Unsupervised Representation Learning with Recognition-Parametrised Probabilistic Models

William I. Walker* **Hugo Soulat*** **Changmin Yu** **Maneesh Sahani***
Gatsby Computational Neuroscience Unit, University College London

Abstract

We introduce a new approach to probabilistic unsupervised learning based on the *recognition-parametrised model* (RPM): a normalised semi-parametric hypothesis class for joint distributions over observed and latent variables. Under the key assumption that observations are conditionally independent given latents, the RPM combines parametric prior and observation-conditioned latent distributions with non-parametric observation marginals. This approach leads to a flexible learnt recognition model capturing latent dependence between observations, without the need for an explicit, parametric generative model. The RPM admits exact maximum-likelihood learning for discrete latents, even for powerful neural-network-based recognition. We develop effective approximations applicable in the continuous-latent case. Experiments demonstrate the effectiveness of the RPM on high-dimensional data, learning image classification from weak indirect supervision; direct image-level latent Dirichlet allocation; and recognition-parametrised Gaussian process factor analysis (RP-GPFA) applied to multi-factorial spatiotemporal datasets. The RPM provides a powerful framework to discover meaningful latent structure underlying observational data, a function critical to both animal and artificial intelligence.

1 INTRODUCTION

Unsupervised representation learning plays a key role in systems that seek to learn and estimate world state and latent structure from observations, including those that address real-world reinforcement learning and robotics, tracking, semi-supervised task learning, and scientific discovery. Such systems have long been underpinned by the methods of probabilistic latent-variable modelling (Bishop, 2006; Barber, 2011; Murphy, 2022). In the most common approach, a model describes a family of distributions over a set of latent variables and the conditional dependence of the observed variables on those latents. Together, these define a directed acyclic graphical model (Table 1) or DAG. Marginalising over the latents then results in a hypothesis class of joint distributions on the observations. Distribution parameters can be found by standard estimation techniques, identifying a model within the class (or a posterior over models) that best matches the data distribution.

Although latent-variable models may also be used for sample simulation (Goodfellow et al., 2014; Kingma et al., 2021) or density estimation (Rezende and Mohamed, 2015), they play a key role in representation learning. Many data sets exhibit complex dependence amongst observations, which arise through common influences from unobserved but causally relevant features of the data generating process. By estimating models that render observations independent conditioned on latent state it is often possible to tease out and represent such underlying features. Indeed, it is this assumption of latent-conditioned independence—between the inputs to different sensors, between sensor modalities, or between future and past—that provides the basis for learning underlying structure in the absence of strong distributional assumptions.

In this representation-learning view, the generative model serves to encode structural priors about dependence and distribution, and the associated marginal on observations underlies the choice of estimation objective, such as likelihood. However, once learnt, neither is used directly. Instead, the model structure and parameters are used for inference or *recognition*—to estimate the state of the world from sensory data (Helmholtz, 1867). This mismatch between the way the model is specified and how it is eventually used poses a challenge to effective learning. Generative models that are sufficiently complex, flexible and non-linear to parametrise real-world observations do not generally admit efficient tractable inference, and so recognition

Proceedings of the 26th International Conference on Artificial Intelligence and Statistics (AISTATS) 2023, Valencia, Spain. PMLR: Volume 206.

* These authors contributed equally

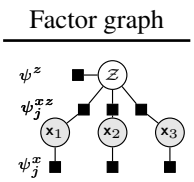
is often approximated (e.g. Dayan et al., 1995; Jordan et al., 1999; Rezende et al., 2014; Kingma and Welling, 2014). These approximate methods lead to biases in parameter estimates (Turner and Sahani, 2011), and result in learnt representations that do not, in fact, match the learnt generative model.

Our goal here is to address such challenges to probabilistic representation learning. We do so by introducing a form of semi-parametric model in which an explicit parametrisation of the *recognition* process is paired with a simplified non-parametric description of the observations. This *recognition parametrised model* (RPM) defines a properly normalised joint distribution, and thus (implicitly) a proper semi-parametric marginal distribution on observations. Maximum-likelihood (ML) learning can be achieved exactly for models with discrete latent variables (and a tractable internal graph), whilst in other settings it depends on potentially milder approximations than do methods that pair recognition modelling with explicit generative models such as the Helmholtz machine (Dayan et al., 1995; Vértes and Sahani, 2018) or variational autoencoder (VAE) (Rezende et al., 2014; Kingma and Welling, 2014). The RPM allows many different distributional and structural assumptions on the latent variables to be combined with a recognition parametrisation, and pairs effectively with established techniques such as variational message passing (Winn et al., 2005) or variational Bayesian learning (Attias, 2000) to estimate models with complex latent dependence.

Below, we first present a general formulation of the RPM (Section 2), discuss inference and learning in this general case (Section 3), and relate it to existing models (Section 4). Thereafter we demonstrate the breadth of the framework by instantiating different conditional structures and prior assumptions on the latent factors, and applying these to appropriate data sets (Section 5).

2 THE RPM

Consider a set of observed (possibly vector-valued) random variables $\mathcal{X} = \{\mathbf{x}_j : j = 1 \dots J\}$. We seek to learn a model based on a set of underlying latent variables $\mathcal{Z} = \{\mathbf{z}_l : l = 1 \dots L\}$, given which the different $\mathbf{x}_j$ are conditionally independent. These variables may be loosely interpreted as causally relevant features responsible for the statistical interdependence of the observations. Our goal is to learn the joint distribution of the latents, along with a parametrised model that infers a suitable belief over their values from observations. We use the symbol P (often with subscripts) to indicate complete (normalised) model distributions, and italicised symbols for factors within the models (Table 1), noting where these are individually normalised. We write $\mathcal{X}^{(n)} = \{\mathbf{x}_j^{(n)} : j = 1 \dots J\}$ for the nth joint data observation, and $\mathbb{X}^{(N)} = \{\mathcal{X}^{(1)} \dots \mathcal{X}^{(N)}\}$ for the entire set of N observations.



Factor graph	DAG	RPM
$\psi^z(\mathcal{Z})$	$p_{\theta z}(\mathcal{Z})$	$p_{\theta z}(\mathcal{Z})$
$\psi_j^{xz}(\mathbf{x}_j, \mathcal{Z})$	$p_{\theta_j}(\mathbf{x}_j \vert \mathcal{Z})$	$\frac{f_{\theta j}(\mathcal{Z}\vert\mathbf{x}_j)}{F_{\theta j}(\mathcal{Z})}$
$\psi_j^x(\mathbf{x}_j)$	1	$p_{0j}(\mathbf{x}_j)=\frac{1}{N}\sum_n \delta(\mathbf{x}_j-\mathbf{x}_j^{(n)})$

Table 1: Conditional independence models and factor definitions. Left column shows a generic factor graph with corresponding unnormalised factors. Central column shows a directed graph. Right column shows the corresponding RPM factors.

The conditional independence assumption implies a factorisation (and corresponding factor graph)

$$\mathsf{P}(\mathcal{X}, \mathcal{Z}) \propto \psi^z(\mathcal{Z}) \prod_j \psi_j^x(\mathbf{x}_j) \prod_j \psi_j^{xz}(\mathbf{x}_j, \mathcal{Z}) \,. \quad (1)$$

In the RPM, these factors are parametrised as follows:

$\psi^z(\mathcal{Z}) \to p_{\theta z}(\mathcal{Z})$: a normalised distribution on the latent variables. For multivariate $\mathcal{Z}$ this may itself be factored, with a corresponding latent graphical model.

$\psi_j^x(\mathbf{x}_j) \to p_{0j}(\mathbf{x}_j)$: a summary of the empirical marginal distribution of each observed variable, with the property that it converges to the true distribution of $\mathbf{x}_j$ in the limit of infinite data. In this paper we take $p_{0j}(\mathbf{x}_j) = \frac{1}{N}\sum_n \delta\big(\mathbf{x}_j - \mathbf{x}_j^{(n)}\big)$, the empirical measure with atoms at the N data points $\mathbf{x}_j^{(n)}$. However, the key definitions and results extend to alternatives, such as an adaptive kernel density estimate with kernel width that approaches 0 as N grows, or (if known) a member of the true marginal distributional family specified by a sufficient statistic of the data. The key is that p_{0j} is determined by the corresponding observations, with learning of the joint distribution focused on the other factors of the RPM.

$\psi^{xz}(\mathbf{x}_j, \mathcal{Z}) \to \frac{f_{\theta j}(\mathcal{Z}\vert\mathbf{x}_j)}{\int d\mathbf{x}_j \, p_{0j}(\mathbf{x}_j) f_{\theta j}(\mathcal{Z}\vert\mathbf{x}_j)}$ where $f_{\theta j}(\mathcal{Z}\vert\mathbf{x}_j)$ is a parametrised normalised distribution possibly, but not necessarily, defined on only a subset of the $\mathcal{Z}$ (often on a single $\mathbf{z}_l$). We write $F_{\theta j}(\mathcal{Z})$ for the mixture with respect to p_{0j} that appears in the denominator. The numerator terms $f_{\theta j}(\mathcal{Z}\vert\mathbf{x}_j)$ will be referred to as *recognition factors*.

Thus the full joint RPM model becomes

$$\mathsf{P}_{\theta, \mathbb{X}^{(N)}}(\mathcal{X}, \mathcal{Z}) = p_{\theta z}(\mathcal{Z}) \prod_j \left(p_{0j}(\mathbf{x}_j) \frac{f_{\theta j}(\mathcal{Z}\vert\mathbf{x}_j)}{F_{\theta j}(\mathcal{Z})} \right) , \quad (2)$$

where the observed dataset $\mathbb{X}^{(N)}$ appears in the subscript

to emphasise that the model parametrisation itself depends on the data through p_{0j} and so $F_{\theta j}$.

With this choice of parametrisation we have

$$\begin{aligned}\mathsf{P}_{\theta,\mathbb{X}^{(N)}}(\mathcal{Z}) &= \prod_j \int d\mathbf{x}_j \left(\frac{p_{0j}(\mathbf{x}_j) f_{\theta j}(\mathcal{Z}|\mathbf{x}_j)}{F_{\theta j}(\mathcal{Z})}\right) p_{\theta z}(\mathcal{Z}) \\ &= p_{\theta z}(\mathcal{Z})\,, \end{aligned} \tag{3}$$

so that the parametrised factor on the latents corresponds to the prior distribution implied by the joint (as is also the case for a DAG). This result confirms that the RPM is properly normalised. The posterior

$$\begin{aligned}\mathsf{P}_{\theta,\mathbb{X}^{(N)}}\left(\mathcal{Z}|\mathcal{X}^{(n)}\right) &\propto \prod_j \frac{f_{\theta j}(\mathcal{Z}|\mathbf{x}_j^{(n)})}{\int d\mathbf{x}_j\, p_{0j}(\mathbf{x}_j) f_{\theta j}(\mathcal{Z}|\mathbf{x}_j)} p_{\theta z}(\mathcal{Z}) \\ &= \frac{1}{W_\theta(\mathcal{X}^{(n)})} \prod_j f_{\theta j}(\mathcal{Z}|\mathbf{x}_j^{(n)}) \, \frac{p_{\theta z}(\mathcal{Z})}{\prod_j F_{\theta j}(\mathcal{Z})} \end{aligned} \tag{4}$$

can be found by normalising the product of learnt factors. The normaliser $W_\theta(\mathcal{X})$ also gives the relative density of the implicit RPM joint on the observed variables, $\mathsf{P}_{\theta,\mathbb{X}^{(N)}}(\mathcal{X}) = \prod_j p_{0j}(\mathbf{x}_j) W_\theta(\mathcal{X})$. The joint is supported on the Cartesian product of the supports of $p_{0j}(\mathbf{x}_j)$—an irregular grid of atoms for atomic p_{0j} as we assume here.

Exponential-Family Parametrisation Beyond the need for normalisation, the factors $p_{\theta z}$ and $f_{\theta j}(\cdot|\mathbf{x}_j^{(n)})$ in the RPM as defined above can be chosen freely. In practice, we will often assume that they lie within a common exponential family with combined sufficient statistic $t(\mathcal{Z})$ defined on all the latent variables, and log-normaliser Φ. The prior factor will be taken to have natural parameter η_0. The natural parameters of the recognition factors are now parametrised functions of the observations given by $\eta_j(\mathbf{x}_j^{(n)})$ (which may be constrained to have constant outputs along some dimensions when the recognition factors target a subset of the latent variables). See also Table A1.

