The Deterministic Information Bottleneck

DJ Strouse
Physics Department
Princeton University
dstrouse@princeton.edu

David J Schwab
Physics Department
Northwestern University
david.schwab@northwestern.edu

Abstract

Lossy compression fundamentally involves a decision about what is relevant and what is not. The information bottleneck (IB) by Tishby, Pereira, and Bialek formalized this notion as an information-theoretic optimization problem and proposed an optimal tradeoff between throwing away as many bits as possible, and selectively keeping those that are most important. Here, we introduce an alternative formulation, the deterministic information bottleneck (DIB), that we argue better captures this notion of compression. As suggested by its name, the solution to the DIB problem is a deterministic encoder, as opposed to the stochastic encoder that is optimal under the IB. We then compare the IB and DIB on synthetic data, showing that the IB and DIB perform similarly in terms of the IB cost function, but that the DIB vastly outperforms the IB in terms of the DIB cost function. Moreover, the DIB offered a 1-2 order of magnitude speedup over the IB in our experiments. Our derivation of the DIB also offers a method for continuously interpolating between the soft clustering of the IB and the hard clustering of the DIB.

1 INTRODUCTION

Compression is a ubiquitous task for humans and machines alike [Cover & Thomas (2006), MacKay (2002)]. For example, machines must turn the large pixel grids of color that form pictures into small files capable of being shared quickly on the web [Wallace (1991)], humans must compress the vast stream of ongoing sensory information they receive into small changes in the brain that form memories [Kandel et al (2000)], and data scientists must turn large amounts of high-dimensional and messy data into more manageable and interpretable clusters [MacKay (2002)].

Lossy compression involve an implicit decision about what is relevant and what is not [Cover & Thomas (2006), MacKay (2002)]. In the example of image compression, the algorithms we use deem some features essential to representing the subject matter well, and others are thrown away. In the example of human memory, our brains deem some details important enough to warrant attention, and others are forgotten. And in the example of data clustering, information about some features is preserved in the mapping from data point to cluster ID, while information about others is discarded.

In many cases, the criterion for "relevance" can be described as information about some other variable(s) of interest. Let's call X the signal we are compressing, T the compressed version, Y the other variable of interest, and $I(T;Y)$ the "information" that T has about Y (we will formally define this later). For human memory, X is past sensory input, T the brain's internal representation (e.g. the activity of a neural population, or the strengths of a set of synapses), and Y the features of the future environment that the brain is interested in predicting, such as extrapolating the position of a moving object. Thus, $I(T;Y)$ represents the predictive power of the memories formed [Palmer et al (2015)]. For data clustering, X is the original data, T is the cluster ID, and Y is the target for prediction, for example purchasing or ad-clicking behavior in a user segmentation problem. In summary, a good compression algorithm can be described as a tradeoff between the compression of a signal and the selective maintenance of the "relevant" bits that help predict another signal.

This problem was formalized as the "information bottleneck" (IB) by Tishby, Pereira, and Bialek [Tishby (1999)]. In their formulation, compression was measured by the mutual information $I(X;T)$. This compression metric has its origins in rate-distortion theory and channel coding, where $I(X;T)$ represents the maximal information transfer rate, or capacity, of the communication channel between X and T [Cover & Thomas (2006)]. While this approach has its applications, often one is more interested in directly restricting the amount of resources required to represent T, represented by the entropy $H(T)$. This latter notion comes

from the source coding literature and implies a restriction on the *representational cost* of T. In the case of human memory, for example, $H(T)$ would roughly correspond to the number of neurons or synapses required to represent or store a sensory signal X. In the case of data clustering, $H(T)$ is related to the number of clusters.

In the following paper, we introduce an alternative formulation of the IB, replacing the compression measure $I(X;T)$ with $H(T)$, thus emphasizing contraints on representation, rather than communication. We begin with a general introduction to the IB. Then, we introduce our alternative formulation, which we call the deterministic information bottleneck (DIB). Finally, we compare the IB and DIB solutions on synthetic data to help illustrate their differences.

