Scalable Training with Information Bottleneck Objectives

Andreas Kirsch [1] **Clare Lyle** [1] **Yarin Gal** [1]

Abstract

The Information Bottleneck principle offers both a mechanism to explain how deep neural networks train and generalize, as well as a regularized objective with which to train models, with multiple competing objectives proposed in the literature. Moreover, the information-theoretic quantities used in these objectives are difficult to compute for large deep neural networks, often relying on density estimation using generative models. This, in turn, limits their use as a training objective. In this work, we review these quantities, compare and unify previously proposed objectives and relate them to surrogate objectives more friendly to optimization without relying on cumbersome tools such as density estimation. We find that these surrogate objectives allow us to apply the information bottleneck to modern neural network architectures with stochastic latent representations. We demonstrate our insights on MNIST and CIFAR10 with modern neural network architectures.

1. Introduction

The Information Bottleneck (IB) principle, introduced by Tishby et al. (2000), proposes that training and generalization in deep neural networks (DNNs) can be explained by information-theoretic principles (Tishby & Zaslavsky, 2015; Shwartz-Ziv & Tishby, 2017; Achille & Soatto, 2018a). This is appealing as the success of DNNs remains largely unexplained by tools from computational learning theory (Zhang et al., 2016; Bengio et al., 2009). However, training with IB objectives presents a computational challenge as the mutual information terms involved are intractable for the complex distributions induced by neural networks.

In this paper, we analyze information quantities and relate them to surrogate objectives for the IB principle which are more friendly to optimization, showing that complex or intractable IB objectives can be replaced with simple, easy-to-compute surrogates that produce similar performance and similar behaviour of information quantities over training. We expand on the findings of Alemi et al. (2016) in their variational IB approximation and conclude that *this upper bound is equal to the commonly-used cross-entropy loss* [1] *under Dropout regularization*. We further examine pathologies of differential entropies that hinder optimization and show how adding Gaussian noise can force differential entropies to become non-negative, which produces a surrogate regularizer. Altogether this leads to simple and tractable surrogate IB objectives such as the following:

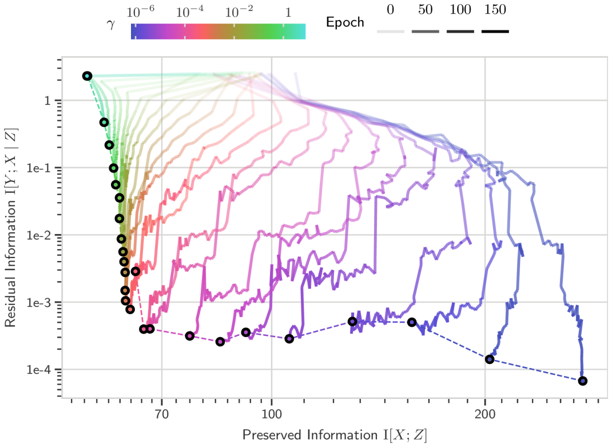


Figure 1. Training trajectories for the surrogate objective $\min_\theta \mathrm{H}_\theta[\mathrm{Y} \mid \mathrm{Z}] + \gamma \mathbb{E}\, \|\mathrm{Z}\|^2$ *from* (9) *with a ResNet18 model on CIFAR10.* The trajectories are colored by their respective γ; their transparency changes by epoch. Compression (Preserved Information ↓) trades-off with performance (Residual Information ↓). See section 4 for more details.

$$\min_{\theta} \mathbb{E}_{\substack{x,y \sim \hat{\mathrm{p}}(x,y), \epsilon \sim \mathcal{N} \\ \eta \sim \text{dropout mask}}} \left[-\log \mathrm{p}(\hat{\mathrm{Y}} = y \mid z = f_\theta(x; \eta) + \epsilon) + \gamma \, \| f_\theta(x; \eta) + \epsilon \|_2^2 \right]. \tag{1}$$

We finally validate our insights qualitatively and quantitatively on MNIST and CIFAR10, and shows that with objectives similar to equation (1) we obtain information plane plots (as in figure 1) similar to those predicted by Tishby & Zaslavsky (2015).

[1] OATML, Department of Computer Science University of Oxford, Oxford, UK. Correspondence to: Andreas Kirsch <andreas.kirsch@cs.ox.ac.uk>.

Presented at the ICML 2020 Workshop on Uncertainty and Robustness in Deep Learning. Copyright 2020 by the author(s).

[1] This connection was assumed without proof by Achille & Soatto (2018a;b).

2. IB Objectives and Their Limitations

2.1. Stochastic Neural Networks

We assume the following probabilistic model given a neural network with parameters θ, where X, Y are the data inputs and labels, $Z = f_\theta(X) + \epsilon$ denotes the neural network's stochastic latent representation with distribution $p_\theta(z|x)$, and $\hat{Y}$ is the prediction obtained from Z, and ϵ denotes zero-entropy noise (which will be introduced in section 3.2).

$$Y \leftarrow X \rightarrow Z \rightarrow \hat{Y} \tag{2}$$

We will use $H[\cdot]$ for the entropy of a random variable, $I[\cdot;\cdot]$ for the mutual information, and let $h(\cdot) = -\ln(\cdot)$. We will further be interested in two cross-entropy loss terms for the neural network predictions: $\mathrm{H}_\theta[\mathrm{Y} \mid \mathrm{Z}]$ to denote the cross-entropy between the prediction $\hat{\mathrm{Y}}$ and the target Y (given the latent Z), and letting $\mathrm{H}_\theta[\mathrm{Y} \mid \mathrm{X}]$ denote the expected cross-entropy loss after marginalizing over the stochastic latent representation. This is similar to notation by Xu et al. (2020). Incorporating noise into the latent representation is known to improve generalization and robustness, and to provide uncertainty estimates (Srivastava et al., 2014; Gal & Ghahramani, 2016).

$$\begin{aligned}\mathrm{H}_\theta[\mathrm{Y} \mid \mathrm{Z}] &:= \mathrm{H}(\mathrm{p}(y \mid z) \,\|\, \mathrm{p}_\theta(\hat{\mathrm{Y}} = y \mid z)) \\ &= \mathbb{E}_{\hat{\mathrm{p}}(x,y)}\, \mathbb{E}_{\mathrm{p}_\theta(z|x)}\, h\left(\mathrm{p}_\theta(\hat{\mathrm{Y}} = y \mid z)\right) \\ &\geq \mathbb{E}_{\hat{\mathrm{p}}(x,y)}\, h\left(\mathbb{E}_{\mathrm{p}_\theta(z|x)}\, \mathrm{p}_\theta(\hat{\mathrm{Y}} = y \mid z)\right) =: \mathrm{H}_\theta[\mathrm{Y} \mid \mathrm{X}]\end{aligned}$$

2.2. Overview of IB Objectives

In their canonical work, Tishby et al. (2000) present the following **Information Bottleneck** objective

$$\min \mathrm{I}[\mathrm{X};\mathrm{Z}] - \beta \mathrm{I}[\mathrm{Y};\mathrm{Z}] \tag{3}$$

and further provide an optimal algorithm for the tabular case, when X, Y and Z are all categorical. The IB principle suggests that learning consists of two competing objectives: maximizing the mutual information between the latent representation and the label to promote accuracy, while at the same time minimizing the mutual information between the latent representation and the input to promote generalization.

Following this principle, many variations of IB objectives have been proposed (Alemi et al., 2016; Strouse & Schwab, 2017; Fisher, 2019; Gondek & Hofmann, 2003; Achille & Soatto, 2018a), which, in supervised learning, have been demonstrated to benefit robustness to adversarial attacks (Alemi et al., 2016; Fisher, 2019) and generalization and regularization against overfitting to random labels (Fisher, 2019). However, whether the benefits of training with IB *objectives* are due to the IB *principle*, or some other unrelated mechanism, remains unclear (Saxe et al., 2019; Amjad & Geiger, 2019; Tschannen et al., 2019)—although recent work has also tied the principle to successful results in both unsupervised and self-supervised learning (Oord et al., 2018; Belghazi et al., 2018; Zhang et al., 2018, among others).

The **Deterministic Information Bottleneck** (DIB) presents a variation on the standard IB objective. Via the observation that $\mathrm{I}[\mathrm{X};\mathrm{Z}] = \mathrm{H}[\mathrm{Z}] - \mathrm{H}[\mathrm{Z} \mid \mathrm{X}]$, and that $\mathrm{H}[\mathrm{Z} \mid \mathrm{X}] = 0$ when Z is a deterministic function of X (when Z is categorical), Strouse & Schwab (2017) introduce the *deterministic* information bottleneck objective (DIB)

$$\min \mathrm{H}[\mathrm{Z}] - \beta \mathrm{I}[\mathrm{Y};\mathrm{Z}] \tag{4}$$

The DIB objective induces subtly different behaviour in the latent representation, but its practical implementation faces similar hurdles to IB.

In contrast, the **Deep Variational Information Bottleneck** (DVIB) directly addresses the challenge of estimating the mutual information. Alemi et al. (2016) rewrite the terms in the bottleneck as maximization problem "max $\mathrm{I}[\mathrm{Y};\mathrm{Z}] - \beta \mathrm{I}[\mathrm{X};\mathrm{Z}]$", and compute a variational lower bound on this objective, using a unit Gaussian as prior $r(z)$ on the distribution of latent representations.

$$\min \mathbb{E}_{p_\theta(z|x_n)}[-\log q_\theta(\hat{Y} = y|z)] - \beta \mathrm{KL}(p_\theta(z|x_n)||r(z)). \tag{5}$$

In principle, the distributions q_θ and p_θ could be given by arbitrary parameterizations and function approximators. In practice, the implementation of DVIB presented by Alemi et al. (2016) constructs p_θ as a multivariate Gaussian with parameterized mean and parameterized diagonal covariance using a neural network followed by a linear decoder feeding into a softmax to yield q_θ. The requirement for the latent distribution p_θ to have a closed-form Kullback-Leibler divergence with respect to the prior $r(z)$ in this variational lower bound limits the applicability of the DVIB objective. A number of other variations also exist in the literature (Fisher, 2019, among others).

2.3. Entropy Estimation for Continuous Variables

One of the principal challenges in training with IB objectives is the computation of the mutual information quantities required. Because neural network representations are continuous (modulo floating point precision), computing the mutual information is equivalent to estimating differential entropies (via e.g. $\mathrm{I}[X;Z] = \mathrm{H}[Z] - \mathrm{H}[Z \mid X]$). Differential entropies have a number of undesirable properties, and chief among them is that they are unbounded from below. This means that in principle a neural network could minimize $\mathrm{H}[Z]$ by scaling the latent representation Z to be arbitrarily close to zero, thus obtaining monotonically 'improving' and unbounded objective values despite not meaningfully changing the representation.

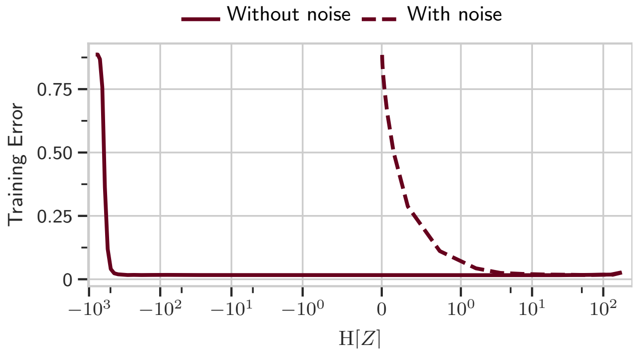


Figure 2. Decreasing the entropy of a noise-free latent does not affect the training error (conceptual toy experiment). Floating-point issues will start affecting it negatively eventually. When adding zero-entropy noise, the training error increases as the entropy approaches zero.

While progress has been made in developing mutual information estimators for DNNs (Poole et al., 2019; Belghazi et al., 2018; Noshad et al., 2019; McAllester & Stratos, 2018; Kraskov et al., 2004), current methods still face many limitations when concerned with high-dimensional random variables (McAllester & Stratos, 2018) and rely on complex estimators or generative models. This presents a challenge to training with IB objectives, and motivates the next section, in which we present our proposed surrogate objective.

3. Surrogate Objectives

Although the mutual information terms in the IB objective can be expensive to compute, we observe that we can re-express the IB and DIB objectives using four entropy terms, each of which can be bounded in a relatively straightforward manner.

Observation 1. *For IB, we obtain*

$$\arg\min \mathrm{I}[\mathrm{X};\mathrm{Z}] - \beta \mathrm{I}[\mathrm{Y};\mathrm{Z}] = \arg\min \mathrm{H}[\mathrm{Y} \mid \mathrm{Z}] + \beta' \underbrace{\mathrm{I}[\mathrm{X};\mathrm{Z} \mid \mathrm{Y}]}_{=\mathrm{H}[\mathrm{Z}|\mathrm{Y}]-\mathrm{H}[\mathrm{Z}|\mathrm{X}]},$$

and, for DIB,

$$\begin{aligned}\arg\min \mathrm{H}[\mathrm{Z}] - \beta \mathrm{I}[\mathrm{Y};\mathrm{Z}] &= \arg\min \mathrm{H}[\mathrm{Y} \mid \mathrm{Z}] + \beta' \mathrm{H}[\mathrm{Z} \mid \mathrm{Y}] \\ &= \arg\min \mathrm{H}[\mathrm{Y} \mid \mathrm{Z}] + \beta'' \mathrm{H}[\mathrm{Z}]\end{aligned}$$

with $\beta' := \frac{1}{\beta-1} \in [0, \infty)$ *and* $\beta'' := \frac{1}{\beta} \in [0, 1)$. *The derivation can be found in section C.3.*

In this section, we will show how to bound these two terms in order to obtain a surrogate objective.

3.1. Bounding $H[Y|Z]$

The first step in our surrogate objective is to bound H[Y | Z]. We obtain this bound in a similar manner as the DVIB objective is obtained, by observing that the cross-entropy between $\hat{Y}$ and Y yields an upper bound on $H[Y|Z]$.

Observation 2. *The Decoder Cross-Entropy provides an upper bound on the Decoder Uncertainty:*

$$\mathrm{H}[\mathrm{Y} \mid \mathrm{Z}] \leq \mathrm{H}[\mathrm{Y} \mid \mathrm{Z}] + D_{\mathrm{KL}}(\mathrm{p}(y \mid z) \parallel \mathrm{p}_\theta(\hat{y} \mid z)) = \mathrm{H}_\theta[\mathrm{Y} \mid \mathrm{Z}], .$$

See section D.2 for a derivation.

3.2. Bounding the Regularization Term

It is not generally possible to compute H[Z | Y] exactly for continuous latent representations Z, but we can derive an upper bound. First, we note that in order to obtain a meaningful regularization term, it is necessary to add noise to the latent representation. Specifically, we add *zero-entropy noise* which we define as $\epsilon \sim \mathcal{N}(0, \frac{1}{2\pi e} I_k)$, such that $\mathrm{H}[\epsilon] = 0$. This also solves the problems described in section 2.3 and is visualized in figure 2.

Observation 3. *After adding zero-entropy noise, the inequality* $\mathrm{I}[\mathrm{X};\mathrm{Z} \mid \mathrm{Y}] \leq \mathrm{H}[\mathrm{Z} \mid \mathrm{Y}] \leq \mathrm{H}[\mathrm{Z}]$ *also holds in the continuous case, and we can minimize* $\mathrm{I}[\mathrm{X};\mathrm{Z} \mid \mathrm{Y}]$ *in the IB objective by minimizing* $\mathrm{H}[\mathrm{Z} \mid \mathrm{Y}]$ *or* $\mathrm{H}[\mathrm{Z}]$, *similarly to the DIB objective. We present a formal proof in section F.1.*

Having shown that the inequality from observation 1 holds for continuous latent representations, we can then consider estimators of $H[Z|Y]$ and $H[Z]$. To do so, we take advantage of the fact that the maximum-entropy distribution for a given covariance matrix Σ is a Gaussian with the same covariance.

Observation 4. *The Reverse Decoder Uncertainty can be approximately bounded using the empirical variance* $\widehat{\mathrm{Var}}[\mathrm{Z}_i \mid y]$:

$$\mathrm{H}[\mathrm{Z} \mid \mathrm{Y}] \leq \mathbb{E}_{\hat{\mathrm{p}}(y)} \sum_i \tfrac{1}{2} \ln(2\pi e \, \mathrm{Var}[\mathrm{Z}_i \mid y]) \tag{6}$$

$$\approx \mathbb{E}_{\hat{\mathrm{p}}(y)} \sum_i \tfrac{1}{2} \ln(2\pi e \, \widehat{\mathrm{Var}}[\mathrm{Z}_i \mid y]), \tag{7}$$

where Z_i *are the individual components of* Z. H[Z] *can be bound similarly. More generally, we can create an even looser upper bound by bounding the mean squared norm of the latent:*

$$\mathbb{E} \, \|\mathrm{Z}\|^2 \leq C' \Rightarrow \mathrm{H}[\mathrm{Z} \mid \mathrm{Y}] \leq \mathrm{H}[\mathrm{Z}] \leq C, \tag{8}$$

with $C' := \frac{k e^{2C/k}}{2\pi e}$ *for* $\mathrm{Z} \in \mathbb{R}^k$. *See section F.2 for a formal proof.*

We can thus replace the regularization term with $\mathbb{E} \, \|Z\|^2$ directly. Constraining the mean squared norm can also be seen as constraining the average power of a communication channel (MacKay, 2003).

