our experiments.

Notice that if $\mathbf{Y}_*^{\mathcal{A}} = \mathbf{Y}^{\mathcal{A}}$, then the aforementioned nearest neighbour search for finding $\tilde{\mathbf{X}}$ in essence constitutes a means of finding correspondences between the two sets of views through a simpler latent space. This case is demonstrated in Section 5.2.

5. Experiments

In this section we will show the experimental evaluation of the model proposed in this paper. We will apply the model to several different types of data, with different number of

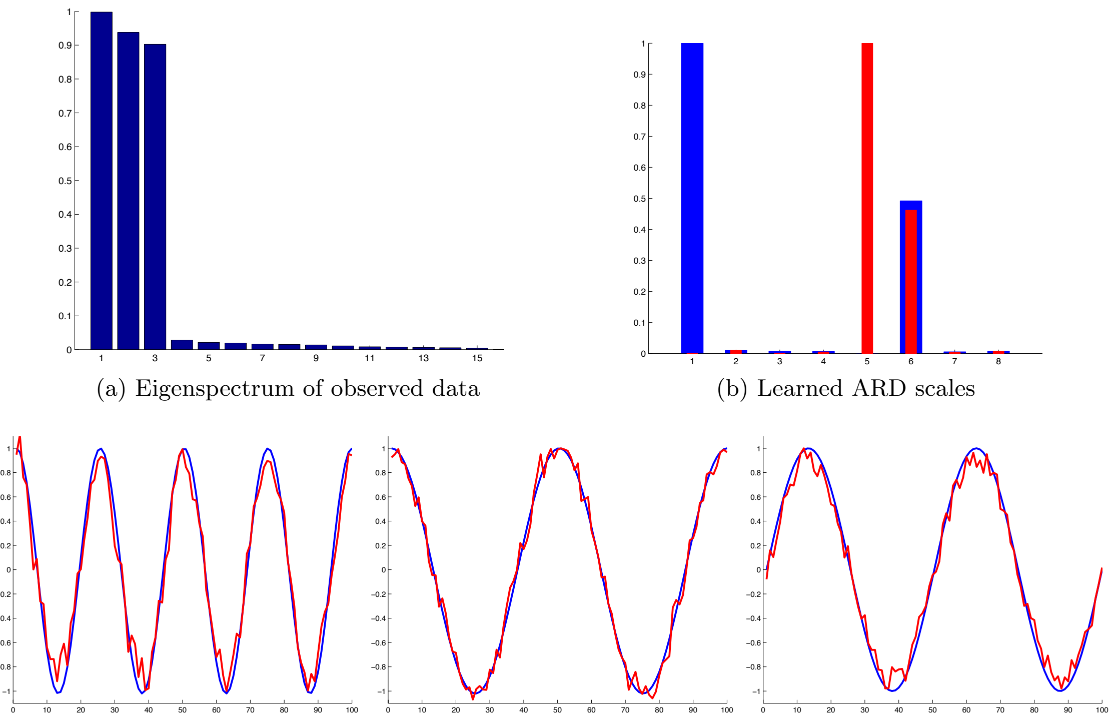
Algorithm 1 Inference algorithm in MRD, assuming two sets of views $\mathbf{Y}^{\mathcal{A}}$ and $\mathbf{Y}^{\mathcal{B}}$

```
1: Given: MRD model trained on views $(\mathbf{Y}^{\mathcal{A}}, \mathbf{Y}^{\mathcal{B}})$ to obtain $q(\mathbf{X}) \sim \mathcal{N}(\mathbf{M}, \mathbf{\Sigma})$.
2: Define the split $\mathbf{M} = (\mathbf{M}^{\mathcal{A}}, \mathbf{M}^{\mathcal{AB}}, \mathbf{M}^{\mathcal{B}})$ following Section 4.4.
3: Given: A test point $\mathbf{y}_*^{\mathcal{A}}$.
4: Optimize $q(\mathbf{X}, \mathbf{x}_*) \approx p(\mathbf{X}, \mathbf{x}_* | \mathbf{Y}^{\mathcal{A}}, \mathbf{y}_*^{\mathcal{A}})$.
5: Get the marginal $q(\mathbf{x}_*)$ with mean $\boldsymbol{\mu}_* = (\boldsymbol{\mu}_*^{\mathcal{A}}, \boldsymbol{\mu}_*^{\mathcal{AB}}, \boldsymbol{\mu}_*^{\mathcal{B}})$ and variances $\mathbf{s}_*$.
6: Find $\mathcal{K}$ points $\tilde{\boldsymbol{\mu}}_{(\kappa)}$ from a $\mathcal{K}-$nearest neigbour search between $\boldsymbol{\mu}_*^{\mathcal{AB}}$ and $\mathbf{M}^{\mathcal{AB}}$.
7: for $\kappa = 1, \cdots, \mathcal{K}$ do
8:     Generate $\tilde{\boldsymbol{\mu}}_{(\kappa),*}$ by combining the dimensions of $\tilde{\boldsymbol{\mu}}_{(\kappa)}$ and $\boldsymbol{\mu}_*$ according to sets $\mathcal{A}$
       and $\mathcal{B}$, i.e. set $\tilde{\boldsymbol{\mu}}_{(\kappa),*} = (\boldsymbol{\mu}_*^{\mathcal{A}}, \boldsymbol{\mu}_*^{\mathcal{AB}}, \tilde{\boldsymbol{\mu}}_{(\kappa)}^{\mathcal{B}})$. (Another heuristic is to explicitly consider the
       relevance weights in this step by "blending" the information coming from $\tilde{\boldsymbol{\mu}}$ and $\boldsymbol{\mu}_*$ according
       to $\mathbf{w}^{\mathcal{A}}$ and $\mathbf{w}^{\mathcal{B}}$ respectively. Then, $\tilde{\boldsymbol{\mu}}_{(\kappa)}^{\mathcal{B}}$ is replaced by a weighted average of $\boldsymbol{\mu}_*^{\mathcal{B}}$ and $\tilde{\boldsymbol{\mu}}_{(\kappa)}^{\mathcal{B}}$).
9:     Construct $q(\tilde{\mathbf{x}}_{(\kappa),*}) \sim \mathcal{N}(\tilde{\boldsymbol{\mu}}_{(\kappa),*}, \mathbf{s}_*)$.
10:    Generate $\mathbf{y}_{(\kappa),*}^{\mathcal{B}}$ from $\int_{\tilde{\mathbf{x}}_{(\kappa),*}} p(\mathbf{y}_{(\kappa),*}^{\mathcal{B}} | \tilde{\mathbf{x}}_{(\kappa),*}) q(\tilde{\mathbf{x}}_{(\kappa),*})$ using equation (15).
11: end for
12: Most likely predictions are obtained for $\kappa = 1$. We only use $\kappa > 1$ if we are interested
    in finding correspondences in the training set (see Section 5.2).
13: In the dynamical MRD version, all test points $\mathbf{Y}_*^{\mathcal{A}}$ are considered together, so that the
    variational distribution $q(\mathbf{X}, \mathbf{X}_*)$ of state 4 will form a timeseries.
```

views and associated with different tasks. We perform experiments on 6 different types of data with 2, 3, 16 and 63 number of different views where the data is parametrised both by continous and discrete variables.

5.1 Toy Data

As a first experiment we will use an intuitive toy example similar to the one proposed by Salzmann et al. (2010). We generate three separate signals: a cosine and a sine, which will be our private signal generators, and a squared cosine as shared signal. We then independently draw three separate random matrices which map the two private signals to 10 dimensions and the shared signal to 5 dimensions. The two sets of observations $\mathbf{Y}^{\mathcal{A}}$ and $\mathbf{Y}^{\mathcal{B}}$ are then formed by concatenating each respective $10-$dimensional private signal with the $5-$dimensional shared one, also adding isotropic Gaussian noise. Therefore, $\mathbf{y}_{i,:}^{\mathcal{A}}, \mathbf{y}_{i,:}^{\mathcal{B}} \in \Re^{15}$. Using a GP prior with a linear covariance function, the model should be able to learn a latent representation of the two data sets by recovering the three generating signals, a sine and a cosine as private and the squared cosine as shared. In Figure 5 the result of the experiment is shown. The model is initialized with $q = 8$ latent dimensions, of which 5 are correctly switched-off during training. Indeed, the model learns the intrinsic dimensionality of the data and, additionally, is able to recover the factorization used for generating the data. We also experimented with adding a temporal prior on the latent space. In this case we encapsulate the prior knowledge that the recovered signals should be smooth. In this



(a) Eigenspectrum of observed data

(b) Learned ARD scales

(c) Shared signal

(d) Private signal for view $\mathbf{Y}^{\mathcal{A}}$

(e) Private signal for view $\mathbf{Y}^{\mathcal{A}}$

Figure 5: MRD on a toy data set. Initialized with 8 latent dimensions the model switched off all dimensions except for three (panel (b)): one private for each observation space and one shared, which corresponds well to the eigenspectrum of the data that clearly shows three variables (panel (a)). The numbers on the $y-$axis are normalized so that the maximum is 1. The second row depicts the recovered latent signals in red and the generating signals in blue. For easier interpretation, they have been post-processed to remove the scale degree of freedom such that the learned latent space matches the generating signals. Note that no temporal information was given (i.e. all points were treated as independent). When the temporal information was given through a prior $p(\mathbf{X}|\mathbf{t})$, the recovered signals were smooth and matched almost exactly the true ones.

case the recovered signals almost exactly match the true ones and, therefore, we have not included this plot. We will therefore now proceed to apply the model to more challenging data where the generating parameters and their structure are truly unobserved.

5.2 Yale Faces

The Yale face database B (Georghiades et al., 2001) is a collection of images depicting different individuals in different poses under controlled lighting conditions. The data set contains 10 individuals in 9 different poses each lighted from 64 different directions. The different lighting directions are positions on a half sphere as can be seen in Figure 6. The

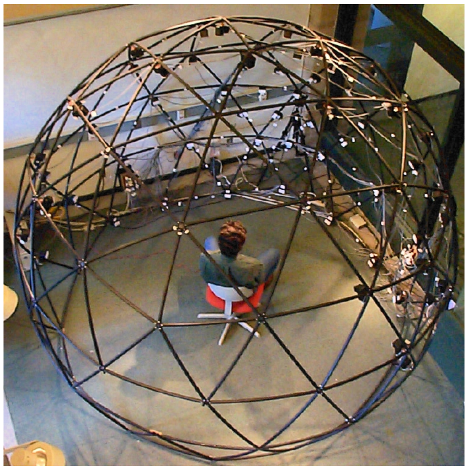
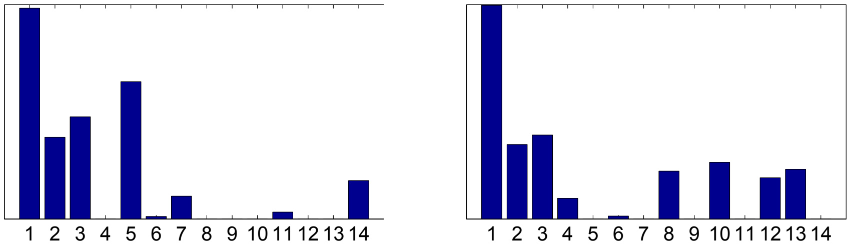


Figure 6: The mechanism used to generate the Yale Faces data set (Georghiades et al., 2001).

images for a specific pose are captured in rapid procession such that the variations in the image for a specific person and pose should mainly be associated with the light direction. This makes the data interesting from a dimensionality reduction point of view, as the representation is very high-dimensional, $192 \times 168 = 32256$ pixels, while the generating parameters, i.e. the lighting directions and pose parameters, are very low dimensional. There are several different ways of using this data in the MRD framework, depending on which correspondence aspect of the data is used to align the different views. We chose to use all illuminations for a single pose. We generate two separate data sets, $\mathbf{Y}^{\mathcal{A}}$ and $\mathbf{Y}^{\mathcal{B}}$, by splitting the images into two sets such that the two views contain three different subjects. The order of the data was such that the lighting direction of each $\mathbf{y}_{i,:}$ matched that of $\mathbf{y}^{\mathcal{B}}_{i,:}$ while the subject identity was random, such that no correspondence was induced between different faces. As such, the model should learn a latent structure factorized into lighting-related parameters (a point on a half-sphere) and subject-related parameters, where the first are shared and the latter private to each observation space.

The model is initialized with $q = 14$ latent dimensions and the optimized relevance weights $\{\mathbf{w}^{\mathcal{A}}, \mathbf{w}^{\mathcal{B}}\}$ are visualized as bar graphs in Figure 7. The latent space is clearly factorized into a shared part, consisting of dimensions indexed[5] as 1,2 and 3, two private and an irrelevant part (dimension 9). The two data views correspond to approximately equal weights for the shared latent dimensions. Projections onto these dimensions are visualized in Figures 8(a) and 8(b). Even though not immediately obvious from these two-dimensional projections, interaction with the shared latent space reveals that it actually has the structure of a half sphere, recovering the shape of the space defined by the fixed locations of the light source shown in Figure 6.

