Sparsity-Inducing Categorical Prior Improves Robustness of the Information Bottleneck

Anirban Samaddar
Michigan State University

Sandeep Madireddy
Argonne National Laboratory

Prasanna Balaprakash
Argonne National Laboratory

Tapabrata Maiti
Michigan State University

Gustavo de los Campos
Michigan State University

Ian Fischer
Google Research

Abstract

The information bottleneck framework provides a systematic approach to learning representations that compress nuisance information in the input and extract semantically meaningful information about predictions. However, the choice of a prior distribution that fixes the dimensionality across all the data can restrict the flexibility of this approach for learning robust representations. We present a novel sparsity-inducing spike-slab categorical prior that uses sparsity as a mechanism to provide the flexibility that allows each data point to learn its own dimension distribution. In addition, it provides a mechanism for learning a joint distribution of the latent variable and the sparsity, and hence it can account for the complete uncertainty in the latent space. Through a series of experiments using in-distribution and out-of-distribution learning scenarios on the MNIST, CIFAR-10, and ImageNet data, we show that the proposed approach improves accuracy and robustness compared to traditional fixed-dimensional priors, as well as other sparsity induction mechanisms for latent variable models proposed in the literature.

1 INTRODUCTION

Information bottleneck (IB) (Tishby et al. (2000)) is a deep latent variable model that poses representation compression as a constrained optimization problem to find representations Z that are maximally informative about the outputs Y while being maximally compressive about the inputs X, using a loss function expressed using a mu-
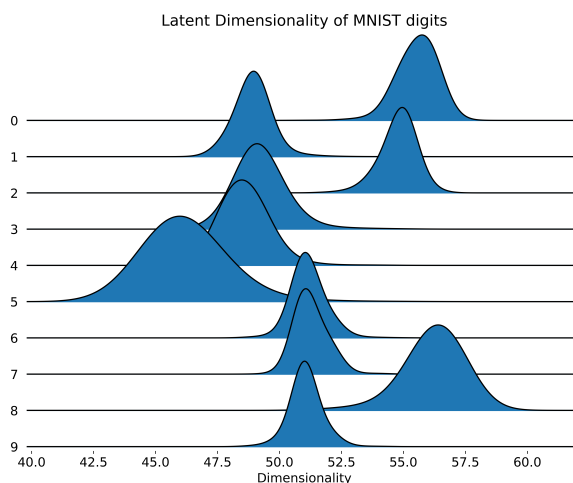

Figure 1: Plot of latent dimension distribution learned by `SparC-IB` aggregated for each of the MNIST data classes. In the vertical axis, we have 10 digits classes and in the horizontal axis we have their distribution of posterior modes of latent dimension aggregated across the test-set data points. It shows our `SparC-IB` prior provides flexibility to learn the data-specific latent dimension distribution, in contrast to fixing to a single value.

tual information (MI) metric and a Lagrangian formulation of the constrained optimization: $\mathcal{L}_{IB} = \mathrm{MI}(X;Z) - \beta \mathrm{MI}(Z;Y)$. Here, $\mathrm{MI}(X;Z)$ is the MI that reflects how much the representation compresses X, and $\mathrm{MI}(Z;Y)$ reflects how much information the representation has kept from Y.

It is standard practice to use parametric priors, such as a mean-field Gaussian prior for the latent variable Z, as seen with most latent-variable models in the literature (Tomczak (2022)). In general, however, a major limitation of these priors is the requirement to preselect a latent space complexity for all data, which can be very restrictive and lead to models that are less robust. Sparsity, when used as a mechanism to choose the complexity of the model in a flexible and data-driven fashion, has the potential to improve

the robustness of machine learning systems without loss of accuracy, especially when dealing with high-dimensional data (Ahuja et al. (2021)).

Sparsity has been considered in the context of latent variable models in a handful of works. In linear latent variable modeling, Lan et al. (2014) proposes a sparse factor model in the context of learning analytics, Chun and Keleş (2010) proposes a sparse partial least squares (sPLS) method to resolve the inconsistency issue that arises in standard PLS in a high-dimensional setting, and Yang et al. (2019) reviews advances in sparse canonical correlation analysis. Most of the work in sparse linear latent variable modeling fixes the latent dimensionality or treats it as a hyperparameter. In nonlinear latent variable modeling, sparsity was proposed primarily in the context of unsupervised learning. Notable works include sparse Dirichlet variational autoencoder (sDVAE) (Burkhardt and Kramer (2019)), epitome VAE (eVAE) (Yeung et al. (2017)), variational sparse coding (VSC) (Tonolini et al. (2020)), and InteL-VAE (Miao et al. (2021)). Most of these approaches do not take into account the uncertainty involved in introducing sparsity, and treat this as a deterministic selection problem. Approaches such as the Indian buffet process VAE (Singh et al. (2017)) relax this and allow learning a distribution over the selection parameters that induce sparsity, but only allow global sparsity. Ignoring uncertainty in selection and the flexibility to learn a local data-driven dimensionality of the latent space for each data point can lead to a loss of robustness in inference and prediction.

We also note that the aforementioned approaches have been proposed for unsupervised learning, and we are interested in the supervised learning scenario introduced with the IB approach, which poses a different set of challenges because the sparsity has to accommodate accurate prediction of Y. Only one recent work (Kim et al. (2021)) that we know of has considered sparsity in the latent variables of the IB model, but here the latent variable is assumed to be deterministic, and the sparsity applied indirectly by weighting each dimension using a Bernoulli distribution, where zero weight is equivalent to sparsification.

To that end, we make the following contributions:

1. We introduce a novel sparsity-inducing Bayesian spike-slab prior that is based on a beta-binomial formulation, where the sparsity in the latent variables of the IB model is modeled stochastically through a categorical distribution, and thus the joint distribution of latent variable and sparsity is learned with this categorical prior IB (`SparC-IB`) model through Bayesian inference. Unlike traditional spike-and-slab priors that are based on the beta-Bernoulli distribution (George and McCulloch (1993),Ishwaran and Rao (2005)) and can select dimensions randomly, `SparC-IB` imposes an order in which the dimensions are selected/activated for

each data point. This helps to infer the distribution of dimensionality more effectively.

2. We derive variational lower bounds for efficient inference of the proposed `SparC-IB` model.

3. Using in-distribution and out-of-distribution experiments with MNIST, CIFAR-10, and ImageNet data, we show an improvement in accuracy and robustness with `SparC-IB` compared to vanilla VIB models and other sparsity-inducing strategies.

4. Through extensive analysis of the latent space, we show that learning the joint distribution provides the flexibility to systematically learn parsimonious data-specific (local) latent dimension complexity (as shown in Fig. 1), enabling them to learn robust representations.

2 RELATED WORKS

Previous works in the literature on latent-variable models has looked at different sparsity-inducing mechanisms in the latent space in supervised and unsupervised settings. Burkhardt and Kramer (2019) propose a sparse Dirichlet variational autoencoder (sDVAE) that assumes a degenerate Dirichlet distribution on the latent variable. They introduce sparsity in the concentration parameter of the Dirichlet distribution through a deterministic neural network. Epitome VAE (eVAE) by Yeung et al. (2017) imposes sparsity through epitomes that select adjacent coordinates of the mean-field Gaussian latent vector and mask the others before feeding to the decoder. The authors propose training a deterministic epitome selector to select from all possible epitomes. Similar to eVAE, the variational sparse coding (VSC) proposed in Tonolini et al. (2020) introduces sparsity through a deterministic classifier.

Approach	Latent Variable	Sparsity (global/local)
sDVAE	S	D (L)
eVAE	S	D (L)
VSC	S	D (L)
InteL-VAE	S	D (L)
Drop-B	D	S (G)
IBP	S	S (G)
SparC-IB	S	S (L)

Table 1: Latent-variable models with different sparsity induction strategies, where D=Deterministic and S=Stochastic. The type of induced sparsity is in parentheses, where G is global sparsity and L the local sparsity.

In this case, the classifier outputs an index from a set of pseudo-inputs that define the prior for the latent variable. In a more recent work, InteL-VAE (Miao et al. (2021)) introduces sparsity via a dimension selector (DS) network on top of the Gaussian encoder layer in standard VAEs. The output of DS is multiplied by the Gaussian encoder to induce sparsity and then fed to the decoder. InteL-VAE has empirically shown an improvement over VSC in unsuper-

vised learning tasks, such as image generation. The Indian buffet process (IBP) (Singh et al. (2017)) learns a distribution on infinite-dimensional sparse matrices where each element of the matrix is an independent Bernoulli random variable, where the probabilities are global and come from a Beta distribution. Therefore, the sparsity induced by IBP is global and can make the coordinates of the latent variable zero for all data points.

In the IB literature, we find the aspect of sparsity in the latent space rarely explored. To the best of our knowledge, Drop-Bottleneck (Drop-B) by Kim et al. (2021) is the only work that attempts this problem. In Drop-B, a neural network extracts features from the data; then, a Bernoulli random variable stochastically drops certain features before passing them to the decoder. The probabilities of the Bernoulli distribution, similar to IBP, are assumed as global parameters and trained with other parameters of the model.

In this paper, with `SparC-IB`, we model stochasticity in both latent variables and sparsity, and we relax the global sparsity assumption by learning the distribution of (local) sparsity for each data point. In this regard, we differ from other latent-variable models. In Table 1 we summarize these different approaches by the types, stochastic (S) or deterministic (D), of the latent variable and the sparsity. Furthermore, we characterize the sparsity induced by each method by whether they impose global (G) or local (L) sparsity. The table shows that very few works incorporate stochasticity in both latent variable and sparsity-inducing mechanism.

