Joint Learning of Discrete and Continuous Variability with Coupled Autoencoding Agents

Yeganeh M. Marghi, Rohan Gala, Uygar Sümbül
Allen Institute, WA, USA
{yeganeh.marghi; rohang; uygars}@alleninstitute.org

Abstract

Jointly identifying discrete and continuous factors of variability can help unravel complex phenomena. In neuroscience, a high-priority instance of this problem is the analysis of neuronal identity. Here, we study this problem in a variational framework by utilizing interacting autoencoding agents, designed to function in the absence of prior distribution over the discrete variable and reach collective decisions. We provide theoretical justification for our method and demonstrate improvements in terms of interpretability, stability, and accuracy over comparable approaches with experimental results on two benchmark datasets and a recent dataset of gene expression profiles of mouse cortical neurons. Furthermore, we demonstrate how our method can determine the neuronal cell types in an unsupervised setting, while identifying the genes implicated in regulating biologically relevant neuronal states.

1 Introduction

Complex phenomena often jointly exhibit discrete variability and continuous within-cluster variability, the former driving the need for clustering and the latter reflecting cluster-dependent processes beyond acquisition noise. Joint dependence on discrete and continuous factors arises in many contexts. Here, our main motivation comes from neuroscience: the notions of cell types and cell states can be considered as biological interpretations of discrete and continuous variability. Identifying such factors can be useful to study canonical brain circuits in terms of their generic components [1], and to identify gene expression programs [2], both of which are high-priority research areas. Although significant progress has been achieved in documenting the diversity of neurons, seemingly continuous transitions between types have posed unresolved challenges [3,4]. Single-cell molecular datasets shed new light on this problem by obtaining detailed measurements for each neuron, and significantly expanding the sample sizes [5]. While deep generative models have previously been applied to such datasets [6], the focus has been on demonstrating the diversity of expression and defining consensus clusters; the issue of neuronal state remains understudied.

Learning representations beyond assigning clusters has received attention from the deep learning community [7-9]. Combining neural networks with Gaussian mixture models [10-12] was among the first attempts to generalize the latent representation by decomposing it into discrete classes. However, identifying interpretable variational factors in the absence of structured prior information remains a significant challenge. Both adversarial learning [9] and variational autoencoder (VAE) [7,13,14] based approaches recently proposed representation learning methods, in which the general goal is to relate the latent factors of variability to explanatory factors of variation in the dataset. However, adversarial learning is susceptible to stability issues [7,8,15], and VAE-based methods with a single autoencoding agent rely either on a heuristic data-dependent embedding capacity [7], or on solving a separate optimization problem for the discrete variable [8]. Thus, learning interpretable and stable mixture representations remains challenging.

To address these issues, we propose a multi-agent variational framework to infer the discrete and continuous factors of variability in an unsupervised fashion. Coupling of classical (continuous factor only), non-probabilistic autoencoders via a cost term on the latent representations has been previously
studied in the context of multi-modal recordings, where each autoencoder is responsible for one of the data modalities [16,17]. Here, we utilize pairwise coupling architecture for joint representation learning for uni-modal data, in which agents collaborate with each other by sharing their categorical variables, while each agent receives a noisy (augmented) copy of the same sample. The key intuition is that, when such augmentation is representative of within-cluster variability, categorical assignments can be regularized by seeking an agreement across agents, thereby increasing inference robustness and removing the need to specify a prior on the relative abundances of the categories.

Our contributions can be summarized as follows: (i) We first show that collective decision making based on augmented copies of an observation can improve categorical assignments. Motivated by that, we posit a constrained optimization problem for multi-agent variational inference that enables a more general mixture representation that is not a function of the categorical prior distribution. (ii) We derive a relaxation of the original problem for efficient inference. We avoid the mode collapse problem by introducing a distance measure for categorical variables in the simplex, which we prove to be an approximation of the Aitchison distance. (iii) In addition to benchmarking the proposed framework with the MNIST and dSprites datasets, we apply it to a publicly available single cell gene expression dataset of cortical neurons to evaluate the utility of our approach in identifying cell types and cell states. (iv) We demonstrate that our approach can identify biologically meaningful clusters and states. We show that changing the state variable for a given category leaves certain marker genes essentially unchanged while correlating well with the expression levels of other, state-related genes.

2 Preliminaries

For an observation \( x \in \mathbb{R}^D \), the variational autoencoder (VAE) learns a generative model \( \log p_\theta (x|z) \) and a variational distribution \( \log q_\phi (z|x) \), where \( z \in \mathbb{R}^M \) for \( M \ll D \) is a latent variable with a parameterized distribution \( p(z) \) [18]. Disentangling different sources of variability into different dimensions of \( z \) enables an interpretable selection of latent variables [19,20], however, in many practical datasets the inherent continuous and discrete sources of variability are often overlooked. This problem can be addressed within the VAE framework in an unsupervised fashion by introducing a categorical latent variable \( c \in \mathcal{S}^K \), denoting the class label defined in a simplex \( \mathcal{S} \) with \( K \) components, alongside the continuous latent variable \( s \in \mathbb{R}^M \), representing the state or style [7,8]. Assuming \( s \) and \( c \) are independent random variables (RVs), the evidence lower bound (ELBO) [21] for a single autoencoding agent with the distributions parameterized by \( \theta \) and \( \phi \) is given by,

\[
\mathcal{L} = \mathbb{E}_{q_\phi (s,c|x)} [\log p_\theta (x|s,c)] - D_{KL} (q_\phi (s|x)||p(s)) - D_{KL} (q_\phi (c|x)||p(c)).
\]  

(1)

While the Gumbel-softmax distribution enables gradient-based optimization by providing a continuous relaxation of the discrete variable [13,14], maximizing ELBO in Eq. (1) for joint learning of \( q(s|x) \) and \( q(c|x) \) is challenging due to the posterior collapse problem, where the network ignores a subset of categorical variables [7].

2.1 Joint representation learning with single-agent variational autoencoders

Akin to \( \beta \)-VAE [19,22], JointVAE [7] has been suggested as an alternative for joint representation learning per Eq. (2).

\[
\mathcal{L} = \mathbb{E}_{q_\phi (s,c|x)} [\log p_\theta (x|s,c)] - \beta_s D_{KL} (q_\phi (s|x)||p(s)) - \mathcal{C}_s - \beta_c D_{KL} (q_\phi (c|x)||p(c)) - \mathcal{C}_c \quad (2)
\]

where \( \beta_s, \beta_c \geq 1 \) are adjustable hyperparameters to encourage disentanglement for continuous and categorical RVs, respectively. Here, \( \mathcal{C}_s \) and \( \mathcal{C}_c \) are the controlled capacities, which are gradually increased over training iterations. A drawback of the added flexibility afforded by embedding capacities is the need for dataset-dependent and empirical tuning [7,22].

Another approach, CascadeVAE [8], maximizes the ELBO in Eq. (1) by iterating over two separate optimizations for the continuous and categorical variables after a warm-up period in which a minimum cost flow algorithm is used to minimize \( \bar{D}_{KL} (q_\phi (c|x)||p(c)) \) instead of the fully gradient-based optimization. Although the discrete optimization problem in [8] has an approximately linear dependence on the number of categories and batch size, the training cost can still be a deterrent for problems with numerous categories and unbalanced datasets requiring larger batch sizes.

In summary, single-agent VAEs fall short of efficiently learning interpretable mixture representations, either due to their reliance on a heuristic embedding capacity (JointVAE), or lacking a fully variational approach (CascadeVAE).
We first prove a motivating statement on whether collective decision making on multiple augmented
where interpretability of the latent variables is desired. More clearly, if the role of the categorical
(augmented) copies of a single observation can be advantageous. Suppose the variable $z$ is to merely provide a discrete code [13], then the uniform prior on the latent variables
$\{z_a\}$ for all $m,n \in A$.

To overcome the issues discussed in Section 2, we propose a novel joint VAE framework using
multiple VAE agents, where each agent learns a mixture representation of continuous and discrete
RVs, in which the state variable is a function of the categorical assignment (Figure 1.b). The idea
behind this multi-agent framework is inspired by the “wisdom of the crowd” approach [23], which
often depends on class label. For instance, even for the well-studied MNIST dataset, the histograms
of two common digit styles (“rotation” and “width”) can markedly vary for different digits (Figure 1.a). Moreover, in JointVAE the unsupervised clustering of only the learned continuous variable $s$ achieves a relatively high classification accuracy ($\sim 66\%$) of the categorical variable $c$, underscoring that the independence assumption is invalid (see supplementary material, Sections F and G).

2.2 Common assumptions: Uniform prior and independent joint representation
In addition to the issues already discussed, the performance and interpretability of VAE models in
[7] and [8] is further limited by two common assumption. The first such assumption is the use of the
uniform distribution as the categorical prior $p(c)$, which encourages similar abundances of samples in each category. In practice, uniformity is the exception rather than the rule, as exemplified by the
highly non-uniformly distributed cortical single-cell gene expression dataset [5] we study in this paper. While prior distributions over latent variables are mostly inconsequential, problems arise when interpretability of the latent variables is desired. More clearly, if the role of the categorical variable(s) is to merely provide a discrete code [13], then the uniform prior on the latent variables is akin to the common Gaussian prior used in classical variational autoencoders. However, if the categorical variable is meant to represent the various innate clusters within the data, then the prior can significantly occlude such interpretations. The second common assumption is that the continuous state variable $s$ representing the style of the data is independent of the class label. In practice, style often depends on class label. For instance, even for the well-studied MNIST dataset, the histograms of two common digit styles (“rotation” and “width”) can markedly vary for different digits (Figure 1.a). Moreover, in JointVAE the unsupervised clustering of only the learned continuous variable $s$ achieves a relatively high classification accuracy ($\sim 66\%$) of the categorical variable $c$, underscoring that the independence assumption is invalid (see supplementary material, Sections F and G).

3 Coupled mix-VAE framework
To overcome the issues discussed in Section 2 we propose a novel joint VAE framework using
multiple VAE agents, where each agent learns a mixture representation of continuous and discrete
RVs, in which the state variable is a function of the categorical assignment (Figure 1.b). The idea
behind this multi-agent framework is inspired by the “wisdom of the crowd” approach [23], which
proposes the utilization of collective decision making.

