Using deep generative models for simultaneous representational and predictive modeling of brain and behavior: A graded supervised-to-unsupervised modeling framework.

Kieran McVeigh[a,1]

Ashutosh Singh[a,2]

Deniz Erdogmus[2]

Lisa Feldman Barrett[b,1,3,4]

Ajay B. Satpute[b,1]

[1]Department of Psychology, Northeastern University, 360 Huntington Ave, 125 NI, Boston, MA, 02130

[2]Department of Electrical and Computer Engineering, Northeastern University, 360 Huntington Ave, Boston, MA, 02130

[3]Athinoula A. Martinos Center for Biomedical Imaging, Massachusetts General Hospital, Charlestown, MA

[4]Psychiatric Neuroimaging Division, Department of Psychiatry, Massachusetts General Hospital, Charlestown, MA

[a,b] Denotes equal contributions

Correspondence to:
Kieran McVeigh, mcveigh.k@northeastern.edu
Ajay Satpute, a.satpute@northeastern.edu

Keywords: deep learning, brain mapping, supervised, unsupervised, brain state

Abstract

Research in cognitive neuroscience has increasingly used machine learning algorithms and multivariate pattern analysis (MVPA) to model brain-behavior relationships. These algorithms typically fall into two main types: unsupervised or supervised. In cognitive neuroscience, most studies assume that brain states map to behavior in a linear, one-to-one fashion. If such mapping exists, unsupervised and supervised approaches should lead to converging conclusions. Conclusions diverge, however, when the mapping is more complex. Unfortunately, the ground truth of brain-behavior relationships is rarely, if ever, known in advance, leading to the possibility of incorrect conclusions when scientists use a modeling approach guided by a single set of assumptions that may not be justified. In this paper, we introduce a possible solution to this dilemma. We combine unsupervised and discriminative supervised models with a model comparison strategy, and apply this approach on simulated data where the ground truth brain-behavior relationships are known. The combined modeling approach learns a latent space that models the distribution of simulated brain states (similar to unsupervised approaches). At the same time, the latent space contains information that can be used to predict behavior, thereby contributing to brain-behavior characterizations (similar to supervised approaches). We use four simulated datasets that vary in the linearity and homogeneity of the brain-behavior relationship to compare the modeling results for each simulated dataset along a continuum from fully unsupervised to fully supervised. More importantly, we examine what happens to the latent space as a consequence of incorrect assumptions that applied to a dataset during modeling. We further show how our framework can model diverse brain-behavior relationships in a way that fully unsupervised v. fully supervised approaches do not, by comparing the modeling results for each simulated dataset along a continuum from fully unsupervised to fully supervised.

Introduction

Research in cognitive neuroscience has increasingly used machine learning algorithms and multivariate pattern analysis (MVPA) to model brain-behavior relationships (Haxby, 2012; Norman et al., 2006; Peelen & Downing, 2023; Pereira et al., 2009). These algorithms typically fall into two main types: unsupervised or supervised (Hastie et al., 2001). Unsupervised methods estimate the distributions of brain states derived from measurements during an experiment but without reference to behavior (Fukushima, 1980; Goutte et al., 1999; Han et al., 2019; Huang et al., 2018). Supervised methods learn how brain states predict behavior (e.g. condition labels; Haynes & Rees, 2006; Mitchell et al., 2004; O'Toole et al., 2007; Poldrack et al., 2009; Tong & Pratte, 2012). Discriminative supervised methods, which are the most commonly used supervised methods in cognitive neuroscience, do not estimate the distribution of brain states (generative supervised approaches do, but are limited because they have stringent assumptions about data distributions; Bishop, 2006). In cognitive neuroscience, most studies assume that brain states map to behavior in a linear, one-to-one fashion (Barrett & Satpute, 2013; Poldrack, 2008; Rugg & Thompson-Schill, 2013; Uttal, 2003). If such mapping exists, unsupervised and supervised approaches should lead to converging conclusions (see **Figure 1A**; Azari et al., 2020). Conclusions diverge, however, when the mapping is more complex (**Figure 1B-D**; e.g., Westlin et al., 2023). It could be, for example, that the brain-behavior mapping is not one-to-one, that the brain-behavior mapping is not linearly separable, or that the dimensions of highest variance in the brain data do not relate to behavior. Unfortunately, the ground truth of brain-behavior relationships is rarely, if ever, known in advance, leading to the possibility of incorrect conclusions when scientists use a modeling approach guided by a single set of assumptions that may not be justified (e.g., using a linear discriminative supervised method on brain data that maps to behavior in a nonlinear and/or many-to-one, brain states-to-behavior fashion, see Azari et al., 2020).

In this paper, we introduce a possible solution to this dilemma. We combine unsupervised and discriminative supervised models with a model comparison strategy, and apply this approach on simulated data where the ground truth brain-behavior relationships are known. The combined modeling approach learns a latent space that models the distribution of simulated brain states (similar to unsupervised approaches). At the same time, the latent space contains information that can be used to predict behavior, thereby contributing to brain-behavior characterizations (similar to supervised approaches). We use four simulated datasets that vary in the linearity and homogeneity of the brain-behavior relationship to compare the modeling results for each simulated dataset along a continuum from fully unsupervised to fully supervised. More importantly, we examine what happens to the latent space as a consequence of incorrect assumptions that applied to a dataset during modeling. We further show how our framework can model diverse brain-behavior relationships in a way that fully unsupervised v. fully supervised approaches do not, by comparing the modeling results for each simulated dataset along a continuum from fully unsupervised to fully supervised.

Supervised Approaches

Supervised algorithms are explicitly designed to estimate a predictive function from brain data to behavioral labels (Bishop, 2006). They have been successfully used to predict labels across multiple psychological domains including memory, social cognition, emotion, language,

etc., from whole-brain signal patterns (Pereira et al., 2009; Poldrack et al., 2009; Tong & Pratte, 2012; Weaverdyck et al., 2020). Supervised methods offer two key advantages for making inferences about brain-behavior mappings. First, measurements of brain activity include many sources of variance including noise and supervised approaches use only the fraction of that variance that predicts the behavioral labels, given specific modeling assumptions. Second, brain-behavior relationships may be complex (e.g., nonlinear, many-to-one mappings; **Figure 1C-D**) and supervised approaches can nonetheless learn to predict behavior even in the presence of such complexities (e.g., by using neural network models as a "universal function approximators"; Hornik et al., 1989).

These strengths are directly related to the key weakness of the supervised approach. A complete understanding of brain-behavior relations *requires* modeling the distribution of brain states. Popular supervised algorithms do not explicitly represent the brain data, however. In the absence of such modeling, it is possible to predict behavioral labels with the same solution in datasets that have different brain-behavior relations, obscuring those differences (Clark-Polner et al., 2017; Lindquist et al., 2022). For example, consider **Figure 1A** vs. **Figure 1B**. Both datasets could be modeled with the same discriminative algorithm and distinguish between two classes (open circles above the dark circles), even though the two datasets involve different brain-behavior relations. Discriminative algorithms such as support vector machines or LASSO-PCR model a decision-boundary that is used to distinguish one class from another to predict behavioral labels, called a hyperplane. These hyperplanes are then analyzed to understand brain-behavior relations. Since these hyperplanes fail to model brain states, it is quite possible for these algorithms to estimate the same hyperplane across fundamentally different brain-behavior relations (Kragel et al., 2018; Wang et al., 2024).

As long as the behavioral predictions are successful, it could be argued that understanding the characteristics of the underlying brain behavior relationships (e.g. the distribution of brain states) is unnecessary. This argument comes with some fine print: Because many relationships can be described by the same solution, *inferences* about ground-truth relationships are unjustified. The risk of ill-advised inferences is particularly pronounced in cognitive neuroscience. Researchers often fit a single linear model and, if it performs significantly above chance, interpret it as accurately capturing the brain-behavior relationship. Yet this model assumes—rather than tests—key aspects of the relationship, such as its linearity, possibly obscuring its true nature. Thus, while supervised approaches can be used to develop predictive models, they offer limited insight into the brain-behavior relations that are the main theoretical focus of cognitive neuroscience.

Unsupervised Approaches

The learning objective of unsupervised algorithms is to model structured variance (i.e., reliable patterns) in the data irrespective of any labels (Hastie et al., 2001). Researchers have used unsupervised models to take an "inside out" approach to brain-behavior mapping wherein models of structured variance in brain data are estimated first, and then the model's estimates are related to behavioral data such as stimulus features, behavioral responses, etc. (Azari et al., 2020; Buzsáki, 2019; Sennesh et al., 2020; for discussion, see Westlin et al., 2023). Unsupervised approaches offer several strengths over supervised algorithms. First, they explicitly model brain states. That is, they model brain data as distributions (when using

clustering algorithms) or as points in a lower dimensional space (when using other dimensionality reduction algorithms). Second, unsupervised approaches allow for more flexible, data-driven conclusions to emerge. Observed clusters or dimensions are not inherently constrained by experimenter-defined condition labels, participant categories, or any other theory-laden assumptions of how brain and behavior "should" relate (e.g., the solution is not constrained by assuming there is a one to one mapping between brain and behavior; Khan et al., 2022; Westlin et al., 2023). The solution is still constrained by theory-laden aspects of study design and sampling, but those issues are better discussed elsewhere (Barrett, 2022; Lee et al., 2021; Dubova & Goldstone, 2023). Here, we focus on analytical choices. For example, unsupervised algorithms could reveal that multiple brain states occur during the trials of a given, experimenter-defined task condition (illustrated in **Figure 1B-D**, Azari et al., 2020; Doyle et al., 2022; Khan et al., 2022; Nakuci et al., 2023), consistent with degeneracy or multiple realizability (Edelman & Gally, 2001; Price & Friston, 2002; Putnam, 1960).

These strengths of unsupervised approach approaches are directly related to a key weakness: They assume that the behavior of interest is related to the dominant dimension of variance in the brain data and therefore strong enough to drive model estimates, but that isn't always the case (for an illustrative case, see **Figure 1D**). Structured variance in brain data during a task may also include signals that are irrelevant to the behavior of interest (by contrast, supervised approaches only model variance related to behavioral labels). Furthermore, it can be difficult to determine whether the signals driving data-driven brain clusters (or dimensions) are in fact behaviorally-relevant (again, see Azari et al., 2020). Thus, while unsupervised approaches model the brain states themselves, the model estimates are not always useful for understanding the brain-behavior relationship.

Present Study

In this paper, we use recent advances in neural networks that are designed to simultaneously capture the distribution of brain states (the unsupervised learning objective) and model the relationships of brain states to class labels (the supervised learning objective; see **Figure 2A** for architecture diagram). There are a number of ways to implement these neural networks (e.g., Bzdok et al., 2015; Kingma et al., 2014; Zabihi et al., 2024). We implemented the unsupervised portion of the model using a variational auto-encoder (VAE), which is a neural network that learns a latent (lower-dimensional) representation of fMRI BOLD signal data, similar to principal component analysis (PCA). The VAE consisted of an encoder that compressed the BOLD data into a latent space and a decoder that reconstructed the data from the latent representation (**Figure 2A**). The supervised portion of the model used a classification network, also called a classification head, to predict the behavioral labels from the same latent representation (**Figure 2A**). All components (the encoder and decoder of the autoencoder, and the classification head) were trained jointly in an integrated "VAE-C" model (i.e., variational auto-encoder with a classification head). This joint training optimized the learned latent space to capture the underlying structure of the fMRI data and to enable classification of the data directly from the latent space (see **Figure 2A**). For an example of a VAE-C approach applied to EEG data see Bethge et al., (2022).

The VAE-C framework allowed us to examine the consequence of introducing increasing supervision during modeling, on a continuum from fully unsupervised to fully supervised (**Figure**

2**B**). In the fully unsupervised modeling condition, no classification was performed (see left most column of **Figure 2B**). The VAE-C ignored the labels when learning the latent representation of the BOLD data (i.e. there was no weight on the classification objective) and only used the BOLD data itself to estimate the latent space of brain states and reconstruct the data. Both the latent representation and the reconstructed data should recover the ground-truth structure in the simulated BOLD data. At the other end of the continuum in the fully supervised modeling condition (right most column, **Figure 2B**), the behavioral labels strongly shaped the latent representation of the BOLD data (i.e., there was no weight on the reconstruction objective). At various points along the continuum, we successively increased the influence of the labels to examine how the labels impacted the latent representation, the reconstructed data, and the resulting classification: 1 (reconstruction):1 (classification), 1:10, 1:100, and 1:500. (**see Figure 2B**).