2 THE ORIGINAL INFORMATION BOTTLENECK (IB)

Given the joint distribution $p(x,y)$, the encoding distribution $q(t \mid x)$ is obtained through the following "information bottleneck" (IB) optimization problem:

$$\min_{q(t|x)} L[q(t \mid x)] = I(X;T) - \beta I(T;Y)\,, \qquad (1)$$

subject to the Markov constraint $T \leftrightarrow X \leftrightarrow Y$. Here $I(X;T)$ denotes the mutual information between X and T, that is $I(X;T) \equiv H(T) - H(T \mid X) = \sum_{x,t} p(x,t) \log\left(\frac{p(x,t)}{p(x)p(t)}\right) = D_{KL}[p(x,t) \mid p(x)\,p(t)]$,[1] where D_{KL} denotes the Kullback-Leibler divergence.[2] The first term in the cost function is meant to encourage compression, while the second relevance. β is a non-negative free parameter representing the relative importance of compression and relevance, and our solution will be a function of it. The Markov constraint simply enforces the probabilistic graphical structure of the task; the compressed representation T is a (possibly stochastic) function of X and can only get information about Y through X. Note that we are using p to denote distributions that are given and fixed, and q to denote distributions that we are free to change and that are being updated throughout the optimization process.

Through a standard application of variational calculus (see Section 7 for a detailed derivation of the solution to a more general problem introduced below), one finds the formal solution:[3]

$$q(t \mid x) = \frac{q(t)}{Z(x,\beta)} \exp[-\beta D_{\mathrm{KL}}[p(y \mid x) \mid q(y \mid t)]] \qquad (2)$$

$$q(y \mid t) = \frac{1}{q(t)} \sum_x q(t \mid x)\, p(x,y)\,, \qquad (3)$$

where $Z(x,\beta) \equiv \exp\left[-\frac{\lambda(x)}{p(x)} - \beta \sum_y p(y \mid x) \log \frac{p(y|x)}{p(y)}\right]$ is a normalization factor, and $\lambda(x)$ is a Lagrange multiplier that enters enforcing normalization of $q(t \mid x)$.[4] To get an intuition for this solution, it is useful to take a clustering perspective - since we are compressing X into T, many X will be mapped to the same T and so we can think of the IB as "clustering" xs into their cluster labels t. The solution $q(t \mid x)$ is then likely to map x to t when $D_{\mathrm{KL}}[p(y \mid x) \mid q(y \mid t)]$ is small, or in other words, when the distributions $p(y \mid x)$ and $q(y \mid t)$ are similar. These distributions are similar to the extent that x and t provide similar information about y. In summary, inputs x get mapped to clusters t that maintain information about y, as was desired.

This solution is "formal" because the first equation depends on the second and vice versa. However, [Tishby (1999)] showed that an iterative approach can be built on the the above equations which provably converges to a local optimum of the IB cost function (eqn 1).

Starting with some initial distributions $q^{(0)}(t \mid x)$, $q^{(0)}(t)$, and $q^{(0)}(y \mid t)$, the n^{th} update is given by:

$$d^{(n-1)}(x,t) \equiv D_{\mathrm{KL}}\left[p(y \mid x) \mid q^{(n-1)}(y \mid t)\right] \qquad (4)$$

$$q^{(n)}(t \mid x) = \frac{q^{(n-1)}(t)}{Z(x,\beta)} \exp\left[-\beta d^{(n-1)}(x,t)\right] \qquad (5)$$

$$q^{(n)}(t) = \sum_x p(x)\, q^{(n)}(t \mid x) \qquad (6)$$

$$q^{(n)}(y \mid t) = \frac{1}{q^{(n)}(t)} \sum_x q^{(n)}(t \mid x)\, p(x,y)\,. \qquad (7)$$

[1] Implicit in the summation here, we have assumed that X, Y, and T are discrete. We will be keeping this assumption throughout for convenience of notation, but note that the IB generalizes naturally to X, Y, and T continuous by simply replacing the sums with integrals (see, for example, [Chechik et al (2005)]).