3.1 Type-preserving augmentation
We first prove a motivating statement on whether collective decision making on multiple augmented
(noisy) copies of a single observation can be advantageous. Suppose the variable $z$ is generated by a
multivariate mixture distribution with $K$ components so that $p(z) = \sum_{k=1}^{K} p(\phi = k)p(z|\phi = k)$,
where $p(\phi)$ denotes an arbitrary prior for cluster abundance.

**Proposition 1.** Let $\tilde{p}_m(z) = \tilde{p}(z|\phi = m)$ denote the distribution of samples in the $m$-th cluster
observed through an augmentation (noise) step. If the augmented distribution satisfies

$$D_{KL}(\tilde{p}_m||\tilde{p}_n) \geq \frac{1}{A} D_{KL}(p_m||p_n) - \left(\frac{A-1}{A}\right) \log \frac{p(\phi = m)}{p(\phi = n)}$$

(3)

for all $m, n \in \{1, \ldots, K\}$, then

$$\mathbb{E}_{z_1:\text{A} \sim P\{z_1:\text{A}\}} \left[ \log \frac{p(\phi_1 = m, \ldots, \phi_A = m|z_1\text{A})}{p(\phi_1 = n, \ldots, \phi_A = n|z_1\text{A})} \right] \geq \mathbb{E}_{z \sim p(z|m)} \left[ \log \frac{p(\phi = m|z)}{p(\phi = n|z)} \right]$$

(4)

where $\{z_a\}_{1:A}$ denotes $A$ independent RVs receiving augmented data sampled from the same family
of augmentations for cluster $m$, i.e., $\tilde{p}_m(z)$. (Proof in supplementary material, Section A)
Thus, when Eq. (5) is satisfied, the expected clustering performance due to the collective decision of multiple augmented versions of a single observation (left side of Inequality (4)) is superior to a lone observation (right side of Inequality (4)). We refer to the augmentations satisfying Eq. (3) as type-preserving augmentations. In the supplementary material, Section A, we state two corollaries of Proposition 1 that enhance interpretability and relate to the uniform abundance assumption commonly used in machine learning.

3.2 Multi-agent VAE

To use the insight obtained from Proposition 1, collective decision making for an $A$-agent VAE network can be formulated as the following constrained optimization on augmented copies $x_1, \ldots, x_A$.

$$
\max_{\phi_1, \theta_1, x_1, s_1, c_1} \ell(\phi_1, \theta_1, x_1, s_1, c_1) + \cdots + \ell(\phi_A, \theta_A, x_A, s_A, c_A)
$$

$$
\text{s.t. } c_1 = \cdots = c_A
$$

Here $\ell(\phi_a, \theta_a, x_n, s_n, c_n)$ is the variational loss function for agent $a$, which approximates two discriminative models, $q(c_a|x_n)$ and $q(s_n|x_n, c_n)$, representing the type and the state variables respectively, using (suppressed) parameterizations $\theta_a$ and $\phi_a$ (see supplementary material for derivation, Section B).

$$
\ell(\phi, \theta, x, s, c) = \mathbb{E}_{q(s, c|\theta)} \left[ \log p(\theta|x, c) - \mathbb{E}_{q(c|x)} \left[ D_{KL} (q(s|c, x) \| p(s|c)) \right] \right]
$$

$$
- \mathbb{E}_{q(s|x, c)} \left[ D_{KL} (q(c|x) \| p(c)) \right].
$$

Not only it is challenging to solve the equality constrained maximization (ECM) problem in Eq. (5), it also remains a function of $p(c)$. To overcome this, we introduce an equivalent ECM formulation for Eq. (5) based on the pairwise coupling paradigm as follows (see supplementary material for a derivation of the equivalence between solving Eq. (7) and the ECM problem in Eq. (5), Section C).

$$
\max_{a=1}^A (A - 1) \left[ \mathbb{E}_{q(s_n, c_n|x_n)} \left[ \log p(x_n|s_n, c_n) \right] - \mathbb{E}_{q(c_n|x_n)} \left[ D_{KL} (q(s_n|c_n, x_n) \| p(s_n|c_n)) \right] \right]
$$

$$
- \sum_{a < b} \mathbb{E}_{q(s_n, c_n|x_n)} \mathbb{E}_{q(s_b, c_b|x_b)} \left[ D_{KL} (q(c_a|x_n)q(c_b|x_b) \| p(c_a, c_b)) \right]
$$

$$
\text{s.t. } c_a = c_b \forall a, b \in [1, A], a < b
$$

where the last term, the KL divergence across coupled agents, is a function of the joint distribution $p(c_a, c_b)$, rather than $p(c)$. We relax Eq. (7) into an unconstrained problem by assuming a differentiable form for $p(c_a, c_b)$ (full derivation in supplementary material, Section D).

$$
\max_{a=1}^A (A - 1) \left[ \mathbb{E}_{q(s_n, c_n|x_n)} \left[ \log p(x_n|s_n, c_n) \right] - \mathbb{E}_{q(c_n|x_n)} \left[ D_{KL} (q(s_n|c_n, x_n) \| p(s_n|c_n)) \right] \right]
$$

$$
+ \sum_{a < b} \mathbb{H}(c_a|x_n) + \mathbb{H}(c_b|x_n) - \lambda \mathbb{E}_{q(c_a|x_n)} \mathbb{E}_{q(c_b|x_b)} \left[ d^2(c_a, c_b) \right],
$$

where $\lambda$ is referred to as the coupling hyperparameter enforcing pairwise alignment between agents, and $d(c_a, c_b)$ denotes a distance between $c_a$ and $c_b$. To benefit from collective decision making, the VAE agents in Eq. (8) interact with each other by sharing their categorical latent variables (Figure 1b) and trying to achieve identical assignments while independently learning their own style variables. In the rest of this manuscript, we refer to the model in Eq. (8) as cpl-mixVAE. Note that, while we chose not to introduce a further hyperparameter here, this formulation can be trivially extended to include a $\beta_n \geq 1$ variable for each agent, as in Eq. (2), to regularize the continuous latent variable.

We use Aitchison geometry to compute $d(c_a, c_b) = ||\text{clr}(c_a) - \text{clr}(c_b)||^2_2$, where $\text{clr}(\cdot)$ denotes the isometric centered-log-ratio transformation introduced by Aitchison and $d$ satisfies the conditions of a mathematical metric $[24, 25]$. To sample from $q(c_a|x_n)$ in a gradient descent framework, we use the Gumbel-softmax distribution $[13, 14]$ with a temperature parameter $0 < \tau < 1$.

3.3 Stabilizing the training by mini-batch variance

The solution to the maximization problem in Eq. (8), which includes minimization of $d(c_a, c_b)$, has trivial local optima that result in the posterior collapse issue $[26]$, in which the network learns to ignore a subset of the discrete latent space. For instance, in the extreme case of $c_{a_n} = c_{b_n} = c, \forall x_{a_n}, x_{b_n} \in \mathbf{X}$, where $n$ denotes the sample index, the distance is trivially minimized. In this scenario, the state variable is compelled to act as the latent variable of a classical variational autoencoder, while the
We assess the performance of cpl-mixV AE for three different datasets. To facilitate comparisons with other methods, first we conducted experiments on two benchmark datasets: MNIST and dSprites. Additionally, we used a publicly available single cell RNA-sequencing dataset (FACS) [5], to evaluate the other methods, first we conducted experiments on two benchmark datasets: MNIST and dSprites. For cpl-mixVAE models, the coupling factor is set to $\lambda = 1$.

![Table 1: Clustering results for the MNIST dataset, over 10 runs with 15,000 training iterations, $|c| = 10, |s| = 10$, and $\tau = 0.67$. For cpl-mixVAE models, the coupling factor is set to $\lambda = 1$.]

| Method                  | $\mathcal{L}_{\text{rec}}$ | ACC $\uparrow$ (mean ± s.d.) |
|-------------------------|-----------------------------|--------------------------------|
| JointVAE                | 122.0                       | 68.99 ± 11.76                  |
| JointVAE$^\dagger$      | 126.7                       | 62.19 ± 05.73                  |
| JointVAE$^\ddagger$     | 130.5                       | 66.61 ± 09.20                  |
| CascadeVAE              | -                           | 81.41 ± 09.54                  |
| cpl-mixVAE(s|c)          | 109.5                       | 78.82 ± 09.43                  |
| cpl-mixVAE(s|c)          | 106.1                       | 82.92 ± 04.64                  |

Accordingly, as the Gumbel-softmax approximations of the categorical variable of the agents move closer to each other on the simplex, $d_\sigma$ converges to $d$.

### 4 Experiments

We assess the performance of cpl-mixVAE for three different datasets. To facilitate comparisons with other methods, first we conducted experiments on two benchmark datasets: MNIST and dSprites. Additionally, we used a publicly available single cell RNA-sequencing dataset (FACS) [5], to evaluate the utility of our approach in identifying cell types and cell states. Compared to typical machine learning datasets, the FACS dataset is very high-dimensional, with over 10,000 genes. It includes over 100 cell types with sizeable difference between the most and the least abundant clusters. The network architecture and training details are reported in the supplementary material, Section H.

#### 4.1 Benchmark datasets

**MNIST**: To learn interpretable representations for the MNIST dataset, we used a 10-dimensional categorical variable representing digits (type), and a 10-dimensional continuous random variable representing the state variable, which can be perceived as the writing style. During training, each agent receives an augmented copy of the original image consisting of random rotation, brightness and saturation changes. To interpret the role of the continuous factor, we fix the discrete latent variable and change the state variable according to the conditional state distribution learned for each category. Figure 2 illustrates these continuous latent traversal results for four dimensions of the state variable obtained from a bi-agent cpl-mixVAE with shared categorical variables. Each row corresponds to a different dimension of the categorical variable, and the state variable monotonically changes across the columns. Panels (a), (b), and (c) represent commonly-identified continuous factors with global attributes, while panel (d) represents roundness, all in a digit-dependent manner.