With these choices (and assuming uniform base measure for $p_{\theta z}$), we can write the implied *generative* conditionals of the RPM as:

$$\begin{aligned}\mathsf{P}_{\theta,\mathbb{X}^{(N)}}(\mathbf{x}_j|\mathcal{Z}) &= p_{0j}(\mathbf{x}_j) \frac{e^{\eta_j(\mathbf{x}_j)^\mathsf{T} t(\mathcal{Z}) - \Phi(\eta_j(\mathbf{x}_j))}}{F_{\theta j}(\mathcal{Z})} \\ &= \chi_j(\mathbf{x}_j)\, e^{t(\mathcal{Z})^\mathsf{T} \eta_j(\mathbf{x}_j) - \Phi_{\mathbf{x}_j}(t(\mathcal{Z}))} \end{aligned}$$

with

$$\chi_j(\mathbf{x}_j) = \frac{1}{C}\, p_0(\mathbf{x}_j) e^{-\Phi(\eta_j(\mathbf{x}_j))}$$

for constant C and

$$\begin{aligned}\Phi_{\mathbf{x}_j}(t(\mathcal{Z})) &= \log \int d\mathbf{x}_j\, \frac{1}{C}\, p_0(\mathbf{x}_j) e^{-\Phi(\eta_j(\mathbf{x}_j))} e^{\eta_j(\mathbf{x}_j)^\mathsf{T} t(\mathcal{Z})} \\ &= \log \int d\mathbf{x}_j\, \chi_j(\mathbf{x}_j) e^{\eta_j(\mathbf{x}_j)^\mathsf{T} t(\mathcal{Z})}\,. \end{aligned}$$

Thus, this form of RPM induces an exponential family conditional on each $\mathbf{x}_j$ in which the parameters of $\eta_j(\mathbf{x}_j)$ determine both the sufficient statistic and the base measure, the latter also depending on the observed marginal. This expression underlines the expressiveness of the RPM with flexibly parametrised recognition factors.

3 MAXIMUM-LIKELIHOOD LEARNING

3.1 Variational Free Energy

As is the case for other latent-variable models, ML estimation in the RPM can be achieved using the Expectation-Maximisation (EM) algorithm and related methods. We adopt the viewpoint of Neal and Hinton (1998) and frame EM as coordinate ascent of a variational free energy derived by applying Jensen's inequality to the log likelihood:

$$\begin{aligned}\sum_n \log \mathsf{P}_{\theta,\mathbb{X}^{(N)}}\left(\mathcal{X}^{(n)}\right) &\geq \mathcal{F}(\theta, q(\{\mathcal{Z}^{(n)}\})) \\ &= \left\langle \sum_n \log \mathsf{P}_{\theta,\mathbb{X}^{(N)}}\left(\mathcal{X}^{(n)}, \mathcal{Z}^{(n)}\right)\right\rangle + \mathbf{H}[q] \end{aligned}$$

where angle brackets indicate expectations over the variational distribution q and $\mathbf{H}[\cdot]$ is the entropy. Dropping the term in p_{0j} which is independent of θ and q, $\mathcal{F}$ can be written in terms of Kullback-Leibler (KL) divergences as

$$\begin{aligned}-\mathcal{F}(\theta, \{q^{(n)}(\mathcal{Z}^{(n)})\}) &\underset{+C}{=} \sum_n \mathbf{KL}\left[q^{(n)} \middle\| p_{\theta z}\right] \\ &+ \sum_{nj} \mathbf{KL}\left[q^{(n)} \middle\| f_{\theta j}(\cdot|\mathbf{x}_j^{(n)})\right] - \sum_{nj} \mathbf{KL}\left[q^{(n)} \middle\| F_{\theta j}\right], \end{aligned} \tag{5}$$

where we have used the fact that the optimal q has the form $\prod_n q^{(n)}(\mathcal{Z}^{(n)})$, and the distributions in the latter KL divergences range over only the $\mathbf{z}_l$ that are targeted by the corresponding $f_{\theta j}$ or $F_{\theta j}$.

Alternating maximisation of $\mathcal{F}$ with respect to q (the "E-step") and θ ("M-step") will converge to a (possibly local) mode of the likelihood, provided that each maximum can be achieved. This is straightforward in cases where the latent targets of each $f_{\theta j}$ are discrete-valued variables (so that $F_{\theta j}(\mathbf{z}_k) = \int d\mathbf{x}_j\, p_{0j}(\mathbf{x}_j) f_{\theta j}(\mathbf{z}_k|\mathbf{x}_j)$ is an easily computed discrete distribution) and $p_{\theta z}$ has conjugate structure and sufficiently small junction tree width to be computationally tractable. Examples of such exact ML learning in an RPM are explored below in Sections 5.1 and 5.2.

3.2 E-step for Continuous-Valued Latent Variables

The situation is more complex when the latent-variable targets of $f_{\theta j}$ are continuous-valued. Even assuming that the graphical structure and factor potentials that compose $p_{\theta z}(\mathcal{Z})$ allow tractable marginalisation, and that the terms $f_{\theta j}(\mathcal{Z}|\mathbf{x}_j)$ provide conjugate factors, the inverse expectation factors $\left(\int d\mathbf{x}_j\, p_{0j}(\mathbf{x}_j) f_{\theta j}(\mathcal{Z}|\mathbf{x}_j)\right)^{-1}$ will generally

break conjugacy and thus analytic tractability. A natural approach in this case is to constrain $q^{(n)}$ to live within the conjugate family defined by $p_{\theta z}$ and $f_{\theta j}(\cdot|\mathbf{x}_j)$. This constraint renders the first two KL divergences in Eq. (5) tractable, but requires approximation to evaluate the third. However, by contrast to the analogous standard parametric variational assumption made in the context of non-conjugate parametrised *generative* models (e.g. the VAE) the impact of the constraint in the RPM may be negligible in the large-data in-model limit. Specifically, if the true posterior on $\mathcal{Z}$ lies within the parametric class of $f_{\theta j}$ then, in the limit of large data one potential set of ML parameters will be such that $F_{\theta j}(\mathcal{Z}) = \int d\mathbf{x}_j\, p_{0j}(\mathbf{x}_j) f_{\theta j}(\mathcal{Z}|\mathbf{x}_j) \rightarrow p_{\theta z}(\mathcal{Z})$. This implies that the penalty of *assuming* that $F_{\theta j}(\mathcal{Z})$ has the exponential family form will become negligible in the large data limit for this in-model conjugate case.

There are at least three approaches to optimising q in the continuous case. We consider the exponential family parametrisation, and constrain $q(\mathcal{Z}^{(n)})$ to be in the same family, with natural parameter $\eta_q^{(n)}$. We will sometimes also require the moment parameters of the various distributions (i.e. the expectations of $t(\mathcal{Z})$ under the corresponding natural parameter). These will be written μ_0, $\mu_j(\mathbf{x}_j^{(n)})$ and $\mu_q^{(n)}$ for the prior, recognition factors and $q^{(n)}$ respectively. See Table A1.

Reparametrised Monte-Carlo The first approach adopts a strategy used extensively in the VAE literature where it is known as "reparametrisation". It is often possible to express a sample from the exponential family of interest as a parametrised function of a sample from a fixed distribution. A common example is the normal family, where samples from $\mathcal{N}(\boldsymbol{\mu}, \Sigma)$ can be expressed in terms of $\boldsymbol{\epsilon}_i \sim \mathcal{N}(0, I)$ as $\Sigma^{\frac{1}{2}}\boldsymbol{\epsilon}_i + \boldsymbol{\mu}$. This allows a Monte-Carlo estimate of the expectation of $F_{\theta j}$ to be written as a function of the parameters $\eta_q^{(n)}$ and a fixed set of samples $\{\boldsymbol{\epsilon}_i\}$. The other expectations can be evaluated analytically under our conjugacy assumptions. Thus, it becomes possible to optimise $\mathcal{F}$ with respect to $\eta_q^{(n)}$ by gradient ascent. While accurate with large numbers of samples, this approach may be computationally expensive for high-dimensional problems.

Second-order Approximation An efficient approximation of $\langle \log F_{\theta j} \rangle$ can be obtained by generalising an approach taken by Braun and McAuliffe (2010). We expand $\log F_{\theta j}(\mathcal{Z})$ to second order in $t(\mathcal{Z})$ around its expectation $\mu_q^{(n)}$ under the variational distribution $q^{(n)}$. Then, writing $V_q^{(n)}$ for the variance of $t(\mathcal{Z})$ under $q^{(n)}$, we have

$$\left\langle \log F_{\theta j}(\mathcal{Z}) \right\rangle \approx \log \frac{1}{N} \sum_{m=1}^{N} e^{\eta_j(\mathbf{x}_j^{(m)})^\top \mu_q^{(n)} - \Phi(\eta_j(\mathbf{x}_j^{(m)}))} + \frac{1}{2}\mathrm{tr}\left(\boldsymbol{\eta}_j^\top V_q^{(n)} \boldsymbol{\eta}_j \left[\mathrm{diag}(\boldsymbol{\pi}_j^{(n)}) - \boldsymbol{\pi}_j^{(n)} \boldsymbol{\pi}_j^{(n)\top}\right]\right) \quad (6)$$

where $\boldsymbol{\eta}_j = \left[\eta_j(\mathbf{x}_j^{(1)}), \ldots, \eta_j(\mathbf{x}_j^{(N)})\right]$ and $\boldsymbol{\pi}_j^{(n)}$ is an N-dimensional vector with components (for $m = 1 \ldots N$)

$$\pi_{mj}^{(n)} = \frac{e^{\eta_j(\mathbf{x}_j^{(m)})^\intercal \mu_q^{(n)} - \Phi(\eta_j(\mathbf{x}_j^{(m)}))}}{\sum_p e^{\eta(\eta_j(\mathbf{x}_j^{(p)}))^\intercal \mu_q^{(n)} - \Phi(\eta_j(\mathbf{x}_j^{(p)}))}}\,. \quad (7)$$

This approximation no longer guarantees a lower bound on the free energy but we demonstrate its efficacy in practice.

Interior Variational Bound A third approach introduces auxiliary variational parameters to obtain a second bound on $\mathcal{F}$. Focusing on the $F_{\theta j}$-dependent terms as above (and again using angle brackets for expectations under q) we introduce functions $\tilde{f}_j^{(n)}(\mathcal{Z})$ and use Jensen's inequality to write

$$\left\langle \log \frac{f_{\theta j}(\cdot|\mathbf{x}_j^{(n)})}{F_{\theta j}} \right\rangle \geq \left\langle \log \frac{f_{\theta j}(\cdot|\mathbf{x}_j^{(n)})}{\tilde{f}_j^{(n)} q^{(n)}} \right\rangle - \log \left\langle \frac{F_{\theta j}}{\tilde{f}_j^{(n)} q^{(n)}} \right\rangle = -\mathbf{KL}\left[q^{(n)} \middle\| f_{\theta j}(\cdot|\mathbf{x}_j^{(n)})\right] - \left\langle \log \tilde{f}_j^{(n)} \right\rangle - \log \Gamma_j^{(n)}\,, \quad (8)$$

where $\tilde{\Gamma}_j^{(n)} = \int d\mathcal{Z}\, F_{\theta j}(\mathcal{Z}) / \tilde{f}_j^{(n)}(\mathcal{Z})$. Inserting this bound into Eq. (5) and rearranging gives

$$\widetilde{\mathcal{F}} = \sum_n \log \mathsf{P}_{\theta, \mathbb{X}^{(N)}}\left(\mathcal{X}^{(n)}\right) - \sum_n \mathbf{KL}\left[q^{(n)} \middle\| \mathsf{P}_{\theta, \mathbb{X}^{(N)}}\left(\cdot|\mathcal{X}^{(n)}\right)\right] - \sum_{nj} \mathbf{KL}\left[q^{(n)} \middle\| \frac{1}{\tilde{\Gamma}_j^{(n)}} \frac{F_{\theta j}}{\tilde{f}_j^{(n)}}\right] \quad (9)$$

with $\widetilde{\mathcal{F}}\left(\theta, q, \{\tilde{f}_j^{(n)}\}\right) \leq \mathcal{F}(\theta, q)$ lower-bounding the conventional free energy. If we now choose $\tilde{f}_j^{(n)}(\mathcal{Z}) = \exp\left(t(\mathcal{Z})^\intercal \tilde{\eta}_j^{(n)}\right)$ with the constraint that $\eta_j(\mathbf{x}_j^{(m)}) - \tilde{\eta}_j^{(n)}$ is a valid natural parameter for all (m, n), then the right hand-side of Eq. (8) is closed-form. Furthermore, if $F_{\theta j}$ approaches the exponential family with statistic $t(\mathcal{Z})$, as in the in-model conjugate case, then it will be possible to choose $\tilde{\eta}_j^{(n)}$ to set the final KL-divergence in Eq. (9) close to 0, restoring a tight bound.

3.3 M-step

The generalised M-step of EM increases the free energy with respect to the parameters θ while holding the variational distribution q fixed (Neal and Hinton, 1998). For many RPMs, the parameter vector will divide into disjoint

subsets that determine $p_{\theta z}$ and the $f_{\theta j}$ (possibly shared for multiple j). In this case, the update for the $p_{\theta z}$ group will be broadly identical to the usual EM update. For $f_{\theta j}$ the update requires gradients of both $\left\langle \log f_{\theta j}(\mathcal{Z}^{(n)}|\mathbf{x}_j^{(n)})\right\rangle$ and $\left\langle \log F_{\theta j}(\mathcal{Z}^{(n)})\right\rangle$.