3 INFORMATION BOTTLENECK WITH SPARSITY-INDUCING CATEGORICAL PRIOR

3.1 INFORMATION BOTTLENECK: PRELIMINARIES

Taking into account a joint distribution $P(X, Y)$ of the input variable X and the corresponding target variable Y, the information bottleneck principle aims to find a (low-dimensional) latent encoding Z by maximizing prediction accuracy, formulated in terms of mutual information $\text{MI}(Z; Y)$, given a constraint on compressing the latent encoding, formulated in terms of mutual information $\text{MI}(X; Z)$. This can be cast as a constrained optimization problem:

$$\begin{aligned} \max_Z \ & \text{MI}(Z; Y) \\ \text{s.t.} \quad & \text{MI}(X; Z) \leq C, \end{aligned} \tag{1}$$

where C can be interpreted as the compression rate or the minimum number of bits needed to describe the input data. Mutual information is obtained through a multidimensional integral that depends on the joint distribution and the marginal distribution of random variables given by $\int_{\mathcal{Z}} \int_{\mathcal{X}} P_\theta(x, z) \log \left(\frac{P_\theta(x,z)}{P(x) P(z)} \right) \, dx \, dz$, where $P_\theta(x, z) :=$

$P_\theta(z|x)P(x)$; a similar expression for $\text{MI}(Z; Y)$ needs $P_\theta(y, z) := \int P(y|x)P_\theta(z|x)P(x)\mathrm{d}x$. The integral presented to calculate MI is generally computationally intractable for large data.

Thus, in practice, the Lagrangian relaxation of the constrained optimization problem is adopted Tishby and Zaslavsky (2015):

$$\mathcal{L}_{IB}(Z) = \text{MI}(Z; Y) - \beta \, \text{MI}(X; Z) \tag{2}$$

where β is a Lagrange multiplier that enforces the constraint $\text{MI}(X; Z) \leq C$ such that a latent encoding Z is desired that is maximally expressive about Y while being maximally compressive about X. In other words, $\text{MI}(X; Z)$ is the mutual information that reflects how much the representation (Z) compresses X, and $\text{MI}(Z; Y)$ reflects how much information the representation has been kept from Y. Several approaches have been proposed in the literature to approximate mutual information $\text{MI}(X; Z)$, ranging from parametric bounds defined by variational lower bounds (Alemi et al. (2016)) to non-parametric bounds (based on kernel density estimate) (Kolchinsky et al. (2019)) and adversarial f-divergences Zhai and Zhang (2021). In this research, we focus primarily on the variational lower bounds-based approximation. Furthermore, we take a square transformation of the term $\text{MI}(X; Z)$ following Kolchinsky et al. (2019). Taking a convex transformation of the compression term makes the solution of the IB Lagrangian identifiable w.r.t. β (Rodríguez Gálvez et al. (2020)). However, for the sake of clarity, we drop this transformation from the loss function derivation. From the convexity property, all derivations with standard IB loss carry over to the loss function with this transformation.

Role of Prior Distribution and Stochasticity of Latent Variable: In the IB formulation presented above, the latent variable is assumed to be stochastic, and hence the posterior distribution of it is learned using Bayesian inference. In Alemi et al. (2016), the variational lower bound of $\mathcal{L}_{IB}(Z)$ is given as

$$\mathcal{L}_{VIB}(Z) = \mathbb{E}_{X,Y} \left[\mathbb{E}_{Z|X} \log q(Y|Z) - \beta \, \mathrm{D}_{\mathrm{KL}}(q(Z|X) || q(Z)) \right] \tag{3}$$

In equation (3), the prior $q(z)$ serves as a penalty to the variational encoder $q(z|x)$ and $q(y|z)$ is the decoder that is the variational approximation to $p(y|z)$. Alemi et al. (2016) choose the encoder family and the prior to be a fixed K-dimensional isotropic multivariate Gaussian distribution ($N(0_K, I_{K \times K})$). This choice is motivated by the simplicity and efficiency of inference using differentiable reparameterization of the Gaussian random variable Z in terms of its mean and sigma. *For complex datasets, however, this can be restrictive since the same dimensionality is imposed on all the data and hence can prohibit the latent space from learning robust features* (Miao et al. (2021)).
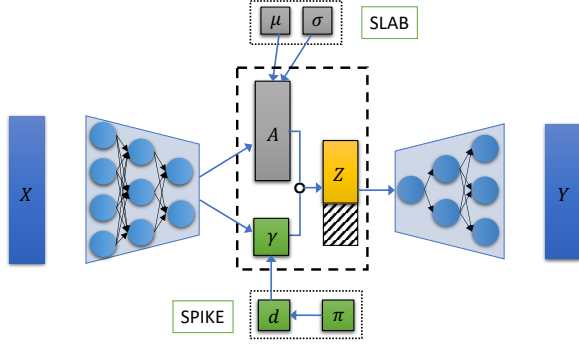
Figure 2: Schematic of the categorical prior information bottleneck (SparC-IB) model. The inputs are passed through the encoder layer to estimate the feature allocation vector A, while simultaneously using a categorical prior to sample the number of active dimensions d of A for given data that selects the complexity of latent variable Z. The obtained Z is then fed to the decoder to predict the supervised learning responses.

We describe a new family of sparsity-inducing priors that allow stochastic exploration of different dimensional latent spaces. For the proposed variational family, we derive the variational lower bound that consists of discrete and continuous variables, and show that simple reparameterization steps can be taken to draw samples efficiently from the encoder for inference and prediction.

3.2 SPARSITY-INDUCING CATEGORICAL PRIOR

A key aspect of the IB model and latent variable models in general is the dimension of Z. Fixing a very low dimension of Z can impose a limitation on the capacity of the model, while a very large Z can lead to learning a lot more nuisance information that can be detrimental to model robustness and generalizability. We formulate a data-driven approach to learn the distribution of dimensionality Z through the design of the sparsity-inducing prior.

Let $\{X_n, Y_n\}_{n=1}^N$ be N data points with $X_n \in \mathbb{R}^{1 \times p}$ and let the latent variable $Z_n = (Z_{n,1}, Z_{n,2}, ..., Z_{n,K})$, where K is the latent dimensionality. The idea is that we will fix K to be large a priori and make the prior assumption that the $k < K$ columns of Z_n are zero and therefore do not contribute to the prediction of Y_n. Therefore, the prior distribution of Z_n can be specified as follows.

$$Z_{n,k}|\gamma_{n,k} \sim (1-\gamma_{n,k})\mathbb{1}(Z_{n,k}=0) + \gamma_{n,k}\mathrm{N}(\mu_{n,k}, \sigma_{n,k}^2)$$
$$\text{OR}$$
$$Z_n = A_n \circ \gamma_n; A_n \sim \mathcal{N}(\mu_n, \Sigma_n) \tag{4}$$
$$\gamma_{n,k}|d_n = \mathbb{1}(k \leqslant d_n); d_n \sim \text{Categorical}(\pi_n) \tag{5}$$

This prior is a variation of the spike-slab family (George and McCulloch (1993)) where the sparsity-inducing parameters follow a categorical distribution. The latent vari-

able Z_n is an element-wise product between the feature allocation vector A_n and a sparsity-inducing vector γ_n. The K dimensional latent A_n is assumed to follow a K dimensional Gaussian distribution with mean vector μ_n and diagonal covariance matrix Σ_n with the variance vector σ_n^2. Note that γ_n is a vector whose first d_n coordinates are 1s and the rest are 0s, and hence $A_n \circ \gamma_n$, with the result that the first d_n coordinates of Z_n are nonzero and the rest are 0. Therefore, as we sample d_n, we consider the different dimensionality of the latent space; d_n follows a categorical distribution over the K categories with π_n as the vector of categorical probabilities whose elements $\pi_{n,k}$ denote the prior probability of the kth dimension for the data point n. In Fig. 2 we present the flow of the model from the input X to the output Y through the latent encoding layer.

3.3 VARIATIONAL BOUND DERIVATION

The variable d_n augments Z_n for latent space characterization. With a Markov chain assumption, $Y \leftrightarrow X \leftrightarrow Z$, the latent variable distribution factorization as $p(Z, d) = \prod_{n=1}^N p(Z_n|d_n)p(d_n)$, and the same family with the same factorization of the variational posterior $q(Z_n, d_n|X)$ as the prior, we derive the variational lower bound of the IB Lagrangian as follows. In the following derivation, for simplicity, we use X interchangeably for both the random variable $\in \mathbb{R}^{1 \times p}$ and the sample covariate matrix $\in \mathbb{R}^{N \times p}$.

$$\mathcal{L}_{IB}(Z) = \mathrm{MI}(Z; Y) - \beta \, \mathrm{MI}(X; Z)$$
$$\geqslant \mathcal{L}_{VIB}(Z)$$
$$= \mathbb{E}_{X,Y}\left[\mathbb{E}_{Z|X} \log q(Y|Z) - \beta \, \mathrm{D_{KL}}(q(Z|X)||q(Z))\right]$$
$$\text{(from (3))}$$
$$= \mathbb{E}_{X,Y}\left[\mathbb{E}_{Z|X} \log q(Y|Z) - \beta \, \mathrm{D_{KL}}(q(Z,d|X)||q(Z,d))\right]$$
$$= \mathcal{L}_{\text{SparC-IB}}(Z, d)$$

Note that we have introduced the random variable d along with Z in the density function of the second KL term. This is because the KL divergence between $q(Z|X)$ and $q(Z)$ is intractable. The equality of the loss function is valid since $d_n = \sum_{k=1}^K \mathbb{1}(Z_{n,k} \neq 0)$ is a deterministic function of Z_n, and we can write the density $q(z, d) = q(z)q(d|z) = q(z)\delta_{d_z}(d)$, where $\delta_c(d)$ is the Dirac delta function at c. Therefore, we have the following.