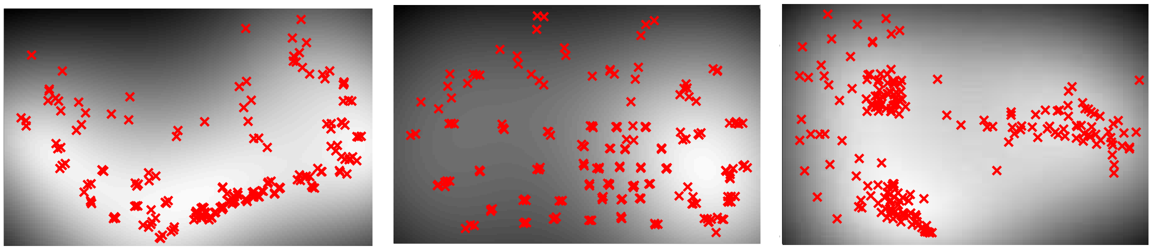
[5] Dimension 6 also encodes shared information, but of almost negligible amount ($\mathbf{w}^{\mathcal{A}}_6, \mathbf{w}^{\mathcal{B}}_6 \approx 0$).



(a) The scale vector $\mathbf{w}^{\mathcal{A}}$

(b) The scale vector $\mathbf{w}^{\mathcal{B}}$

Figure 7: The relevance weights for the faces data. Despite allowing for soft sharing, the first 3 dimensions are switched on with approximately the same weight for both views of the data. Most of the remaining dimensions are used to explain private variance.



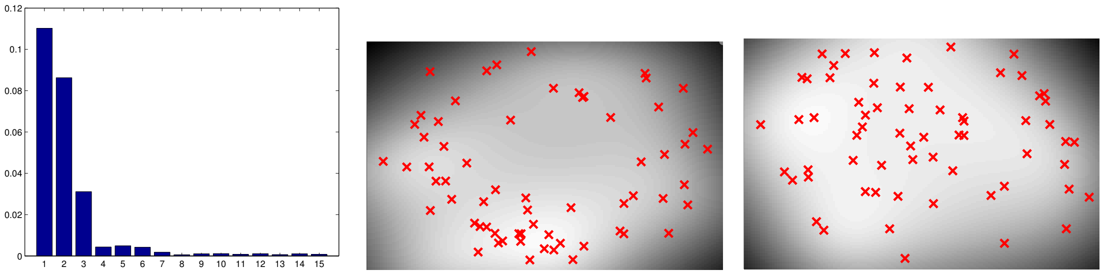
(a)

(b)

(c)

Figure 8: Projection of the shared latent space into dimensions $\{1, 2\}$ and $\{1, 3\}$ (figures (a) and (b)) and projection of the $\mathbf{Y}^{\mathcal{A}}$−private dimensions $\{5, 14\}$ (figure (c)). Red x's represent (projected) locations of latent points that correspond to the training data. The greyscale intensities of the background are proportional to the predicted variance of the GP mapping, if the corresponding locations were given as inputs. MRD successfully factorized the latent space so that the latent points in Figure (c) form three clusters, each responsible for modelling one of the three faces in $\mathbf{Y}^{\mathcal{A}}$.

By projecting the latent space onto the dimensions corresponding to the private spaces, we essentially factor out the variations generated by the light direction. As can be seen in Figure 8(c), the model then separately represents the face characteristics of each of the subjects. This indicates that the shared space successfully encodes the information about the position of the light source and not the face characteristics. This indication is enhanced by the results found when we performed dimensionality reduction with the Bayesian GP-LVM for pictures corresponding to all illumination conditions of a single face (i.e. a data set with one modality). Specifically, the latent space discovered by the Bayesian GP-LVM and the shared subspace discovered by MRD have the same dimensionality and similar



(a) (b) (c)

Figure 9: Latent space learned by the Bayesian GP-LVM for a single face view. The weight set $\mathbf{w}$ associated with the learned latent space is shown in (a). In figures (b) and (c) we plotted pairs of the 3 dominant latent dimensions against each other. Dimensions $4, 5$ and 6 have a very small but not negligible weight and represent other minor differences between pictures of the same face, as the subjects often blink, smile etc.

structure, as can be seen in Figure 9. As for the private manifolds discovered by MRD, these correspond to subspaces for disambiguating between faces of the same view. Indeed, plotting the largest two dimensions of the first latent private subspace against each other in Figure 8(c) reveals three clusters, corresponding to the three different faces within the data set. Similarly to the Bayesian GP-LVM applied to a single face, here the private dimensions with very small weight model slight changes across faces of the same subject (blinking etc).

We can also confirm visually the subspaces' properties by sampling a set of novel inputs $\mathbf{X}_{\text{samp}}$ from each subspace and then mapping back to the observed data space using the likelihoods $p(\mathbf{Y}^{\mathcal{A}}|\mathbf{X}_{\text{samp}})$ or $p(\mathbf{Y}^{\mathcal{B}}|\mathbf{X}_{\text{samp}})$, thus obtaining novel outputs (images). To better understand what kind of information is encoded in each of the dimensions of the shared or private spaces, we sampled new latent points by varying only one dimension at a time, while keeping the rest fixed. The first two rows of Figure 10 show some of the outputs obtained after sampling across each of the shared dimensions 1 and 3 respectively, which clearly encode the coordinates of the light source, whereas dimension 2 was found to model the overall brightness. The sampling procedure can intuitively be thought as a walk in the space shown in Figure 8(b) from left to right and from the bottom to the top. Although the set of learned latent inputs is discrete, the corresponding latent subspace is continuous, and we can interpolate images in new illumination conditions by sampling from areas where there are no training inputs (i.e. in between the red crosses shown in Figure 8). Similarly, we can sample from the private subspaces and obtain novel outputs which interpolate the non-shared characteristics of the involved data. This results in a morphing effect across different faces, which is shown in the last row of Figure 10. The two interpolation effects can be combined. Specifically, we can interactively obtain a set of shared dimensions corresponding to a specific lighting direction, and by fixing these dimensions we can then sample in the private dimensions, effectively obtaining interpolations between faces under the desired lighting condition. This demonstration, and the rest of the results, are illustrated in the online videos (`http://git.io/vwLhH`). As can be seen, MRD allows structured generation

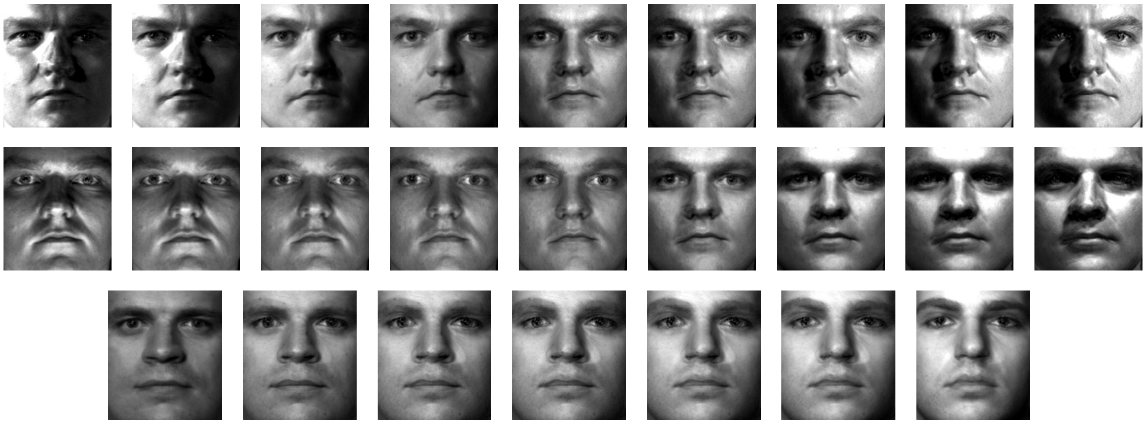


Figure 10: Sampling inputs to produce novel outputs. First row shows interpolation between positions of the light source in the x coordinate and second row in the y coordinate (elevation), obtained by structured sampling in the shared latent space. Last row shows interpolation between face characteristics to produce a morphing effect, obtained by structured sampling in the private latent space. Note that these images are presented scaled here, the original size is 192 × 198 pixels.

of novel high-dimensional outputs (images) by using low-dimensional inputs (latent points) as "controls".

As a final test, we confirm the efficient factorization of the latent space into private and shared parts by automatically recovering all the illumination similarities found in the training set. More specifically, given a data point $\mathbf{y}^{\mathcal{A}}_{i,:}$ from the first view, we search the whole space of training inputs $\mathbf{X}$ to find the 6 nearest neigbours to the latent representation $\mathbf{x}_{i,:}$ of $\mathbf{y}^{\mathcal{A}}_{i,:}$, *based only on the shared dimensions.* From these latent points, we can then obtain points in the output space of the second view, by using the likelihood $p(\mathbf{Y}^{\mathcal{B}}|\mathbf{X})$. This procedure is a special case of Algorithm 1 where the test point given is already in the training set. As can be seen in Figure 11, the model returns images with matching illumination condition. Moreover, the fact that, typically, the first neighbours of each given point correspond to outputs belonging to different faces, indicates that the shared latent space is "pure", and is not polluted by information that encodes the face appearance.

5.3 Pose Estimation and Ambiguity Modelling

For our next experiment we will use the MRD model to perform human pose estimation from silhouette data. The purpose of this experiment is to show how a factorized latent variable model can be used to perform efficient inference when the task is ambiguous. We consider a set of 3D human poses and associated silhouettes, coming from the data set of Agarwal and Triggs (2006). We used a subset of 5 sequences, totaling 649 frames, corresponding to walking motions in various directions and patterns. A separate walking sequence of

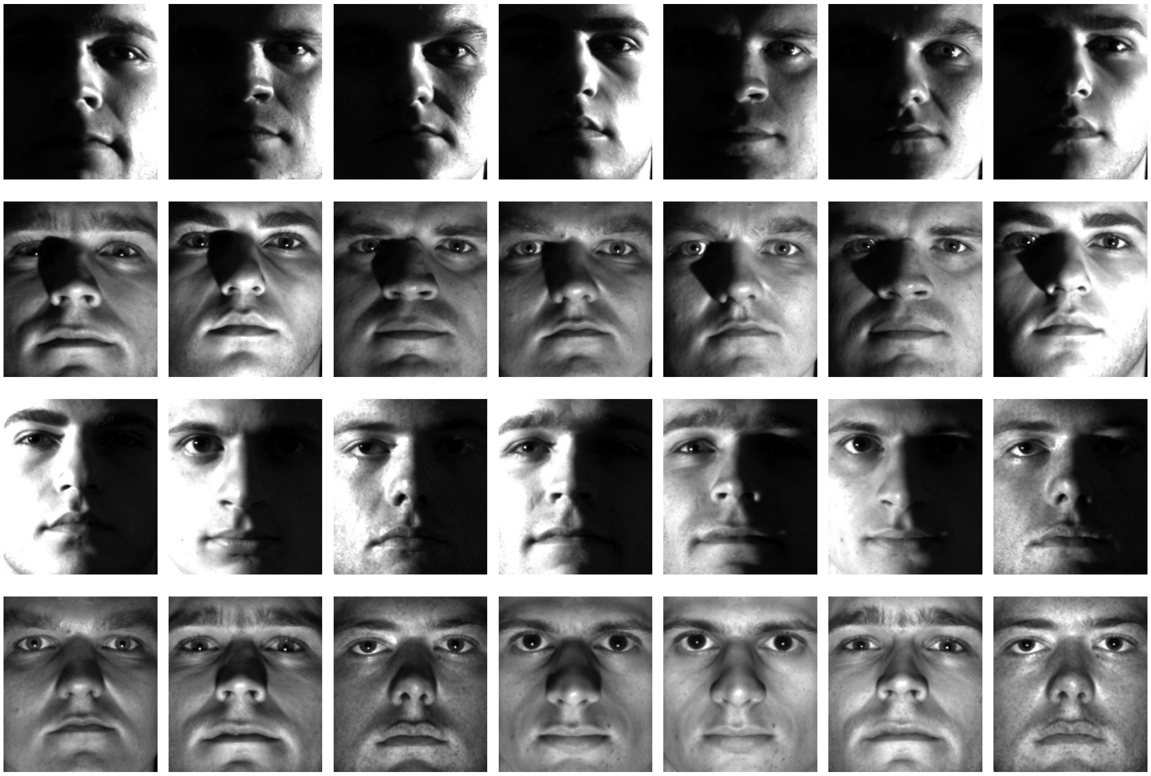


Figure 11: Solving the correspondence problem: given the images of the first column, the model searches only in the shared latent space to find the pictures of the opposite view which have the same illumination condition. The images found, are sorted in columns 2 - 7 by relevance.