For discrete-valued latent variables where the E-step is exact, the corresponding M-step is straightforward, possibly incorporating backpropagation of gradients where $f_{\theta j}$ has neural network form and amenable to automated gradient-based optimisation. For continuous-valued latent variables the corresponding step depends on the E-step approach used. Reparametrisation and the second-order approximation both provide an explicit estimate of $\mathcal{F}$ which can be increased directly. When employing the interior variational bound, we instead increase the term $\widetilde{\mathcal{F}}$ (see Appendix).

It is worth noting that, although the discussion above has focused on cases where the latent distribution $p_{\theta z}$ is tractable (in the sense that the marginals needed for learning can be computed efficiently) the RPM can also be seamlessly combined with standard approximate variational inference and learning methods. This includes variational Bayesian methods to obtain approximate posteriors on parameters.

4 RELATIONSHIPS TO OTHER MODELS

Dual Generation-Recognition Models Many archetypal learning architectures for latent-variable models, including the variational autoencoder (VAE; Kingma and Welling, 2014) and Helmholtz machine (Dayan et al., 1995), employ parametrised recognition networks in support of learning an explicit generative model for data. The associated objective functions are usually derived from the likelihood of the generative parameters, with the recognition model supplying 'E-step' inference in an EM-like approach (Neal and Hinton, 1998; Jordan et al., 1999). However, the true posterior distribution over the latents implied by the generative structure rarely lies within the class of functions described by the recognition model parametrisation. This mismatch induces an intrinsic bias in the estimates of the generative parameters (e.g. Turner and Sahani, 2011), which can be seen either as a necessary compromise or (for the VAE) as a reframing of the objective function from the likelihood to the variational lower bound (Jordan et al., 1999).

Recent work has sought to lessen the bias by introducing a more flexible posterior representation (Rezende and Mohamed, 2015; Vértes and Sahani, 2018; Wenliang et al., 2020), or tighter variational bounds than the classic free energy form (Burda et al., 2016; Maddison et al., 2017; Masrani et al., 2019). However, these extensions retain the emphasis on approximate ML estimation of a parametric *generative* process with a specific noise model, potentially guiding the latent representation towards details of individual data elements that may not be representationally useful. By contrast, the RPM likelihood emphasises latent structure that captures dependence between data elements, dispensing with a parametrisation of the marginal distributions of individual elements and corresponding noise. Intuition suggests that this joint structure is most likely to reflect latent "causal" elements, and so may be most valuable for decision making. Furthermore, although approximation is necessary for RPM models with continuous-valued latent variables, the impact of the approximation will not always persist as the data set grows (see the discussion of in-model conjugacy in Section 3.2).

The RPM is also directly compatible with graphical (i.e. conditional-independence-based) prior structure within the latents, as explored in various models below. Analogous structured versions have been explored in the context of generation-recognition parametrisations; but complications arise from the need to backpropagate gradients through message passing in the latent graph of structured VAEs (Johnson et al., 2016) or from the need to approximate message passing in complex Helmholtz machines (Vértes and Sahani, 2019; Wenliang and Sahani, 2019).

Undirected Models Latent-variable models may also be parametrised in a factored form corresponding to an undirected graph, exemplified by the Boltzmann machine (Ackley et al., 1985). Factor models with observations conditionally independent given the latents *and vice versa* (such as the restricted Boltzmann machine (RBM; Smolensky, 1986; Hinton, 2002) or exponential-family harmonium (Welling et al., 2004)) may be viewed as restricted and unnormalised variants of the RPM. Inference follows directly from the parametric form, but only because the latents are also conditionally independent given the observations. In other words, whereas the RPM can incorporate factors that link *all* the latents to each observation separately (see Eq. (1)), the RBM is restricted to pairwise factors linking individual latents and observations, or more generally factors that link disjoint subsets of each. Furthermore, the marginal prior on the latents is implicit and typically inaccessible and ML learning requires sampling from the model, most often by Markov-chain methods. Again this contrasts with the efficient ML learning of the RPM.

Noise-contrastive Estimation and InfoNCE An alternative to ML estimation is often applied to "energy-based" models, where an unnormalised data density is expressed as a parametrised non-negative function of the observations (the logarithm of this function is the "energy"). The idea behind noise-contrastive estimation (Gutmann and Hyvärinen, 2010) is to train the energy as though it were the log-odds of a classifier that seeks to distinguish genuine observations from corrupted ones. This makes sense because,

in the large-data limit, the optimal log-odds values correspond to the ratio of the model log-likelihood on the genuine data to that on the corruptions. One common form of corruption is to break each observation into two components and shuffle these components around. In this case, known as InfoNCE (Oord et al., 2018), the log-odds-like cost function approaches the mutual information between the components.

Recall that the RPM data measure is defined on the cross-product of the empirical marginal summaries $p_{0j}(\mathbf{x}_j)$, weighted by $W_\theta(\mathcal{X})$. RPM learning can thus be viewed as a process of maximising weights on the observations, which—as the distribution is normalised—must come at the expense of weights elsewhere. Thus, with the empirical delta-function measure, the RPM also learns to contrast real observations from shuffled versions. Indeed, this link between InfoNCE and a probabilistic model has been noted previously (Aitchison and Ganev, 2023), though the model proposed there was based on the (unknown) true data marginals rather than the empirical measures, and so remained intractable.

The "shuffling" in the RPM is implicit, and involves all J conditionally independent observed variables rather than just pairs. Furthermore, the normalised latent variable formulation (missing in energy-based approaches) provides access to efficient message-passing inference in complex models, as well as to variational and other well-developed tools of learning in probabilistic graphical models. And the learnt recognition model provides proper posterior beliefs over latent variables which can, as argued above, form the basis of optimal Bayesian decision making.

5 EXPERIMENTS

We demonstrate the flexibility of the RPM on a range of discrete- and continuous-latent problems: weakly supervised categorisation, a pixel-level extension of Latent-Dirichlet Allocation (LDA) (Blei et al., 2003) to images, and non-linear recognition-parametrised Gaussian Process Factor Analysis (RP-GPFA) (Yu et al., 2008; Duncker and Sahani, 2018).

The RPM performance was compared to that of appropriate VAEs to provide the most appropriate baseline. Both VAE and RPM are normalised probabilistic models where we could equate distributional assumptions and recognition architecture. In all experiments the training data, prior and recognition architecture were identical for RPM and VAE. The only differences were in the generative model that had to be instantiated for the VAE (which was set to an artificial neural network plus noise), and the corresponding learning algorithms. Derivations and details are provided in the appendices. A comparison of compute time for two of the experiments is shown in Fig. A3.

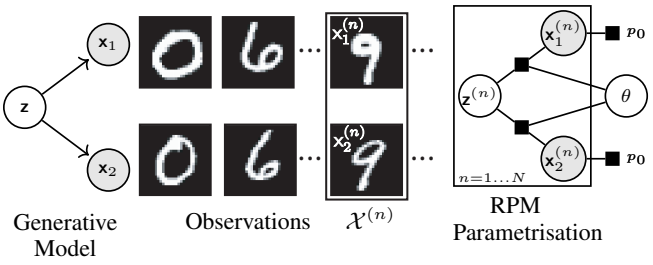


Figure 1: Peer-supervised learning. Each pair of observations $\mathcal{X} = \{\mathbf{x}_1, \mathbf{x}_2\}$ is conditionally independent given their shared digit identity $\mathbf{z}$.

VQ-VAE	GS-VAE	GS-S VQ-VAE	RPM
0.26 ± 0.25	0.46 ± 0.06	0.77 ± 0.04	$\mathbf{0.87 \pm 0.09}$

Table 2: Accuracy (higher better) on test MNIST data of recognition networks trained by peer supervision.

5.1 Peer Supervision

In the first experiment, the observations $\mathbf{x}_j$ are groups of MNIST (Deng, 2012) images representing J (here 2) different renderings of the same digit. The data set is structured in this way so that the J images are conditionally independent given the (unknown) digit identity. Thus, we expect the RPM to extract identity without explicit label information – a setting we term "peer supervision" (Fig. 1). The RPM is constructed with a single discrete-valued latent $\mathbf{z}$, and a recognition network (two convolutional layers, pooling, two linear layers and rectified linear activation function (ReLU) trained using Adam) with shared parameters θ for both factors. The learned recognition network achieved an average test set classification accuracy of 0.87 ± 0.09 over different random seeds, achieving an accuracy of 0.96 on 4 out of 10 runs. Failures occurred predominantly when multiple (usually two) digits were systematically mapped to the same latent—a phenomenon possible in the absence of explicit label supervision. This effect results in a correlation between the average posterior entropy and the classification accuracy (see Fig. A1). When comparing the recognition network accuracy on MNIST test set, RPM outperforms both Vector Quantised-VAE and VQ-VAE trained using the Gumbel Softmax categorical reparametrisation (GS-Soft VQ-VAE) (Sønderby et al., 2017; Van Den Oord et al., 2017; Maddison et al., 2016) See Table 2. Implementation details can be found in Appendix B.1.

5.2 RP-LDA

A second RPM instance builds on latent Dirichlet allocation (LDA) models and variational Bayes to identify pixel-level statistical regularities corresponding to textural prop-

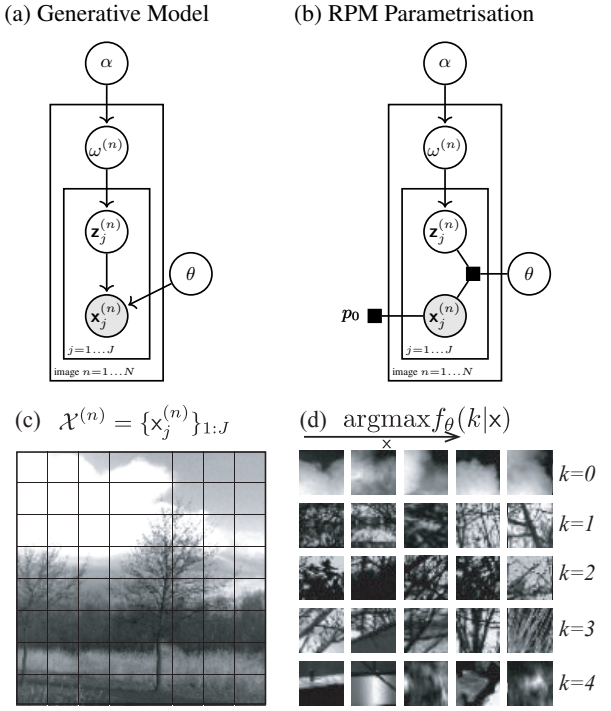


Figure 2: (a) Latent Dirichlet Allocation and (b) RP-LDA. (c) Splitting of an image in patches. (d) Five most representative patches for five texture categories sorted using the recognition network outputs.

erties of image subpatches (Fig. 2). Images, indexed by n, are decomposed into J smaller non-overlapping subpatches $\mathbf{x}_j^{(n)}$ (Fig. 2c) which are assumed to be conditionally independent given a discrete latent texture identity $\mathbf{z}_j^{(n)}$. The $\mathbf{z}_j^{(n)}$ are drawn from random categorical distributions $\omega^{(n)}$, which each gives the distribution of textures in the corresponding image, and is in turn drawn from a Dirichlet prior (with uniform parameter α). A recognition network (with the same structure as in Section 5.1) is shared across patches, and outputs a categorical distribution over texture identities given the patch pixel values. Writing $\mathcal{Z} = \{\mathbf{z}_j : j = 1 \dots J\} \cup \{\omega\}$, RP-LDA takes the form

$$\mathsf{P}(\mathcal{X}, \mathcal{Z}) = \prod_{j=1}^{J} p_{0j}(\mathbf{x}_j) \frac{f_\theta(\mathbf{z}_j|\mathbf{x}_j)}{F_\theta(\mathbf{z}_j)} p\left(\mathbf{z}_j|\omega\right) p\left(\omega|\alpha\right). \quad (10)$$

Applied to images from the van Hateren database (Van Hateren and van der Schaaf, 1998), RP-LDA recovers textural components (clouds, branches, pavements, etc.). Fig. 2d shows representative patches $\mathbf{x}_j$ that are most robustly assigned to a single textural category. Further details and derivations are given in Appendix B.2.

