This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2017.2784440, IEEE Transactions on Pattern Analysis and Machine Intelligence

Information Dropout: Learning Optimal Representations Through Noisy Computation

Alessandro Achille and Stefano Soatto
Department of Computer Science
University of California, Los Angeles
405 Hilgard Ave, Los Angeles, 90095, CA, USA
Email: {achille,soatto}@cs.ucla.edu

Abstract—The cross-entropy loss commonly used in deep learning is closely related to the defining properties of optimal representations, but does not enforce some of the key properties. We show that this can be solved by adding a regularization term, which is in turn related to injecting multiplicative noise in the activations of a Deep Neural Network, a special case of which is the common practice of dropout. We show that our regularized loss function can be efficiently minimized using Information Dropout, a generalization of dropout rooted in information theoretic principles that automatically adapts to the data and can better exploit architectures of limited capacity. When the task is the reconstruction of the input, we show that our loss function yields a Variational Autoencoder as a special case, thus providing a link between representation learning, information theory and variational inference. Finally, we prove that we can promote the creation of optimal disentangled representations simply by enforcing a factorized prior, a fact that has been observed empirically in recent work. Our experiments validate the theoretical intuitions behind our method, and we find that information dropout achieves a comparable or better generalization performance than binary dropout, especially on smaller models, since it can automatically adapt the noise to the structure of the network, as well as to the test sample.

Index Terms—Representation learning, deep learning, information bottleneck, nuisances, invariants, minimality

1 Introduction

We call "representation" any function of the data that is useful for a task. An optimal representation is most useful (sufficient), parsimonious (minimal), and minimally affected by nuisance factors (invariant). Do deep neural networks approximate such *sufficient invariants*?

The cross-entropy loss most commonly used in deep learning does indeed enforce the creation of sufficient representations, but the other defining properties of optimal representations do not seem to be explicitly enforced by the commonly used training procedures. However, (we show that) this can be done by adding a regularizer, which is related to the injection of multiplicative noise in the activations, with the surprising result that *noisy computation facilitates the approximation of optimal representations*. In this paper we establish connections between the theory of optimal representations for classification tasks, variational inference, dropout and "disentangling" in deep neural networks. Our contributions can be summarized in the following steps:

(1) We define optimal representations using established principles of statistical decision and information theory: sufficiency, minimality, invariance (cf. [1], [2]) (Section 3).
(2) We relate the defining properties of optimal representations for classification to the loss function most commonly used in deep learning, but with an added regularizer (Section 4, eq. (3)).
(3) We show that, counter-intuitively, injecting multiplicative noise to the computation improves the properties of a representation and results in better approximation of an optimal one (Section 6).
(4) We relate such a multiplicative noise to the regularizer, and show that in the special case of Bernoulli noise, regularization reduces to dropout [3], thus establishing a connection to information theoretic principles. We also provide a more efficient alternative, called Information Dropout, that makes better use of limited capacity, adapts to the data, and is related to Variational Dropout [4] (Section 6).
(5) We show that, when the task is reconstruction, the procedure above yields a generalization of the Variational Autoencoder, which is instead derived from a Bayesian inference perspective [5]. This establishes a connection between information theoretic and Bayesian representations, where the former explains the use of a multiplier used in practice but unexplained by Bayesian theory (Section 7).
(6) We show that "disentanglement of the hidden causes," an often-cited but seldom formalized desideratum for deep networks, can be achieved by assuming a factorized prior for the components of the optimal representation. Specifically, we prove that computing the regularizer term under the simplifying assumption of an independent prior has the effect of minimizing the total correlation of the components, a phenomenon previously observed empirically by [6] (Section 5).
(7) We validate the theory with several experiments including: improved insensitivity/invariance to nuisance

Dedicated to Naftali Tishby in the occasion of the conference Information, Control and Learning *held in his honor in Jerusalem, September 26-28, 2016. Registered as Tech Report UCLA-CSD160009 and arXiv:1611.01353 on November 6, 2016*

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2017.2784440, IEEE Transactions on Pattern Analysis and Machine Intelligence

factors using Information Dropout using (a) Cluttered MNIST [7] and (b) MNIST+CIFAR, a newly introduced dataset to test sensitivity to occlusion phenomena critical in Vision applications; (c) we show improved efficiency of Information Dropout compared to regular dropout for limited capacity networks, (d) we show that Information Dropout favors disentangled representations; (e) we show that Information Dropout adapts to the data and allows different amounts of information to flow between different layers in a deep network (Section 8).

In the next section we introduce the basic formalism to make the above statements more precise, which we do in subsequent sections.

2 Preliminaries

In the general supervised setting, we want to learn the conditional distribution $p(\mathbf{y}|\mathbf{x})$ of some random variable $\mathbf{y}$, which we refer to as the *task*, given (samples of the) input data $\mathbf{x}$. In typical applications, $\mathbf{x}$ is often high dimensional (for example an image or a video), while $\mathbf{y}$ is low dimensional, such as a label or a coarsely-quantized location. In such cases, a large part of the variability in $\mathbf{x}$ is actually due to *nuisance factors* that affect the data, but are otherwise irrelevant for the task [1]. Since by definition these nuisance factors are not predictive of the task, they should be disregarded during the inference process. However, it often happens that modern machine learning algorithms, in part due to their high flexibility, will fit spurious correlations, present in the training data, between the nuisances and the task, thus leading to poor generalization performance.

In view of this, [8] argue that the success of deep learning is in part due to the capability of neural networks to build incrementally better representations that expose the relevant variability, while at the same time discarding nuisances. This interpretation is intriguing, as it establishes a connection between machine learning, probabilistic inference, and information theory. However, common training practice does not seem to stem from this insight, and indeed deep networks may maintain even in the top layers dependencies on easily ignorable nuisances (see for example Figure 2).

To bring the practice in line with the theory, and to better understand these connections, we introduce a modified cost function, that can be seen as an approximation of the Information Bottleneck Lagrangian of [2], which encourages the creation of representations of the data which are increasingly disentangled and insensitive to the action of nuisances, and we show that this loss can be minimized using a new layer, which we call *Information Dropout*, that allows the network to selectively introduce multiplicative noise in the layer activations, and thus to control the flow of information. As we show in various experiments, this method improves the generalization performance by building better representations and preventing overfitting, and it considerably improves over binary dropout on smaller models, since, unlike dropout, Information Dropout also adapts the noise to the structure of the network and to the individual sample at test time.

Apart from the practical interest of Information Dropout, one of our main results is that Information Dropout can be seen as a generalization to several existing dropout methods, providing a unified framework to analyze them, together with some additional insights on empirical results. As we discuss in Section 3, the introduction of noise to prevent overfitting has already been studied from several points of view. For example the original formulation of dropout of [3], which introduces binary multiplicative noise, was motivated as a way of efficiently training an ensemble of exponentially many networks, that would be averaged at testing time. [4] introduce *Variational Dropout*, a dropout method which closely resemble ours, and is instead derived from a Bayesian analysis of neural networks. Information Dropout gives an alternative information-theoretic interpretation of those methods.

As we show in Section 7, other than being very closely related to Variational Dropout, Information Dropout directly yields a variational autoencoder as a special case when the task is the reconstruction of the input. This result is in part expected, since our loss function seeks an optimal representation of the input for the task of reconstruction, and the representation given by the latent variables of a variational autoencoder fits the criteria. However, it still rises the question of exactly what and how deep are the links between information theory, representation learning, variational inference and nuisance invariance. This work can be seen as a small step in answering this question.

3 Related work

The main contribution of our work is to establish how two seemingly different areas of research, namely dropout methods to prevent overfitting, and the study of optimal representations, can be linked through the Information Bottleneck principle.

Dropout was introduced by Srivastava et al. [3]. The original motivation was that by randomly dropping the activations during training, we can effectively train an ensemble of exponentially many networks, that are then averaged during testing, therefore reducing overfitting. Wang et al. [9] suggested that dropout could be seen as performing a Monte-Carlo approximation of an implicit loss function, and that instead of multiplying the activations by binary noise, like in the original dropout, multiplicative Gaussian noise with mean 1 can be used as a way of better approximating the implicit loss function. This led to a comparable performance but faster training than binary dropout.