158 frames was used as a test set. Each pose is represented by a 63−dimensional vector of joint locations and each silhouette is represented by a 100−dimensional vector of HoG (histogram of oriented gradients) features. To model this data we used MRD initialized with $q = 15$ latent dimensions. Given the test silhouette features $\{\mathbf{y}^{\mathcal{A}}_{i,*}\}_{i=1}^{n_*}$, we used our model to generate the corresponding poses $\{\mathbf{y}^{\mathcal{B}}_{i,*}\}_{i=1}^{n_*}$. This is challenging, as the data are multi-modal and ambiguous, i.e. a silhouette representation may be generated from more than one pose (e.g. Figure 12).

The inference procedure proceeds as described in Algorithm 1. Specifically, given a test point $\mathbf{y}^{\mathcal{A}}_{i,*}$ we firstly estimate the corresponding latent point $\mathbf{x}_{i,*}$ and then through a nearest neighbour search we seek the training latent point $\tilde{\mathbf{x}}$ which is closest to $\mathbf{x}_{i,*}$ in the shared dimensions. It is interesting to investigate which latent points $\tilde{\mathbf{x}}$ are returned by the nearest neighbour search, as this will reveal properties of the shared latent space. While exploring this aspect, we found that the training points suggested as most similar in the shared space typically correspond to silhouettes (outputs) similar to the given test one, $\mathbf{y}^{\mathcal{A}}_{i,*}$. This

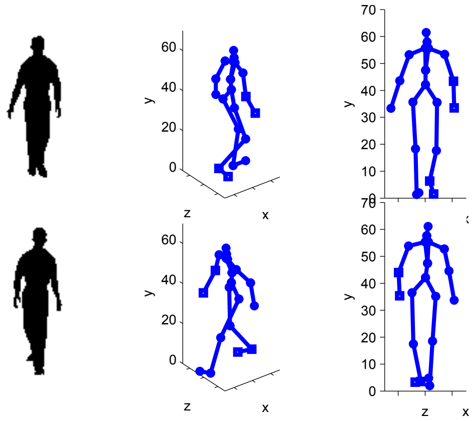


Figure 12: Although the two poses in the second column are very dissimilar, they correspond to resembling silhouettes that have similar feature vectors. This happens because the 3D information is lost in the silhouette space, as can also be seen in the third column, depicting the same poses from the silhouettes' viewpoint.

confirms that the factorization of the latent space is efficient in representing the correct kind of information in each subspace. However, when ambiguities arise, as the example shown in Figure 12, the non-dynamical version of our model has no way of selecting the correct input, since all points of the test sequence are treated independently. Intuitively, this means that two very similar test givens $(\mathbf{y}^{\mathcal{A}}_{i,*}, \mathbf{y}^{\mathcal{A}}_{i+1,*})$ can be mapped to generated latent vectors $(\tilde{\mathbf{x}}_{i,*}, \tilde{\mathbf{x}}_{(i+1),*})$ from which predictions $(\mathbf{y}^{\mathcal{B}}_{i,*}, \mathbf{y}^{\mathcal{B}}_{i+1,*})$ in the other modality can be drastically different. But when the dynamical version is employed, the model forces the whole set of training and test inputs (and, therefore, also the test outputs) to form smooth paths. In other words, the dynamics disambiguate the model.

To perform pose inference in the dynamical scenario a test silhouette $\mathbf{y}^{\mathcal{A}}_{i,*}$ is accompanied by its timestamp, $\mathbf{t}_{i,*}$, which is used to disambiguate the temporal approximate posterior $q(\mathbf{x}_{i,*}|\mathbf{t}_{i,*})$ and, consequently, the prediction $\mathbf{y}^{\mathcal{B}}_{i,*}$. This temporal disambiguation effect is demonstrated in Figure 13, where from a set of test silhouettes $\mathbf{Y}^{\mathcal{A}}_{*}$ we find the corresponding set of nearest training silhouettes $\widetilde{\mathbf{Y}}^{\mathcal{A}}$ through the shared latent space which respects dynamics. In this case, we see that each $\mathbf{y}^{\mathcal{A}}_{i,*}$ is not necessarily the most similar to the corresponding $\tilde{\mathbf{y}}^{\mathcal{A}}_{i,:}$, because that would mean that the dynamics would "break", as in Figure 13, column 1 versus column 3 of row 2. Instead, our model treats the whole test set as a sequence, so in column 2 of Figure 13 we see that the silhouette is more dissimilar to the given one (column 1) but it represents a walk in the same direction. What is more, if we assume that the test *pose* $\mathbf{y}^{\mathcal{B}}_{i,*}$ is known and look for its nearest training neighbour *in the pose space*, we find that the corresponding silhouette is very similar to the one found by our model, which is only given information in the silhouette space.

After the above analysis regarding the properties of the latent space, we now proceed to quantitatively evaluate the generation of test poses $\mathbf{y}^{\mathcal{B}}_{*}$ from test silhouettes $\mathbf{y}^{\mathcal{A}}_{*}$. Figure

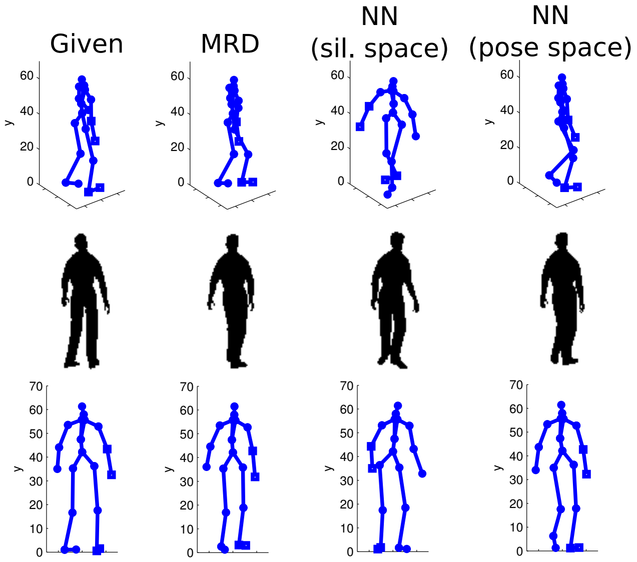


Figure 13: Given the HoG features $\mathbf{y}^{\mathcal{A}}_{i,*}$ for the test silhouette in column one, we predict the corresponding pose $\mathbf{y}^{\mathcal{B}}_{i,*}$ using the dynamical MRD and nearest neighbour (NN) in the silhouette space obtaining the results in the first row, columns 2 and 3 respectively. The last row is the same as the first one, but the poses are rotated to highlight the ambiguities. Notice that the silhouette shown in the second row for MRD does not correspond exactly to the pose of the first row, as the model generates a *novel* pose given a test silhouette. Instead, it is the training silhouette found by performing NN in the shared latent space to obtain $\tilde{\mathbf{x}}_{i,:}$. As some form of "ground truth", in column 4 we plot the NN of the training *pose* $\mathbf{y}^{\mathcal{A}}_{i,:}$ given the test pose $\mathbf{y}^{\mathcal{B}}_{i,*}$ (which is normally unknown during the test phase).

12 shows one encouraging example of this result. To more reliably quantify the results, we compare our method with linear and Gaussian process regression and with nearest neighbour in the silhouette space. We also compared against the shared GP-LVM (Ek, 2009) which optimizes the latent points using MAP and, thus, requires an initial factorization of the inputs to be given a priori. Finally, we compared to a dynamical version of nearest neighbour where we kept multiple nearest neighbours and selected the coherent ones over a sequence. The errors shown in Table 1 as well as the on-line videos (`http://git.io/vwLhH`) show that MRD performs better than the other methods in this task.

5.4 Discriminative - Generative MRD

So far the experiments have considered two continuous views and the task was either to generate novel data or to transfer information between the views. We now consider a hybrid discriminative - generative model and task, where one view contains labels of the features in the other view. This experimental setting is quite different from the ones considered so far, since the two views contain very diverse types of data; in particular, the class-label

	Error
Mean Training Pose	6.16
Linear Regression	5.86
GP Regression	4.27
Nearest Neighbour (sil. space)	4.88
Nearest Neighbour with sequences (sil. space)	4.04
Nearest Neighbour (pose space)	2.08
Shared GP-LVM	5.13
MRD without Dynamics	4.67
MRD with Dynamics	**2.94**

Table 1: The mean of the Euclidean distances of the joint locations between the predicted and the true poses. The nearest neighbour in the pose space is not a fair comparison (since the test pose is supposed to be unseen), but is reported as it provides some insight about the lower bound on the error that can be achieved for this task.

view contains discrete, low dimensional features. Further, these features are noise-free and very informative for the task at hand and, therefore, applying MRD in this data set can be seen as a form of supervised dimensionality reduction. The challenge for the model is to successfully cope with the different levels of noise in the views, while managing to recover a continuous shared latent space from two very diverse information sources, one of which is discriminative. In particular, we would ideally expect to obtain a shared latent space which encodes class information and a nonexistent private space for the class-label modality.

To test our hypotheses, we used the "oil flow" database (Bishop and James, 1993) which contains 1000 $12-$dimensional examples split in 3 classes. We selected 10 random subsets of the data with increasing number of training examples and compared to the nearest neighbor (NN) method in the data space. The label $\mathbf{y}^{\mathcal{B}}_{i,:} = [y^{\mathcal{B}}_{i,1}, y^{\mathcal{B}}_{i,2}, y^{\mathcal{B}}_{i,3}]^\top$ corresponding to the training instance $\mathbf{y}^{\mathcal{A}}_{i,:}$ was encoded so that $y^{\mathcal{B}}_{i,j} = -1$ if $\mathbf{y}^{\mathcal{A}}_{i,:}$ does not belong to class j, and $y^{\mathcal{B}}_{i,j} = 1$ otherwise. Given a test instance $\mathbf{y}^{\mathcal{A}}_{i,*}$, we predict the corresponding label vector $\mathbf{y}^{\mathcal{B}}_{i,*}$ as before. Since this vector contains continuous values, we use 0 as a threshold to obtain values $y^{\mathcal{B}}_{i,j} \in \{-1, 1\}$. With this technique, we can perform multi-class and multi-label classification, where an instance can belong to more than one classes. The specific data set considered in this section, however, is not multi-label and so we select the label for which the corresponding weight is the largest. To evaluate this technique, we computed the classification accuracy as a proportion of correctly classified instances. As can be seen in Figure 14, MRD successfully determines the shared information between the data and the label space and outperforms NN. This result suggests that MRD manages to factor out the non class-specific information found in $\mathbf{Y}^{\mathcal{A}}$ and perform classification based on more informative features (i.e. the shared latent space). With this experiment we wish to demonstrate the power of the method for learning from very diverse kinds of modalities; we clarify that we do not wish to present our method as a state-of-the-art discriminative classifier.

It is worth mentioning that, as expected, the models trained in each experimental trial defined a latent space factorization where there is no private space for the label view, whereas the shared space is one or two dimensional and is composed of three clusters (corresponding to the three distinct labels). Therefore, by factorizing out signal in $\mathbf{Y}^{\mathcal{A}}$ that is irrelevant to the classification task, we manage to obtain a better classification accuracy. The above are confirmed in Figure 15, where we plot the shared latent space and the relevance weights for the model trained on the full data set.

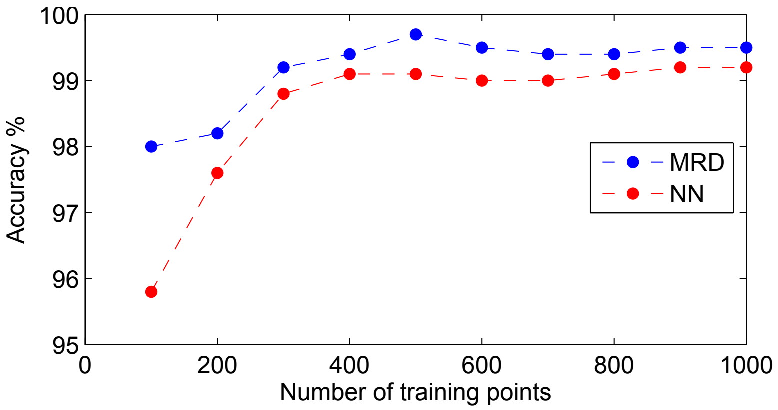


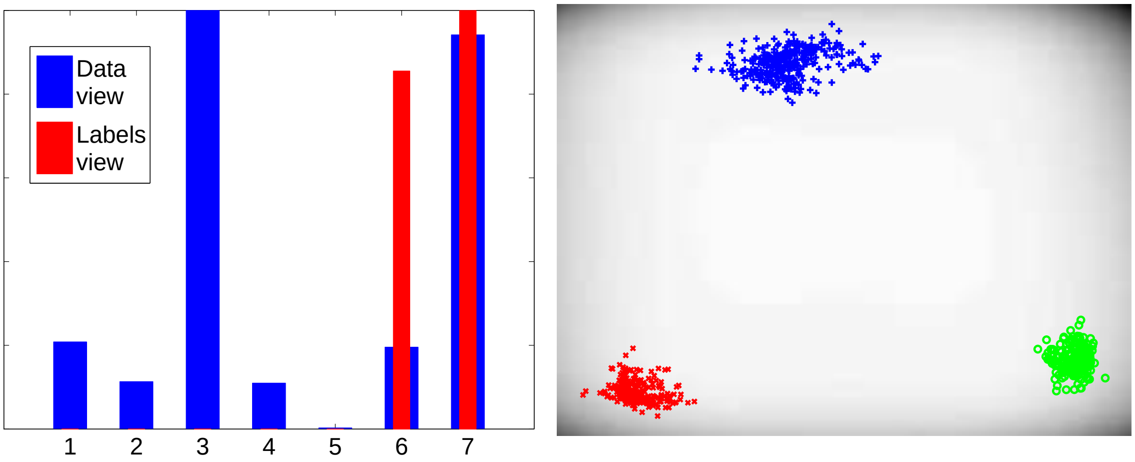
Figure 14: Accuracy obtained after testing MRD and NN on the full test set of the "oil flow" data set.