In addition, the VAE-C framework contained four different modeling architectures that allowed us to vary whether a network could learn nonlinear relationships to examine the impact on the latent representation (via the encoder function), the reconstructed data (via the decoder function), and the resulting classification (via the classification function). We varied whether the encoder/decoder was restricted to learning a linear relation between the BOLD data and the latent space or whether an encoder/decoder could also learn nonlinear relations, as well as whether the classifier was restricted to learning a linear mapping between the estimated distribution of brain states and the behavioral labels or whether nonlinear mappings could also be estimated. These four architectures were examined for each point along the supervision continuum, resulting in a 2 (linearly or nonlinearly capable auto-encoder) x 2 (a linearly or nonlinearly capable classifier) x W (a weight ratio, W, for the learning objectives from fully unsupervised to fully supervised) framework (see **Figure 2B**).

We then applied this 2x2xW framework to four simulated data sets with different ground truth brain-behavior relationships to examine what happens when modeling assumptions do not respect the ground truth relationships in the modeled data (see Figure 1). We used four ground-truth simulations. Dataset 1 was simulated with a linearly separable, one-to-one brain-behavior mapping, in line with traditional assumptions in cognitive neuroscience (Figure 1A). Dataset 2 was simulated with a linearly separable, but many-to-one brain-behavior relationship (**Figure 1B**). Datasets 3 and 4 embodied the same ground truth relations (nonlinear separable many-to-one brain-behavior mappings) but dataset 3 contained cluster-like mappings whereas dataset 4 contained a continuous mapping (**Figure 1C-D**).

In all simulations, we started with a ground truth latent brain space that contained the distribution of brain states and their groupings (Dataset 1 through 3) or not (Dataset 4). We then generated “observed” BOLD data that was linearly related to that latent space. We also generated “observed” behavioral labels that were either linearly or nonlinearly related to that space. Using the “observed” (i.e., “simulated”) data, we then learned a latent brain space, and from this space reconstructed the BOLD data as well as computed classifications with the labels. This created a unique opportunity to observe the consequences of using different modeling assumptions (embedded in architectures 1 through 4) when brain-behavior mappings are complex (e.g., not one-to-one or nonlinear), and how these consequences might change as the labels had an increasing impact on the learned latent space.

The resulting analysis space was 4 (VAE-C architectures) by 6 weightings (along the supervision continuum from fully unsupervised to fully supervised) by 4 (ground-truth datasets) or 96 analyses. In each analysis, we measured the accuracy of the learned latent space by comparing it to the ground truth latent space for each simulated data set using representational similarity analysis (Kriegeskorte et al., 2008). We measured the accuracy of the reconstructed data by comparing it to the "observed" (simulated) BOLD data using the coefficient of determination (Ezekiel, 1930). We measured the accuracy of the classification by comparing the behavioral labels predicted by the VAE-C analysis to the "observed" (i.e., simulated) behavioral labels using an F1 score (Rijsbergen, 1979).

We predicted and found that, in general, the latent space would be modeled with reasonable accuracy for data that contained straightforward brain-behavior relations (in Dataset 1, the learned and ground-truth latent spaces would converge). We predicted and found that, in general, as the weighting of the classification objective (i.e., the behavioral labels) increased to fully supervised, complex brain-behavior relations (Datasets 2 through 4) would be modeled with distortion or bias (i.e., the learned and ground-truth latent spaces would diverge). As distortions increase, it's tempting to expect that the reconstructed BOLD data would correspondingly diverge from the "observed" (simulated) BOLD data, but to some extent, this is not what we observed. Instead, as we show, the data reconstructions were largely robust to violated modeling assumptions because more than one latent space produced the same reconstructed data. Also, all functions (encoding, decoding, and classification) proceeded simultaneously and were jointly optimized, such that the decoding function accommodated the distorted space when reconstructing the BOLD data. We predicted and found that all models classified well only when their linearity assumptions matched the ground truth in the simulated data, with one exception as we explain below. Taken together, these analyses make clear that neither supervised or unsupervised modeling alone is sufficient and modeling assumptions should always be tested rather than overlooked.

Methods

Modeling Framework

The VAE-C framework learns latent representations which contain information about both the original fMRI signal and the behavioral class associated with the observation. Below we formally describe the more general modeling framework and how our VAE-C architecture implements this framework. We begin by defining our set of variables. Let $D = \{X, Y\}_n$ be the input dataset where $X \in R^d$ which represents the matrix of fmri data and $Y \in R$ represents the behavior label for *n* observations. Let $z \in R$d be the random variable that describes the latent distribution. We assume that the latent distribution of the generative model is Gaussian $\{0, I\}$. We specified the learning objective to be,

$$max[I(Z;Y) + I(Z;\hat{X}) - I(X;Z)] \quad [1]$$

This equation describes a constrained optimization that makes it possible for us to learn a latent representation Z such that it contains information about both the class labels Y and the neural

data X. More formally, maximizing the mutual information, *I*, between Z and Y ensures the latent representation Z contains information about the class labels. Maximizing the mutual information between Z and $\hat{X}$ (i.e. the reconstruction of X from Z, wherein the optimal case $\hat{X}$ = X), optimizes the latent representation to be able to reconstruct the data accurately ensuring Z contains information about the X (the fMRI data). If we only included these first two terms in our model, the model is at greater risk of overfitting the data and generating an uninterpretable latent space. Hence, the final term minimizes mutual information between Z and X, which regularizes the learned Z, taking inspiration from the information bottleneck theory (Tishby et al. 2000).

VAE Implementation

To implement this broader modeling framework, we used a variational autoencoder with a classification head neural network (VAE-C). The variational autoencoder portion of this network consists of an encoder, and a decoder, implements a generative model of the brain data (X) accomplishing the unsupervised learning objective. While the classifier network from the VAE's latent space (See **Figure 2A**) accomplishes the supervised learning objective. We use the following notation for describing the parameters of the networks $[\theta, \phi, \gamma]$ respectively. Theta is the set of parameters for the encoder, phi is the set of parameters for the decoder, and gamma is the set of parameters for the classifier. We use a VAE derived loss for learning the parameters for $[\theta, \phi]$ which consist of reconstruction loss and a KL (Kulback-leibler) divergence term. Simultaneously we also minimize the cross entropy loss which serves as the proxy for maximizing $I(Z;Y)$. The objective in [1] can be expressed as the following loss to learn the model parameter set $[\theta, \phi, \gamma]$,

$$L_{VAE} = min_{\theta,\phi}||\hat{x} - D_{\phi}[E_{\theta}(x)]||_2 + \beta\, KL[N(\mu_x, \sigma_x), N(0, I)],$$

$$L_{clf} = min_{\gamma} E_{Py}[-log_{PY}(y)]$$

$$Lvae - c = Lvae + w\, Lclf$$

Network Architecture

To implement our framework, we first specified four VAE-C architectures that varied in the linearity or nonlinearity of the autoencoder and classifier components (**Figure 2B**), that is: (i) linear-autoencoder and linear-classifier, (ii) linear-autoencoder and nonlinear-classifier, (iii) nonlinear-autoencoder and linear-classifier, (iv) nonlinear-autoencoder and nonlinear-classifier. The linear-autoencoders consisted of two fully connected linear layers, one transforming the input (X) to the latent representation (Z), and the other transforming the latent representation (Z) to the reconstruction of the input (X_hat). This architecture is only capable of learning linear encoding and decoding functions. The nonlinear autoencoder adds an additional fully connected hidden layer (size 25), with a tanh activation function, to both the decoder and encoder. The linear-classifier consisted of a single, fully connected linear layer between the latent space (Z)

and the class labels (Y). This architecture is only capable of learning a linear function to map from the latent space to class labels. For the nonlinear classifiers, we used two hidden layers between the latent space and the class labels with ReLU activation functions. Additional details and justification for the hidden layer sizes and architecture are located in the hyperparameters section.

Weighting Learning Objectives

The above equations formally describe how our framework integrates reconstruction and classification objectives. This integration enables a key feature of our architecture - a classification weight - to examine how the modeling framework performs when emphasizing one or the other objective. At one end of the continuum, a classification weight of zero changes the network to be fully unsupervised, which allows us to examine the distribution of the learned Z when only fMRI measurements are taken into account. At the other end of the continuum, a reconstruction weight of zero changes the network to be fully supervised, which reveals the brain representations the network learns when functioning as a supervised classifier.

Of particular theoretical interest we can examine how the latent space of the network changes as we gradually adjust the weights of the network to move from a fully unsupervised analysis to a fully supervised analysis. While in relatively simple scenarios we expect the networks learned representations to be consistent across a variety of weights, of greater interest is how in more complex brain behavior mapping (i.e. many-to-one and nonlinear) is understanding how a networks architecture and weights on learning objective interact to shape the learned latent space.

Simulated Data

We aimed to test our model in four different data simulations that span hypothetical mappings as shown in Figure 1. Dataset 1 was simulated using the conventional assumption of one (brain state) to one (behavior) mapping. Instances (i.e, trials) from behavioral class “A” were sampled from one brain state distribution, whereas those from behavioral class “B” were sampled from another brain state distribution (**Figure 1A**). Latent brain states, which are represented as points in a 2-dimensional space, were sampled from two distinct Gaussian distributions. For each of these gaussians we sampled 500 data points. We then generated behavioral class labels and fMRI data from each point. Behavioral labels were generated by assigning all data points from each gaussian to a given class label (i.e., “A” or “B”). fMRI data from these latent data points was generated by projecting observations from the 2d latent space to the 3d fMRI space (represented as 100 parcels, or regions of interest) using a randomly generated linear function.

To generate the other simulated datasets, this same general procedure was modified as follows. Dataset 2 was simulated using the assumption of a many (brain state) to one (behavior)linearly-separable mapping. Two different brain state distributions were used to generate instances for class “A”, and two different brain state distributions were used to generate instances for class “B” (**Figure 1B**). Latent brain states were sampled from four distinct Gaussian distributions. The simulation generated fMRI and behavioral data that are consistent with a linearly separable, but many-to-one, brain-behavior relationship. Dataset 3 was simulated

using the assumption of a many-to-one nonlinearly-separable, brain-behavior mapping with a cluster-like structure (**Figure 1C**). Data was sampled from four gaussians distributions, however the fMRI and behavioral data were simulated such that there was a nonlinearly separable many-to-one, brain-behavior relationship. Dataset 4 was another version of the many-to-one nonlinearly-separable, brain-behavior mapping but with a continuous or dimensional structure (**Figure 1D**). We generated a single distribution of brain states and assigned the 50% of the instances with the greatest similarity to the mean to behavioral class A, and all other instances to behavior class B. For additional details on sampling from the data generating distribution for each simulation, see supplemental materials. Here it's worth noting that since we did not vary assumptions about the latent space is projected to fMRI observations (i.e. we used the same linear function to project the latent simulated data to the fMRI space) we expect all models to be able to reconstruct the fMRI data nearly perfect at baseline.

Network Training and Evaluation

All networks were trained with the Adam optimization technique starting with an initial learning rate of .0001. Networks were trained for 1500 epochs. Data was split 70/30 into train and test. To account for the auto-encoder style infrastructure of the networks, where the smallest dimensional layer of the network occurred in the middle of the network, the first 50 epochs of all networks were trained with only the reconstruction and kl divergence objective functions, after which networks were trained with the reported weights on the classification and reconstruction objectives.

For each network, we evaluated reconstruction performance using the coefficient of determination (R^2), the classification performance using the F1 score, and the representational similarity analysis (RSA) to compare the structure of the learned latent space with the ground truth. We used the F1 score over accuracy as the F1 score balances precision (the proportion of true positive predictions among all positive predictions) and recall (the proportion of true positive predictions among all actual positives). We used RSA because of its common application in cognitive neuroscience and the low dimensionality of our latent spaces also allowed for qualitative comparisons via visual inspections. When calculating the similarity matrices for our RSA analyses we used the euclidean distance as our distance metric and compared the similarity of the matrices with the Pearson's correlation coefficient. Network performance metrics were calculated at the end of the 1500 epochs on the left out testing data.