[2] For those unfamiliar with it, mutual information is a very general measure of how related two variables are. Classic correlation measures typically assume a certain form of the relationship between two variables, say linear, whereas mutual information is agnostic as to the details of the relationship. One intuitive picture comes from the entropy decomposition: $I(X;Y) \equiv H(X) - H(X \mid Y)$. Since entropy measures uncertainty, mutual information measures the *reduction in uncertainty* in one variable when observing the other. Moreover, it is symmetric ($I(X;Y) = I(Y;X)$), so the information is *mutual*. Another intuitive picture comes from the D_{KL} form: $I(X;Y) \equiv D_{KL}[p(x,y) \mid p(x)\,p(y)]$. Since D_{KL} measures the distance between two probability distributions, the mutual information quantifies how far the relationship between x and y is from a probabilistically independent one, that is how far the joint $p(x,y)$ is from the factorized $p(x)\,p(y)$.

[3] For the reader familiar with rate-distortion theory, eqn 2 can be viewed as the solution to a rate-distortion problem with distortion measure given by the KL-divergence term in the exponent.

[4] More explicitly, our cost function L also implicitly includes a term $\sum_x \lambda(x)\left[1 - \sum_t q(t|x)\right]$ and this is where $\lambda(x)$ comes in to the equation. See Section 7 for details.

Note that the first pair of equations is the only "meaty" bit; the rest are just there to enforce consistency with the laws of probability (e.g. that marginals are related to joints as they should be). In principle, with no proof of convergence to a global optimum, it might be possible for the solution obtained to vary with the initialization, but in practice, the cost function is "smooth enough" that this does not seem to happen. This algorithm is summarized in algorithm 1. Note that while the general solution is iterative, there is at least one known case in which an analytic solution is possible, name when X and Y are jointly Gaussian [Chechik et al (2005)].

Algorithm 1 - The information bottleneck (IB) method.

Given $p(x, y)$, $\beta \geq 0$
Initialize $q^{(0)}(t \mid x)$ and set $n = 0$
$q^{(0)}(t) = \sum_x p(x)\, q^{(0)}(t \mid x)$
$q^{(0)}(y \mid t) = \frac{1}{q^{(0)}(t)} \sum_x p(x, y)\, q^{(0)}(t \mid x)$
while not converged **do**
    $n = n + 1$
    $d^{(n-1)}(x, t) \equiv D_{\mathrm{KL}}\left[p(y \mid x) \mid q^{(n-1)}(y \mid t)\right]$
    $q^{(n)}(t \mid x) = \frac{q^{(n-1)}(t)}{Z(x,\beta)} \exp\left[-\beta d^{(n-1)}(x, t)\right]$
    $q^{(n)}(t) = \sum_x p(x)\, q^{(n)}(t \mid x)$
    $q^{(n)}(y \mid t) = \frac{1}{q^{(n)}(t)} \sum_x q^{(n)}(t \mid x)\, p(x, y)$
end while

In summary, given the joint distribution $p(x, y)$, the IB method extracts a compressive encoder $q(t \mid x)$ that selectively maintains the bits from X that are informative about Y. As the encoder is a function of the free parameter β, we can visualize the entire family of solutions on a curve (figure 1), showing the tradeoff between compression (on the x-axis) and relevance (on the y-axis). For small β, compression is more important than prediction and we find ourselves at the bottom left of the curve in the high compression, low prediction regime. As β increases, prediction becomes more important relative to compression, and we see that both $I(X;T)$ and $I(T;Y)$ increase. At some point, $I(T;Y)$ saturates, because there is no more information about Y that can be extracted from X (either because $I(T;Y)$ has reached $I(X;Y)$ or because T has too small cardinality). Note that the region below the curve is shaded because this area is feasible; for suboptimal $q(t \mid x)$, solutions will lie in this region. Optimal solutions will of course lie on the curve, and no solutions will lie above the curve.