Table 1 displays the classification performance of the discrete latent variable (as the predicted class label) for different methods including JointVAE and CascadeVAE, where we report the loss value based on the reconstruction objective, $\log p(x|s, c)$, and the accuracy of the categorical assignments (ACC) across 10 initializations with different random seeds. For CascadeVAE, we used the numbers reported in [8] and for JointVAE, we used the publicly available implementation and hyperparameters detailed in [2]. We report the performance of the proposed coupled VAE for two cases: (i) cpl-mixVAE(s|c), in which the state variable is independent of the discrete variable, and (ii) cpl-mixVAE(s|c), which is the proposed VAE model in Eq. (5). The reported numbers for cpl-mixVAE models belong to only one of the agents (Further accuracy/stability improvements could be achieved by considering the results of both agents, but that is besides the focus of the present work). For a fair
Figure 2: Continuous latent traversals of cpl-mixVAE with 10-dimensional continuous and 10-dimensional
categorical variables. Examples of (a) rotation angle, (b) stroke thickness, (c) character width and (d) roundness
of looped features are presented. The discrete variable $c$ is constant for all reconstructions in the same row.

In many real-life applications, the true number of types is unknown. Accordingly, in a separate set
of experiments, we studied the performance of cpl-mixVAE($s|c$) for different cardinalities of the
categorical variable, $c$. Figure 3.a shows the performance of the proposed cpl-mixVAE in terms of
ACC and AMP as a function of $|c| = K \in [7, 13]$. Here, AMP denotes the average of maximum
probability of categories i.e., $1/K \sum_{k=1}^{K} \max p(c_k|x)$. Our experimental results show that the best
performance is obtained for $K = 10$. Figure 3.b demonstrates the sorted probabilistic assignment of $c$
for all digits. As illustrated, an inadequate number of categorical variables (left panel) results in some
dimensions being used for more than one digit, while extra $c_k$ leaves some categories under-utilized
(right panel).

dSprites: Similar to JointVAE and CascadeVAE, we used a 3-dimensional categorical variable
for learning the shape (type), and a 6-dimensional state variable representing the style of each
shape. During training, each agent received an augmented copy of the original image consisting
of random rotation, scaling, and translation. To interpret the role of the continuous factor, we fix
the discrete latent variable and change the state variable, as in the MNIST experiment. Figure 4
illustrates these traversal results for three dimensions of the state variable obtained from a 2-agent
cpl-mixVAE. Each row corresponds to a different dimension of the categorical variable, and the state
variable monotonically changes across the columns. In addition, Table 2 shows the degree to which
cpl-mixVAE outperforms the other two methods in clustering.

Figure 3: Clustering performance of cpl-mixVAE for MNIST, when the number of clusters ($|c|$) is unknown.
Figure 4: Continuous latent traversals of the trained cpl-mixVAE model with 6-dimensional continuous variable and 3-dimensional categorical variable. Examples of (a) rotation, (b) scale, (c) position features are presented. The discrete variable \( c \) is held as fixed for all reconstructions in the same row.

Table 2: Clustering results for the dSprites dataset, over 10 runs with \( |c| = 3, |s| = 6, \tau = 0.67, \) and \( \lambda = 10. \)

| JointVAE          | CascadeVAE | cpl-mixVAE(s \mid c) |
|-------------------|------------|----------------------|
| ACC (mean ± s.d.) | 44.79 ± 03.88 | 78.84 ± 15.65 | 96.30 ± 09.15 |

4.2 Analysis of cortical single-cell gene expression

As our main motivation, we applied the proposed method to a challenging dataset with practical importance, i.e. a single-cell RNA sequencing dataset (FACS). This dataset profiles gene expression of 22,365 cortical neurons for adult mouse [5]. Here, we excluded non-neuronal cells and used a subset of 5,000 most expressed genes based on their peak values. The original publication partitioned the neurons into 115 clusters referred to as cell types using a biologically-motivated, hierarchical clustering of marker gene expression [5]. From the perspective of neuroscience, it is not yet clear whether these genetic types correspond to meaningful circuit components with matching physiological and/or anatomical characteristics, or if the detailed categorization is a case of over-splitting. Moreover, individual cells can be different either because they belong to different types or different cell states [2, 27], the latter perhaps better explained by continuous factors of variability around cell type representatives. One final challenging aspect of categorical assignment in this dataset is its substantial imbalance, where for the most- and the least-abundant types, there exist 1404 and 16 samples, respectively.

We used a 115-dimensional discrete and a 2-dimensional continuous variable for cell type and cell state representations, respectively. We applied gene-wise dropout augmentation to train each VAE agent [28]. Figure 5.a illustrates the performance of the 2-agent cpl-mixVAE model trained over 45,000 iterations. The dendrograms on the y-axis displays the hierarchical relationship between neuron types according to [5]. For inhibitory neuronal cells, whether the observed diversity corresponds to discrete variability or a continuum is an ongoing debate [29, 30]. Consistently, while the classification accuracy of our model is lower for inhibitory neurons and 32% across the whole dataset (chance level is \( \sim 6\% \), based on the most abundant type), the accuracy significantly increases as clusters are merged according to the hierarchy defined by [5] (see supplementary material). Figure 5.b compares the clustering performance of JointVAE and cpl-mixVAE for different numbers of categories obtained for merged types. For instance, \( |c| = 2 \) corresponds to the highest node in the dendrogram, where there are only two neuronal types, excitatory and inhibitory. The top x-axis denotes the state variable dimension, which is gradually increased by merging neuronal types that leads to an increment in within-cluster variations.

Figure 5: Categorical assignments for the FACS dataset. (a) Confusion matrices of cpl-mixVAE and JointVAE using \( |c| = 115, |s| = 2, \tau = 1, \) and \( \lambda = 1. \) The dendrogram on the y-axis shows marker-based hierarchical classification with 115 cell types. (b) Classification accuracy, over 3 runs with 2M training iterations, as a function of number of categories by progressively merging the 115 types according to the hierarchical dendrogram.
To examine the role of the continuous latent variable (the cell state), we applied a similar traversal analysis to that used for the MNIST and dSprites datasets. For a given cell sample and its discrete type, we changed every dimension of the continuous state variable using conditional distribution $q(s|c)$, and inspected gene expression changes caused by state variable alterations. Figure 6 shows the results of the state traversal experiment for two excitatory neurons belonging to the “L5 NP” (near-projecting) and “L6 CT” (corticothalamic) sub-classes, and an inhibitory neuron belonging to the “PV” (parvalbumin) class. In each sub-figure, the latent traversal is color-mapped to normalized reconstructed expression values, where the y-axis corresponds to one dimension of the state variable, and the x-axis corresponds to four gene subsets, namely (i) marker genes (MG) for the two excitatory types identified by [5], (ii) immediate early genes (IEG), and two housekeeping gene (HKG) subgroups (iii) cytochrome c oxidase (COX), and (iv) cell cycle (CC) regulators [31, 32]. MGs are ideally expected to function as indicators for particular cell types such that their normalized expression is not affected by the state of the cell. Indeed, the expression of the reported excitatory MGs remains constant for excitatory traversals but not necessarily for the inhibitory traversal (i.e., Calb2, Gad2, Pde11a in Fig. 6). In contrast, the expression of IEGs and HKGs should depend strongly on the metabolic and environmental conditions. Indeed, we find that the expression changes of IEGs and HKGs are for the most part monotonically linked to the state variable, confirming that the state variable captures relevant, interpretable continuous variability, as in the MNIST and dSprites examples. Lastly, the expression of IEGs and HKGs (state-dependent genes) depends on the cell type e.g., not all IEGs are activated for all cell types. Notably, for the excitatory “L5 NP” and “L6 CT” cells that are proximate in the hierarchy (suggested in [5]), state traversal is quite similar. These results suggest that the state variable inferred by cpl-mixVAE provides insight when deciphering the molecular mechanisms shaping the landscape of cell states.

5 Conclusion

We have introduced cpl-mixVAE as a general framework to apply the power of collective decision making in unsupervised joint learning of discrete and continuous generative factors, without imposing a prior distribution on the discrete variable. The proposed framework utilizes multiple pairwise-coupled autoencoding agents with a shared categorical variable, while independently learning the continuous variables. We derived a variational formulation for this joint learning machine that conditions the continuous factor on the discrete latent variable, and showed that the resulting joint representations are more accurate, more stable, and yet interpretable. Our experimental results for all three datasets show that cpl-mixVAE outperforms the comparable VAE models. Finally, our results for a challenging gene expression dataset showed that the proposed framework can identify annotated neuronal types and differentiate between type-dependent and state-dependent genes. (See supplementary material for a discussion on future directions.)
**Broader Impact**

We do not expect our work to have immediate societal impact. Longer term, we consider two different kinds of potential impact: (i) The basic statistical machine learning aspect of our work improves the field of unsupervised clustering and representation learning. It is important to emphasize that these algorithms are data-driven, and as such, are bound to capture the biases present in the training datasets. We would like to warn against explicit and implicit sources of bias (e.g., gender, race, income) in the training examples. (ii) We expect that the computational tool studied in this paper will improve our understanding of the organization and function of biological systems, in particular the nervous system. Naturally, we are hopeful that our work will bring us closer to understanding how the brain works and the diseases of the nervous system. For example, many diseases are thought to have a cell type bias, where the progression of the disease may be reflected in the state of the cells. If the community finds our research useful, pharmaceutical companies may utilize it to develop drugs against certain diseases. In its current form, this is at best a remote possibility. While drug development would be a welcome outcome, we would like to warn against the fact that pharmaceutical productions are almost always protected by patents, which adversely affects the affordability and accessibility of drugs based on income and geographical location.

**Acknowledgements**

We wish to thank the Allen Institute for Brain Science founder, Paul G Allen, for his vision, encouragement and support.

**References**

[1] Cornelia Bargmann, William Newsome, A Anderson, E Brown, K Deisseroth, J Donoghue, P MacLeish, E Marder, R Normann, J Sanes, et al. Brain 2025: a scientific vision. *Brain Research through Advancing Innovative Neurotechnologies (BRAIN) Working Group Report to the Advisory Committee to the Director, NIH*, 2014.

[2] Cole Trapnell. Defining cell types and states with single-cell genomics. *Genome research*, 25(10):1491–1498, 2015.

[3] H Sebastian Seung and Uygar Sümbül. Neuronal cell types and connectivity: lessons from the retina. *Neuron*, 83(6):1262–1272, 2014.

[4] Hongkui Zeng and Joshua R Sanes. Neuronal cell-type classification: challenges, opportunities and the path forward. *Nature Reviews Neuroscience*, 18(9):530, 2017.

