INFORMATION PLANE ANALYSIS FOR DROPOUT NEURAL NETWORKS

Anonymous authors
Paper under double-blind review

ABSTRACT

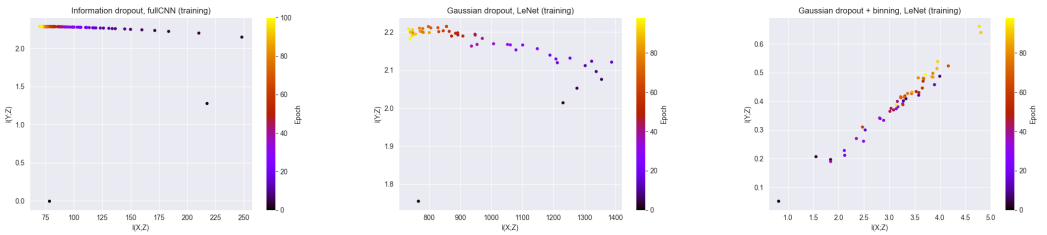
The information theoretic framework promises to explain the predictive power of neural networks. In particular, the information plane analysis, which measures mutual information (MI) between input and representation as well as representation and output, should give rich insights into the training process. This approach, however, was shown to strongly depend on the choice of estimator of the MI: measuring discrete MI does not capture the nature of deterministic neural networks and continuous data distributions, and different approaches for discretization arbitrarily change results. On the other hand, measuring continuous MI for a deterministic network is not mathematically meaningful. In this work we show how the stochasticity induced by dropout layers can be utilized to estimate MI in a theoretically sound manner. We demonstrate in a range of experiments that this approach enables a meaningful information plane analysis for the large class of dropout neural networks that is widely used in practice.

1 INTRODUCTION

The information bottleneck hypothesis for deep learning conjectures two phases of training feed-forward neural networks (Shwartz-Ziv and Tishby, 2017): the fitting phase and the compression phase. The first corresponds to extracting information from the input into the learned representations, so that the mutual information (MI) between inputs and hidden representations grows. The latter corresponds to forgetting the information that is not needed to predict the target, reflected in a decrease of the MI between hidden representations and inputs, while MI between representations and targets stays the same or grows. The phases can be observed via an information plane analysis, i.e., by analyzing the development of MI between inputs and representations in contrast to the MI between representations and targets during training (see Fig. 1 for an example). For an overview we refer the reader to Geiger (2021).

While being elegant and plausible, this hypothesis is challenging to investigate empirically. As shown by Amjad and Geiger (2019), the MI of a deterministic neural network is infinite, if the input distribution is continuous. The standard approach is therefore to assume the input distribution to be discrete (e.g., equivalent to the empirical distribution of the dataset S at hand) and to discretize the real-valued hidden representations by binning to allow for non-trivial measurements, i.e. to avoid that the MI always takes the maximum value of $\log(|S|)$ (Shwartz-Ziv and Tishby, 2017). In this discrete setting the MI theoretically gets equivalent to the Shannon entropy of the hidden representations and thus MI decrease is equivalent to geometrical compression (Basirat et al., 2021). Moreover, the binning-based estimate highly dependents on the chosen bin size (Ross, 2014). To instead work with continuous input distributions, Goldfeld et al. (2019) suggest to replace deterministic neural networks by stochastic ones via adding Gaussian noise to each of the hidden representations. This kind of stochastic networks is rarely used in practice, which limits the insights brought by the analysis. However, another class of stochastic neural networks, namely those using dropout, is heavily used in practice due to their effective regularizing properties.

This leads to the core questions investigated in this work: Can we obtain accurate and meaningful MI estimates in neural networks with dropout noise? And if so, do information planes built for dropout networks confirm the information bottleneck hypothesis? To that end, our main contributions are the following: We present a theoretical analysis showing that binary dropout noise does not introduce sufficient stochasticity that allows us to estimate the MI between inputs and representations. At the



(a) Information dropout (b) Gaussian dropout (our) (c) Gaussian dropout (binning)

Figure 1: Information planes for the representation with information and Gaussian dropout layer of a LeNet network. Compared to an information plane analysis based on a (discrete) binning estimation of the MI, our approach (both with Gaussian and information dropout) clearly shows compression.

same moment we confirm that dropout noise with any continuous distribution not only results in finite MI, but also provides an elegant way to estimate it. In particular, Gaussian dropout is known to benefit generalization even more than binary dropout (Srivastava et al., 2014). We empirically analyse the quality of the MI estimation in the setup with Gaussian dropout in a range of experiments on benchmark neural networks and datasets.

2 Mutual Information Estimation for Neural Networks

In this paper we use the following notations: $H(A)$ denotes the Shannon entropy of a discrete random variable A whose distribution is denoted p_A; $h(B)$ is the differential entropy of a continuous random variable B whose distribution is described by the probability density function p_B; $I(A;B)$ is the MI between random variables A and B; $X \in \mathcal{X}$ and $Y \in \mathcal{Y}$ are the random variables describing inputs to a neural network and corresponding targets; $f(X)$ is the result of the forward pass of the input through the network to the hidden layer of interest; Z is an n-dimensional random variable describing the hidden representations that are being analyzed. Note, that while $f(\cdot)$ describes a deterministic function implemented by a neural network, Z may contain additional stochasticity, e.g., via additive or multiplicative noise.

There are several equally valid choices to theoretically describe the distributions of X and Z. The caveats of different approaches with respect to measuring MI were discussed widely in the literature (Saxe et al., 2019; Kolchinsky et al., 2019). In the following we shortly summarize the problems motivating the approach proposed in this paper. It should be noted that these problems do not appear for the MI measured between representations and targets, since they are not connected via a deterministic function. We therefore concentrate on $I(X;Z)$, where we assume that Z does not contain additional stochasticity, i.e., we describe a deterministic neural network with $Z = f(X)$.

The first option is to assume the input to be drawn from a discrete distribution. This view makes it easy to use a finite dataset S at hand to describe the distribution and is supported by the finiteness of the accuracy of the used computational resources (Lorenzen et al., 2021). More precisely, it describes the input distribution as a uniform on the training data and fixes the discretization corresponding to the computer precision (or other selected bin size for ease of experimental setup). In this case Z is discrete as well and the MI between X and Z is computed like $I(X;Z) = H(Z) - H(Z|X) = H(Z) - H(f(X)|X) = H(Z) - 0 = H(Z)$, where we assume that the network forward function $f(\cdot)$ is deterministic. Thus, estimated MI between input and representation essentially corresponds to the entropy of the representation, which is equal to the entropy of the dataset $\log |S|$ unless the forward pass maps some of the different data points from the dataset to the same value in the latent space.