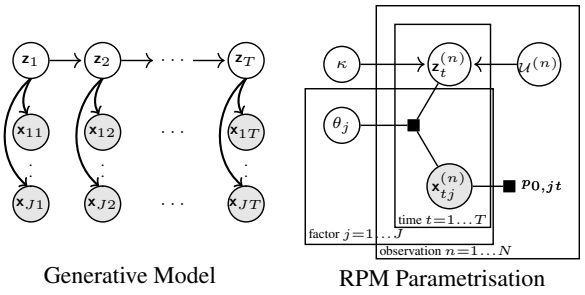


Figure 3: RP-GPFA

5.3 RP-GPFA

Finally, we model continuous multi-factorial temporal dependencies by introducing recognition-parametrised Gaussian process factor analysis (RP-GPFA; Fig. 3). We consider J observed time-series measured over T timesteps: $\mathcal{X} = \{\mathbf{x}_{jt} : j = 1 \dots J,\ t = 1 \dots T\}$. We seek to capture both spatial and temporal structure in a set of K-dimensional underlying latent time-series $\mathcal{Z} = \{\mathbf{z}_t : t = 1 \dots T\}$, such that the observations are conditionally independent across series and across time. The RPM thus takes the form

$$\mathsf{P}_{\theta,\mathbb{X}^{(N)}}(\mathcal{X}, \mathcal{Z}) = \prod_{j=1}^{J} \prod_{t=1}^{T} \left(p_{0,jt}(\mathbf{x}_{jt}) \frac{f_{\theta j}(\mathbf{z}_t|\mathbf{x}_{jt})}{F_{\theta j}(\mathbf{z}_t)} \right) p_{\theta z}(\mathcal{Z}) . \quad (11)$$

The prior on $\mathcal{Z}$ comprises independent Gaussian Process priors over each latent dimension $\mathcal{Z}_k = \{z_{kt} : t = 1 \dots T\}$

$$p_{\theta z}(\mathcal{Z}) = \prod_{k=1}^{K} p_k(\mathcal{Z}_k) ; \quad p_k(\cdot) = \mathcal{GP}(0, \kappa_k(\cdot,\cdot)) . \quad (12)$$

The recognition factors are parametrised by a neural network with weight θ_j that outputs the parameters of a multivariate normal distribution, i.e. $f_{\theta j}(\mathbf{z}_t|\mathbf{x}_{jt}) = \mathcal{N}(\mathbf{z}_t; \mu_\theta[\mathbf{x}_{jt}], \Sigma_\theta[\mathbf{x}_{jt}])$. We use the sparse variational GP approximation (Titsias, 2009) to improve scalability. The model is augmented with M inducing points (IP) for each latent dimension ($k = 1 \dots K$) and each observation ($n = 1 \dots N$). This smaller set ($M < T$) of fictitious measurements is optimised to efficiently represent function evaluations. For simplicity, IP are defined at fixed and shared locations. We restrict our experiments to using the radial basis function kernel, but the method can accommodate any GP prior.

Performance range

Table 3 gives the median and inter-quartile range for each of the RP-GPFA experiments described in the main text.

		sGP-VAE		RP-GPFA		
		1D	2D	Monte-Carlo	2nd Order	Variational
Textured Bouncing Ball	$-\mathcal{F}$ ($\times 10^4$)	$3.1\,[3.0-3.2]$	$2.3\,[2.2-2.4]$	$0.90\,[0.76-0.97]$	$0.91\,[0.77-0.98]$	$\mathbf{0.74\,[0.59-1.2}$]
	nMSE	≥ 0.99	≥ 0.99	$0.85\,[0.06-1.00]$	$0.14\,[0.04-0.94]$	$\mathbf{0.05\,[0.03-0.26]}$
Structured Background	$-\mathcal{F}$ ($\times 10^3$)	$40\,[28-46]$	$26\,[21-30]$	1.1 [1.1 -1.1]	$0.96\,[0.93-1.0$]	$\mathbf{0.77\,[0.75-0.80]}$
	nMSE	≥ 0.99	≥ 0.9	$0.11\,[0.10-0.14]$	$0.11\,[0.10-0.12]$	$\mathbf{0.09[0.09-0.10]}$

Table 3: Performance on the Textured and Structured Background Bouncing Ball Experiments using negative free energy ($-\mathcal{F}$; lower better) and normalised mean squared regression error to the true latent (nMSE; lower better). We compare different fitting procedures for RP-GPFA with a single latent dimension. sGP-VAE is fitted with both one or two latent dimensions. Values indicate median and inter-quartile range over 20 random seeds.

5.3.1 Textured Bouncing Ball

We illustrate RP-GPFA on a modified version of the bouncing ball experiment (Johnson et al., 2016), in which a one-dimensional latent modulates the intensity of observed pixels across time (Fig. 4). The stochastic mapping from latent to observation is defined such that the mean and variance of pixel intensity is independent of the latent position. We compare our approach to sparse Gaussian process VAE (sGP-VAE) (Ashman et al., 2020) and report the negative free energy ($-\mathcal{F}$) and the normalised mean squared error (nMSE) obtained by linear regression from inferred to true latent (Table 3). RP-GPFA is fit using a one dimensional latent space with each of the E-step methods described in Section 3.2. Reparametrisation employed only 20 samples to maintain computational comparability. All methods shared the same recognition network structure (two fully connected layers of size 50, ReLU activation function) and were trained using Adam. Kingma and Ba (2014).

The latent variable influences the higher order statistics of the image (i.e., the texture) but the standard sGP-VAE generative model maps latent to observations through multivariate Gaussian distributions. As a consequence, the one-dimensional version of this model is predictably blind to the latent oscillations. Interestingly, this was still the case when using a two-dimensional latent space. In contrast, the implicit generative process of RP-GPFA is not subject to model mismatch and recovers the latent dynamics accurately. The best performance was reached using the interior variational bound, albeit with high variability. The second order approximation yielded competitive and more reliable results across the 20 random seeds.

5.3.2 Structured-Background Bouncing Ball

In a second variant of the bouncing ball experiment, the ball appeared as a local Gaussian blur imposed over a structured, striped, moving background (Fig. 4). This example helps to illustrate another shortcoming of explicit generation, beyond the risk of model mismatch illustrated above. The generative likelihood depends on the capacity to reconstruct the entire observation, including any structured but independent features that cannot be ascribed

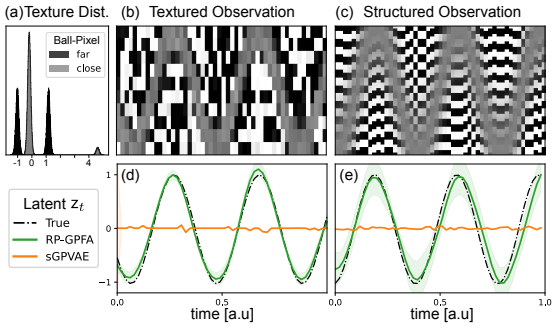


Figure 4: Bouncing ball experiments: a latent variable $\mathbf{z}_t$ modulates pixel intensity. (a,b) $\mathbf{z}_t$ influences the higher order statistics of the image (i.e., the texture). (c) Intensity modulation is imposed over structured background. (d,e) Latent recovery using RP-GPFA (variational bound method) or sGP-VAE. Shades indicates 2 standard deviations.

to noise. In this example, the sGP-VAE must work to model both structured background and ball-related features, which proves impossible with one or two latent processes. By contrast, the RPM likelihood focuses on latent structure which renders observations conditionally independent in time (Fig. 4). The difference is again reflected quantitatively in the free energies achieved and match between the recovered latent and ball position (Table 3).

5.3.3 Multi-factorial Integration across Time

Last, we considered conditional independence structured across time and observed signal. Three independent agent navigate in a bounded environment, moving inanimate blocks to a designated target. Observations of the agents' locations are collected in the form of 3D-rendered image frames of the entire environment $\mathbf{x}_1 = \{\mathbf{x}_{1t} : t = 1 \dots T\}$, as well as noisy range-finding sensor data giving the distances of one of the agents from the four corners of the room $\mathbf{x}_2 = \{\mathbf{x}_{2t} : t = 1 \dots T\}$ (Fig. 5). Renderings and trajectories were generated using Unity Machine Learning Agents Toolkit (Juliani et al., 2018). In this setting, a (2D) latent inducing conditional independence should recover

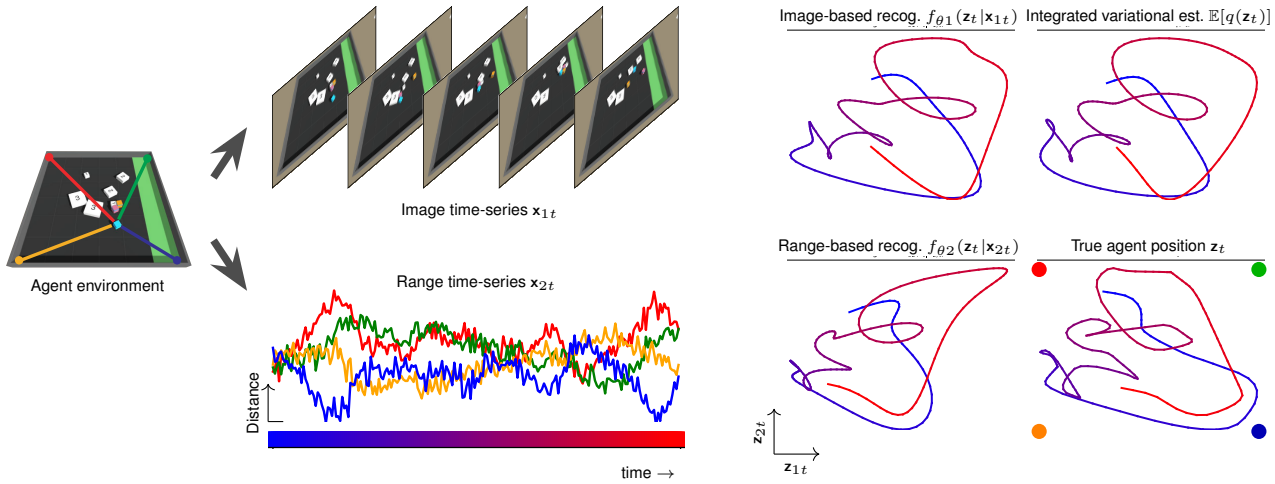


Figure 5: Multi-factorial integration across time. RP-GPFA combines image frames $\mathbf{x}_{1t}$ and noisy range-finding data $\mathbf{x}_{2t}$ tracking an agent moving amongst distractors. No structural priors on the source of the data streams are included. By identifying a 2D signal that renders the complex data streams conditionally independent, it recovers the position $\mathbf{z}_t$ of the range-finder-equipped agent. In particular the noisy range-finder data is related to only one of the many agents in the environment, which the RPM-reconstructed $\mathbf{z}_t$ learns to select.

the location of the range-finder-equipped agent, ignoring the other agents and the rest of the image data.

We trained RP-GPFA using the 2$^{\text{nd}}$-order approximation method, 40 inducing points, a convolutional network θ_1 acting on image data and a two-layer perceptron θ_2 on range data (resp. similar to Section 5.1 and Section 5.3.1).

Fig. 5 shows one full trajectory $\mathcal{Z} = \{\mathbf{z}_t : t = 1 \dots T\}$, the recovered mean of the variational distribution q, and the individual video and range recording factors $f_{\theta j}(\cdot|\mathbf{x}_{jt})$ combined with $p_{\theta z}$. These results illustrate how the pursuit of conditional independence underlying RP-GPFA makes it possible to (i) learn the nonlinear mapping from distance to position signal and (ii) learn to track a moving agent from video recordings. Perhaps more importantly, it is the conditional independence structure across distance sensors and video that provides the signal guiding the video network to (iii) learn which agent to track amongst the distractors.

6 CONCLUSION

We have introduced the *recognition-parametrised model*, a normalised semi-parametric family in which the latent variables model the joint dependence of observations but not their individual marginals. As the parametric part of the likelihood is defined in terms of the recognition parameters alone, the RPM avoids issues of mismatch between generative and recognition models, and enables rapid computations of latent posterior distributions from observed data. Furthermore, by incorporating the empirical marginal distribution of individual latents, the RPM is able to capture joint structure, regardless of details of the noise distribution, or of unrelated distractors.

The RPM may be defined using simple exponential family forms on the latents, allowing access to the wide range of probabilistic tools. The capacity for structured probabilistic inference was exploited in the experiments here, with RP-LDA exemplifying the use of hierarchical models and variational Bayes, while RP-GPFA combined RPM inference with the sparse variational GP approximation. The model can be learned through maximum-likelihood exactly in the case of discrete latents and we present several approximations to a variational bound for the continuous case.

Animals and artificial agents acting in the world need to learn structure in sensory input to build representations of their environments and infer state, but they rarely need to generate synthetic observations. The assumptions of the RPM: that recognition is probabilistic, detailed simulation is avoided, and learning is unsupervised are likely to be those that shape natural intelligence. Behavioural studies reveal Bayesian perception and decision making under noise, uncertainty and risk; dense cortico-fugal connections do not extend to the sensory periphery; and natural human "supervision" in fact corresponds to the RPM principle: object category is the thing that makes the utterance of a caregiver conditionally independent of the picture or object to which they are pointing. We are unaware of other learning frameworks that are fully probabilistic, unsupervised, tractable, and avoid explicit instantiation of a generative model. Thus models of the RPM type may be central to replicating general animal-like intelligence. and so we believe it holds promise both as a model of biological learning and as a basis for efficient state discovery and action learning in artificial settings.