Observation 5. *When we add zero-entropy noise to the latent* Z *and, for example, estimate the Decoder Cross-Entropy* $\mathrm{H}_\theta[\mathrm{Y} \mid \mathrm{Z}] = \mathrm{H}(\mathrm{p}(y \mid z) \parallel \mathrm{p}_\theta(\hat{\mathrm{Y}} = y \mid z))$, *using the regular*

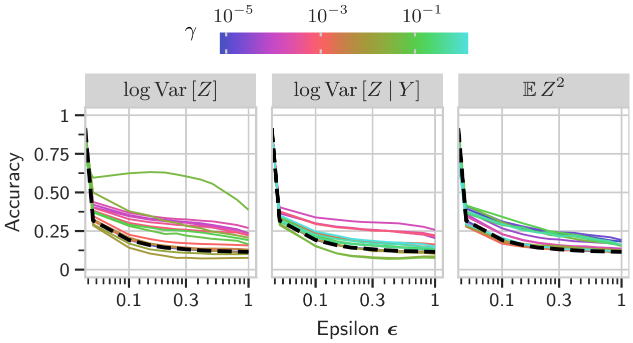


Figure 3. Adversarial robustness of models trained with surrogate objectives. Models are trained on CIFAR-10 using the surrogate objectives described, then evaluated on their robustness to FGSM attacks of varying ϵ values. We see that models trained with surrogate IB objectives (shown in coloured lines) largely see improved robustness over models trained only to minimize the cross-entropy training objective (shown in black).

cross-entropy while using a single stochastic sample during training, we obtain as surrogate objective for $\mathbb{E}\,\|Z\|^2$*:*

$$\min \mathrm{H}_\theta[\mathrm{Y} \mid \mathrm{Z}] + \gamma \mathbb{E}\,\|\mathrm{Z}\|^2; \tag{9}$$

for $\log \mathrm{Var}[\mathrm{Z} \mid \mathrm{Y}]$*:*

$$\min \mathrm{H}_\theta[\mathrm{Y} \mid \mathrm{Z}] + \gamma\, \mathbb{E}_{\hat{\mathrm{p}}(y)} \sum_i \frac{1}{2} \ln(2\pi e\ \widehat{\mathrm{Var}}[\mathrm{Z}_i \mid y]); \tag{10}$$

and for $\log \mathrm{Var}[\mathrm{Z}]$*:*

$$\min \mathrm{H}_\theta[\mathrm{Y} \mid \mathrm{Z}] + \gamma \sum_i \frac{1}{2} \ln(2\pi e\ \widehat{\mathrm{Var}}[\mathrm{Z}_i]). \tag{11}$$

4. Experiments

IB objectives have been shown to lead to improved adversarial robustness, and to induce training trajectories that demonstrate trade-offs between fitting and compressing the data. However, previous work has considered toy synthetic datasets or the MNIST digit classification dataset, neither of which guarantees that these results are applicable to the more complex tasks to which deep neural networks are often applied. In this section, we empirically demonstrate that our proposed surrogate objectives produce qualitatively similar behaviour on the CIFAR-10 dataset to that seen by other IB objectives on simpler datasets. For details about our experiment setup, DNN architectures, hyperparameters and more insights, see section G.

Robustness to adversarial attacks Alemi et al. (2016) observe that their DVIB objective leads to improved adversarial robustness over standard training objectives. We perform a similar evaluation to see whether our surrogate objectives also see improved robustness. We train a fully-connected residual network on CIFAR-10, incorporating stochasticity into the latent representation via Dropout and DropConnect (Srivastava et al., 2014; Wan et al., 2013). After training, we evaluate the models on adversarially perturbed images using the Fast Gradient Sign Method (Szegedy et al., 2013) for varying levels of the perturbation magnitude parameter ϵ, comparing to the same model trained with a regular cross-entropy loss (black dashed-line).

We consider each of the three surrogate objectives proposed in Equations 9, 10, and 11 for a range of regularization coefficients γ. We also consider training with $\gamma = 0$. For all surrogate objectives, we find that the optimal regularization coefficient γ yields significantly more robust models while obtaining similar test accuracy on the unperturbed data.

Surrogate objectives & information plane plots To compare the different surrogate regularizers on CIFAR10, we use a deterministic ResNet18 model as an encoder with a logistic regression layer as a decoder. We use a deterministic ResNet18 model because we can estimate $\mathrm{I}[\mathrm{X};\mathrm{Z}] = \mathrm{H}[\mathrm{Z}]$ as $\mathrm{H}[\mathrm{Z} \mid \mathrm{X}] = 0$ (with injected zero-entropy noise) using the entropy estimator presented from Kraskov et al. (2004). We measure $\mathrm{I}[\mathrm{X};\mathrm{Y} \mid \mathrm{Z}] = \mathrm{H}[\mathrm{Y} \mid \mathrm{Z}]$, as $\mathrm{H}[\mathrm{Y} \mid \mathrm{X}] = 0$ for CIFAR-10, on the training set by using its upper bound $\mathrm{H}[\mathrm{Y} \mid \mathrm{Z}]$ as approximiation (Xu et al., 2020; McAllester & Stratos, 2018). We train with the surrogate objectives from 3.2 for various γ, chosen in logspace from different ranges to compensate for their relationship to β as noted in section 3.2: for $\log \mathrm{Var}[\mathrm{Z}]$, $\gamma \in [10^{-5}, 1]$; for $\log \mathrm{Var}[\mathrm{Z} \mid \mathrm{Y}]$, $\gamma \in [10^{-5}, 10]$; and for $\mathbb{E}\,\|\mathrm{Z}\|^2$, by trial and error, $\gamma \in [10^{-6}, 10]$.

Figure 1 shows an information plane plot for regularizing with $\mathbb{E}\,\|\mathrm{Z}\|^2$ for different γ over different epochs for the training set. Similar to Shwartz-Ziv & Tishby (2017), we observe that there is an initial expansion phase followed by compression. The jumps in performance (reduction of the Residual Information) are due to drops in the learning rate. Figure G.1 shows the difference between the regularizers more clearly, and figure G.3 shows the training trajectories for all three regularizers.

5. Conclusion

We have demonstrated that by decomposing the mutual information terms in IB objectives into their component entropy terms, one obtains tractable, scalable surrogate objectives. We show that these objectives can capture many of the desirable properties of IB methods while also scaling to problems of interest in deep learning; these surrogate objectives yield similar information plane plots and adversarial robustness properties as IB objectives based on direct mutual information estimation (or a variational bound thereof). Promising future directions include improving the regularizers presented in this paper, and exploring the connection to Bayesian neural networks.

References

Achille, A. and Soatto, S. Emergence of invariance and disentanglement in deep representations. *The Journal of Machine Learning Research*, 19(1):1947–1980, 2018a.

Achille, A. and Soatto, S. Information dropout: Learning optimal representations through noisy computation. *IEEE transactions on pattern analysis and machine intelligence*, 40(12):2897–2905, 2018b.

Alemi, A. A., Fischer, I., Dillon, J. V., and Murphy, K. Deep variational information bottleneck. *arXiv preprint arXiv:1612.00410*, 2016.

Amjad, R. A. and Geiger, B. C. Learning representations for neural network-based classification using the information bottleneck principle. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2019.

Belghazi, M. I., Baratin, A., Rajeshwar, S., Ozair, S., Bengio, Y., Courville, A., and Hjelm, D. Mutual information neural estimation. In *International Conference on Machine Learning*, pp. 531–540, 2018.

Bengio, Y. et al. Learning deep architectures for ai. *Foundations and trends® in Machine Learning*, 2(1):1–127, 2009.

Bercher, J.-F. and Vignat, C. A renyi entropy convolution inequality with application. In *2002 11th European Signal Processing Conference*, pp. 1–4. IEEE, 2002.

Cover, T. M. and Thomas, J. A. *Elements of information theory*. John Wiley & Sons, 2012.

Fisher, I. The Conditional Entropy Bottleneck. *Submission to ICLR 2019, International Conference on Learning Representations*, 2019.

Gal, Y. and Ghahramani, Z. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *international conference on machine learning*, pp. 1050–1059, 2016.

Gondek, D. and Hofmann, T. Conditional information bottleneck clustering. In *3rd ieee international conference on data mining, workshop on clustering large data sets*, pp. 36–42. Citeseer, 2003.

Goodfellow, I. J., Mirza, M., Xiao, D., Courville, A., and Bengio, Y. An empirical investigation of catastrophic forgetting in gradient-based neural networks. *arXiv preprint arXiv:1312.6211*, 2013.

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016a.

He, K., Zhang, X., Ren, S., and Sun, J. Identity mappings in deep residual networks. *Lecture Notes in Computer Science*, pp. 630–645, 2016b.

Houlsby, N., Huszár, F., Ghahramani, Z., and Lengyel, M. Bayesian active learning for classification and preference learning. *arXiv preprint arXiv:1112.5745*, 2011.

Jette, M. A., Yoo, A. B., and Grondona, M. Slurm: Simple linux utility for resource management. In *In Lecture Notes in Computer Science: Proceedings of Job Scheduling Strategies for Parallel Processing (JSSPP) 2003*, pp. 44–60. Springer-Verlag, 2002.

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

Kirsch, A., van Amersfoort, J., and Gal, Y. Batchbald: Efficient and diverse batch acquisition for deep bayesian active learning. In *Advances in Neural Information Processing Systems*, pp. 7024–7035, 2019.

Kraskov, A., Stögbauer, H., and Grassberger, P. Estimating mutual information. *Physical review E*, 69(6):066138, 2004.

Krizhevsky, A., Hinton, G., et al. Learning multiple layers of features from tiny images. 2009.

MacKay, D. J. C. *Information Theory, Inference, and Learning Algorithms*. Cambridge University Press, 2003.

McAllester, D. and Stratos, K. Formal limitations on the measurement of mutual information. *arXiv preprint arXiv:1811.04251*, 2018.

McGill, W. Multivariate information transmission. *Transactions of the IRE Professional Group on Information Theory*, 4(4):93–111, 1954.

Noshad, M., Zeng, Y., and Hero, A. O. Scalable mutual information estimation using dependence graphs. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 2962–2966. IEEE, 2019.

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

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., et al. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems*, pp. 8024–8035, 2019.

Poole, B., Ozair, S., Oord, A. v. d., Alemi, A. A., and Tucker, G. On variational bounds of mutual information. *arXiv preprint arXiv:1905.06922*, 2019.

Saxe, A. M., Bansal, Y., Dapello, J., Advani, M., Kolchinsky, A., Tracey, B. D., and Cox, D. D. On the information bottleneck theory of deep learning. *Journal of Statistical Mechanics: Theory and Experiment*, 2019(12):124020, 2019.

Shamir, O., Sabato, S., and Tishby, N. Learning and generalization with the information bottleneck. *Theoretical Computer Science*, 411(29-30):2696–2711, 2010.

Shannon, C. E. A mathematical theory of communication. *Bell system technical journal*, 27(3):379–423, 1948.

Shwartz-Ziv, R. and Tishby, N. Opening the black box of deep neural networks via information. *arXiv preprint arXiv:1703.00810*, 2017.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1):1929–1958, 2014.

Strouse, D. and Schwab, D. J. The deterministic information bottleneck. *Neural computation*, 29(6):1611–1630, 2017.

Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I., and Fergus, R. Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*, 2013.

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

Tishby, N., Pereira, F. C., and Bialek, W. The information bottleneck method. *arXiv preprint physics/0004057*, 2000.

Tschannen, M., Djolonga, J., Rubenstein, P. K., Gelly, S., and Lucic, M. On mutual information maximization for representation learning. *arXiv preprint arXiv:1907.13625*, 2019.

Wan, L., Zeiler, M., Zhang, S., Le Cun, Y., and Fergus, R. Regularization of neural networks using dropconnect. In *International conference on machine learning*, pp. 1058–1066, 2013.

Xu, Y., Zhao, S., Song, J., Stewart, R., and Ermon, S. A theory of usable information under computational constraints, 2020.

Yeung, R. W. A new outlook on shannon's information measures. *IEEE transactions on information theory*, 37 (3):466–474, 1991.

Zhang, C., Bengio, S., Hardt, M., Recht, B., and Vinyals, O. Understanding deep learning requires rethinking generalization. *arXiv preprint arXiv:1611.03530*, 2016.

Zhang, Y., Xiang, T., Hospedales, T. M., and Lu, H. Deep mutual learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4320–4328, 2018.

A. Information quantities & information diagrams

Here we introduce notation and terminology in greater detail than in the main paper. We review well-known information quantities and provide more details on using information diagrams (Yeung, 1991).

A.1. Information quantities

We denote entropy $\mathrm{H}[\cdot]$, joint entropy $\mathrm{H}[\cdot, \cdot]$, conditional entropy $\mathrm{H}[\cdot \mid \cdot]$, mutual information $\mathrm{I}[\cdot ; \cdot]$ and Shannon's information content $h(\cdot)$ following Cover & Thomas (2012); MacKay (2003); Shannon (1948) :

$$
\begin{aligned}
h(x) &= -\ln x \\
\mathrm{H}[X] &= \mathbb{E}_{\mathrm{p}(x)}\, h(\mathrm{p}(x)) \\
\mathrm{H}[X, Y] &= \mathbb{E}_{\mathrm{p}(x,y)}\, h(\mathrm{p}(x, y)) \\
\mathrm{H}[X \mid Y] &= \mathrm{H}[X, Y] - \mathrm{H}[Y] \\
&= \mathbb{E}_{\mathrm{p}(y)}\, \mathrm{H}[X \mid y] = \mathbb{E}_{\mathrm{p}(x,y)}\, h(\mathrm{p}(x \mid y)) \\
\mathrm{I}[X; Y] &= \mathrm{H}[X] + \mathrm{H}[Y] - \mathrm{H}[X, Y] \\
&= \mathbb{E}_{\mathrm{p}(x,y)}\, h\left(\tfrac{\mathrm{p}(x)\,\mathrm{p}(y)}{\mathrm{p}(x,y)}\right) \\
\mathrm{I}[X; Y \mid Z] &= \mathrm{H}[X \mid Z] + \mathrm{H}[Y \mid Z] - \mathrm{H}[X, Y \mid Z],
\end{aligned}
$$

where X, Y, Z are random variables and x, y, z are outcomes these random variables can take.

We use differential entropies interchangeably with entropies. We can do so because equalities between them hold as can be verified by symbolic expansions. For example,

$$
\begin{aligned}
&\mathrm{H}[X, Y] = \mathrm{H}[X \mid Y] + \mathrm{H}[Y] \\
\Leftrightarrow\ &\mathbb{E}_{\mathrm{p}(x,y)}\, h(\mathrm{p}(x, y)) = \mathbb{E}_{\mathrm{p}(x,y)}\, [h(\mathrm{p}(x \mid y)) + h(\mathrm{p}(y))] = \mathbb{E}_{\mathrm{p}(x,y)}\, [h(\mathrm{p}(x \mid y))] + \mathbb{E}_{\mathrm{p}(y)}\, h(\mathrm{p}(y)),
\end{aligned}
$$

which is valid in both the discrete and continuous case (if the integrals all exist). The question of how to transfer inequalities in the discrete case to the continuous case is dealt with in section 2.3.

We will further require the Kullback-Leibler divergence $D_{\mathrm{KL}}(\cdot \,\|\, \cdot)$ and cross-entropy $\mathrm{H}(\cdot \,\|\, \cdot)$:

$$
\begin{aligned}
\mathrm{H}(\mathrm{p}(x) \,\|\, \mathrm{q}(x)) &= \mathbb{E}_{\mathrm{p}(x)}\, h(\mathrm{q}(x)) \\
D_{\mathrm{KL}}(\mathrm{p}(x) \,\|\, \mathrm{q}(x)) &= \mathbb{E}_{\mathrm{p}(x)}\, h\left(\tfrac{\mathrm{q}(x)}{\mathrm{p}(x)}\right) \\
\mathrm{H}(\mathrm{p}(y \mid x) \,\|\, \mathrm{q}(y \mid x)) &= \mathbb{E}_{\mathrm{p}(x)}\, \mathbb{E}_{\mathrm{p}(y|x)}\, h(\mathrm{q}(y \mid x)) \\
&= \mathbb{E}_{\mathrm{p}(x,y)}\, h(\mathrm{q}(y \mid x)) \\
D_{\mathrm{KL}}(\mathrm{p}(y \mid x) \,\|\, \mathrm{q}(y \mid x)) &= \mathbb{E}_{\mathrm{p}(x,y)}\, h\left(\tfrac{\mathrm{q}(y|x)}{\mathrm{p}(y|x)}\right)
\end{aligned}
$$

A.2. Information diagrams

Information diagrams (I-diagrams), like the one depicted in figure H.1 (or figure H.1 for a bigger version), visualize the relationship between information quantities: Yeung (1991) shows that we can define a signed measure μ^* such that these well-known quantities map to abstract sets and are consistent with set operations.

$$
\begin{aligned}
\mathrm{H}[A] &= \mu^*(A) \\
\mathrm{H}[A_1, \dots, A_n] &= \mu^*(\cup_i A_i) \\
\mathrm{H}[A_1, \dots, A_n \mid B_1, \dots, B_n] &= \mu^*(\cup_i A_i - \cup_i B_i) \\
\mathrm{I}[A_1; \dots; A_n] &= \mu^*(\cap_i A_i) \\
\mathrm{I}[A_1; \dots; A_n \mid B_1, \dots, B_n] &= \mu^*(\cap_i A_i - \cup_i B_i)
\end{aligned}
$$

Note that interaction information (McGill, 1954) follows as canonical generalization of the mutual information to multiple variables from that work, whereas total correlation does not.

In other words, equalities can be read off directly from I-diagrams: an information quantity is the sum of its parts in the corresponding I-diagram. This is similar to Venn diagrams. The sets used in I-diagrams are just abstract symbolic objects, however.

An important distinction between I-diagrams and Venn diagrams is that while we can always read off inequalities in Venn diagrams, this is not true for I-diagrams in general because mutual information terms in more than two variables can be negative. In Venn diagrams, a set is always larger or equal any subset.

However, if we show that all information quantities are non-negative, we can read off inequalities again. We do this for figure H.1 at the end of section 2.1 for categorical Z and expand this to continuous Z in section 2.3. Thus, we can treat the Mickey Mouse I-diagram like a Venn diagram to read off equalities and inequalities.

Nevertheless, caution is warranted sometimes. As the signed measure can be negative, $\mu^*(X \cap Y) = 0$ does *not* imply $X \cap Y = \emptyset$: deducing that a mutual information term is 0 does not imply that one can simply remove the corresponding area in the I-diagram. There could be Z with $\mu^*((X \cap Y) \cap Z) < 0$, such that $\mu^*(X \cap Y) = \mu^*(X \cap Y \cap Z) + \mu^*(X \cap Y - Z) = 0$ but $X \cap Y \neq \emptyset$. This also means that we cannot drop the term from expressions when performing symbolic manipulations. This is of particular importance because a mutual information of zero means two random variables are independent, which might invite one drawing them as disjoint areas.