5.5 Multi-view Models and Data Exploration

We have so far demonstrated MRD in data sets with two modalities. However, there is no theoretical constraint on the number of modalities that can be handled by MRD, it naturally extends beyond two views. Even when multiple views of scarce data are considered, the principled Bayesian framework will provide strong regularization. This is one of the important and powerful aspects of MRD compared to previous work. In this section we will use the AVletters database (Matthews et al., 2002) to generate multiple views of data. This audio-visual data set was generated by recording the audio and visual signals of 10 speakers that uttered the letters A to Z three times each (i.e. three *trials*). The audio signal was processed to obtain a 299− dimensional vector per utterance. The video signal per utterance is a sequence of 24 frames, each being represented by the raw values of the 60×80 pixels around the lips, as can be seen in Figure 16. Thus, a single instance of the video modality of this data set is a 115200−dimensional vector. With different formulation of these data into views we construct three different scenarios, detailed in the following, in order to demonstrate MRD with large number of views.

Data Exploration

Depending on the desired predictive or exploratory task, different subsets of the data can be split across different views. To explore the connections and commonalities in the information encoded in different subjects, letters and type of signal (video or audio), we first performed data exploration by considering the following generic setting: we created a data set where



(a) Relevance weights

(b) Shared latent space.

Figure 15: Results from testing the MRD as a classifier on the full "oil flow" data set, where one of the views was the data and one the class labels. Three important observations can be made: firstly, both views "agree" on using dimensions 6 and 7 (and the data view even switches off one other dimension). Secondly, the labels' view has no private space. Thirdly, the shared latent space projection clearly clusters the data according to their class label.

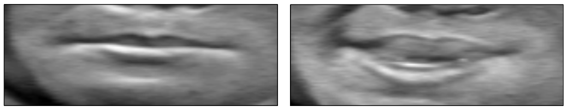


Figure 16: Two example frames of the AVletters data set.

the modalities were split across all subjects and across type of signal. We only considered 8 of the subjects. Thus, we ended up with 16 different modalities, where modalities $i, i+1$ contained the video and audio signal respectively for the $i-$th subject. The alignment was therefore made with respect to the different letters. We used all three available trials but letters "B", "M" and "T" were left out of the training set completely to be used at test time. For each modality, we thus had 69 rows (23 letters × 3 trials). The split across instances and modalities is summarized in Table 2. In the test set, each modality had only 9 rows (3 letters × 3 trials). Notice that this is a rather extreme scenario: the number of training instances is only 4.3 times larger than the number of modalities. We applied MRD to reveal the strength of commonality between signal corresponding to different subjects and to different recording type (video/audio). The visualization of the ARD weights can be seen in Figure 17.

This figure shows that similar weights (particularly in dimension 12) are typically found for modalities $1, 3, 5, ...$, i.e. the ones that correspond to the video signal. This visualization

		view 1	view 2	···	view 15	view 16
		(subj.1, video)	(subj.1, audio)		(subj.8, video)	(subj.8, audio)
1	‘A’ trial 1	*video data*	*audio data*		*video data*	*audio data*
2	‘A’ trial 2	*video data*	*audio data*		*video data*	*audio data*
3	‘A’ trial 3	*video data*	*audio data*		*video data*	*audio data*
⋮	...	...	...		...	...
$n = 69$	‘Z’ trial 3	*video data*	*audio data*		*video data*	*audio data*

Table 2: View/row split for the first AVletters experiment. Column k represents $\mathbf{Y}^{(k)}$ and row i represents $[\mathbf{y}_{i,:}^{(1)}, \ldots, \mathbf{y}_{i,:}^{(K)}]$.

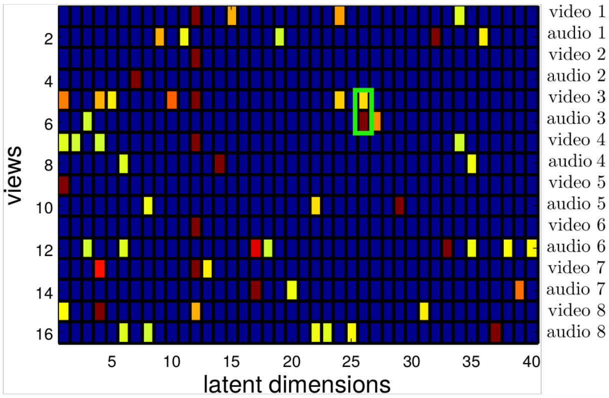


Figure 17: The optimized weights for the first version of the AVletters experiment represented as a heat-map. “Warm” (red) colors on column i, j indicate a large weight for latent dimension j and modality i. Notice that for visualization of these weights we normalized them to be between 0 and 1 and used a threshold so that $w_{i,j} < \varepsilon$ (with $\varepsilon \to 0$) was set to zero. The green box highlights a shared latent space (dimension 26) for views 5 and 6 (subject 3).

is instructive, as it reveals that to predict the lip movements in a test scenario, the other pieces of information that can help the most in this prediction is the lip movements of the rest of the subjects. We can also draw other sorts of conclusions from this kind of data exploration. For example, we can see that the subject number 3 is the one that has the best variance alignment between the video and audio signal. This can be understood by observing that the 5th and 6th row of the matrix in Figure 17 share some weights (highlighted with a green box), i.e. modality 5 and 6 share a latent subspace.

Generation Task

Given the above analysis, we attempted to recover the lip movements of subject 3 for uttering the three test letters “B”, “M”, “T”. The given information was the audio signal of this subject as well as the video signal of all the rest (corresponding to the same letters). The RMSE error for MRD was 0.3 while for NN it was 0.35.

Incorporating Labels

We subsequently considered a different scenario for modelling the AVletters data base in which we also wanted to include some sort of label information, following the promising results of the discriminative model detailed in Section 5.4. Specifically, we selected the utterances of the first three subjects for the first two trials and for letters A to Q and constructed three views as follows; view 1 contained the audio signal by stacking the relevant information for all considered subjects, trials and letters. Similarly, view 2 contained the video signal. Each row in views 1 and 2 corresponds to one of three subjects, and to encode this information we use a third view. Thus, view 3 contains the discrete labels $C \in \{000, 010, 100\}$ which specify the subject identity. This construction resulted in a training set of 102 data points per view (3 subjects × 17 letters × 2 trials) and is summarized in Table 3. Therefore, the subject identity is directly encoded by a single view containing discrete labels corresponding to each row of all the other views. This comes in contrast to the representation described in the previous paragraph, where the subject identity was implicitly encoded by having a separate view per subject. The ordering of the rows in the three views does not matter, as long as the same permutation is applied to all three views.

		View 1	View 2	View 3
1	subj 1, ‘A’, trial 1	audio	video	001
2	subj 1, ‘A’, trial 2	audio	video	001
⋮	…	…	…	…
34	subj 1, ‘Q’, trial 2	audio	video	001
35	subj 2, ‘A’, trial 1	audio	video	010
⋮	…	…	…	…
$n = 102$	subj 3, ‘Q’, trial 2	audio	video	100

Table 3: View/row split for the second ‘AVletters’ experiment.

With the above setting we are, thus, interested in modelling the commonality between the two types of signal for each person with regards to global characteristics, like accent, voice, lip movement and shape. This is because the data were aligned across modalities so that audio and video utterances were matched irrespective of the pronounced letter. The factorization learned by MRD is shown by the optimized weights in Figure 18.

As can be seen, in this scenario the video and audio signals only share one latent dimension (4) while the subject identifier modality shares weights with both of the other two signals (dimensions 1 and 23). This means that, given this small training set, MRD manages to reveal the commonality between the audio and video signals and, at the same time, learn to differentiate between subjects in both the video and the audio domain. It can also be seen that the video modality which is higher dimensional than the audio modality is nevertheless represented by fewer latent dimensions. This is because there is greater redundancy in the video modality but also because MRD is a nonlinear method and, hence, the number of active dimensions is not proportional to the complexity of the latent space

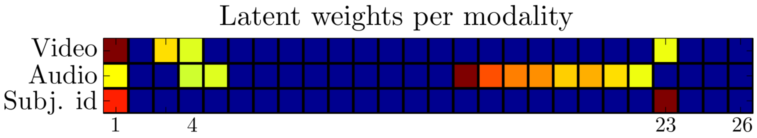


Figure 18: The optimized weights for the second version of the AVletters experiment represented as a heat-map which has the same format as for Figure 17.

learned. To further demonstrate the power of MRD in representation learning, we attempted to transfer information between views. To do that, we created a test set in a similar way as for the training one but for the third trial of the recordings. From this test data set we attempted to predict the video signal in two ways: firstly by giving only the corresponding audio signal and, secondly, by giving both the audio and subject identity information. For comparison, we used nearest neighbour and standard Gaussian process regression, as can be seen in Table 4 which presents the corresponding RMSE.

Given	Predicted	MRD	NN	GP
audio	video	0.498	0.485	0.494
audio, labels	video	0.434	0.485	0.472

Table 4: RMSE of predicting the video information of test data, given only the audio or the audio and subject id information.

Notice that the model could also end up with a completely separate space for the third modality (labels); the fact that it didn't means that the way in which video and audio are associated in this data set is also dependent on the subject, something which is expected, especially since two of the subjects are females and one is male. One can see from Table 4 that when MRD is given the label information, it can disambiguate better in the prediction phase and produce smaller error. Figure 19 illustrates some example video frames generated from this experiment. As can be seen, in general our model produces sharp images which are similar to the ground truth. However, occasionaly the model predicts in the wrong modality (i.e. last column), which is a sign that the label (subject id) signal can occasionaly be overwhelmed by the signal of the audio modality.

Finally, we tested the model in the opposite direction: specifically, we presented the video and audio signal of the test (third) trial of the recordings to the model and tried to recover the identity of the subject. To make this more challenging, we randomly erased 50% of the information from each of the given views. The vast dimensionality of the given space did not allow us to compare against a standard GP regression model, so we only compared against nearest neighbour. The result was an F-measure[6] of 0.92 for MRD compared to 0.76 for NN.

[6] The F-measure is given by $F_1 = 2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}} = \frac{2 \cdot \text{true positives}}{2 \cdot \text{true positives} + \text{false negatives} + \text{false positives}}$.

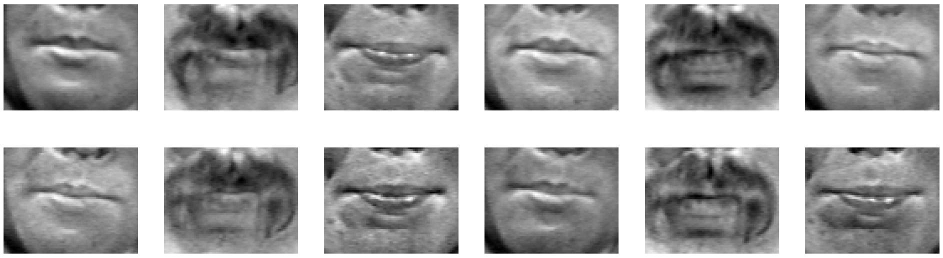


Figure 19: Generated frames for the results corresponding to Table 4. Top row presents generated frames and bottom row presents the corresponding ground truth.

5.6 Automatic Correlation Learning of Output Dimensions

GP-LVM-based models make the assumption that all dimensions of the latent space, $\{\mathbf{x}_{:,j}\}_{j=1}^{q}$, affect all output dimensions, $\{\mathbf{y}_{:,j}\}_{j=1}^{p}$, since a single GP mapping is used[7]. In MRD we make conditional independence assumptions for *subspaces* of $\mathbf{X}$, so that different views are generated by different GP mappings. In this section, we consider the extreme case where we want to learn automatically *all* conditional independencies of a multivariate data set's dimensions. In some sense, this is a means of simultaneously performing graphical model learning (learn conditional independencies) and manifold learning (learn manifolds). To achieve this goal, we reformulate the observed data set $\mathbf{Y} \in \Re^{n \times p}$ so that each dimension $\mathbf{y}_{:,j} \in \Re^{n}$ constitutes a separate, one-dimensional view of the data. Then, following the MRD formulation, we consider a single latent space $\mathbf{X}$ which encapsulates output correlations through the p sets of ARD weights. These weights constitute the hyperparameters of p separate, independent Gaussian processes.