Acknowledgements

This work was funded by the Gatsby Charitable Foundation and Simons Foundation (SCGB 543039). We thank Ted Moskovitz, Marcel Nonnenmacher and Peter Orbanz for helpful discussions.

References

D. H. Ackley, G. E. Hinton, and T. J. Sejnowski. A learning algorithm for Boltzmann machines. *Cognitive Science*, 9(1):147–169, 1985.

L. Aitchison and S. Ganev. InfoNCE is a variational autoencoder. *arXiv*, arxiv:2107.02495v2, 2023.

M. Ashman, J. So, W. Tebbutt, V. Fortuin, M. Pearce, and R. E. Turner. Sparse Gaussian process variational autoencoders. *arXiv preprint arXiv:2010.10177*, 2020.

H. Attias. Inferring parameters and structure of graphical models by variational Bayes. In S. A. Solla, T. K. Leen, and K.-R. Müller, editors, *Adv Neural Info Processing Sys*, volume 12, Cambridge, MA, 2000. MIT Press.

D. Barber. *Bayesian Reasoning and Machine Learning*. Cambridge University Press, 2011.

C. M. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006.

D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent Dirichlet allocation. *J Mach Learn Res*, 3(Jan):993–1022, 2003.

M. Braun and J. McAuliffe. Variational inference for large-scale models of discrete choice. *Journal of the American Statistical Association*, 105(489):324–335, 2010.

Y. Burda, R. Grosse, and R. Salakhutdinov. Importance weighted autoencoders. *ICLR*, 2016.

P. Dayan, G. E. Hinton, R. M. Neal, and R. S. Zemel. The Helmholtz machine. *Neural Comput*, 7(5):889–904, 1995.

L. Deng. The MNIST database of handwritten digit images for machine learning research. *IEEE Sig Proc Mag*, 29 (6):141–142, 2012.

L. Duncker and M. Sahani. Temporal alignment and latent Gaussian process factor inference in population spike trains. In *Adv Neural Info Processing Sys*, volume 31, 2018.

I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In *Adv Neural Info Processing Sys*, volume 27, 2014.

M. Gutmann and A. Hyvärinen. Noise-contrastive estimation: A new estimation principle for unnormalized statistical models. In *AISTATS*, pages 297–304. JMLR Workshop and Conference Proceedings, 2010.

H. L. F. Helmholtz. *Handbuch der physiologischen Optik*. Voss, 1867. Republished: Thoemmes Continuum.

G. E. Hinton. Training products of experts by minimizing contrastive divergence. *Neural Comput*, 14(8):1771–1800, 2002.

M. J. Johnson, D. K. Duvenaud, A. Wiltschko, R. P. Adams, and S. R. Datta. Composing graphical models with neural networks for structured representations and fast inference. In *Adv Neural Info Processing Sys*, volume 29, 2016.

M. I. Jordan, Z. Ghahramani, T. Jaakkola, and L. K. Saul. An introduction to variational methods for graphical models. *Mach Learn*, 37(2):183–233, 1999.

A. Juliani, V.-P. Berges, E. Teng, A. Cohen, J. Harper, C. Elion, C. Goy, Y. Gao, H. Henry, M. Mattar, et al. Unity: A general platform for intelligent agents. *arXiv preprint arXiv:1809.02627*, 2018.

D. Kingma, T. Salimans, B. Poole, and J. Ho. Variational diffusion models. In *Adv Neural Info Processing Sys*, volume 34, pages 21696–21707, 2021.

D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

D. P. Kingma and M. Welling. Auto-encoding variational Bayes. *ICLR*, 2014.

C. J. Maddison, A. Mnih, and Y. W. Teh. The concrete distribution: A continuous relaxation of discrete random variables. *arXiv preprint arXiv:1611.00712*, 2016.

C. J. Maddison, J. Lawson, G. Tucker, N. Heess, M. Norouzi, A. Mnih, A. Doucet, and Y. Teh. Filtering variational objectives. In *Adv Neural Info Processing Sys*, volume 30, 2017.

V. Masrani, T. A. Le, and F. Wood. The thermodynamic variational objective. In *Adv Neural Info Processing Sys*, volume 32, 2019.

K. P. Murphy. *Probabilistic Machine learning: An Introduction*. MIT Press, Cambridge, Mass., 2022.

R. M. Neal and G. E. Hinton. A view of the EM algorithm that justifies incremental, sparse, and other variants. In M. I. Jordan, editor, *Learning in Graphical Models*, pages 355–370. Kluwer Academic Press, 1998.

A. v. d. Oord, Y. Li, and O. Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

D. Rezende and S. Mohamed. Variational inference with normalizing flows. In *International conference on machine learning*, pages 1530–1538. PMLR, 2015.

D. J. Rezende, S. Mohamed, and D. Wierstra. Stochastic backpropagation and approximate inference in deep generative models. In *PMLR*, pages 1278–1286, 2014.

P. Smolensky. Information processing in dynamical systems: Foundations of harmony theory. In D. E. Rumelhart and J. L. McClelland, editors, *Parallel Distributed*

Processing: Volume 1: Foundations, pages 194–281. MIT Press, Cambridge, MA., 1986.

C. K. Sønderby, B. Poole, and A. Mnih. Continuous relaxation training of discrete latent variable image models. *Baysian DeepLearning workshop, NIPS 2017*, 2017.

M. Titsias. Variational learning of inducing variables in sparse Gaussian processes. In *AISTATS*, pages 567–574. PMLR, 2009.

R. E. Turner and M. Sahani. Two problems with variational expectation maximisation for time-series models. In D. Barber, A. T. Cemgil, and S. Chiappa, editors, *Bayesian Time Series Models*. Cambridge University Press, 2011.

A. Van Den Oord, O. Vinyals, and K. Kavukcuoglu. Neural discrete representation learning. In *Adv Neural Info Processing Sys*, volume 30, pages 6309–6318, 2017.

J. H. Van Hateren and A. van der Schaaf. Independent component filters of natural images compared with simple cells in primary visual cortex. *P Roy Soc B: Biol Sci*, 265(1394):359–366, 1998.

E. Vértes and M. Sahani. Flexible and accurate inference and learning for deep generative models. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Adv Neural Info Processing Sys*, volume 31, pages 4169–4178. Curran Associates, Inc., 2018.

E. Vértes and M. Sahani. A neurally plausible model learns successor representations in partially observable environments. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché Buc, E. Fox, and R. Garnett, editors, *Adv Neural Info Processing Sys*, volume 32, pages 13692–13702. Curran Associates, Inc., 2019.

M. Welling, M. Rosen-Zvi, and G. E. Hinton. Exponential family harmoniums with an application to information retrieval. In *Adv Neural Info Processing Sys*, volume 17, 2004.

L. K. Wenliang and M. Sahani. A neurally plausible model for online recognition and postdiction. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché Buc, E. Fox, and R. Garnett, editors, *Adv Neural Info Processing Sys*, volume 32, pages 9641–9652. Curran Associates, Inc., 2019.

L. K. Wenliang, T. Moskovitz, H. Kanagawa, and M. Sahani. Amortised learning by wake-sleep. In *Proceedings of the 37th International Conference on Machine Learning*, volume 98 of *Proceedings of Machine Learning Research*. PMLR, 2020.

J. Winn, C. M. Bishop, and T. Jaakkola. Variational message passing. *J Mach Learn Res*, 6(4), 2005.

B. M. Yu, J. P. Cunningham, G. Santhanam, S. Ryu, K. V. Shenoy, and M. Sahani. Gaussian-process factor analysis for low-dimensional single-trial analysis of neural population activity. In *Adv Neural Info Processing Sys*, volume 21, 2008.

A MAXIMUM LIKELIHOOD LEARNING

We provide further details and complete derivations of key results. Equation numbers without an 'A' prefix correspond to those in the main text.

Recall that the full joint distribution associated with a Recognition-Parametrised Model (RPM) takes the form

$$labeleq : joint\mathsf{P}_{\theta,\mathbb{X}^{(N)}}(\mathcal{X},\mathcal{Z}) = p_{\theta z}(\mathcal{Z})\prod_j \left(p_{0j}(\mathbf{x}_j)\frac{f_{\theta j}(\mathcal{Z}|\mathbf{x}_j)}{F_{\theta j}(\mathcal{Z})}\right), \tag{2}$$

where $\mathcal{X} = \{\mathbf{x}_j : j = 1 \dots J\}$ is a set random variables and $\mathcal{Z} = \{\mathbf{z}_l : l = 1 \dots L\}$ is a set of underlying latent variables given which $\mathbf{x}_j$ are conditionally independent. The factors are defined as

$p_{\theta z}(\mathcal{Z})$: a normalised distribution on the latent variables whose factorisation depends on a latent graphical model.

$p_{0j}(\mathbf{x}_j) = \frac{1}{N}\sum_n \delta(\mathbf{x}_j - \mathbf{x}_j^{(n)})$: the empirical measures with atoms at the N data points $\mathbf{x}_j^{(n)}$.

$f_{\theta j}(\mathcal{Z}|\mathbf{x}_j)$: parameterised distributions that we call "recognition factors".

$F_{\theta j}(\mathcal{Z}) = \int d\mathbf{x}_j p_{0j}(\mathbf{x}_j) f_{\theta j}(\mathcal{Z}|\mathbf{x}_j)$: mixture of recognition factors with respect to the empirical measures.

As we take $p_{0j}(\mathbf{x}_j)$ to be the empirical measures throughout, the mixtures have the forms $F_{\theta j}(\mathcal{Z}) = \frac{1}{N}\sum_n f_{\theta j}(\mathcal{Z}|\mathbf{x}_j^{(n)})$.

A.1 Variational Free Energy

We use Expectation-Maximisation coordinate ascent of the variational free energy (sometimes referred to as Evidence Lower Bound or ELBO) derived by applying Jensen's inequality to the log likelihood (Neal and Hinton, 1998):

$$\begin{aligned}
&\sum_n \log \mathsf{P}_{\theta,\mathbb{X}^{(N)}}\left(\mathcal{X}^{(n)}\right) \geq \mathcal{F}(\theta, q(\{\mathcal{Z}^{(n)}\})) \\
&\quad = \left\langle \sum_n \log \mathsf{P}_{\theta,\mathbb{X}^{(N)}}\left(\mathcal{X}^{(n)},\mathcal{Z}^{(n)}\right)\right\rangle + \mathbf{H}[q] \\
&\quad = \sum_n \left(\left\langle \log p_{\theta z}(\mathcal{Z}^{(n)})\right\rangle + \sum_j \left(\left\langle \log f_{\theta j}(\mathcal{Z}^{(n)}|\mathbf{x}_j^{(n)})\right\rangle - \left\langle \log F_{\theta j}(\mathcal{Z}^{(n)})\right\rangle\right)\right) \\
&\qquad + \sum_{jn} \log p_{0j}(\mathbf{x}_j^{(n)}) + \sum_n \mathbf{H}\big[q^{(n)}\big]
\end{aligned} \tag{A1}$$

where, as in the main text, angle brackets indicate expectations with respect to the variational distribution q, $\mathbf{H}[\cdot]$ is the entropy, and we have used the fact that the optimal q has the form $\prod q^{(n)}(\mathcal{Z}^{(n)})$. When the latent variables that appear in $F_{\theta j}(\mathcal{Z})$ are discrete, and the graphical structure of $p_{\theta z}$ admits exact belief propagation, the expressions in Eq. (A1) can be evaluated in closed form and optimisation is straightforward. For the more challenging case of continuous-valued latent variables, we introduced three approximation approaches in the main text. These are reviewed and developed further below.

A.2 E-step for Continuous-Valued Latent Variables

We consider the case in which $p_{\theta z}$, $f_{\theta j}(.|\mathbf{x}_j)$ and $q^{(n)}$ are all members of the same exponential family with natural parameters η_0, $\eta_j(\mathbf{x}_j^{(n)})$ and $\eta_q^{(n)}$ respectively, corresponding to minimal sufficient statistic $t(\mathcal{Z})$ and log-normaliser Φ. The expectations of $t(\mathcal{Z})$ under the corresponding distribution are written μ_0, $\mu_j(\mathbf{x}_j^{(n)})$ and $\mu_q^{(n)}$ respectively (notation is summarised in table A1). With this set of assumptions, the only term from Eq. (A1) that cannot be expressed analytically is $\left\langle \log F_{\theta j}(\mathcal{Z}^{(n)})\right\rangle$. As discussed in the main paper, if a sample from q can be expressed as a parametrised function of a sample from a fixed distribution (as is the case with multivariate normal distributions) it can be evaluated using Monte-Carlo estimates. Nevertheless, this approach may be computationally expensive for high dimensional problems. We therefore propose two additional approaches to handle intractable terms.