The only time where one can safely remove an area from the diagram is for *atomic* quantities, which are quantities which reference all the available random variables (Yeung, 1991). For example, when we only have three variables X, Y, Z, $\mathrm{I}[X; Y; Z]$ and $\mathrm{I}[X; Y \mid Z]$ are atomic quantities. We can safely remove atomic quantities from I-diagrams when they are 0 as there are no random variables left to apply that could lead to the problem explored above.

Continuing the example, $0 = \mathrm{I}[X; Y; Z] = \mu^*(X \cap Y \cap Z)$ would imply $X \cap Y \cap Z = \emptyset$, and we could remove it from the diagram without loss of generality. Moreover, atomic $\mathrm{I}[X; Y \mid Z] = \mu^*(X \cap Y - Z) = 0$ then and could be removed from the diagram as well.

We only use I-diagrams for the three variable case, but they supply us with tools to easily come up with equalities and inequalities for information quantities. In the general case with multiple variables, they can be difficult to draw, but for Markov chains they can be of great use.

B. Mickey Mouse I-diagram

B.1. Intuition for the Mickey Mouse information quantities

We base the names of information quantities on existing conventions and come up with sensible extensions. For example, the name $PreservedRelevantInformation$ for $\mathrm{I[Y;Z]}$ was introduced by Tishby & Zaslavsky (2015). It can be seen as the intersection of $\mathrm{I[X;Z]}$ and $\mathrm{I[X;Y]}$ in the I-diagram, and hence we denote $\mathrm{I[X;Z]}$ Preserved Information and $\mathrm{I[X;Y]}$ Relevant Information, which are sensible names as we detail below.

We identify the following six atomic quantities:

Label Uncertainty $\mathrm{H[Y \mid X]}$ quantifies the uncertainty in our labels. If we have multiple labels for the same data sample, it will be > 0. It is 0 otherwise.

Encoding Uncertainty $\mathrm{H[Z \mid X]}$ quantifies the uncertainty in our latent encoding given a sample. When using a Bayesian model with random variable ω for the weights, one can further split this term into $\mathrm{H[Z \mid X]} = \mathrm{I[Z; \omega \mid X]} + \mathrm{H[Z \mid X, \omega]}$, so uncertainty stemming from weight uncertainty and independent noise (Houlsby et al., 2011; Kirsch et al., 2019).

Preserved Relevant Information $\mathrm{I[Y;Z]}$ quantifies information in the latent that is relevant for our task of predicting the labels (Tishby & Zaslavsky, 2015). Intuitively, we want to maximize it for good predictive performance.

Residual Information $\mathrm{I[X;Y \mid Z]}$ quantifies information for the labels that is not captured by the latent (Tishby & Zaslavsky, 2015) but would be useful to be captured.

Redundant Information $\mathrm{I[X;Z \mid Y]}$ quantifies information in the latent that is not needed for predicting the labels[2].

[2] Fisher (2019) uses the term “Residual Information” for this, which conflicts with Tishby & Zaslavsky (2015).

We also identify the following composite information quantities:

Relevant Information $\mathrm{I}[X; Y] = \mathrm{I}[X; Y \mid Z] + \mathrm{I}[Y; Z]$ quantifies the information in the data that is relevant for the labels and which our model needs to capture to be able to predict the labels.

Preserved Information $\mathrm{I}[X; Z] = \mathrm{I}[X; Z \mid Y] + \mathrm{I}[Y; Z]$ quantifies information from the data that is preserved in the latent.

Decoder Uncertainty $\mathrm{H}[Y \mid Z] = \mathrm{I}[X; Y \mid Z] + \mathrm{H}[Y \mid X]$ quantifies the uncertainty about the labels after learning about the latent Z. If $\mathrm{H}[Y \mid Z]$ reaches 0, it means that no additional information is needed to infer the correct label Y from the latent Z: the optimal decoder can be a deterministic mapping. Intuitively, we want to minimize this quantity for good predictive performance.

Reverse Decoder Uncertainty $\mathrm{H}[Z \mid Y] = \mathrm{I}[X; Z \mid Y] + \mathrm{H}[Z \mid X]$ quantifies the uncertainty about the latent Z given the label Y. We can imagine training a new model to predict Z given Y and minimizing $\mathrm{H}[Z \mid Y]$ to 0 would allow for a deterministic decoder from the latent to given the label.

Nuisance[3] $\mathrm{H}[X \mid Y] = \mathrm{H}[X \mid Y, Z] + \mathrm{I}[X; Z]$ quantifies the information in the data that is not relevant for the task (Achille & Soatto, 2018a).

B.2. Definitions & equivalences

The following equalities can be read off from figure H.1. For completeness and to provide a handy reference, we list them explicitly here. They can also be verified using symbolic manipulations and the properties of information quantities.

Equalities for composite quantities:

$$\mathrm{I}[X; Y] = \mathrm{I}[X; Y \mid Z] + \mathrm{I}[Y; Z] \tag{12}$$
$$\mathrm{I}[X; Z] = \mathrm{I}[X; Z \mid Y] + \mathrm{I}[Y; Z] \tag{13}$$
$$\mathrm{H}[Y \mid Z] = \mathrm{I}[X; Y \mid Z] + \mathrm{H}[Y \mid X] \tag{14}$$
$$\mathrm{H}[Z \mid Y] = \mathrm{I}[X; Z \mid Y] + \mathrm{H}[Z \mid X] \tag{15}$$
$$\mathrm{H}[X \mid Y] = \mathrm{H}[X \mid Y, Z] + \mathrm{I}[X; Z] \tag{16}$$

We can combine the atomic quantities into the overall Label Entropy and Encoding Entropy:

$$\mathrm{H}[Y] = \mathrm{H}[Y \mid X] + \mathrm{I}[Y; Z] + \mathrm{I}[X; Y \mid Z] \tag{17}$$
$$\mathrm{H}[Z] = \mathrm{H}[Z \mid X] + \mathrm{I}[Y; Z] + \mathrm{I}[X; Z \mid Y]. \tag{18}$$

We can express the Relevant Information $\mathrm{I}[X; Y]$, Residual Information $\mathrm{I}[X; Y \mid Z]$, Redundant Information $\mathrm{I}[X; Z \mid Y]$ and Preserved Information $\mathrm{I}[X; Z]$ without X on the left-hand side:

$$\mathrm{I}[X; Y] = \mathrm{H}[Y] - \mathrm{H}[Y \mid X], \tag{19}$$
$$\mathrm{I}[X; Z] = \mathrm{H}[Z] - \mathrm{H}[Z \mid X], \tag{20}$$
$$\mathrm{I}[X; Y \mid Z] = \mathrm{H}[Y \mid Z] - \mathrm{H}[Y \mid X], \tag{21}$$
$$\mathrm{I}[X; Z \mid Y] = \mathrm{H}[Z \mid Y] - \mathrm{H}[Z \mid X]. \tag{22}$$

This simplifies estimating these expressions as X is usually much higher-dimensional and irregular than the labels or latent encodings. We also can rewrite the Preserved Relevant Information $\mathrm{I}[Y; Z]$ as:

$$\mathrm{I}[Y; Z] = \mathrm{H}[Y] - \mathrm{H}[Y \mid Z] \tag{23}$$
$$\mathrm{I}[Y; Z] = \mathrm{H}[Z] - \mathrm{H}[Z \mid Y] \tag{24}$$

[3] Not depicted in figure H.1.

C. Information bottleneck & related works

C.1. Goals & motivation

The IB principle from Tishby et al. (2000) can be recast as a generalization of finding minimal sufficient statistics for the labels given the data (Shamir et al., 2010; Tishby & Zaslavsky, 2015; Fisher, 2019): it strives for minimality and sufficiency of the latent Z. Minimality is about minimizing amount of information necessary of X for the task, so minimizing the Preserved Information $\mathrm{I}[\mathrm{X};\mathrm{Z}]$; while sufficiency is about preserving the information to solve the task, so maximizing the Preserved Relevant Information $\mathrm{I}[\mathrm{Y};\mathrm{Z}]$.

From figure H.1, we can read off the definitions of Relevant Information and Preserved Information:

$$\mathrm{I}[\mathrm{X};\mathrm{Y}] = \mathrm{I}[\mathrm{Y};\mathrm{Z}] + \mathrm{I}[\mathrm{X};\mathrm{Y} \mid \mathrm{Z}] \tag{25}$$

$$\mathrm{I}[\mathrm{X};\mathrm{Z}] = \mathrm{I}[\mathrm{Y};\mathrm{Z}] + \mathrm{I}[\mathrm{X};\mathrm{Z} \mid \mathrm{Y}], \tag{26}$$

and see that maximizing the Preserved Relevant Information $\mathrm{I}[\mathrm{Y};\mathrm{Z}]$ is equivalent to minimizing the Residual Information $\mathrm{I}[\mathrm{X};\mathrm{Y} \mid \mathrm{Z}]$, while minimizing the Preserved Information $\mathrm{I}[\mathrm{X};\mathrm{Z}]$ at the same time means minimizing the Redundant Information $\mathrm{I}[\mathrm{X};\mathrm{Z} \mid \mathrm{Y}]$, too, as $\mathrm{I}[\mathrm{X};\mathrm{Y}]$ is constant for the given dataset[4]. Moreover, we also see that the Preserved Relevant Information $\mathrm{I}[\mathrm{Y};\mathrm{Z}]$ is upper-bounded by Relevant Information $\mathrm{I}[\mathrm{X};\mathrm{Y}]$, so to capture all relevant information in our latent, we want $\mathrm{I}[\mathrm{X};\mathrm{Y}] = \mathrm{I}[\mathrm{Y};\mathrm{Z}]$.

Using the diagram, we can also see that minimizing the Residual Information is the same as minimizing the Decoder Uncertainty $\mathrm{H}[\mathrm{Y} \mid \mathrm{Z}]$:

$$\mathrm{I}[\mathrm{X};\mathrm{Y} \mid \mathrm{Z}] = \mathrm{H}[\mathrm{Y} \mid \mathrm{Z}] - \mathrm{H}[\mathrm{Y} \mid \mathrm{X}].$$

Ideally, we also want to minimize the Encoding Uncertainty $\mathrm{H}[\mathrm{Z} \mid \mathrm{X}]$ to find the most deterministic latent encoding Z. Minimizing the Encoding Uncertainty and the Redundant Information $\mathrm{I}[\mathrm{X};\mathrm{Z} \mid \mathrm{Y}]$ together is the same as minimizing the Reverse Decoder Uncertainty $\mathrm{H}[\mathrm{Z} \mid \mathrm{Y}]$.

All in all, we want to minimize both the Decoder Uncertainty $\mathrm{H}[\mathrm{Y} \mid \mathrm{Z}]$ and the Reverse Decoder Uncertainty $\mathrm{H}[\mathrm{Z} \mid \mathrm{Y}]$.

C.2. IB objectives

"The Information Bottleneck Method" (IB)

Tishby et al. (2000) introduce $MI(X;\hat{X}) - \beta MI(\hat{X};Y)$ as optimization objective for the Information Bottleneck. We can relate this to our notation by renaming $\hat{X} = \mathrm{Z}$, such that the objective becomes "$\min \mathrm{I}[\mathrm{X};\mathrm{Z}] - \beta \mathrm{I}[\mathrm{Y};\mathrm{Z}]$". The IB objective minimizes the Preserved Information $\mathrm{I}[\mathrm{X};\mathrm{Z}]$ and trades it off with maximizing the Preserved Relevant Information $\mathrm{I}[\mathrm{Y};\mathrm{Z}]$. Tishby & Zaslavsky (2015) mention that the IB objective is equivalent to minimizing $\mathrm{I}[\mathrm{X};\mathrm{Z}]+\beta\mathrm{I}[\mathrm{X};\mathrm{Y} \mid \mathrm{Z}]$, see our discussion above. Tishby et al. (2000) provide an optimal algorithm for the tabular case, when X, Y and Z are all categorical. This has spawned additional research to optimize the objective for other cases and specifically for DNNs.

"Deterministic Information Bottleneck" (DIB)

Strouse & Schwab (2017) introduce as objective "$\min \mathrm{H}[\mathrm{Z}] - \beta \mathrm{I}[\mathrm{Y};\mathrm{Z}]$". Compared to the IB objective, this also minimizes $\mathrm{H}[\mathrm{Z} \mid \mathrm{X}]$ and encourages determinism. Vice-versa, for deterministic encoders, $\mathrm{H}[\mathrm{Z} \mid \mathrm{X}] = 0$, and their objective matches the IB objective. Like Tishby et al. (2000), they provide an algorithm for the tabular case. To do so, they examine an analytical solution for their objective as it is unbounded: $\mathrm{H}[\mathrm{Z} \mid \mathrm{X}] \to -\infty$ for the optimal solution. As we discuss in section 2.3, it does not easily translate to a continuous latent representation.

"Deep Variational Information Bottleneck"

Alemi et al. (2016) rewrite the terms in the bottleneck as maximization problem "$\max \mathrm{I}[\mathrm{Y};\mathrm{Z}] - \beta \mathrm{I}[\mathrm{X};\mathrm{Z}]$" and swap the β parameter. Their β would be $\frac{1}{\beta}$ in IB above, which emphasizes that $\mathrm{I}[\mathrm{Y};\mathrm{Z}]$ is important for performance and $\mathrm{I}[\mathrm{X};\mathrm{Z}]$ acts as regularizer.

The paper derives the following variational approximation to the IB objective, where $z = f_\theta(x, \epsilon)$ denotes a stochastic latent embedding with distribution $\mathrm{p}_\theta(z \mid x)$, $\mathrm{p}_\theta(\hat{y} \mid z)$ denotes the decoder, and $\mathrm{r}(z)$ is some fixed prior distribution on the latent

[4] That is, it does not depend on θ.

embedding:

$$\min \mathbb{E}_{\hat{p}(x,y)}\, \mathbb{E}_{\epsilon\sim p(\epsilon)} \Big[-\log p_\theta(\hat{Y} = y \mid z = f_\theta(x_n, \epsilon)) + \gamma\, D_{\mathrm{KL}}(p(z|x_n) \| r(z))\Big]. \tag{27}$$

In principle, the distributions $p_\theta(\hat{y} \mid z)$ and $p_\theta(z \mid x)$ could be given by arbitrary parameterizations and function approximators. In practice, the implementation of DVIB presented by Alemi et al. (2016) constructs $p_\theta(z \mid x)$ as a multivariate Gaussian with parameterized mean and parameterized diagonal covariance using a neural network, and then uses a simple logistic regression to obtain $p_\theta(\hat{y} \mid z)$, while arbitrarily setting $r(z)$ to be a unit Gaussian around the origin. The requirement for $p_\theta(z \mid x)$ to have a closed-form Kullback-Leibler divergence limits the applicability of the DVIB objective.

The DVIB objective can be written more concisely as

$$\min H_\theta[Y \mid Z] + \gamma\, D_{\mathrm{KL}}(p(z \mid x) \,\|\, r(z))$$

in the notation introduced in section 2.1. We discuss the regularizer in more detail in section F.3.

“Conditional Entropy Bottleneck”

Fisher (2019) introduce their Conditional Entropy Bottleneck as “$\min I[X; Z \mid Y] - I[Y; Z]$”. We can rewrite the objective as $I[X; Z \mid Y] + I[X; Y \mid Z] - I[X; Y]$, using equations (25) and (26). The last term is constant for the dataset and can thus be dropped. Likewise, the IB objective can be rewritten as minimizing $I[X; Z \mid Y] + (\beta - 1) I[X; Y \mid Z]$. The two match for $\beta = 2$. Fisher (2019) provides experimental results that favorably compare to Alemi et al. (2016), possibly due to additional flexibility as Fisher (2019) do not constrain $p(z)$ to be a unit Gaussian and employ variational approximations for all terms.

C.3. Canonical IB & DIB objectives

We expand the IB and DIB objectives into “disjoint” terms and drop constant ones to find a more canonical form. This leads us to focus on the optimization of the Decoder Uncertainty $H[Y \mid Z]$ along with additional regularization terms. In section 3.1, we discuss the properties of $H[Y \mid Z]$, and in section 3.2 we examine the regularization terms.

Observation. *For IB, we obtain*

$$\arg\min I[X; Z] - \beta I[Y; Z] = \arg\min H[Y \mid Z] + \beta' \underbrace{I[X; Z \mid Y]}_{=H[Z|Y]-H[Z|X]}, \tag{28}$$

and, for DIB,

$$\arg\min H[Z] - \beta I[Y; Z] = \arg\min H[Y \mid Z] + \beta' H[Z \mid Y] = \arg\min H[Y \mid Z] + \beta'' H[Z] \tag{29}$$

with $\beta' := \frac{1}{\beta - 1} \in [0, \infty)$ *and* $\beta'' := \frac{1}{\beta} \in [0, 1)$.

Proof. For the steps marked with *, we make use of $\beta > 1$. For IB, we obtain

$$\begin{aligned}\arg\min I[X; Z] - \beta I[Y; Z] &= \arg\min I[X; Z \mid Y] + (\beta - 1) H[Y \mid Z] \\ &\overset{(*)}{=} \arg\min H[Y \mid Z] + \beta'\ I[X; Z \mid Y] \\ &\arg\min H[Y \mid Z] + \beta' (H[Z \mid Y] - H[Z \mid X]),\end{aligned} \tag{IB}$$

and, for DIB,

$$\begin{aligned}\arg\min H[Z] - \beta I[Y; Z] &= \arg\min H[Z \mid Y] + (\beta - 1) H[Y \mid Z] \\ &\overset{(*)}{=} \arg\min H[Y \mid Z] + \beta' H[Z \mid Y],\end{aligned} \tag{DIB}$$

with $\beta' := \frac{1}{\beta - 1} \in [0, \infty)$. Similarly, we show for DIB

$$\begin{aligned}\arg\min H[Z] - \beta I[Y; Z] &= \arg\min H[Z] + \beta H[Y \mid Z] \\ &\overset{(*)}{=} \arg\min H[Y \mid Z] + \beta'' H[Z],\end{aligned}$$

with $\beta'' := \frac{1}{\beta} \in [0, 1)$, which is relevant in section 3.2.