Results

Overview of Analyses

A summary of our analyses is presented in **Table 1**. The correspondence or "match" of the ground truth brain-behavior relationships in simulated data of and the assumptions built into the architectures in the framework is shown in each cell in the table. As seen in **table 1**, the main mismatches between the ground-truth and VAE-C architectures occur in terms of whether the classification head assumes linear or nonlinear relationships between the brain states and the labels. This distinction corresponds to the type of brain-behavior mapping simulated (linearly or nonlinearly separable). The linearity or nonlinearity of the auto-encoder is of less importance to

the current analyses as all relations between the simulated latent and "observed" fMRI data are linear.

Dataset 1: Linearly Separable, One-to-One Mapping

Dataset 1 was simulated to represent linearly separable Gaussian distributions, such that each behavioral label mapped to one and only one grouping of brain states. This is the brain-behavior mappings most commonly assumed in the majority of analytical approaches used in cognitive neuroscience. The linear encoder/linear classifier architecture contained assumptions that best matched the ground truth represented in Dataset 1 (**Table 1,** column 1). As expected, the learned and ground truth latent state estimates generally converged (i.e. brain states associated with each behavioral label were represented as single grouping in the latent spaces). As the weighting of the labels increased along the supervision continuum, the learned latent space was distorted along the axis perpendicular to the classification boundary (see **Figure 3**), as reflected in decreased RSA scores for the more heavily supervised models (**Table 2**). This distortion did not affect the accuracy of the data reconstruction or classification, as indexed by the R2 and the F1 values, which remained high across the supervision continuum (R^2s ~ 1.00, F1-Scores ~.95 **Table 2**). This same pattern of results was present for all VAE-C architectures (**Table 2**), as expected, because none of the other VAE-C architectures contained assumptions that violated the linearly separable, one-to-one mapping in Dataset 1.

Dataset 2: Linearly Separable, Many-to-One Mapping

Dataset 2 was simulated to represent linearly separable Gaussian distributions, like Dataset 1, except that each behavioral label mapped to more than one distribution of brain states. Similar to Dataset 1, the linear encoder/linear classifier architecture contained assumptions that best matched the ground truth represented in Dataset 2, although none of the other VAE-C architectures contained assumptions that violated that ground truth (**Figure 1**, column 2).

As expected, the fully unsupervised model learned a latent space that converged with the ground truth space (Pearson's r = .94). The learned latent state became increasingly distorted along the axis perpendicular to the classification boundary as the semi-supervised modeling moved towards fully supervised, as observed for Dataset 1 (**see Figure 4**). The fully supervised model produced the most distortion (Pearson's r = .26); instead of modeling two distinct groupings for each behavioral label, the fully supervised variant of the the linear encoder/linear classifier architecture produced one grouping for each behavioral label, essentially producing a linearly separable, one-to-one mapping (see **Figure 4** last column). Despite these distortions, data reconstruction and classification remained accurate (**Table 2**). This same pattern of results was present for all VAE-C architectures (**Table 2**), as expected.

Dataset 3: Nonlinearly Separable, Many-to-One Mapping (Discrete)

The linear encoding/nonlinear classifier architecture best matched the ground truth represented in Dataset 3 (**Figure 2**, column 3). Dataset 3 was simulated to represent a many-to-one mapping, such that each behavioral label mapped to more than one distribution of brain states, except that the mapping was nonlinear. As expected, the fully unsupervised model

learned a latent space that converged with the ground truth space (Pearson's r = .98). The semi-supervised models performed similarly. The fully supervised model produced the most distortion (Pearson's r = .41) in the learned latent space; instead of modeling two distinct distributions for each behavioral label, the fully supervised variant produced one distribution for each behavioral label, essentially creating a linearly separable, one-to-one mapping from the latent space, instead of a nonlinearly separable, many to one mapping (see **figure 5** last column). The classification accuracy for the fully supervised model was similarly low (F1 = .66). Data reconstruction accuracy was not computed for the fully supervised model (for which data reconstruction was irrelevant), but was high for all other variants of this architecture (**Table 4**). All semi-supervised variants classified well (F1 = .95), with the 1:1 variant being somewhat less accurate (F1 = .75).

The same pattern of results was generally observed for the architecture that was next best in terms of matching assumptions for Dataset 3: the nonlinear encoding/nonlinear classifier architecture (see Figure 2, column 3). The classifier assumption matched the ground-truth relation between the ground-truth latent space and the "observed" behavioral labels, whereas the nonlinear encoder assumption was capable (but not optimized) for learning the linear encoding relationship between ground-truth latent space and the "observed" (i.e., simulated) BOLD data (**Table 4**). The one exception was the classification accuracy for the fully supervised model, which was high (F1 = .93).

The two remaining VAE-C architectures (with linear classifiers) explicitly violated the ground-truth, nonlinear relations between the latent space and the behavioral labels in Dataset 3. The fully unsupervised versions of these two architectures produced accurate learned latent spaces. The fully supervised versions of these two architectures produced distortions in the learned latent space (**Table 4**), but this distortion was unlikely the result of assumption violations because they also emerged in the prior two architectures (which did not explicitly violate any assumptions). The remaining pattern of results for the semi-supervised version of these architectures was complex.

The semi-supervised variants of VAE-C architecture with a linear encoder and a linear classifier learned accurate latent spaces and accurately reconstructed the BOLD data, but their classification accuracy was reduced (F1 =.60). The semi-supervised version of the nonlinear encoder combined with a linear classifier learned accurate latent spaces at lower degrees of supervision, but their classification accuracy was reduced. At higher weightings, they produced slight inaccuracies in the learned latent space but, paradoxically, produced high classification accuracy **Table 4**). This effect was a result of this specific architecture (nonlinear encoder/linear classifier). To achieve high classification this architecture requires a latent space where the classes are linearly separable. To achieve this linear separability in dataset 3 (nonlinearly separable many to one mapping) required the nonlinear encoder to learn a distorted latent space (see **figure 5** third row). Despite this distortion to the latent space the reconstruction performance was essentially unaffected.

Dataset 4: Nonlinearly Separable, Many-to-One Mapping (Continuous)

The results for Dataset 4 largely mirrored those for Dataset 3. Increasing weight on the classification objective increasingly shaped the learned latent space, that space becomes distorted but with little decrement for classification accuracy (and, as we observed in Dataset 3,

classification accuracy can even improve, see **Table 5, Figure 6**). Distortions in the latent space are associated with slightly less accurate data reconstructions (on the order of a few percentages of variance explained see **Table 5**).

Discussion

In cognitive neuroscience, it is important to model both the distribution of brain states and how these brain states relate with behavior. Conventional unsupervised and supervised analytical approaches tend to focus only on one or the other objective, which offer an incomplete account of brain behavior relations and may lead to conclusions that are not supported by the data. Our VAE-C framework addresses both objectives simultaneously. In doing so, the framework offers opportunities to test fundamental assumptions about brain-behavior relationships that are often overlooked. Indeed, the mapping between brain states and behavior is rarely, if ever, known at the outset of an experiment. Despite this uncertainty, many current approaches assume that brain-behavior relationships are one-to-one and linearly separable without testing if these assumptions are in fact supported. Indeed, a growing body of evidence suggests that brain-behavior relationships are complex and may consist of nonlinear, many-to-one mappings (Doyle et al., 2022; Marder & Taylor, 2011; Mizusaki & O'Donnell, 2021; Price & Friston, 2002; Sajid et al., 2023; Wang et al., 2024). To address these complexities, our framework combines two key components: (1) a VAE-C neural network architecture that simultaneously models the distribution of brain states and their mappings to behavior and (2) a model comparison strategy that varies the degree of linearity in architecture and the amount of weight on the supervised objective akin to a multiverse analysis (Steegen et al., 2016) . This framework allows scientists to discover whether more complex relations might be present and to adjust their modeling approaches and theoretical conclusions accordingly.

We fit this framework to four simulated ground-truth datasets to show what happens when data was modelled with assumptions that are inconsistent with the ground-truth. Our results showed that for datasets in which there were many-to-one mappings, fully supervised models failed to capture these mappings in the latent space despite classifying well. This demonstrates how high classification accuracy alone does not imply a good model of the brain-behavior relationship. Our results also showed that high degrees of supervision often introduced distortions in the latent space, even when the models continued to reconstruct and classify well, suggesting that multiple different latent space structures can encode approximately equivalent brain-behavior relationships. These findings demonstrate how the latent representations learned by the model are shaped by the researchers' choices about learning objectives, the relative weight of the different objectives, and model architectures.

When ground-truth relations contained nonlinear brain-behavior mappings, our results (unsurprisingly) showed that nonlinear models were necessary to accurately capture the brain-behavior relationship. Importantly, while linear models achieved predictive accuracy above chance, they were far less accurate than their nonlinear counterparts. This finding highlights the value of our framework's model comparison across architectures. These results underscore the risk of using predictive performance above chance as the sole criterion for model evaluation. Researchers should test rather than assume the form of brain-behavior mappings.

Our framework offers ways to gain insights into these questions. If, upon applying the 2x2xW framework to a dataset, there is little change in reconstruction, accuracy, and RSA metrics across architectures and the supervision continuum, a reasonable assumption is that the data follows a linearly separable brain-behavior relationship. On the other hand if architectures with nonlinearities have improved classification accuracies this suggests a nonlinear brain-behavior relation.

Our framework also illustrates that inferences about one-to-one versus many-to-one mappings are more challenging to make. Many-to-one brain-behavior mappings exhibit greater distortions in the latent space as models transition from lightly to fully supervised. Additionally, many-to-one mappings can be explored using clustering methods applied to the model's latent space. In this context, the semi-supervised VAE-C models provide an advantage over fully unsupervised methods. Because each location in the latent space of a trained VAE-C model is associated with a class label, researchers can condition the latent space based on these learned label associations. This conditioning facilitates the identification of many-to-one relationships, as demonstrated in dataset 4, where brain states are clustered into a single group regardless of class labels.

Moreover, VAE-C models offer a generative framework that reconstructs both BOLD data and class labels, enabling researchers to investigate the extent of many-to-one mappings in terms of BOLD correlates, not just in the latent space. For example, if the samples in the latent space contain both positive and negative values for a latent dimension, its BOLD correlates likely reflect diverse measurements (e.g., increased and decreased BOLD signals in particular regions). This heterogeneity can be assessed by sampling from the model's latent space and reconstructing the data in brain space. Overall the VAE-C framework offers researchers a new tool to discover and model more complex brain-behavior relationships.

Comparisons to other methods and limitations

Combining supervised and unsupervised objectives, our framework models representations interpretable in terms of both fMRI data and behavioral classes. However, because our model architecture is designed for these objectives and optimized jointly, it does not guarantee optimal performance on either objective individually. A fully supervised, purely discriminative model might classify behavior more accurately, albeit it is also possible that it may not since it would not have the advantage of using learned latent dimensions as features that could be more informative for classification (as offered by the VAE-C). Various unsupervised methods (such as clustering) might fit a researcher's purposes for a specific set of analyses, but again may be dominated by dimensions of variance in the fMRI data that are not necessarily relevant to behavior (Azari et al., 2020). Ultimately, neither approach alone would capture the joint distribution of neural and behavioral data as our framework does.

The VAE-C models used in our framework can also be thought of as a form of self-supervised representation learning, an increasingly popular technique in machine learning. The VAE-Cs' unsupervised reconstruction ensures that the model learns a well-structured distribution of brain states, creating a latent space with strong interpretability in terms of BOLD measurements, this is a specific choice we made in the framework. There are many other methods for self-supervised representation learning, such as contrastive learning (Hadsell et al., 2006; Schneider et al., 2023). These other methods provide more flexibility in the latent space,

allowing potentially novel brain representations that are not as tightly constrained by data reconstruction. This flexibility might yield novel insights, yet it comes with the risk of reduced interpretability in terms of neural correlates. For the purpose of understanding brain-behavior mappings, we believe the VAE-C framework strikes a valuable balance, leveraging the flexibility of neural networks while maintaining a strong anchoring in both neural and behavioral measurements.

Conclusion

The VAE-C framework provides a novel and versatile tool for testing assumptions about brain-behavior relationships. This framework empowers researchers to move beyond assumptions of linearity and one-to-one mappings that dominate much of cognitive neuroscience. By integrating supervised and unsupervised objectives, our approach offers insights that traditional methods obscure. Ultimately, frameworks like VAE-C can help progress in uncovering the diversity and richness of brain-behavior relations, bridging gaps between theoretical and empirical understanding.