The second option is to assume X to be drawn from a continuous distribution. This is more aligned to the common description of real world data where we assume that it is drawn from some data generating distribution $\mathcal{D}$ over the whole space of $\mathcal{X} \times \mathcal{Y}$. In this case, if the network transformation $f(\cdot)$ results in a discrete distribution of representations Z one can still use the decomposition $I(X,Z) = H(Z) - H(Z|X)$ to estimate and describe the MI based on Shannon entropy. As shown

in Theorem 1 of Amjad and Geiger (2019) this is however not the case for neural networks with commonly used activation functions. If $f(\cdot)$ is deterministic and Z is not purely discrete, the MI between X and Z is infinite. This happens because the joint distribution is not absolutely continuous with respect to the product of the marginals. [1] Thus, the approach to estimate the MI of continuous input and hidden representation in practice is to modify the space of inputs and/or representations to be discrete, e.g., by binning. For example, binning the representation Z to $\hat{Z}$ again yields $I(X; \hat{Z}) = H(\hat{Z})$, and the qualitative behavior of this entropy will be defined by properties of activation functions and selected bin size (Saxe et al., 2019).

From the discussion above it follows that estimating $I(X; Z)$ in deterministic neural networks requires discretization and always reflects geometric properties of representation space or errors of the used estimators. As a solution to the discussed challenges, in their work Goldfeld et al. (2019) propose to introduce zero mean Gaussian noise added to the representations during training. This transforms a deterministic neural network into a stochastic one, but essentially does not change the training results and predictive abilities of the model. Such setup allows for applying estimation techniques for entropy values, based on Monte Carlo sampling, and then uniting them into the estimation of MI, with a known error bound on the estimation. The authors claim that in such setup MI estimation is tracking the clustering of the representations, thus also their geometric properties. It allows for the theoretical explanation and justification of the controversial information compression phase hypothesis (Shwartz-Ziv and Tishby, 2017): When the MI between input and representation is dropping, it means that noise-induced Gaussians centered in the representations are overlapping more. Then it is becoming harder to distinguish between same label inputs by their representations, which means lower MI with input. Nevertheless, measuring geometric properties of the representation space via MI, as the empirical results of Goldfeld et al. (2019) demonstrate, still vague about the most interesting aspect: Can information plane development be predictive about generalization abilities of the trained network?

Overall, the existing research indicates the need for a method to measure MI for networks with continuous input distributions, since the known approaches to MI estimation lead to an analysis of the development of neural network representations that is limited to geometric interpretation (Geiger, 2021). Based on the knowledge about benefits of the stochasticity added to the training of the neural networks, e.g., in the form of dropout, we suggest to concentrate on the setup where representations are not deterministic, which allows for theoretically justified values of MI. While additive noise is a valuable step towards measuring such MI, it requires noise introduced on all the layers of a network and samples from multiple copies of the network, moreover, it is centered around geometrical meaning of the information compression. In this work, we analyse the stochastic representation obtained via a dropout layer, and show that such noise has a great potential for revealing the underlying information flow of the neural network.

3 Mutual Information in Dropout Networks

As discussed in the previous section, the MI between inputs and hidden representations of deterministic networks is infinite, if we assume the input distribution to be described by some continuous probability density function. To overcome this problem, some form of stochasticity has to be introduced. While adding noise to activations (Goldfeld et al., 2019) indeed allows to compute MI, it is not a widely used type of the neural networks. Neural networks with dropout instead are one of the most popular classes of neural networks used in practice and are indeed stochastic in nature: Adding a dropout layer to a neural network corresponds to multiplying the hidden representation with some form of random noise. Formally, denoting the random noise by a random variable D of the same dimension as $f(X)$, the hidden representation is $Z = f(X) \circ D$, where $\circ$ denotes element-wise multiplication. In the most basic form D follows a Bernoulli distribution (Srivastava et al., 2014). Currently, such binary dropout is widely used since it allows us to intuitively understand it as "turning off" a fraction of neurons during training. There is a variety of other dropout layers, including multiplicative Gaussian noise, fast dropout (Wang and Manning, 2013), or variational dropout (Kingma et al., 2015). Information dropout (Achille and Soatto, 2018) is a variant that allows us to approximate mutual information in a closed form—while the approximation itself is rather crude, it helps to train a network and is therefore used in the original work as a weighted reg-

[1] See Appendix A.1 for a more detailed discussion.

ularization term complementing the cross-entropy loss. In order to obtain such closed form, dropout noise is sampled from a log-normal distribution and a prior distribution on representations is chosen depending on the activation function (ReLU or Softplus). We provide details on the derivation in Appendix A.2.

Instead of approximating MI for regularization, our goal is to compute MI accurately using dropout as a source of stochasticity. For that, we first show a negative result: binary dropout leads to MI being infinite in theory. We then show that any continuous dropout prevents MI from becoming infinite and thus allows for its meaningful estimation.

3.1 Binary Dropout

We start by analysing binary dropout which forces individual neurons to be "turned off" with some (discrete) probability. More formally, the output of each neuron is multiplied with an independent Bernoulli random variable that is equal to 1 with a predefined probability p. The following theorem shows that this kind of (combinatorial) stochasticity is insufficient to prevent MI from becoming infinite.

Theorem 3.1. *In the setting of (Amjad and Geiger, 2019, Th. 1), let the output $f(\cdot)$ of a hidden layer be parameterized as a deterministic neural network with N neurons, let $B \in \{0,1\}^N$ be the set of independent Bernoulli random variables characterizing the dropout pattern, and let Z be the stochastic representation obtained by applying the dropout pattern to the neural network function $f(\cdot)$. Then it holds that $I(X;Z) = \infty$.*

Proof. Using the chain rule of MI, we have

$$\begin{aligned} I(X;Z) &= I(X;Z,B) - I(B;X|Z) \\ &= I(X;Z|B) + I(B;X) - I(B;X|Z) \geq I(X;Z|B) - H(B) \end{aligned}$$

where the last line follows from dropping $I(B;X)$ since they are independent and the fact that $I(B;X|Z) \leq H(B)$. Having $B \in \{0,1\}^N$ as a discrete random variable, it immediately follows that $H(B) \leq N \log 2$. Now note that

$$I(X;Z|B) = \sum_{b \in \{0,1\}^N} \mathbb{P}(B=b) I(X;Z|B=b).$$

Since the Bernoulli variables are independent, positive probability mass is assigned to $B = (1,1,\ldots,1)$, i.e., to the case where all neurons are active. Evidently, when $b = (1,1,\ldots,1)$ it follows that $Z = f(X)$. Thus, with (Amjad and Geiger, 2019, Th. 1)

$$I(X;Z|B) \geq \mathbb{P}(B=(1,1,\ldots,1)) I(X;f(X)) = \infty$$

and thus $I(X;Z) = \infty$. □

While the Bernoulli distribution guarantees that $B = (1,1,\ldots,1)$ always has non-zero probability, other distributions over $\{0,1\}^N$ might not have this property. Theorem 3.1 can however be generalized to arbitrary distributions over $\{0,1\}^N$:

Theorem 3.2. *In the setting of (Amjad and Geiger, 2019, Th. 1), let the output $f(\cdot)$ of a hidden layer be parameterized as a deterministic neural network with N neurons, let $B \in \{0,1\}^N$ be the binary random vector characterizing the dropout pattern, and let Z be the stochastic representation obtained by applying the dropout pattern to the neural network defining $f(\cdot)$. Then, it either holds that $I(X;Z) = \infty$ or that $I(X;Z) = 0$ if the dropout patterns almost surely disrupt information flow through the network.*

Proof. If the binary dropout is such that nonzero probability is assigned to the dropout mask $B = (1,1,\ldots 1)$, then the statement of the theorem follows as in the proof of the previous theorem.