$$\mathrm{D_{KL}}(q(Z,d|X)||q(Z,d)) = \mathbb{E}_{Z,d|X} \log \frac{q(Z|X)\delta_{d_z(d)}}{q(Z)\delta_{d_z(d)}}$$
$$= \mathbb{E}_{Z|X}\mathbb{E}_{d|Z} \log \frac{q(Z|X)\delta_{d_z}(d)}{q(Z)\delta_{d_z}(d)} = \mathbb{E}_{Z|X} \log \frac{q(Z|X)}{q(Z)}$$
$$= \mathrm{D_{KL}}(q(Z|X)||q(Z))$$

Note that $\delta_{d_z}(d) = 1$, given Z, almost everywhere since $\delta_{d_z}(d) = 0 \iff d \neq d_z$ has measure 0 under $q(d|z) =$

$\delta_{d_z}(d)$. We now replace $\mathbb{E}_{X,Y}$ with the empirical version.

$$\mathcal{L}_{\texttt{SparC-IB}}(Z, d) \hat{=} \frac{1}{N} \sum_{n=1}^{N} \Bigg[\mathbb{E}_{Z_n|X}[\log q(Y_n|Z_n)]$$
$$- \beta \ \mathrm{D}_{\mathrm{KL}}(q(Z_n, d_n|X)||q(Z_n, d_n)) \Bigg]$$
$$= \frac{1}{N} \sum_{n=1}^{N} \Bigg[\mathbb{E}_{d_n|X} \mathbb{E}_{Z_n|X,d_n}[\log q(Y_n|Z_n)]$$
$$- \beta \ \big[\mathbb{E}_{d_n|X}[\mathrm{D}_{\mathrm{KL}}(q(Z_n|X, d_n)||q(Z_n|d_n))]$$
$$+ \mathrm{D}_{\mathrm{KL}}(q(d_n|X)||q(d_n)) \big] \Bigg]$$

We analyze the three terms in the above decomposition as follows.

$(i) \ \mathbb{E}_{d_n|X} \mathbb{E}_{Z_n|X,d_n}[\log q(Y_n|Z_n)]$
$$= \sum_{k=1}^{K} \mathbb{E}_{Z_n|X,d_n=k}[\log q(Y_n|Z_n, d_n = k)]\pi_{n,k}(X)$$

This term is a weighted average of the negative cross-entropy losses from models with increasing dimension of latent space, where the weights are the posterior probabilities of the dimension encoder. Therefore, *maximizing this term implies putting large weights on the dimensions of the latent space where log-likelihood is high*. During training, this term can be computed using the Monte Carlo approximation, that is, $(i) \hat{=} \frac{1}{J} \sum_{j=1}^{J} \log q(Y_n|Z_n = Z_n^{(j)}, d_n = d_n^{(j)})$, where we draw J randomly drawn samples from $q(Z_n, d_n|X)$. In our experiments, we fixed $J = 10$ everywhere during training.

$(ii) \ \mathbb{E}_{d_n|X}[\mathrm{D}_{\mathrm{KL}}(q(Z_n|X, d_n)||q(Z_n|d_n))]$
$$= \sum_{k=1}^{K} \mathrm{D}_{\mathrm{KL}}(q(Z_n|X, d_n = k)||q(Z_n|d_n = k))\pi_{n,k}(X)$$

Note that $q(z_n|d_n) = q(z_n|\gamma_n) = \mathcal{N}(z_n; \tilde{\mu}_n, \tilde{\Sigma}_n)$, where $\tilde{\mu}_n = \mu_n \circ \gamma_n$ and $\tilde{\Sigma}_n$ are diagonal with the entries $\tilde{\sigma}_n^2 = \sigma_n^2 \circ \gamma_n$. When $k < K$, this density does not exist w.r.t. the Lebesgue measure in $\mathbb{R}^K$. However, we can still define a density w.r.t. the Lebesgue measure restricted to $\mathbb{R}^k$ (see Chapter 8 in Rao (1973)), and it is the k-dimensional multivariate normal density with mean $\mu_{n,-\overline{K-k}} = (\mu_{n,1}, ..., \mu_{n,k})'$ and diagonal covariance matrix $\Sigma_{n,-\overline{K-k}}$ with diagonal entries $\sigma_{n,-\overline{K-k}}^2 = (\sigma_{1,n}^2, ..., \sigma_{n,k}^2)'$. Denoting $\mu_{n,-0} = \mu_n$, we have the following.

$$(ii)term = \sum_{k=1}^{K} \mathrm{D}_{\mathrm{KL}} \Bigg(\mathcal{N}(\mu_{n,-\overline{K-k}}(X), \Sigma_{n,-\overline{K-k}}(X))$$
$$||\mathcal{N}(\mu_{n,-\overline{K-k}}, \Sigma_{n,-\overline{K-k}}) \Bigg) \pi_{n,k}(X)$$
$$= \sum_{k=1}^{K} \sum_{\ell=1}^{k} \mathrm{D}_{\mathrm{KL}} \Bigg(\mathcal{N}(\mu_{n,\ell}(X), \sigma_{n,\ell}^2(X))$$
$$||\mathcal{N}(\mu_{n,\ell}, \sigma_{n,\ell}^2) \Bigg) \pi_{n,k}(X)$$
$$= \frac{1}{2} \sum_{k=1}^{K} \sum_{\ell=1}^{k} [\sigma_{n,\ell}^2(X) - 1 - \log(\sigma_{n,\ell}^2(X))$$
$$+ \mu_{n,\ell}^2(X)]\pi_{n,k}(X)$$

The second-last equality is due to the fact that the KL divergence of multivariate Gaussian densities whose covariances are diagonal can be written as a sum of coordinate-wise KL divergences. Since KL-divergence is always non-negative, *minimizing the above expression implies putting more probability to the smaller-dimensional latent space models since the second summation term is expected to grow with dimension k.*

$$(iii) \ \mathrm{D}_{\mathrm{KL}}(q(d_n|X)||q(d_n)) = \sum_{k=1}^{K} \log \frac{\pi_{n,k}(X)}{\pi_{n,k}} \pi_{n,k}(X)$$

Minimizing this term forces the learned probabilities to be close to the prior. Note that we are learning these probabilities for each data point (since $\pi_{n,k}(X)$ is indexed by n). In this respect, we differ from most of the stochastic sparsity-inducing approaches, such as Drop-B (Kim et al. (2021)) and IBP (Singh et al. (2017)). In these approaches, the sparsity is induced from a probability distribution with global parameterization and is not learned for each data point.

Modeling Choices for the `SparC-IB` Components: Since $Z_n = A_n \circ \gamma_n$, we are required to fix the priors for (A_n, γ_n) or (A_n, d_n). We chose K-dimensional spherical Gaussian $\mathcal{N}(0, I_K)$ as the prior for the latent variable A_n. For d_n, we assume that the kth categorical probability comes from the compound distribution of a beta-binomial model, also known as the *Polya urn* model Mahmoud (2008). Therefore,

$$\pi_n = \mathbb{P}(d_n = k) = \binom{K-1}{k-1} \frac{\mathrm{B}(a_n + k - 1, b_n + K - k)}{\mathrm{B}(a_n, b_n)}. \tag{6}$$

For simplicity, we set the prior value to be constant across the data points, that is, $(a_n, b_n) = (a, b)$. The key advantage of this choice is that we can write the probability as a differentiable function of the two shape parameters (a_n, b_n). Therefore, we can assume the same categorical distribution for the encoder; and instead of learning K probabilities $\pi_{n,k}(X)$ we can learn $(a_n(X), b_n(X))$,

which significantly reduces the dimensionality of the parameter space. However, learning $\pi_{n,k}(X)$ provides more flexibility because the shape of the distribution is not constrained, while learning $(a_n(X), b_n(X))$ constrains $\pi_{n,k}(X)$ to follow according to the shape of the compound distribution, which depends on $a_n(X)$ and $b_n(X)$. In our experiments, we tested both approaches and found that learning $(a_n(X), b_n(X))$ produces better results.

4 EXPERIMENTAL RESULTS

In this section, we present experimental results that compare the performance of `SparC-IB` with the most recent sparsity-inducing strategies proposed in the literature: the Drop-B model (Kim et al. (2021)) and InteL-VAE (Miao et al. (2021)). We could not find an open source implementation for either of the two approaches and therefore we have coded our own implementation where we adapt them in the context of information bottleneck (link to the code base is provided in the Appendix A.5). In this section, these two approaches are called Drop-VIB and Intel-VIB. In addition, we compare our model with the baseline mean-field Gaussian VIB approach, where the latent dimension is fixed to a single value across all data. Note that we apply the square transformation (Kolchinsky et al. (2019)) to the estimator of $\text{MI}(X; Z)$ for all the methods.

We evaluated the `SparC-IB` model for *in-distribution* prediction accuracy in a supervised classification scenario using the MNIST and CIFAR-10 data, which have a small number of classes, and also the ImageNet data, where the number of classes is large. Furthermore, we evaluated the robustness of `SparC-IB` trained on these three datasets in *out-of-distribution* scenarios, specifically with rotation, white-box attacks with noise corruptions, and black-box adversarial attacks.