We limit ourselves to $\beta > 1$, because, for $\beta < 1$, we would be maximizing the Decoder Uncertainty, which does not make sense: the obvious solution to this is one where Z contains no information on Y, that is $\mathrm{p}(y \mid z)$ is uniform. In the case of DIB, it is to map every input deterministically to a single latent; whereas for IB, we only minimize the Redundant Information, and the solution is free to contain noise. For $\beta = 1$, we would not care about Decoder Uncertainty and only minimize Redundant Information and Reverse Decoder Uncertainty, respectively, which allows for arbitrarily bad predictions. □

We note that we have $\beta' = \frac{\beta''}{1-\beta''}$ using the relations above.

C.4. IB objectives and the Entropy Distance Metric

Another perspective on the IB objectives is by expressing them using the Entropy Distance Metric. MacKay (2003, p. 140) introduces the entropy distance

$$EDM\,(\mathrm{Y}, \mathrm{Z}) = \mathrm{H}[\mathrm{Y} \mid \mathrm{Z}] + \mathrm{H}[\mathrm{Z} \mid \mathrm{Y}]. \tag{30}$$

as a metric when we identify random variables up to permutations of the labels for categorical variables: if the entropy distance is 0, Y and Z are the same distribution up to a consistent permutation of the labels (independent of X). If the entropy distance becomes 0, both $\mathrm{H}[\mathrm{Y} \mid \mathrm{Z}] = 0 = \mathrm{H}[\mathrm{Z} \mid \mathrm{Y}]$, and we can find a bijective map from Z to Y.[5]

We can express the Reverse Decoder Uncertainty $\mathrm{H}[\mathrm{Z} \mid \mathrm{Y}]$ using the Decoder Uncertainty $\mathrm{H}[\mathrm{Y} \mid \mathrm{Z}]$ and the entropies:

$$\mathrm{H}[\mathrm{Z} \mid \mathrm{Y}] + \mathrm{H}[\mathrm{Y}] = \mathrm{H}[\mathrm{Y} \mid \mathrm{Z}] + \mathrm{H}[\mathrm{Z}],$$

and rewrite equation (30) as

$$EDM\,(\mathrm{Y}, \mathrm{Z}) = 2\mathrm{H}[\mathrm{Y} \mid \mathrm{Z}] + \mathrm{H}[\mathrm{Z}] - \mathrm{H}[\mathrm{Y}].$$

For optimization purposes, we can drop constant terms and rearrange:

$$\arg\min EDM\,(\mathrm{Y}, \mathrm{Z}) = \arg\min \mathrm{H}[\mathrm{Y} \mid \mathrm{Z}] + \tfrac{1}{2}\mathrm{H}[\mathrm{Z}].$$

C.4.1. Rewriting IB and DIB using the Entropy Distance Metric

For $\beta \geq 1$, we can rewrite equations (IB) and (DIB) as:

$$\arg\min EDM\,(\mathrm{Y}, \mathrm{Z}) + \gamma\,(\mathrm{H}[\mathrm{Y} \mid \mathrm{Z}] - \mathrm{H}[\mathrm{Z} \mid \mathrm{Y}]) + (\gamma - 1)\,\mathrm{H}[\mathrm{Z} \mid \mathrm{X}] \tag{31}$$

for IB, and

$$\arg\min EDM\,(\mathrm{Y}, \mathrm{Z}) + \gamma\,(\mathrm{H}[\mathrm{Y} \mid \mathrm{Z}] - \mathrm{H}[\mathrm{Z} \mid \mathrm{Y}]) \tag{32}$$

for DIB and replace β with $\gamma = 1 - \frac{2}{\beta} \in [-1, 1]$ which allows for a linear mix between $\mathrm{H}[\mathrm{Y} \mid \mathrm{Z}]$ and $\mathrm{H}[\mathrm{Z} \mid \mathrm{Y}]$.

DIB will encourage the model to match both distributions for $\gamma = 0$ ($\beta = 2$), as we obtain a term that matches the Entropy Distance Metric from section C.4, and otherwise trades off Decoder Uncertainty and Reverse Decoder Uncertainty. IB behaves similarly but tends to maximize Encoding Uncertainty as $\gamma - 1 \in [-2, 0]$. Fisher (2019) argues for picking this configuration similar to the arguments in section C.1. DIB will force both distributions to become exactly the same, which would turn the decoder into a permutation matrix for categorical variables.

D. Decoder Uncertainty $\mathrm{H}[\mathrm{Y} \mid \mathrm{Z}]$

D.1. Cross-entropy loss

The cross-entropy loss features prominently in section 3.1. We can derive the usual cross-entropy loss for our model by minimizing the Kullback-Leibler divergence between the empirical sample distribution $\hat{\mathrm{p}}(x, y)$ and the parameterized

[5] The argument for continuous variables is the same. We need to identify distributions up to "isentropic" bijections.

distribution $p_\theta(x)\, p_\theta(\hat{y} \mid x)$. For discriminative models, we are only interested in $p_\theta(\hat{y} \mid x)$, and can simply set $p_\theta(x) = \hat{p}(x)$:

$$\begin{aligned}
&\arg\min_\theta D_{\mathrm{KL}}(\hat{p}(x, y) \,\|\, p_\theta(x)\, p_\theta(\hat{Y} = y \mid x)) \\
&= \arg\min_\theta D_{\mathrm{KL}}(\hat{p}(y \mid x) \,\|\, p_\theta(\hat{Y} = y \mid x)) + \underbrace{D_{\mathrm{KL}}(\hat{p}(x) \,\|\, p_\theta(x))}_{=0} \\
&= \arg\min_\theta H(\hat{p}(y \mid x) \,\|\, p_\theta(\hat{Y} = y \mid x)) - \underbrace{H[Y \mid X]}_{\text{const.}} \\
&= \arg\min_\theta H(\hat{p}(y \mid x) \,\|\, p_\theta(\hat{Y} = y \mid x)).
\end{aligned}$$

In section 3.1, we introduce the shorthand $H_\theta[Y \mid X]$ for $H(\hat{p}(y \mid x) \,\|\, p_\theta(\hat{Y} = y \mid x))$ and refer to it as Prediction Cross-Entropy.

D.2. Upper bounds & training error minimization

To motivate that $H[Y \mid Z]$ (or $H_\theta[Y \mid Z]$) can be used as main loss term, we show that it can bound the (training) error probability since *accuracy* is often the true objective when machine learning models are deployed on real-world problems[6].

Observation. *The Decoder Cross-Entropy provides an upper bound on the Decoder Uncertainty:*

$$H[Y \mid Z] \le H[Y \mid Z] + D_{\mathrm{KL}}(p(y \mid z) \,\|\, p_\theta(\hat{y} \mid z)) = H_\theta[Y \mid Z],$$

and further bounds the training error:

$$p(\text{“}\hat{Y} \textit{ is wrong”}) \le 1 - e^{-H_\theta[Y|Z]} = 1 - e^{-\left(H[Y|Z] + D_{\mathrm{KL}}(p(y|z) \| p_\theta(\hat{y}|z))\right)}.$$

Likewise, for the Prediction Cross-Entropy $H_\theta[Y \mid X]$ *and the Label Uncertainty* $H[Y \mid X]$.

Proof. The upper bounds for Decoder Uncertainty $H[Y \mid Z]$ and Label Uncertainty $H[Y \mid X]$ follow from the non-negativity of the Kullback-Leibler divergence, for example:

$$\begin{aligned}
0 &\le D_{\mathrm{KL}}(p(y \mid z) \,\|\, p_\theta(\hat{y} \mid z)) = H_\theta[Y \mid Z] - H[Y \mid Z], \\
0 &\le D_{\mathrm{KL}}(\hat{p}(y \mid x) \,\|\, p_\theta(\hat{y} \mid x)) = H_\theta[Y \mid X] - H[Y \mid X].
\end{aligned}$$

The derivation for the training error probability is as follows:

$$\begin{aligned}
p(\text{“}\hat{Y} \text{ is correct”}) &= \mathbb{E}_{\hat{p}(x,y)}\, p(\text{“}\hat{Y} \text{ is correct”} \mid x, y) = \mathbb{E}_{\hat{p}(x,y)}\, \mathbb{E}_{p_\theta(z|x)}\, p_\theta(\hat{Y} = y \mid z) \\
&= \mathbb{E}_{p(y,z)}\, p_\theta(\hat{Y} = y \mid z).
\end{aligned}$$

We can then apply Jensen's inequality using convex $h(x) = -\ln x$:

$$\begin{aligned}
& h\left(\mathbb{E}_{p(y,z)}\, p_\theta(\hat{Y} = y \mid z)\right) \le \mathbb{E}_{p(y,z)}\, h\left(p_\theta(\hat{Y} = y \mid z)\right) \\
&\Leftrightarrow p(\text{“}\hat{Y} \text{ is correct”}) \ge e^{-H(p(y|z) \| p_\theta(\hat{Y}=y|z))} \\
&\Leftrightarrow p(\text{“}\hat{Y} \text{ is wrong”}) \le 1 - e^{-H_\theta[Y|Z]}.
\end{aligned}$$

For small $H_\theta[Y \mid Z]$, we note that one can use the approximation $e^x \approx 1 + x$ to obtain:

$$p(\text{“}\hat{Y} \text{ is wrong”}) \lessapprox H_\theta[Y \mid Z]. \tag{33}$$

Finally, we split the Decoder Cross-Entropy into the Decoder Uncertainty and a Kullback-Leibler divergence:

$$H_\theta[Y \mid Z] = H[Y \mid Z] + D_{\mathrm{KL}}(p(y \mid z) \,\|\, p_\theta(\hat{Y} = y \mid z)).$$

If we upper-bound $D_{\mathrm{KL}}(p(y|z) \| p_\theta(\hat{Y} = y|z))$, minimizing the Decoder Uncertainty $H[Y \mid Z]$ becomes a sensible minimization objective as it reduces the probability of misclassification.

We can similarly show that the training error is bounded by the Prediction Cross-Entropy $H_\theta[Y \mid X]$. □

In the next section, we examine categorical Z for which optimal decoders can be constructed and $D_{\mathrm{KL}}(p(y \mid z) \,\|\, p_\theta(\hat{Y} = y \mid z))$ becomes zero.

[6] As we only take into account the empirical distribution $\hat{p}(x, y)$ available for training, the following derivation refers only to the empirical risk, and not to the expected risk of the estimator $\hat{Y}$.

E. Categorical Z

For categorical Z, $\mathrm{p}(y \mid z)$ can be computed exactly for a given encoder $\mathrm{p}_\theta(z \mid x)$ by using the empirical data distribution, which, in turn, allows us to compute $\mathrm{H}[\mathrm{Y} \mid \mathrm{Z}]$[7]. This is similar to computing a confusion matrix between Y and Z but using information content instead of probabilities.

Moreover, if we set $\mathrm{p}_\theta(\hat{y} \mid z) := \mathrm{p}(\mathrm{Y} = \hat{y} \mid z)$ to have an optimal decoder, we obtain equality in equation (6), and obtain $\mathrm{H}_\theta[\mathrm{Y} \mid \mathrm{X}] \leq \mathrm{H}_\theta[\mathrm{Y} \mid \mathrm{Z}] = \mathrm{H}[\mathrm{Y} \mid \mathrm{Z}]$. If the encoder were also deterministic, we would obtain $\mathrm{H}_\theta[\mathrm{Y} \mid \mathrm{X}] = \mathrm{H}_\theta[\mathrm{Y} \mid \mathrm{Z}] = \mathrm{H}[\mathrm{Y} \mid \mathrm{Z}]$. We can minimize $\mathrm{H}[\mathrm{Y} \mid \mathrm{Z}]$ directly using gradient descent. $\frac{d}{d\theta}\mathrm{H}[\mathrm{Y} \mid \mathrm{Z}]$ only depends on $\mathrm{p}(y \mid z)$ and $\frac{d}{d\theta}\,\mathrm{p}_\theta(z \mid x)$:

$$\frac{d}{d\theta}\mathrm{H}[\mathrm{Y} \mid \mathrm{Z}] = \mathbb{E}_{\mathrm{p}(x,z)}\left[\frac{d}{d\theta}\left[\ln \mathrm{p}_\theta(z \mid x)\right] \mathbb{E}_{\hat{\mathrm{p}}(y \mid x)}\, h\left(\mathrm{p}(y \mid z)\right)\right].$$

Proof.

$$\begin{aligned}\frac{d}{d\theta}\mathrm{H}[\mathrm{Y} \mid \mathrm{Z}] &= \frac{d}{d\theta}\,\mathbb{E}_{\mathrm{p}(y,z)}\, h\left(\mathrm{p}(y \mid z)\right) = \frac{d}{d\theta}\,\mathbb{E}_{\mathrm{p}(x,y,z)}\, h\left(\mathrm{p}(y \mid z)\right) = \mathbb{E}_{\hat{\mathrm{p}}(x,y)}\,\frac{d}{d\theta}\,\mathbb{E}_{\mathrm{p}_\theta(z \mid x)}\, h\left(\mathrm{p}(y \mid z)\right)\\ &= \mathbb{E}_{\mathrm{p}_\theta(z \mid x)}\,\mathbb{E}_{\hat{\mathrm{p}}(x,y)}\,\frac{d}{d\theta}\left[h\left(\mathrm{p}(y \mid z)\right)\right] + h\left(\mathrm{p}(y \mid z)\right)\frac{d}{d\theta}\left[\ln \mathrm{p}_\theta(z \mid x)\right]\\ &= \mathbb{E}_{\mathrm{p}(x,y,z)}\,\frac{d}{d\theta}\left[h\left(\mathrm{p}(y \mid z)\right)\right] + h\left(\mathrm{p}(y \mid z)\right)\frac{d}{d\theta}\left[\ln \mathrm{p}_\theta(z \mid x)\right].\end{aligned}$$

And now we show that $\mathbb{E}_{\mathrm{p}(x,y,z)}\,\frac{d}{d\theta}\left[h\left(\mathrm{p}(y \mid z)\right)\right] = 0$:

$$\begin{aligned}\mathbb{E}_{\mathrm{p}(x,y,z)}\,\frac{d}{d\theta}\left[h\left(\mathrm{p}(y \mid z)\right)\right] &= \mathbb{E}_{\mathrm{p}(y,z)}\,\frac{d}{d\theta}\left[h\left(\mathrm{p}(y \mid z)\right)\right] = \mathbb{E}_{\mathrm{p}(y,z)}\,\frac{-1}{\mathrm{p}(y \mid z)}\frac{d}{d\theta}\,\mathrm{p}(y \mid z)\\ &= -\int \frac{\mathrm{p}(y,z)}{\mathrm{p}(y \mid z)}\frac{d}{d\theta}\,\mathrm{p}(y \mid z)\,dy\,dz = -\int \mathrm{p}(z)\int \frac{d}{d\theta}\,\mathrm{p}(y \mid z)\,dy\,dz\\ &= -\int \mathrm{p}(z)\frac{d}{d\theta}\Big[\underbrace{\int \mathrm{p}(y \mid z)\,dy}_{=1}\Big]\,dz = 0.\end{aligned}$$

Splitting the expectation and reordering of $\mathbb{E}_{\mathrm{p}(x,y,z)}\, h\left(\mathrm{p}(y \mid z)\right)\frac{d}{d\theta}\left[\ln \mathrm{p}_\theta(z \mid x)\right]$, we obtain the result. □

The same holds for Reverse Decoder Uncertainty $\mathrm{H}[\mathrm{Z} \mid \mathrm{Y}]$ and for the other quantities as can be verified easily.

If we minimize $\mathrm{H}[\mathrm{Y} \mid \mathrm{Z}]$ directly, we can compute $\mathrm{p}(y \mid z)$ after every training epoch and fix $\mathrm{p}_\theta(\hat{y} \mid z) := \mathrm{p}(\mathrm{Y} = \hat{y} \mid z)$ to create the discriminative model $\mathrm{p}_\theta(\hat{y} \mid x)$. This is a different perspective on the self-consistent equations from Tishby et al. (2000); Gondek & Hofmann (2003).

E.1. Empirical evaluation of $D_{\mathrm{KL}}(\mathrm{p}(y \mid z) \,\|\, \mathrm{p}_\theta(\hat{\mathrm{Y}} = y \mid z))$ during training

We examine the size of the gap between Decoder Uncertainty and Decoder Cross-Entropy and the training behavior of the two cross-entropies with *categorical* latent Z on Permutation MNIST and CIFAR10. For Permutation MNIST (Goodfellow et al., 2013), we use the common fully-connected ReLU $784-1024-1024-C$ encoder architecture, with $C = 100$ categories for Z. For CIFAR10 (Krizhevsky et al., 2009), we use a standard ResNet18 model with C many output classes as encoder (He et al., 2016a). See section G for more details about the hyperparameters. Even though a $C \times 10$ matrix and a SoftMax would suffice to describe the decoder matrix $\mathrm{p}_\theta(\hat{y} \mid z)$[8], we have found that over-parameterization using a separate DNN benefits optimization a lot. Thus, to parameterize the decoder matrix, we use fully-connected ReLUs $C - 1024 - 1024 - 10$ with a final SoftMax layer. We compute it once per batch during training and back-propagate into it.

Figure E.1 shows the three metrics as we train with each of them in turn. Our results do not achieve SOTA accuracy on the test set—we impose a harder optimization problem as Z is categorical, and we are essentially solving a hard-clustering problem first and then map these clusters to $\hat{\mathrm{Y}}$. Results are provided for the training set in order to compare with the optimal decoder.

[7] $\mathrm{p}(y \mid z)$ depends on θ through $\mathrm{p}_\theta(z \mid x)$: $\mathrm{p}(y \mid z) = \frac{\sum_x \hat{\mathrm{p}}(x,y)\,\mathrm{p}_\theta(z \mid x)}{\sum_x \hat{\mathrm{p}}(x)\,\mathrm{p}_\theta(z \mid x)}$.

[8] For categorical Z, $\mathrm{p}_\theta(\hat{y} \mid z)$ is a stochastic matrix which sums to 1 along the $\hat{\mathrm{Y}}$ dimension.