The above formulation, from now on referred to as *fully independent MRD (FI-MRD)*, allows us to discover correlations of the output dimensions via the latent space. We have already seen examples of this task in Figure 10, where we sampled from specific latent dimensions and observed the obtained variance in the outputs. A similar task could be achieved with the motion capture data set of Figure 12. For example, we might observe that a specific latent dimension is responsible for encoding the variance in both legs' movements. In this section we are interested in discovering this kind of effects *automatically* (without the need to sample) and also by considering all possible *subsets* of latent dimensions at the same time. This task reminds the objectives of multi-output Gaussian process inference (see e.g. Alvarez et al., 2010), according to which the GP output function correlations are sought to be modelled explicitly by defining special covariance functions. However, in our framework the inputs to the covariance functions are latent, and the output correlations are rather captured by defining a special noise model.

To test this model, we considered again the motion capture data set introduced in Section 5.3 (without the silhouette views). We used a smaller subset of 120 frames that represent 4 distinct walking motions (two facing to the North, one facing to the South, and a semi-

[7] More precisely, multiple GP mappings with *shared parameters* are used.

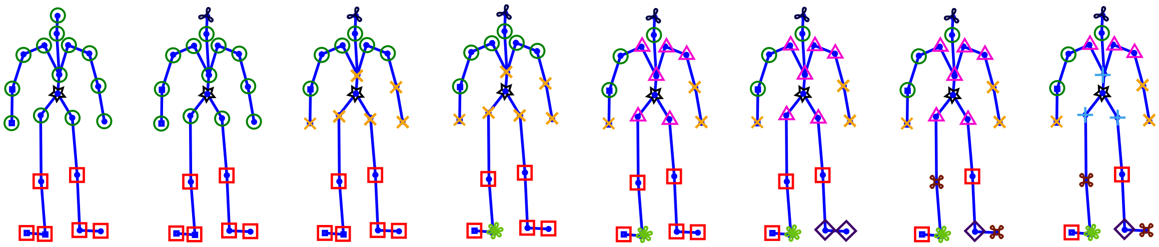


Figure 20: In the fully independent MRD experiment we have 21 joints × 3 degrees of freedom (x,y,z coordinates) = 63 one-dimensional views. The 3 sets of ARD weights for each joint are clustered in $\mathcal{K}$ clusters. The figure shows the cluster assignments for each joint, for $\mathcal{K} = 3$ (far left) to $\mathcal{K} = 10$ (far right). Each cluster is represented as a separate combination of symbol/color and corresponds to output dimensions controlled by similar latent subspaces.

circular motion). In this motion capture data set, the human subjects are represented in the 3D coordinate space. That is, each of the 21 body joints is represented as a 3 dimensional vector, corresponding to the degrees of freedom along the (x, y, z) axes. Therefore, we have $p = 63$ columns in the original data set which we use to form 63 single-dimensional views. The motion is centered with respect to the root node, so that the subject moves in place. Each view has its own set of ARD weights, but since every 3 views (and thus every 3 sets of weights) correspond to the same joint (for different degrees of freedom), we can group these 63 weight sets as $\mathbf{w}_{xyz}^{(j)} = [\mathbf{w}_x^{(j)} \mathbf{w}_y^{(j)} \mathbf{w}_z^{(j)}], j = 1, \cdots, 21$. Notice, however, that each of the 63 sets of weights is a priori independent and learned separately, we just group them by 3 according to their corresponding joints *after optimization*, just for performing our data analysis (explained below) more easily. For the data analysis, we perform $\mathcal{K}$-means clustering on the 21 vectors $\mathbf{w}_{xyz}^{(j)}, j = 1, ..., 21$, where each vector has dimensionality $3q \times 1$. In this way, we can investigate which parts of the latent space are operating together to control a particular joint.

The $q = 10$ dimensional latent space was not constrained with dynamics, and was initialized by performing PCA on the whole $\mathbf{Y}$ data set. As can be seen in Figure 20 the model uncovers very intuitive correlations in the joints of the body, via the cluster assignments. For example, the model tends to represent the head with a separate latent subspace, whereas joints belonging to the same limb typically share parts of the latent space. Moreover, the discovered clusters are similar (but not identical) to the ones obtained if we directly cluster the output dimensions corresponding to each joint (after we group them by 3, aggregating information for the x, y, z coordinates). Therefore, the fully independent MRD formulation manages to maintain and uncover the cluster related properties of the original data set in the efficiently factorized latent space. Achieving this via a low-dimensional latent space is important, since very high-dimensional output spaces might not be easily clustered in the high-dimensional Euclidean space. Further, the MRD formulation allows us to transfer information between output dimensions (which here constitute separate views), a task which is typically solved (in supervised learning) using multi-output GPs. Finally, this experiment highlights the advantage of our Bayesian framework, since the number of data points is only less than double of the number of views (120 datapoints, 63 views). To the best of our knowledge, MRD is the only multi-view model operating well in this kind of extreme scenarios.

6. Conclusions

In this paper we have presented a model capable of consolidating several different views using a single factorized latent variable. The model is fully probabilistic, handles a large variety of different data and is capable of modelling very high dimensional observations. We have shown how the model can recover the generating parameters from extremely high-dimensional data and how the factorized latent space can be used to model ambiguities in the data. Its power has been demonstrated in challenging cases of representation learning, generation of novel data, discriminative tasks and information transfer between views. We also highlighted the utility of the model in data exploration and developed the FI-MRD (Section 5.6) as a particular extension.

The model is trained in a Bayesian manner using an efficient variational approximation which makes it possible to learn even from very small amounts of data and naturally allows for additional priors to be included. Compared to previous work, our approach significantly advances the state-of-the-art by equipping the standard IBFA model with two very important properties: nonlinearity and nonparametric mappings. We also provided an alternative Bayesian formulation and link to IBFA, in a similar vein to the work of Virtanen et al. (2011); Klami et al. (2013); Damianou et al. (2012). Previous IBFA models have typically been demonstrated in scenarios with two or, more rarely, very few views compared to the number of data. To the best of our knowledge, our paper is the first to present results in cases comprising a truly large number of views with simultaneous data-efficient learning.

The presented model can be readily used in many interesting application scenarios, especially when data is scarce and comes from several noisy views that individually cannot disambiguate the task. Currently we are exploring the use of MRD in robotics and computational biology. Exploiting the typically rich prior knowledge in tasks associated with these fields is possible with future research that targets sophisticated priors for MRD. Scaling up our model to tens of thousands of datapoints is also possible, following recent work in distributed (Gal et al., 2014; Dai et al., 2014) and stochastic (Hoffman et al., 2013; Hensman et al., 2013) variational inference. This promising direction is left for future work.

Acknowledgments

We thank Michalis Titsias, Max Zwiessele, Zhenwen Dai for helpful discussions related to this research.

Appendix A. Details on the Derivation of the Variational Lower Bound

In this appendix we demonstrate the detailed derivation of the variational lower bound. For simplicity, we will assume that we only have two views, $\mathbf{Y}^{\mathcal{A}} \triangleq \mathbf{Y}^{(1)}$ and $\mathbf{Y}^{\mathcal{B}} \triangleq \mathbf{Y}^{(2)}$, and the extensions to multiple views follows trivially.

As explained in the main paper, we assume Gaussian process priors on the mappings, so that:

$$\begin{aligned} f^{\mathcal{A}} \sim \mathcal{GP}(\mathbf{y}^{\mathcal{A}}, k^{\mathcal{A}}) \Rightarrow p(\mathbf{F}^{\mathcal{A}}|\mathbf{X}, \boldsymbol{\theta}^{\mathcal{A}}) = \prod_{j=1}^{p_{\mathcal{A}}} \mathcal{N}(\mathbf{f}^{\mathcal{A}}_{:,j}|\mathbf{0}, \mathbf{K}^{\mathcal{A}}) \\ f^{\mathcal{B}} \sim \mathcal{GP}(\mathbf{y}^{\mathcal{B}}, k^{\mathcal{B}}) \Rightarrow p(\mathbf{F}^{\mathcal{B}}|\mathbf{X}, \boldsymbol{\theta}^{\mathcal{B}}) = \prod_{j=1}^{p_{\mathcal{B}}} \mathcal{N}(\mathbf{f}^{\mathcal{B}}_{:,j}|\mathbf{0}, \mathbf{K}^{\mathcal{B}}), \end{aligned} \tag{16}$$

where $\mathbf{K}^{\{\mathcal{A},\mathcal{B}\}} = k^{\{\mathcal{A},\mathcal{B}\}}(\mathbf{x}, \mathbf{x}')$ are the covariance matrices evaluated at the latent points.

The first step in defining a Bayesian training procedure, is to place a prior distribution $p(\mathbf{X}|\boldsymbol{\theta}_x)$ over the latent space, where $\boldsymbol{\theta}_x$ denotes any parameters associated with this prior. For the moment we will not assume any particular form for this distribution and we will omit the conditioning on $\boldsymbol{\theta}_x$. Then, the joint distribution of the model is written as

$$\begin{aligned} p(\mathbf{Y}^{\mathcal{A}}, \mathbf{Y}^{\mathcal{B}}, \mathbf{F}^{\mathcal{A}}, \mathbf{F}^{\mathcal{B}}, \mathbf{X}) =& p(\mathbf{Y}^{\mathcal{A}}|\mathbf{F}^{\mathcal{A}})p(\mathbf{F}^{\mathcal{A}}|\mathbf{X})p(\mathbf{Y}^{\mathcal{B}}|\mathbf{F}^{\mathcal{B}})p(\mathbf{F}^{\mathcal{B}}|\mathbf{X})p(\mathbf{X}) \\ =& p(\mathbf{X}) \prod_{j=1}^{p_{\mathcal{A}}} p(\mathbf{y}^{\mathcal{A}}_{:,j}|\mathbf{f}^{\mathcal{A}}_{:,j})p(\mathbf{f}^{\mathcal{A}}_{:,j}|\mathbf{X}) \prod_{j=1}^{p_{\mathcal{B}}} p(\mathbf{y}^{\mathcal{B}}_{:,j}|\mathbf{f}^{\mathcal{B}}_{:,j})p(\mathbf{f}^{\mathcal{B}}_{:,j}|\mathbf{X}), \end{aligned} \tag{17}$$

where we assume independence in the data features given the latent variables. Then, we seek to optimize the model by computing the marginal likelihood

$$p(\mathbf{Y}^{\mathcal{A}}, \mathbf{Y}^{\mathcal{B}}) = \int_{\mathbf{X},\mathbf{F}^{\mathcal{A}},\mathbf{F}^{\mathcal{B}}} p(\mathbf{Y}^{\mathcal{A}}|\mathbf{F}^{\mathcal{A}})p(\mathbf{F}^{\mathcal{A}}|\mathbf{X})p(\mathbf{Y}^{\mathcal{B}}|\mathbf{F}^{\mathcal{B}})p(\mathbf{F}^{\mathcal{B}}|\mathbf{X})p(\mathbf{X}). \tag{18}$$

The key difficulty with this Bayesian approach is propagating the prior density $p(\mathbf{X})$ through the nonlinear mapping. This mapping gives the expressive power to the model, but simultaneously renders the associated marginal likelihood (18) intractable.

We now invoke the variational Bayesian methodology to approximate the integral. Following a standard variational inference procedure, we introduce a variational distribution which we assume to factorise as $q(\boldsymbol{\Theta})q(\mathbf{X})$ where $q(\mathbf{X}) \sim \mathcal{N}(\boldsymbol{\mu}, \mathbf{S})$ and $\mathbf{U}$ denotes additional variational parameters which we will define later on. We now compute the Jensen's lower bound $\mathcal{L}_{\mathcal{A},\mathcal{B}}$ on the logarithm of (18),

$$
\begin{aligned}
\mathcal{L}_{\mathcal{A},\mathcal{B}} &= \int_{\mathbf{X},\mathbf{U}} q(\mathbf{U})q(\mathbf{X}) \log \frac{p(\mathbf{Y}^{\mathcal{A}}, \mathbf{Y}^{\mathcal{B}}|\mathbf{X})p(\mathbf{X})}{q(\mathbf{U})q(\mathbf{X})} \qquad (19)\\
&= \int_{\mathbf{X},\mathbf{U},\mathbf{F}^{\mathcal{A}},\mathbf{F}^{\mathcal{B}}} q(\mathbf{U})q(\mathbf{X}) \log \left(\frac{p(\mathbf{Y}^{\mathcal{A}}|\mathbf{F}^{\mathcal{A}})p(\mathbf{F}^{\mathcal{A}}|\mathbf{X})p(\mathbf{Y}^{\mathcal{A}}|\mathbf{F}^{\mathcal{B}})p(\mathbf{F}^{\mathcal{B}}|\mathbf{X})}{q(\mathbf{U})} \frac{p(\mathbf{X})}{q(\mathbf{X})} \right)\\
&= \int_{\mathbf{F}^{\mathcal{A}},\mathbf{X},\mathbf{U}} q(\mathbf{U})q(\mathbf{X}) \log \frac{p(\mathbf{Y}^{\mathcal{A}}|\mathbf{F}^{\mathcal{A}})p(\mathbf{F}^{\mathcal{A}}|\mathbf{X})}{q(\mathbf{U})}\\
&\quad + \int_{\mathbf{F}^{\mathcal{B}},\mathbf{X},\mathbf{U}} q(\mathbf{U})q(\mathbf{X}) \log \frac{p(\mathbf{Y}^{\mathcal{A}}|\mathbf{F}^{\mathcal{B}})p(\mathbf{F}^{\mathcal{B}}|\mathbf{X})}{q(\mathbf{U})}\\
&\quad - \int_{\mathbf{X}} \cancel{q(\mathbf{U})} q(\mathbf{X}) \log \frac{q(\mathbf{X})}{p(\mathbf{X})}\\
&= \tilde{\mathcal{L}}_{\mathcal{A}} + \tilde{\mathcal{L}}_{\mathcal{B}} - \mathrm{KL}\left(q(\mathbf{X}) \,\|\, p(\mathbf{X})\right), \qquad (20)
\end{aligned}
$$

where $\boldsymbol{\theta}$ denotes the model's parameters $\boldsymbol{\theta} = \{\boldsymbol{\theta}^{\mathcal{A}}, \boldsymbol{\theta}^{\mathcal{B}}\}$. However, the above form of the lower bound is problematic because $\mathbf{X}$ (in the GP terms $p(\mathbf{F}^{\mathcal{A}}|\mathbf{X})$ and $p(\mathbf{F}^{\mathcal{B}}|\mathbf{X})$) appears nonlinearly inside the kernel matrices $\mathbf{K}^{\mathcal{A}}$ and $\mathbf{K}^{\mathcal{B}}$ of equation (16), making the integration over $\mathbf{X}$ difficult. It is, thus, obvious that standard mean field variational methodologies do not lead to an analytically tractable algorithm.