The selection of the Lagrange multiplier β controls the amount of information learned from the input (that is, $\text{MI}(X; Z)$) by the latent space. We have chosen a common β where $(\text{MI}(X; Z), \text{MI}(Z; Y))$ is close to the *minimum necessary information* or MNI (suggested in Fischer (2020)) for all models. MNI is a point in the information plane where $\text{MI}(X; Z) = \text{MI}(Z; Y) = \text{H}(Y)$, where the entropy is indicated by $\text{H}(Y)$. We evaluated the robustness of each model using a single value of β. The value of β we chose to compare the models for MNIST is ~ 0.08, for CIFAR-10 it is ~ 0.04, and for ImageNet it is ~ 0.02. The choice of β is discussed in more detail in the Appendix A.6.

We use the encoder-decoder architecture from Rodríguez Gálvez et al. (2020) for MNIST, from Yu et al. (2021) for CIFAR-10, and from Alemi et al. (2016) for ImageNet. Note that we learn the parameters of the dimension distribution in both compound and categorical strategies by splitting the encoder network head into two parts, as depicted in Fig. 2. Full details of the architectures

have been discussed in the Appendix A.1. Furthermore, following Fischer and Alemi (2020), we pass the mean of the encoder Z to the decoder at the test time to make a prediction.

Prior probabilities act as regularizers in learning the dimension probabilities in both categorical and compound strategies (the third term of $\mathcal{L}_{\texttt{SparC-IB}}(Z, d)$). They also model the prior knowledge or inductive bias that one may have. The prior probability distribution in this case (8) can be set by the choice of hyperparameters (a, b). we evaluated two different cases, $(a, b) = (1, 3)$ and $(2, 2)$ for both the categorical and compound distribution strategies. The choice $(a, b) = (1, 3)$ puts more probability mass on the lower dimensions, and gradually decays with dimension, whereas $(a, b) = (2, 2)$ penalizes models of too high or too low dimensions. The appendix A.7 shows the prior probabilities of the dimensions for both choices.

4.1 IN-DISTRIBUTION DATA

In this section, we compare the performance of `SparC-IB` with Intel-VIB, Drop-VIB, and the vanilla fixed-dimensional VIB approach on the MNIST, CIFAR-10 test set, and ImageNet validation set. Beyond these choices, we have also compared `SparC-IB` with a discrete latent space IB model following VQ-VAE (Van Den Oord et al. (2017)) on MNIST data. We train each model for 5 values of the Lagrange multiplier β in the set $(0.02, 0.04, 0.06, 0.08, 0.1)$. We calculate the validation set error for each model for these β values. Since increasing β penalizes the amount of information retained by the latent space about the inputs, we expect the error to increase as β increases.

Methods	MNIST		CIFAR-10	
	Acc %	LL	Acc %	LL
SparC-IB	**98.65** (0.001)	**3.24** (0.004)	84.44 (0.015)	**2.53** (0.010)
Drop-VIB	98.25 (0.000)	3.12 (0.003)	**85.43** (0.004)	2.37 (0.015)
Intel-VIB	98.51 (0.001)	3.23 (0.007)	82.77 (0.010)	2.49 (0.063)
Fixed K: 6	98.27 (0.001)	3.22 (0.004)	82.29 (0.050)	2.49 (0.107)
Fixed K: 32	98.54 (0.001)	3.23 (0.003)	83.76 (0.012)	2.44 (0.013)
Fixed K: 128	85.28 (0.165)	2.85 (0.358)	81.88 (0.001)	2.48 (0.036)

Table 2: In-distribution performance of all methods in terms of accuracy and log-likelihood (SD in the parenthesis) at MNI for MNIST and CIFAR-10 (maximum for each column highlighted). `SparC-IB` performs as good as the best performing model in terms of both metrics.

For MNIST, we find that, across β, the compound distribution prior with $(a, b) = (2, 2)$ performs best in terms of in-distribution prediction accuracy across β as compared to other `SparC-IB` choices, fixed-dimensional VIB approaches and Drop-VIB and Intel-VIB, as shown in Fig. 3(a) . For CIFAR-10 data, we observe that compound strategy with prior $(a, b) = (1, 3)$ has the best accuracy compared to other `SparC-IB` choices at MNI, fixed-dimensional VIB models and Intel-VIB but is slightly
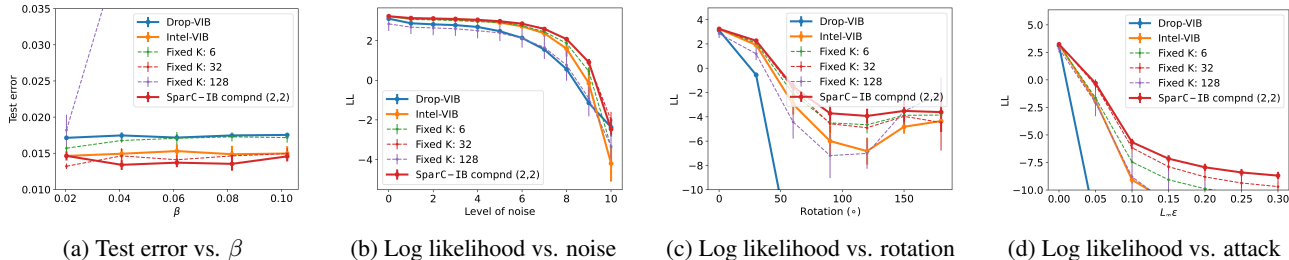
(a) Test error vs. β (b) Log likelihood vs. noise (c) Log likelihood vs. rotation (d) Log likelihood vs. attack

Figure 3: In- and out-of- distribution performance on MNIST.

(a) Test error vs. β (b) Log likelihood vs. noise (c) Log likelihood vs. attack
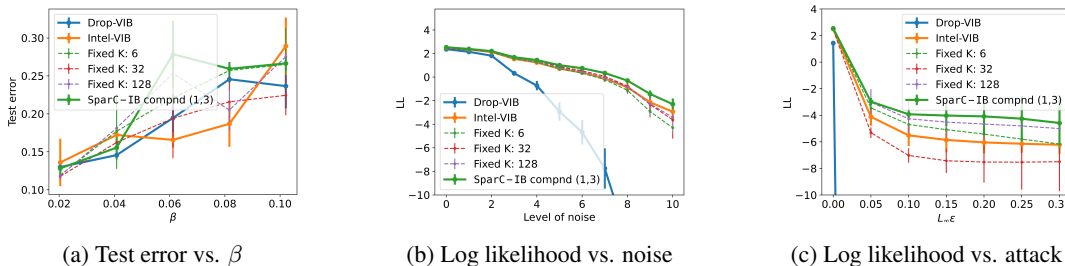
Figure 4: In- and out-of- distribution performance on CIFAR-10.

lower than Drop-VIB (numbers are shown in Table 2). However, we find from Table 2 that SparC-IB compound (1,3) has the highest log-likelihood at MNI among the other approaches. Furthermore, we have found that the best test accuracy for the discrete latent space IB is 97.79% (avg. over 3 seeds) in MNIST, which is lower than SparC-IB.

Methods	ImageNet	
	Acc %	LL
SparC-IB	79.71 (0.001)	8.48 (0.007)
Drop-VIB	79.86 (0.000)	8.19 (0.002)
Intel-VIB	**79.88 (0.000)**	**8.53 (0.003)**
Fixed K: 1024	79.88 (0.000)	8.52 (0.005)

Table 3: In-distribution performance of all methods in terms of accuracy and log-likelihood (SD in the parenthesis) at MNI for ImageNet (maximum for each column highlighted). All models are close in terms of both metrics (especially log-likelihood).

For Imagenet data, we observe that the compound strategy with prior $(a, b) = (2, 2)$ has the best validation accuracy compared to other SparC-IB choices. Furthermore, the in-distribution performance is at least as good as the fixed-dimensional VIB model (Table 3), but it has a slightly lower accuracy compared to the Drop-VIB and Intel-VIB. The test error for Intel-VIB is very high when $\beta > 0.04$. This behavior is due to the fact that the dimension selector in Intel-VIB (Section 6.3 in Miao et al. (2021)) is pruning almost all values of the latent allocation vector A for values of $\beta > 0.04$.

4.2 OUT-OF-DISTRIBUTION DATA

We consider three out-of-distribution scenarios to measure the robustness of our approach and compare it with vanilla VIB with fixed latent dimension capacity, as well

as with other sparsity-inducing strategies. The first is a white-box attack, in which we systematically introduce shot noise corruptions into the test data Mu and Gilmer (2019); Hendrycks and Dietterich (2019). The second is a rotation transform (only for MNIST data). The third is the black-box adversarial attack simulated using the projected gradient descent (PGD) strategy Madry et al. (2017). We use the log-likelihood metric to compare the out-of-distribution performance of the methods (Ovadia et al. (2019)). Comparison in terms of other metrics is included in the Appendix A.8.

4.2.1 NOISE CORRUPTION

White-box attack or noise corruption is generated by adding shot noise. Following Mu and Gilmer (2019), Poisson noise is generated pixel-by-pixel and added to the test images for both MNIST and CIFAR-10. The levels of noise along the horizontal axis of panel (b) of Fig. 3 and Fig. 4 represent an increasing degree of noise added to the images of the validation set. For our approach and each of the five models that are being compared, we plot the log-likelihood as a function of the level of noise to assess the robustness of these approaches. We find that with both MNIST (panels (b) in Fig. 3) and CIFAR-10 (panels (b) in Fig. 4), for each of these three metrics, SparC-IB outperforms all other approaches compared. For ImageNet (panel (b) in Fig. 5), we find that Drop-VIB has a higher likelihood than our approach, possibly due to a large amount of input information (estimated $MI(X; Z) = 57.56$ for Drop-VIB and $= 10.35$ for SparC-IB) learned in the latent space.