Kingma et al. [4] take a similar view of dropout as introducing multiplicative (Gaussian) noise, but instead study the problem from a Bayesian point of view. In this setting, given a training dataset $\mathcal{D} = \{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1,\dots,N}$ and a prior distribution $p(\mathbf{w})$, we want to compute the posterior distribution $p(\mathbf{w}|\mathcal{D})$ of the weights $\mathbf{w}$ of the network. As is customary in variational inference, the true posterior can be approximated by minimizing the negative variational lower bound $\mathcal{L}(\theta)$ of the marginal log-likelihood of the data,

$$\mathcal{L}(\theta) = \frac{1}{N}\sum_{i=1}^{N} \mathbb{E}_{\mathbf{w}\sim p_\theta(\mathbf{w}|\mathcal{D})}[-\log p(\mathbf{y}_i|\mathbf{x}_i,\mathbf{w})] + \\ + \frac{1}{N}\,\mathrm{KL}(p_\theta(\mathbf{w}|\mathcal{D}) \parallel p(\mathbf{w})). \quad (1)$$

This minimization is difficult to perform, since it requires to repeatedly sample new weights for each sample of

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2017.2784440, IEEE Transactions on Pattern Analysis and Machine Intelligence

the dataset. As an alternative, [4] suggest that the uncertainty about the weights that is expressed by the posterior distribution $p_\theta(\mathbf{w}|\mathcal{D})$ can equivalently be encoded as a multiplicative noise in the activations of the layers (the so called *local reparametrization trick*). As we will see in the following sections, this loss function closely resemble the one of Information Dropout, which however is derived from a purely information theoretic argument based on the Information Bottleneck principle. One difference is that we allow the parameters of the noise to change on a per-sample basis (which, as we show in the experiments, can be useful to deal with nuisances), and that we allow a scaling constant β in front of the KL-divergence term, which can be changed freely. Interestingly, even if the Bayesian derivation does not allow a rescaling of the KL-divergence, Kingma et al. notice that choosing a different scale for the KL-divergence term can indeed lead to improvements in practice. A related method, but derived from an information theoretic perspective was also suggested previously by [10].

The interpretation of deep neural network as a way of creating successively better representations of the data has already been suggested and explored by many. Most recently, Tishby et al. [8] put forth an interpretation of deep neural networks as creating sufficient representations of the data that are increasingly minimal. In parallel simultaneous work, [11] approximate the information bottleneck similarly to us, but focus on empirical analysis of robustness to adversarial perturbations rather than tackling disentanglement, invariance and minimality analytically.

Sufficient dimensionality reduction [12] and Optimal Component Analysis [13] follow a similar idea to us, in that they focus on finding the smallest (usually linear) sufficient statistic of the data that is sufficient for a given task. However, while they define small in term of dimension of the representation, we focus on finding a (non-linear) representation with minimal information content, but whose dimension can, in fact, be even larger than the original data. By allowing large non-linear representations, we can exploit the full representational power of deep networks, while the minimality of the information content still promotes nuisance invariance and prevents overfitting. Our framework also has connections with Independent Component Analysis (ICA), which we discuss further in Section 7.

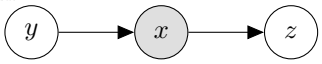
Some have focused on creating representations that are *maximally* invariant to nuisances, especially when they have the structure of a (possibly infinite-dimensional) group acting on the data, like [14], or, when the nuisance is a locally compact group acting on each layer, by successive approximations implemented by hierarchical convolutional architectures, like [15] and [16]. In these cases, which cover common nuisances such as translations and rotations of an image (affine group), or small diffeomorphic deformations due to a slight change of point of view (group of diffeomorphisms), the representation is equivalent to the data modulo the action of the group. However, when the nuisances are not a group, as is the case for occlusions, it is not possible to achieve such equivalence, that is, there is a loss. To address this problem, [1] defined optimal representations not in terms of maximality, but in terms of *sufficiency*, and characterized representations that are both sufficient and invariant. They argue that the management of nuisance factors common in visual data, such as changes of viewpoint, local deformations, and changes of illumination, is directly tied to the specific structure of deep convolutional networks, where local marginalization of simple nuisances at each layer results in marginalization of complex nuisances in the network as a whole.

Our work fits in this last line of thinking, where the goal is not equivalence to the data up to the action of (group) nuisances, but instead sufficiency for the task. **Our main contribution** in this sense is to show that injecting noise into the layers, and therefore using a non-deterministic function of the data, can actually simplify the theoretical analysis and lead to disentangling and improved insensitivity to nuisances. This is an alternate explanation to that put forth by the references above.

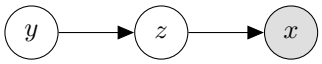
4 Optimal representations and the Information Bottleneck loss

Given some input data $\mathbf{x}$, we want to compute some (possibly nondeterministic) function of $\mathbf{x}$, called a *representation*, that has some desirable properties in view of the task $\mathbf{y}$, for instance by being more convenient to work with, exposing relevant statistics, or being easier to store. Ideally, we want this representation to be as good as the original data for the task, and not squander resources modeling parts of the data that are irrelevant to the task. Formally, this means that we want to find a random variable $\mathbf{z}$ satisfying the following conditions:

(i) $\mathbf{z}$ is a **representation** of $\mathbf{x}$; that is, its distribution depends only on $\mathbf{x}$, as expressed by the following Markov chain:



(ii) $\mathbf{z}$ is **sufficient** for the task $\mathbf{y}$, that is $I(\mathbf{x};\mathbf{y}) = I(\mathbf{z};\mathbf{y})$, expressed by the Markov chain:



(iii) among all random variables satisfying these requirements, the mutual information $I(\mathbf{x};\mathbf{z})$ is **minimal**. This means that $\mathbf{z}$ discards all variability in the data that is not relevant to the task.

Using the identity $I(\mathbf{x};\mathbf{y}) - I(\mathbf{z};\mathbf{y}) = I(\mathbf{x};\mathbf{y}|\mathbf{z})$, where I the mutual information, it is easy to see that the above conditions are equivalent to finding a distribution $p(\mathbf{z}|\mathbf{x})$ which solves the optimization problem

$$\begin{aligned} \text{minimize} \quad & I(\mathbf{x};\mathbf{z}) \\ \text{s.t.} \quad & I(\mathbf{x};\mathbf{y}|\mathbf{z}) = 0. \end{aligned}$$

The minimization above is difficult in general. For this reason, Tishby et al. have introduced a generalization known as the *Information Bottleneck Principle* and the associated Lagrangian to be minimized [2]:

$$\mathcal{L} = I(\mathbf{x},\mathbf{y}|\mathbf{z}) + \beta I(\mathbf{x};\mathbf{z}).$$

When y is a discrete random variable, such as a label, as we will often assume through this work, we can further use the identity $I(\mathbf{x};\mathbf{y}|\mathbf{z}) = H(\mathbf{y}|\mathbf{z}) - H(\mathbf{y}|\mathbf{x})$ and the fact that $H(\mathbf{y}|\mathbf{x})$ is constant to obtain the equivalent Lagrangian

$$\mathcal{L} = H(\mathbf{y}|\mathbf{z}) + \beta I(\mathbf{x};\mathbf{z}), \tag{2}$$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2017.2784440, IEEE Transactions on Pattern Analysis and Machine Intelligence

where β is a positive constant that manages the trade-off between sufficiency (the performance on the task, as measured by the first term) and minimality (the complexity of the representation, measured by the second term). It is easy to see that, in the limit $\beta \to 0^+$, this is equivalent to the original problem, where $\mathbf{z}$ is a minimal sufficient statistic. When all random variables are discrete and $\mathbf{z} = T(\mathbf{x})$ is a deterministic function of $\mathbf{x}$, the algorithm proposed by [2] can be used to minimize the IB Lagrangian efficiently. However, no algorithm is known to minimize the IB Lagrangian for non-Gaussian, high-dimensional continuous random variables.