Name	Distribution	Natural Parameter	$\mathbb{E}(t(\mathcal{Z}))$	$\mathbb{V}(t(\mathcal{Z}))$	Normalised ?
Prior	$p_{\theta z}$	η_0	μ_0	N/A	Yes
Recognition Factors	$f_{\theta j}(.\|\mathbf{x}_j)$	$\eta_j(\mathbf{x}_j^{(n)})$	$\mu_j(\mathbf{x}_j^{(n)})$	N/A	Yes
Variational	$q^{(n)}$	$\eta_q^{(n)}$	$\mu_q^{(n)}$	$V_q^{(n)}$	Yes
Auxiliary Factors	$\tilde{f}_j^{(n)}$	$\tilde{\eta}_j^{(n)}$	N/A	N/A	No
Normalised Auxiliary	$\hat{f}_j^{(n)}$	$\eta_j(\mathbf{x}_j^{(n)}) - \tilde{\eta}_j^{(n)}$	N/A	N/A	Yes
Mixture	$F_{\theta j}$	N/A	N/A	N/A	Yes

Table A1: Notation Glossary for Continuous Exponential Family Case

A.2.1 Second-order Approximation

First, we generalise the approach introduced by (Braun and McAuliffe, 2010) and expand $g(t(\mathcal{Z})) = \log F_{\theta j}(\mathcal{Z})$ to second order in $t(\mathcal{Z})$ around its expectation $\mu_q^{(n)}$. This gives

$$g(t(\mathcal{Z})) \approx g(\mu_q^{(n)}) + \partial g^\top \left(t(\mathcal{Z}) - \mu_q^{(n)}\right) + \frac{1}{2}\left(t(\mathcal{Z}) - \mu_q^{(n)}\right)^\top \partial^2 g \left(t(\mathcal{Z}) - \mu_q^{(n)}\right), \tag{A2}$$

where

$$\partial g = \sum_m \eta(\mathbf{x}_j^{(m)})\pi_{jm}^{(n)}, \tag{A3}$$

$$\partial^2 g = \sum_m \eta(\mathbf{x}_j^{(m)})\eta(\mathbf{x}_j^{(m)})^\top \pi_{jm}^{(n)} - \sum_{m,m'} \eta(\mathbf{x}_j^{(m)})\eta(\mathbf{x}_j^{(m')})^\top \pi_{jm}^{(n)}\pi_{jm'}^{(n)}, \tag{A4}$$

with

$$\pi_{jm}^{(n)} = \frac{e^{\eta(\mathbf{x}_j^{(m)})^\top \mu_q^{(n)} - \Phi(\eta(\mathbf{x}_j^{(m)}))}}{\sum_p e^{\eta(\mathbf{x}_j^{(p)})^\top \mu_q^{(n)} - \Phi(\eta(\mathbf{x}_j^{(p)}))}}. \tag{7}$$

The first order term vanishes when taking the expectation over $q^{(n)}$ so that

$$\langle g(t(\mathcal{Z}))\rangle \approx g(\mu_q^{(n)}) + \frac{1}{2}\mathrm{tr}\left(V_q^{(n)}\partial^2 g\right). \tag{A5}$$

Finally, we gather the recognition factor natural parameters in

$$\boldsymbol{\eta}_j = \left[\eta(\mathbf{x}_j^{(1)}), \ldots, \eta(\mathbf{x}_j^{(N)})\right]$$

and the weights in

$$\boldsymbol{\pi}_j^{(n)} = \left[\pi_{1j}^{(n)}, \ldots, \pi_{Nj}^{(n)}\right]^\top,$$

yielding

$$\left\langle \log F_{\theta j}\right\rangle \approx \log \frac{1}{N}\sum_{m=1}^{N} e^{\eta(\mathbf{x}_j^{(m)})^\top \mu_q^{(n)} - \Phi(\eta(\mathbf{x}_j^{(m)}))} + \frac{1}{2}\mathrm{tr}\left(\boldsymbol{\eta}_j^\top V_q^{(n)} \boldsymbol{\eta}_j \left[\mathrm{diag}(\boldsymbol{\pi}_j^{(n)}) - \boldsymbol{\pi}_j^{(n)}\boldsymbol{\pi}_j^{(n)\top}\right]\right). \tag{6}$$

This form can be inserted into Eq. (A1) to yield a tractable, approximate free energy.

In the case where $q^{(n)}$ is a multivariate distribution with mean $\mathbf{m}^{(n)}$ and variance $S^{(n)}$, we recall

$$\mu_q^{(n)} = \begin{bmatrix} \mathbf{m}^{(n)} \\ \mathrm{Vec}\left(S^{(n)} + \mathbf{m}^{(n)}\mathbf{m}^{(n)\top}\right)\end{bmatrix} \tag{A6}$$

and

$$V_q^{(n)} = \begin{bmatrix} S^{(n)} & \mathbf{m}^{(n)\top} \otimes S^{(n)} + S^{(n)} \otimes \mathbf{m}^{(n)\top} \\ \mathbf{m}^{(n)} \otimes S^{(n)} + S^{(n)} \otimes \mathbf{m}^{(n)} & \mathbb{S}^{(n)}\end{bmatrix} \tag{A7}$$

where

$$\mathbb{S}^{(n)} = h(S^{(n)}, S^{(n)}) + h(S^{(n)}, \mathbf{m}^{(n)}\mathbf{m}^{(n)\top}) + h(\mathbf{m}^{(n)}\mathbf{m}^{(n)\top}, S^{(n)})\,.$$

and

$$h(A, B) = A \otimes B + \left(\Gamma^\top \otimes A \otimes \Gamma\right) \odot \left(\Gamma \otimes B \otimes \Gamma^\top\right) \text{ with } \Gamma = 1_{K\times 1} \tag{A8}$$

$\otimes$ and $\odot$ are the Kronecker and Hadamard products.

A.2.2 Interior Variational Bound

The previous approach gives a compact approximation of the free energy but it is not guaranteed to lower bound the log-likelihood. Thus, we considered a second strategy in which we introduced a further relaxation of the free-energy bound, by introducing auxiliary functions $\tilde{f}_j^{(n)}(\mathcal{Z})$. Focusing on the $F_{\theta j}$-dependent terms as above, we have

$$\begin{aligned}
\left\langle \log \frac{f_{\theta j}(\cdot|\mathbf{x}_j^{(n)})}{F_{\theta j}} \right\rangle &= \left\langle \log \frac{f_{\theta j}(\cdot|\mathbf{x}_j^{(n)})}{\tilde{f}_j^{(n)} q^{(n)}} \right\rangle - \left\langle \log \frac{F_{\theta j}}{\tilde{f}_j^{(n)} q^{(n)}} \right\rangle \\
&\geq \left\langle \log \frac{f_{\theta j}(\cdot|\mathbf{x}_j^{(n)})}{\tilde{f}_j^{(n)} q^{(n)}} \right\rangle - \log \left\langle \frac{F_{\theta j}}{\tilde{f}_j^{(n)} q^{(n)}} \right\rangle \quad \text{(by Jensen)} \\
&= \left\langle \log \frac{f_{\theta j}(\cdot|\mathbf{x}_j^{(n)})}{\tilde{f}_j^{(n)} q^{(n)}} \right\rangle - \log \int d\mathcal{Z}\, \frac{F_{\theta j}(\mathcal{Z})}{\tilde{f}_j^{(n)}(\mathcal{Z})}\,.
\end{aligned} \tag{A9}$$

If we now choose $\tilde{f}_j^{(n)}(\mathcal{Z}) = \exp\left(t(\mathcal{Z})^\mathsf{T} \tilde{\eta}_j^{(n)}\right)$ with the constraint that $\eta_j(\mathbf{x}_j^{(m)}) - \tilde{\eta}_j^{(n)}$ is a valid natural parameter for all (m, n), then the right hand-side of Eq. (A9) is closed-form

$$\tilde{\Gamma}_j^{(n)} = \int d\mathcal{Z} \frac{F_{\theta j}(\mathcal{Z})}{\tilde{f}_j^{(n)}(\mathcal{Z})} = \frac{1}{N} \sum_m e^{\Phi(\eta_j(\mathbf{x}_j^{(m)}) - \tilde{\eta}_j^{(n)}) - \Phi_j(\eta_j(\mathbf{x}_j^{(m)}))}\,. \tag{A10}$$

By rearranging terms, we obtain (c.f. main text eq. 8)

$$\left\langle \log \frac{f_{\theta j}(\mathbf{z}_j|\mathbf{x}_j^{(n)})}{F_{\theta j}(\mathbf{z}_j)} \right\rangle \geq -\,\mathbf{KL}\left[q^{(n)} \middle\| \hat{f}_j^{(n)}\right] + \log \Gamma_j^{(n)}\,, \tag{A11}$$

where $\hat{f}_j^{(n)}$ is a properly normalised exponential family distribution with natural parameter $\eta_j(\mathbf{x}_j^{(n)}) - \tilde{\eta}_j^{(n)}$ and

$$\Gamma_j^{(n)} = \frac{e^{\Phi(\eta_j(\mathbf{x}_j^{(n)}) - \tilde{\eta}_j^{(n)}) - \Phi(\eta_j(\mathbf{x}_j^{(n)}))}}{\frac{1}{N}\sum_m e^{\Phi(\eta_j(\mathbf{x}_j^{(m)}) - \tilde{\eta}_j^{(n)}) - \Phi_j(\eta_t(\mathbf{x}_j^{(m)}))}} = \frac{e^{\Phi(\eta_j(\mathbf{x}_j^{(n)}) - \tilde{\eta}_j^{(n)}) - \Phi(\eta_j(\mathbf{x}_j^{(n)}))}}{\tilde{\Gamma}_j^{(n)}}\,. \tag{A12}$$

This fully tractable expression can then be inserted in Eq. (A1). Furthermore, the terms of the resulting expression can be rearranged to make explicit the bound to the conventional free energy and the log-likelihood

$$\sum_n \log \mathsf{P}_{\theta, \mathbb{X}^{(N)}}\left(\mathcal{X}^{(n)}\right) \geq \mathcal{F}(\theta, q) \geq \widetilde{\mathcal{F}}\left(\theta, q, \{\tilde{f}_j^{(n)}\}\right), \tag{A13}$$

where

$$\widetilde{\mathcal{F}}\left(\theta, q, \{\tilde{f}_j^{(n)}\}\right) = \sum_n \log \mathsf{P}_{\theta, \mathbb{X}^{(N)}}\left(\mathcal{X}^{(n)}\right) - \sum_n \mathbf{KL}\left[q^{(n)} \middle\| \mathsf{P}_{\theta, \mathbb{X}^{(N)}}\left(\cdot|\mathcal{X}^{(n)}\right)\right] - \sum_{nj} \mathbf{KL}\left[q^{(n)} \middle\| \frac{1}{\tilde{\Gamma}_j^{(n)}} \frac{F_{\theta j}}{\tilde{f}_j^{(n)}}\right]. \tag{9}$$

Thus, as might be expected, the second variational relaxation introduces a further KL-divergence penalty, beyond the term in $\mathbf{KL}[q(\mathcal{Z})\|p(\mathcal{Z}|\mathcal{X})]$ introduced by the standard variational approach.

B DISCRETE EXPERIMENTS

In all discrete experiments, the recognition network θ was shared across factors and comprised 2 convolutional layers with max pooling and one fully connected layer of 50 units followed by a ReLU (Rectified Linear Unit) activation function.

B.1 Peer-Supervision

In this case, the RPM had a single categorical latent variable $\mathbf{z}$ (so $L = 1$) with uniform prior, and $J = 2$ observations $\mathbf{x}_j$ each corresponding to an MNIST image. The data set comprised random pairs of images of the same digit, with each digit appearing in only one pair. The factors $f_{\theta j}(\mathbf{z}|\mathbf{x}_j)$ were parametrised by a single convolutional neural network (i.e., the parameters θ_j were tied), which outputs categorical probabilities. Inference is thus conjugate. Assuming that $\mathbf{z}$ can take $K = 10$ values, the E-step has the closed form:

$$q^{(n)}(\mathbf{z} = k) \propto \prod_j \frac{f_\theta(k|\mathbf{x}_j^{(n)})}{\sum_m f_\theta(k|\mathbf{x}_j^{(m)})} \,. \tag{A14}$$

The RPM is compared to Gumbel Softmax Variational Autoencoder (GS-VAE) (Maddison et al., 2016), Vector Quantised Variational Autoencoder (VQ-VAE) (Van Den Oord et al., 2017)[1] and Gumbel-Softmax VQ-VAE (GS-VQVAE) (Sønderby et al., 2017)[2] (temperature of 0.5) . The Gumbel-Max reparametrisation trick allows the sampling of discrete random variables to be a sum of a deterministic function of the discrete probabilities and a fixed noise distribution, followed by an argmax operation. The Gumbel Softmax replaces the argmax with a softmax operation such that the gradients of the probabilities can be calculated. Thus this allows the VAE to learn using samples of the discrete latents in the loss. The VQ-VAE is a deterministic autoencoder whose encoder produces a continuous vector that then gets compared to a nearest neighbour embedding. The nearest neighbour is then used in the decoder for reconstructing data. Gradients are passed using the straight through estimator and the encoder, decoder, and nearest neighbour embedding is learned. The GS-VQVAE computes the variational posterior using the distances from encoder output to nearest neighbour embedding vectors as logits of a categorical distribution. Then learns to maximise the free energy using the Gumbel Softmax reparametrisation trick.