References

Azari, B., Westlin, C., Satpute, A. B., Hutchinson, J. B., Kragel, P. A., Hoemann, K., Khan, Z., Wormwood, J. B., Quigley, K. S., Erdogmus, D., Dy, J., Brooks, D. H., & Barrett, L. F. (2020). Comparing supervised and unsupervised approaches to emotion categorization in the human brain, body, and subjective experience. *Scientific Reports*, *10*(1), 20284. https://doi.org/10.1038/s41598-020-77117-8

Barrett, L. F. (2022). Context reconsidered: Complex signal ensembles, relational meaning, and population thinking in psychological science. *The American Psychologist*, *77*(8), 894–920. https://doi.org/10.1037/amp0001054

Barrett, L. F., & Satpute, A. B. (2013). Large-scale brain networks in affective and social neuroscience: Towards an integrative functional architecture of the brain. *Current Opinion in Neurobiology*, *23*(3), 361–372. https://doi.org/10.1016/j.conb.2012.12.012

Bethge, D., Hallgarten, P., Grosse-Puppendahl, T., Kari, M., Chuang, L. L., Özdenizci, O., & Schmidt, A. (2022). EEG2Vec: Learning Affective EEG Representations via Variational Autoencoders. *2022 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, 3150–3157. https://doi.org/10.1109/SMC53654.2022.9945517

Bishop, C. M. (2006). *Pattern recognition and machine learning*. Springer.

Buzsáki, G. (2019). *The Brain from Inside Out*. Oxford University Press.

Bzdok, D., Eickenberg, M., Grisel, O., Thirion, B., & Varoquaux, G. (2015). Semi-Supervised Factored Logistic Regression for High-Dimensional Neuroimaging Data. *Advances in Neural Information Processing Systems*, *28*. https://proceedings.neurips.cc/paper/2015/hash/06a15eb1c3836723b53e4abca8d9b879-Abstract.html

Clark-Polner, E., Johnson, T. D., & Barrett, L. F. (2017). Multivoxel Pattern Analysis Does Not Provide Evidence to Support the Existence of Basic Emotions. *Cerebral Cortex*, *27*(3),

1944–1948. https://doi.org/10.1093/cercor/bhw028

Doyle, C. M., Lane, S. T., Brooks, J. A., Wilkins, R. W., Gates, K. M., & Lindquist, K. A. (2022). Unsupervised classification reveals consistency and degeneracy in neural network patterns of emotion. *Social Cognitive and Affective Neuroscience*, *17*(11), 995–1006. https://doi.org/10.1093/scan/nsac028

Dubova, M., & Goldstone, R. L. (2023). Carving joints into nature: Reengineering scientific concepts in light of concept-laden evidence. *Trends in Cognitive Sciences*, *27*(7), 656–670. https://doi.org/10.1016/j.tics.2023.04.006

Edelman, G. M., & Gally, J. A. (2001). Degeneracy and complexity in biological systems. *Proceedings of the National Academy of Sciences*, *98*(24), 13763–13768. https://doi.org/10.1073/pnas.231499798

Ezekiel, M. (1930). *Methods of correlation analysis* (pp. xiv, 427). Wiley.

Finn, E. S., Glerean, E., Khojandi, A. Y., Nielson, D., Molfese, P. J., Handwerker, D. A., & Bandettini, P. A. (2020). Idiosynchrony: From shared responses to individual differences during naturalistic neuroimaging. *NeuroImage*, *215*, 116828. https://doi.org/10.1016/j.neuroimage.2020.116828

Fukushima, K. (1980). Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. *Biological Cybernetics*, *36*(4), 193–202. https://doi.org/10.1007/BF00344251

Goutte, C., Toft, P., Rostrup, E., Nielsen, F. Å., & Hansen, L. K. (1999). On Clustering fMRI Time Series. *NeuroImage*, *9*(3), 298–310. https://doi.org/10.1006/nimg.1998.0391

Hadsell, R., Chopra, S., & LeCun, Y. (2006). Dimensionality Reduction by Learning an Invariant Mapping. *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, *2*, 1735–1742. https://doi.org/10.1109/CVPR.2006.100

Han, K., Wen, H., Shi, J., Lu, K.-H., Zhang, Y., Fu, D., & Liu, Z. (2019). Variational autoencoder: An unsupervised model for encoding and decoding fMRI activity in visual cortex.

NeuroImage, *198*, 125–136. https://doi.org/10.1016/j.neuroimage.2019.05.039

Hastie, T., Friedman, J., & Tibshirani, R. (2001). Unsupervised Learning. In T. Hastie, J. Friedman, & R. Tibshirani (Eds.), *The Elements of Statistical Learning: Data Mining, Inference, and Prediction* (pp. 437–508). Springer. https://doi.org/10.1007/978-0-387-21606-5_14

Haxby, J. V. (2012). Multivariate pattern analysis of fMRI: The early beginnings. *NeuroImage*, *62*(2), 852–855. https://doi.org/10.1016/j.neuroimage.2012.03.016

Haynes, J.-D., & Rees, G. (2006). Decoding mental states from brain activity in humans. *Nature Reviews Neuroscience*, *7*(7), 523–534. https://doi.org/10.1038/nrn1931

Hornik, K., Stinchcombe, M., & White, H. (1989). Multilayer feedforward networks are universal approximators. *Neural Networks*, *2*(5), 359–366. https://doi.org/10.1016/0893-6080(89)90020-8

Huang, H., Hu, X., Zhao, Y., Makkie, M., Dong, Q., Zhao, S., Guo, L., & Liu, T. (2018). Modeling Task fMRI Data Via Deep Convolutional Autoencoder. *IEEE Transactions on Medical Imaging*, *37*(7), 1551–1561. IEEE Transactions on Medical Imaging. https://doi.org/10.1109/TMI.2017.2715285

Khan, Z., Wang, Y., Sennesh, E., Dy, J., Ostadabbas, S., van de Meent, J.-W., Hutchinson, J. B., & Satpute, A. B. (2022). A computational neural model for mapping degenerate neural architectures. *Neuroinformatics*, 1–15.

Kingma, D. P., Rezende, D. J., Mohamed, S., & Welling, M. (2014). *Semi-Supervised Learning with Deep Generative Models* (arXiv:1406.5298). arXiv. https://doi.org/10.48550/arXiv.1406.5298

Kragel, P. A., Koban, L., Barrett, L. F., & Wager, T. D. (2018). Representation, Pattern Information, and Brain Signatures: From Neurons to Neuroimaging. *Neuron*, *99*(2), 257–273. https://doi.org/10.1016/j.neuron.2018.06.009

Kriegeskorte, N., Mur, M., & Bandettini, P. A. (2008). Representational similarity

analysis—Connecting the branches of systems neuroscience. *Frontiers in Systems Neuroscience*, *2*. https://doi.org/10.3389/neuro.06.004.2008

Lee, K. M., Ferreira-Santos, F., & Satpute, A. B. (2021). Predictive processing models and affective neuroscience. *Neuroscience & Biobehavioral Reviews*, *131*, 211–228. https://doi.org/10.1016/j.neubiorev.2021.09.009

Lindquist, K. A., Jackson, J. C., Leshin, J., Satpute, A. B., & Gendron, M. (2022). The cultural evolution of emotion. *Nature Reviews Psychology*, *1*(11), Article 11. https://doi.org/10.1038/s44159-022-00105-4

Marder, E., & Taylor, A. L. (2011). Multiple models to capture the variability in biological neurons and networks. *Nature Neuroscience*, *14*(2), 133–138. https://doi.org/10.1038/nn.2735

Mitchell, T. M., Hutchinson, R., Niculescu, R. S., Pereira, F., Wang, X., Just, M., & Newman, S. (2004). Learning to Decode Cognitive States from Brain Images. *Machine Learning*, *57*(1), 145–175. https://doi.org/10.1023/B:MACH.0000035475.85309.1b

Mizusaki, B. E. P., & O'Donnell, C. (2021). Neural circuit function redundancy in brain disorders. *Current Opinion in Neurobiology*, *70*, 74–80. https://doi.org/10.1016/j.conb.2021.07.008

Nakuci, J., Yeon, J., Kim, J.-H., Kim, S.-P., & Rahnev, D. (2023). Multiple brain activation patterns for the same task. *bioRxiv*. https://www.ncbi.nlm.nih.gov/pmc/articles/PMC10104176/

Norman, K. A., Polyn, S. M., Detre, G. J., & Haxby, J. V. (2006). Beyond mind-reading: Multi-voxel pattern analysis of fMRI data. *Trends in Cognitive Sciences*, *10*(9), 424–430. https://doi.org/10.1016/j.tics.2006.07.005

O'Toole, A. J., Jiang, F., Abdi, H., Pénard, N., Dunlop, J. P., & Parent, M. A. (2007). Theoretical, Statistical, and Practical Perspectives on Pattern-based Classification Approaches to the Analysis of Functional Neuroimaging Data. *Journal of Cognitive Neuroscience*, *19*(11), 1735–1752. https://doi.org/10.1162/jocn.2007.19.11.1735

Peelen, M. V., & Downing, P. E. (2023). Testing cognitive theories with multivariate pattern

analysis of neuroimaging data. *Nature Human Behaviour*, *7*(9), 1430–1441. https://doi.org/10.1038/s41562-023-01680-z

Pereira, F., Mitchell, T., & Botvinick, M. (2009). Machine learning classifiers and fMRI: A tutorial overview. *NeuroImage*, *45*(1, Supplement 1), S199–S209. https://doi.org/10.1016/j.neuroimage.2008.11.007

Poldrack, R. A. (2008). The role of fMRI in Cognitive Neuroscience: Where do we stand? *Current Opinion in Neurobiology*, *18*(2), 223–227. https://doi.org/10.1016/j.conb.2008.07.006

Poldrack, R. A., Halchenko, Y. O., & Hanson, S. J. (2009). Decoding the Large-Scale Structure of Brain Function by Classifying Mental States Across Individuals. *Psychological Science*, *20*(11), 1364–1372. https://doi.org/10.1111/j.1467-9280.2009.02460.x

Price, C. J., & Friston, K. J. (2002). Degeneracy and cognitive anatomy. *Trends in Cognitive Sciences*, *6*(10), 416–421.

Putnam, H. (1960). Minds and Machines. In S. Hook (Ed.), *Dimensions Of Mind: A Symposium.* (pp. 138–164). NEW YORK University Press.

Rijsbergen, C. J. V. (1979). *Information Retrieval* (2nd ed.). Butterworth-Heinemann.

Rugg, M. D., & Thompson-Schill, S. L. (2013). Moving Forward With fMRI Data. *Perspectives on Psychological Science*, *8*(1), 84–87. https://doi.org/10.1177/1745691612469030

Sajid, N., Gajardo-Vidal, A., Ekert, J. O., Lorca-Puls, D. L., Hope, T. M. H., Green, D. W., Friston, K. J., & Price, C. J. (2023). Degeneracy in the neurological model of auditory speech repetition. *Communications Biology*, *6*(1), 1–10. https://doi.org/10.1038/s42003-023-05515-5

Schneider, S., Lee, J. H., & Mathis, M. W. (2023). Learnable latent embeddings for joint behavioural and neural analysis. *Nature*, 1–9. https://doi.org/10.1038/s41586-023-06031-6

Sennesh, E., Khan, Z., Wang, Y., Hutchinson, J. B., Satpute, A., Dy, J., & van de Meent, J.-W.

(2020). Neural Topographic Factor Analysis for fMRI Data. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, & H. Lin (Eds.), *Advances in Neural Information Processing Systems* (Vol. 33, pp. 12046–12056). Curran Associates, Inc. https://proceedings.neurips.cc/paper/2020/file/8c3c27ac7d298331a1bdfd0a5e8703d3-Paper.pdf

Steegen, S., Tuerlinckx, F., Gelman, A., & Vanpaemel, W. (2016). Increasing Transparency Through a Multiverse Analysis. *Perspectives on Psychological Science*, *11*(5), 702–712. https://doi.org/10.1177/1745691616658637

Tong, F., & Pratte, M. S. (2012). Decoding Patterns of Human Brain Activity. *Annual Review of Psychology*, *63*(Volume 63, 2012), 483–509. https://doi.org/10.1146/annurev-psych-120710-100412

Uttal, W. (2003). *The New Phrenology*. MIT Press.