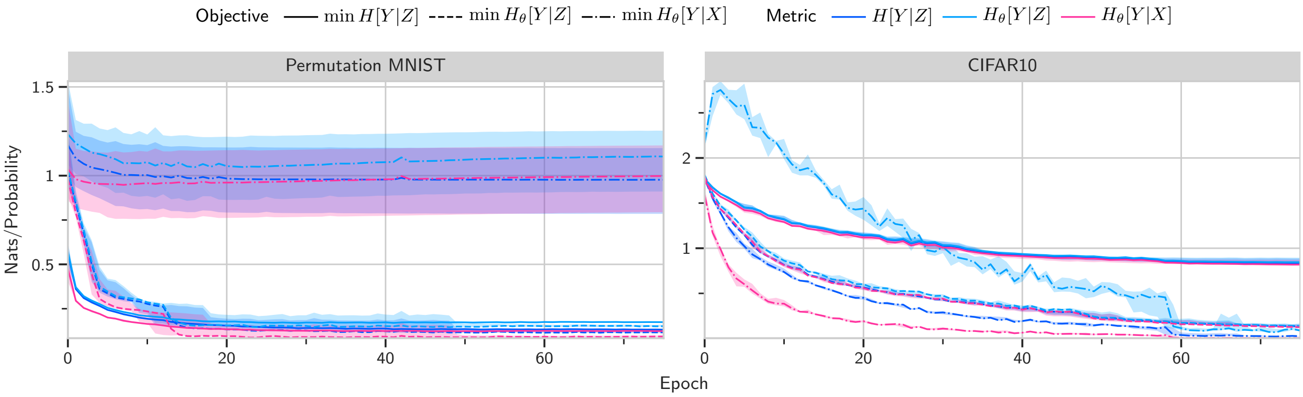


Figure E.1. Decoder Uncertainty, Decoder Cross-Entropy and Prediction Cross-Entropy for Permutation-MNIST and CIFAR10 with a categorical Z*.* $C = 100$ categories are used for Z. We optimize with different minimization objectives in turn and plot the metrics. $D_{\mathrm{KL}}(\mathrm{p}(y|z)||\mathrm{p}_\theta(\hat{\mathrm{Y}} = y|z))$ is small when training with $\mathrm{H}_\theta[\mathrm{Y} \mid \mathrm{Z}]$ or $\mathrm{H}[\mathrm{Y} \mid \mathrm{Z}]$. When training with $\mathrm{H}_\theta[\mathrm{Y} \mid \mathrm{X}]$ on CIFAR10, $D_{\mathrm{KL}}(\mathrm{p}(y|z)||\mathrm{p}_\theta(\hat{\mathrm{Y}} = y \mid z))$ remains quite large. We run 8 trials each and plot the median with confidence bounds (25% and 75% quartiles). See section E.1 for more details.

As predicted, the Decoder Cross-Entropy upper-bounds both the Decoder Uncertainty $\mathrm{H}[\mathrm{Y} \mid \mathrm{Z}]$ and the Prediction Cross-Entropy in all cases. Likewise, the gap between $\mathrm{H}_\theta[\mathrm{Y} \mid \mathrm{Z}]$ and $\mathrm{H}[\mathrm{Y} \mid \mathrm{Z}]$ is tiny when we minimize $\mathrm{H}_\theta[\mathrm{Y} \mid \mathrm{Z}]$. On the other hand, minimizing Prediction Cross-Entropy can lead to large gaps between $\mathrm{H}_\theta[\mathrm{Y} \mid \mathrm{Z}]$ and $\mathrm{H}[\mathrm{Y} \mid \mathrm{Z}]$, as can be seen for CIFAR10.

Very interestingly, on MNIST Decoder Cross-Entropy provides a better training objective whereas on CIFAR10 Prediction Cross-Entropy trains lower. Decoder Uncertainty does not train very well on CIFAR10, and Prediction Cross-Entropy does not train well on Permutation MNIST at all. We suspect DNN architectures in the literature have evolved to train well with cross-entropies, but we are surprised by the heterogeneity of the results for the two datasets and models.

F. Surrogates for regularization terms

F.1. Differential entropies

Observation. *After adding zero-entropy noise, the inequality* $\mathrm{I}[\mathrm{X}; \mathrm{Z} \mid \mathrm{Y}] \le \mathrm{H}[\mathrm{Z} \mid \mathrm{Y}] \le \mathrm{H}[\mathrm{Z}]$ *also holds in the continuous case, and we can minimize* $\mathrm{I}[\mathrm{X}; \mathrm{Z} \mid \mathrm{Y}]$ *in the IB objective by minimizing* $\mathrm{H}[\mathrm{Z} \mid \mathrm{Y}]$ *or* $\mathrm{H}[\mathrm{Z}]$*, similarly to the DIB objective. We present a formal proof in section F.1.*

Theorem 1. *For random variables A, B, we have*

$$\mathrm{H}[A + B] \ge \mathrm{H}[B].$$

Proof. See Bercher & Vignat (2002, section 2.2). □

Proposition 1. *Let Y, Z and X be random variables satisfying the independence property* $Z \perp Y|X$*, and F a possibly stochastic function such that* $Z = F(X) + \epsilon$*, with independent noise* ϵ *satisfying* $\epsilon \perp F(X), \epsilon \perp Y$ *and* $H(\epsilon) = 0$*. Then the following holds whenever* $\mathrm{I}[\mathrm{Y}; \mathrm{Z}]$ *is well-defined.*

$$\mathrm{I}[\mathrm{X}; \mathrm{Z} \mid \mathrm{Y}] \le \mathrm{H}[\mathrm{Z} \mid \mathrm{Y}] \le \mathrm{H}[\mathrm{Z}].$$

Proof. First, we note that $\mathrm{H}[\mathrm{Z} \mid \mathrm{X}] = \mathrm{H}[F(X) + \epsilon \mid X] \ge \mathrm{H}[\epsilon \mid X] = \mathrm{H}[\epsilon]$ with theorem 1, as ϵ is independent of X, and thus $\mathrm{H}[\mathrm{Z} \mid \mathrm{X}] \ge 0$. We have $\mathrm{H}[\mathrm{Z} \mid \mathrm{X}] = \mathrm{H}[Z \mid X, Y]$ by the conditional independence assumption, and by the non-negativity of mutual information, $\mathrm{I}[\mathrm{Y}; \mathrm{Z}] \ge 0$. Then:

$$\mathrm{I}[\mathrm{X}; \mathrm{Z} \mid \mathrm{Y}] + \underbrace{\mathrm{H}[\mathrm{Z} \mid \mathrm{X}]}_{\ge 0} = \mathrm{H}[\mathrm{Z} \mid \mathrm{Y}]$$

$$\mathrm{H}[\mathrm{Z} \mid \mathrm{Y}] + \underbrace{\mathrm{I}[\mathrm{Y}; \mathrm{Z}]}_{\ge 0} = \mathrm{H}[\mathrm{Z}]$$

□

The probabilistic model from section 2.1 fulfills the conditions exactly, and the two statements motivate our observation.

It is important to note that while zero-entropy noise is necessary for preserving inequalities like $\mathrm{I}[\mathrm{X};\mathrm{Z} \mid \mathrm{Y}] \leq \mathrm{H}[\mathrm{Z} \mid \mathrm{Y}] \leq \mathrm{H}[\mathrm{Z}]$ in the continuous case, any Gaussian noise will suffice for optimization purposes: we optimize via pushing down an upper bound, and constant offsets will not affect this.

Thus, if we had $\mathrm{H}[\epsilon] \neq 0$, even though $\mathrm{I}[\mathrm{X};\mathrm{Z} \mid \mathrm{Y}] + \mathrm{H}[\mathrm{Z} \mid \mathrm{X}] \nleq \mathrm{H}[\mathrm{Z} \mid \mathrm{Y}]$, we could instead use

$$\mathrm{I}[\mathrm{X};\mathrm{Z} \mid \mathrm{Y}] + \mathrm{H}[\mathrm{Z} \mid \mathrm{X}] - \mathrm{H}[\epsilon] \leq \mathrm{H}[\mathrm{Z} \mid \mathrm{Y}] - \mathrm{H}[\epsilon]$$

as upper bound to minimize. The gradients remain the same.

This also points to the nature of differential entropies as lacking a proper point of origin by themselves. We choose one by fixing $\mathrm{H}[\epsilon]$. Just like other literature usually only considers mutual information as meaningful, we consider $\mathrm{H}[\mathrm{Z} \mid \mathrm{X}] - \mathrm{H}[\epsilon]$ as more meaningful than $\mathrm{H}[\mathrm{Z} \mid \mathrm{X}]$. However, we can side-step this discussion conveniently by picking a canonical noise as point of origin in the form of zero-entropy noise $\mathrm{H}[\epsilon] = 0$.

F.2. Upper bounds

We derive this result as follows:

$$\begin{aligned}\mathrm{H}[\mathrm{Z} \mid \mathrm{Y}] &= \mathbb{E}_{\hat{\mathrm{p}}(y)} \mathrm{H}[\mathrm{Z} \mid y] \\ &\leq \mathbb{E}_{\hat{\mathrm{p}}(y)} \tfrac{1}{2} \ln \det(2\pi e \operatorname{Cov}[\mathrm{Z} \mid y]) \\ &\leq \mathbb{E}_{\hat{\mathrm{p}}(y)} \sum_i \tfrac{1}{2} \ln(2\pi e \operatorname{Var}[\mathrm{Z}_i \mid y]) \\ &\approx \mathbb{E}_{\hat{\mathrm{p}}(y)} \sum_i \tfrac{1}{2} \ln(2\pi e \widehat{\operatorname{Var}}[\mathrm{Z}_i \mid y]),\end{aligned}$$

Theorem 2. *Given a k-dimensional random variable $X = (X_i)_{i=1}^k$ with $\operatorname{Var}[X_i] > 0$ for all i,*

$$\begin{aligned}\mathrm{H}[X] &\leq \tfrac{1}{2} \ln \det(2\pi e \operatorname{Cov}[X]) \\ &\leq \sum_i \tfrac{1}{2} \ln(2\pi e \operatorname{Var}[X_i]).\end{aligned}$$

Proof. First, the multivariate normal distribution with same covariance is the maximum entropy distribution for that covariance, and thus $\mathrm{H}[X] \leq \ln \det(2\pi e \operatorname{Cov}[X])$, when we substitute the differential entropy for a multivariate normal distribution with covariance $\operatorname{Cov}[X]$. Let $\Sigma_0 := \operatorname{Cov}[X]$ be the covariance matrix and $\Sigma_1 := \operatorname{diag}(\operatorname{Var}[X_i])_i$ the matrix that only contains the diagonal. Because we add independent noise, $\operatorname{Var}[X_i] > 0$ and thus Σ_1^{-1} exists. It is clear that $\operatorname{tr}(\Sigma_1^{-1}\Sigma_0) = k$. Then, we can use the KL-Divergence between two multivariate normal distributions $\mathcal{N}_0, \mathcal{N}_1$ with same mean 0 and covariances Σ_0 and Σ_1 to show that $\ln \det \Sigma_0 \leq \ln \det \Sigma_1$:

$$\begin{aligned}0 \leq D_{\mathrm{KL}}(\mathcal{N}_0 \parallel \mathcal{N}_1) &= \tfrac{1}{2}\left(\operatorname{tr}(\Sigma_1^{-1}\Sigma_0) - k + \ln\left(\frac{\det \Sigma_1}{\det \Sigma_0}\right)\right) \\ \Leftrightarrow 0 \leq \tfrac{1}{2} \ln\left(\frac{\det \Sigma_1}{\det \Sigma_0}\right) &\Leftrightarrow \tfrac{1}{2} \ln \det \Sigma_0 \leq \tfrac{1}{2} \ln \det \Sigma_1.\end{aligned}$$

We substitute the definitions of Σ_0 and Σ_1, and obtain the second inequality after adding $k \ln(2\pi e)$ on both sides. □

Theorem 3. *Given a k-dimensional real-valued random variable $X = (X_i)_{i=1}^k \in \mathbb{R}^k$, we can bound the entropy by the mean squared norm of the latent:*

$$\mathbb{E}\, \|X\|^2 \leq C' \Rightarrow \mathrm{H}[X] \leq C, \tag{34}$$

with $C' := \frac{k e^{2C/k}}{2\pi e}$.

Proof. We begin with the previous bound:

$$\begin{aligned}\mathrm{H}[X] &\leq \sum_i \tfrac{1}{2} \ln(2\pi e \operatorname{Var}[X_i]) = \tfrac{k}{2} \ln 2\pi e + \tfrac{1}{2} \ln \prod_i \operatorname{Var}[X_i] \\ &\leq \tfrac{k}{2} \ln 2\pi e + \tfrac{1}{2} \ln \left(\tfrac{1}{k} \sum_i \operatorname{Var}[X_i] \right)^k = \tfrac{k}{2} \ln \tfrac{2\pi e}{k} \sum_i \operatorname{Var}[X_i] \\ &\leq \tfrac{k}{2} \ln \tfrac{2\pi e}{k} \mathbb{E} \, \|X\|^2,\end{aligned}$$

where we use the AM-GM inequality:

$$\left(\prod_i \operatorname{Var}[X_i] \right)^{\frac{1}{k}} \leq \tfrac{1}{k} \sum_i \operatorname{Var}[X_i]$$

and the monotony of the logarithm with:

$$\sum_i \operatorname{Var}[X_i] = \sum_i \mathbb{E}\left[X_i^2\right] - \mathbb{E}\,[X_i]^2 \leq \sum_i \mathbb{E}\left[X_i^2\right] = \mathbb{E}\,\|X\|^2$$

Bounding using $\mathbb{E}\,\|X\|^2 \leq C'$, we obtain

$$\mathrm{H}[X] \leq \tfrac{k}{2} \ln \tfrac{2\pi e}{k} C' = C,$$

and solving for C' yields the statement. □

This theorem provides justification for the use of $\ln \mathbb{E}\,\|Z\|^2$ as a regularizer, but does not justify the use of $\mathbb{E}\,\|Z\|^2$ directly. Here, we give two motivations. We first observe that $\ln x \leq x - 1$ due to ln's strict convexity and $\ln 1 = 0$, and thus:

$$\mathrm{H}[X] \leq \tfrac{k}{2} \ln \tfrac{2\pi e}{k} \mathbb{E}\,\|X\|^2 = \tfrac{k}{2} \left(\ln \tfrac{2\pi}{k} \mathbb{E}\,\|X\|^2 - 1 \right) \leq \pi \mathbb{E}\,\|X\|^2.$$

We can also take a step back and remind ourselves that IB objectives are actually Lagrangians, and β in $\min \mathrm{I}[\mathrm{X};\mathrm{Z}] - \beta \mathrm{I}[\mathrm{Y};\mathrm{Z}]$ is introduced as Lagrangian multiplier for the constrained objective:

$$\min \mathrm{I}[\mathrm{X};\mathrm{Z}] \text{ s.t. } \mathrm{I}[\mathrm{Y};\mathrm{Z}] \geq C.$$

We can similarly write our canonical DIB objective $\mathrm{H}[\mathrm{Y} \mid \mathrm{Z}] + \beta'' \mathrm{H}[\mathrm{Z}]$ as constrained objective

$$\min \mathrm{H}[\mathrm{Y} \mid \mathrm{Z}] \text{ s.t. } \mathrm{H}[\mathrm{Z}] \leq C,$$

and use above statement to find the approximate form

$$\min \mathrm{H}[\mathrm{Y} \mid \mathrm{Z}] \text{ s.t. } \mathbb{E}\,\|Z\|^2 \leq C'.$$

Reintroducing a Lagrangian multiplier recovers our reguralized $\mathbb{E}\,\|Z\|^2$ objective:

$$\min \mathrm{H}[\mathrm{Y} \mid \mathrm{Z}] + \gamma \mathbb{E}\,\|Z\|^2.$$

F.3. Alemi et al. (2016) and $\mathbb{E}\,\|Z\|^2$

Alemi et al. (2016) model $\mathrm{p}_\theta(z \mid x)$ explicitly as multivariate Gaussian with parameterized mean and parameterized diagonal covariance in their encoder and regularize it to become close to $\mathcal{N}(0, I_k)$ by minimizing the Kullback-Leibler divergence $D_{\mathrm{KL}}(\mathrm{p}_\theta(z \mid x) \,\|\, \mathcal{N}(0, I_k))$ alongside the cross-entropy:

$$\min \mathrm{H}_\theta[\mathrm{Y} \mid \mathrm{Z}] + \gamma\, D_{\mathrm{KL}}(\mathrm{p}(z \mid x) \,\|\, \mathrm{r}(z)),$$

as detailed in section C.2.

We can expand the regularization term to

$$
\begin{aligned}
&D_{\mathrm{KL}}(\mathrm{p}(z \mid x) \,\|\, \mathcal{N}(0,\, I_k)) \\
&\quad = \mathbb{E}_{\hat{\mathrm{p}}(x)}\, \mathbb{E}_{\mathrm{p}(z|x)}\, h\left((2\pi)^{-\frac{k}{2}}\, e^{-\frac{1}{2}\|\mathrm{Z}\|^2|}\right) - \mathrm{H}[\mathrm{Z} \mid \mathrm{X}] \\
&\quad = \mathbb{E}_{\mathrm{p}(z)}\left[\frac{k}{2}\ln(2\pi) + \frac{1}{2}\|\mathrm{Z}\|^2\right] - \mathrm{H}[\mathrm{Z} \mid \mathrm{X}].
\end{aligned}
$$

After dropping constant terms (as they don't matter for optimization purposes), we obtain

$$
= \frac{1}{2}\mathbb{E}\, \|\mathrm{Z}\|^2 - \mathrm{H}[\mathrm{Z} \mid \mathrm{X}].
$$

When we inject zero-entropy noise into the latent Z, we have $\mathrm{H}[\mathrm{Z} \mid \mathrm{X}] \geq 0$ and thus $\mathbb{E}\,\|\mathrm{Z}\|^2 - \mathrm{H}[\mathrm{Z} \mid \mathrm{X}] \leq \mathbb{E}\,\|\mathrm{Z}\|^2$. Thus, the $\mathbb{E}\,\|\mathrm{Z}\|^2$ regularizer also upper-bounds DVIB's regularizer in this case.