Assume now that B is such that zero mass is assigned to $B = (1,1,\ldots,1)$. To treat this case, we suppose that the distribution of X has a portion with a continuous probability density function on a compact set and that the neural network has activation functions that are either bi-Lipschitz or continuously differentiable with a strictly positive derivative (following the requirements of Amjad

and Geiger (2019, Th. 1)). Then, we obtain $I(X; f(X)) = \infty$ from (Amjad and Geiger, 2019, Th. 1) for almost all parameterizations of the neural network. Under this setting, $f(X) \circ b$ is again a neural network with activation functions that are either bi-Lipschitz or continuously differentiable with a strictly positive derivative. Assuming that b is such that the input of the network is not completely disconnected from the considered layer, for this pattern we have $I(X; Z|B = b) = \infty$. Otherwise, we obviously have $I(X; Z|B = b) = 0$. The statement of the theorem follows from taking the expectation over all patterns b. □

Thus, binary dropout cannot be used to estimate MI reliably and is not helpful for the information plane analysis. The main obstacle is the finiteness of the combinatorial space induced by the binary noise on the space of representations.

3.2 Dropout with Continuous Noise

As proposed by Srivastava et al. (2014), dropout can be implemented using continuous Gaussian noise with mean vector $\mu = \mathbf{1}$. Formally, the hidden representation Z is given by $f(X) \circ D$, where $D \sim \mathcal{N}(\mathbf{1}, I\sigma^2)$ with σ^2 being the selected variance of the noise. In contrast to binary noise sampled from a discrete distribution, multiplicative Gaussian noise turns the joint distribution of (Z, X) to be absolutely continuous with respect to the marginals of Z and X allowing for finite values of MI between the input X and the hidden representation. The following theorem states that the MI between input and hidden dropout representation indeed is finite.

Theorem 3.3. *Let p_x be a probability density function on X, $f(\cdot)$ be parameterized by a deterministic neural network with Lipschitz activation functions, and let $Z = f(X) \circ D$, with $D \sim \mathcal{N}(\mathbf{1}, I\sigma^2)$. Then, under assumption of $\|X\|_\infty < \infty$, it holds that $I(X, Z) < \infty$.*

Proof. We first restrict our attention to representations Z for which all dimensions are different from zero. This can be done w.l.o.g. as the following argument shows. Specifically, suppose that $Z = (Z_1, \dots, Z_n)$ and that $B = (B_1, \dots, B_n)$ with $B_i = 0$ if $Z_i = 0$ and $B_i = 1$ otherwise. Clearly, B is a function of Z, hence $I(X; Z) = I(X; Z, B) = I(B; X) + I(Z; X|B)$. Since B is binary, we have that $I(X; B) \leq H(B) \leq n \ln 2$. Letting $Z_B = (Z_i | i\colon B_i = 1)$ denote the sub-vector of non-zero elements of Z, we thus obtain

$$I(X; Z) \leq n \ln 2 + \sum_b \mathbb{P}(B = b) I(Z_b; X). \tag{1}$$

Therefore, $I(X; Z)$ is finite iff $I(Z_b; X) = I(Z; X|B = b)$ is finite B-almost surely. We thus now fix an arbitrary $B = b$ and continue the proof for $Z = Z_b$.

Since the distribution of Z is continuous, we consider the following decomposition of MI into differential entropies: $I(X; Z) = h(Z) - h(Z|X)$. The entropy of the representations $h(Z)$ is upper-bounded by the entropy of a Gaussian with the same covariance matrix Σ as the distribution of $Z = (Z_1, \dots, Z_n)$, i.e., by $\frac{n}{2} \ln(2\pi) + \frac{1}{2} \ln(\det(\Sigma)) + \frac{1}{2} n$, where n is the dimensionality of Z. From Hadamard's inequality and since Σ is positive semidefinite it follows that $\det(\Sigma) \leq \prod_{i=1}^n \sigma_{ii}$, where σ_{ii} are diagonal elements of the covariance matrix, i.e., $\sigma_{ii} = Var[Z_i]$. Due to the independence of $f(X)$ and D the variance of Z_i is given by

$$Var[Z_i] = Var[f(X)_i] Var[D_i] + Var[f(X)_i](\mathbb{E}[D_i])^2 + Var[D_i](\mathbb{E}[f(X)_i])^2 \ .$$

From the fact that the activation functions are Lipschitz, it follows that the composition $f(\cdot)$ of activation functions and linear maps is also Lipschitz and has Lipschitz constant L. Thus, $Var[f(X)_i] \leq 2L Var[X_i]$. Moreover, since X_i is bounded and $f(\cdot)$ is Lipschitz, $E[f(X)_i]$ is bounded as well. Therefore, $Var[Z_i]$ is bounded for all i and the determinant of the covarinace matrix is finite, leading to $h(Z) < \infty$.

It remains to show that the $h(Z|X) > -\infty$. Due to the independence of D_i and D_j, for all $i, j = 1, \dots n, i \neq j$, the conditional entropy of Z factorises in the sum of conditional entropy of its components, i.e., $h(Z|X) = \sum_{i=1}^n h(Z_i|X)$. The conditional entropy $h(Z_i|X)$ is defined as

$$h(Z_i|X) = -\int p_{z_i,x} \log(p_{z_i|x}) dz_i dx = -\int p_x (p_{z_i|x} \log(p_{z_i|x})) dz_i dx = \mathbb{E}_x[h(p_{z_i|x})] \ .^2$$

[2]Note that we do not change the order of integration here (it is only possible if the integral is finite, which is exactly what we are proving here), but simply use the definition of joint probability.

Since for every fixed $X = x$, each $p_{z_i|x}$ is a Gaussian with mean $f(x)_i$ and variance $|f(x)_i| \cdot \sigma$, i.e., $Z_i|X = x \sim \mathcal{N}(f(x)_i, (|f(x)_i| \cdot \sigma)^2)$, where $f(x)_i$ is the i-th component of $f(x)$ that is being multiplied by the i-th component of D, the entropy is given by

$$h(Z_i|X = x) = \mathbb{E}_x[\log(|f(x)_i|\sigma\sqrt{2\pi e})] = \mathbb{E}_x[\log(|f(x)_i|)] + \mathbb{E}_x[\log(\sigma\sqrt{2\pi e})] =$$
$$= \mathbb{E}_x[\log(|f(x)_i|)] + \log(\sigma\sqrt{2\pi e})$$

Remember that all $|f(x)_i|$ are strictly positive. By the law of the unconscious statistician we can exchange the integral with the following one $\mathbb{E}_x[\log(|f(x)|_i)] = \int \log(\hat{x}) p_{\hat{x}} d\hat{x}$, where $p_{\hat{x}}$ corresponds to the density function of $|f(x)|_i$. Under assumption that $p_{\hat{x}} < K$, where K is some number, we get $\int \log(\hat{x}) p_{\hat{x}} d\hat{x} < K \int \log(\hat{x}) d\hat{x} = K\hat{x}(\log(\hat{x}) - 1)$ which is finite. The properties of $p_{\hat{x}}$ are defined by the unknown p_x and unknown $f(\cdot)$. Seeing $|f(x)_i|$ as a result of mapping an input vector x to a scalar, we can write down

$$p_{\hat{x}} = \int_{\mathcal{X}} p_x \delta(\hat{x} - |f(x)|_i) dx \ ,$$

where δ is a delta Dirac function. Essentially, if we assume that p_x is such that mathematical expectation is finite, then $p_{\hat{x}}$ is also bounded.