One of our key results is that, when we restrict to the family of distributions obtained by injecting noise to one layer of a neural network, we can efficiently approximate and minimize the IB Lagrangian.[1] As we will show, this process can be effectively implemented through a generalization of the dropout layer that we call *Information Dropout*.

To set the stage, we rewrite the IB Lagrangian as a per-sample loss function. Let $p(\mathbf{x}, \mathbf{y})$ denote the true distribution of the data, from which the training set $\{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1,\dots,N}$ is sampled, and let $p_\theta(\mathbf{z}|\mathbf{x})$ and $p_\theta(\mathbf{y}|\mathbf{z})$ denote the unknown distributions that we wish to estimate, parametrized by θ. Then, we can write the two terms in the IB Lagrangian as

$$H(\mathbf{y}|\mathbf{z}) \simeq \mathbb{E}_{\mathbf{x},\mathbf{y}\sim p(\mathbf{x},\mathbf{y})}\left[\mathbb{E}_{\mathbf{z}\sim p_\theta(\mathbf{z}|\mathbf{x})}[-\log p_\theta(\mathbf{y}|\mathbf{z})]\right]$$
$$I(\mathbf{x};\mathbf{z}) = \mathbb{E}_{\mathbf{x}\sim p(\mathbf{x})}[\mathrm{KL}(p_\theta(\mathbf{z}|\mathbf{x}) \parallel p_\theta(\mathbf{z}))],$$

where KL denotes the Kullback-Leibler divergence. We can therefore approximate the IB Lagrangian empirically as

$$\mathcal{L} = \frac{1}{N}\sum_{i=1}^{N} \mathbb{E}_{z\sim p(\mathbf{z}|\mathbf{x}_i)}[-\log p(\mathbf{y}_i|\mathbf{z})] + \beta\, \mathrm{KL}(p_\theta(\mathbf{z}|\mathbf{x}_i) \parallel p_\theta(\mathbf{z})). \tag{3}$$

Notice that the first term simply is the average cross-entropy, which is the most commonly used loss function in deep learning. The second term can then be seen as a regularization term. In fact, many classical regularizers, like the L_2 penalty, can be expressed in the form of eq. (3) (see also [17]). In this work, we interpret the KL term as a reuglarizer that penalizes the transfer of information from $\mathbf{x}$ to $\mathbf{z}$. In the next section, we discuss ways to control such information transfer through the injection of noise.

Remark (Deterministic vs. stochastic representations)**.** Aside from being easier to work with, stochastic representations can attain a lower value of the IB Lagrangian than any deterministic representation. For example, consider the task of reconstructing single random bit y given a noisy observation x. The only deterministic representations are equivalent to the either the noisy observation itself or to the trivial constant map. It is not difficult to check that for opportune values of β and of the noise, neither realize the optimal tradeoff reached by a suitable stochastic representation.

Remark (Approximate sufficiency)**.** The quantity $I(x;y|z) = H(y|z) - H(y|x) \geq 0$ can be seen as a measure of the distance between $p(x,y,z)$ and the closest distribution $q(x,y,z)$ such that $\mathbf{x} \to \mathbf{z} \to \mathbf{y}$ is a Markov chain. Therefore, by minimizing eq. (2) we find representations that are increasingly "more sufficient", meaning that they are closer to an actual Markov chain.

1. Since we restrict the family of distributions, there is no guarantee that the resulting representation will be optimal. We can, however, iterate the process to obtain incrementally improved approximations.

5 Disentanglement

In addition to sufficiency and minimality, "disentanglement of hidden factors" is often cited as a desirable property of a representation [18], but seldom formalized. We may think that the observed data is generated by a complex interplay of independent causes, or factors. Ideally, the components of the learned representation should capture these independent factors by disentangling the correlations in the observed data. We can then quantify disentanglement by measuring the *total correlation* [19], also known as *multiinformation* [20],[2] defined as

$$\mathrm{TC}(\mathbf{z}) := \mathrm{KL}(q(\mathbf{z}) \parallel \textstyle\prod_j q_j(z_j)).$$

Notice that the components of $\mathbf{z}$ are mutually independent if and only if $\mathrm{TC}(z)$ is zero. Adding this as a penalty in the IB Lagrangian, with a factor γ yields

$$\mathcal{L} = \frac{1}{N}\sum_{i=1}^{N} \mathbb{E}_{z\sim p(\mathbf{z}|\mathbf{x}_i)}[-\log p(\mathbf{y}_i|\mathbf{z})] + \\ + \beta\, \mathrm{KL}(p_\theta(\mathbf{z}|\mathbf{x}_i) \parallel p_\theta(\mathbf{z})) + \gamma\, \mathrm{TC}(\mathbf{z}). \tag{4}$$

In general, minimizing this augmented loss is intractable, since to compute both the KL term and the total correlation, we need to know the marginal distribution $p_\theta(z)$, which is not easily computable. However, the following proposition, that we prove in Section B, shows that if we choose $\gamma = \beta$, then the problem simplifies, and can be easily solved by adding an auxiliary variable.

Proposition 1. *The minimization problem*

$$\mathrm{minimize}_p \frac{1}{N}\sum_{i=1}^{N} \mathbb{E}_{z\sim p(\mathbf{z}|\mathbf{x}_i)}[-\log p(\mathbf{y}_i|\mathbf{z})] + \\ + \beta\left\{\mathrm{KL}(p(\mathbf{z}|\mathbf{x}_i) \parallel p(\mathbf{z})) + \mathrm{TC}(\mathbf{z})\right\},$$

is equivalent to the following minimization in two variables

$$\mathrm{minimize}_{p,q} \frac{1}{N}\sum_{i=1}^{N} \mathbb{E}_{z\sim p(\mathbf{z}|\mathbf{x}_i)}[-\log p(\mathbf{y}_i|\mathbf{z})] + \\ + \beta\, \mathrm{KL}(p(\mathbf{z}|\mathbf{x}_i) \parallel \textstyle\prod_{i=1}^{|z|} q_i(\mathbf{z}_i)).$$

In other words, minimizing the standard IB Lagrangian assuming that the activations are independent, i.e. having $q(\mathbf{z}) = \prod_i q_i(\mathbf{z}_i)$, is equivalent to enforcing disentanglement of the hidden factors. It is interesting to note that this independence assumption is already adopted often by practitioners on grounds of simplicity, since the actual marginal $p(\mathbf{z}) = \int_{\mathbf{x}} p(\mathbf{x},\mathbf{z})d\mathbf{x}$ is often incomputable. That using a factorized model results in "disentanglement" was also observed empirically by [6] which, however, introduced

2. As pointed out by a reviewer, multi-information would be a more appropriate name for this quantity. We chose to use Total Correlation both for historical reasons, after its introduction in [19], and to emphasize the relation with disentanglement also in recent work on unsupervised learning [21]. Other measures of independence are of course possible. Total Correlation has the advantage of being enforced naturally when optimizing other information-theoretic quantities.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2017.2784440, IEEE Transactions on Pattern Analysis and Machine Intelligence

an ad-hoc metric based on classifiers of low VC-dimension, rather than the more natural Total Correlation adopted here.

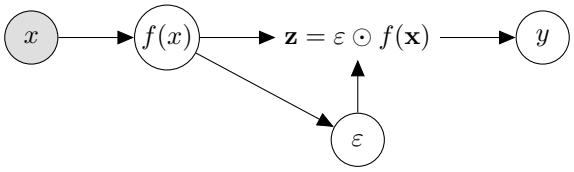
In view of the previous proposition, from now on we will assume that the activations are independent and ignore the total correlation term.