In contrast, our framework allows us to compute a closed-form Jensen's lower bound by applying variational inference after expanding the GP prior so as to include auxiliary inducing variables. Originally, inducing variables were introduced for computational speed ups in GP regression models. In our approach, these extra variables are used as in the variational sparse GP method of Titsias (2009). More specifically, we expand the joint probability model in equation (17) with m extra samples $\mathbf{U}^{\mathcal{A}}$ and $\mathbf{U}^{\mathcal{B}}$ of the latent functions $\mathbf{f}^{\mathcal{A}}$ and $\mathbf{f}^{\mathcal{B}}$ evaluated at a set of pseudo-inputs (known as "inducing points") $\mathbf{Z}^{\mathcal{A}}$ and $\mathbf{Z}^{\mathcal{B}}$ respectively. Here, $\mathbf{U}^{\mathcal{A}} \in \mathbb{R}^{m_{\mathcal{A}} \times p_{\mathcal{A}}}$, $\mathbf{U}^{\mathcal{B}} \in \mathbb{R}^{m_{\mathcal{B}} \times p_{\mathcal{B}}}$, $\mathbf{Z}^{\mathcal{A}} \in \mathbb{R}^{m_{\mathcal{A}} \times q}$, $\mathbf{Z}^{\mathcal{B}} \in \mathbb{R}^{m_{\mathcal{B}} \times q}$ and $m = m_{\mathcal{A}} + m_{\mathcal{B}}$. Typically we select $m \ll n$ to also gain in computational speed.

The augmented joint probability density takes the form

$$
\begin{aligned}
p(\mathbf{Y}^{\mathcal{A}}, \mathbf{Y}^{\mathcal{B}}, \mathbf{F}^{\mathcal{A}}, \mathbf{F}^{\mathcal{B}}, \mathbf{U}^{\mathcal{A}}, \mathbf{U}^{\mathcal{B}}, \mathbf{X}|\mathbf{Z}^{\mathcal{A}}, \mathbf{Z}^{\mathcal{B}}) = p(\mathbf{X}) &\prod_{j=1}^{p_{\mathcal{A}}} p(\mathbf{y}^{\mathcal{A}}_{:,j}|\mathbf{f}^{\mathcal{A}}_{:,j})p(\mathbf{f}^{\mathcal{A}}_{:,j}|\mathbf{u}^{\mathcal{A}}_{:,j}, \mathbf{X})p(\mathbf{u}^{\mathcal{A}}_{:,j}|\mathbf{Z}^{\mathcal{A}})\\
&\prod_{j=1}^{p_{\mathcal{B}}} p(\mathbf{y}^{\mathcal{B}}_{:,j}|\mathbf{f}^{\mathcal{B}}_{:,j})p(\mathbf{f}^{\mathcal{B}}_{:,j}|\mathbf{u}^{\mathcal{B}}_{:,j}, \mathbf{X})p(\mathbf{u}^{\mathcal{B}}_{:,j}|\mathbf{Z}^{\mathcal{B}}) \qquad (21)
\end{aligned}
$$

where $p(\mathbf{u}^{\{\mathcal{A},\mathcal{B}\}}_{:,j}|\mathbf{Z}^{\{\mathcal{A},\mathcal{B}\}})$ are zero-mean Gaussians with covariance matrices $\mathbf{K}^{\mathcal{A}}_{uu}$ and $\mathbf{K}^{\mathcal{B}}_{uu}$ respectively, constructed using the same functions as for the GP priors (16). We write the augmented GP prior analytically (see Rasmussen and Williams (2006)) as

$$
p(\mathbf{f}^{\mathcal{A}}_{:,j}|\mathbf{u}^{\mathcal{A}}_{:,j}, \mathbf{X}) = \mathcal{N}\left(\mathbf{f}^{\mathcal{A}}_{:,j}|\mathbf{K}^{\mathcal{A}}_{fu}\left(\mathbf{K}^{\mathcal{A}}_{uu}\right)^{-1}\mathbf{u}^{\mathcal{A}}_{:,j}, \mathbf{K}^{\mathcal{A}}_{ff} - \mathbf{K}^{\mathcal{A}}_{fu}\left(\mathbf{K}^{\mathcal{A}}_{uu}\right)^{-1}\mathbf{K}^{\mathcal{A}}_{uf}\right), \qquad (22)
$$

and similarly for $\mathcal{B}$. Here, $\mathbf{K}^{\mathcal{A}}_{ff} = k^{\mathcal{A}}(\mathbf{X}, \mathbf{X})$ and $\mathbf{K}^{\mathcal{A}}_{fu}$ denotes the cross-covariance between the function values of $k^{\mathcal{A}}$ evaluated at the latent points $\mathbf{X}$ and the inducing points $\mathbf{Z}^{\mathcal{A}}$, i.e. $\mathbf{K}^{\mathcal{A}}_{fu} = k^{\mathcal{A}}(\mathbf{X}, \mathbf{Z}^{\mathcal{A}})$.

Analogously to Titsias and Lawrence (2010), we are now able to obtain a tractable lower bound through the variational density:

$$q(\mathbf{U})q(\mathbf{X}) = \{q(\mathbf{U}^{\mathcal{K}})p(\mathbf{F}^{\mathcal{K}}|\mathbf{U}^{\mathcal{K}}, \mathbf{X}, \mathbf{Z}^{\mathcal{K}})\}_{\mathcal{K}=\{\mathcal{A},\mathcal{B}\}}q(\mathbf{X}), \tag{23}$$

where $q(\mathbf{U}^{\{\mathcal{A},\mathcal{B}\}})$ are free form distributions and $q(\mathbf{X})$ a Gaussian with parameters $\boldsymbol{\mu}$ and $\mathbf{S}$. Notice also that (23) factorises across dimensions. Optimization of the variational lower bound provides an approximation to the true posterior $p(\mathbf{X}|\mathbf{Y}^{\mathcal{A}}, \mathbf{Y}^{\mathcal{B}})$ by $q(\mathbf{X})$.

After defining a variational distribution, we can continue our derivation by returning to the expression for the lower bound (20) and replacing the joint distribution with its augmented version (21) and the variational distribution with its factorised version (23). Since the variational bound breaks to separate terms for each of the observations spaces, here we will drop the subscripts $\mathcal{A}$ and $\mathcal{B}$ and show how, in general, we can calculate the $\tilde{\mathcal{L}}$ terms of equation (20) for a general observation space $\mathbf{Y}$. We have:

$$\begin{aligned}
\tilde{\mathcal{L}} &= \int_{\mathbf{X},\mathbf{F},\mathbf{U}} q(\mathbf{U})q(\mathbf{X}) \log \frac{p(\mathbf{Y}, \mathbf{F}, \mathbf{U}|\mathbf{X}, \mathbf{Z})}{q(\mathbf{U})} \\
&= \int_{\mathbf{X},\mathbf{F},\mathbf{U}} \prod_{j=1}^{p} p(\mathbf{f}_{:,j}|\mathbf{u}_{:,j}, \mathbf{X}, \mathbf{Z})q(\mathbf{u}_{:,j}) \log \frac{\prod_{j=1}^{p} p(\mathbf{y}_{:,j}|\mathbf{f}_{:,j})\cancel{p(\mathbf{f}_{:,j}|\mathbf{u}_{:,j}, \mathbf{X}, \mathbf{Z})}p(\mathbf{u}_{:,j}|\mathbf{Z})}{\prod_{j=1}^{p} \cancel{p(\mathbf{f}_{:,j}|\mathbf{u}_{:,j}, \mathbf{X}, \mathbf{Z})}q(\mathbf{u}_{:,j})q(\mathbf{X})} \\
&= \int_{\mathbf{X},\mathbf{F},\mathbf{U}} \prod_{j=1}^{p} p(\mathbf{f}_{:,j}|\mathbf{u}_{:,j}, \mathbf{X}, \mathbf{Z})q(\mathbf{u}_{:,j})q(\mathbf{X}) \log \frac{\prod_{j=1}^{p} p(\mathbf{y}_{:,j}|\mathbf{f}_{:,j})p(\mathbf{u}_{:,j}|\mathbf{Z})}{\prod_{j=1}^{p} q(\mathbf{u}_{:,j})q(\mathbf{X}))}.
\end{aligned}$$

Dropping $\mathbf{Z}$ from our expressions, for simplicity, we finally obtain:

$$\tilde{\mathcal{L}} = \sum_{j=1}^{p} \left(\int_{\mathbf{u}_{:,j},\mathbf{X}} q(\mathbf{u}_{:,j})q(\mathbf{X})\mathbb{E}_{p(\mathbf{f}_{:,j}|\mathbf{u}_{:,j},\mathbf{X})} \left[\log p(\mathbf{y}_{:,j}|\mathbf{f}_{:,j})\right] + \log \mathbb{E}_{q(\mathbf{u}_{:,j})} \left[\frac{p(\mathbf{u}_{:,j})}{q(\mathbf{u}_{:,j})}\right] \right) = \sum_{j=1}^{p} \tilde{\mathcal{L}}_{:,j}, \tag{24}$$

Calculating (24) in the same manner for every available observation space and replacing back in the variational bound (20) we obtain the final form of the bound which is now analytically tractable. In particular, the KL term is tractable and easy for certain priors $p(\mathbf{X})$. As for the $\tilde{\mathcal{L}}$ terms, we can calculate the expectation over $p(\mathbf{f}_{:,j}|\mathbf{u}_{:,j}, \mathbf{X})$ and reveal that the optimal setting for $q(\mathbf{u}_{:,j})$ is also a Gaussian. More specifically, we have:

$$\tilde{\mathcal{L}}_{:,j} = \int_{\mathbf{u}_{:,j}} q(\mathbf{u}_{:,j}) \log \frac{e^{\mathbb{E}_{q(\mathbf{X})}\left[\log N\left(\mathbf{y}_{:,j}|\mathbf{a}_{:,j},\beta^{-1}\mathbf{I}_{:,j}\right)\right]}p(\mathbf{u}_{:,j})}{q(\mathbf{u}_{:,j})} - \mathbf{C}, \tag{25}$$

where $\mathbf{a}_{:,j}$ is the mean of (22) and $\mathbf{C} = \frac{\beta}{2}\mathrm{Tr}(\mathbb{E}_{q(\mathbf{X})}\left[\mathbf{K}_{ff}\right]) + \frac{\beta}{2}\mathrm{Tr}\left(\mathbf{K}_{uu}^{-1}\mathbb{E}_{q(\mathbf{X})}\left[\mathbf{K}_{uf}\mathbf{K}_{fu}\right]\right)$. The expression in (25) is a KL-like quantity and, therefore, $q(\mathbf{u}_{:,j})$ is optimally set to be the quantity appearing in the numerator of the above equation. So:

$$q(\mathbf{u}_{:,j}) = e^{\mathbb{E}_{q(\mathbf{X})}\left[\log \mathcal{N}\left(\mathbf{y}_{:,j}|\mathbf{a}_{:,j},\beta^{-1}\mathbf{I}_{:,j}\right)\right]}p(\mathbf{u}_{:,j}),$$

exactly as in Titsias and Lawrence (2010). This is a Gaussian distribution since we have assumed $p(\mathbf{u}_{:,j}) = \mathcal{N}(\mathbf{u}_{:,j}|\mathbf{0}, \mathbf{K}_{uu})$.