[5] Bosiljka Tasic, Zizhen Yao, Lucas T Graybuck, Kimberly A Smith, Thuc Nghi Nguyen, Darren Bertagnolli, Jeff Goldy, Emma Garren, Michael N Economou, Sarada Viswanathan, et al. Shared and distinct transcriptomic cell types across neocortical areas. *Nature*, 563(7729):72–78, 2018.

[6] Romain Lopez, Jeffrey Regier, Michael B Cole, Michael I Jordan, and Nir Yosef. Deep generative modeling for single-cell transcriptomics. *Nature methods*, 15(12):1053–1058, 2018.

[7] Emilien Dupont. Learning disentangled joint continuous and discrete representations. In *Advances in Neural Information Processing Systems*, pages 710–720, 2018.

[8] Yeonwoo Jeong and Hyun Oh Song. Learning discrete and continuous factors of data via alternating disentanglement. *arXiv preprint arXiv:1905.09432*, 2019.

[9] Xi Chen, Yan Duan, Rein Houthooft, John Schulman, Ilya Sutskever, and Pieter Abbeel. Infogan: Interpretable representation learning by information maximizing generative adversarial nets. In *Advances in neural information processing systems*, pages 2172–2180, 2016.

[10] Matthew J Johnson, David K Duvenaud, Alex Wiltshko, Ryan P Adams, and Sandeep R Datta. Composing graphical models with neural networks for structured representations and fast inference. In *Advances in neural information processing systems*, pages 2946–2954, 2016.

[11] Nat Dilokthanakul, Pedro AM Mediano, Marta Garnelo, Matthew CH Lee, Hugh Salimbeni, Kai Arulkumar, and Murray Shanahan. Deep unsupervised clustering with gaussian mixture variational autoencoders. *arXiv preprint arXiv:1611.02648*, 2016.
[12] Zhuxi Jiang, Yin Zheng, Huachun Tan, Bangsheng Tang, and Hanning Zhou. Variational deep embedding: an unsupervised and generative approach to clustering. In Proceedings of the 26th International Joint Conference on Artificial Intelligence, pages 1965–1972, 2017.

[13] Eric Jang, Shixiang Gu, and Ben Poole. Categorical reparameterization with gumbel-softmax. arXiv preprint arXiv:1611.01144, 2016.

[14] Chris J Maddison, Daniel Tarlow, and Tom Minka. A* sampling. In Advances in Neural Information Processing Systems, pages 3086–3094, 2014.

[15] Hyunjik Kim and Andriy Mnih. Disentangling by factorising. arXiv preprint arXiv:1802.05983, 2018.

[16] Fangxiang Feng, Xiaojie Wang, and Ruifan Li. Cross-modal retrieval with correspondence autoencoder. In Proceedings of the 22nd ACM international conference on Multimedia, pages 7–16, 2014.

[17] Rohan Gala, Nathan Gouwens, Zizhen Yao, Agata Budzillo, Osnat Penn, Bosiljka Tasic, Gabe Murphy, Hongkui Zeng, and Uygar Sümübil. A coupled autoencoder approach for multi-modal analysis of cell types. In Advances in Neural Information Processing Systems, pages 9263–9272, 2019.

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

[19] Irina Higgins, Loic Matthey, Arka Pal, Christopher Burgess, Xavier Glorot, Matthew Botvinick, Shakir Mohamed, and Alexander Lerchner. beta-vae: Learning basic visual concepts with a constrained variational framework. Iclr, 2(5):6, 2017.

[20] Francesco Locatello, Stefan Bauer, Mario Lucic, Gunnar Rätsch, Sylvain Gelly, Bernhard Schölkopf, and Olivier Bachem. Challenging common assumptions in the unsupervised learning of disentangled representations. arXiv preprint arXiv:1811.12359, 2018.

[21] John Aitchison. The statistical analysis of compositional data. Journal of the Royal Statistical Society: Series B (Methodological), 44(2):139–160, 1982.

[22] Juan José Egozcue, Vera Pawlowsky-Glahn, Glòria Mateu-Figueras, and Carles Barcelo-Vidal. Isometric logratio transformations for compositional data analysis. Mathematical Geology, 35(3):279–300, 2003.

[23] James Surowiecki. The wisdom of crowds. Anchor, 2005.

[24] Giorgio A Ascoli, Lidia Alonso-Nanclares, Stewart A Anderson, German Barrionuevo, Ruth Benavides-Piccione, Andreas Burkhalter, György Buzsáki, Bruno Cauli, Javier DeFelipe, Alfonso Fairén, et al. Petilla terminology: nomenclature of features of gabaergic interneurons of the cerebral cortex. Nature Reviews Neuroscience, 9(7):557, 2008.

[25] Kenneth D Harris, Hannah Hochgerner, Nathan G Skene, Lorenza Magno, Linda Katona, Carolina Bengts-son Gonzales, Peter Somogyi, Nicoletta Kessaris, Sten Linnarsson, and Jens Hjerling-Leffler. Classes and continua of hippocampal ca1 inhibitory neurons revealed by single-cell transcriptomics. PLoS biology, 16(6):e2006387, 2018.

[26] Sinisa Hrvatin, Daniel R Hochbaum, M Aurel Nagy, Marcelo Cicconet, Keiramarie Robertson, Lucas Cheadle, Rapolas Zilionis, Alex Ratner, Rebeca Borges-Monroy, Alon M Klein, et al. Single-cell analysis of experience-dependent transcriptomic states in the mouse visual cortex. Nature neuroscience, 21(1):120–129, 2018.
[32] Tatyana N Tarasenko, Susan E Pacheco, Mary Kay Koenig, Julio Gomez-Rodriguez, Senta M Kapnick, Francisca Diaz, Patricia M Zerfas, Emanuele Barca, Jessica Sudderth, Ralph J DeBerardinis, et al. Cytochrome c oxidase activity is a metabolic checkpoint that regulates cell fate decisions during T cell activation and differentiation. *Cell metabolism*, 25(6):1254–1268, 2017.
A Type-preserving augmentation

Proposition 1. Suppose $\tilde{p}_m(z)$ is the distribution of augmented samples in the $m$-th cluster i.e., $z \sim p_m(z)$, where $p_m(z) = p(z|\phi = m)$. If the augmented distribution satisfies

$$D_{KL}(\tilde{p}_m || p_n) \geq \frac{1}{A} D_{KL}(p_m || p_n) - \left(\frac{A-1}{A}\right) \log \frac{p(\phi = m)}{p(\phi = n)}$$

for $A \geq 1$ and all $m, n \in \{1, \ldots, K\}$, then

$$E_{\{z_a\}_{1:A} \sim \Pi_n \tilde{p}(z_a|m)} \left[ \log \frac{p(\phi = m|z)}{p(\phi = n|z)} \right] \geq E_{z \sim \tilde{p}(z|m)} \left[ \log \frac{p(\phi = m|z)}{p(\phi = n|z)} \right]$$

where $\{z_a\}_{1:A}$ denotes $A$ independent RVs receiving augmented data sampled from the same family of augmentations for cluster $m$, i.e., $\tilde{p}_m(z)$.

Proof. According to Eq. (9), we have,

$$D_{KL}(p_m || p_n) + (A-1) \log \frac{p(\phi = n)}{p(\phi = m)} \leq AD_{KL}(\tilde{p}_m || \tilde{p}_n)$$

and

$$E_{p_m(z)} \left[ \log \frac{p_m(z)}{p_n(z)} \right] + A \log \frac{p(\phi = m)}{p(\phi = n)} \leq \frac{1}{A} D_{KL}(\tilde{p}_m || \tilde{p}_n) - \left(\frac{A-1}{A}\right) \log \frac{p(\phi = m)}{p(\phi = n)}$$

Therefore, to satisfy Eq. (10) for all categorical components, $A \geq \max_{m,n} \left\{ D_{KL}(p_m || p_n) / D_{KL}(\tilde{p}_m || \tilde{p}_n) \right\}$.

Corollary 1. For uniformly distributed categorical components and augmentations satisfying Eq. (9), Eq. (10) is satisfied when the number of agents $A$ satisfies

$$A \geq \max_{m,n} \left\{ D_{KL}(p_m || p_n) / D_{KL}(\tilde{p}_m || \tilde{p}_n) \right\}$$

Proof. For uniformly distributed clusters, $\log \frac{p(\phi = m)}{p(\phi = n)} = 0$, so that Eq. (9) simplifies to

$$A \geq \frac{D_{KL}(p_m || p_n)}{D_{KL}(\tilde{p}_m || \tilde{p}_n)}$$

Therefore, to satisfy Eq. (10) for all categorical components, $A \geq \max_{m,n} \left\{ D_{KL}(p_m || p_n) / D_{KL}(\tilde{p}_m || \tilde{p}_n) \right\}$.

Corollary 2. If $D_{KL}(\tilde{p}_m || \tilde{p}_n) = D_{KL}(p_m || p_n)$, Eq. (10) is satisfied when

$$D_{KL}(\tilde{p}_m || \tilde{p}_n) \geq \log \frac{p(\phi = n)}{p(\phi = m)}$$

Proof. According to $D_{KL}(p_m || p_n) = D_{KL}(\tilde{p}_m || \tilde{p}_n)$, we can rewrite Eq. (17) as follows.

$$D_{KL}(\tilde{p}_m || \tilde{p}_n) \geq \log \frac{p(\phi = n)}{p(\phi = m)}$$

$$(A-1) D_{KL}(\tilde{p}_m || \tilde{p}_n) \geq (A-1) \log \frac{p(\phi = n)}{p(\phi = m)}$$

$$A D_{KL}(\tilde{p}_m || \tilde{p}_n) \geq (A-1) \log \frac{p(\phi = n)}{p(\phi = m)} + D_{KL}(\tilde{p}_m || \tilde{p}_n)$$

$$\geq (A-1) \log \frac{p(\phi = n)}{p(\phi = m)} + D_{KL}(p_m || p_n)$$

Supplementary Materials
which is equivalent to Eq. (9). Therefore, based on Proposition 1, the categorical assignment is guaranteed to be improved when there are multiple observations.