All VAE models shared the same generative neural network and all methods fundamentally shared the same recognition network. They differ in that the output dimension of RPM and GS-VAE was of dimension 10, while Vector Quantised Models recognition networks first output to an embedding space of dimension 64 before being mapped to one of 10 categories.

Each model was fit 10 times with different random initialisation. Once fit, the output of the recognition factor neural network was evaluated for classification accuracy on the MNIST test dataset on the basis of the best mapping from network output to digit identity (using Kuhn–Munkres algorithm).

Fig. A1 shows the accuracy achieved for each random seed as a function of the entropy of the average posterior. The RPM (alone) achieved performance of 96.5% for 3/10 random initialisations, but in other cases drew sharp classification boundaries that confused or divided single digit classes, as seems reasonable given the lack of label supervision. This effect can be seen in the confusion matrices shown in Fig. A1. None of the baseline models achieved better than about 80% accuracy, and all of them created more distributed errors, confusing examples of many digit types.

B.2 Latent Dirichlet Allocation (LDA)

The goal of the RPM-LDA is to infer the statistics of local image properties in natural images. We start by decomposing each image into sub-patches and denote:

[1] https://github.com/bshall/VectorQuantizedVAE

[2] https://github.com/YongfeiYan/Gumbel_Softmax_VAE

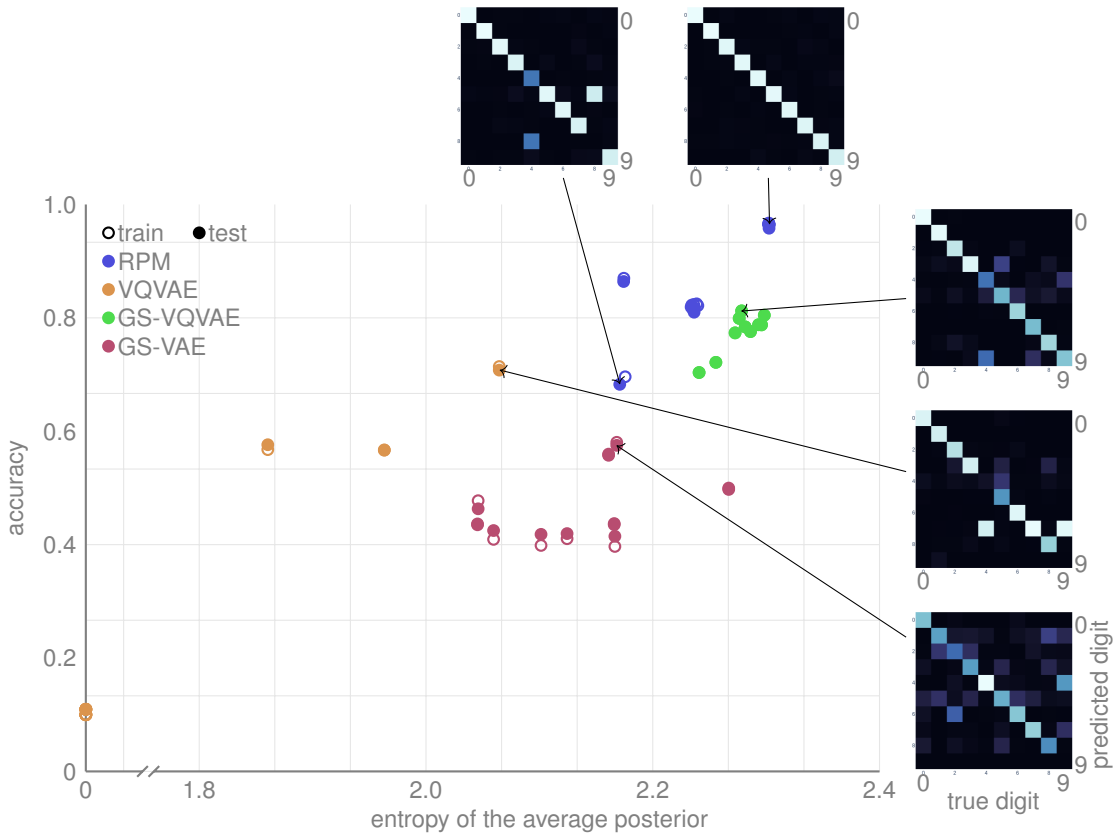
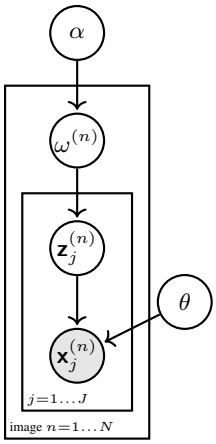
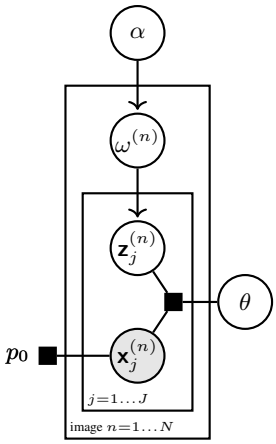


Figure A1: Accuracy and entropy of average posterior achieved by each of the four model types initialised with 10 different random seeds. Insets show confusion matrices (for the best digit assignment of latent values) for the least and most accurate RPM, and most accurate example of the baseline models.



LDA DAG



RPM parametrisation

- $\mathbf{x}_j^{(n)}$ the j-th patch of image n
- $\mathbf{z}_j^{(n)}$ the categorical identity of $\mathbf{x}_j^{(n)}$
- $\omega^{(n)}$ the distribution of categories for image n
- $p(\omega) = \text{Dirichlet}(\alpha, ..., \alpha)$ the prior over the category distribution
- θ the recognition network shared for all the patches that outputs the probabilities that a patch $\mathbf{x}_j^{(n)}$ belongs to each category.

The RPM has the form

$$\mathsf{P}_{\theta,\alpha,\mathbb{X}^{(N)}}(\mathcal{X},\mathcal{Z}) = \prod_{n=1}^{N}\prod_{j=1}^{J} p_0(\mathbf{x}_j^{(n)}) \frac{f_\theta(\mathbf{z}_j^{(n)}|\mathbf{x}_j^{(n)})}{\frac{1}{N}\sum_m f_\theta(\mathbf{z}_j^{(n)}|\mathbf{x}_j^{(m)})} p\big(\mathbf{z}_j^{(n)}|\omega^{(n)}\big) p\big(\omega^{(n)}|\alpha\big)\,, \tag{A15}$$

where $\mathcal{Z} = \big\{\{\mathbf{z}_j^{(n)}\}_j, \omega^{(n)}\big\}_n$.

We model the variational distribution as

$$q(\mathcal{Z}) = \prod_{n=1}^{N} q_\omega^{(n)}(\omega^{(n)}) \prod_{j=1}^{J} q_j^{(n)}(\mathbf{z}_j^{(n)}) \tag{A16}$$

where

$$q_\omega^{(n)} = \text{Dirichlet}(\alpha_1^{(n)}, ..., \alpha_K^{(n)}) \quad \text{and} \quad q_j^{(n)}(\mathbf{z}_j^{(n)} = k) = \gamma_{jk}^{(n)}\,. \tag{A17}$$

The E-Step is closed form and follows

$$\alpha_k^{(n)} = \alpha + \sum_{j=1}^{J} \gamma_{jk}^{(n)} \quad \text{and} \quad \gamma_{jk}^{(n)} \propto \exp\Big(\Psi(\alpha_k^{(n)}) + \log f_\theta(k|\mathbf{x}_j^{(n)}) - \log f_\theta(k)\Big) \tag{A18}$$

where Ψ is the digamma function.

During the M-Step, the recognition model is updated using Adam (Kingma and Ba, 2014) on the free energy. We applied RPM-LDA to 100 images from the van Hateren database and fixed $K = 10$. Given a texture k, its most representative patch is the one maximising the probability of being assigned to k: $f_\theta(k|\mathbf{x})$. We plot such patches Fig. A2-(a), and see that RPM-LDA learns meaningful textural information (clouds, branches, etc.). The statistics of each image can be described by $\bar{\omega}^{(n)} = \langle\omega^{(n)}\rangle_q$. We confirmed the inferred textural grouping by reporting examples of images with low entropy on A2-(b) and one where $\omega^{(n)}$ is multimodal A2-(c).

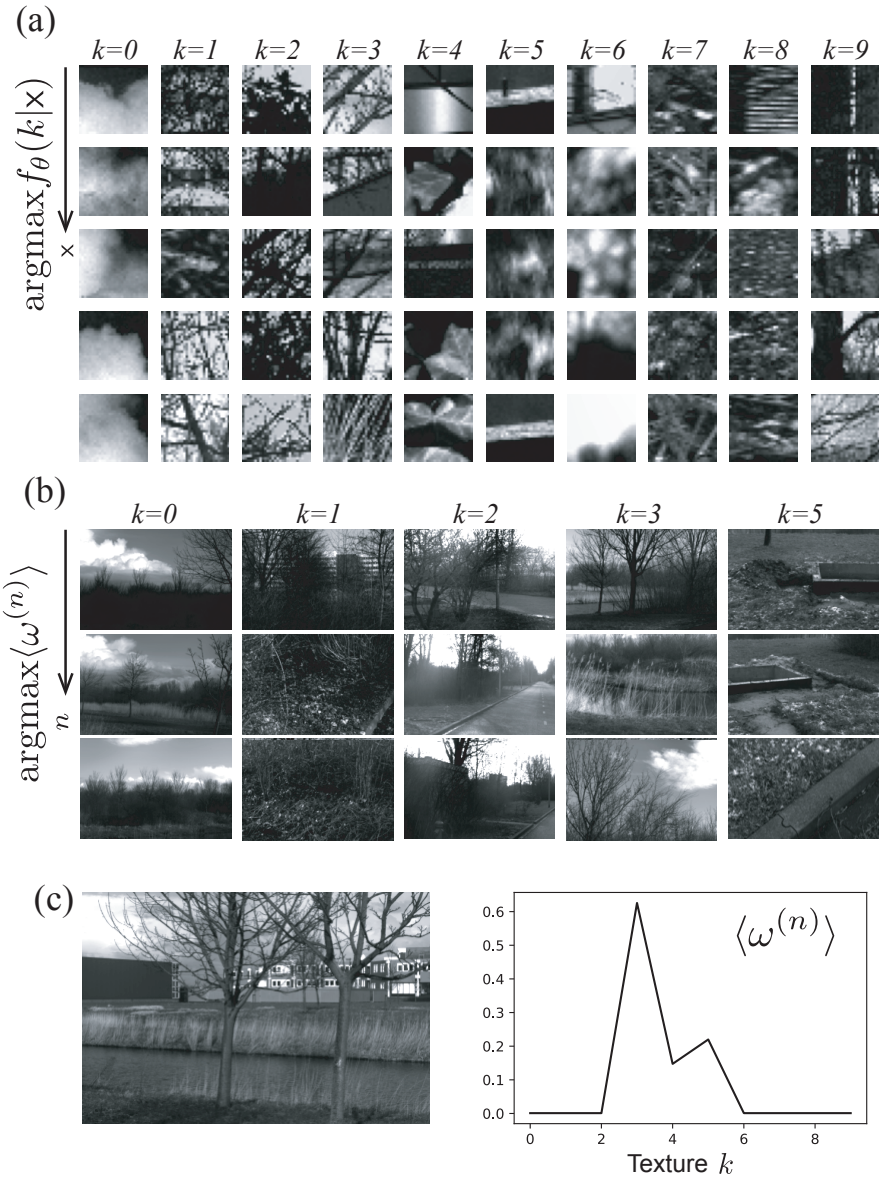
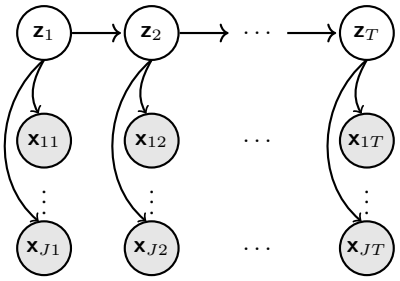
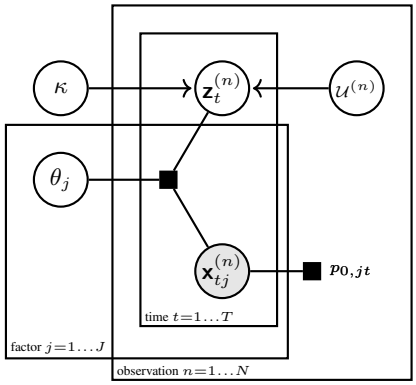


Figure A2: RPM-LDA. (a) Five most representative patches for all textures. (b) Three most representative images for some textures. (c) Texture distribution of a given image.