6 Information Dropout

Guided by the analysis in the previous sections, and to emphasize the role of stochasticity, we consider representations $\mathbf{z}$ obtained by computing a deterministic map $f(\mathbf{x})$ of the data (for instance a sequence of convolutional and/or fully-connected layers of a neural network), and then multiplying the result component-wise by a random sample ϵ drawn from a parametric noise distribution p_α with unit mean and variance that depends on the input $\mathbf{x}$:

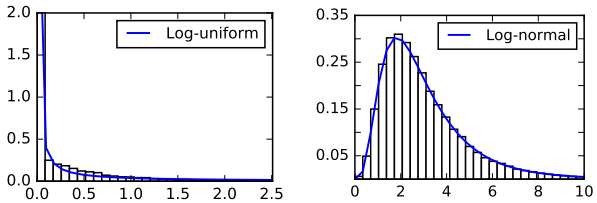
$$\varepsilon \sim p_{\alpha(\mathbf{x})}(\varepsilon),$$
$$\mathbf{z} = \varepsilon \odot f(\mathbf{x}),$$

where "$\odot$" denotes the element-wise product. Notice that, if $p_{\alpha(\mathbf{x})}(\varepsilon)$ is a Bernoulli distribution rescaled to have mean 1, this reduces exactly to the classic binary dropout layer. As we discussed in Section 3, there are also variants of dropout that use different distributions.



A natural choice for the distribution $p_{\alpha(\mathbf{x})}(\varepsilon)$, which also simplifies the theoretical analysis, is the log-normal distribution $p_{\alpha(\mathbf{x})}(\varepsilon) = \log\mathcal{N}(0, \alpha^2_\theta(\mathbf{x}))$. Once we fix this noise distribution, given the above expression for $\mathbf{z}$, we can easily compute the distribution $p_\theta(\mathbf{z}|\mathbf{x})$ that appears in eq. (3). However, to be able to compute the KL-divergence term, we still need to fix a prior distribution $q_\theta(\mathbf{z})$. The choice of this prior largely depends on the expected distribution of the activations $f(\mathbf{x})$. Recall that, by Section 5, we can assume that all activations are independent, thus simplifying the computation. Now, we concentrate on two of the most common activation functions, the *rectified linear unit* (ReLU), which is easy to compute and works well in practice, and the *Softplus* function, which can be seen as a strictly positive and differentiable approximation of ReLU.

A network implemented using only ReLU and a final Softmax layer has the remarkable property of being scale-invariant, meaning that multiplying all weights, biases, and activations by a constant does not change the final result. Therefore, from a theoretical point of view, it would be desirable to use a scale-invariant prior. The only such prior is the improper log-uniform, $q(\log(z)) = c$, or equivalently $q(z) = c/z$, which was also suggested by [4], but as a prior for the weights of the network, rather than the activations. Since the ReLU activations are frequently zero, we also assume $q(z = 0) = q_0$ for some constant $0 \leq q_0 \leq 1$. Therefore, the final prior has the form $q(z) = q_0\delta_0(z) + c/z$, where δ_0 is the Dirac delta in zero. In Figure 1a, we compare this prior distribution with the actual empirical distribution $p(z)$ of a network with ReLU activations.



(a) Histogram of ReLU activations (b) Histogram of Softplus activations

Fig. 1: Comparison of the empirical distribution $p(z)$ of the post-noise activations with our proposed prior when using: (a) ReLU activations, for which we propose a log-uniform prior, and (b) Softplus activations, for which we propose a log-normal prior. In both cases, the empirical distribution approximately follows the proposed prior. Both histograms where obtained from the last dropout layer of the All-CNN-32 network described in Table 2, trained on CIFAR-10.

In a network implemented using Softplus activations, a log-normal is a good fit of the distribution of the activations. This is to be expected, especially when using batch-normalization, since the pre-activations will approximately follow a normal distribution with zero mean, and the Softplus approximately resembles a scaled exponential near zero. Therefore, in this case we suggest using a log-normal distribution as our prior $q(z)$. In Figure 1b, we compare this prior with the empirical distribution $p(z)$ of a network with Softplus activations.

Using these priors, we can finally compute the KL divergence term in eq. (3) for both ReLU activations and Softplus activations. We prove the following two propositions in Section A.

Proposition 2 (Information dropout cost for ReLU)**.** *Let $z = \varepsilon \cdot f(x)$, where $\varepsilon \sim p_\alpha(\varepsilon)$, and assume $p(z) = q\delta_0(z) + c/z$. Then, assuming $f(x) \neq 0$, we have*

$$\mathrm{KL}(p_\theta(z|x) \parallel p(z)) = -H(p_{\alpha(x)}(\log\varepsilon)) + \log c$$

In particular, if $p_\alpha(\varepsilon)$ is chosen to be the log-normal distribution $p_\alpha(\varepsilon) = \log\mathcal{N}(0, \alpha^2_\theta(x))$, we have

$$\mathrm{KL}(p_\theta(z|x) \parallel p(z)) = -\log\alpha_\theta(x) + const. \qquad (5)$$

If instead $f(x) = 0$, we have

$$\mathrm{KL}(p_\theta(z|x) \parallel p(z)) = -\log q.$$

Proposition 3 (Information dropout cost for Softplus)**.** *Let $z = \varepsilon \cdot f(x)$, where $\varepsilon \sim p_\alpha(\varepsilon) = \log\mathcal{N}(0, \alpha^2_\theta(x))$, and assume $p_\theta(z) = \log\mathcal{N}(\mu, \sigma^2)$. Then, we have*

$$\mathrm{KL}(p_\theta(z|x) \parallel p(z)) = \frac{1}{2\sigma^2}\left(\alpha^2(x) + \mu^2\right) - \log\frac{\alpha(x)}{\sigma} - \frac{1}{2}. \qquad (6)$$

Substituting the expression for the KL divergence in eq. (5) inside eq. (3), and ignoring for simplicity the special case $f(x) = 0$, we obtain the following loss function for ReLU activations

$$\mathcal{L} = \frac{1}{N}\sum_{i=1}^{N} \mathbb{E}_{\mathbf{z}\sim p_\theta(\mathbf{z}|\mathbf{x}_i)}[\log p(\mathbf{y}_i|\mathbf{z})] + \beta\log\alpha_\theta(\mathbf{x}_i), \qquad (7)$$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2017.2784440, IEEE Transactions on Pattern Analysis and Machine Intelligence

and a similar expression for Softplus. Notice that the first expectation can be approximated by sampling (in the experiments we use one single sample, as customary for dropout), and is just the average cross-entropy term that is typical in deep learning. The second term, which is new, penalizes the network for choosing a low variance for the noise, i.e. for letting more information pass through to the next layer. This loss can be optimized easily using stochastic gradient descent and the reparametrization trick of [5] to back-propagate the gradient through the sampling operation.

7 Connections with other frameworks

In this section, we outline strong connections between Information Dropout, Variational Autoencoders (VAEs) [5], and Independent Component Analysis (ICA) [22], [23].

Variational autoencoders aim to reconstruct, given a training dataset $\mathcal{D} = \{\mathbf{x}_i\}$, a latent random variable $\mathbf{z}$ such that the observed data $\mathbf{x}$ can be thought as being generated by the, usually simpler, variable $\mathbf{z}$ through some unknown generative process $p_\theta(\mathbf{x}|\mathbf{z})$. In practice, this is done by minimizing the negative variational lower-bound to the marginal log-likelihood of the data

$$\mathcal{L}(\theta) = \frac{1}{N}\sum_{i=1}^{N} \mathbb{E}_{\mathbf{z}\sim p_\theta(\mathbf{z}|\mathbf{x}_i)}[-\log p_\theta(\mathbf{x}_i|\mathbf{z})] + \mathrm{KL}(p_\theta(\mathbf{z}|\mathbf{x}_i) \parallel \textstyle\prod_i q_\theta(z_i)),$$

where the optimization is joint over the factorized prior $q_\theta(\mathbf{z})$, which is often assumed to be factorized, and the posterior $p_\theta(\mathbf{z}|\mathbf{x})$. The optimization can now be performed easily through sampling using the SGVB method of [5].