Trivially, from Corollaries 1 and 2, we can conclude that for uniformly distributed categorical components and for any augmentation satisfying \( D_{KL}(\tilde{p}_m||\tilde{p}_n) \approx D_{KL}(p_m||p_n) \), the inequality in Eq. (10) is always satisfied, which exemplifies the advantage of utilizing collective decision making on multiple augmented observations rather than a single one.

B Variational lower bound for conditional single mix-VAE

For completeness, we first derive the evidence lower bound (ELBO) for an observation \( x \) described by one categorical random variable (RV), \( c \), and one continuous RV, \( s \), without assuming conditional independence of \( c \) and \( s \) given \( x \). The variational approach to choosing the latent variables corresponds to solving the optimization equation

\[
q^*(s, c|x) = \arg \min_{q(s, c|x) \in \mathcal{D}} D_{KL}(q(s, c|x)||p(s, c|x))
\]

where \( \mathcal{D} \) is a family of density functions over the latent variables. However, evaluating the objective function requires knowledge of \( p(x) \), which is usually unknown. Therefore, we rewrite the divergence term as

\[
D_{KL}(q(s, c|x)||p(s, c|x)) = \int_s \sum_c q(s|c, x)q(c|x) \log \frac{q(s|c, x)q(c|x)}{p(x|s, c)p(s|c)p(c)} ds
\]

\[
= \int_s \sum_c q(s|c, x)q(c|x) \log \frac{q(s|c, x)}{p(s|c)} ds
+ \int_s \sum_c q(s|c, x)q(c|x) \log \frac{q(c|x)}{p(c)} ds
+ \int_s \sum_c q(s|c, x)q(c|x) \log p(x|s, c) ds
- \int_s \sum_c q(s|c, x)q(c|x) \log p(x|s, c) ds
= \log p(x) - \mathbb{E}_{q(c|x)} \left[ \mathbb{E}_{q(s|c, x)} [\log p(x|s, c)] \right]

+ \mathbb{E}_{q(c|x)} [D_{KL}(q(s|c, x)||p(s|c))] + \mathbb{E}_{q(s|c, x)} [D_{KL}(q(c|x)||p(c))] \tag{20}
\]

Since \( \log p(x) \) is unknown, instead of minimizing \( \mathcal{L}_a \), the variational lower bound

\[
\mathcal{L}_a = \mathbb{E}_{q(c|x)} \left[ \mathbb{E}_{q(s|c, x)} [\log p(x|s, c)] \right] - \mathbb{E}_{q(c|x)} [D_{KL}(q(s|c, x)||p(s|c))] - \mathbb{E}_{q(s|c, x)} [D_{KL}(q(c|x)||p(c))] \tag{22}
\]

can be maximized. We choose \( q(s|c, x) \) to be a factorized Gaussian, parametrized using the reparametrization trick, and assume that the corresponding prior distribution is also a factorized Gaussian, \( s|c \sim \mathcal{N}(0, I) \). Similarly, for the categorical variable, we assume a uniform prior, \( c \sim U(K) \).

C Variational inference for multi-agent autoencoding networks

As discussed in the main text, the collective decision making for an A-agent VAE network can be formulated as an equality constrained optimization as follows.

\[
\max \mathcal{L}(\phi_1, \theta_1, x_1, s_1, c_1) + \cdots + \mathcal{L}(\phi_A, \theta_A, x_A, s_A, c_A)
\text{s.t. } c_1 = \cdots = c_A \tag{23}
\]

Without loss of generality, the optimization in Eq. (23) can be rephrased as follows.

\[
\max \mathcal{L}(\phi_1, \theta_1, s_1, c_1) + \mathcal{L}(\phi_2, \theta_2, s_2, c_2) + \cdots + \mathcal{L}(\phi_A, \theta_A, s_A, c_A)
\text{s.t. } c_1 = c_2
\text{ } c_1 = c_3
\text{ } \cdots 
\text{ } c_1 = c_A
\text{ } \cdots 
\text{ } c_{A-1} = c_A \tag{24}
\]
where the equality constraint is represented as $\binom{A}{2}$ pairs of categorical agreements. Multiplying the objective term in Eq. (23) by a constant value, $A - 1$, we obtain,

$$\max_{\text{s.t. } c_a = c_b \forall a, b \in [1, A], a < b} \left( A - 1 \right) \left( \mathcal{L}(\phi_1, \theta_1, s_1, c_1) + \mathcal{L}(\phi_2, \theta_2, s_2, c_2) + \cdots + \mathcal{L}(\phi_A, \theta_A, s_A, c_A) \right)$$

(25)

Consider one pair of $\mathcal{L}$ objectives for two agents $a$ and $b$:

$$\mathcal{L}(\phi_a, \theta_a, s_a, c_a) + \mathcal{L}(\phi_b, \theta_b, s_b, c_b) = \mathbb{E}_{q\phi_a(s_a | c_a)} \left[ \log p_{\theta_a}(x_a | s_a, c_a) \right] + \mathbb{E}_{q\phi_b(s_b | c_b)} \left[ \log p_{\theta_b}(x_b | s_b, c_b) \right]$$

$$- \mathbb{E}_{q\phi_a(s_a | x_a)} \left[ D_{KL}(q(\phi_a(s_a | c_a | x_a)) || p(s_a | c_a)) \right] - \mathbb{E}_{q\phi_b(s_b | x_b)} \left[ D_{KL}(q(\phi_b(s_b | c_b | x_b)) || p(s_b | c_b)) \right]$$

(26)

Since all agents receive augmented samples from the same original distribution, we have $p(c_a) = p(c_b) = p(c)$. Using a simplified notation, $q_a = q\phi_a(s_a | x_a)$, the last two KL divergence terms can be expressed as,

$$D_{KL}(q_a || p(c)) + D_{KL}(q_b || p(c)) = \sum_{c_a} q_a \log \frac{q_a}{p(c)} + \sum_{c_b} q_b \log \frac{q_b}{p(c)}$$

$$= \sum_{c_a} \sum_{c_b} q_a q_b \log \frac{q_a}{p(c)} + \sum_{c_a} \sum_{c_b} q_a q_b \log \frac{q_b}{p(c)}$$

(27)

Now, if we marginalize $p(c)$ over the joint distribution $p(c_a, c_b)$, we can represent the categorical prior distribution as follows.

$$p(c) = \sum_{c_a, c_b} p(c | c_a, c_b) p(c_a, c_b)$$

(28)

Since there is a categorical agreement condition i.e., $c_a = c_b$, $p(c)$ can be expressed as,

$$p(c) = \sum_m p(c | c_a = c_b = m) p(c_a = c_b = m)$$

(29)

where

$$p(c | c_a = c_b = m) = \begin{cases} 1 & m = c \\ 0 & \text{otherwise} \end{cases}$$

(30)

Accordingly, under the $c_a = c_b$ constraint, we merge those KL divergence terms as follows:

$$D_{KL}(q_a || p(c)) + D_{KL}(q_b || p(c)) = \sum_{c_a} \sum_{c_b} q_a q_b \log \frac{q_a q_b}{p(c_a, c_b)}$$

(31)

Finally, the optimization in Eq. (25) can be expressed as

$$\max_{\text{s.t. } c_a = c_b \forall a, b \in [1, A], a < b} \left( A - 1 \right) \left( \mathcal{L}(\phi_a, \theta_a, s_a, c_a) - \mathbb{E}_{q(\phi_a | c_a)} \left[ D_{KL}(q(s_a | c_a) || p(s_a | c_a)) \right] - \mathbb{E}_{q(\phi_b | c_b)} \left[ D_{KL}(q(s_b | c_b) || p(s_b | c_b)) \right] \right)$$

(32)

### D Variational lower bound for coupled mix-VAE

In this section, using a pair of VAE agents, first we generalize the loss function for the single mix-VAE i.e., $\mathcal{L}_a$ in Eq. (22), to the multi-agent case, and show that we can achieve the same objective function in Eq. (22). Then, we derive a relaxation for the equality constrained optimization.

Given input data $x_a$, an agent approximates two models $q(c_a | x_a)$ and $q(s_a | x_a, c_a)$. If we use pairwise coupling to allow interactions between the agents, then, for a pair of VAE agents, $a$ and $b$, the variational lower bound
obtained from the KL divergence in Equation (20) can be generalized as
\[
\Delta(a, b) \triangleq D_{KL} (q(s_a, s_b, c_a, c_b|x_a, x_b) || p(s_a, s_b, c_a, c_b|x_a, x_b))
\]
\[
= \int_{s_a} \int_{s_b} \sum_{c_a} \sum_{c_b} q(s_a, s_b|c_a, c_b, x_a, x_b) q(c_a, c_b|x_a, x_b)
\]
\[
\times \log \left( \frac{q(s_a, s_b|c_a, c_b, x_a, x_b) q(c_a, c_b|x_a, x_b)}{p(s_a, s_b|c_a, c_b, x_a, x_b) p(c_a, c_b)} \right) ds_a ds_b
\]

When each agent learns the continuous factor independent of other agents, we have \(q(s_a, s_b|c_a, c_b, x_a, x_b) = q(s_a|c_a, x_a) q(s_b|c_b, x_b)\). Equivalently, for independent samples \(x_a\) and \(x_b\), we have \(q(c_a, c_b|x_a, x_b) = q(c_a|x_a) q(c_b|x_b)\). Hence,
\[
\Delta(a, b) = \log p(x_a, x_b) + \int_{s_a} \sum_{c_a} q(s_a|c_a, x_a) q(c_a|x_a) \log \frac{q(s_a|c_a, x_a)}{p(s_a|c_a)} ds_a
\]
\[
+ \int_{s_b} \sum_{c_b} q(s_b|c_b, x_b) q(c_b|x_b) \log \frac{q(s_b|c_b, x_b)}{p(s_b|c_b)} ds_b
\]
\[
- \int_{s_a} \sum_{c_a} q(s_a|c_a, x_a) q(c_a|x_a) \log p(x_a|s_a, c_a) ds_a
\]
\[
- \int_{s_b} \sum_{c_b} q(s_b|c_b, x_b) q(c_b|x_b) \log p(x_b|s_b, c_b) ds_b
\]
\[
+ \int_{s_a} \int_{s_b} \sum_{c_a} \sum_{c_b} q(s_a|c_a, x_a) q(s_b|c_b, x_b) q(c_a|x_a) q(c_b|x_b) \log \frac{q(c_a|x_a) q(c_b|x_b)}{p(c_a, c_b)} ds_a ds_b
\]

(34)

Therefore, the variational lower bound for a pair of coupled VAE agents can be expressed as,
\[
\mathcal{L}_{pair}(a, b) = \mathbb{E}_{q(s_a|c_a, x_a)}[\log p(x_a|s_a, c_a)] + \mathbb{E}_{q(s_b|c_b, x_b)}[\log p(x_b|s_b, c_b)]
\]
\[
- \mathbb{E}_{q(s_a|c_a, x_a)}[D_{KL} (q(s_a|c_a, x_a) || p(s_a|c_a))] - \mathbb{E}_{q(s_b|c_b, x_b)}[D_{KL} (q(s_b|c_b, x_b) || p(s_b|c_b))]
\]
\[
+ \mathbb{E}_{q(s_a|c_a, x_a)}[\mathbb{E}_{q(s_b|c_b, x_b)}[D_{KL} (q(c_a|x_a) q(c_b|x_b) || p(c_a, c_b))]] + \log p(x_a, x_b)
\]

(35)

which is equivalent to the loss function in Eq. (32), for \(A = 2\).