C CONTINUOUS EXPERIMENTS: RP-GPFA



Generative Model



RP-GPFA parametrisation

Recognition-Parametrised Gaussian Process Factor Analysis (RP-GPFA) models continuous multi-factorial temporal dependencies. We consider J observed time-series measured over T timesteps: $\mathcal{X} = \{\mathbf{x}_{jt} : j = 1 \dots J, t = 1 \dots T\}$. We seek to capture both spatial and temporal structure in a set of K-dimensional underlying latent time-series $\mathcal{Z} = \{\mathbf{z}_t : t = 1 \dots T\}$, such that the observations are conditionally independent across series and across time. The full joint has the form:

$$\mathsf{P}_{\theta,\mathbb{X}^{(N)}}(\mathcal{X}, \mathcal{Z}) = \prod_{j=1}^{J}\prod_{t=1}^{T}\left(p_{0,jt}(\mathbf{x}_{jt})\frac{f_{\theta j}(\mathbf{z}_t|\mathbf{x}_{jt})}{F_{\theta j}(\mathbf{z}_t)}\right)p_{\theta z}(\mathcal{Z})\,, \tag{A19}$$

Each recognition factor is parametrised by a neural network θ_j that outputs the natural parameters $\eta_j(\mathbf{x}_j^{(n)})$ of a multivariate normal distribution given input $\mathbf{x}_{jt}^{(n)}$ and we recall that

$$F_{\theta j}(\mathbf{z}_t) = \frac{1}{N}\sum_{n=1}^{N} f_{\theta j}\left(\mathbf{z}_t|\mathbf{x}_{jt}^{(n)}\right)\,. \tag{A20}$$

The prior on $\mathcal{Z}$ comprises independent Gaussian Process priors over each latent dimension $\mathcal{Z}_k = \{z_{kt} : t = 1 \dots T\}$

$$p_{\theta z}(\mathcal{Z}) = \prod_{k=1}^{K} p_k(\mathcal{Z}_k)\,;\, p_k(\cdot) = \mathcal{GP}(0, \kappa_k(\cdot,\cdot))\,, \tag{A21}$$

C.1 Variational Distribution and inducing points

We use sparse variational GP approximations (Titsias, 2009) to improve scalability of RP-GPFA. The model is augmented with M inducing points (IP) for each latent dimension ($k = 1 \dots K$) and each observation ($n = 1 \dots N$). This smaller set ($M < T$) of fictitious measurements is optimised to efficiently represent function evaluations. For simplicity, IP are defined at fixed and shared locations $\tau = [\tau_1, \dots, \tau_M]^\top$. We denote them

$$\mathcal{U}^{(n)} = \left[\mathcal{U}_1^{(n)}, \dots, \mathcal{U}_K^{(n)}\right] \sim M \times K\,. \tag{A22}$$

Given an observation n, the variational distribution writes

$$q\left(\mathcal{U}^{(n)}, \mathcal{Z}^{(n)}\right) = \prod_{k=1}^{K} q\left(\mathcal{U}_k^{(n)}, \mathcal{Z}_k^{(n)}\right)\,. \tag{A23}$$

In practice, we only need the marginals over inducing points and latents. The former are optimised numerically and denoted

$$q\left(\mathcal{U}_k^{(n)}\right) = \mathcal{N}\left(\mu_k^{(n)}, \Sigma_k^{(n)}\right)\,. \tag{A24}$$

For the latter, we simplify inference by adopting the form

$$q\left(\mathcal{U}_k^{(n)}, \mathcal{Z}_k^{(n)}\right) = p_k\left(\mathcal{Z}_k^{(n)} | \mathcal{U}_k^{(n)}\right) q\left(\mathcal{U}_k^{(n)}\right) , \tag{A25}$$

which gives closed form expression for

$$q\left(z_{k,t}^{(n)}\right) = \mathcal{N}\left(m_{k,t}^{(n)}, S_{k,t}^{(n)}\right) . \tag{A26}$$

Indeed, we denote $\kappa_k^\tau = \kappa_k(\tau, \tau)$ and use the law of total expectations to obtain

$$m_{kt}^{(n)} = \mathbb{E}_u(\mathbb{E}_{z|u}(z)) = \kappa_k(t,\tau)\kappa_k^{\tau-1}\mu_k^{(n)} \tag{A27}$$

and

$$s_{kt}^{(n)} = \mathbb{V}_u(\mathbb{E}_{z|u}(z)) + \mathbb{E}_u(\mathbb{V}_{z|u}(z)) = \kappa_k(t,\tau)\left(\kappa_k^{\tau-1}\Sigma_k^{(n)}\kappa_k^{\tau-1} - \kappa_k^{\tau-1}\right)\kappa_k(\tau,t) + \kappa_k(t,t) . \tag{A28}$$

We gather those centred moments in the K dimensional vector $\mathbf{m}_t^{(n)}$ and diagonal matrix $S_t^{(n)}$.

C.2 Variational Free Energy

The free energy is given by

$$\begin{aligned}
\log \mathsf{P}_{\theta,\mathbb{X}^{(N)}}(\mathcal{X}) &= \iint d\mathcal{Z}\, d\mathcal{U}\, \mathsf{P}_{\theta,\mathbb{X}^{(N)}}(\mathcal{X},\mathcal{Z},\mathcal{U}) \\
&\geq \iint d\mathcal{Z}\, d\mathcal{U}\, q\left(\mathcal{Z},\mathcal{U}\right) \log \frac{\mathsf{P}_{\theta,\mathbb{X}^{(N)}}(\mathcal{X}|\mathcal{Z}) p_{\theta z}(\mathcal{Z}|\mathcal{U}) p_{\theta z}(\mathcal{U})}{p_{\theta z}(\mathcal{Z}|\mathcal{U}) q(\mathcal{U})} \\
&= \langle \log \mathsf{P}_{\theta,\mathbb{X}^{(N)}}(\mathcal{X}|\mathcal{Z})\rangle_{q(\mathcal{Z},\mathcal{U})} - \mathsf{KL}\Big[q(\mathcal{U})\Big\|p_{\theta z}(\mathcal{U})\Big] = \mathcal{F}
\end{aligned} \tag{A29}$$

The KL divergence between the variational and the prior distribution over IP is closed form and can be broken down to

$$\mathsf{KL}\Big[q(\mathcal{U})\Big\|P(\mathcal{U})\Big] = \sum_{n,k} \mathsf{KL}\Big[q(\mathcal{U}_k^{(n)})\Big\|p_k(\mathcal{U}_k^{(n)})\Big] \tag{A30}$$

The remaining term has the RPM form

$$\left\langle \mathsf{P}_{\theta,\mathbb{X}^{(N)}}(\mathcal{X}|\mathcal{Z})\right\rangle_{q(\mathcal{Z},\mathcal{U})} = NJT\log\frac{1}{N} + \sum_{njt}\left\langle \log f_{\theta j}(\mathbf{z}_t|\mathbf{x}_{jt}^{(n)})\right\rangle_{q\left(\mathbf{z}_t^{(n)}\right)} - \left\langle \log F_{\theta j}(\mathbf{z}_t)\right\rangle_{q\left(\mathbf{z}_t^{(n)}\right)} , \tag{A31}$$

and is estimated by using one of the inference methods described above. Finally, the free energy (or its lower bound) is optimised with respect to the kernel parameters, the inducing point variational distributions, and the recognition networks (and the auxiliary factors) using Adam. When necessary, we ensure the validity of $\hat{f}_j^{(n)}$ by soft-thresholding the eigenvalues of the natural parameters.

C.3 RP-GPFA Experiments

In all RP-GPFA experiments, the recognition networks consisted in at least two fully connected hidden layers of size 50. When input included image frames, they were preceded by two convolutional layer with max pooling. All layers were followed by Rectified Linear Unit (ReLU) and trained with Adam.

Bouncing Balls

In Bouncing ball experiments, the latent was generated with a randomly initialised two dimensional oscillating linear system from which we extracted the first components. We fixed the number of observation to $N = 50$, the number of time points to $T = 50$, and used $M = 20$ inducing points. As described in the main text, in the textured experiment, the stochastic mapping from latent to observation is defined such that the mean and variance of pixel intensity is independent of the latent position (respectively fixed to 0 and 1). This is achieved with a mixture of Gaussian distributions with fixed variance, but whose weights and position depend on the latent.

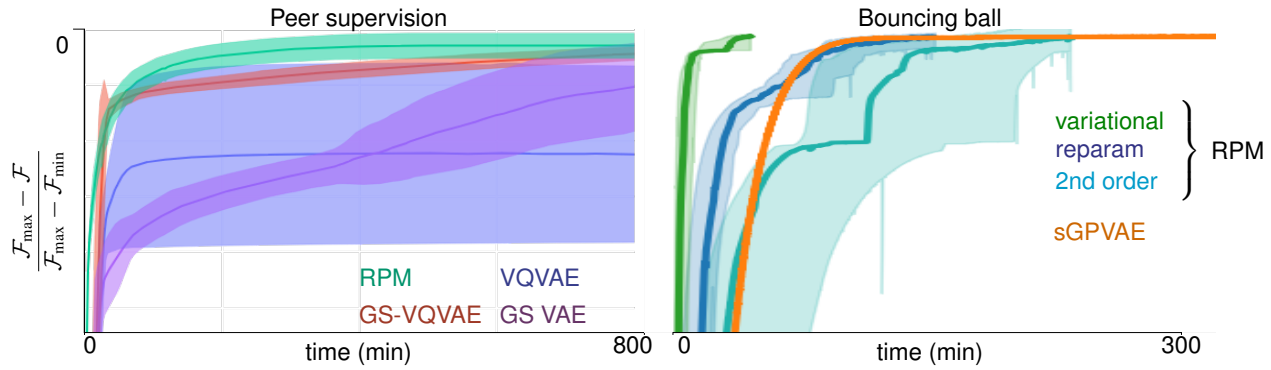


Figure A3: Relative free energy vs clock time for MNIST peer supervision (left) and a Bouncing ball data set (right). Free energy is shown relative to the highest value at convergence (rather than on an absolute scale) to emphasise relative timing. In both cases the RPM approaches converged to higher absolute values of free energy than alternatives.

Multi-factorial experiment

In the multi-factorial experiment, three independent agents are placed in a bounded environment where they work to move inanimate blocks to a designated target. Once the task is complete, the arena ground colour changes. Observations of the agents' locations are collected in the form of 3D-rendered image frames of the entire environment and noisy range-finding sensor data giving the distances of one of the agents from the four corners of the room. Sensor noise is modelled as additive Gaussian with zero mean and variance 0.1. Renderings and trajectories were generated using Unity Machine Learning Agents Toolkit (Juliani et al., 2018). We used $N = 50$ observations of length $T = 200$ and $M = 40$ inducing points.

D Compute time

Compute time for the RPM experiments was competitive with the baseline comparison methods for full-batch training with both discrete and continuous latents.

Fig. A3 shows wall-clock comparisons for MNIST peer supervision, and for a bouncing ball (with noise better matched for sGPVAE so that it converges to a non-trivial value). Learning curves are scaled vertically to emphasise relative timing. The RPM always converged to a higher value of free energy on an absolute scale.

We used 20 samples per latent in reparametrisation to ensure that the compute time was comparable. Note that although the variational method is fast here, the current implementation scales poorly with GP dimension.

E Code

Implementation and code of all discrete and continuous latent experiments can be found at `https://github.com/gatsby-sahani/rpm-aistats-2023`

References

M. Braun and J. McAuliffe. Variational inference for large-scale models of discrete choice. *Journal of the American Statistical Association*, 105(489):324–335, 2010.

A. Juliani, V.-P. Berges, E. Teng, A. Cohen, J. Harper, C. Elion, C. Goy, Y. Gao, H. Henry, M. Mattar, et al. Unity: A general platform for intelligent agents. *arXiv preprint arXiv:1809.02627*, 2018.

D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

C. J. Maddison, A. Mnih, and Y. W. Teh. The concrete distribution: A continuous relaxation of discrete random variables. *arXiv preprint arXiv:1611.00712*, 2016.

R. M. Neal and G. E. Hinton. A view of the EM algorithm that justifies incremental, sparse, and other variants. In M. I. Jordan, editor, *Learning in Graphical Models*, pages 355–370. Kluwer Academic Press, 1998.

C. K. Sønderby, B. Poole, and A. Mnih. Continuous relaxation training of discrete latent variable image models. *Baysian DeepLearning workshop, NIPS 2017*, 2017.

M. Titsias. Variational learning of inducing variables in sparse Gaussian processes. In *AISTATS*, pages 567–574. PMLR, 2009.

A. Van Den Oord, O. Vinyals, and K. Kavukcuoglu. Neural discrete representation learning. In *Adv Neural Info Processing Sys*, volume 30, pages 6309–6318, 2017.