We now show that this procedure can be seen as a special case of Information Dropout: Consider again the loss in eq. (4) in the special case $\mathbf{y} = \mathbf{x}$, that is, when the task is reconstruction of the input:

$$\mathcal{L} = \frac{1}{N}\sum_{i=1}^{N} \mathbb{E}_{z\sim p(\mathbf{z}|\mathbf{x}_i)}[-\log p(\mathbf{x}_i|\mathbf{z})] + \beta\,\mathrm{KL}(p_\theta(\mathbf{z}|\mathbf{x}_i) \parallel p_\theta(\mathbf{z})) + \gamma\,\mathrm{TC}(\mathbf{z}). \quad (8)$$

By Proposition 1, in the special case $\beta = \gamma$, this reduces to

$$\mathcal{L}(\theta) = \frac{1}{N}\sum_{i=1}^{N} \mathbb{E}_{\mathbf{z}\sim p_\theta(\mathbf{z}|\mathbf{x}_i)}[-\log p_\theta(\mathbf{x}_i|\mathbf{z})] + \beta\,\mathrm{KL}(p_\theta(\mathbf{z}|\mathbf{x}_i) \parallel \textstyle\prod_i q_\theta(z_i)), \quad (9)$$

where again the optimization is joint over prior $q_\theta(\mathbf{z})$ and posterior $p_\theta(\mathbf{z}|\mathbf{x})$, leading to the same optimization problem of a VAE when $\beta = 1$, that is when all quantities have the same weight in the loss function. This derivation also provides some additional insights: when using a factorized prior, a VAE will try to find a representation of the data which is sufficient for reconstruction (cross-entropy term), maximally compressed (KL term) *and* disentangled (total correlation term). We can also see that, while using instead a non factorized prior increases the complexity of the optimization problem, it spares the VAE from having to find a disentangled representation, allowing it to obtain a better compression result [24]. In the same setting as eq. (9) we can use larger values of β to force Information Dropout, and hence, in the case of reconstruction, a VAE, to recover representations that are increasingly more compressed and also disentangled. This fact is implicitly used in contemporary work [6], that derive the loss in eq. (9) taking inspiration from experimental evidence in neuroscience. They empirically verify that, as expected from this theoretical derivation, for higher values of β the representation $\mathbf{z}$ recovered by the VAE is increasingly more disentangled.

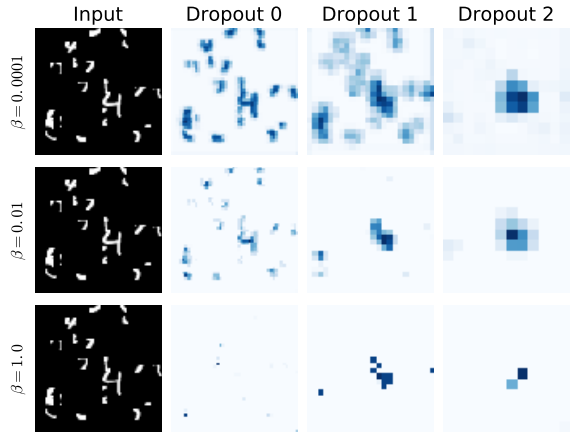


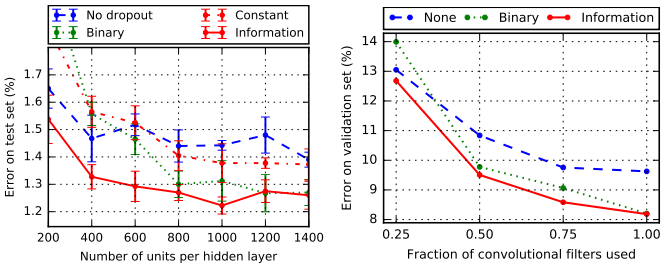
Fig. 2: Plot of the total KL-divergence at each spatial location in the first three Information Dropout layers (of sizes 48x48, 24x24 and 12x12 respectively) of All-CNN-96 (see Table 2) trained on Cluttered MNIST with different values of β. This measures how much information from each part of the image the Information Dropout layer is transmitting to the next layer. For small β information about the nuisances is transmitted to the next layers, while for higher values of β the dropout layers drop the information as soon as the receptive field is big enough to recognize it as a nuisance. The resulting representation is thus more robust to nuisances, improving generalization. Notice that the noise added by Information Dropout is tailored to the specific sample, to the point that the digit can be localized from the noise mask.

Equation (8) has two other important cases: As we have already seen, the case $\gamma = 0$ and $\beta > 0$ is the standard Information Bottleneck Lagrangian: A VAE trained with this loss will focus purely on compression of the input, without squandering resources to also disentangle the representation. In the case $\beta = 0$ and $\gamma > 0$ we obtain instead the standard loss function of Independent Component Analysis (ICA), whereby we try to reconstruct a perfectly disentangled representation of the data, without any constraint on its complexity (quantity of information). While both cases are important on their own right, Proposition 1 does not apply for them, thus the loss function does not generally simplify and cannot be computed in closed form.

8 Experiments

The goal of our experiments is to validate the theory, by showing that indeed increasing noise level yields reduced dependency on nuisance factors, a more disentangled representation, and that by adapting the noise level to the data we can better exploit architectures of limited capacity.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2017.2784440, IEEE Transactions on Pattern Analysis and Machine Intelligence



(a) MNIST (b) CIFAR-10

Fig. 3: (a) Average classification error on MNIST over 3 runs of several dropout methods applied to a fully connected network with three hidden layers and ReLU activations. Information dropout outperforms binary dropout, especially on smaller networks, possibly because dropout severely reduces the already limited capacity of the network, while Information Dropout can adapt the amount of noise to the data and the size of the network. Information dropout also outperforms a dropout layer that uses constant log-normal noise with the same variance, confirming the benefits of adaptive noise. (b) Classification error on CIFAR-10 for several dropout methods applied to the All-CNN-32 network (see Table 2) using Softplus activations.

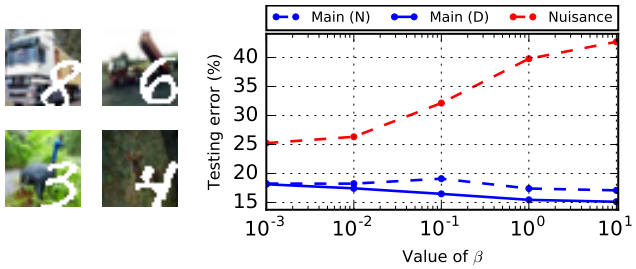


Fig. 4: A few samples from our Occluded CIFAR dataset and the plot of the testing error on the main task (classifying the CIFAR image) and on the nuisance task (classifying the occluding MNIST digit) as β varies. For both tasks, we use the same representation of the data trained for the main task using Information Dropout. For larger values of β the representation is increasingly more invariant to nuisances, making the nuisance classification task harder, but improving the performance on the main task by preventing overfitting. For the nuisance task, we test using the learned noisy representation of the data, since we are interested specifically in the effects of the noise. For the main task, we show the result both using the noisy representation (N), and the deterministic representation (D) obtained by disabling the noise at testing time.

To this end, we first compare Information Dropout with the Dropout baseline on several standard benchmark datasets using different networks architecture, and highlight a few key properties. All the models were implemented using TensorFlow [25]. As [4] also notice, letting the variance of the noise grow excessively leads to poor generalization. To avoid this problem, we constraint $\alpha(x) < 0.7$, so that the maximum variance of the log-normal error distribution will be approximately 1, the same as binary dropout when using a drop probability of 0.5. In all experiments we divide the KL-divergence term by the number of training samples, so that for $\beta = 1$ the scaling of the KL-divergence term in similar to the one used by Variational Dropout (see Section 3).

Cluttered MNIST. To visually asses the ability of Information Dropout to create a representation that is increasingly insensitive to nuisance factors, we train the All-CNN-96 network (Table 2) for classification on a Cluttered MNIST dataset [7], consisting of 96×96 images containing a single MNIST digit together with 21 distractors. The dataset is divided in 50,000 training images and 10,000 testing images. As shown in Figure 2, for small values of β, the network lets through both the objects of interest (digits) and distractors, to upper layers. By increasing the value of β, we force the network to disregard the least discriminative components of the data, thereby building a better representation for the task. This behavior depends on the ability of Information Dropout to learn the structure of the nuisances in the dataset which, unlike other methods, is facilitated by the ability to select noise level on a per-sample basis.