4.2.2 ROTATION

In this scenario, we evaluate the models trained on MNIST data with increasingly rotated digits following Ovadia et al.

(a) Test error vs. β value

(b) Log likelihood vs. noise
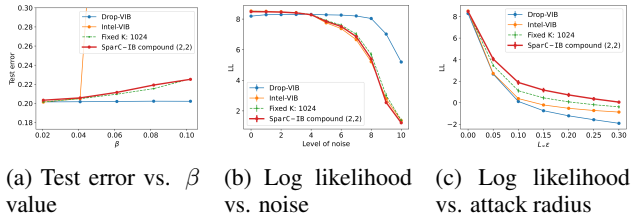
(c) Log likelihood vs. attack radius

Figure 5: In- and out-of- distribution results on ImageNet.

(2019), which simulates the scenario of data that are moderately out of distribution to the data used for training the model. We show the results of the experiments in panel (c) in Fig. 3. We can see that `SparC-IB` with the compound distribution prior outperforms the rest of the five models in terms of log-likelihood, we also note that the performance of the Drop-VIB has the lowest in this scenario.

4.2.3 ADVERSARIAL ROBUSTNESS

Multiple approaches have been proposed in the literature to assess the adversarial robustness of a model Tramer et al. (2020). Among them, perhaps the most commonly adopted approach is to test the accuracy in adversarial examples that are generated by adversarially perturbing the test data. This perturbation is in the same spirit as the noise corruption presented in the preceding section, but is designed to be more catastrophic by adopting a black-box attack approach that chooses a gradient direction of the image pixels at some loss and then takes a single step in that direction. The projected gradient descent is an example of such an adversarial attack. Following Alemi et al. (2016), we evaluated the robustness of the model to the PGD attack with 10 iterations. We use the L_∞ norm to measure the size of the perturbation, which in this case is a measure of the largest single change in any pixel. Log-likelihood as a function of the perturbed L_∞ distance for MNIST (panel (d) in Fig. 3), for CIFAR-10 (panel (c) in Fig. 4), and for ImageNet (panel (c) in Fig. 5) *show that the* `SparC-IB` *approach provides the highest log-likelihood across the attack radius in all three datasets.*

4.3 ANALYSIS OF THE LATENT SPACE

A key property of `SparC-IB` is the ability to jointly learn the latent allocation and the dimension of the latent space for each data point. In this section, our aim is to disentangle the information learned in the latent space of the `SparC-IB` approach by analyzing the posterior distribution of the dimension variable and the information learned in the latent allocation vector for MNIST data. Similar analyses for CIFAR-10 and ImageNet are in the Appendix A.8.

4.3.1 FLEXIBLE LATENT SPACE DIMENSION

A distinct feature of the proposed approach is the flexibility enabled by the sparsity-inducing prior for learning a data-
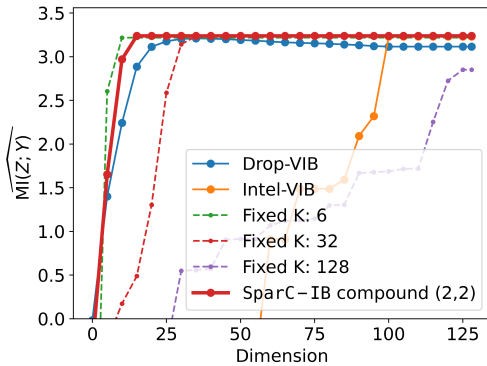

Figure 6: Information content plot for different latent dimensions (averaged across seeds). We observe that `SparC-IB` learns the maximum information in a small dimensional latent space.

dependent dimension distribution. To demonstrate this, for a compound model with $(a, b) = (2, 2)$ in Fig. 1 we show the distribution of posterior modes of dimension distribution across data points, aggregated per MNIST digit. We see that, in fact, each digit, on average, preferred to have a different latent dimension. We further note that the mode values depicted by the plot for digits 5 and 8 are farther away from the rest of the digits. Note that the pattern in Fig. 1 is for a single seed. Although the pattern in dimension distribution modes changes across the seeds, the separation between the classes remains (see A.9.1 in the Appendix for details). We also observe a similar separation of the latent dimensionality across classes with CIFAR-10 and ImageNet data (Appendix Fig. A7).

4.3.2 ANALYSIS OF INFORMATION CONTENT

The `SparC-IB` prior in (6) induces an ordered selection of the dimensions of the latent allocation vector A_n based on the dimension d_n. In Fig. 6, we plotted the estimated mutual information or $\widehat{\text{MI}(Z; Y)}$ against the increasing dimension of the latent space (in increments of 5) for all models in MNIST. For a given dimension d, $\widehat{\text{MI}(Z; Y)} = \frac{1}{N} \sum_{n=1}^{N} \log q(Y_n | Z_n = \mu_n(X) \circ \gamma(d))$, where the first d coordinates of $\gamma(d)$ are 1s and the rest are 0s. We observe that the `SparC-IB` model has been able to code the learned information in the first few (~ 15) dimensions of the latent space. In contrast, we notice that the fixed-dimensional VIB models, Drop-VIB, and Intel-VIB require close to the full latent space to encode similar information levels. This characteristic perhaps hinders these models in achieving good robustness performance consistently on all data. However, we believe that further investigation is needed to establish this claim. For CIFAR-10, we observed the same characteristics of the information content plot in Fig. 6 whereas for ImageNet the information plateaus at a larger dimension than Fig. 6 (see Fig. A8 in the Appendix).
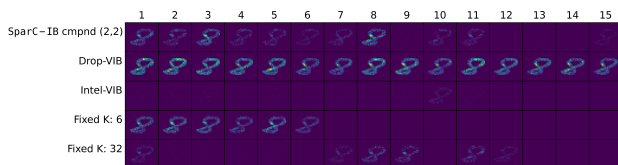
Figure 7: Plot of pixel importance in encoding the latent allocation mean (averaged across seeds) for a sample from the MNIST test data. We observe that `SparC-IB` learns important features of the latent space in the first few dimension of the latent space.

4.3.3 VISUALIZING THE LATENT DIMENSIONS

In deep neural networks, the conductance of a hidden unit, proposed in Dhamdhere et al. (2018), is the flow of information from the input through that unit to the predictions. In this spirit, we measure the importance scores of individual pixels in the dimensions of the latent mean allocation vector μ for the MNIST data. The computation details have been added in the Appendix A.9.4. In Fig. 7, we plotted these measures for different methods (averaged across seeds) for the first 15 dimensions of μ for a sample from the MNIST test data. We observe that `SparC-IB` encodes important features in the first few dimensions of the latent space. For Intel-VIB and VIB with K = 32, we see that the pixel information is spread over a large set of dimensions of the latent space. For Drop-VIB, we notice that it learns a lot of information in all the 15 dimensions. Fig. 7 also helps explain the information jumps in Fig. 6. We notice that the jump in the information content occurs when we include the dimensions that have learned important pixel information, e.g. for `SparC-IB` dimensions 3 and 8. Note that the Intel-VIB and fixed-dimensional VIB models with high dimensions have many dimensions with very little information about the input. Although Fig. 7 exhibits some features of the digit learned by the latent space (e.g., the middle part of the digit has high importance for `SparC-IB`), in our experiments, we have not been able to extract meaningful features in the latent dimensions that are common across all the digits. In our view, this demands further investigation.

5 CONCLUSIONS

In summary, we propose the `SparC-IB` method in this paper, which models the latent variable and its dimension through a Bayesian spike-and-slab categorical prior and derived a variational lower bound for efficient Bayesian inference. This approach accounts for the full uncertainty in the latent space by learning a joint distribution of the latent variable and the sparsity. We compare our approach with commonly used fixed-dimensional priors, as well as using those sparsity-inducing strategies previously proposed in the literature through experiments on MNIST, CIFAR-10, and ImageNet in both the in-distribution and out-of-

distribution scenarios (such as noise corruption, adversarial attacks, and rotation). We find that our approach obtains as good accuracy and robustness as the best-performing model in all the cases, and in some cases it outperforms the other models. This is important because we found that other VIB algorithms considered performed well on a few datasets but have significantly poor performance on the others. In addition, we show that enabling each data to learn their own dimension distribution leads to separation of dimension distribution between output classes, thus substantiating that latent dimension varies class-wise. Furthermore, we find that the `SparC-IB` approach provides a compact latent space in which it learns important data features in the first few dimensions of the latent space, which is known to lead to superior robustness properties.

There are several avenues for future research based on the proposed model `SparC-IB`. Since the latent dimensionality of the data is modeled through a Bayesian spike-and-slab prior with a categorical spike distribution over the dimension, an interesting avenue for future research could be to find a rich class of hierarchical priors or a non-parametric stick-breaking prior. Another direction might be to explore other approaches to data-driven mutual information estimation to tighten the lower bound of $\mathrm{MI}(X; Z)$. A key aspect of this work is its broader appeal in the field of adaptive selection of model complexity in machine learning. Beyond IB, the proposed sparsity-inducing prior is a promising candidate for the probabilistic node and depth selection of DNN. Our approach for data-specific complexity learning can potentially be applied to have a NN with an optimal number of layers with ordered and sparse information captured in each layer.

6 Acknowledgments

References

Kartik Ahuja, Ethan Caballero, Dinghuai Zhang, Jean-Christophe Gagnon-Audet, Yoshua Bengio, Ioannis Mitliagkas, and Irina Rish. Invariance principle meets information bottleneck for out-of-distribution generalization. *Advances in Neural Information Processing Systems*, 34:3438–3450, 2021.