Wang, Y., Kragel, P., & Satpute, A. B. (2024). Neural predictors of fear depend on the situation. *Journal of Neuroscience*. https://doi.org/10.1523/JNEUROSCI.0142-23.2024

Weaverdyck, M. E., Lieberman, M. D., & Parkinson, C. (2020). Tools of the Trade Multivoxel pattern analysis in fMRI: A practical introduction for social and affective neuroscientists. *Social Cognitive and Affective Neuroscience*, *15*(4), 487–509. https://doi.org/10.1093/scan/nsaa057

Westlin, C., Theriault, J. E., Katsumi, Y., Nieto-Castanon, A., Kucyi, A., Ruf, S. F., Brown, S. M., Pavel, M., Erdogmus, D., Brooks, D. H., Quigley, K. S., Whitfield-Gabrieli, S., & Barrett, L. F. (2023). Improving the study of brain-behavior relationships by revisiting basic assumptions. *Trends in Cognitive Sciences*, *27*(3), 246–257. https://doi.org/10.1016/j.tics.2022.12.015

Zabihi, M., Kia, S. M., Wolfers, T., Boer, S. de, Fraza, C., Dinga, R., Arenas, A. L., Bzdok, D., Beckmann, C. F., & Marquand, A. (2024). Nonlinear latent representations of high-dimensional task-fMRI data: Unveiling cognitive and behavioral insights in

heterogeneous spatial maps. *PLOS ONE*, *19*(8), e0308329.

https://doi.org/10.1371/journal.pone.0308329

Figures and Tables

Table 1

		Brain - Behavior Relationship			
		Linear, One to One	Linear, Many to One	Nonlinear, Many to One (Cluster-like)	Nonlinear, Many to One (Continuous)
AutoEncoder Architecture	Classifier Architecture				
Linear	Linear	match	match	violation	violation
Linear	Nonlinear	allowable	allowable	match	match
Nonlinear	Linear	allowable	allowable	violation	violation
Nonlinear	Nonlinear	allowable	allowable	allowable	allowable

Modeling Assumptions and Simulated Brain-Behavior Relationships. Modeling architectures (along rows) had either linearly or nonlinearly capable encoders and/or classifiers. The brain-behavior relationships in the simulated datasets (along columns) were linearly or nonlinearly separable and also had one-to-one or different versions of many-to-one mappings. Model architectures can “match” (green highlighted cells) or “violate” (red highlighted cells) the data generative process of the ground truth. “Allowable” (yellow highlighted cells) architectures are flexible enough to model the data even if they are not a direct match in terms of the linearity of the encoder and/or the classifier to the data generative process. Whether an architecture is a “match” for a brain-behavior relationship depends on two factors. The first factor is the assumptions of the classifier. Linearly capable classifiers match linear, but violate nonlinear, brain-behavior relationships whereas nonlinearly capable classifiers match nonlinear relationships. The second factor is the assumptions of the encoder. For simplicity, we simulated the “observed” BOLD data using a linear encoding function from the latent space to the brain space. This was helpful for limiting our study to four ground-truth datasets and to maintain our focus on complex brain-*behavior* relationships. But as a consequence, architectures with a nonlinear encoder - regardless of whether they are paired with linear or nonlinear classifiers - can only be allowable (or a violation) and not a match. Intriguingly, the model architectures do not have straightforward matches or mismatches for one-to-one or many-to-one brain behavior mappings. That aspect of brain-behavior mappings is informed in our VAE-C by examining how the latent space is shaped with greater weight on supervision.

Table 2

Dataset 1. One-to-one, linearly separable brain-behavior relationship.

Classifier Weight	Fully Unsupervised	1	10	100	500	Fully Supervised
Linear Autoencoder, Linear Classifier						
R Squared	1.00	1.00	1.00	1.00	1.00	
F1-Score		0.95	0.95	0.95	0.96	0.93
Accuracy		0.95	0.95	0.95	0.96	0.93
RSA	0.98	0.98	0.99	0.99	0.97	0.55
kl-divergence	0.06	0.06	0.06	0.07	0.10	0.01
Linear Autoencoder, Nonlinear Classifier						
R Squared	1.00	1.00	1.00	1.00	1.00	
F1-Score		0.96	0.96	0.95	0.96	0.89
Accuracy		0.96	0.96	0.95	0.96	0.89
RSA	0.98	0.98	0.98	0.99	0.96	0.57
kl-divergence	0.06	0.06	0.06	0.06	0.10	0.01
Nonlinear Autoencoder, Linear Classifier						
R Squared	1.00	1.00	1.00	1.00	0.99	
F1-Score		0.95	0.95	0.96	0.96	0.94
Accuracy		0.96	0.95	0.96	0.96	0.94
RSA	0.97	0.98	0.99	0.93	0.88	0.49
kl-divergence	0.04	0.04	0.04	0.05	0.09	0.01
Nonlinear Autoencoder, Nonlinear Classifier						
R Squared	1.00	1.00	1.00	1.00	0.99	
F1-Score		0.00	0.95	0.95	0.95	0.92
Accuracy		0.50	0.95	0.95	0.96	0.92
RSA	0.96	0.97	0.98	0.98	0.91	0.47
kl-divergence	0.04	0.04	0.04	0.04	0.06	0

Columns of this table are the location on the unsupervised to the supervised continuum models were trained at. This starts with full unsupervised in the left columns and increases to fully supervised in the rightmost columns, in between the ratio of the unsupervised to supervised weight is listed. Rows are grouped by the VAE-C architecture a model was trained with. Rows within each architecture group report models performance in terms of reconstruction (R Squared), classification (F1-Score), and similarity of the learned and ground truth latent space (Representational Similarity Analysis; RSA). RSA scores are reported as the Pearson correlation of the ground truth and learned latent space for each model. All models perform well in terms of reconstruction, and classification. RSA is high for all unsupervised and semi-supervised models, and although is considerably lower for the fully supervised models, is still generally high.

Table 3

Dataset 2. Many-to-one, linearly separable brain-behavior relationship.

Classifier Weight	Fully Unsupervised	1	10	100	500	Fully Supervised
Linear Autoencoder, Linear Classifier						
R Squared	1.00	1.00	1.00	1.00	1.00	
F1-Score		0.98	0.99	0.99	0.98	0.94
Accuracy		0.98	0.99	0.99	0.98	0.94
RSA	0.94	0.94	0.93	0.83	0.77	0.26
kl-divergence	0.11	0.11	0.11	0.13	0.17	0.01
Linear Autoencoder, Nonlinear Classifier						
R Squared	1.00	1.00	1.00	1.00	1.00	
F1-Score		0.99	0.98	0.98	0.98	0.95
Accuracy		0.99	0.98	0.98	0.98	0.95
RSA	0.94	0.94	0.94	0.88	0.82	0.24
kl-divergence	0.12	0.11	0.11	0.13	0.16	0.01
Nonlinear Autoencoder, Linear Classifier						
R Squared	1.00	1.00	1.00	1.00	1.00	
F1-Score		0.99	0.98	0.98	0.99	0.96
Accuracy		0.99	0.99	0.99	0.99	0.96
RSA	0.95	0.94	0.90	0.67	0.57	0.20
kl-divergence	0.05	0.05	0.05	0.06	0.08	0.01
Nonlinear Autoencoder, Nonlinear Classifier						
R Squared	1.00	1.00	1.00	1.00	1.00	
F1-Score		0.98	0.98	0.98	0.99	0.96
Accuracy		0.99	0.98	0.99	0.99	0.96
RSA	0.96	0.95	0.95	0.83	0.70	0.22
kl-divergence	0.05	0.05	0.05	0.05	0.06	0.01

For dataset 2 all models reconstructed and classified the data accurately. However, as the weight on the supervised training objective increased to fully supervised accuracy in terms of the structure of the latent space showed large decrements. Suggesting in the fully supervised case models failed to capture the many to one brain-behavior mappings in dataset 2. Interestingly for many of the semi-supervised models further towards the supervised end of the continuum which showed some distortions to the latent space (RSA scores), the reconstruction was unaffected.

Table 4

Dataset 3. Many-to-one, nonlinearly separable brain-behavior relationship (clusters)

Classifier Weight	Fully Unsupervised	1	10	100	500	Fully Supervised
Linear Autoencoder, Linear Classifier						
R Squared	1.00	1.00	1.00	1.00	1.00	
F1-Score		0.58	0.59	0.59	0.59	0.66
Accuracy		0.59	0.59	0.59	0.59	0.68
RSA	0.98	0.98	0.98	0.98	0.99	0.36
kl-divergence	0.13	0.13	0.13	0.13	0.15	0.00
Linear Autoencoder, Nonlinear Classifier						
R Squared	1.00	1.00	1.00	1.00	1.00	
F1-Score		0.66	0.95	0.95	0.95	0.64
Accuracy		0.74	0.95	0.95	0.95	0.64
RSA	0.98	0.98	0.98	0.99	1.00	0.41
kl-divergence	0.13	0.13	0.13	0.16	0.31	0.00
Nonlinear Autoencoder, Linear Classifier						
R Squared	1.00	1.00	1.00	1.00	0.99	
F1-Score		0.61	0.63	0.71	0.96	0.93
Accuracy		0.60	0.61	0.70	0.97	0.93
RSA	0.97	0.97	0.97	0.96	0.93	0.29
kl-divergence	0.05	0.05	0.05	0.06	0.09	0.00
Nonlinear Autoencoder, Nonlinear Classifier						
R Squared	1.00	1.00	1.00	1.00	1.00	
F1-Score		0.95	0.95	0.95	0.96	0.93
Accuracy		0.95	0.96	0.96	0.96	0.94
RSA	0.98	0.98	0.99	0.99	0.99	0.22
kl-divergence	0.05	0.05	0.05	0.06	0.08	0.00

In dataset 3 all models continued to reconstruct well. Models with non-linear classifiers are generally classified well. All unsupervised and semi-supervised models also performed well in terms of Representational Similarity. Again as models became fully supervised, representational similarity greatly decreased. The nonlinear/linear model showed an interesting set of results whereas classification weight increased the F1-Score increased but representational similarity decreased.

Table 5

Dataset 4. Many-to-one, nonlinearly separable brain-behavior relationship (dimensional)

Classifier Weight	Fully Unsupervised	1	10	100	500	Fully Supervised
Linear Autoencoder, Linear Classifier						
R Squared	1.00	1.00	1.00	1.00	1.00	
F1-Score		0.69	0.70	0.70	0.70	0.48
Accuracy		0.60	0.61	0.61	0.61	0.47
RSA	0.99	0.99	0.99	1.00	0.94	-0.01
kl-divergence	0.05	0.05	0.06	0.07	0.11	0.00
Linear Autoencoder, Nonlinear Classifier						
R Squared	1.00	1.00	1.00	1.00	1.00	
F1-Score		0.96	0.96	0.97	0.97	0.75
Accuracy		0.96	0.96	0.97	0.97	0.74
RSA	0.99	0.99	0.99	0.97	1.00	0.48
kl-divergence	0.05	0.06	0.06	0.10	0.31	0.01
Nonlinear Autoencoder, Linear Classifier						
R Squared	1.00	1.00	1.00	0.99	0.95	
F1-Score		0.66	0.73	0.85	0.95	0.89
Accuracy		0.58	0.66	0.83	0.95	0.90
RSA	0.99	0.99	0.97	0.76	0.71	0.12
kl-divergence	0.04	0.04	0.04	0.05	0.05	0.01
Nonlinear Autoencoder, Nonlinear Classifier						
R Squared	1.00	1.00	1.00	0.99	0.97	
F1-Score		0.96	0.95	0.96	0.97	0.92
Accuracy		0.96	0.96	0.96	0.98	0.92
RSA	0.99	0.99	0.97	0.90	0.84	0.05
kl-divergence	0.04	0.04	0.04	0.05	0.06	0.01

In dataset 4 all models continued to reconstruct well. Models with non-linear classifiers are generally classified well. All unsupervised and semi-supervised models also performed well in terms of Representational Similarity. As models became fully supervised, representational similarity greatly decreased. The nonlinear/linear model showed an interesting set of results where as classification weight increased the F1-Score increased but representational similarity decreased. This was accompanied by a slight decrement in reconstruction accuracy (R-squared). The nonlinear/nonlinear model also showed a similar set of results where despite classifying well at all levels of supervision as the supervised weight increased the model showed slight decrements in representational similarity (RSA) and reconstruction performance (R-squared) suggesting distortions of the latent space.