To approximate the joint distribution \(p(c_a, c_b)\), here, we define an auxiliary continuous random variable \(\epsilon\) with an exponential probability density function with parameter \(\lambda\), i.e., \(f(\epsilon; \lambda)\), representing the mismatch error between \(c_a\) and \(c_b\) such that \(\forall c_a, c_b \in S^K, \epsilon > 0\), and \(\lambda > 0\,
\]
\[
p(c_a, c_b|\epsilon) = \begin{cases} 1 & \left| \epsilon - d^2(c_a, c_b) \right| < \epsilon/2 \\ 0 & \text{otherwise} \end{cases}
\]

(37)

Here, \(d(c_a, c_b)\) denotes the distance between \(c_a\) and \(c_b\) in the simplex \(S^K\), as a measure of mismatch between categorical variables. Accordingly, the joint categorical distribution can be represented as,
\[
p(c_a, c_b) = \int p(c_a, c_b|\epsilon) f(\epsilon) d\epsilon
\]
\[
= \int_{\epsilon/2}^{\epsilon/2 + d^2(c_a, c_b)} f(\epsilon, \lambda) d\epsilon \approx \epsilon f(\epsilon^2, \lambda), \lambda
\]
\[
\approx \epsilon \lambda \exp(-\lambda d^2(c_a, c_b)),
\]

(39)

where the approximation is valid for small values of \(\epsilon\). Thus, the last KL divergence in Eq. (36) can be approximated as,
\[
D_{KL} (q(c_a|x_a) q(c_b|x_b) || p(c_a, c_b)) = \sum_{c_a} \sum_{c_b} q(c_a|x_a) q(c_b|x_b) \log \frac{q(c_a|x_a) q(c_b|x_b)}{p(c_a, c_b)}
\]
\[
= -H(c_a|x_a) - H(c_b|x_b) - \sum_{c_a} \sum_{c_b} q(c_a|x_a) q(c_b|x_b) \log p(c_a, c_b)
\]
\[
\approx -H(c_a|x_a) - H(c_b|x_b) + \lambda \mathbb{E}_{q(c_a|x_a)} \mathbb{E}_{q(c_b|x_b)} \left[ d^2(c_a, c_b) \right] - \log(41)
\]

(40)
Therefore, the approximated variational cost for a pair of VAE agents can be written as follows (since log ϵλ is a constant, not a function of the variational parameters, it can be discarded from the loss function).

\[
\mathcal{L}_{\text{pair}}(a, b) = E_{q(s_u, c_u | x_u)} \left[ \log p(x_1 | s_u, c_u) + E_{q(s_v, c_v | x_v)} \left[ \log p(x_1 | s_v, c_v) \right] \right]
- E_{q(c_v | x_v)} \left[ D_{KL} \left( q(s_v | c_v, x_v) \parallel p(s_v | c_v) \right) \right] - E_{q(c_v | x_v)} \left[ D_{KL} \left( q(s_v | c_v, x_v) \parallel p(s_v | c_v) \right) \right]
+ H(c_v | x_v) - \lambda E_{q(c_v | x_v)} E_{q(c_v | x_v)} \left[ d^2(c_v, c_v) \right]
\]

Equation (42)

Now, by extending \( \mathcal{L}_{\text{pair}} \) from two agents to \( A \) agents, in which there are \( \binom{A}{2} \) paired networks, the total loss function for \( A \) agents can be written as

\[
\mathcal{L}_{\text{cpl}} = \sum_{a=1}^{A} \sum_{b=a+1}^{A} \mathcal{L}_{\text{pair}}(a, b)
\]

Equation (43)

\[
= \sum_{a=1}^{A} (A - 1)E_{q(s_a, c_a | x_a)} \left[ \log p(x_1 | s_a, c_a) \right] - (A - 1)E_{q(c_a | x_a)} \left[ D_{KL} \left( q(s_a | c_a, x_a) \parallel p(s_a | c_a) \right) \right]
+ \sum_{a < b} H(c_a | x_a) + H(c_b | x_b) - \lambda E_{q(c_a | x_a)} E_{q(c_b | x_b)} \left[ d^2(c_a, c_b) \right] .
\]

E  Proof of Proposition 2

In this section, we first briefly review some critical definitions in Aitchison geometry. Then, to support the proof of Proposition 2, here we introduce Lemma 1 and Propositions 3 and 4.

According to Aitchison geometry, a simplex of \( K \) parts can be considered as a vector space \( (S^K, \oplus, \odot) \), in which \( \oplus \) and \( \odot \) corresponds to perturbation and power operations, respectively, as follows.

**Perturbation**: \( \forall x, y \in S^K, \ x \oplus y = C(x_1y_1, \ldots, x_Ky_K) \)

**Power**: \( \forall x \in S^K \) and \( \forall \alpha \in \mathbb{R}, \ \alpha \odot x = C(x_1^\alpha, \ldots, x_K^\alpha) \)

where \( C \) denotes the closure operation as follows.

\[
C(x) = \left( \frac{c x_1}{\sum_{k=1}^{K} x_k}, \ldots, \frac{c x_K}{\sum_{k=1}^{K} x_k} \right) .
\]

In the simplex vector space, for any \( x, y \in S^K \), the distance is defined as,

\[
d_{S^K}(x, y) = \left( \frac{1}{K} \sum_{i<j} \left( \log \frac{x_i}{x_j} - \log \frac{y_i}{y_j} \right)^2 \right)^{1/2} .
\]

Equation (44)

Furthermore, Aitchison has introduced centered-logratio transformation (CLR), which is an isometric transformation from a simplex to a \( K \)–dimensional real space, \( clr(x) \in \mathbb{R}^K \). The CLR transformation involves the logratio of each \( x_k \) over geometric means in the simplex as follows.

\[
clr(x) = \left( \log \frac{x_1}{g(x)}, \ldots, \log \frac{x_K}{g(x)} \right) .
\]

Equation (45)

where \( g(x) = \left( \prod_{k=1}^{K} x_k \right)^{1/K} \) and \( \sum_{k=1}^{K} \log \frac{x_k}{g(x)} = 0 \).

Since CLR is an isometric transformation, we have

\[
d_{S^K}(x, y) = d_{S^K}(clr(x), clr(y)) = \|clr(x) - clr(y)\|_2
\]

The algebraic-geometric definition of \( S^K \) satisfies standard properties, such as

\[
d_{S^K}(x \oplus u, y \oplus u) = d_{S^K}(x \oplus u, y \oplus u) = d_{S^K}(x, y)
\]

Equation (46)

where \( u \in S^K \) could be any arbitrary vector in the simplex.
Lemma 1. Given a set of vectors \( \{x_1, \ldots, x_N\} \in S^K \) where \( S^K \) is a simplex of \( K \) parts, then
\[
clr(x_1 \oplus x_2 \oplus \cdots \oplus x_N) = clr(x_1) + clr(x_2) + \cdots + clr(x_N).
\]

Proof. According to Aitchison geometry, addition of vectors in the simplex is defined as,
\[
x_1 \oplus \cdots \oplus x_N = \left(\prod_{n=1}^{N} x_{n1}, \ldots, \prod_{n=1}^{N} x_{nK}\right)
\]
where \( c_N = \sum_{k=1}^{K} \prod_{n=1}^{N} x_{nk} \).

By applying the centered-logratio transformation, we have
\[
clr(x_1 \oplus \cdots \oplus x_N) = \left(\log \frac{x_{1,n1}}{\delta_{K,N}}, \ldots, \log \frac{x_{1,nK}}{\delta_{K,N}}\right)
\]
where \( \delta_{K,N} = c_N \left(\prod_{k=1}^{K} \prod_{n=1}^{N} x_{nk}\right)^{1/K} = \left(\prod_{k=1}^{K} \prod_{n=1}^{N} x_{nk}\right)^{1/K} \).

Now, we can rewrite (48) as,
\[
clr(x_1 \oplus \cdots \oplus x_N) = \left(\log \frac{x_{1,n1}}{\prod_{k=1}^{K} x_{1k}}^{1/K}, \ldots, \log \frac{x_{1,nK}}{\prod_{k=1}^{K} x_{1k}}^{1/K}\right)
\]
\[
= \left(\sum_{n} \log \frac{x_{n1}}{\prod_{k=1}^{K} x_{nk}}^{1/K}, \ldots, \sum_{n} \log \frac{x_{nK}}{\prod_{k=1}^{K} x_{nk}}^{1/K}\right)
\]
\[
= clr(x_1) + \cdots + clr(x_N)
\]
(49)

Proposition 3. Given vectors \( x, y, u_x, u_y \in S^K \) where \( S^K \) is a simplex of \( K \) parts, then
\[
d_{S^K}^2 (x \oplus u_x, y \oplus u_y) - d_{S^K}^2 (x, y) \leq K \tau^2 - \frac{\Delta^2}{K}
\]
where \( \tau = \max_k \{\log u_{xk} - \log u_{yk}\} \) and \( \Delta = \sum_k (\log u_{xk} - \log u_{yk}) \).