Alexander A Alemi, Ian Fischer, Joshua V Dillon, and Kevin Murphy. Deep variational information bottleneck. *arXiv preprint arXiv:1612.00410*, 2016.

Sophie Burkhardt and Stefan Kramer. Decoupling sparsity and smoothness in the Dirichlet variational autoencoder topic model. *J. Mach. Learn. Res.*, 20(131):1–27, 2019.

Hyonho Chun and Sündüz Keleş. Sparse partial least squares regression for simultaneous dimension reduction and variable selection. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 72(1):3–25, 2010.

Kedar Dhamdhere, Mukund Sundararajan, and Qiqi Yan. How important is a neuron?, 2018. URL https://arxiv.org/abs/1805.12233.

Ting-Han Fan, Ta-Chung Chi, Alexander I. Rudnicky, and Peter J. Ramadge. Training discrete deep generative models via gapped straight-through estimator, 2022. URL https://arxiv.org/abs/2206.07235.

Ian Fischer. The conditional entropy bottleneck. *Entropy*, 22(9), 2020. ISSN 1099-4300. doi: 10.3390/e22090999. URL https://www.mdpi.com/1099-4300/22/9/999.

Ian Fischer and Alexander A Alemi. Ceb improves model robustness. *Entropy*, 22(10):1081, 2020.

Edward I George and Robert E McCulloch. Variable selection via Gibbs sampling. *Journal of the American Statistical Association*, 88(423):881–889, 1993.

Dan Hendrycks and Thomas Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. In *7th International Conference on Learning Representations (ICLR-2019)*, 2019. URL https://openreview.net/pdf?id=HJz6tiCqYm.

Hemant Ishwaran and J. Sunil Rao. Spike and slab variable selection: Frequentist and Bayesian strategies. *The Annals of Statistics*, 33(2):730 – 773, 2005. doi: 10.1214/009053604000001147. URL https://doi.org/10.1214/009053604000001147.

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

Jaekyeom Kim, Minjung Kim, Dongyeon Woo, and Gunhee Kim. Drop-bottleneck: Learning discrete compressed representation for noise-robust exploration. *arXiv preprint arXiv:2103.12300*, 2021.

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

Narine Kokhlikyan, Vivek Miglani, Miguel Martin, Edward Wang, Bilal Alsallakh, Jonathan Reynolds, Alexander Melnikov, Natalia Kliushkina, Carlos Araya, Siqi Yan, et al. Captum: A unified and generic model interpretability library for pytorch. *arXiv preprint arXiv:2009.07896*, 2020.

Artemy Kolchinsky, Brendan D. Tracey, and David H. Wolpert. Nonlinear information bottleneck. *Entropy*, 21(12), 2019. ISSN 1099-4300. doi: 10.3390/e21121181. URL https://www.mdpi.com/1099-4300/21/12/1181.

Andrew S Lan, Andrew E Waters, Christoph Studer, and Richard G Baraniuk. Sparse factor analysis for learning and content analytics. *Journal of Machine Learning Research (JMLR)*, 15(57):1959–2008, 2014.

Chris J Maddison, Andriy Mnih, and Yee Whye Teh. The concrete distribution: A continuous relaxation of discrete random variables. *arXiv preprint arXiv:1611.00712*, 2016.

Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. *arXiv preprint arXiv:1706.06083*, 2017.

Hosam Mahmoud. *Pólya urn models*. Chapman and Hall/CRC, 2008.

Ning Miao, Emile Mathieu, N Siddharth, Yee Whye Teh, and Tom Rainforth. On incorporating inductive biases into vaes. *arXiv preprint arXiv:2106.13746*, 2021.

Norman Mu and Justin Gilmer. Mnist-c: A robustness benchmark for computer vision. *arXiv preprint arXiv:1906.02337*, 2019.

Yaniv Ovadia, Emily Fertig, Jie Ren, Zachary Nado, David Sculley, Sebastian Nowozin, Joshua Dillon, Balaji Lakshminarayanan, and Jasper Snoek. Can you trust your model's uncertainty? Evaluating predictive uncertainty under dataset shift. *Advances in Neural Information Processing Systems*, 32, 2019.

Calyampudi Radhakrishna Rao. *Linear statistical inference and its applications*, volume 2. Wiley New York, 1973.

Borja Rodríguez Gálvez, Ragnar Thobaben, and Mikael Skoglund. The convex information bottleneck Lagrangian. *Entropy*, 22(1):98, Jan. 2020. ISSN 1099-4300. doi: 10.3390/e22010098. URL http://dx.doi.org/10.3390/e22010098.

Rachit Singh, Jeffrey Ling, and Finale Doshi-Velez. Structured variational autoencoders for the Beta–Bernoulli process. In *NIPS 2017 Workshop on Advances in Approximate Bayesian Inference*, 2017.

Christian Szegedy, Sergey Ioffe, Vincent Vanhoucke, and Alex Alemi. Inception-v4, inception-resnet and the impact of residual connections on learning, 2016. URL https://arxiv.org/abs/1602.07261.

Naftali Tishby and Noga Zaslavsky. Deep learning and the information bottleneck principle. In *2015 IEEE Information Theory Workshop (ITW)*, pages 1–5. IEEE, 2015.

Naftali Tishby, Fernando C Pereira, and William Bialek. The information bottleneck method. *arXiv preprint physics/0004057*, 2000.

Jakub M Tomczak. *Deep Generative Modeling*. Springer Nature, 2022.

Francesco Tonolini, Bjørn Sand Jensen, and Roderick Murray-Smith. Variational sparse coding. In *Uncertainty in Artificial Intelligence*, pages 690–700. PMLR, 2020.

Florian Tramer, Nicholas Carlini, Wieland Brendel, and Aleksander Madry. On adaptive attacks to adversarial example defenses. *Advances in Neural Information Processing Systems*, 33:1633–1645, 2020.

Aaron Van Den Oord, Oriol Vinyals, et al. Neural discrete representation learning. *Advances in neural information processing systems*, 30, 2017.

Xinghao Yang, Weifeng Liu, Wei Liu, and Dacheng Tao. A survey on canonical correlation analysis. *IEEE Transactions on Knowledge and Data Engineering*, 33(6):2349–2368, 2019.

Serena Yeung, Anitha Kannan, Yann Dauphin, and Li Fei-Fei. Tackling over-pruning in variational autoencoders. *arXiv preprint arXiv:1706.03643*, 2017.

Xi Yu, Shujian Yu, and José C Príncipe. Deep deterministic information bottleneck with matrix-based entropy functional. In *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 3160–3164. IEEE, 2021.

Penglong Zhai and Shihua Zhang. Adversarial information bottleneck. *arXiv preprint arXiv:2103.00381*, 2021.

A   APPENDIX

A.1   MODEL ARCHITECTURES AND HYPERPARAMETER SETTINGS

For the proposed `SparC-IB` model, we need to select the neural architecture to be used for the latent space mean and variance, as well as the dimension encoder that learns categorical probabilities or compound distribution parameters. To learn the parameters of the dimension distribution in both compound and categorical strategies, we split the head of the encoder network into two parts, as depicted in Fig. 2. The first part estimates the parameters of the latent allocation variable A and the second part estimates the parameters of the dimension variable d. For MNIST, we follow Rodríguez Gálvez et al. (2020) and use an MLP encoder with three fully connected layers, the first two of dimensions 800 and ReLu activation, and the last layer with dimension $2K + 2$ for the compound strategy and $3K$ for the categorical strategy. The decoder consists of two fully connected layers, the first of which has dimension 800 and the second predicts the softmax class probabilities in the last layer. For CIFAR-10, following Yu et al. (2021), we adopt a VGG16 encoder and a single-layered neural network as decoder. We choose the final sequential layer of VGG16 as the bottleneck layer that outputs the parameters of the latent space.

For ImageNet, we crop the images at its center to make them 299×299 pixels and normalize them to have mean = (0.5, 0.5, 0.5) and standard deviation = (0.5, 0.5, 0.5). We have followed the implementation of Alemi et al. (2016) where we transform the ImageNet data with a pre-trained Inception Resnet V2 (Szegedy et al. (2016)) network without the output layer. Under this transformation, the original ImageNet images reduce to a 1534-dimensional representation, which we used for all our results. Following Alemi et al. (2016), we use an encoder with two fully connected layers, each with 1024 hidden units, and a single-layer decoder architecture.

For comparison with other sparsity-inducing approaches, we chose the two most recent works: the Drop-B model (Kim et al. (2021)) and the InteL-VAE (Miao et al. (2021)). The Drop-B implementation requires a feature extractor. For all three data sets, we choose the same architecture for the feature extractor as the encoder of `SparC-IB` until the final layer, which has K dimensions. We assume the same decoder architecture for Drop-B as for `SparC-IB`. Additionally, for Drop-B, the K Bernoulli probabilities are trained with the other parameters of the model. For InteL-VAE, the encoder and decoder architectures are chosen to be the same as in `SparC-IB` for all data sets. In addition, this model requires a dimension selector (DS) network. Following the experiments in Miao et al. (2021), we select three fully connected layers for the DS network with ReLu activations between, where the first two layers have dimension 10 and the last layer has dimension K. We fix K (that is, the prior assumption of dimensionality) to be 100 for MNIST and CIFAR-10 and 1024 for the ImageNet data.