Occluded CIFAR. Occlusions are a fundamental phenomenon in vision, for which it is difficult to hand-design invariant representations. To assess that the approximate minimal sufficient representation produced by Information Dropout has this invariance property, we created a new dataset by occluding images from CIFAR-10 with digits from MNIST (Figure 4). We train the All-CNN-32 network (Table 2) to classify the CIFAR image. The information relative to the occluding MNIST digit is then a nuisance for the task, and therefore should be excluded from the final representation. To test this, we train a secondary network to classify the nuisance MNIST digit using only the the representation learned for the main task. When training with small values of β, the network has very little pressure to limit the effect of nuisances in the representation, so we expect the nuisance classifier to perform better. On the other hand, increasing the value of β we expect its performance to degrade, since the representation will become increasingly minimal, and therefore invariant to nuisances. The results in Figure 4 confirm this intuition.

MNIST and CIFAR-10. Similar to [4], to see the effect of Information Dropout on different network sizes and architectures, we train on MNIST a network with 3 fully connected hidden layers with a variable number of hidden units, and we train on CIFAR-10 [26] the All-CNN-32 convolutional network described in Table 2, using a variable percentage of all the filters. The fully connected network was trained for 80 epochs, using stochastic gradient descent with momentum with initial learning rate 0.07 and dropping the learning rate by 0.1 at 30 and 70 epochs. The CNN was trained for 160 epochs with initial learning rate 0.1 and dropping the learning rate by 0.1 at 80 and 120 epochs. We show the results in Figure 3. Information Dropout is comparable or outperforms binary dropout, especially on smaller networks. A possible explanation is that dropout severely reduces the already limited capacity of the network, while Information Dropout can adapt the amount of noise to the data and to the size of the network so that the relevant information can still flow to the successive layers. Figure 6 shows how the amount of transmitted information adapts to the size and

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2017.2784440, IEEE Transactions on Pattern Analysis and Machine Intelligence

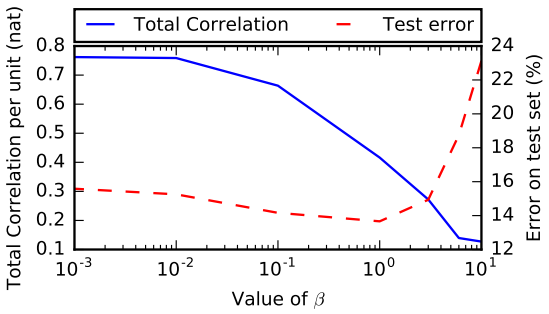
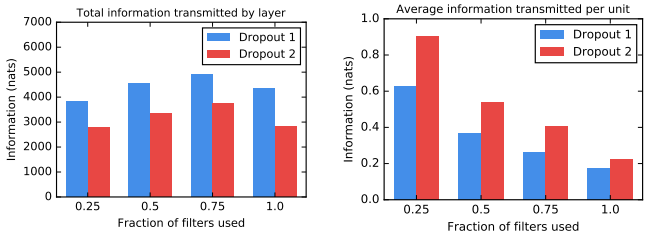


Fig. 5: For different values of β, plot of the test error and total correlation of the final layer of the All-CNN-32 network with Softplus activations trained on CIFAR-10 with 25% of the filters. Increasing β the test error decreases (we prevent overfitting) and the representation becomes increasingly disentangled. When β is too large, it prevents information from passing through, jeopardizing sufficiency and causingi a drastic increase in error.



(a) (b)

Fig. 6: Plots of (a) the total information transmitted through the two dropout layers of a All-CNN-32 network with Softplus activations trained on CIFAR and (b) the average quantity of information transmitted through each unit in the two layers. From (a) we see that the total quantity of information transmitted does not vary much with the number of filters and that, as expected, the second layer transmits less information than the first layer, since prior to it more nuisances have been disentangled and discarded. In (b) we see that when we decrease the number of filters, we force each single unit to let more information flow (i.e. we apply less noise), and that the units in the top dropout layer contain on average more information relevant to the task than the units in the bottom dropout layer.

hierarchical level of the layer.

Disentangling. As we saw Section 6, in the case of Softplus activations, the logarithm of the activations approximately follow a normal distribution. We can then approximate the total correlation using the associated covariance matrix Σ. Precisely, we have

$$\mathrm{TC}(\mathbf{z}) = -\log|\Sigma_0^{-1}\Sigma|$$

where $\Sigma_0 = \operatorname{diag}\Sigma$ is the variance of the marginal distribution. In Figure 5 we plot the testing error and the total correlation of the representation learned by All-CNN-32 on CIFAR-10 when using 25% of the filters for different values of β. As predicted, when β increases the total correlation diminishes, that is, the representation becomes disentangled, and the testing error improves, since we prevent overfitting. When β is to large, information flow is insufficient, and the testing error rapidly increases.

VAE. To validate Section 7, we replicate the basic variational autoencoder of [5], implementing it both with Gaussian latent variables, as in the original, and with an Information Dropout layer. We trained both implementations for 300 epochs dropping the learning rate by 0.1 at 30 and 120 epochs. We report the results in Table 1. The Information Dropout implementation has similar performance to the original, confirming that a variational autoencoder can be considered a special case of Information Dropout.

TABLE 1: Average variational lower-bound $\mathcal{L}$ on the testing dataset for a simple VAE, where the size of the latent variable $\mathbf{z}$ is $256 \cdot k$ and the encoder/decoder each contain $512 \cdot k$ hidden units. The latent variable $\mathbf{z}$ is implemented either using a Gaussian vector or using Information Dropout. Both methods achieve a similar performance.

k	Gaussian	Information
1	-98.8	-100.0
2	-99.0	-99.1
3	-98.7	-99.1

TABLE 2: Structure of the networks used in the experiments. The design of network is based on [27], but we also add batch normalization before the activations of each layer. Depending on the experiment, the ReLU activations are replaced by Softplus activations, and the dropout layer is implemented with binary dropout, Information Dropout or completely removed.

(a) All-CNN-32

Input 32x32
3x3 conv 96 ReLU 3x3 conv 96 ReLU 3x3 conv 96 ReLU stride 2
dropout
3x3 conv 192 ReLU 3x3 conv 192 ReLU 3x3 conv 192 ReLU stride 2
dropout
3x3 conv 192 ReLU 1x1 conv 192 ReLU 1x1 conv 10 ReLU
spatial average
softmax

(b) All-CNN-96

Input 96x96
3x3 conv 32 ReLU 3x3 conv 32 ReLU 3x3 conv 32 ReLU stride 2
dropout
3x3 conv 64 ReLU 3x3 conv 64 ReLU 3x3 conv 64 ReLU stride 2
dropout
3x3 conv 96 ReLU 3x3 conv 96 ReLU 3x3 conv 96 ReLU stride 2
dropout
3x3 conv 192 ReLU 3x3 conv 192 ReLU 3x3 conv 192 ReLU stride 2
dropout
3x3 conv 192 ReLU 1x1 conv 192 ReLU 1x1 conv 10 ReLU
spatial average
softmax

9 Discussion

We relate the Information Bottleneck principle and its associated Lagrangian to seemingly unrelated practices and concepts in deep learning, including dropout, disentanglement, variational autoencoding. For classification tasks, we show how an optimal representation can be achieved by injecting multiplicative noise in the activation functions, and therefore into the gradient computation during learning.

A special case of noise (Bernoulli) results in dropout, which is standard practice originally motivated by ensemble averaging rather than information-theoretic considerations. Better (adaptive) noise models result better exploitation of

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2017.2784440, IEEE Transactions on Pattern Analysis and Machine Intelligence

limited capacity, leading to a method we call Information Dropout. We also establish connections with variational inference and variational autoencoding, and show that "disentangling of the hidden causes" can be measured by total correlation and achieved simply by enforcing independence of the components in the representation prior.

So, what may be done by necessity in some computational systems (noisy computation), turns out to be beneficial towards achieving invariance and minimality. Analogously, what has been done for convenience (assuming a factorized prior) turns out to be the beneficial towards achieving "disentanglement."