In particular, we have equality when we use a deterministic encoder. When we inject zero-entropy noise and use a deterministic encoder, we are optimizing the DVIB objective function when we use the $\mathbb{E}\,\|\mathrm{Z}\|^2$ regularizer. In other words, in this particular case, we could reinterpret "$\min \mathrm{H}_\theta[\mathrm{Y} \mid \mathrm{Z}] + \gamma\, \mathbb{E}\,\|\mathrm{Z}\|^2$" as optimzing the DVIB objective from Alemi et al. (2016) if they were using a constant covariance instead of parameterizing it in their encoder. This does not hold for stochastic encoders.

F.4. Soft clustering by entropy Minimization with Gaussian noise

Consider the problem of minimizing H[Z | Y] and H[Y | Z], in the setting where $Z = f_\theta(X) + \epsilon \sim \mathcal{N}(0, \sigma^2)$—i.e. the embedding Z is obtained by adding Gaussian noise to a deterministic function of the input. Let the training set be enumerated $x_1, \ldots, x_n$, with $\mu_i = f_\theta(x_i)$. Then the distribution of Z is given by a mixture of Gaussians with the following density, where $d(x, \mu_i) := \|x - \mu_i\|/\sigma^2$.

$$
\mathrm{p}(z) \propto \frac{1}{n}\sum_{i=1}^{n} \exp(-d(z, \mu_i))
$$

Assuming that each x_i has a deterministic label y_i, we then find that the conditional distributions $\mathrm{p}(y \mid z)$ and $\mathrm{p}(z \mid y)$ are given as follows:

$$
\begin{aligned}
\mathrm{p}(z \mid y) &\propto \frac{1}{n_y}\sum_{i:y_i=y} \exp(-d(z, \mu_i)) \\
\mathrm{p}(y \mid z) &= \sum_{i:y_i=y} \mathrm{p}(\mu_i \mid z) = \sum_{i:y_i=y} \frac{\mathrm{p}(z \mid \mu_i)\,\mathrm{p}(\mu_i)}{\mathrm{p}(z)} \\
&= \frac{\sum_{i:y_i=y} \mathrm{p}(z \mid \mu_i)}{\sum_{k=1}^{n} \mathrm{p}(z \mid \mu_k)} = \frac{\sum_{i:y_i=y} \exp(-d(z, \mu_i))}{\sum_{k=1}^{n} \exp(-d(z, \mu_k))},
\end{aligned}
$$

where n_y is the number of x_i with class $y_i = y$. Thus, the conditional $Z|Y$ can be interpreted as a mixture of Gaussians and $Y|Z$ as a Softmax marginal with respect to the distances between Z and the mean embeddings. We observe that H[Z | Y] is lower-bounded by the entropy of the random noise added to the embeddings:

$$
\mathrm{H}[\mathrm{Z} \mid \mathrm{Y}] \geq \mathrm{H}[f_\theta(X) + \epsilon \mid \mathrm{Y}] \geq \mathrm{H}[\epsilon]
$$

with equality when the distribution of $f_\theta(X)|Y$ is deterministic – that is f_θ is constant for each equivalence class.

Further, the entropy H[Y | Z] is minimized when H[Z] is large compared to H[Z | Y] as we have the decomposition

$$
\mathrm{H}[\mathrm{Y} \mid \mathrm{Z}] = \mathrm{H}[\mathrm{Z} \mid \mathrm{Y}] - \mathrm{H}[\mathrm{Z}] + \mathrm{H}[\mathrm{Y}].
$$

In particular, when f_θ is constant over equivalence classes of the input, then H[Y | Z] is minimized when the entropy $H[f_\theta(X) + \epsilon]$ is large – i.e. the values of $f_\theta(x_i)$ for each equivalence class are distant from each other and there is minimal overlap between the clusters. Therefore, the optima of the information bottleneck objective under Gaussian noise share similar properties to the optima of geometric clustering of the inputs according to their output class.

To gain a better understanding of local optimization behavior, we decompose the objective terms as follows:

$$\begin{aligned}\mathrm{H}[\mathrm{Z} \mid \mathrm{Y}] &= \mathbb{E}_{\hat{\mathrm{p}}(y)} \mathrm{H}(\mathrm{p}(z \mid y) \,\|\, \mathrm{p}(z \mid y)) \\ &= \mathbb{E}_{\hat{\mathrm{p}}(x,y)} \mathrm{H}(\mathrm{p}(z \mid x) \,\|\, \mathrm{p}(z \mid y)) \\ &= \mathbb{E}_{\hat{\mathrm{p}}(x,y)} D_{\mathrm{KL}}(\mathrm{p}(z \mid x) \,\|\, \mathrm{p}(z \mid y)) + \mathrm{H}[Z \mid x] \\ &= \mathbb{E}_{\hat{\mathrm{p}}(x,y)} D_{\mathrm{KL}}(\mathrm{p}(z \mid x) \,\|\, \mathrm{p}(z \mid y)) \\ &\quad + \underbrace{\mathrm{H}[\mathrm{Z} \mid \mathrm{X}]}_{=const}.\end{aligned}$$

To examine how the mean embedding μ_k of a single datapoint x_k affects this entropy term, we look at the derivative of this expression with respect to $\mu_k = f_\theta(x_k)$. We obtain:

$$\begin{aligned}\frac{d}{d\mu_k}\mathrm{H}[\mathrm{Z} \mid \mathrm{Y}] &= \frac{d}{d\mu_k} \mathrm{H}[\mathrm{Z} \mid y_k] \\ &= \frac{d}{d\mu_k} \mathbb{E}_{\mathrm{p}(x|y_k)} D_{\mathrm{KL}}(\mathrm{p}(z \mid x) \,\|\, \mathrm{p}(z \mid y)) \\ &= \sum_{i \neq i : y_i = y_k} \frac{1}{n_{y_k}} \frac{d}{d\mu_k} D_{\mathrm{KL}}(\mathrm{p}(z \mid x_i) \,\|\, \mathrm{p}(z \mid y_k)) \\ &\quad + \frac{1}{n_{y_k}} \frac{d}{d\mu_k} D_{\mathrm{KL}}(\mathrm{p}(z \mid x_k) \,\|\, \mathrm{p}(z \mid y_k)).\end{aligned}$$

While these derivatives do not have a simple analytic form, we can use known properties of the KL divergence to develop an intuition on how the gradient will behave. We observe that in the left-hand sum μ_k only affects the distribution of $Z|Y$ (that is we are differentiating a sum of terms that look like a reverse KL), whereas it has greater influence on $\mathrm{p}(z \mid x_k)$ in the right-hand term, and so its gradient will more closely resemble that of the forward KL. The left-hand-side term will therefore push μ_k towards the centroid of the means of inputs mapping to y, whereas the right-hand side term is mode-seeking.

F.5. A note on differential and discrete entropies

The mutual information between two random variables can be defined in terms of the KL divergence between the product of their marginals and their joint distribution. However, the KL divergence is only well-defined when the Radon-Nikodym derivative of the density of the joint with respect to the product exists. Mixing continuous and discrete distributions—and thus differential and continuous entropies—can violate this requirement, and so lead to negative values of the "mutual information". This is particularly worrying in the setting of training stochastic neural networks, as we often assume that an stochastic embedding is generated as a deterministic transformation of an input from a finite dataset to which a continuous perturbation is added. We provide an examples where naive computation without ensuring that the product and joint distributions of the two random variables have a well-defined Radon-Nikodym derivative yields negative mutual information.

Let $X \sim U([0, 0.1])$, $Z = X + R$ with $R \sim U(\{0, 1\})$. Then

$$\mathrm{I}[X; Z] = \mathrm{H}[X] = \log \tfrac{1}{10} \leq 0.$$

Generally, given X as above and an invertible function f such that $Z = f(X)$, $\mathrm{I}[X; Z] = \mathrm{H}[X]$ and can thus be negative. In a way, these cases can be reduced to (degenerate) expressions of the form $\mathrm{I}[X; X] = \mathrm{H}[X]$.

We can avoid these cases by adding independent continuous noise.

These examples show that not adding noise can lead to unexpected results. While they still yield finite quantities that bear a relation to the entropies of the random variables, they violate some of the core assumptions we have such that mutual information is always positive.

G. Experiment details

G.1. DNN architectures and hyperparameters

For our experiments, we use PyTorch (Paszke et al., 2019) and the Adam optimizer (Kingma & Ba, 2014). In general, we use an initial learning rate of 0.5×10^{-3} and multiply the learning rate by $\sqrt{0.1}$ whenever the loss plateaus for more than 10 epochs for CIFAR10. For MNIST and Permutation MNIST, we use an initial learning rate of 10^{-4} and multiply the learning rate by 0.8 whenever the loss plateaus for more than 3 epochs.

Sadly, we deviate from this in the following experiments: when optimizing the decoder uncertainty for *categorical* Z for CIFAR10, we used 5 epochs patience for the decoder uncertainty objective and a initial learning rate of 10^{-4}. We do not expect this difference to affect the qualitative results mentioned in section E when comparing to other objectives. We also only used 5 epochs patience when comparing the two cross-entropies on CIFAR10 in section 3.1. As this was used for both sets of experiments, it does not matter.

We train the experiments for creating the information plane plots for 150 epochs.

We use a batchsize of 128 for most experiments. We use a batchsize of 32 for comparing the cross-entropies for CIFAR10 (where we take 8 Dropout samples each), and a batchsize of 16 for MNIST (where we take 64 Dropout sampels each).

For MNIST, we use a standard Dropout CNN, following `https://github.com/pytorch/examples/blob/master/mnist/main.py`. For Permutation MNIST, we use a fully-connected model (for experiments with categorical Z in section E): $784 \times 1024 \times 1024 \times C$. For CIFAR10, we use use a ResNet18v2 (He et al., 2016b) with $K = 256$ dimensions for Z, and use a $K \times 10$ linear unit with softmax as decoder. When we need a stochastic model for CIFAR10 (for *continuous* Z), we add DropConnect with rate 0.1 to all but the first convolutional layers and Dropout with rate 0.1 before the final fully-connected layer. Because of memory issues, we reuse the Dropout masks within one batch. The model trains to 94% accuracy on CIFAR10.

For CIFAR10, we always remove the maximum pooling layer and change the first convolutional layer to have kernel size 3 with stride 1 and padding 1. We also use dataset augmentation during training, but not during evaluation on the training set and test set for purposes of computing metrics. We crop randomly after the padding the training images by 4 pixels in every direction and randomly flip images horizontally.

We generally sample 30 values of γ for the information plane plots from the specified ranges, using a log scale. For the ablation studies mentioned below, we sample 10 values of γ each. We always sample $\gamma = 0$ separately and run a trial with it.

Baselines were tuned by hand (without regularization) using grad-student descent and small grid searches.

G.2. Cluster setup & used resources

We make use of a local SLURM cluster (Jette et al., 2002). We run our experiments on GPUs (Geforce RTX 2080 Ti). We estimate reproducing all results would take 94 GPU days.

G.3. Differential entropies and noise

We demonstrate the importance of adding noise to continuous latents by constructing a pathological sequence of parameters which attain monotonically improving and unbounded regularized objective values (H[Z]) while all computing *the same function*. We use MNIST with a standard Dropout CNN as encoder, with $K = 128$ continuous dimensions in Z, and a $K \times 10$ linear layer as decoder. After every training epoch, we decrease the entropy of the latent by normalizing and then scaling the latent to bound the entropy. We multiply the weights of the decoder to not change the overall function.

G.4. Comparison of the surrogate objectives

G.4.1. Measurements of information quantities

Measuring information quantities can be challenging. As mentionend in the introduction, there are many complex ways of measuring entropies and mutual information terms. We can side-step the issue by making use of the bounds we have established and the zero-entropy noise we are injecting, and design experiments around that.

First, to estimate the Preserved Information $\mathrm{I}[X;Z]$, we note that when we use a deterministic model as encoder and only inject zero-entropy noise, we have $\mathrm{H}[Z \mid X] = 0$ and $\mathrm{I}[X;Z] = \mathrm{I}[X;Z] + \mathrm{H}[Z \mid X] = \mathrm{H}[Z]$. We use the entropy estimator from Kraskov et al. (2004, equation (20)) to estimate the Encoding Entropy $\mathrm{H}[Z]$ and thus $\mathrm{I}[X;Z]$.

To estimate the Residual Information $\mathrm{I}[X;Y \mid Z]$, we similarly note that $\mathrm{I}[X;Y \mid Z] = \mathrm{I}[X;Y \mid Z] + \mathrm{H}[Y \mid X] = \mathrm{H}[Y \mid Z]$. Instead of estimating the entropy using Kraskov et al. (2004), we can use the Decoder Cross-Entropy $\mathrm{H}_\theta[Y \mid Z]$ which provides a tighter bound as long as we also minimize $\mathrm{H}_\theta[Y \mid Z]$ as part of the training objective.

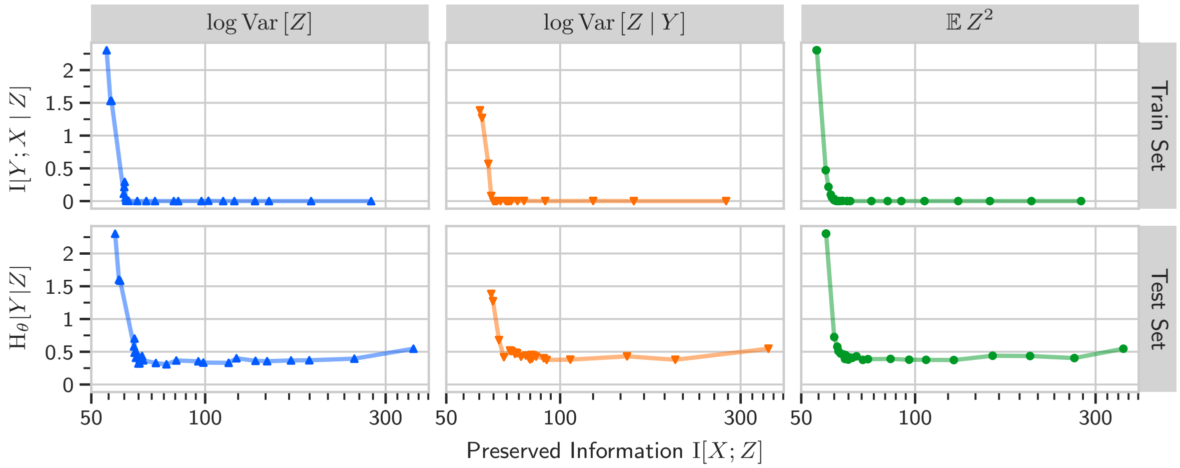


Figure G.1. Information Plane Plot of the latent Z *similar to Tishby & Zaslavsky (2015) but using a* ResNet18 *model on* CIFAR10 *using the different regularizes from section 3 (*without Dropout, but with zero-entropy noise*).* The dots are colored by γ. See section 4 for more details.

When we use stochastic models as encoder, we cannot easily compute I[X; Z] anymore. In the ablation study in the next section, we thus change the X axis accordingly.

Similarly, when we look at the trajectories on the test set instead of the training set, for example in figure G.2, we change the Y axis to signify the Decoder Uncertainty $\mathrm{H}_\theta[\mathrm{Y} \mid \mathrm{Z}]$. It is still an upper-bound, but we do not minimize it directly anymore.

At this point, it is important to recall that the Decoder Uncertainty is also the negative log-likelihood (when training with a single Dropout sample), which provides a different perspective on the plots. It makes it clear that we can see how much a model overfits by comparing the best and final epochs of a trajectory in the plot (marked by a circle and a square, respectively).

G.4.2. Ablation study

We perform an ablation study to determine whether injecting noise is necessary. Furthermore, we investigate the more interesting case of using a stochastic model as encoder, and if we can use a stochastic model without injecting zero-entropy noise.

We also investigate whether log Var[Z | Y] performs better when we increase batchsize as we hypothesized that a batchsize of 128 does not suffice as it leaves only ≈ 13 samples per class to approximate H[Z | Y]).

Figure G.3 shows a larger version of figure 1 for all three regularizers and also training trajectories on the test set. As described in the previous section, this allows us to validate that the regularizers prevent overfitting on the training set: with increasing γ, the model overfits less.

Figure G.6 and figure G.5 shows that injecting noise is necessary independently of whether we use Dropout or not. Regularizing with $\mathbb{E}\, \|Z\|^2$ still has a very weak effect. We hypothesize that floating-point precision issues might provide a natural noise source eventually. This would change the effectiveness of γ and might require much higher values to observe similar regularization effects as when we do inject zero-entropy noise.

Figure G.4 shows trajectories for a stochastic encoder (as described above with DropConnect/Dropout rate 0.1). It overfits less than a deterministic one.

Figure G.7 shows the effects of using higher dropout rates (using DropConnect/Dropout rates of 0.3/0.5). It overfits less than model with DropConnect/Dropout rates of 0.1/0.1.

The plots in figure G.10 show the effects of different γ with different regularizers more clearly. On both training and test set, one can clearly see the effects of regularization.