The workflow of `SparC-IB` overlaps with the standard VIB when encoding the mean and sigma of the full dimension of the latent variable. Furthermore, `SparC-IB` encodes categorical probabilities and then draws samples from a categorical distribution. Unlike the reparameterization trick (Kingma and Welling (2013)) for Gaussian variables, there does not exist a differentiable transformation from categorical probabilities to the samples. Therefore, we use the Gumbel-Softmax approximation (Maddison et al. (2016), Jang et al. (2016)) to draw categorical samples. We apply the transformation in Eq. 4 to the samples and take the element-wise product with the Gaussian samples before passing it to the decoder. Note that there exists other differentiable reparameterization of the discrete samples, e.g., the Gapped Straight-Through (GST) estimator Fan et al. (2022). However, in our experiments, the use of the Gumbel-Softmax approximation has led to a lower loss value than that of the GST.

Fitting deep learning models involves several key hyperparameters. In Table A1, we provide the necessary hyperparameters for training and evaluation of all fitted models in three data sets.

Hyperparameters	MNIST	CIFAR-10	ImageNet
Train set size	60,000	50,000	128,1167
Validation set size	10,000	10,000	50,000
# epochs	100	400	200
Training batch size	128	100	2000
Optimizer	Adam	SGD	Adam
Initial learning rate	1e-4	0.1	1e-4
Learning rate drop	0.6	0.1	0.97
Learning rate drop steps	10 epochs	100 epochs	2 epochs
Weight decay	Not used	5e-4	Not used

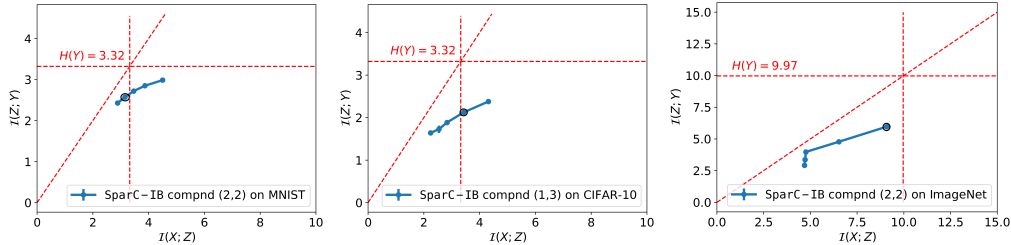Table A1: Hyperparameter settings used to model the three data sets.

Figure A1: Information curve on MNIST, CIFAR-10, and ImageNet.

A.2 DATA AUGMENTATION

For the CIFAR-10 data, we augmented the training data using random transformations. We pad each training set image by 4 pixels on all sides and crop at a random location to return an original-sized image. We flip each training set image horizontally with probability 0.5. Furthermore, we normalize each training and validation set image with mean = $(0.4914, 0.4822, 0.4465)$ and standard deviation = $(0.2023, 0.1994, 0.2010)$.

A.3 CONVERGENCE CHECKS

For CIFAR-10, we observed overfitting after 100 epochs for the Drop-VIB model, where the validation loss started to increase. Therefore, we saved the model at epoch 100 for our robustness analysis. For other models in our experiments on MNIST and CIFAR-10, we observed the convergence of train and validation loss, and we have considered models at the final epoch as the final models. For ImageNet, we saved the model with the lowest validation loss as our final model for all the methods.

A.4 EVALUATION METRICS

We calculated three evaluation metrics for each method in each scenario: test error, log-likelihood, and Brier score. For prediction in all in- and out-of-distribution scenarios, following Fischer and Alemi (2020), we have used the mean latent space $\mathbb{E}(Z|X)$ as input to the decoder for all methods. Note that for SparC-IB we calculate the marginal expectation by following, $\mathbb{E}(Z|X) = \mathbb{E}_{q(d|X)}\mathbb{E}_{q(Z|d,X)}(Z) \hat{=} \frac{1}{J}\sum_{j=1}^{J}\mathbb{E}_{q(Z|d=d_j,X)}(Z)$, where d_j is a sample from $q(d|X)$. In our experiments, we have fixed $J = 10$.

A.5 SOFTWARE AND HARDWARE

We have forked the code base `https://github.com/burklight/convex-IB-Lagrangian-PyTorch.git` that implements the Convex-IB method (Rodríguez Gálvez et al. (2020)) using PyTorch. The code to run the models used in the experiments can be found in the following repository `https://github.com/AnirbanSamaddar/SparC-IB`. For modeling MNIST, we used NVIDIA V100 GPUs and for CIFAR-10 and ImageNet experiments, we used NVIDIA A100 GPUs.

A.6 INFORMATION CURVE: SELECTION OF β

In the IB Lagrangian, the Lagrange multiplier β controls the trade-off between two MI terms. By optimizing the IB objective for different values of β, we can explore the information curve, which is the plot of $(\mathrm{MI}(X; Z), \mathrm{MI}(Z; Y))$ in the 2-d plane. Fig. A1 shows the information curve on the validation set for the models selected for MNIST, CIFAR-10 and ImageNet for the robustness studies in the main article. For the fixed K VIB models, the information curves are similar to Fig. A1. *Minimum necessary information* (Fischer (2020)) is a point in the information plane where $\mathrm{MI}(X; Z) = \mathrm{MI}(Z; Y) = \mathrm{H}(Y)$, where the entropy is indicated by $\mathrm{H}(Y)$. For classification tasks, where labels are deterministic given the images, the entropy $\mathrm{H}(Y) = \log_2 n_c$, where n_c is the number of classes. Therefore, for MNIST and CIFAR-10 $\mathrm{H}(Y) = \log_2 10 \sim 3.32$ and for ImageNet $\mathrm{H}(Y) = \log_2 10 \sim 9.97$. Therefore, we choose $\beta \sim 0.08$ for MNIST, $\beta \sim 0.04$ for CIFAR-10, and $\beta \sim 0.02$ for ImageNet, which gives us the closest proximity to MNI. The points are circled in Fig. A1.

A.7 SELECTION OF PRIOR PARAMETERS FOR SparC-IB

Fig. A2 shows the probabilities of the prior dimension considered for modeling the three data sets. These prior probabilities are from the compound distribution (Eq. 6) with $K = 100$ (left figure) and $K = 1024$ (right figure).
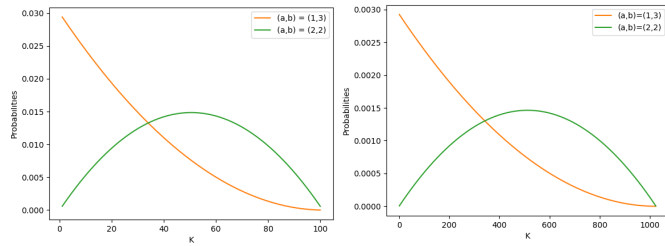
Figure A2: Plot of prior dimension probabilities for choices of (a, b).

A.8 PERFORMANCE BASED ON OTHER EVALUATION METRICS ON *OUT-OF-DISTRIBUTION* DATA

In the main paper, we have presented the robustness results for the three data sets in terms of the log-likelihood. Here we present the robustness results in terms of the test set error and the Brier score for all the methods across the three data sets.

Fig. A3 (a)-(e) show the results for the *out-of-distribution* MNIST test data. From the figures, we observe that in all the scenarios `SparC-IB` performs as well as the best performing model. We further observe that the separation between the models in terms of the test error and the Brier score is less than the log-likelihood which is our chosen metric in the main paper. Similar results for CIFAR10 and ImageNet have been presented in Fig. A4 and Fig. A5 respectively. For CIFAR10 (Fig. A4), we observe similar behavior in terms of the test error and the Brier score as for MNIST. For ImageNet (Fig. A5), we observe that `SparC-IB` is doing better in terms of test error in the L_∞ attacks. However, Drop-VIB seems to be performing better than other approaches for the white-box attacks. This behavior has been discussed in the main paper (Sec. 4.2.1).
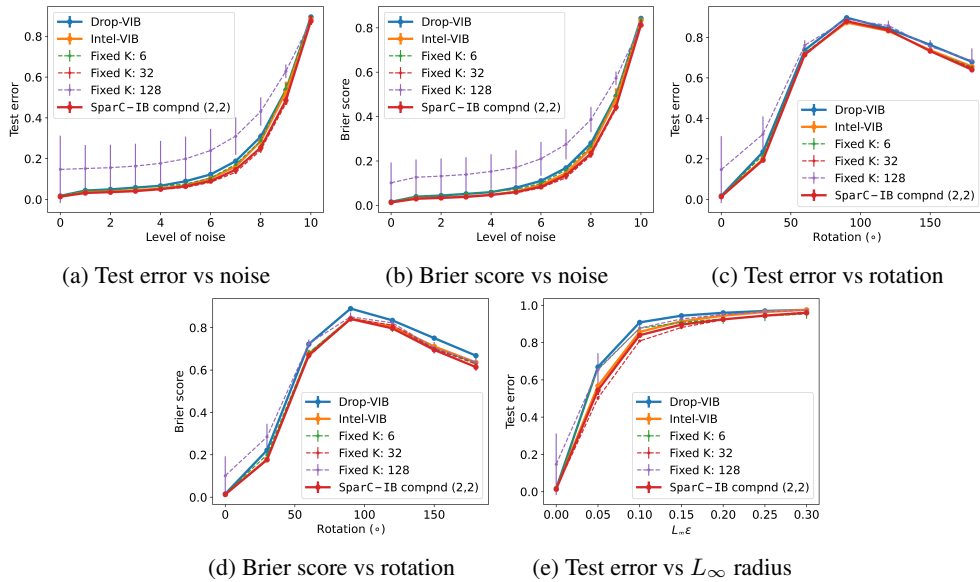

(a) Test error vs noise (b) Brier score vs noise (c) Test error vs rotation

(d) Brier score vs rotation (e) Test error vs L_∞ radius

Figure A3: Out-of-distribution performance in terms of the test error, and the Brier score on MNIST. We observe that `SparC-IB` approach with compound strategy and $(a, b) = (2, 2)$ (red line) performs as well as the best-performing model in all the cases.