Therefore, $I(X; Z) < \infty$ for Z being representation under Gaussian dropout. □

This proof can be straightforwardly extended to any continuous distribution—including the distributions used for information dropout (Achille and Soatto, 2018). As long as the noise distribution has finite entropy, the resulting MI is finite, since its first term is bounded by the Gaussian entropy and the second term modifies to

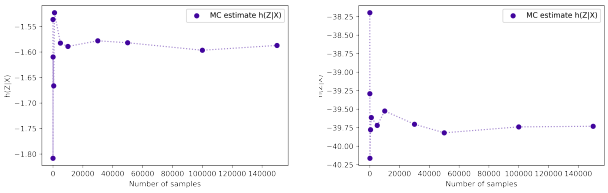
$$\mathbb{E}_x[h(|f(x)| \circ D)] = h(D) + \mathbb{E}_x[\log(|f(x)|)] \ .$$

Thus the same argument as in the proof for Gaussian dropout holds. Moreover, if Thm. 3.3 holds for a particular layer, then it follows from the data processing inequality that MI is finite for all subsequent layers.

Corollary 3.4. *Let p_x be a probability density function on X, $f(\cdot)$ be parameterized by a deterministic neural network, and let $Z = f(X) \circ D$, with $D \sim \mathcal{N}(\mathbf{1}, I\sigma^2)$. Then, under assumption of $\|X\|_\infty < \infty$, it holds that $I(X, Z_l) \leq \infty$, where Z_l corresponds to the representations on the layers l that follow after dropout.*

4 Estimation of MI under Continuous Dropout

First, let's consider information dropout network. The computational benefit of information dropout is that the values necessary to produce an IP are contained readily in the training loss. That is, the variational approximation of $I(X; Z)$ is the regularization term of the information dropout objective with additional summands for ReLU (see Appendix A.2). Since ReLU networks require a known weight on the delta Dirac in the prior distribution in order to compute the full $I(X; Z)$, we use Softplus activations with a known log-normal prior. $I(Y; Z)$ is approximated by the cross entropy loss l_{ce} subtracted from the approximation of $H(Y)$. For the classification problems $H(Y)$ can be for example approximated by $log(C)$ where C is number of classes.

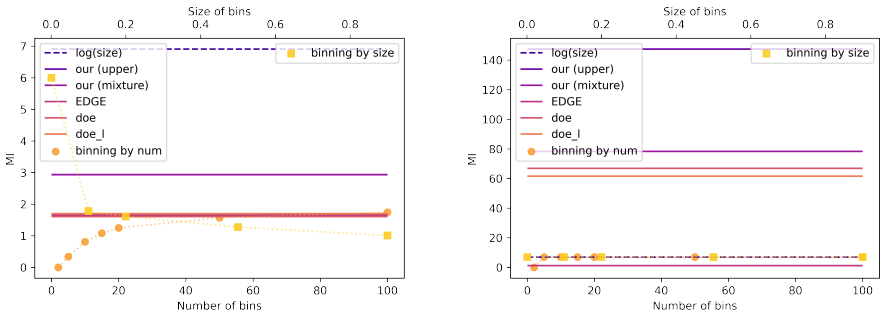


(a) $\sigma = 0.1, n = 2$ (b) $\sigma = 0.1, n = 50$

Figure 2: Independent of the dimensionality, MC estimation of $h(Z|X)$ stabilizes with the amount of samples.

In the case of Gaussian dropout, to estimate $I(X; Z)$ we approximate $h(Z)$ and $h(Z|X)$ separately.

For estimating $h(Z)$ we employ a Monte Carlo (MC) estimate, similar to the one proposed by Goldfeld et al. (2019). That is, we approximate the distribution of Z as a Gaussian mixture, where



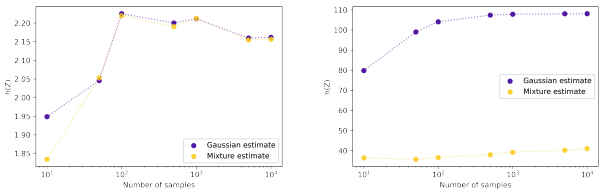
(a) $\sigma = 0.1, n = 1$ (b) $\sigma = 0.1, n = 50$

Figure 4: Comparison of various approaches to MI estimation for the toy example with multiplicative Gaussian noise. For low dimensional X and Z different bin sizes lead to different MI estimates of the binning estimator. For higher dimensions the binning based estimate is almost always maximal. Our estimation approaches the lower bound estimation proposed by McAllester and Stratos (2020) tightly from above.

we draw samples $f(x^{(j)}), j = 1, \dots$ and place Gaussians with a diagonal covariance matrix with variances $\sigma|f(x^{(j)})_i|, i = 1, \dots, n$ on each sample of $f(x^{(j)})$. For a sanity check we also compute an upper bound of $h(Z)$ given by the entropy of a Gaussian with the same covariance matrix as Z. Note, that the estimation of the upper bound needs a sufficiently large number of samples to guarantee that the estimate from the sample covariance is not singular.

For each fixed $X = x$ the conditional distribution $p_{Z|X=x}$ is a Gaussian distribution $\mathcal{N}(f(x), I(\sigma|f(x)|)^2)$. Moreover, when the input is fixed, the components of $Z|X = x$ are independent, since components of the noise are independent. This allows to compute $h(Z|X)$ as a sum of $h(Z_i|X)$ where Z_i is the i-th component of the representation vector. The computation of $h(Z_i|X)$ requires integration over the input space for computing the mathematical expectation $\mathbb{E}_x[h(Z_i|X = x)]$. This can be approximated via MC sampling. That is, we approximate $h(Z_i|X)$ by $1/N \sum_{k=1}^{N} h(Z_i|X = x_k)$ where $h(Z_i|X = x_k) = \log(|f(x_k)_i|\sigma\sqrt{2\pi e})$. [3]

We consider a simple toy problem for validating our approach to computing MI where input X is generated from an n-dimensional standard normal distribution, then modified with a function $f(X) = 2X + 0.5$ and Gaussian dropout distributed according to $\mathcal{N}(1, \sigma^2)$ is applied. We investigate the convergence of our estimator for $h(Z|X)$ for increasing number of samples. For each input sample we generate 10 noise masks, thus getting 10 samples of Z. Results are shown in Fig. 2. As it can be seen the estimation stabilizes with larger amount of samples for different dimensionality of the data. We also compare the estimate to the upper bound for $h(Z)$ in Fig 3.