After replacing $q(\mathbf{u}_{:,j})$ with its optimal value, we can reverse Jensen's inequality (i.e. substitute the optimal form back inside the bound) to obtain:

$$\tilde{\mathcal{L}}_{:,j} \geq \log \int_{\mathbf{u}_{:,j}} e^{\mathbb{E}_{q(\mathbf{X})}\left[\log N\left(\mathbf{y}_{:,j}|\mathbf{a}_{:,j},\beta^{-1}\mathbf{I}_{:,j}\right)\right]} p(\mathbf{u}_{:,j}) - \mathbf{C}. \tag{26}$$

Notice that the expectation appearing above is a standard Gaussian integral and (26) can be calculated in closed form, which turns out to be:

$$\tilde{\mathcal{L}}_{:,j}(q,\boldsymbol{\theta}) \geq \log \left[\frac{(\beta)^{\frac{N}{2}} |\mathbf{K}_{uu}|^{\frac{1}{2}}}{(2\pi)^{\frac{N}{2}} |\beta\boldsymbol{\Phi} + \mathbf{K}_{uu}|^{\frac{1}{2}}} e^{-\frac{1}{2}\mathbf{y}_d^\top \mathbf{W} \mathbf{y}_{:,j}} \right] - \frac{\beta\psi}{2} + \frac{\beta}{2} \mathrm{Tr}\left(\mathbf{K}_{uu}^{-1}\boldsymbol{\Phi}\right) \tag{27}$$

where:

$$\psi = \mathrm{Tr}(\mathbb{E}_{q(\mathbf{X})}\left[\mathbf{K}_{ff}\right]) \;, \quad \boldsymbol{\Psi} = \mathbb{E}_{q(\mathbf{X})}\left[\mathbf{K}_{fu}\right] \;, \quad \boldsymbol{\Phi} = \mathbb{E}_{q(\mathbf{X})}\left[\mathbf{K}_{uf}\mathbf{K}_{fu}\right] \tag{28}$$

and $\mathbf{W} = \beta\mathbf{I} - \beta^2\boldsymbol{\Psi}(\beta\boldsymbol{\Phi} + \mathbf{K}_{uu})^{-1}\boldsymbol{\Psi}^\top$. This expression is straight forward to compute, as long as the covariance functions $k^{\mathcal{A}}$ and $k^{\mathcal{B}}$ are selected so that the $\{\psi, \boldsymbol{\Psi}, \boldsymbol{\Phi}\}$ quantities of equation (28) can be computed analytically. As shown in Titsias and Lawrence (2010), these statistics constitute convolutions of the covariance function with Gaussian densities and are tractable for many standard covariance functions, such as the ARD squared exponential or the linear one.

Given the above, we obtain the final variational lower bound of equation (14) by computing equation (24) for each modality (this equation is a summation of the terms of equation (27)) and subtracting the KL term as explained in equation (20).

Appendix B. Inferring a New Latent Point

Given a model which is trained so as to jointly represent two output spaces $\mathbf{Y}^{\mathcal{A}}$ and $\mathbf{Y}^{\mathcal{B}}$ with a common but factorised input space $\mathbf{X}$, we wish to generate a new (or infer a training) set of outputs $\mathbf{Y}_*^{\mathcal{B}} \in \mathbb{R}^{n_* \times p_{\mathcal{B}}}$ given a set of (potentially partially) observed test points $\mathbf{Y}_*^{\mathcal{A}} \in \mathbb{R}^{n_* \times p_{\mathcal{A}}}$. This is done in three steps, as explained in the main paper. Here we explain in more detail the first step, where we need to predict the set of latent points $\mathbf{X}^* \in \mathbb{R}^{n_* \times q}$ which is most likely to have generated $\mathbf{Y}_*^{\mathcal{A}}$.

To achieve this, we use an approximation to the posterior marginal $p(\mathbf{X}_*|\mathbf{Y}_*^{\mathcal{A}}, \mathbf{Y}^{\mathcal{A}})$, which has the same form as for the standard Bayesian GP-LVM model (Titsias and Lawrence, 2010) and is given by a variational distribution $q(\mathbf{X}_*)$ which, in turn, is a marginal of $q(\mathbf{X}, \mathbf{X}_*)$. To find $q(\mathbf{X}, \mathbf{X}_*)$ we optimise a variational lower bound $\mathcal{L}_{\mathcal{A},*}\left(q(\mathbf{X}, \mathbf{X}^*)\right)$ on the marginal likelihood $p(\mathbf{Y}^{\mathcal{A}}, \mathbf{Y}_*^{\mathcal{A}})$ which has analogous form with the training objective function (12). In specific, we ignore $\mathbf{Y}^{\mathcal{B}}$ and replace $\mathbf{Y}^{\mathcal{A}}$ with $(\mathbf{Y}^{\mathcal{A}}, \mathbf{Y}_*^{\mathcal{A}})$ and $\mathbf{X}$ with $(\mathbf{X}, \mathbf{X}_*)$ in (19) so as to get:

$$\begin{aligned}
\mathcal{L}_{\mathcal{A},*} &= \int_{\mathbf{X}_*,\mathbf{X}} p(\mathbf{Y}_*^{\mathcal{A}}, \mathbf{Y}^{\mathcal{A}}|\mathbf{X}_*, \mathbf{X}) p(\mathbf{X}_*, \mathbf{X}) \\
&\leq \int_{\mathbf{X},\mathbf{X}_*,\mathbf{U}} q(\mathbf{X}_*, \mathbf{X}) q(\mathbf{U}) \log \frac{p(\mathbf{Y}_*^{\mathcal{A}}, \mathbf{Y}^{\mathcal{A}}|\mathbf{X}_*, \mathbf{X}) p(\mathbf{X}_*, \mathbf{X})}{q(\mathbf{X}_*, \mathbf{X}) q(\mathbf{U})}.
\end{aligned} \tag{29}$$

This variational lower bound is computed exactly as described in Appendix A.

What now remains is to define $q(\mathbf{X}_*, \mathbf{X})$. At this step, the inference procedure differs depending on the type of prior used for the latent space $\mathbf{X}$. Specifically, if we use a prior that does not couple datapoints, such as a standard normal one, then we are allowed to write that $q(\mathbf{X}, \mathbf{X}_*) = \prod_{i=1}^{n} q(\mathbf{x}_{i,:}) \prod_{i,:=1}^{n_*} q(\mathbf{x}_*)$, where $q(\mathbf{x}_*) = \mathcal{N}(\mathbf{x}_* | \boldsymbol{\mu}_{i,*}, \mathbf{S}_{i,*})$. In this case, equation (29) can be broken into a sum of three terms:

$$\begin{aligned} \mathcal{L}_{\mathcal{A},*} \leq & \int_{\mathbf{X}_*, \mathbf{X}, \mathbf{U}} q(\mathbf{X}_*) q(\mathbf{X}) q(\mathbf{U}) \log \frac{p(\mathbf{Y}_*^{\mathcal{A}}, \mathbf{Y}^{\mathcal{A}} | \mathbf{X}_*, \mathbf{X})}{q(\mathbf{X}_*) q(\mathbf{X}) q(\mathbf{U})} \\ & - \mathrm{KL}\left(q(\mathbf{X}_*) \,\|\, p(\mathbf{X}_*)\right) - \mathrm{KL}\left(q(\mathbf{X}) \,\|\, p(\mathbf{X})\right). \end{aligned} \tag{30}$$

The last term is already computed during training time and does not need to be re-computed. The middle term can also be computed cheaply and in parallel. As for the first term, it has the same form as for equation (27) but augmented with the test data. Importantly, the expensive computations in this expression are in the $\{\psi, \mathbf{\Psi}, \mathbf{\Phi}\}$ statistics. However, these statistics are decomposable across data-points, meaning that we can re-use computations (expectations over $q(\mathbf{X})$) performed during training time.

On the other hand, performing inference in the dynamical model is more challenging, since $q(\mathbf{X}_*, \mathbf{X})$ is fully coupled across $\mathbf{X}$ and $\mathbf{X}_*$. Therefore, if we wish to maintain the correlation of the inputs depending on their times, we should select this distribution to only factorise across features: $q(\mathbf{X}_*, \mathbf{X}) = \prod_{j=1}^{q} \mathcal{N}(\mathbf{x}_{*,j} | \boldsymbol{\mu}_{*,j}, \mathbf{S}_{*,j})$, where $\mathbf{S}_{q,n}$ are $(n + n_*) \times (n + n_*)$ matrices. In this case, the predictive equation (29) will not break as in equation (30) and, therefore, the computational complexity is increased.

Appendix C. Top-down Predictions

In this section we describe the prediction of an output point in some modality k, given latent test posterior marginal $q(\mathbf{X}_*)$ obtained as explained in Appendix B. For simplicity, we will drop the superscsript denoting a specific modality and, instead, refer to a generic output space $\mathbf{Y}$. The quantity of interest is:

$$\begin{aligned} p(\mathbf{Y}) & \approx \int_{\mathbf{F}_*} p(\mathbf{Y}_* | \mathbf{F}_*) \int_{\mathbf{U}, \mathbf{X}_*} p(\mathbf{F}_* | \mathbf{U}, \mathbf{X}_*) q(\mathbf{U}) q(\mathbf{X}_*) \\ & = \int_{\mathbf{F}_*} p(\mathbf{Y}_* | \mathbf{F}_*) q(\mathbf{F}_*) \end{aligned} \tag{31}$$

where $q(\mathbf{F}_*)$ is found as a product of its dimensions with:

$$q(\mathbf{f}_{*,j}) = \int_{\mathbf{u}_{:,j}, \mathbf{X}_*} p(\mathbf{f}_{*,j} | \mathbf{u}_{:,j}, \mathbf{X}_*) q(\mathbf{u}_{:,j}) q(\mathbf{X}_*).$$

The above is found by first using the Gaussian $q(\mathbf{U})$ and equation (22) in order to find the intermediate result $q(\mathbf{F}_* | \mathbf{X}_*)$ and then getting the final result following Girard et al. (2003) to be:

$$\mathbb{E}(\mathbf{F}_*) = \mathbf{B}^\top \mathbf{\Psi}_* \tag{32}$$

$$\mathrm{Cov}(\mathbf{F}_*) = \mathbf{B}^\top \left(\mathbf{\Phi}_* - \mathbf{\Psi}_* \mathbf{\Psi}_*^\top \right) \mathbf{B} + \psi_* \mathbf{I} - \mathrm{tr}\left(\left(\mathbf{K}_{uu}^{-1} - (\mathbf{K}_{uu} + \beta \mathbf{\Phi})^{-1} \right) \mathbf{\Phi}_* \right) \mathbf{I}, \tag{33}$$

where $\psi_* = \mathrm{tr}\left(\mathbb{E}_{q(\mathbf{X}_*)}\left[\mathbf{K}_{**}\right]\right)$, $\mathbf{\Psi}_* = \mathbb{E}_{q(\mathbf{X}_*)}\left[\mathbf{K}_{u*}\right]$, $\mathbf{\Phi}_* = \mathbb{E}_{q(\mathbf{X}_*)}\left[\mathbf{K}_{u*}\mathbf{K}_{u*}^\top\right]$. Further, $\mathbf{B} = \beta\left(\mathbf{K}_{uu} + \beta\mathbf{\Phi}\right)^{-1}\mathbf{\Psi}^\top\mathbf{Y}$, $\mathbf{K}_{**} = k(\mathbf{X}_*, \mathbf{X}_*)$ and $\mathbf{K}_{u*} = k(\mathbf{Z}, \mathbf{X}_*)$. Notice that the above is a moment-matching approach: the true distribution for $q(\mathbf{F}_*)$ is intractable, but all of its moments are analytic (above we computed only the mean and variance). Replacing (32) and (33) into (31) we have that the predicted mean of $\mathbf{Y}_*$ is equal to $\mathbb{E}\left[\mathbf{F}_*\right]$ and the predicted covariance (for each column of $\mathbf{Y}_*$) is equal to $\mathrm{Cov}(\mathbf{F}_*) + \beta^{-1}\mathbf{I}_{n_*}$.

References

Ankur Agarwal and Bill Triggs. Recovering 3d human pose from monocular images. *IEEE transactions on pattern analysis and machine intelligence*, 28(1):44–58, 2006.

Samuel Ainsworth, Nicholas Foti, Adrian KC Lee, and Emily Fox. Interpretable vaes for nonlinear group factor analysis. *arXiv preprint arXiv:1802.06765*, 2018.

Mauricio Alvarez, David Luengo, Michalis Titsias, and Neil D Lawrence. Efficient multioutput gaussian processes through variational inducing kernels. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, pages 25–32, 2010.

Galen Andrew, Raman Arora, Jeff Bilmes, and Karen Livescu. Deep canonical correlation analysis. In *International conference on machine learning*, pages 1247–1255. PMLR, 2013.

Cedric Archambeau and Francis R Bach. Sparse probabilistic projections. In *Advances in Neural Information Processing Systems*, 2008.