A.9 ANALYSIS OF THE LATENT SPACE

A.9.1 DIMENSION DISTRIBUTION MODE PLOT ACROSS SEEDS

Fig. A6 shows the mode plot for `SparC-IB` compound (2,2) across 3 seeds on MNIST data. The overall range of dimensions remains unchanged across the 3 seeds; however, we observe that each digit prefers a different dimension of the latent space.
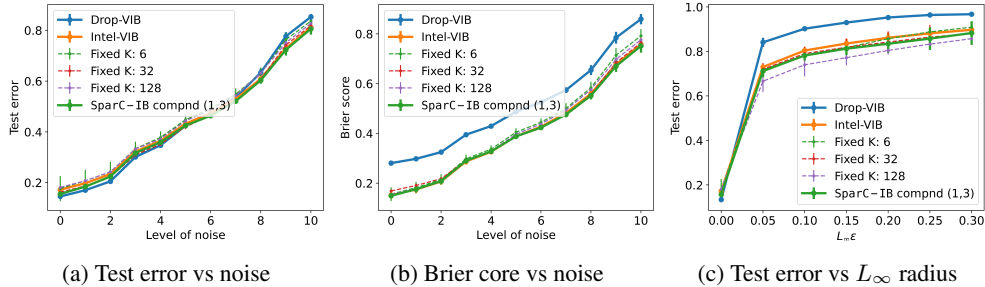
(a) Test error vs noise   (b) Brier core vs noise   (c) Test error vs L_∞ radius

Figure A4: Out-of-distribution performance in terms of the test error, and the Brier score on CIFAR-10. We observe that `SparC-IB` approach with compound strategy and $(a, b) = (1, 3)$ (green line) performs as good as the best performing model in all the cases.

(a) Test error vs noise   (b) Brier score vs noise   (c) Test error vs L_∞ radius
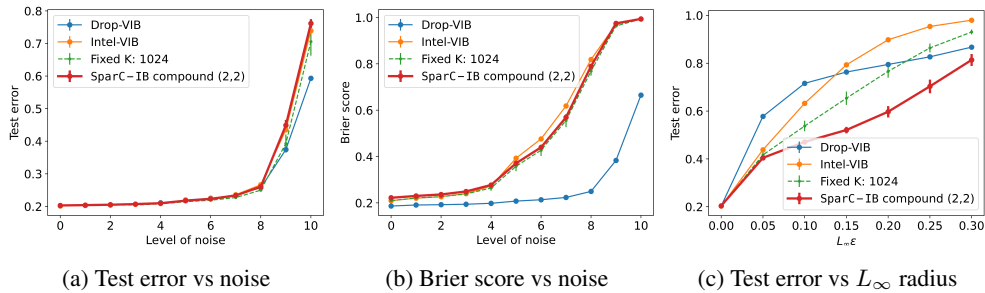
Figure A5: Out-of-distribution performance in terms of the test error, and the Brier score on ImageNet. In the white-noise scenario, the drop-VIB performs better than our approach possibly due to learning more information about X. However, we observe that `SparC-IB` approach *outperforms other models in black-box attacks*.
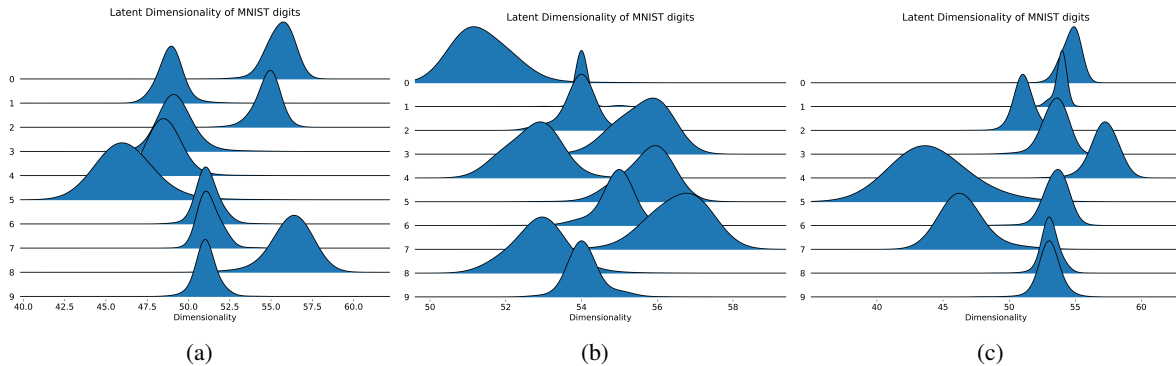

(a)   (b)   (c)

Figure A6: Mode plot for `SparC-IB` compound $(a, b) = (2, 2)$ across the 3 seeds on MNIST. We observe separation of posterior modes between the MNIST digits for all 3 seeds.

A.9.2 THE DIMENSION DISTRIBUTION MODE PLOT FOR CIFAR-10 AND IMAGENET

Fig. A7 shows the dimension distribution mode plot across classes of CIFAR-10 and ImageNet. We show these plots for the `SparC-IB` compound (1,3) in CIFAR-10 and the `SparC-IB` compound (2,2) in ImageNet. In both data sets, we observe that each class prefers a different latent dimension (especially on ImageNet).

A.9.3 INFORMATION CONTENT PLOT FOR CIFAR-10 AND IMAGENET

Fig. A8 shows the estimated $\mathrm{MI}(Z; Y)$ (expression provided in Sec. 4.3.2) against the increasing dimension of the latent space for CIFAR-10 and ImageNet. In CIFAR-10, `SparC-IB` provides the most compact representation among the other models in which the information plateaus within the first dimensions (~ 5) of the latent space. For ImageNet, we observe that the information plateaus around dimension 500 which is smaller than the fixed-dimensional VIB and Intel-VIB models but higher than the Drop-VIB model. In addition, we note that the behavior of the estimated $\mathrm{MI}(Z; Y)$ as a function of dimension is much smoother than those of the other two data sets. The reason for such behavior is perhaps the complexity

(a) Dimension mode plot for CIFAR-10   (b) Dimension mode plot for ImageNet.
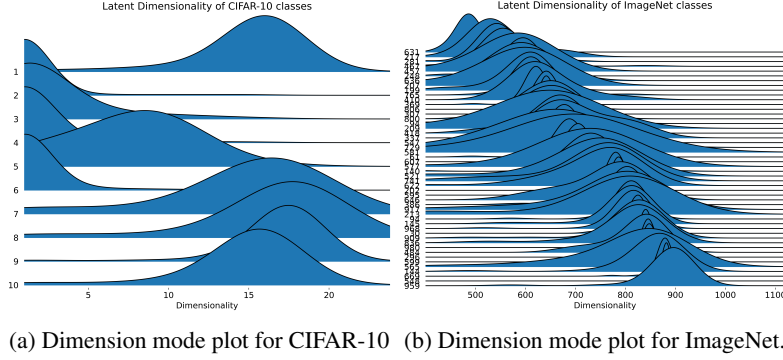
Figure A7: Plot of the posterior modes of the dimension variable for (a) CIFAR-10 and (b) 50 randomly chosen classes of ImageNet. Both plot show separation between the latent dimension of the classes chosen by `SparC-IB`.

in the ImageNet data, where it requires a high-dimensional latent space to encode the necessary information of X where each dimension's contribution is small.

(a) Information content vs dimension plot for CIFAR-10

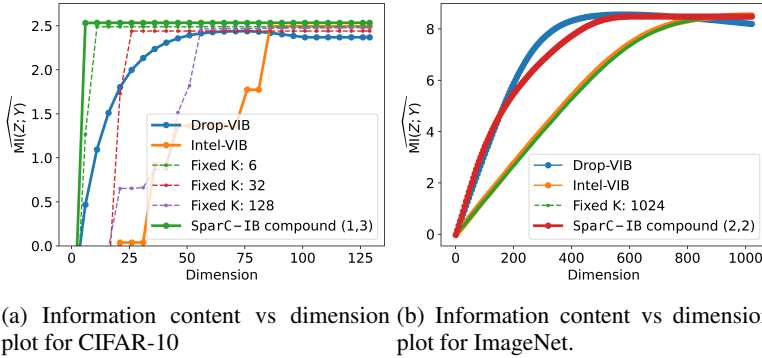(b) Information content vs dimension plot for ImageNet.

Figure A8: Plot of the information content vs dimension of the mean of the encoder for (a) CIFAR-10 and (b) ImageNet. Both plot show `SparC-IB` encodes the maximum information in a smaller dimensional latent space than other models.

A.9.4 CALCULATING PIXEL-WISE IMPORTANCE SCORES FOR LATENT DIMENSIONS VISUALIZATION ON MNIST

We have used the Captum package Kokhlikyan et al. (2020) to calculate the pixel importance scores to visualize the latent space (Sec. 4.3.3) for MNIST. Given an input image x and a baseline image x', the importance score for the i th pixel on the d th dimension of the mean of the latent space is calculated using the following expression.

$$\text{Importance Score}_i^d(x) = (x_i - x'_i) \int_{\alpha=0}^{1} \frac{\partial \mu_d(\alpha x + (1-\alpha)x')}{\partial x_i} d\alpha$$

In the above expression, $\mu(.)$ is the mean vector of the latent space and $\mu_d(.)$ represents its d-th coordinate. We used a blank image where every pixel value is 0 as a baseline x'. We can interpret the score as the sensitivity of the dimensions of $\mu(x)$ to a small change in each pixel integrated on the images that fall on the line given by $\alpha x + (1-\alpha)x'$.