(a) $\sigma = 0.1, n = 1$ (b) $\sigma = 0.1, n = 50$

Figure 3: Estimates of the entropy of the hidden representation Z. With growing dimensionality of X the upper bound of the entropy of a Gaussian becomes very loose, compared to the Gaussian mixture based MC estimation.

Moreover, we compare our estimator to the binning approach, the EDGE estimator (Noshad et al., 2019), and the lower bounds analyzed by McAllester and Stratos (2020). The results are shown in Fig. 4. With binning we underestimate MI when bin size is small and overestimate with large

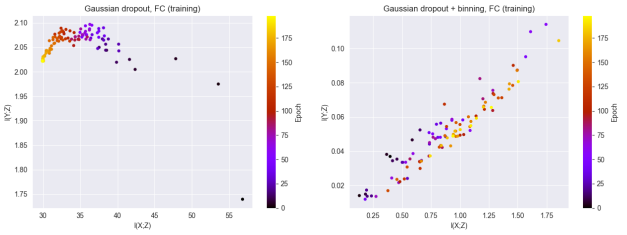
[3]Notice, that different from additive noise analysed by Goldfeld et al. (2019) the conditional entropy depends on the hidden representation $f(x)$ even in the case of single application of the noise.

bin size (Ross, 2014), which can be clearly seen in the plots where bins are organized both by size and by amount. Moreover, in with the high dimensional data, binning approach hits the maximal possible value of $log(|S|)$ (S being the dataset at hand) very fast, not being able to overestimate the MI value. According to McAllester and Stratos (2020) lower bound based MI estimators also need exponentially (in the true value of MI) many data points for correct value prediction, otherwise they will always heavily underestimate the value. Further computations with different noise level are demonstrated in Appendix A.3.

5 Information Plane Analysis of Dropout Networks

We use the estimators described in the previous section for an information plane analysis of dropout networks. In the a first set of experiments, we analysed information planes (IPs) for the training of Gaussian and information dropout networks.

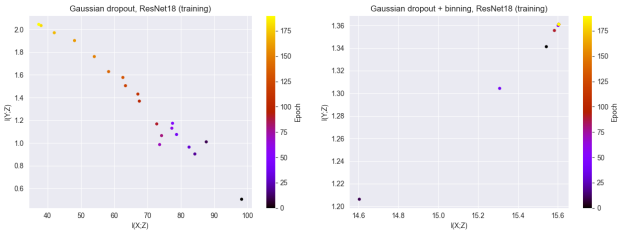
We always consider only the representation corresponding to the first dropout layer [4]. For estimating $I(Y;Z)$ we employ the EDGE estimator (Noshad et al., 2019). The analysis on the MNIST dataset was performed for a LeNet network (LeCun et al., 1998) that achieves 99% accuracy and a simple fully connected (FC) network with three hidden layers ($28 \times 28 - 512 - 128 - 32 - 10$) and softplus activation functions achieving 97% accuracy. We analyze both information dropout and Gaussian dropout in the LeNet network and only Gaussian dropout in the FC network. We compare IPs based on estimators using the binning approach to IPs based on our approach in Fig. 1 and Fig. 5. We also build the IPs for a ResNet18 trained on CIFAR10 where we added an additional bottleneck layer with 128 neurons and Gaussian dropout before the output layer, and which achieves an accuracy of 94%. The resulting IPs for our MC based estimator and binning are shown in Fig. 6.



(a) Our estimator (b) Binning estimator

Figure 5: IPs for a FC network with Gaussian dropout trained on MNIST. Compared to the binning estimation of MI our approach reveals compression.

Interestingly, for all networks and datasets we observe significant compression, which however is not shown when MI is measured based on binning (note, that we got the same finding also for different bin sizes we tried). This indicates that the MI compression measured in dropout networks is different from purely geometrical compression, which would make it fundamentally different to all the discrete MI estimators and the additive noise approach (Basirat et al., 2021).



(a) Our estimator (b) Binning estimator

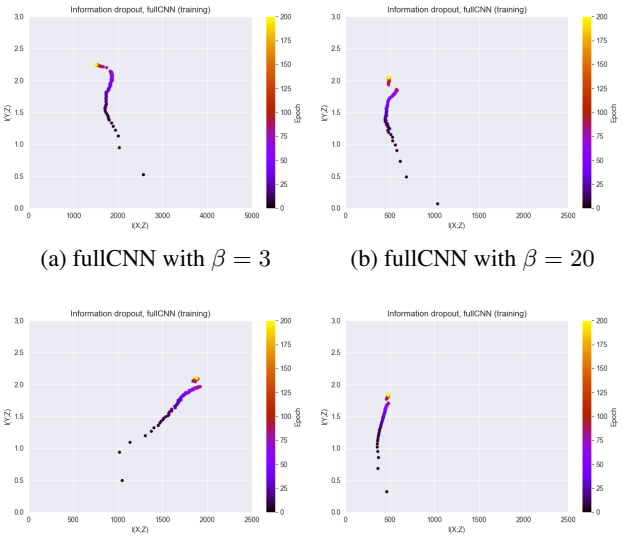
Figure 6: IPs for a ResNet18 network with Gaussian dropout trained on CIFAR10. Comparing to the binning estimation of MI our approach clearly shows compression.

In a second set of experiments, we further analyze IPs in information dropout networks. For this we trained fully convolutional neural network (fullCNN) on CIFAR10 where we used the official code provided by Achille and Soatto (2018). A fullCNN was trained for 200 epochs with SGD with momentum

[4]This makes the MI estimation more efficient, since the previous part of the network is deterministic which allows for an analytical expression of $h(Z|X = x)$. Note however, that the estimation could be extended to higher layers as well as supported by Corollary 3.4.

and, different from the original setup, only one dropout layer after the third convolutional layer. The batch size was set to 100, the learning rate was initially set to 0.05 and was reduced by multiplying it with 0.1 after the 40, 80, and 120 epoch. The network was trained with different values of the regularization weight β and different amount of filters in the convolutional layers. Also different from the original setup, we allowed noise variance to grow up to 0.95, i.e., that the information flow is nearly cut, in order to see the effect of the limited information between representation and input more pronounced.

Results are shown in Fig. 7. With less regularization of information, promoted by a smaller value of β weighting the regularization (i.e., the KL-divergence) term the level of achieved MI with the label is higher and the resulting error is 5% lower for the test set and 10% for the train set. Thus, the larger the value of β the smaller is the amount of information between input and representation throughout the training, which leads to the higher error (both on training and test set). We can see larger information compression with smaller β and almost no compression with the larger β. We conjecture that information can only be compressed if enough of it is allowed to flow through the network. We repeated experiments with the same convolutional network architecture where we reduced the number of filters in the hidden layers. Fig. 7 (a) and (b) show IPs for the original fullCNN and (c), (d) show the IPs for the network that contains 25% of the filters on each of the layers. This indicates that the ability to compress MI between input and representation can be connected with the overall capacity of a neural network.



(a) fullCNN with $\beta = 3$ (b) fullCNN with $\beta = 20$

(c) 0.25fullCNN with $\beta = 3$ (d) 0.25fullCNN with $\beta = 20$

Figure 7: IP demonstrate more (a), (c) and less (b), (d) compression of MI between input and representation.

Additional plots for the performed experiments can be found in Appendix A.4.