Figure 1

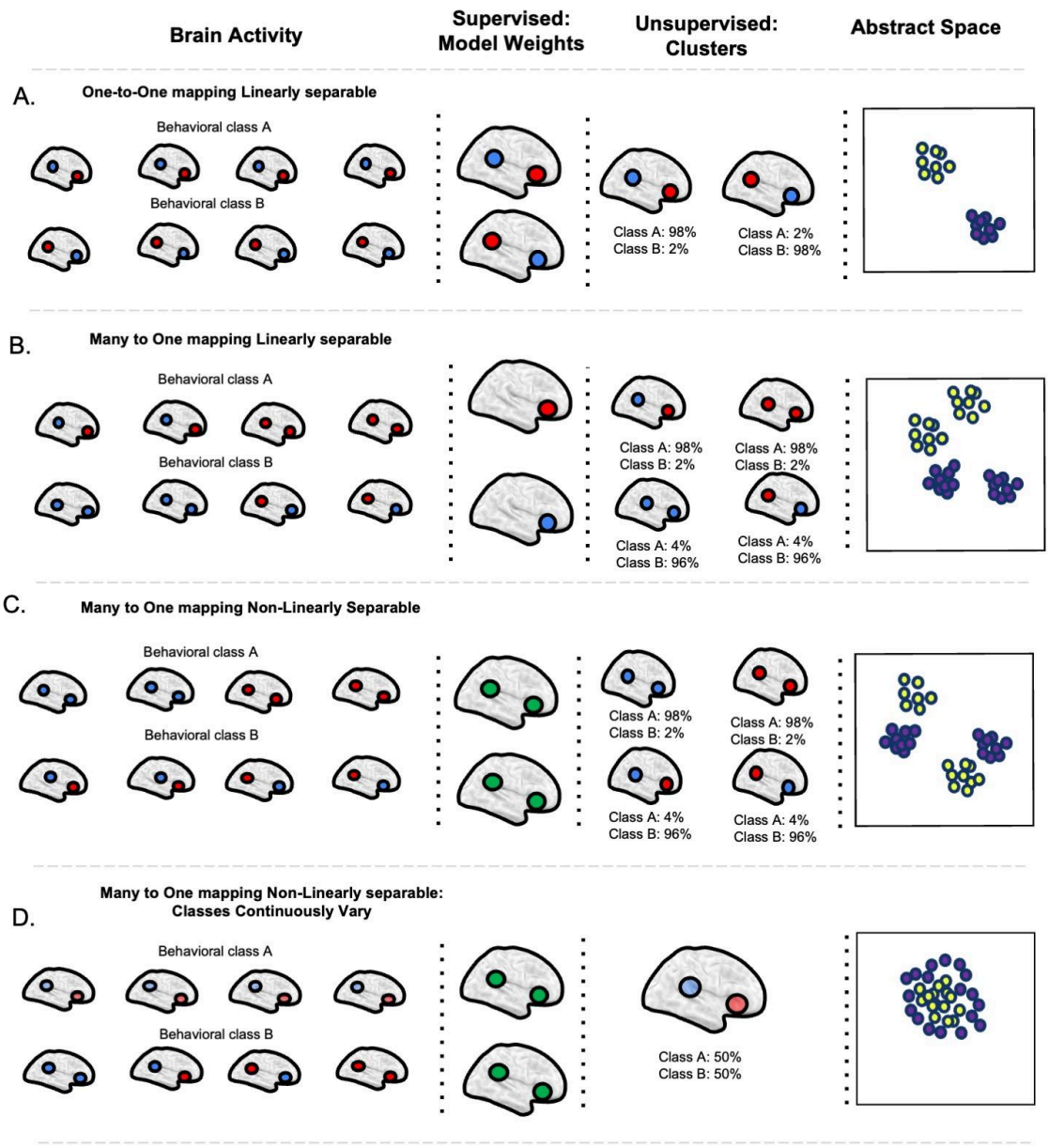


Figure 1. Four hypothetical brain-behavior mappings. This figure provides simplified cartoon examples of possible brain-behavior relationships, and how supervised and unsupervised clustering approaches might model these mappings.

Panel A shows a simple one-to-one brain-behavior mapping. The left most column depicts brain activity occurring during instances of behavioral class A (e.g. four trials in an fMRI study; top row) and behavior class B (bottom row). A one-to-one mapping assumes that the same pattern of brain activity occurs across instances. The second column represents the hypothetical feature weights of behavioral classes as modeled by a linear discriminative supervised algorithm. For instance the features weights of a logistic regression would show that increases in the anterior region and decreases in the posterior region in activity contribute to a prediction of behavioral class A (and vice versa for behavioral class B). **Cluster finding** These feature weights correspond with the underlying brain states, and the unsupervised clustering findings (shown in the third column). Finally the last column shows the same scenario in a two dimensional brain state space. **Of note under the restrictive assumption of a one-to-one mapping these approaches converge.**

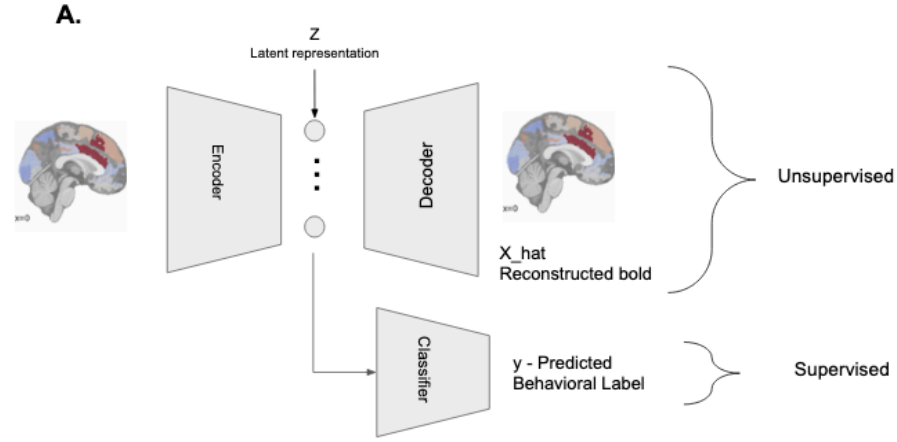
Panel B depicts a hypothetical many-to-one brain-behavior mapping, **wherein** there are multiple brain states associated with each

behavioral class, but **behavioral classes are still linearly separable**. A linear discriminative supervised algorithm will succeed in classifying behavioral classes from brain data, but it will fail in capturing the multiple brain states associated with the same class (i.e., it will only model regions contributing to classification). In contrast, an unsupervised approach may model the four brain states, albeit does not explicitly relate them to behavior. The final column illustrates this scenario in a brain state space, showing 4 clusters of data and linearly-separable, many-to-one mapping.

Panel C depicts a many-to-one nonlinearly separable brain-behavior mapping. Here, a linear supervised model would not classify well in this scenario, and so we assume a nonlinear supervised discriminative algorithm for our example. Although a nonlinear model will succeed in classification, they provide little information about the brain states underlying the behavioral classes. We can calculate average importance weights (indicated in green in the third column), which indicate which features are more or less important, but importance weights are not easily relatable to information about brain states (unlike regression weights which, even while not directly modeling brain states, at least provide indications of both sign and magnitude of importance of activity in a brain region). Unsupervised analyses do capture this many-to-one mapping (but again, do not explicitly relate brain states to behavior). The final column illustrates this scenario in a brain state space, showing 4 clusters of data, and a many-to-one nonlinearly separable mapping.

Panel D depicts another many-to-one, nonlinearly separable brain-behavior mapping (sometimes referred to as the “Anna Karenina” organization; Finn et al., 2020). Instances of behavioral class 1 have a similar pattern of activity whereas instances of class 2 are highly heterogenous. A nonlinear supervised model would successfully classify the behavioral classes, but again the importance weights do not accurately describe the brain states associated with each class. In fact, the average importance weights in this case are identical to those in Panel C even though the brain-behavior mapping is completely different, which further underscores the shortcomings of using feature weights from supervised models to gain insight on brain states. Critically, this scenario also reveals the shortcomings of a fully unsupervised approach. For example, a typical clustering solution would reveal a single cluster, failing to even separate the data into behaviorally relevant classes. The final column illustrates this scenario in a brain state space, showing the Anna Karenina-like organization of data.

Figure 2. Diagram of network architectures



B

Fully Unsupervised ⟷ Fully Supervised

	Reconstruction to Classification Weight Ratio					
Architecture	Unsupervised	1:1	1:10	1:100	1:500	Fully Supervised
Linear Encoder/ Linear Classifier	reconstruction (r2) latent space similarity (RSA)	reconstruction (r2) classification (f1) latent space similarity (RSA)	...	...	...	classification (f1) latent space similarity (RSA)
Linear Encoder/ Non-Linear Classifier	reconstruction (r2) latent space similarity (RSA)	reconstruction (r2) classification (f1) latent space similarity (RSA)	...	...	...	classification (f1) latent space similarity (RSA)
Non-Linear Encoder/ Linear Classifier	reconstruction (r2) latent space similarity (RSA)	reconstruction (r2) classification (f1) latent space similarity (RSA)	...	...	...	classification (f1) latent space similarity (RSA)
Non-Linear Encoder/ Non-Linear Classifier	reconstruction (r2) latent space similarity (RSA)	reconstruction (r2) classification (f1) latent space similarity (RSA)	...	...	...	classification (f1) latent space similarity (RSA)

A. shows a schematic of the proposed VAE-C architecture. During training and inference high dimensional fMRI data is input to encoder, and transformed into a lower dimensional latent representation Z. This latent representation Z is then fed through both decoder and the classifier. The decoder transforms Z into X_hat a reconstruction of the inputted bold data while the classifier transforms Z into a predicted class label y. The encoding to decoding reconstruction path through the network is an unsupervised analysis that motivates Z to model the structure in the inputted bold data. The encoding to classifier path through the network is a supervised analysis that motivates Z to contain information relevant to class labels. **B.** A visual representation of our planned analyses. Where for each dataset we trained each combination of VAE-C's with linear/nonlinear encoder and decoders, and linear/nonlinear classifiers - represented as the 2 x 2 table. Additionally we train each combinations of these network architectures at different locations of the supervision continuum, ranging from a fully unsupervised analysis to a fully supervised analysis.

Figure 3.