Francis R Bach and Michael I Jordan. A probabilistic interpretation of canonical correlation analysis. Technical report, 2005.

M Balasubramanian. The Isomap Algorithm and Topological Stability. *Science*, 295(5552): 7a–7, January 2002.

Christopher M. Bishop. Bayesian PCA. In Michael J. Kearns, Sara A. Solla, and David A. Cohn, editors, *Advances in Neural Information Processing Systems*, volume 11, pages 482–388, Cambridge, MA, 1999. MIT Press.

Christopher M. Bishop and Gwilym D. James. Analysis of multiphase flows using dual-energy gamma densitometry and neural networks. *Nuclear Instruments and Methods in Physics Research*, A327:580–593, 1993. doi: 10.1016/0168-9002(93)90728-Z.

Michael W Browne. The maximum-likelihood solution in inter-battery factor analysis. *British Journal of Mathematical and Statistical Psychology*, 1979.

Noam A Chomsky and Jerry A Fodor. The inductivist fallacy. *Language and Learning: The Debate between Jean Piaget and Noam Chomsky*, 1980.

Michael Cox and Trevor Cox. Multidimensional scaling. *Handbook of data visualization*, January 2008.

Zhenwen Dai, Andreas Damianou, James Hensman, and Neil Lawrence. Gaussian process models with parallelization and GPU acceleration. *arXiv preprint arXiv:1410.4984*, 2014.

Andreas Damianou. Deep Gaussian processes and variational propagation of uncertainty. *PhD Thesis, University of Sheffield*, 2015.

Andreas Damianou and Neil D. Lawrence. Deep Gaussian processes. In Carlos Carvalho and Pradeep Ravikumar, editors, *Proceedings of the Sixteenth International Workshop on Artificial Intelligence and Statistics*, volume 31, pages 207–215, AZ, USA, 4 2013. JMLR W&CP 31.

Andreas Damianou, Michalis K. Titsias, and Neil D. Lawrence. Variational inference for latent variables and uncertain inputs in Gaussian processes. *Journal of Machine Learning Research*, 17, 2016.

Andreas C Damianou, Michalis Titsias, and Neil D Lawrence. Variational Gaussian Process Dynamical Systems. In *Advances in Neural Information Processing Systems*, Granda, December 2011. University of Sheffield, University of Manchester.

Andreas C Damianou, Carl Henrik Ek, Michalis Titsias, and Neil D Lawrence. Manifold Relevance Determination. In *International Conference on Machine Learning*, pages 145–152, June 2012.

Carl Henrik Ek. *Shared Gaussian Process Latent Variable Models.* PhD thesis, Oxford Brookes University, Oxford, 2009.

Carl Henrik Ek, Phil H. S. Torr, and Neil D Lawrence. Gaussian process latent variable models for human pose estimation. *International conference on Machine learning for multimodal interaction*, pages 132–143, 2007.

Carl Henrik Ek, J Rihan, Phil H. S. Torr, G Rogez, and Neil D Lawrence. Ambiguity modeling in latent spaces. *International conference on Machine learning for multimodal interaction*, pages 62–73, 2008a.

Carl Henrik Ek, Phil H. S. Torr, and Neil D Lawrence. GP-LVM for Data Consolidation. In *Neural Information Processing Systems: Workshop on Learning from multiple sources*, October 2008b.

Stefanos Eleftheriadis, Ognjen Rudovic, and Maja Pantic. View-constrained latent variable model for multi-view facial expression classification. In *International symposium on visual computing*, pages 292–303. Springer, 2014.

Stefanos Eleftheriadis, Ognjen Rudovic, and Maja Pantic. Discriminative shared gaussian processes for multiview and view-invariant facial expression recognition. *IEEE transactions on image processing*, 24(1):189–204, 2015.

Yarin Gal, Mark van der Wilk, and Carl E. Rasmussen. Distributed variational inference in sparse Gaussian process regression and latent variable models. In Zoubin Ghahramani, Max Welling, Corinna Cortes, Neil D. Lawrence, and Kilian Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 27, Cambridge, MA, 2014.

Athinodoros S Georghiades, Peter N Belhumeur, and David J Kriegman. From Few to Many: Illumination Cone Models for Face Recognition under Variable Lighting and Pose. *IEEE Transactions Pattern Analysis and Machine Intelligence*, 23(6):643–660, 2001.

Agathe Girard, Carl Edward Rasmussen, Joaquin Quiñonero Candela, and Roderick Murray-Smith. Gaussian process priors with uncertain inputs—application to multiple-step ahead time series forecasting. In Sue Becker, Sebastian Thrun, and Klaus Obermayer, editors, *Advances in Neural Information Processing Systems*, volume 15, pages 529–536, Cambridge, MA, 2003. MIT Press.

Jihun Ham, Daniel D Lee, and Lawrence K Saul. Learning high dimensional correspondences from low dimensional manifolds. In *International Conference on Machine Learning*, 2003.

Jihun Ham, D Lee, and Lawrence K Saul. Semisupervised alignment of manifolds. In *Annual Conference on Uncertainty in Artificial Intelligence*, 2005.

Jihun Ham, Ikkjin Ahn, and Daniel Lee. Learning a manifold-constrained map between image sets: applications to matching and pose estimation. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 817–824. IEEE, 2006.

James Hensman and Neil D. Lawrence. Nested variational compression in deep Gaussian processes. Technical report, University of Sheffield, 2014.

James Hensman, Nicoló Fusi, and Neil D. Lawrence. Gaussian processes for big data. In Ann Nicholson and Padhraic Smyth, editors, *Uncertainty in Artificial Intelligence*, volume 29. AUAI Press, 2013.

Matthew D Hoffman, David M Blei, Chong Wang, and John Paisley. Stochastic variational inference. 14(1):1303–1347, 2013.

Harold Hotelling. Analysis of a complex of statistical variables into principal components. *Journal of Educational Psychology*, 24(6):417–441, September 1933.

Harold Hotelling. Relations between two sets of variates. *Biometrika*, 28(3/4):321–377, 1936.

Yangqing Jia, Mathieu Salzmann, and Trevor Darrell. Factorized Latent Spaces with Structured Sparsity. In *Advances in Neural Information Processing Systems.* UC Berkeley EECS, 2010.

Yangqing Jia, Mathieu Salzmann, and Trevor Darrell. Learning cross-modality similarity for multinomial data. In *IEEE International Conference on Computer Vision*, pages 2407–2414. IEEE, 2011.

Alfredo Kalaitzis and Neil D Lawrence. Residual Component Analysis: Generalising PCA for more flexible inference in linear-Gaussian models . In *International Conference on Machine Learning*, pages 209–216, San Francisco, June 2012.

Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

Arto Klami and Samuel Kaski. Generative Models that Discover Dependencies Between Data Sets. In *IEEE Signal Processing Society Workshop on Machine Learning for Signal Processing*, pages 123–128, 2006.

Arto Klami, Seppo Virtanen, and Samuel Kaski. Bayesian Canonical Correlation Analysis. *Journal of Machine Learning Research*, 14:965–1003, April 2013.

Malte Kuss and Thore Graepel. The Geometry Of Kernel Canonical Correlation Analysis. Technical Report 108, Max Planck Institute for Biological Cybernetics, May 2003.

Neil D Lawrence. Probabilistic non-linear principal component analysis with Gaussian process latent variable models. 6:1783–1816, 2005.

Neil D Lawrence. Probabilistic Spectral Dimensionality Reduction. *Advances in Neural Information Processing Systems*, 2010.

Neil D Lawrence and J Quiñonero-Candela. Local distance preservation in the GP-LVM through back constraints. *Proceedings of the 23rd international conference on Machine learning*, pages 513–520, 2006.

Gayle Leen and C Fyfe. Learning shared and separate features from two related data sets using GPLVM's. In *Learning from Multiple Sources Workshop, Neural Information Processing*, 2008.

Kantilal V. Mardia, John T. Kent, and John M. Bibby. *Multivariate analysis*. Academic Press, London, 1979. ISBN 0-12-471252-5.

Iain Matthews, Timothy F Cootes, J Andrew Bangham, Stephen Cox, and Richard Harvey. Extraction of visual features for lipreading. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(2):198–213, 2002.

Stanley A Mulaik. A brief history of the philosophical foundations of exploratory factor analysis. *Multivariate Behavioral Research*, 22(3):267–305, 1987.

Radford M Neal. *Bayesian Learning for Neural Networks*, volume 8. New York: Springer-Verlag, 1996.

Mihalis A Nicolaou, Vladimir Pavlovic, and Maja Pantic. Dynamic probabilistic cca for analysis of affective behavior and fusion of continuous annotations. *IEEE transactions on pattern analysis and machine intelligence*, 36(7):1299–1311, 2014.

Manfred Opper and Cedric Archambeau. The variational Gaussian approximation revisited. *Neural Computation*, 21(3):786–792, 2009.

Karl Pearson. On lines and planes of closest fit to systems of points in space. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 2(11):559–572, 1901.

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

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

Sam T Roweis and Lawrence K Saul. Nonlinear dimensionality reduction by locally linear embedding. *Science*, (290), 2000.

Jan Rupnik and John Shawe-Taylor. Multi-view canonical correlation analysis. In *Conference on Data Mining and Data Warehouses (SiKDD 2010)*, pages 1–4, 2010.

Mathieu Salzmann, Carl Henrik Ek, Raquel Urtasun, and Trevor Darrell. Factorized Orthogonal Latent Spaces. In *International Conference on Artificial Inteligence and Statistical Learning*, pages 701–708, 2010.

Abhishek Sharma, Abhishek Kumar, Hal Daume, and David W Jacobs. Generalized multiview analysis: A discriminative latent space. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pages 2160–2167. IEEE, 2012.

Aaron Shon, Keith Grochow, Aaron Hertzmann, and Rajesh P Rao. Learning shared latent structure for image synthesis and robotic imitation. In *Advances in neural information processing systems*, pages 1233–1240, 2006.

Charles Spearman. "General Intelligence", Objectively Determined and Measured. *The American Journal of Psychology*, 15(2):201–292, 1904.

Joshua B Tenenbaum, Vin de Silva, and John C Langford. A Global Geometric Framework for Nonlinear Dimensionality Reduction. *Science*, 290(5500):2319–2323, December 2000.

Michael E. Tipping. The relevance vector machine. In Sara A. Solla, Todd K. Leen, and Klaus-Robert Müller, editors, *Advances in Neural Information Processing Systems*, volume 12, pages 652–658, Cambridge, MA, 2000. MIT Press.

Michael E Tipping and Christopher M Bishop. Probabilistic principal component analysis. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 61(3):611–622, 1999.

Michalis Titsias. Variational learning of inducing variables in sparse Gaussian processes. In *International Conference on Artificial Inteligence and Statistical Learning*, pages 567–574, 2009.

Michalis Titsias and Neil D Lawrence. Bayesian Gaussian Process Latent Variable Model. In *International Conference on Artificial Inteligence and Statistical Learning*, pages 844–851, 2010.

Ledyard R Tucker. An Inter-Battery Method of Factory Analysis. *Psychometrika*, 23, June 1958.

Raquel Urtasun and Trevor Darrell. Discriminative Gaussian process latent variable model for classification. *International Conference on Machine Learning*, page 934, 2007.

Raquel Urtasun, David J Fleet, Andreas Geiger, Jovan Popovic, Trevor Darrell, and Neil D Lawrence. Topologically-Constrained Latent Variable Models. *International Conference on Machine Learning*, 2008.

Seppo Virtanen, Arto Klami, and Samuel Kaski. Bayesian cca via group sparsity. In *Proceedings of the 28th International Conference on International Conference on Machine Learning*, pages 457–464. Omnipress, 2011.

Seppo Virtanen, Yangqing Jia, Arto Klami, and Trevor Darrell. Factorized multi-modal topic model. *arXiv preprint arXiv:1210.4920*, 2012.

Jack M Wang, David J Fleet, and Aaron Hertzmann. Gaussian process dynamical models for human motion. *IEEE transactions on pattern analysis and machine intelligence*, 30 (2):283–298, 2008.

Weiran Wang, Raman Arora, Karen Livescu, and Jeff Bilmes. On deep multi-view representation learning. In *International conference on machine learning*, pages 1083–1092, 2015.

Kilian Q Weinberger and L K Saul. Unsupervised Learning of Image Manifolds by Semidefinite Programming. *International journal of computer vision*, 70:77–90, 2006.

Kilian Q Weinberger, Fei Sha, and Lawrence K Saul. Learning a kernel matrix for nonlinear dimensionality reduction. In *International Conference on Machine Learning*, pages 106–113. ACM, July 2004.

Shuicheng Yan, Dong Xu, Benyu Zhang, Hong-jiang Zhang, Qiang Yang, and Stephen Lin. Graph Embedding and Extensions: A General Framework for Dimensionality Reduction. *IEEE Transactions Pattern Analysis and Machine Intelligence*, 29(1):40–51, 2007.

Cheng Zhang, Carl Henrik Ek, Andreas C Damianou, and Hedvig Kjellström. Factorized Topic Models. In *International Conference on Learning Representations*, April 2013.