6 Conclusion

In this paper we analyze the dropout neural networks from the information-theoretic perspective. A theoretically sound estimation of the MI between input and hidden representation of a neural network requires either discretization or the introduction of stochasticity into the network. While discretization has the disadvantage that it limits the analysis to the geometric properties of Z, artificially introducing stochasticity limits insights due to altering the model. Dropout neural networks, however, present a class of neural networks that are inherently stochastic and heavily used in practice. In our theoretical analysis we thus investigate if different variants of dropout noise allow for finite values of MI if we assume a continuous input distribution. Our first result shows that binary dropout does not help to prevent the MI from getting infinite. We then prove, however, that Gaussian dropout –or more generally any form of dropout with noise from a continuous distribution with finite entropy–does. Based on this result we propose an MC based estimate of the MI in Gaussian dropout networks and perform an information plane analysis for different networks with Gaussian and information dropout (the latter offers a natural way to measure MI based on the MI based regularization term) on different datasets. This analysis demonstrates that the proposed estimates lead to substantially different observations than binning based approaches. Particularly, we observe compression for all studied networks and data sets. We hypothesize that in dropout networks compression of MI is not alone driven by geometrical compression. A more detailed information bottleneck analysis of dropout networks is left for the future work.

REFERENCES

Alessandro Achille and Stefano Soatto. Information dropout: Learning optimal representations through noisy computation. IEEE transactions on pattern analysis and machine intelligence, 40 (12):2897–2905, 2018. 3, 6, 8, 11

Rana Ali Amjad and Bernhard C Geiger. Learning representations for neural network-based classification using the information bottleneck principle. IEEE transactions on pattern analysis and machine intelligence, 42(9):2225–2239, 2019. 1, 3, 4, 5

Mina Basirat, Bernhard C Geiger, and Peter M Roth. A geometric perspective on information plane analysis. Entropy, 23(6):711, 2021. 1, 8

Jens Behrmann, Will Grathwohl, Ricky TQ Chen, David Duvenaud, and Jörn-Henrik Jacobsen. Invertible residual networks. In International Conference on Machine Learning, pages 573–582. PMLR, 2019. 11

Bernhard Claus Geiger. On information plane analyses of neural network classifiers – a review, 2021. accepted for publication in IEEE Trans. Neural Netw. Learn. Syst.; preprint available: `arXiv:2003.09671 [cs.LG]`. 1, 3

Ziv Goldfeld, Ewout van den Berg, Kristjan H Greenewald, Igor Melnyk, Nam Nguyen, Brian Kingsbury, and Yury Polyanskiy. Estimating information flow in deep neural networks. In ICML, 2019. 1, 3, 6, 7

Durk P Kingma, Tim Salimans, and Max Welling. Variational dropout and the local reparameterization trick. Advances in neural information processing systems, 28, 2015. 3

Artemy Kolchinsky, Brendan D Tracey, and Steven Van Kuyk. Caveats for information bottleneck in deterministic scenarios. ICLR, 2019. 2, 11

Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. Proceedings of the IEEE, 86(11):2278–2324, 1998. 8

Stephan Sloth Lorenzen, Christian Igel, and Mads Nielsen. Information bottleneck: Exact analysis of (quantized) neural networks. arXiv preprint arXiv:2106.12912, 2021. 2

David McAllester and Karl Stratos. Formal limitations on the measurement of mutual information. In International Conference on Artificial Intelligence and Statistics, pages 875–884. PMLR, 2020. 7, 8

Morteza Noshad, Yu Zeng, and Alfred O Hero. Scalable mutual information estimation using dependence graphs. In ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pages 2962–2966. IEEE, 2019. 7, 8

Brian C Ross. Mutual information between discrete and continuous data sets. PloS one, 9(2): e87357, 2014. 1, 8

Andrew M Saxe, Yamini Bansal, Joel Dapello, Madhu Advani, Artemy Kolchinsky, Brendan D Tracey, and David D Cox. On the information bottleneck theory of deep learning. Journal of Statistical Mechanics: Theory and Experiment, 2019(12):124020, 2019. 2, 3

Ravid Shwartz-Ziv and Naftali Tishby. Opening the black box of deep neural networks via information. arXiv preprint arXiv:1703.00810, 2017. 1, 3

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. The journal of machine learning research, 15(1):1929–1958, 2014. 2, 3, 5, 14

Sida Wang and Christopher Manning. Fast dropout training. In international conference on machine learning, pages 118–126. PMLR, 2013. 3

A MUTUAL INFORMATION ESTIMATION CHALLENGES

A.1 DISCUSSION ON INFINITE MUTUAL INFORMATION

MI is defined (among other definitions) as Kullback-Leibler divergence between the joint distribution and product of marginals for two random variables. According to definition

$$D_{KL}(A||B) = \mathbb{E}_A log \frac{dA}{dB}$$

where A should be absolutely continuous with respect to B (otherwise the value is defined to be infinity). So, for the validity of MI estimation we have to be sure that joint distribution is absolutely continuous with respect to the product of marginals, i.e., claiming that if $P_A \cdot P_B = 0$ it will imply that $P_{A,B} = 0$. Now let's consider the case of a deterministic neural network representation, i.e., put $X \in \mathbb{R}^d$ to be an input random variable, $Z = f(X)$, $Z \in \mathbb{R}^l$ to be an intermediate representation. While joint distribution will be defined by a pdf on $\mathbb{R}^{d+l}$ space, essentially it will have only d degrees of freedom. So the measure of the space defined by $(x, f(x)), x \in X$ will be 1 for the joint distribution pdf. But the measure of product of marginals in the same space $(x, f(x))$ will be 0, since the dimensionality of it is lower (e.g., 3-dimensional integral of a line in 2-dimensional space is 0). Like this, estimation of MI between input and representation for deterministic neural networks becomes a meaningless task, since theoretically it should be infinity.

Another side of the overall problem modeling is possibility that neural network is a bijective (one-to-one) function. Then the information passed through it does not change in terms of MI: According to the basic properties of MI $I(g(A); B) = I(A; B)$ when $g(\circ)$ is a bijective function. This results in $I(X; Z) = I(X; X)$, which is either entropy of the input or infinity, and is detrimental for measuring MI between representation and label, because here $I(Z; Y) = I(f(X); Y) = I(X; Y)$ meaning that MI will not reveal any knowledge about the properties of Z and neural network that produces it. Nevertheless, we know that training an invertable neural network is a hard task, pursued by, for example, metric learning (Behrmann et al., 2019). So, until specifically trained, neural network representations are not a bijection of the input space. On the other hand, the setup is justified by Kolchinsky et al. (2019) on the assumption that with a finite dataset in high dimension at hand the representations are mostly bijective and the information plane is not interesting for analysis of the training process. We nevertheless hold to the standard modeling paradigm in machine learning that there is a distribution defined over inputs and outputs $\mathcal{D}(\mathcal{X} \times \mathcal{Y})$, and even if the dataset at hand can be mapped to its labels one-to-one the overall space does not follow that deterministic trend, thus such analysis just omits important aspects of the underlying process in the training.