Another interpretation of Information Dropout is as a way of biasing the network towards reconstructing representations of the data that are compatible with a Markov chain generative model, making it more suited to data coming from hierarchical models, and in this sense is complementary to architectural constraint, such as convolutions, that instead bias the model toward geometric tasks.

It should be noticed that injecting multiplicative noise to the activations can be thought of as a particular choice of a class of minimizers of the loss function, but can also be interpreted as a regularization terms added to the cost function, or as a particular procedure utilized to carry out the optimization. So the same operation can be interpreted as either of the three key ingredients in the optimization: the function to be minimized, the family over which to minimize, and the procedure with which to minimize. This highlight the intimate interplay between the choice of models and algorithms in deep learning.

Acknowledgments

Work supported by ARO, ONR, AFOSR. We are very grateful to the reviewers for their through analysis of the paper. In particular, we would like to thank one of the anonymous reviewers for for providing an elegant alternative proof to Proposition 1, as well as thoughtful critiques to the theorems.

REFERENCES

[1] S. Soatto and A. Chiuso, "Visual representations: Defining properties and deep approximations," *Proc. of the Intl. Conf. on Learning Representations (ICLR); ArXiv: 1411.7676*, May 2016.

[2] N. Tishby, F. C. Pereira, and W. Bialek, "The information bottleneck method," pp. 368–377, 1999.

[3] N. Srivastava, G. E. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting." *Journal of Machine Learning Research*, vol. 15, no. 1, pp. 1929–1958, 2014.

[4] D. P. Kingma, T. Salimans, and M. Welling, "Variational dropout and the local reparameterization trick," in *Proceedings of the 28th International Conference on Neural Information Processing Systems*, ser. NIPS'15, 2015, pp. 2575–2583.

[5] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," in *Proceedings of the 2nd International Conference on Learning Representations (ICLR)*, no. 2014, 2013.

[6] I. Higgins, L. Matthey, A. Pal, C. Burgess, X. Glorot, M. Botvinick, S. Mohamed, and A. Lerchner, "beta-vae: Learning basic visual concepts with a constrained variational framework," in *Proceedings of the 2nd International Conference on Learning Representations (ICLR)*, no. 2014, 2013.

[7] V. Mnih, N. Heess, A. Graves *et al.*, "Recurrent models of visual attention," in *Advances in Neural Information Processing Systems*, 2014, pp. 2204–2212.

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

[9] S. Wang and C. Manning, "Fast dropout training," in *Proceedings of the 30th International Conference on Machine Learning (ICML-13)*, 2013, pp. 118–126.

[10] G. E. Hinton and D. Van Camp, "Keeping the neural networks simple by minimizing the description length of the weights," in *Proceedings of the sixth annual conference on Computational learning theory*. ACM, 1993, pp. 5–13.

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

[12] K. P. Adragni and R. D. Cook, "Sufficient dimension reduction and prediction in regression," *Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, vol. 367, no. 1906, pp. 4385–4405, 2009.

[13] X. Liu, A. Srivastava, and K. Gallivan, "Optimal linear representations of images for object recognition," in *Computer Vision and Pattern Recognition, 2003. Proceedings. 2003 IEEE Computer Society Conference on*, vol. 1. IEEE, 2003, pp. I–I.

[14] G. Sundaramoorthi, P. Petersen, V. S. Varadarajan, and S. Soatto, "On the set of images modulo viewpoint and contrast changes," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, June 2009.

[15] F. Anselmi, L. Rosasco, and T. Poggio, "On invariance and selectivity in representation learning," *Information and Inference*, 2016.

[16] J. Bruna and S. Mallat, "Classification with scattering operators," in *Proceedings of the 2011 IEEE Conference on Computer Vision and Pattern Recognition*, ser. CVPR '11, 2011, pp. 1561–1566.

[17] Y. Gal and Z. Ghahramani, "Bayesian convolutional neural networks with bernoulli approximate variational inference," *arXiv preprint arXiv:1506.02158*, 2015.

[18] Y. Bengio, A. Courville, and P. Vincent, "Representation learning: A review and new perspectives," *IEEE transactions on pattern analysis and machine intelligence*, vol. 35, no. 8, pp. 1798–1828, 2013.

[19] S. Watanabe, "Information theoretical analysis of multivariate correlation," *IBM Journal of research and development*, vol. 4, no. 1, pp. 66–82, 1960.

[20] M. Studený and J. Vejnarová, "The multiinformation function as a tool for measuring stochastic dependence," in *Learning in graphical models*. Springer, 1998, pp. 261–297.

[21] G. V. Steeg, "Unsupervised learning via total correlation explanation," *arXiv preprint arXiv:1706.08984*, 2017.

[22] P. Comon, "Independent component analysis, a new concept?" *Signal Processing*, vol. 36, pp. 287–314, 1994.

[23] A. J. Bell and T. J. Sejnowski, "An information-maximization approach to blind separation and blind deconvolution," *Neural Computation*, vol. 7, pp. 1129–1159, 1995.

[24] D. P. Kingma, T. Salimans, R. Jozefowicz, X. Chen, I. Sutskever, and M. Welling, "Improved variational inference with inverse autoregressive flow," in *Advances in Neural Information Processing Systems*, 2016, pp. 4743–4751.

[25] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng, "TensorFlow: Large-scale machine learning on heterogeneous systems," 2015, software available from tensorflow.org. [Online]. Available: http://tensorflow.org/

[26] A. Krizhevsky and G. Hinton, "Learning multiple layers of features from tiny images," 2009.

[27] J. T. Springenberg, A. Dosovitskiy, T. Brox, and M. Riedmiller, "Striving for simplicity: The all convolutional net," *arXiv preprint arXiv:1412.6806*, 2014.

Alessandro Achille is a PhD candidate at UCLA, where he joined Prof. Soatto's research group in 2015. He previously obtained his Master's degree in Pure Math (with honors) at the Scuola Normale Superiore di Pisa and the University of Pisa. His research interests include machine learning, variational inference and information theory, with particular focus on artificial and embodied intelligence.

Stefano Soatto received his Ph.D. in Control and Dynamical Systems from the California Institute of Technology in 1996; he joined UCLA in 2000 after being Assistant and then Associate Professor of Electrical and Biomedical Engineering at Washington University, and Research Associate in Applied Sciences at Harvard University. Between 1995 and 1998 he was also Ricercatore in the Department of Mathematics and Computer Science at the University of Udine - Italy. He received his D.Ing. degree (highest honors) from the University of Padova - Italy in 1992. His general research interests are in Computer Vision and Nonlinear Estimation and Control Theory. In particular, he is interested in ways for computers to use sensory information (e.g. vision, sound, touch) to interact with humans and the environment.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2017.2784440, IEEE Transactions on Pattern Analysis and Machine Intelligence

APPENDIX A
COMPUTATIONS

Proposition (Information dropout cost for ReLU activations)**.** *Let $z = \varepsilon \cdot f(x)$, where $\varepsilon \sim p_\alpha(\varepsilon)$, and assume $p(z) = q\delta_0(z) + c/z$. Then, assuming $f(x) \neq 0$, we have*

$$\mathrm{KL}(p_\theta(z|x) \parallel p(z)) = -H(p_{\alpha(x)}(\log \varepsilon)) + \log(c)$$

In particular, if $p_\alpha(\varepsilon)$ is chosen to be the log-normal distribution $p_\alpha(\varepsilon) = \log \mathcal{N}(0, \alpha_\theta^2(x))$, we have

$$\mathrm{KL}(p_\theta(z|x) \parallel p(z)) = -\log \alpha_\theta(x) + const.$$

If instead $f(x) = 0$, we have

$$\mathrm{KL}(p_\theta(z|x) \parallel p(z)) = -\log q.$$

Proof. Since the measures $P_\theta(z|x)$ and $P_\theta(z)$ are not absolutely continuous with respect to the Lebesgue measure, it will be convenient to use the following more general definition of KL divergence:[3]

$$\mathrm{KL}(P(z) \parallel Q(z)) := \int \log \frac{dP}{dQ} dP,$$

where we assume that $P \ll Q$, so that the density dP/dQ exists. Recall that the KL-divergence is invariant under invertible parameter transformations, meaning that for any invertible transformation $\psi(z)$ we have

$$\mathrm{KL}(P(z) \parallel Q(z)) = \mathrm{KL}(P(\psi(z)) \parallel Q(\psi(z)).$$