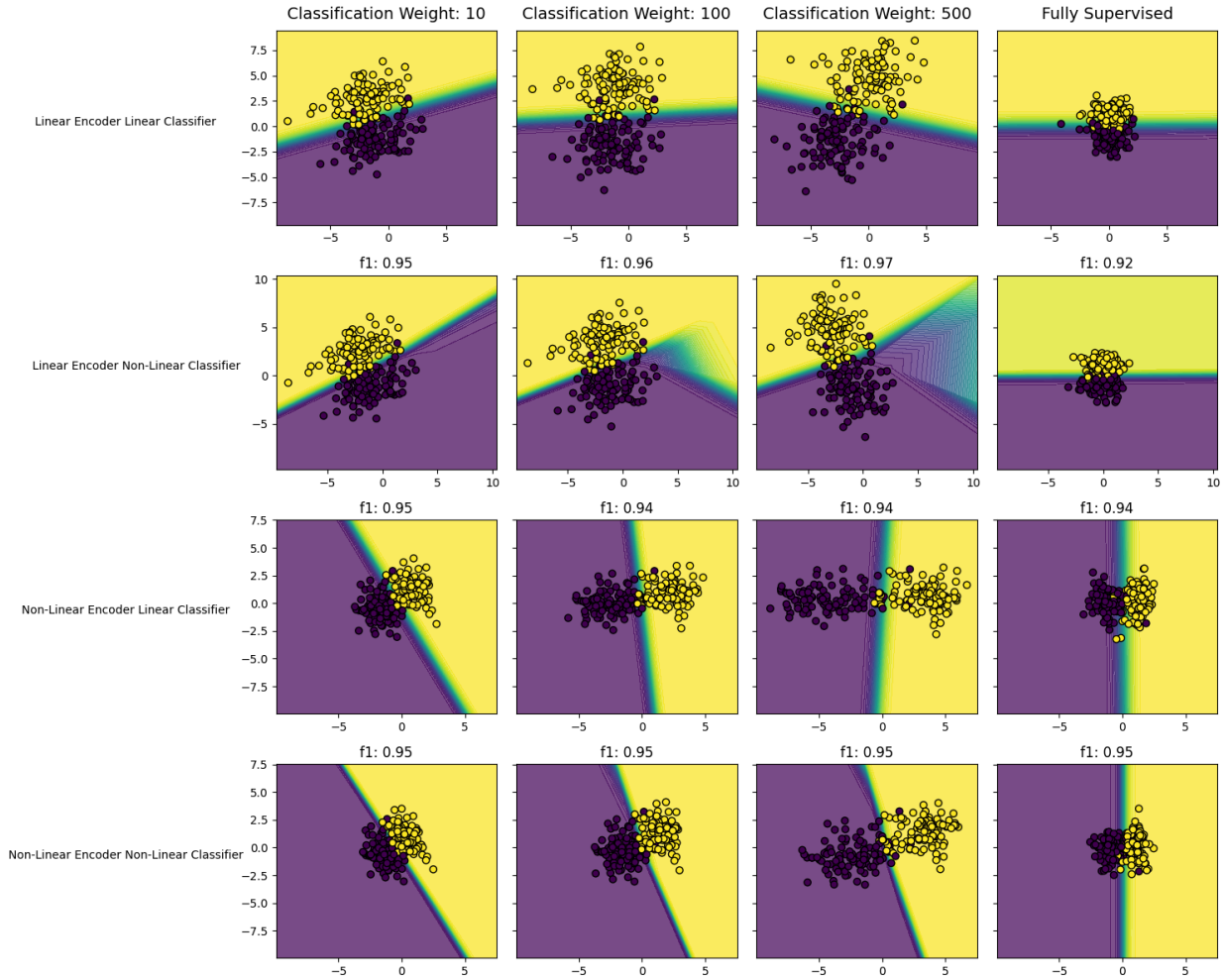
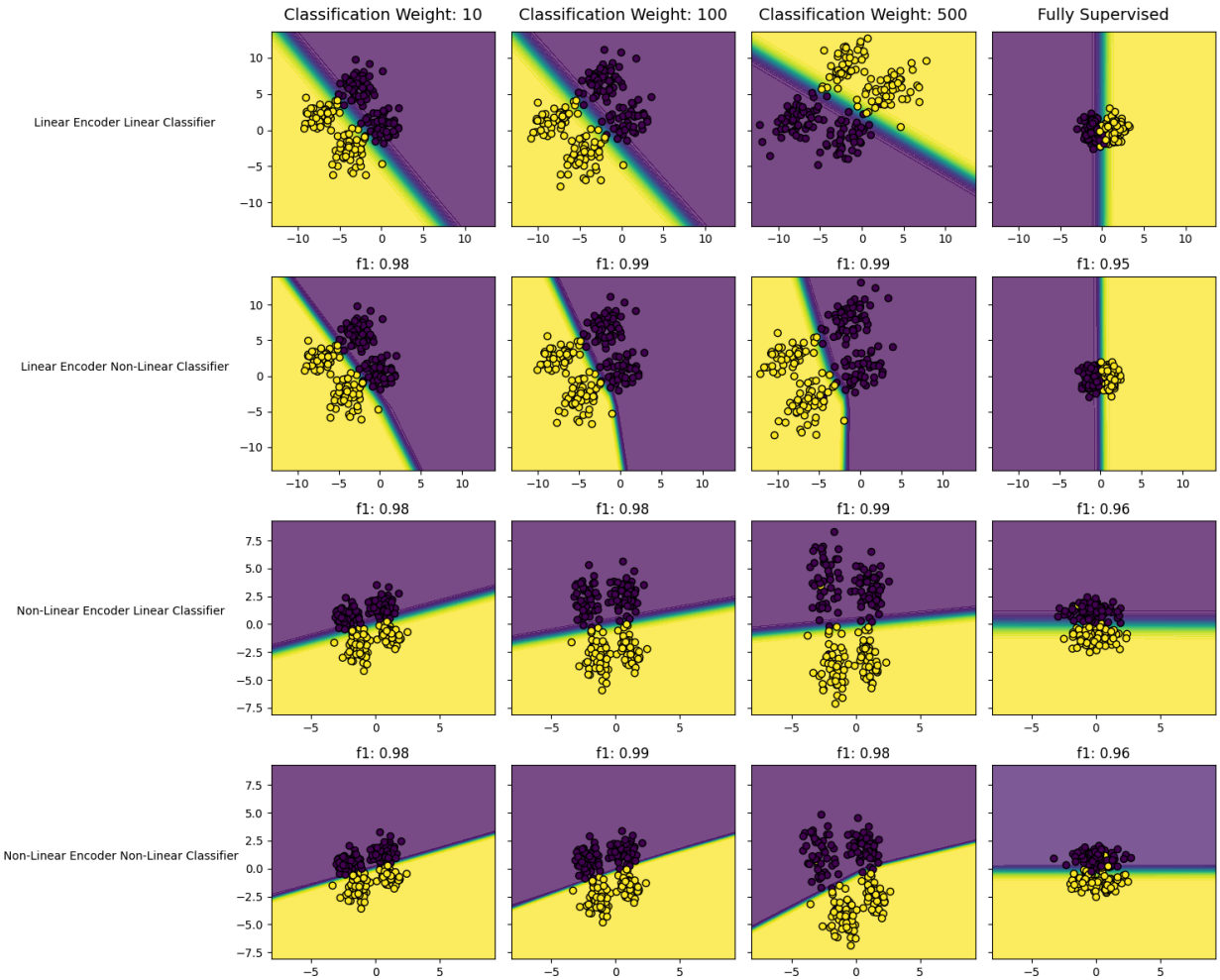


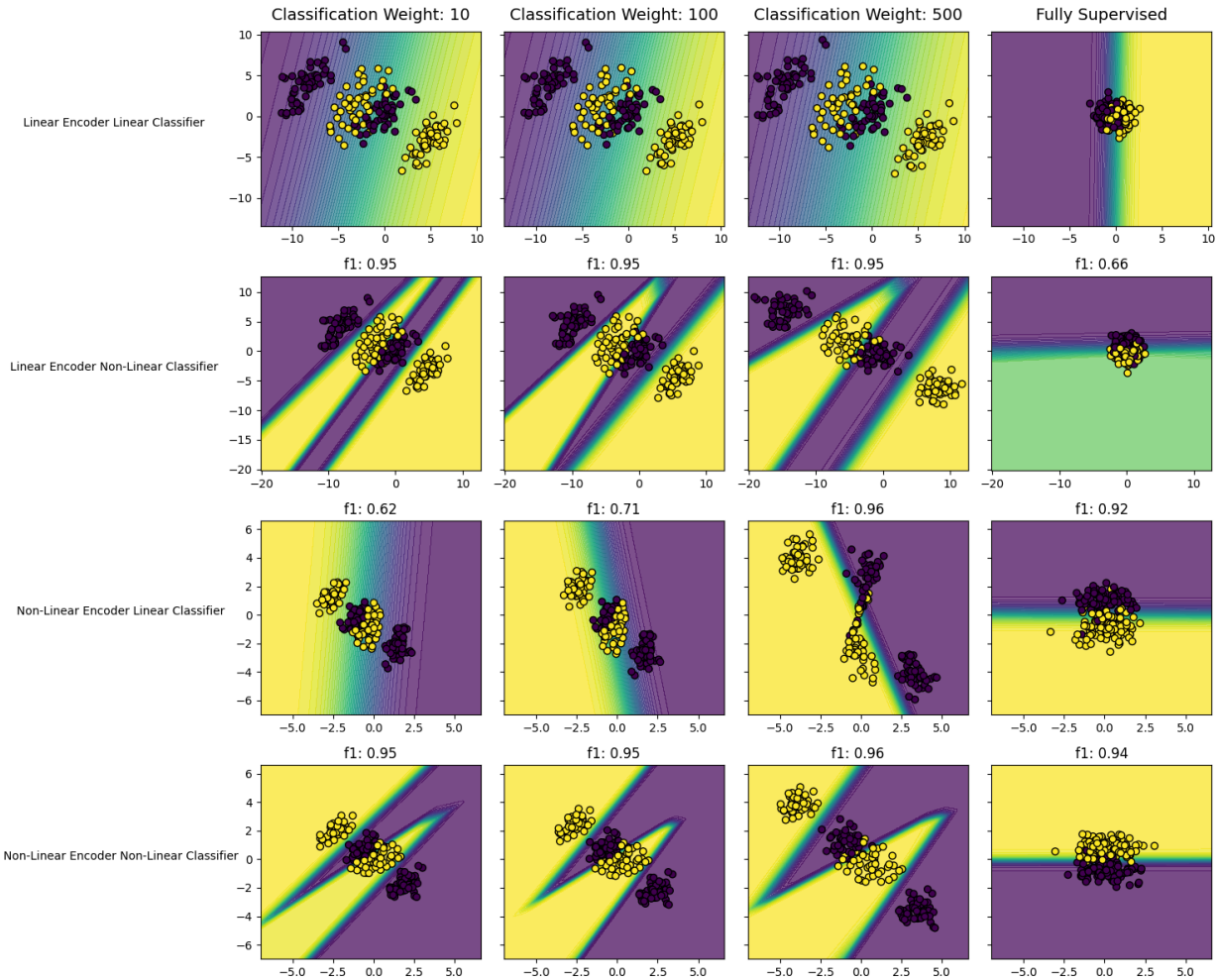
Figure 3. Learned Latent Spaces for Dataset 1. Each panel is the learned latent space for a different iteration of VAE-C model. Rows correspond to different architectures, while columns correspond to locations on the supervision continuum, with classification weights from left to right (10,100,500, and fully supervised). Each dot is the learned latent representation of a simulated fMRI observation. The color of the dot is the “observed” (simulated) behavioral label for that fMRI observation. The background color of each panel represents the models’ prediction of behavior labels for that location in the latent space. Model predictions are probabilistic so areas in which one class is not highly probable show graded coloration. In this Dataset there is relatively little difference in performance across architectures and classification weights, although slight distortions are seen as the degree of supervision increases.

Figure 4.



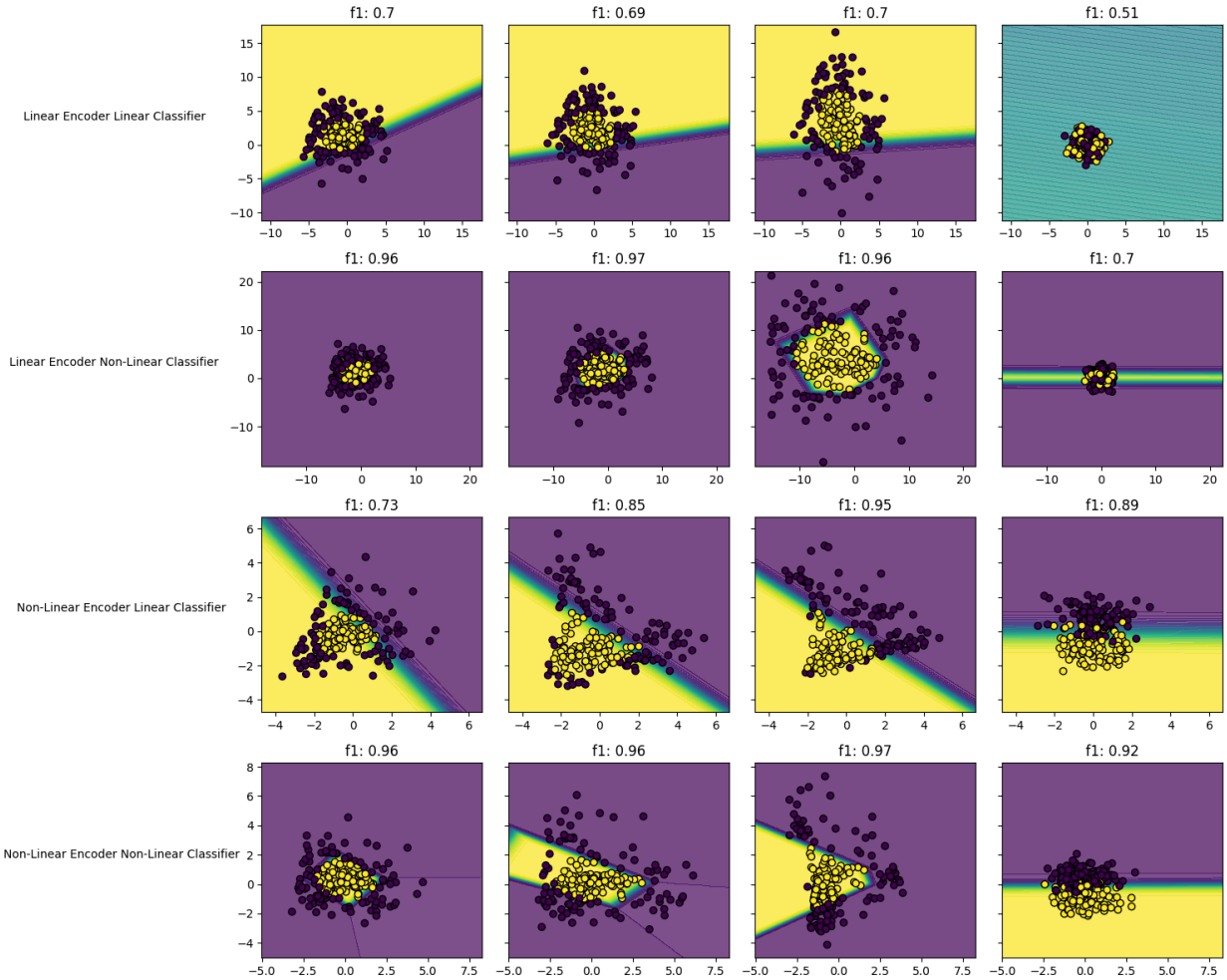
Learned Latent Spaces for Dataset 2. Each panel is the learned latent space for a different iteration of VAE-C model. Rows correspond to different architectures, while columns correspond to locations on the supervision continuum, with classification weights from left to right (10, 100, 500, and fully supervised). Each dot is the learned latent representation of a simulated fMRI observation. The color of the dot is the “observed” (simulated) behavioral label for that fMRI observation. The background color of each panel represents the models’ prediction of behavior labels for that location in the latent space. Model predictions are probabilistic so areas in which one class is not highly probable show graded coloration. In Dataset 2 the main change to the latent space is seen between columns 3 and 4 as models transition to fully supervised models. fully supervised models fail to represent the many to one mappings present in the ground-truth simulated data, despite accurately classifying the brain states.