A.2 INFORMATION DROPOUT

Another type of dropout with continuous dropout noise is information dropout (Achille and Soatto, 2018). It is a technique that combines a variable dropout noise and a continuous dropout variables with dropout noise sampled from a log-normal distribution $p_{\alpha(x)}(\epsilon) = log\mathcal{N}(0, \alpha_\theta^2(x))$, where $\alpha(x)$ is a learnable parameter, and the introduction of a regularization term that is added to the standard cross entropy loss and is given by $\beta KL(p(z|x_i)|| \prod_{i=1}^{|z|} q_i(z_i))$.

This is based on another application of the information bottleneck theory to deep learning: training a neural network using the information bottleneck in the form of Lagrangian. Thus, rewriting IB-Lagrangian and adding a disentanglement term (i.e., we want each element of representation z to be independent of others) results in the used regularization term. Initial $I(x, z)$ is rewritten as $\mathbb{E}_{x\sim p(x)}[KL(p_\theta(z|x)||p_\theta(z)]$ and then mathematical expectation is replaced with empirical average.

In order to be able to optimize this regularization term in a closed form, dropout noise is sampled from log-normal distribution $p_{\alpha(x)}(\epsilon) = log\mathcal{N}(0, \alpha_\theta^2(x))$ where α is a learnable through back-propagation parameter. Additionally, it is proposed to use particular priors ($p(z)$) depending on the activation function (ReLU or Softplus) whose validity is empirically verified. Such priors and dropout noise allow for closed form of KL-divergence, which makes it very easy to directly track information plane while training.

In the following, we provide the closed form of the $I(x, z)$ as proposed by Achille and Soatto (2018)

$$I(x, z) = KL(p_\theta(x, z)||p_\theta(z) \circ p(x)) = \int p_\theta(x, z) log(\frac{p_\theta(x, z)}{p_\theta(z) \circ p(x)})$$

$$= \int p(x)p_\theta(z|x)log(\frac{p(x)p_\theta(z|x)}{p_\theta(z) \circ p(x)}) = \int p(x)KL(p_\theta(z|x)||p_\theta(z)) = \mathbb{E}_{x\sim p(x)}[KL(p_\theta(z|x)||p_\theta(z))]$$

Empirically we can compute this as

$$I(x;z) = \sum_{i=1}^{n} KL(p_\theta(z|x_i)||p_\theta(z))$$

If we fix prior distribution $p_\theta(z)$ to improper log-uniform (empirically valid for ReLU activations) and take into account that $z = \epsilon \circ f(x)$, where $\epsilon \sim log\mathcal{N}(0, \alpha^2(x))$, we can write KL-divergence as following

$$KL(p_\theta(z|x)||p_\theta(z)) =$$

considering only the case when $f(x)! = 0$ (which in turn means that $z! = 0$, since ϵ is log-normal and cannot be 0) and due to invarinace of KL-divergence under parameter transformation with a strictly monotone function

$$= KL(p_\theta(log(z)|x)||p_\theta(log(z)) =$$

by definition of KL divergence

$$= \int p_\theta(log(z)|x)log(\frac{p_\theta(log(z)|x)}{p_\theta(log(z))}dz =$$

since $log(z) = log(\epsilon) + log(f(x))$ and $p_\theta(log(z)) = c$ for improper log-uniform prior

$$= \int p_\theta((log(\epsilon) + log(f(x)))|x)log(\frac{p_\theta((log(\epsilon) + log(f(x)))|x)}{c})dz =$$

taking into account that $p(x + const) = p(x)$ and $log(f(x))|x$ is constant; also that $p_\theta(log(\epsilon)|x) = p_\theta(log(\epsilon))$ because ϵ is independent of x [epsilon is dependant on x - its variance is a function of x]

$$= \int p_{\alpha(x)}(log(\epsilon))log(p_{\alpha(x)}(log(\epsilon)))d\epsilon - \int p_{\alpha(x)}(log(\epsilon))log(c)d\epsilon =$$

since $\int p_{\alpha(x)}(log(\epsilon))d\epsilon = 1$

$$= \int p_{\alpha(x)}(log(\epsilon))log(p_{\alpha(x)}(log(\epsilon)))d\epsilon - log(c) = -H(log(\epsilon)) - log(c) =$$

since $log(\epsilon)$ is normally distributed, its entropy can be computed in closed form

$$= -(log(\alpha(x)) + \frac{1}{2}log(2\pi e)) - log(c) \tag{2}$$

In the case when we put $f(x) = 0$, we also get $z = 0$. In this case $p(z|x) = \delta_0$ and the mutual information will become:

$$= \int p_\theta(z|x)log(\frac{p_\theta(z|x)}{p_\theta(z)}dz = \int \delta_0 log(\frac{\delta_0}{q\delta_0})dz = -log(q)$$

Analogously, if instead of ReLU the network has SoftPlus activations, i.e., the prior on the representations distribution is log-normal instead of log-uniform, the formula changes to

$$KL(p_\theta(z|x)||p_\theta(z)) = \frac{1}{2}(\alpha^2(x)) - log(\alpha(x)) - \frac{1}{2}$$

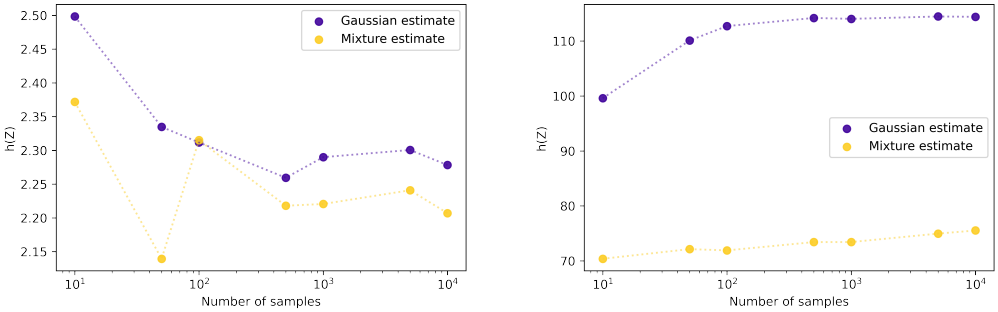
Should be noted, that finiteness of $I(x, z)$ in the case of Information Dropout can be proven in the same way as in the previous section—here the critical term is $\mathbb{E}_x[log(\alpha(x))]$ that is equivalent to the expectation of the logarithm of representation in the continuous noise dropout.

A.3 Estimation of MI under Gaussian Dropout

Fig. 8 shows upper bounds and estimation of $h(Z)$ with a higher noise than in the Fig. 3. Larger noise increases the gap between upper Gaussian estimation and mixture estimation as expected.

In Fig. 9 we see convergence of MC estimate for $h(Z|X)$ under larger noise.