Proof. According to Aitchison geometry, the distance between two vectors \( x, y \in S^K \) is defined as,
\[
d_{S^K}^2 (x, y) = \|clr(x) - clr(y)\|^2
\]
Then,
\[
d_{S^K}^2 (x \oplus u_x, y \oplus u_y) = \|clr(x \oplus u_x) - clr(y \oplus u_y)\|^2
\]
According to Lemma (1),

\[ d_{2S}^2 (x \oplus u_x, y \oplus u_y) = \| (clr(x) - clr(y)) + (clr(u_x) - clr(u_y)) \|_2^2 \]

\[ = (|clr(x) - clr(y)|^2 + |clr(u_x) - clr(u_y)|^2 + (clr(x) - clr(y))^T (clr(u_x) - clr(u_y)) + (clr(u_x) - clr(u_y))^T (clr(x) - clr(y))) \]

\[ = d_{2S}^2 (x, y) + d_{2S}^2 (u_x, u_y) + 2 \sum_{k=1}^{K} \left( \log \frac{x_k}{g(x)} - \log \frac{y_k}{g(y)} \right)^2 \left( \log \frac{u_{x_k}}{g(u_x)} - \log \frac{u_{y_k}}{g(u_y)} \right) \]

For simplicity, let’s define \( d_2^2 := d_{2S}^2 (x, y) \) and \( d_1^2 := d_{2S}^2 (x \oplus u_x, y \oplus u_y) \), then

\[
d_2^2 - d_1^2 = d_{2S}^2 (u_x, u_y) + 2 \sum_{k=1}^{K} \left( \log \frac{x_k}{g(x)} - \log \frac{y_k}{g(y)} \right)^2 \left( \log \frac{u_{x_k}}{g(u_x)} - \log \frac{u_{y_k}}{g(u_y)} \right)
\]

\[
= d_{2S}^2 (u_x, u_y) + 2 \sum_{k=1}^{K} \left( \log \frac{x_k}{g(x)} \right) \left( \log \frac{u_{x_k}}{g(u_x)} - \log \frac{g(u_x)}{g(u_y)} \right) - 2 \sum_{k=1}^{K} \left( \log \frac{y_k}{g(y)} \right) \left( \log \frac{u_{y_k}}{g(u_y)} - \log \frac{g(u_y)}{g(u_x)} \right)
\]

Moreover, we know that \( \log \frac{u_{x_k}}{u_{y_k}} \leq \tau \) and \( \log \frac{g(u_x)}{g(u_y)} = \log \left( \prod_{k=1}^{K} \frac{u_{x_k}}{u_{y_k}} \right)^{1/K} = \frac{1}{K} \sum_{k=1}^{K} \log \frac{u_{x_k}}{u_{y_k}} = \frac{\Delta}{K} \), then

\[
d_2^2 - d_1^2 \leq d_{2S}^2 (u_x, u_y) + \frac{2}{K} \left( \sum_{k=1}^{K} \left( \log \frac{x_k}{g(x)} \right) - \sum_{k=1}^{K} \left( \log \frac{y_k}{g(y)} \right) \right) \leq d_{2S}^2 (u_x, u_y) + \frac{2}{K} \left( \sum_{k=1}^{K} \left( \log \frac{u_{x_k}}{g(u_x)} \right) - \sum_{k=1}^{K} \left( \log \frac{y_k}{g(y)} \right) \right)
\]

Since CLR is a zero-mean transformation, \( \sum_{k=1}^{K} \log \frac{x_k}{g(x)} = 0 \) and \( \sum_{k=1}^{K} \log \frac{y_k}{g(y)} = 0 \). Therefore,

\[
d_2^2 - d_1^2 \leq d_{2S}^2 (u_x, u_y)
\]

In addition, we have

\[
d_{2S}^2 (u_x, u_y) = \sum_{k=1}^{K} \left( \log \frac{u_{x_k}}{u_{y_k}} \right)^2 = \sum_{k=1}^{K} \left( \log \frac{u_{x_k}}{u_{y_k}} \right)^2 + \sum_{k=1}^{K} \left( \log \frac{u_{x_k}}{g(u_x)} \right)^2 - 2 \log \frac{g(u_x)}{g(u_y)} \sum_{k=1}^{K} \log \frac{u_{x_k}}{u_{y_k}}
\]

\[
\leq K \tau^2 - \frac{\Delta^2}{K}
\]

By inserting (54) in (53), we will have

\[
d_2^2 - d_1^2 \leq K \tau^2 - \frac{\Delta^2}{K}
\]

\[
\square
\]

**Proposition 4.** Given samples \( x, y \in S^K \), where \( S^K \) is a simplex of \( K \) parts, we have

\[ 0 \leq d_1^2 (x, y) - d_{2S}^2 (x \oplus u_x, y \oplus u_y) \leq \frac{1}{K} (\tau_1 + \tau_2)^2 \]

where \( d_1^2 (x, y) = \sum_k (\log x_k u_{x_k} - \log y_k u_{y_k})^2 \), \( \tau_1 = \max_k \{ \log u_{x_k} - \log u_{y_k} \} \), and \( \tau_2 = \max_k \{ \log x_k - \log y_k \} \).
where $D = \frac{1}{K} \sum_k (\log x_k u_{x_k} - \log y_k u_{y_k})$. Hence,

$$D^2 = \sum_{k=1}^K (\log x_k u_{x_k} - \log y_k u_{y_k})^2 + K D^2 - 2 D \sum_{k=1}^K (\log x_k u_{x_k} - \log y_k u_{y_k})$$

$$= d^2_a (x, y) = K D^2$$

Moreover we know that $\forall k$, $\log \frac{u_{x_k}}{u_{y_k}} \leq \tau_1$ and $\log \frac{x_k}{y_k} \leq \tau_2$, then

$$d^2_a (x, y) - D^2 \leq \frac{1}{K} \left( \sum_{k=1}^K \left( \log \frac{x_k}{y_k} + \log \frac{u_{x_k}}{u_{y_k}} \right) \right)^2$$

$$\leq \frac{1}{K} (\tau_1 + \tau_2)^2 \tag{58}$$

**Proposition 2.** Suppose $c_{a_n}, c_{b_n} \in S^K$, where $S^K$ is a simplex of $K$ parts and $n$ is the sample index. If $d (c_{a_n}, c_{b_n})$ denotes the Aitchison distance, then

$$d^2_a (c_{a_n}, c_{b_n}) = \frac{1}{K} \left( \sum_{k=1}^K \left( \log c_{a_{n_k}} - \log c_{b_{n_k}} \right) \right)^2$$

where $\tau_c = \max_k \{ \log c_{a_{n_k}} - \log c_{b_{n_k}} \}$, $\tau_d = \max_k \{ (\sigma_{a_k}^{-1} - 1) \log c_{a_{n_k}} - (\sigma_{b_k}^{-1} - 1) \log c_{b_{n_k}} \}$, and $\Delta_d = \sum_k (\sigma_{a_k}^{-1} - 1) \log c_{a_{n_k}} - (\sigma_{b_k}^{-1} - 1) \log c_{b_{n_k}}$.

**Proof.** In Propositions 3 and 4 by considering $x = c_{a_n}, y = c_{b_n}, u_x = u_y = u_n = \left( \frac{c_{a_{n1}}}{\gamma_1}, \ldots, \frac{c_{a_{nK}}}{\gamma_n} \right)$, and $u_y = u_b = \left( \frac{c_{b_{n1}}}{\gamma_{b1}}, \ldots, \frac{c_{b_{nK}}}{\gamma_{bK}} \right)$, where $\gamma_a = \sum_k (\sigma_{a_k}^{-1} - 1)$ and $\gamma_b = \sum_k (\sigma_{b_k}^{-1} - 1)$, we have

$$d^2_{S^K} (c_a + u_a, c_b + u_b) = \sum_{k=1}^K \left( \sigma_{a_k}^{-1} \log c_{a_{n_k}} - \sigma_{b_k}^{-1} \log c_{b_{n_k}} - D \right)^2 \tag{59}$$

where $D = \frac{1}{K} \sum_k (\sigma_{a_k}^{-1} \log c_{a_{n_k}} - \sigma_{b_k}^{-1} \log c_{b_{n_k}})$. Hence,

$$d^2_{S^K} (c_a + u_a, c_b + u_b) - d^2_{S^K} (c_a, c_b) \leq K \tau_d^2 - \frac{\Delta_d^2}{K} \tag{60}$$
We analyzed the effects of the dependency between the continuous and discrete latent factors on the results obtained by state-of-the-art methods for joint representation learning, e.g. JointVAE or CascadeVAE. These methods formulate the continuous and discrete variability as two independent factors such that the discrete factor is expected to determine the cluster to which each sample belongs, while the continuous factor represents the class-independent variability. In many applications, however, the assumption of a discrete-continuous dichotomy may not be satisfied. (Section F analyzes this assumption for the MNIST dataset.)

Calculation of angle and width: We first calculate the inertia matrix for each sample by treating the image as a solid object with a mass distribution given by pixel brightness values. Then, we compute the principal axis of the image based on the inertia matrix. We report the angle between this vector and the vertical axis using the computed angle value. We report the support of this projected signal, normalized by the horizontal size of the image (here 28 pixels).

\[
0 \leq d^2_{c_1} (c_a, c_b) - d^2_{c_2} (c_a \oplus u_a, c_b \oplus u_b) \leq \frac{1}{K} (\tau_e + \tau_o)^2
\]

Therefore,
\[
d^2_{c_1} (c_a, c_b) - d^2_{c_2} (c_a, c_b) \leq \frac{1}{K} \left( (\tau_e + \tau_o)^2 + K^2 \tau_o^2 - \Delta^2 \right)
\]

F MNIST dataset analysis

A common assumption in “disentangling” the continuous and discrete factors of variability is the independence of the categorical and continuous latent variables, conditioned on data. Figure S1 demonstrates that this assumption can be significantly violated for two commonly used, interpretable style variables, “angle” and “width,” in the MNIST dataset.

Calculation of angle and width: We first calculate the inertia matrix for each sample by treating the image as a solid object with a mass distribution given by pixel brightness values. Then, we compute the principal axis of the image based on the inertia matrix. We report the angle between this vector and the vertical axis using the computed angle value. We report the support of this projected signal, normalized by the horizontal size of the image (here 28 pixels).