Let's first consider the case $f(x) \neq 0$, in which case we also have $z \neq 0$ almost surely. Since $p(\log z) = c$ for $z > 0$, we can write

$$\frac{dP(\log z|x)}{dP(\log z)} = p_{\alpha(x)}(\log \epsilon)/c.$$

Therefore, we have

$$\begin{aligned}
\mathrm{KL}(P_\theta(z|x) \parallel P_\theta(z)) &= \mathrm{KL}(P_\theta(\log z|x) \parallel P_\theta(\log z)) \\
&= \int \log \left(\frac{dP_\theta(\log z|x)}{dP_\theta(\log z)} \right) p_\theta(\log z|x) dz \\
&= \int \log \left(p_{\alpha(x)}(\log \varepsilon) \right) p_{\alpha(x)}(\log \varepsilon) d\varepsilon \\
&\quad - \log c \\
&= -H(p_{\alpha(x)}(\log \varepsilon)) - \log c,
\end{aligned}$$

which gives us the first statement. Now, notice that if $p_\alpha(z)(\epsilon) = \log \mathcal{N}(0, \alpha_\theta^2(x))$, then by definition we have $p_{\alpha(x)}(\log \epsilon) = \mathcal{N}(0, \alpha_\theta^2(x))$. We can then use the known formula for the entropy of a Gaussian distribution to obtain

$$H(\mathcal{N}(0, \alpha)) = \log \alpha_\theta(x) + \frac{1}{2} \log(2\pi e),$$

which gives us the second statement.

Finally, consider the case $f(x) = 0$, in which case $z = 0$, and therefore $p(z|x)$ reduces to the singular distribution $p(z|x) = \delta_0(z)$. Again, the measure $P(z|x)$ is absolutely continuous with respect to $P(z)$, and we have

$$\frac{dP(z|x)}{dP(z)} = \begin{cases} 1/q & \text{if } z = 0, \\ 0 & \text{else.} \end{cases}$$

We can then compute

$$\begin{aligned}
\mathrm{KL}(p_\theta(z|x) \parallel p(z)) &= \int \log \frac{dP(z|x)}{dP(z)} dP \\
&= -\log q,
\end{aligned}$$

as wanted. □

Proposition (Information dropout cost for Softplus activations)**.** *Let $z = \varepsilon \cdot f(x)$, where $\varepsilon \sim p_\alpha(\varepsilon) = \log \mathcal{N}(0, \alpha_\theta^2(x))$, and assume $p_\theta(z) = \log \mathcal{N}(\mu, \sigma^2)$. Then, we have*

$$\mathrm{KL}(p_\theta(z|x) \parallel p(z)) = \frac{1}{2\sigma^2} \left(\alpha^2(x) + \mu^2 \right) - \log \frac{\alpha(x)}{\sigma} - \frac{1}{2}.$$

Proof. Since the KL-divergence is invariant for invertible reparametrizations (as in the proof of the previous proposition), the divergence between two log-normal distributions is equal to the divergence between the corresponding normal distributions. Therefore, using the known formula for the KL divergence of normals, we get the desired result. □

APPENDIX B
DISENTANGLEMENT

In this appendix, we show that the minimization problem

$$\min_p \frac{1}{N} \sum_{i=1}^N \mathbb{E}_{z \sim p(\mathbf{z}|\mathbf{x}_i)}[-\log p(\mathbf{y}_i|\mathbf{z})] + \\ + \beta \left\{ \mathrm{KL}(p(\mathbf{z}|\mathbf{x}_i) \parallel p(\mathbf{z})) + \mathrm{TC}(\mathbf{z}) \right\},$$

which is difficult in general since we do not have access to the joint distribution $p(\mathbf{z})$, is equivalent to the following simpler optimization problem in two variables

$$\min_{p,q} \frac{1}{N} \sum_{i=1}^N \mathbb{E}_{z \sim p(\mathbf{z}|\mathbf{x}_i)}[-\log p(\mathbf{y}_i|\mathbf{z})] + \\ + \beta \, \mathrm{KL}(p(\mathbf{z}|\mathbf{x}_i) \parallel \textstyle\prod_{i=1}^{|z|} q_i(\mathbf{z}_i)).$$

In the following proposition, for simplicity, we concentrate on discrete random variables.

Proposition. *Let $\mathbf{z} = (z_1, \ldots, z_n)$ be a discrete random variable, let $p(\mathbf{z}|\mathbf{x})$ be a generic probability distribution, and let $q(\mathbf{z}) = \prod_{i=1}^n q_i(z_i)$ be a factorized prior distribution. Then, for any function $F(p)$, a minimization problem in the form*

$$\text{minimize}_{p,q} \quad F(p) + \beta \mathbb{E}_x[\mathrm{KL}(p(\mathbf{z}|\mathbf{x}) \parallel q(\mathbf{z}))],$$

is equivalent to

$$\text{minimize}_p \quad F(p) + \beta \left\{ I_p(\mathbf{z}; \mathbf{x}) + \mathrm{TC}_p(\mathbf{z}) \right\},$$

where $I_p(\mathbf{z}; \mathbf{x})$ is the mutual information and $\mathrm{TC}_p(\mathbf{z})$ is the total correlation of $\mathbf{z}$, assuming $\mathbf{z} \sim p(\mathbf{z})$.

Proof. First, notice that for any $p(\mathbf{z}|\mathbf{x})$ we have

$$\mathrm{KL}(p(\mathbf{z}) \parallel \textstyle\prod_i p(z_i)) = \min_{q(z_1), \ldots, q(z_n)} \mathbb{E}\Big[\log \frac{p(\mathbf{z}|\mathbf{x})}{\prod_i q(z_i)} \Big],$$

3. An alternative approach working with (improper) density functions instead of measures is to approximate the Dirac delta's with Gaussian distributions and take the limit using the dominated convergence theorem, as suggested by an anonymous reviewer.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2017.2784440, IEEE Transactions on Pattern Analysis and Machine Intelligence

since for any $p(\mathbf{z})$ and $q(\mathbf{z}) = \prod_i q(z_i)$ we have

$$\begin{aligned}
\mathrm{KL}(p(\mathbf{z}) \parallel \textstyle\prod_i p(z_i)) &= \mathbb{E}\left[\log \frac{p(\mathbf{z})}{\prod_i p(z_i)}\right] \\
&= \mathbb{E}\left[\log \frac{p(\mathbf{z})}{\prod_i q(z_i)} \cdot \frac{\prod_i q(z_i)}{\prod_i p(z_i)}\right] \\
&= \mathbb{E}\left[\log \frac{p(\mathbf{z})}{\prod_i q(z_i)}\right] \\
&\quad - \sum_{i=1}^{n} \mathrm{KL}(p(z_i) \parallel q(z_i)) \\
&\leq \mathbb{E}\left[\log \frac{p(\mathbf{z})}{\prod_i q(z_i)}\right],
\end{aligned}$$

and equality trivially holds when $q(z_i) = p(z_i)$. Using the above identity, we now have

$$\begin{aligned}
&I(\mathbf{x}; \mathbf{z}) + \mathrm{KL}(p(\mathbf{z}|\mathbf{x}) \parallel \textstyle\prod_i p(\mathbf{z}_i|x)) \\
&\quad = \min_{q(z_1),\ldots,q(z_n)} \mathbb{E}\Big[\log \frac{p(\mathbf{z}|\mathbf{x})}{p(\mathbf{z})} \cdot \frac{p(\mathbf{z})}{\prod_i q(z_i)}\Big] \\
&\quad = \min_{q(z_1),\ldots,q(z_n)} \mathbb{E}\left[\log \frac{p(\mathbf{z}|\mathbf{x})}{\prod_i q(z_i)}\right].
\end{aligned}$$

□