Figure 5.



Learned Latent Spaces for Dataset 3. Each panel is the learned latent space for a different iteration of VAE-C model. Rows correspond to different architectures, while columns correspond to locations on the supervision continuum, with classification weights from left to right (10, 100, 500, and fully supervised). Each dot is the learned latent representation of a simulated fMRI observation. The color of the dot is the “observed” (simulated) behavioral label for that fMRI observation. The background color of each panel represents the models’ prediction of behavior labels for that location in the latent space. Model predictions are probabilistic so areas in which one class is not highly probable show graded coloration. For dataset 3, the latent spaces change between columns 3 and 4 as models transition to fully supervised models. Fully supervised models latent spaces fail to represent the many to one mappings present in the ground-truth simulated data. Next for the linear encoder, linear classifier (first row) the graded coloration and poor classification performance demonstrates the model’s inability to learn the nonlinear separation between behavioral labels. Finally for the nonlinear encoder, linear classifier, the latent space becomes progressively distorted along the supervision continuum as the nonlinear encoder learns a nonlinear encoding function to introduce linear separability between brain states associated with different behavioral labels.

Figure 6



Learned Latent Spaces for Dataset 4. Each panel is the learned latent space for a different iteration of VAE-C model. Rows correspond to different architectures, while columns correspond to locations on the supervision continuum, with classification weights from left to right (10, 100, 500, and fully supervised). Each dot is the learned latent representation of a simulated fMRI observation. The color of the dot is the “observed” (simulated) behavioral label for that fMRI observation. The background color of each panel represents the models’ prediction of behavior labels for that location in the latent space. Model predictions are probabilistic so areas in which one class is not highly probable show graded coloration. For dataset 4, the latent spaces change between columns 3 and 4 as models transition to fully supervised models. Fully supervised models latent spaces fail to represent the many to one mappings present in the ground-truth simulated data. Next for the linear encoder, linear classifier (first row) the graded coloration and poor classification performance demonstrates the model’s inability to learn the nonlinear separation between behavioral labels. Finally, both models with nonlinear encoders the latent space becomes progressively distorted along the supervision continuum.

Supplemental

Additional Modeling Details

The KL term in the L_{VAE} acts as a regularization term that tries to make sure that the posterior $P(z|x)$ appear closer to prior $P(z)$ i.e. $N(0, I)$. The benefit of adding the KL term is that it introduces the completeness and continuity in the latent space hence allowing the notion of gradient over information embedded in the latent space $P(z)$. This results in natural disentanglement of the information encoded in the latent space.

Neural Network Hyperparameters:

Our primary goal in comparing several neural network architectures is to show more expressive networks (i.e., nonlinear versus linear) allows for modeling a broader range of brain-behavior relationships. Since neural networks are universal function approximators, we do not consider our hyperparameter choices for the nonlinear models to confer any special properties of these networks, and believe a wide range of other architectures would perform similarly (i.e. with different number of hidden layers or number of neurons per layer). It is common in machine learning to explore a wide range of architectures to optimize performance. However, the architectures we report were the first we tested, and they performed near-optimally, so we did not pursue an exhaustive parameter search.

Our hyperparameter choices were as follows:

- For the nonlinear encoder, we used a single hidden layer with 25 neurons and a Tanh activation function. Given that our data were generated with a linear mapping from the latent space to fMRI representations, we chose this simple architecture. Tanh's output range includes negative values, which aligns with the fMRI data's range.
- For the nonlinear classifier, we used two hidden layers of sizes four and three, with ReLU activations. We added a second layer because research has shown that increasing the number of layers can improve the model's ability to learn complex functions. While the relationships in our simulations are relatively simple, we believed the additional layer would help capture the structure. ReLU activations were chosen due to their proven success in classification tasks.

Additionally the set of classification weights were chosen to sample a range a of weights from no weight on the classification objective to all weight on the classification. To do this we used the following set of weights [0, 1, 10, 100, 500, Fully Supervised].

Simulated Datasets

Overview

In this study, we conducted four simulations to evaluate our model's performance across various brain-behavior mapping conditions. The simulations generated synthetic data representing latent brain states, behavioral labels, and fMRI observations. The latent brain states were modeled as points in a 2-dimensional space, sampled from Gaussian distributions. Behavioral labels were generated based on the class membership of these points, and fMRI data was generated by projecting the latent points into a higher-dimensional space using a randomly initialized linear transformation. The parameters used for each simulation are outlined below. See figure S1 for an example for dataset 3

Dataset 1: Linearly Separable Gaussian Classes

When simulating dataset 1, latent brain states were sampled from two distinct Gaussian distributions in a 2-dimensional space. A total of 1000 data points were generated, with 500 points sampled from each Gaussian. The behavioral labels were assigned such that points from the first Gaussian were labeled as "Class A" and points from the second Gaussian as "Class B." These latent points were then projected into a 100-dimensional fMRI space using a randomly generated projection matrix.

Table 1: Parameters for Dataset 1

Parameter	**Class A (Gaussian 1)**	**Class B (Gaussian 2)**
Mean (μ\muμ)	[0.0763, 0.7799]	[2.0763, 0.7799]
Covariance Matrix	[[0.4384, 0], [0, 0.7235]]	[[0.4384, 0], [0, 0.7235]]

Dataset 2: Linearly Separable Many-to-One Mapping

When simulating dataset 2, latent brain states were sampled from four distinct Gaussian distributions, representing a more complex mapping scenario. A total of 1000 data points were sampled, with 250 points drawn from each of the four distributions. Behavioral labels were assigned such that data from two of the distributions were mapped to "Class A" and the other two distributions were mapped to "Class B." The latent points were again projected into a 100-dimensional fMRI space using the same randomly generated projection matrix as in Simulation 1.

Table 2: Parameters for Dataset 2

Parameter	Class A (Dist 1)	Class A (Dist 2)	Class B (Dist 1)	Class B (Dist 2)
Mean	[0.7182, 0.1664]	[6.7182, 0.1664]	[0.7182, 3.1664]	[6.7182, 3.1664]
Covariance Matrix	[[1.4280, 0], [0, 0.6255]]	[[1.1753, 0], [0, 0.5266]]	[[1.4280, 0], [0, 0.6255]]	[[1.1753, 0], [0, 0.5266]]

Dataset 3: Non-Linearly Separable Many-to-One Mapping

When simulating dataset 3, we modeled a nonlinearly separable many-to-one brain-behavior relationship. Latent brain states were again sampled from four distinct Gaussian distributions, with 250 points sampled from each distribution. However, in this simulation, the class labels were assigned such that the data points could not be separated by a simple linear decision boundary. The fMRI data were generated by projecting the latent points into 100-dimensional space using the same projection matrix as before.

Table 3: Parameters for Dataset 3

Parameter	Class A (Dist 1)	Class A (Dist 2)	Class B (Dist 1)	Class B (Dist 2)
Mean	[0.1946, 0.7265]	[6.1946, 6.7265]	[-3.8054, -3.2735]	[2.1946, 2.7265]
Covariance Matrix	[[0.5195, 0], [0, 1.0308]]	[[0.7053, 0], [0, 1.4147]]	[[0.5195, 0], [0, 1.0308]]	[[0.7053, 0], [0, 1.4147]]

Dataset 4: Non-Linearly Separable Many-to-One Mapping

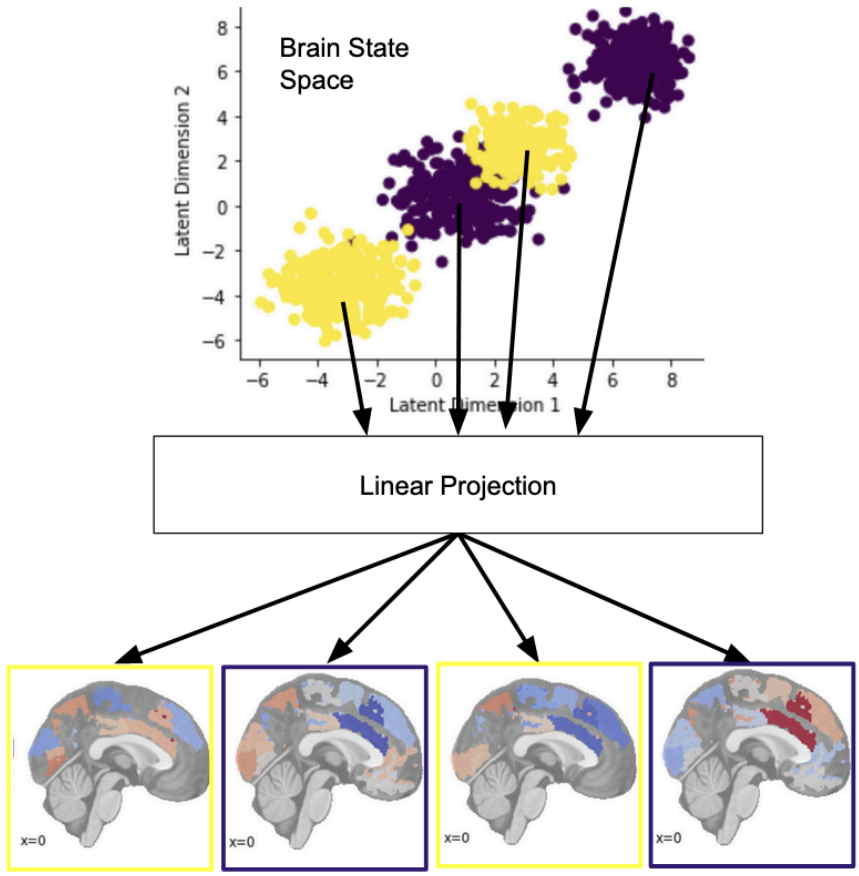
When simulating dataset 4, latent brain states were sampled from a single Gaussian distribution in the 2-dimensional space. The behavioral labels were generated by assigning the 50% of data points closest to the mean of the Gaussian to "Class A" and the remaining points to "Class B," following the "Anna Karenina" model. The fMRI data were again generated by projecting the latent data points into a 100-dimensional space using the same projection matrix as in the previous simulations.

Table 4: Parameters for Dataset 4

Parameter	Value
Mean	[0.9482, 0.0547]
Covariance Matrix	[[1.1911, 0], [0, 0.8295]]

Projection Matrix for fMRI Data Generation

In all simulations, the latent points were projected into 100-dimensional fMRI space using a random projection matrix, this projection matrix is available on the project github



Simulated Neural Data

Examples of Simulated Data Generation.

Example of Simulated data generation. Top Panel - shows simulated data distributions in the latent space for each of the 3 Datasets. Different behavioral classes are indicated by different colors. Bottom panel - a schematic of how the simulated data is mapped from the latent space to the 100d fMRI observation space, and 2 class behavioral space. Note for visualization purposes the simulated fMRI data is threshold at z scores of above 1.5.