G Dependence of state and class label in JointVAE

We analyzed the effects of the dependency between the continuous and discrete latent factors on the results obtained by state-of-the-art methods for joint representation learning, e.g. JointVAE or CascadeVAE. These methods formulate the continuous and discrete variability as two independent factors such that the discrete factor is expected to determine the cluster to which each sample belongs, while the continuous factor represents the class-independent variability. In many applications, however, the assumption of a discrete-continuous dichotomy may not be satisfied. (Section F analyzes this assumption for the MNIST dataset.)

Figure S2 illustrates four dimensions of the continuous latent variable $s$ obtained by the JointVAE model for the MNIST dataset. Here, colors represent the digit type of each $s$ sample. While the prior distribution is assumed to be Gaussian, the dependency of $s|x$ on the digit type, $c$, is visible. To quantify this observation, we applied an unsupervised clustering method, i.e. Gaussian mixture model (GMM) with 10 clusters, to the continuous RV samples obtained from a JointVAE network trained for 150000 iterations. This unsupervised model achieved an overall clustering accuracy of 84%. Figure S2 shows the results for individual digits, e.g. 83% for digit “1” (Figure S2). Together, these results demonstrate the violation of the independence assumption for $q(s|x)$ and $q(c|x)$. 

}\]
Figure S2: (a) 2-dimensional projections of the continuous variable obtained by JointVAE. Each dot represents a sample of the MNIST dataset and colors represent different digits. (b) Confusion matrix for MNIST digit clustering via GMM using only the continuous latent variable learned by JointVAE.

H  Architectures of the coupled networks

Figure S3 shows the network architecture for the 2-coupled mixVAE model applied on the MNIST and FACS datasets, respectively. For the MNIST dataset, each VAE agent received an augmented copy of the original image consisting of uniformly-distributed random alterations in four aspects:

1. rotation from $[-20, 20]$ degrees,
2. brightness from $[4, 6]$,
3. saturation from $[0.5, 1.5]$, and
4. hue saturation from $[-0.25, 0.25]$.

For the dSprites dataset, we have applied the following uniformly-distributed random alterations in three aspects:

1. rotation from $[-180, 180]$ degrees,
2. translation from $[0.1, 0.1]$, and
3. scale from $[0.8, 1.2]$.

To enhance the training process, we also applied 20% and 10% random dropout of the input image and the state variable, respectively. For the FACS dataset, we only used random 50% dropout of the gene expression values as data augmentation.

JointVAE uses the same network architecture as a single agent of cpl-mixVAE. That is, it still uses the same loss function and learning procedure as JointVAE, but its convolutional layers are replaced by fully-connected layers, to demonstrate that these architecture choices do not explain the improvement achieved by cpl-mixVAE.

H.1 Training parameters for the MNIST dataset

Training details used for the MNIST dataset are listed as follows. For JointVAE and JointVAE model, we used the same training parameters as reported in (Dupont, 2018).

cpl-mixVAE

- Continuous and categorical variational factors: $s \in \mathbb{R}^{10}$, $|c| = 10$
- Batch size: 256
- Training epochs: 600
- Temperature for sampling from Gumbel-softmax distribution: 0.67
- Coupling weight, $\lambda$: 1
- Optimizer: Adam with learning rate 1e-4

JointVAE, JointVAE model

- Continuous and categorical variational factors: $s \in \mathbb{R}^{10}$, $|c| = 10$
- Batch size: 64
- Training epochs: 160
- Temperature for sampling from Gumbel-softmax distribution: 0.67
- $\gamma_s, \gamma_c$: 30
- $C_s, C_c$: Increased linearly from 0 to 5 in 25000 iterations
- Optimizer: Adam with learning rate 1e-4

(a) Benchmark datasets including MNIST and dSprites. The dimension of the input and first hidden layers depend on the image resolution i.e., $D$.

(b) FACS dataset

Figure S3: cpl-mixVAE architectures including 2 agents.
H.2 Training parameters for the dSprites dataset

Training details used for the dSprites dataset are listed as follows.

cpl-mixVAE
- Continuous and categorical variational factors: \( s \in \mathbb{R}^6 \), \( |c| = 3 \)
- Batch size: 256
- Training epochs: 600
- Temperature for sampling from Gumbel-softmax distribution: 0.67
- Coupling weight, \( \lambda \): 10
- Optimizer: Adam with learning rate 1e-4

H.3 Training parameters for the FACS dataset

Training details used for the FACS dataset are listed as follows. For the JointVAE\(^\dagger\) model, we tried to set the parameters according to the reported numbers in (Dupont, 2018).

cpl-mixVAE
- Continuous and categorical variational factors: \( s \in \mathbb{R}^2 \), \( |c| = 115 \)
- Batch size: 1000
- size of the last hidden layer, D: 10
- Training epochs: 10000
- Temperature for sampling from Gumbel-softmax distribution: 1
- Temperature for softmax function on \( q(c|x) \): 0.005 (\( \approx 1/|z| \))
- Coupling weight, \( \lambda \): 1
- Optimizer: Adam with learning rate 1e-3

JointVAE\(^\dagger\)
- Continuous and categorical variational factors: \( s \in \mathbb{R}^2 \), \( |c| = 115 \)
- Batch size: 1000
- size of the last hidden layer, D: 10
- Training epochs: 10000
- Temperature for sampling from Gumbel-softmax distribution: 0.005
- \( \gamma_s, \gamma_c \): 100
- \( C_s, C_c \): Increased linearly from 0 to 10 in 100000 iterations
- Optimizer: Adam with learning rate 1e-3

I Supplementary discussion: type-preserving augmentations

The proposed cpl-mixVAE framework relies on data augmentation to generate multiple noisy copies of a given data point, \( x \). Data augmentation is a known technique used in machine learning to improve the generalizability and stability of the learning process, particularly in the classification problem\(^1\). In our framework, however, data augmentation is required to enhance the categorical assignment by exploring within-class diversity across agents and encouraging them to utilize the entire latent space. In the Experiments section, we showed that this framework significantly outperforms JointVAE and CascadeVAE, which use a single VAE agent. While augmentation strategies are already known for many applications such as benchmark image datasets, achieving optimal type-preserving data augmentations is not always trivial and requires prior information about the underlying factors of invariance. Although standard data augmentations, such as rotation, scaling, or translation can provide type-preserving augmentations, these affine operations often fail to capture the richness of the underlying process.

\(^1\)Krizhevsky A., Sutskever I., Hinton G. E. Imagenet classification with deep convolutional neural networks. In Advances in neural information processing systems, 2012.
Moreover, they may fail to generate samples belonging to the same class depending on the parameters of those transformations, which suggests a need to decide on threshold values for those parameters. There are alternative methods that address some shortcomings of standard augmentation, such as (i) transformation learning on a per-class basis through pairwise image alignment within a class; (ii) DAGANs that are able to generate a much broader set of augmentations using adversarial networks; (iii) unsupervised adversarial invariance induction using competitive learning between reconstruction and prediction tasks. Combining our framework with an augmentation method like DAGAN can be a promising future direction.

### J Hierarchical Clustering of the FACS dataset

Figure S4 illustrates the cell type taxonomy proposed by Tasic et al., 2018 for the FACS dataset, which includes 115 inhibitory and excitatory neurons in the anterior lateral motor cortex and primary visual cortex in mice. While the original dataset consists of 133 cell types, including non-neuronal cells, we analyze only the neurons in this manuscript. The dendrogram shows the hierarchical relationship between neuron types, where the first bifurcation from the top represents the split between inhibitory (right branch) and excitatory neurons (left branch). Figure S4b shows the distribution of samples per cell type demonstrating the pronounced imbalance in the dataset, with several types having less than 100 samples.

---

2. Hauberg S., et al. Dreaming more data: Class-dependent distributions over diffeomorphisms for learned data augmentation. In Artificial Intelligence and Statistics, 2016

3. Antoniou A., Storkey A., Edwards, H. Data augmentation generative adversarial networks. arXiv preprint arXiv:1711.04340, 2017.

4. Jaiswal A., et al. Unsupervised adversarial invariance. In Advances in Neural Information Processing Systems, 2018.
We use the hierarchical dendrogram in Figure S4a to assess the performance of the proposed method at different levels of the cell type taxonomy. For this purpose, we obtain a smaller number of cell classes by progressively merging the 115 cell types according to the hierarchical dendrogram. For instance, at the three merging levels from top to bottom indicated in Figure S4a, we obtain 23, 57, and 88 distinct merged cell classes, respectively.

To evaluate the categories learned by cpl-mixVAE with respect to the fewer cell classes resulting from such merging, we pursue two approaches. In the first approach, we match the 115 categories with the cell types of (Tasic et al., 2018) using the Hungarian algorithm and then merge the categories by moving upward in the hierarchy. In the second approach, we train cpl-mixVAE models with a reduced number of categories that matches the number of the merged cell classes. In the latter approach, the resulting categories do not incorporate any hierarchical information. Figure S5 compares classification performance with these two approaches. Based on the confusion matrices in Figure S5 using merged categories (b, d, f) leads to more prominent diagonals and darker off-diagonal entries, suggesting superior performance compared to using categories that do not incorporate any hierarchical information (c, e, g). However, Table S3 shows that while using the hierarchical merging of categories performs better with more number of cell classes (K), the advantage dissipates as the number of cell classes decreases.

Table S3: Classification accuracy of cpl-mixVAE as a function of the number of categories for the FACS dataset. The reported numbers for non-merged categories are the mean values over three model initialization.

| K   | Merged Categories | Not Merged Categories |
|-----|-------------------|-----------------------|
| 88  | 34.7              | 27.0                  |
| 57  | 47.5              | 37.3                  |
| 23  | 73.3              | 65.6                  |
| 10  | 83.4              | 92.6                  |
| 2   | 94.3              | 99.9                  |
Figure S5: The confusion matrices obtained from categorical factors learned by the cpl-mixVAE model with different numbers of clusters, for the FACS dataset. (a) Clustering performance for 115 cell types in the original dendrogram of Figure S4. The left panels show the clustering performance when the categorical variables in (a) are merged according to the hierarchy in Figure S4 at three different levels, producing (b) 88, (d) 57, and (f) 23 neuron classes. The right panels show the performance for (c) 88, (e) 57, and (g) 23 types when independent cpl-mixVAE models are trained with K categories.