Overall, log Var[Z | Y] performs worse as a regularizer. In figure G.8, we compare the effect of doubling batchsize. Indeed,

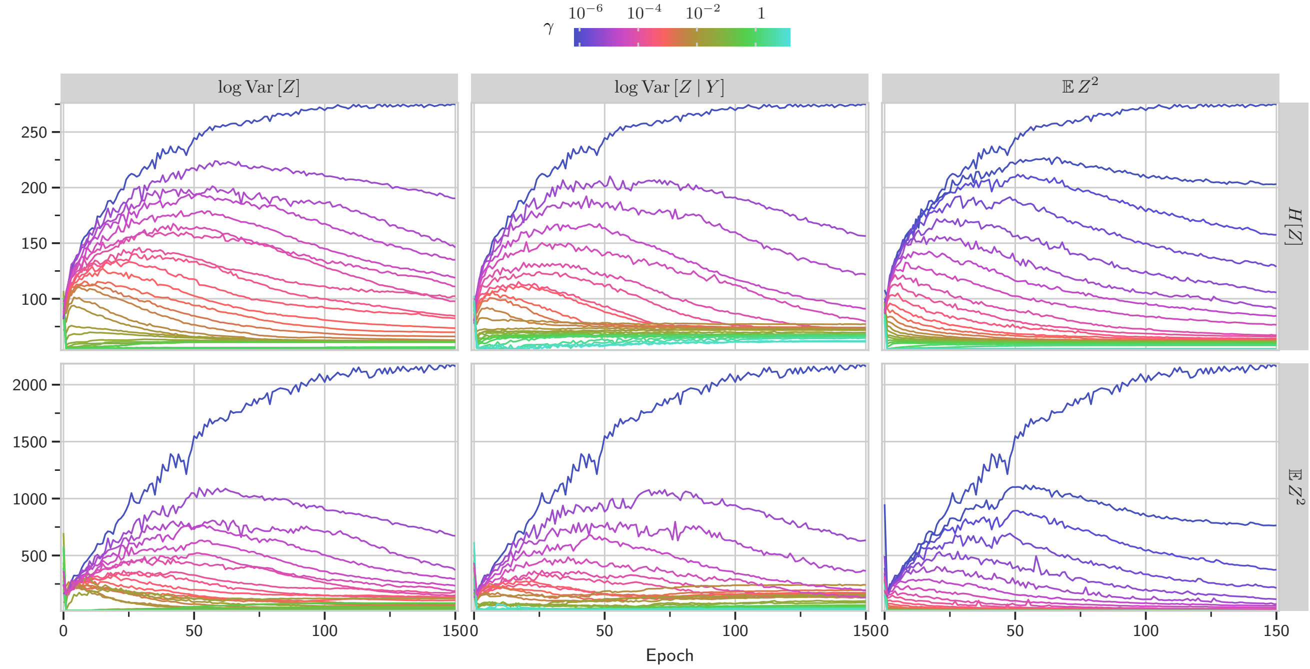


Figure G.2. Entropy estimates while training with different γ and with different surrogate regularizers on CIFAR10 *with a* ResNet18 *model.* Entropies are estimated on training data based on Kraskov et al. (2004). Qualitatively all three regularizers push H[Z] and H[Z | Y] down. H[Z | Y] is not shown here because it always stays very close to H[Z]. $\mathbb{E}\,\|Z\|^2$ tends to regularize entropies more strongly for small γ. See section 4 for more details.

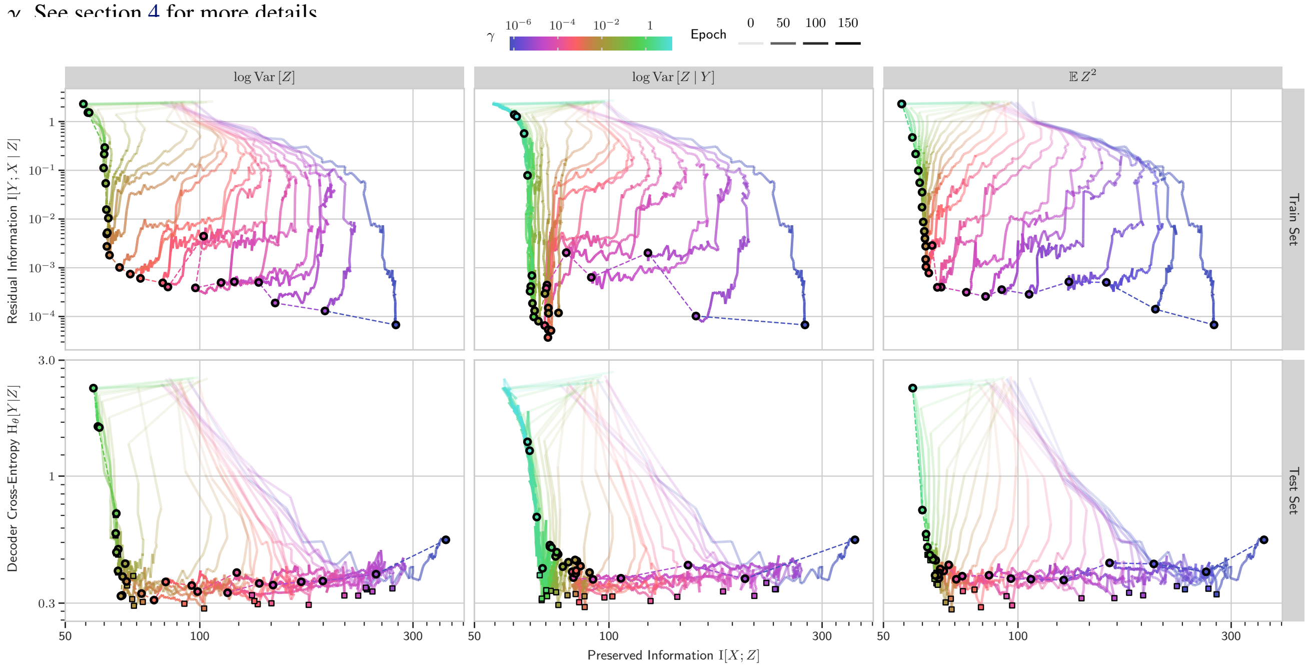


Figure G.3. Without dropout but with zero-entropy noise: *Information Plane Plot of training trajectories for ResNet18 models on CIFAR10 and different regularizers.* The trajectories are colored by their respective γ; their transparency changes by epoch. Compression (Preserved Information ↓) trades-off with performance (Residual Information ↓). See section 4. The circle marks the final epoch of a trajectory. The square marks the best epoch (Residual Information ↓↓).

log Var[Z | Y] performs better with higher batchsize and looks closer to log Var[Z].

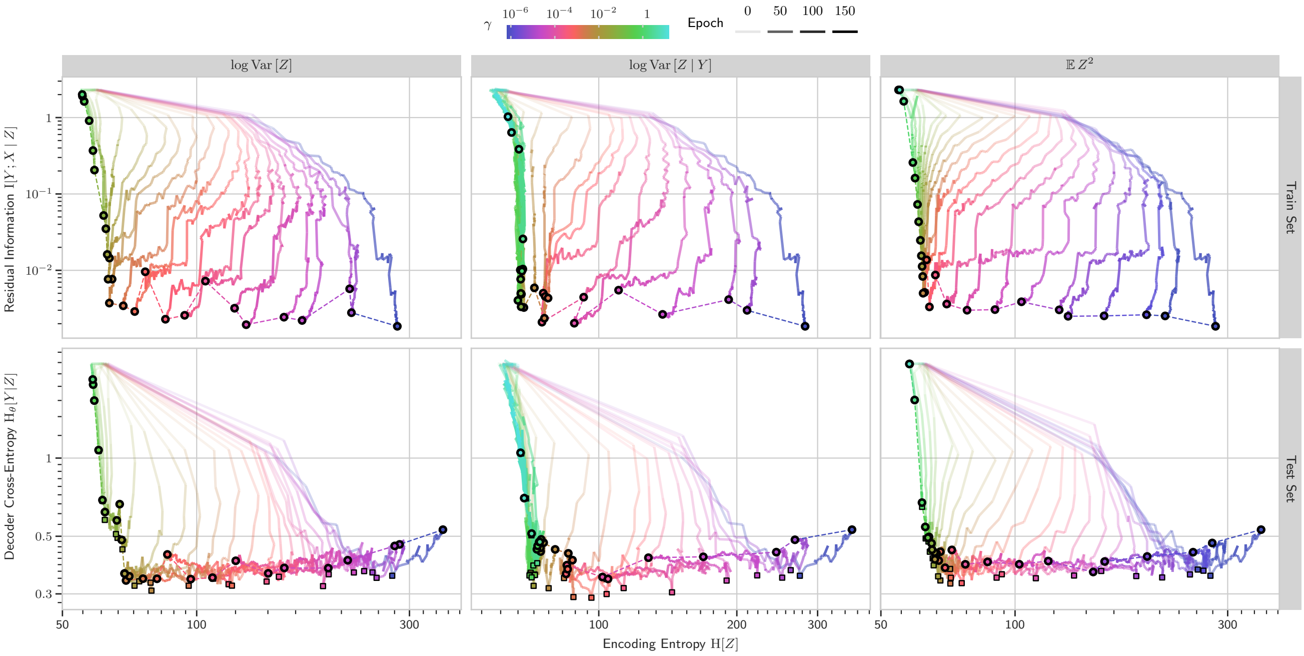


Figure G.4. With dropout and with zero-entropy noise: *Information Plane Plot of training trajectories for ResNet18 models on CIFAR10 and different regularizers.* The trajectories are colored by their respective γ; their transparency changes by epoch. Compression (Preserved Information ↓) trades-off with performance (Residual Information ↓). See section 4. The circle marks the final epoch of a trajectory. The square marks the best epoch (Residual Information ⇊).

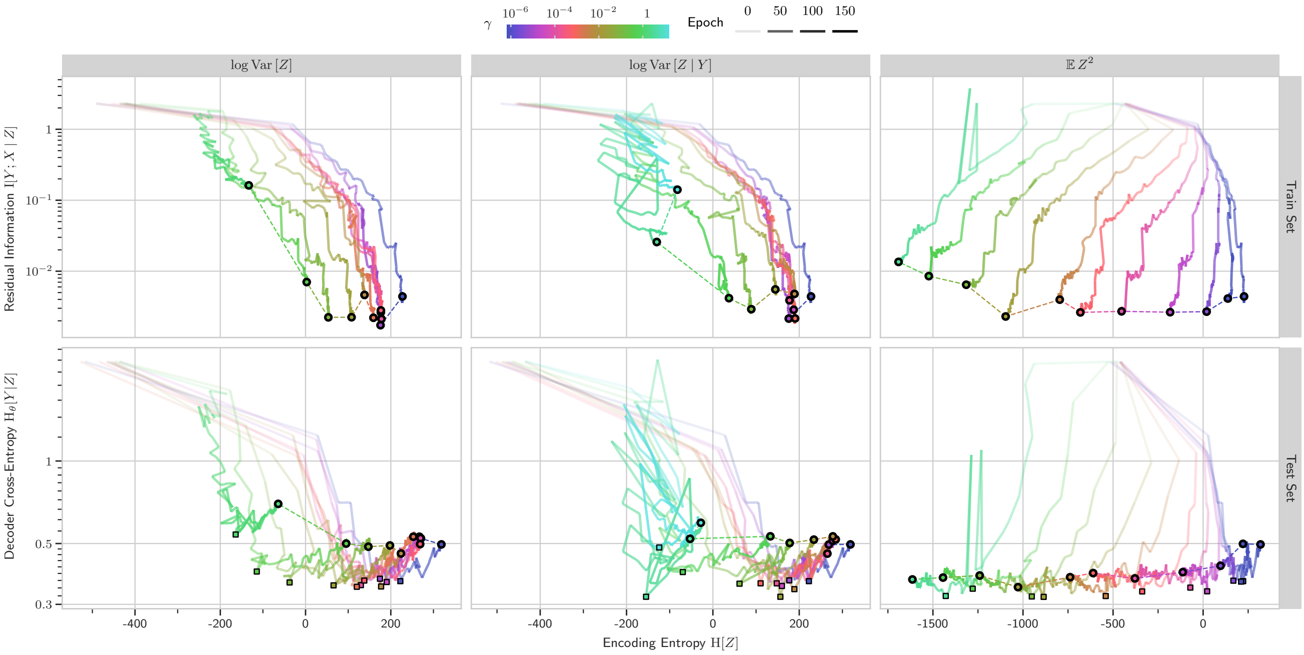


Figure G.5. With dropout but without zero-entropy noise: *Information Plane Plot of training trajectories for ResNet18 models on CIFAR10 and different regularizers.* The trajectories are colored by their respective γ; their transparency changes by epoch. Compression (Preserved Information ↓) trades-off with performance (Residual Information ↓). See section 4. The circle marks the final epoch of a trajectory. The square marks the best epoch (Residual Information ⇊).

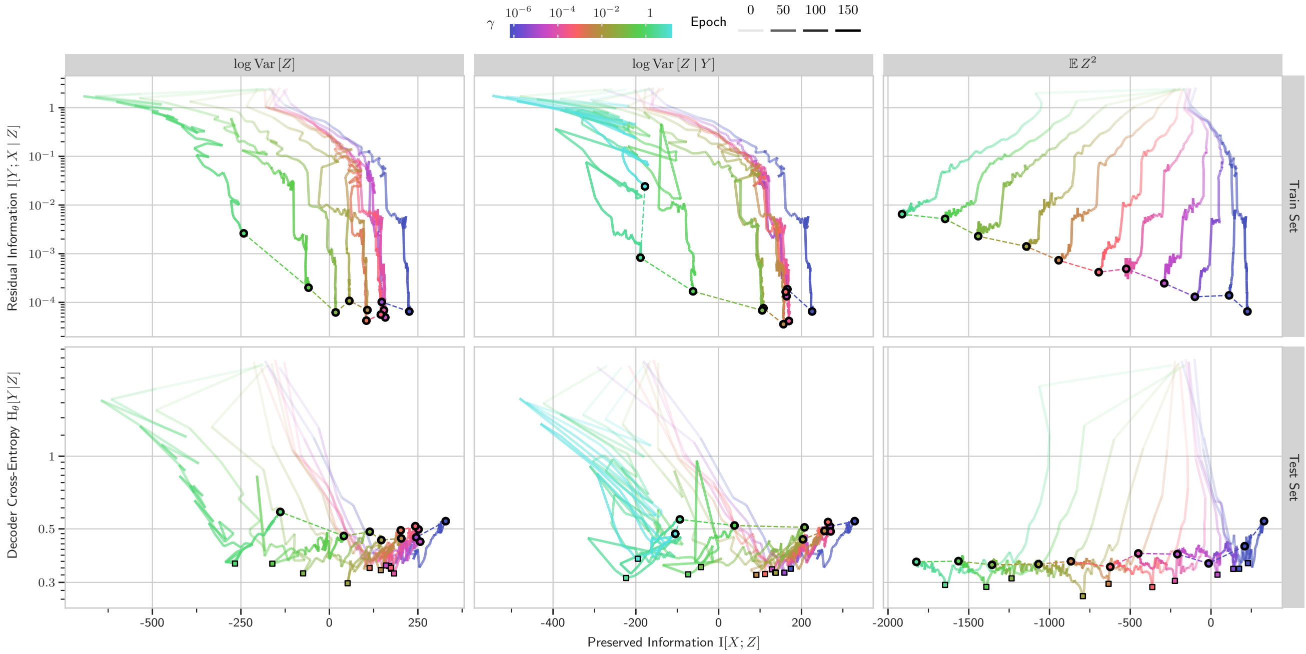


Figure G.6. Without dropout and without zero-entropy noise: *Information Plane Plot of training trajectories for ResNet18 models on CIFAR10 and different regularizers.* The trajectories are colored by their respective γ; their transparency changes by epoch. Compression (Preserved Information ↓) trades-off with performance (Residual Information ↓). See section 4. The circle marks the final epoch of a trajectory. The square marks the best epoch (Residual Information ↓↓).

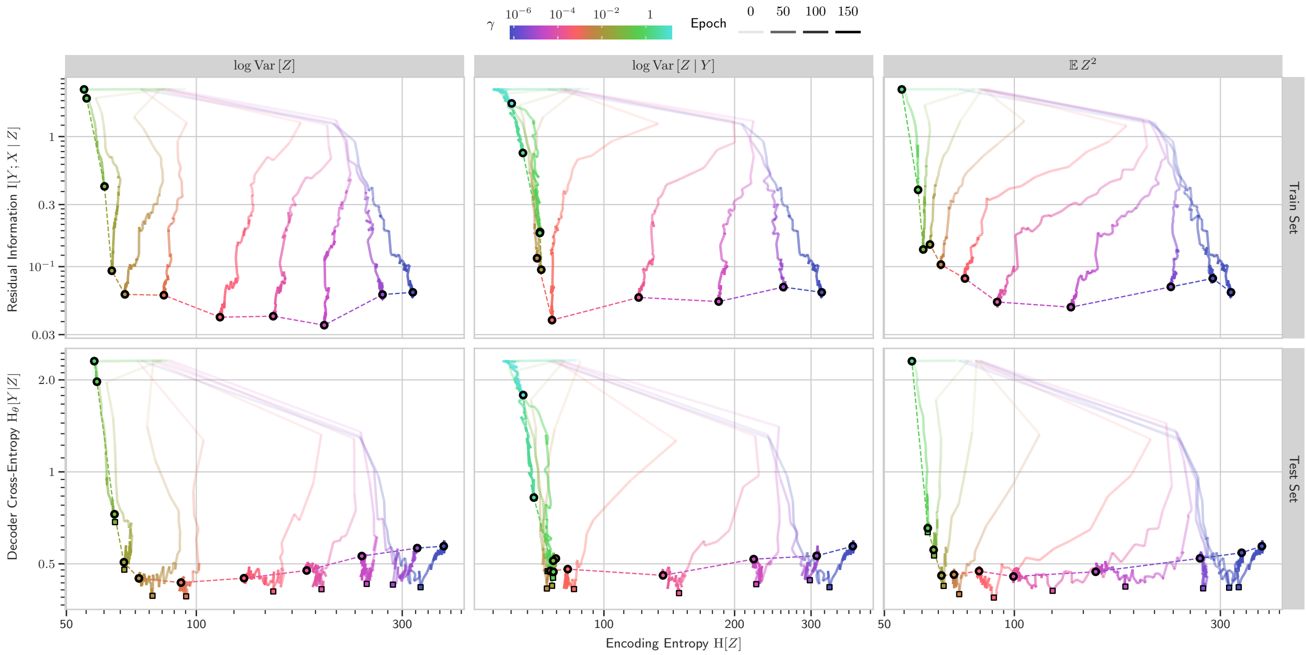


Figure G.7. With more dropout and zero-entropy noise: *Information Plane Plot of training trajectories for ResNet18 models on CIFAR10 and* log Var[Z | Y] *regularizer with batchsizes 128 and 256.* The trajectories are colored by their respective γ; their transparency changes by epoch. Compression (Preserved Information ↓) trades-off with performance (Residual Information ↓). See section 4. The circle marks the final epoch of a trajectory. The square marks the best epoch (Residual Information ↓↓). A DropConnect rate of 0.3 and Dropout rate of 0.4 were used instead of 0.1 for each.