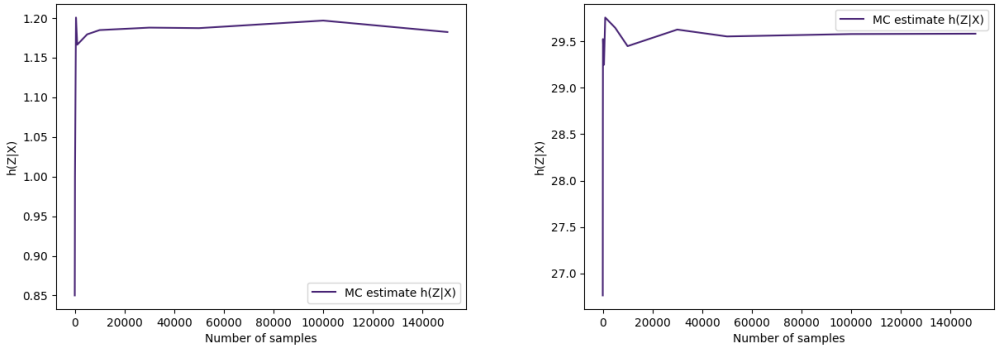
As expected larger noise variance results in smaller MI values (Fig. 10), with it the trend observed when changing dimensionality stays the same.



(a) Noise variance 0.4, dimensionality 1

(b) Noise variance 0.4, dimensionality 50

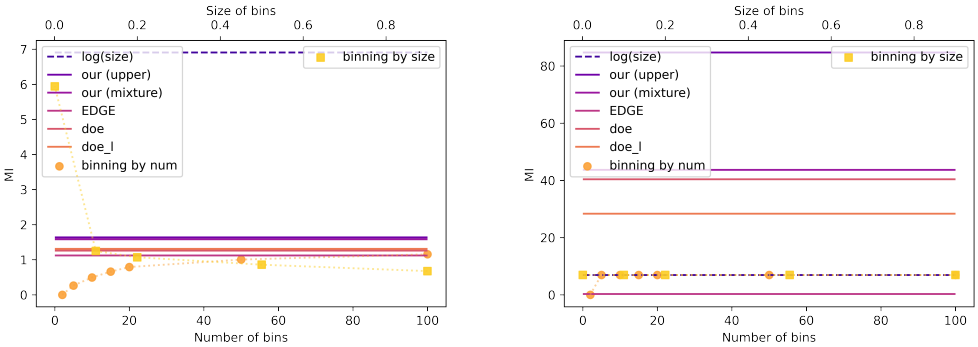
Figure 8: It can be seen that with growing dimensionality the Gaussian estimation becomes very loose, compared to the Gaussian mixture estimation.



(a) Noise variance 0.4, dimensionality 2

(b) Noise variance 0.4, dimensionality 50

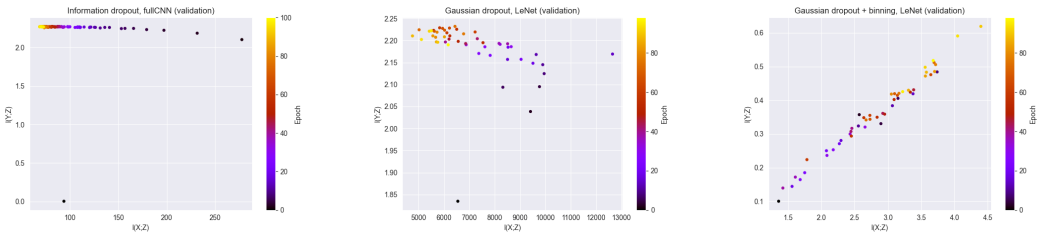
Figure 9: Independent of the dimensionality MC estimation of $h(Z|X)$ stabilizes with the amount of samples.



(a) Noise variance 0.4, dimensionality 1

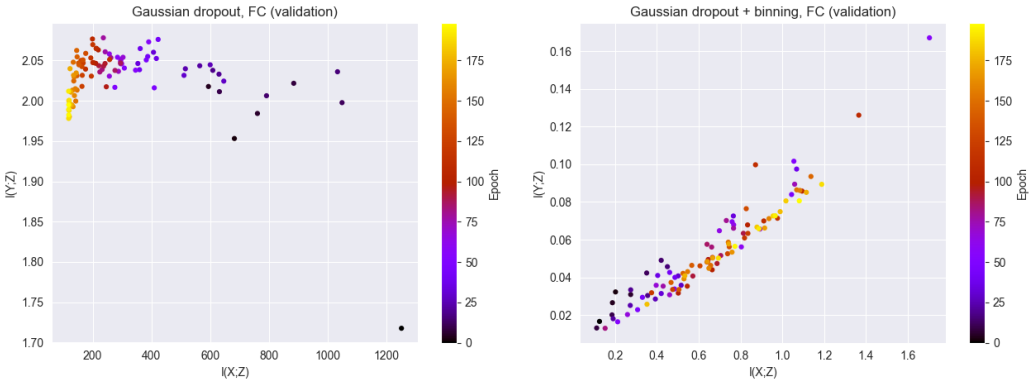
(b) Noise variance 0.4, dimensionality 50

Figure 10: Comparison of various approaches to MI estimation for the setup of the multiplicative Gaussian noise.



(a) Information dropout (b) Gaussian dropout (our) (c) Gaussian dropout (binning)

Figure 11: Compared to the binning (discrete) estimation of MI for an IP analysis, our approach shows compression as well as the compression when information dropout is applied.



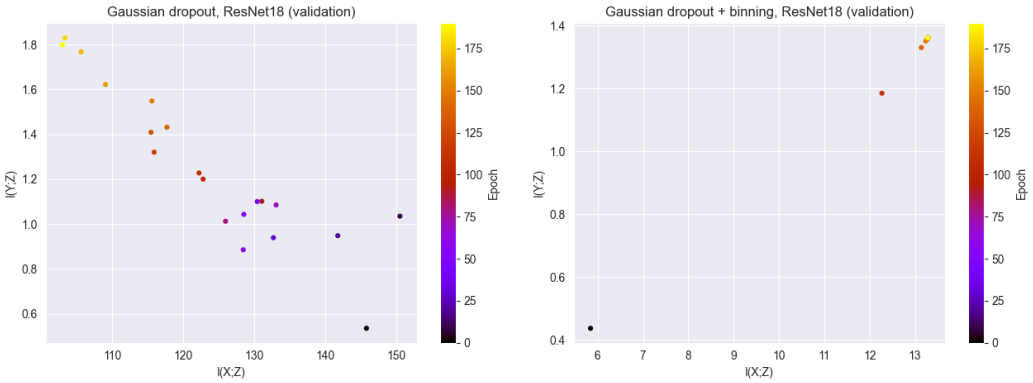
(a) Our estimator (b) Binning estimator

Figure 12: Compared to the binning (discrete) estimation of MI for an IP analysis our approach shows compression.

A.4 Information Plane Analysis

Note, that in the experiments we analyze IP on the train sample and test sample separately. In order to obtain a valid sample of variables for MI estimation during the inference, we apply MC-Dropout, as opposed to the usual way of inference with dropout being turned off. According to Srivastava et al. (2014) this is the theoretically sound way to obtain predictions, while turned off dropout is an approximation to it that allows for faster computation.

In Fig. 11, Fig.12 and Fig. 13 we provide IP-s built on the test set of the corresponding datasets.



(a) Our estimator

(b) Binning estimator

Figure 13: Comparing to the discrete estimation of MI for an IP analysis our approach clearly shows compression.