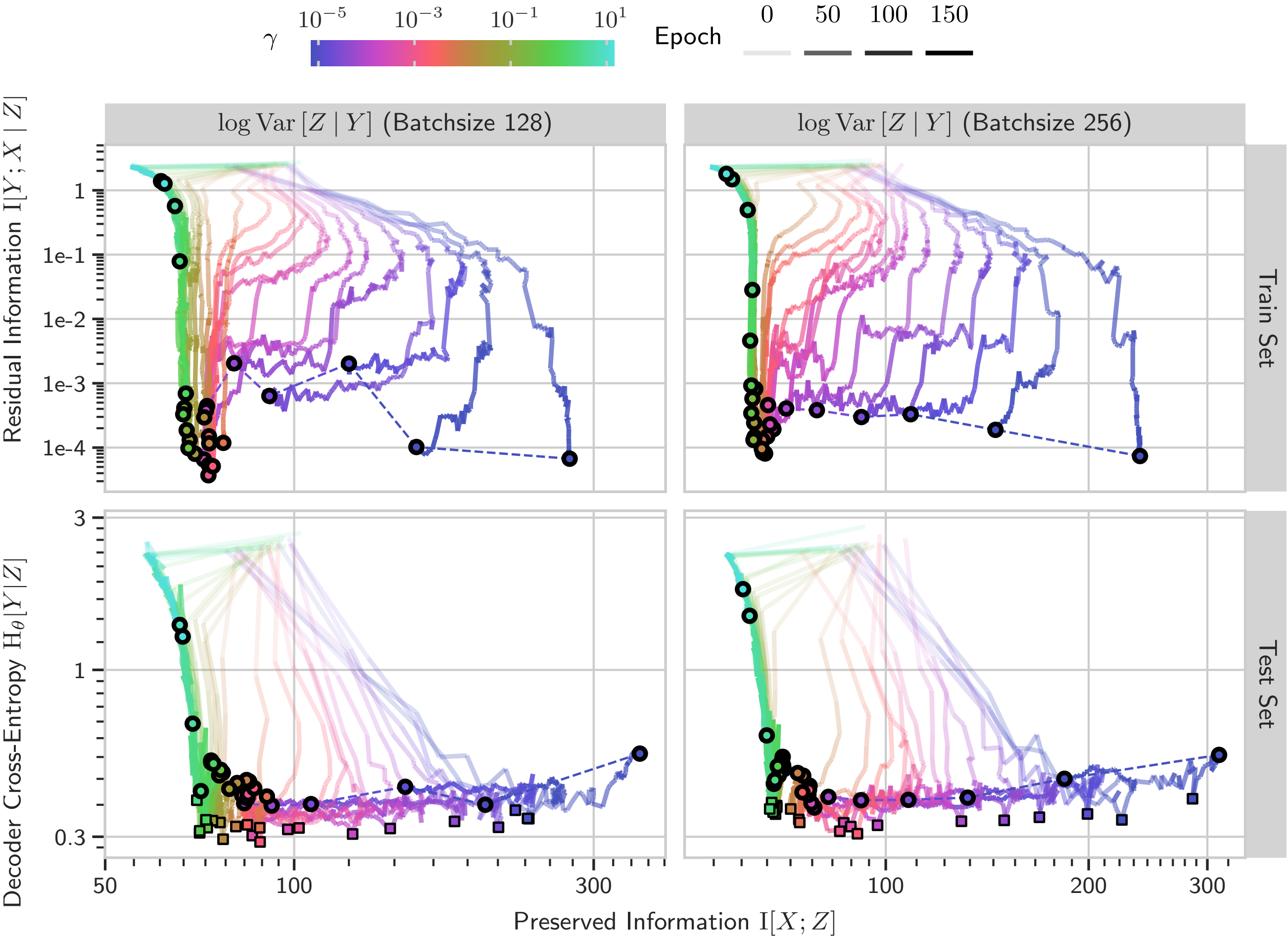


Figure G.8. Without dropout but with zero-entropy noise: *Information Plane Plot of training trajectories for ResNet18 models on CIFAR10 and* log Var[Z | Y] *regularizer with batchsizes 128 and 256.* The trajectories are colored by their respective γ; their transparency changes by epoch. Compression (Preserved Information ↓) trades-off with performance (Residual Information ↓). See section 4. The circle marks the final epoch of a trajectory. The square marks the best epoch (Residual Information ⇊).

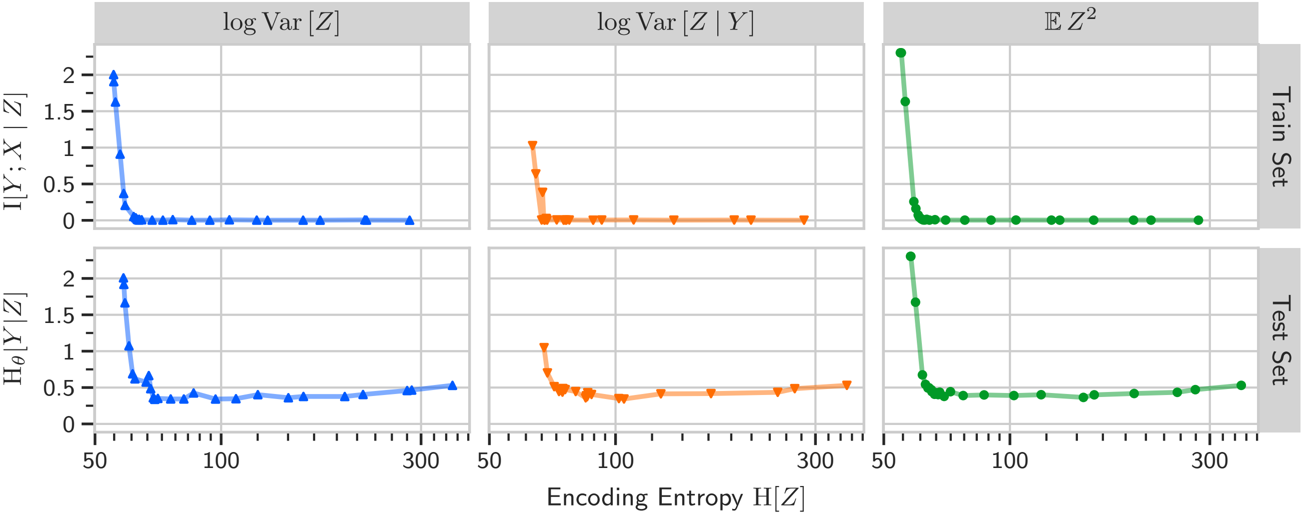
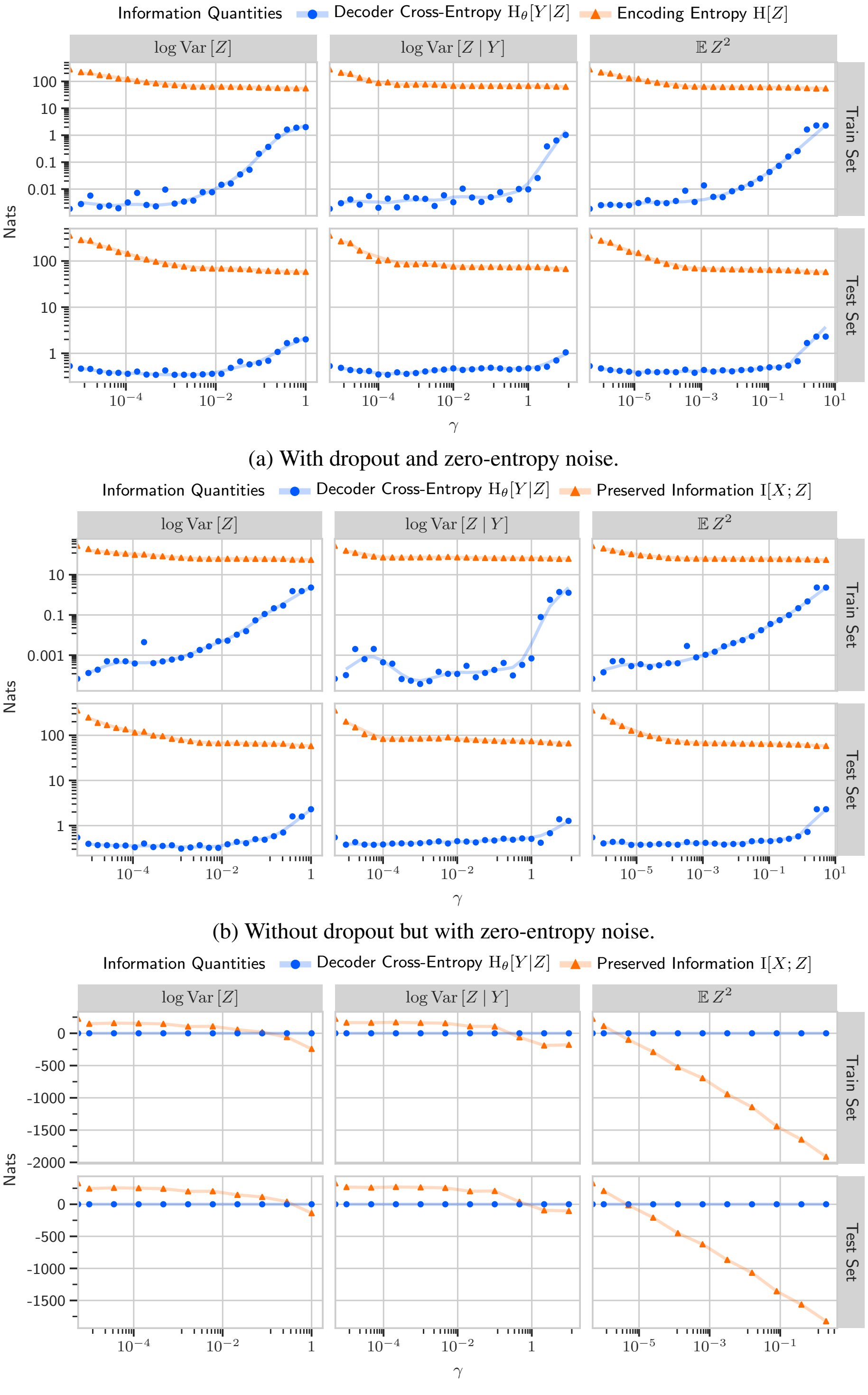


Figure G.9. Information Plane Plot of the latent Z *similar to Tishby & Zaslavsky (2015) but using a* ResNet18 *model on* CIFAR10 *using the different regularizes from section 3 (*with Dropout and zero-entropy noise*).* The dots are colored by γ. See section 4 for more details.



(a) With dropout and zero-entropy noise.

(b) Without dropout but with zero-entropy noise.

(c) Without dropout and without zero-entropy noise.

Figure G.10. Information quantites for different γ at the end of training for ResNet18 models on CIFAR10 and $\log \mathrm{Var}[Z \mid Y]$ *regularizer with batchsizes 128 and 256.* Compression (Preserved Information ↓) trades-off with performance (Residual Information ↓). See section 4.

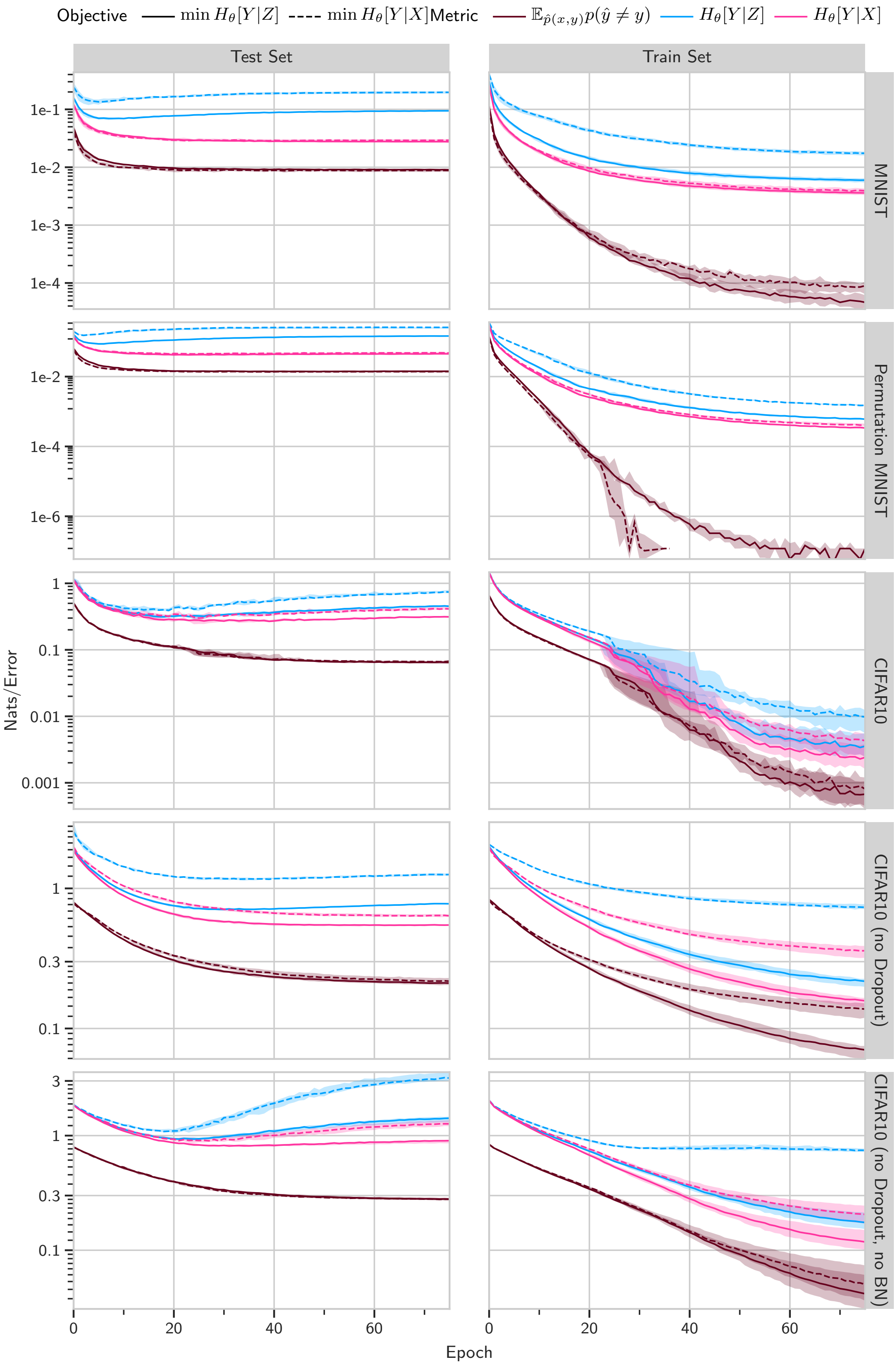


Figure G.11. Training error probability, Decoder Cross-Entropy $\mathrm{H}_\theta[\mathrm{Y} \mid \mathrm{Z}]$ *and Prediction Cross-Entropy* $\mathrm{H}_\theta[\mathrm{Y} \mid \mathrm{X}]$ *with continuous* Z. $K = 100$ dimensions are used for Z, and we use Dropout to obtain stochastic models. Minimizing $\mathrm{H}_\theta[\mathrm{Y} \mid \mathrm{Z}]$ (solid) leads to smaller cross-entropies and lower training error probability than minimizing $\mathrm{H}_\theta[\mathrm{Y} \mid \mathrm{X}]$ (dashed). This suggests a better data fit, which is what we desire for a loss term. We run 8 trials each and plot the median with confidence bounds (25% and 75% quartiles). See section 3.1 and 4 for more details.

H. Large Version of the Mickey Mouse I-Diagram

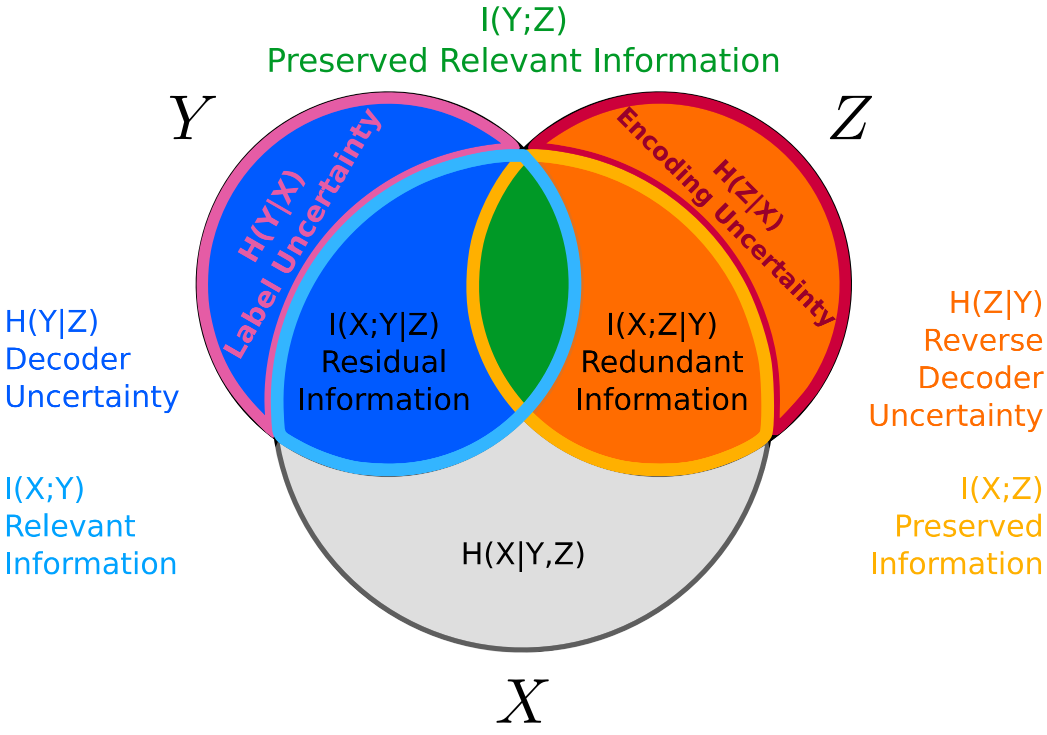


Figure H.1. Mickey Mouse I-diagram.