Additional work on the IB has highlighted its relationship with maximum likelihood on a multinomial mixture model [Slonim & Weiss (2002)] and canonical correlation analysis [Creutzig et al (2009)] (and therefore linear Gaussian models [Bach & Jordan (2005)] and slow feature analysis [Turner & Sahani (2007)]). Applications have included speech recognition [Hecht & Tishby (2005), Hecht & Tishby (2008), Hecht et al (2009)], topic modeling [Slonim & Tishby (2000), Slonim & Tishby (2001), Bekkerman et al (2001), Bekkerman et al (2003)], and neural coding [Schneidman et al (2002), Palmer et al (2015)]. Most recently, the IB has even been proposed as a method for benchmarking the performance of deep neural networks [Tishby & Zaslavsky (2015)].

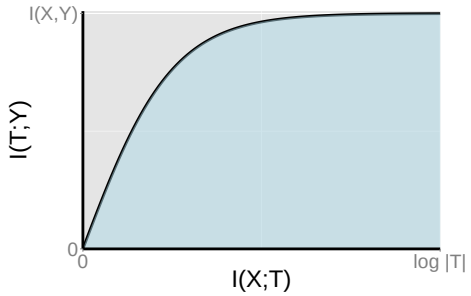

Figure 1: **An illustrative IB curve.** $I(T;Y)$ is the relevance term from eqn 1; $I(X;T)$ is the compression term. $I(X;Y)$ is an upper bound on $I(T;Y)$ since T only gets its information about Y via X. $\log(|T|)$, where $|T|$ is the cardinality of the compression variable, is a bound on $I(X;T)$ since $I(X;T) = H(T) - H(T \mid X) \leq H(T) \leq \log(|T|)$.

3 THE DETERMINISTIC INFORMATION BOTTLENECK (DIB)

Our motivation for introducing an alternative formulation of the information bottleneck is rooted in the "compression term" of the IB cost function; there, the minimization of the mutual information $I(X;T)$ represents compression. As discussed above, this measure of compression comes from the channel coding literature and implies a restriction on the *communication cost* between X and T. Here, we are interested in the source coding notion of compression, which implies a restriction on the *representational cost* of T. For example, in neuroscience, there is a long history of work on "redundancy reduction" in the brain in the form of minimizing $H(T)$ [Barlow (1981), Barlow (2001), Barlow (2001)].

Let us call the original IB cost function L_{IB}, that is $L_{\mathrm{IB}} \equiv I(X;T) - \beta I(T;Y)$. We now introduce the deterministic information bottleneck (DIB) cost function:

$$L_{\mathrm{DIB}}[q(t \mid x)] \equiv H(T) - \beta I(T;Y)\,, \tag{8}$$

which is to be minimized over $q(t \mid x)$ and subject to the same Markov constraint as the original formulation (eqn 1). The "deterministic" in its name will become clear below.

To see the distinction between the two cost functions, note that:

$$L_{\mathrm{IB}} - L_{\mathrm{DIB}} = I(X;T) - H(T) \tag{9}$$
$$= -H(T \mid X)\,, \tag{10}$$

where we have used the decomposition of the mutual information $I(X;T) = H(T) - H(T \mid X)$. $H(T \mid X)$ is sometimes called the "noise entropy" and measures the stochasticity in the mapping from X to T. Since we are minimizing these cost functions, this means that the IB cost function *encourages* stochasticity in the encoding distribution $q(t \mid x)$ relative to the DIB cost function. In fact, we will see that by removing this encouragement of stochasticity, the DIB problem actually produces a deterministic encoding distribution, i.e. an encoding *function*, hence the "deterministic" in its name.

Naively taking the same variational calculus approach as for the IB problem, one cannot solve the DIB problem.[5] To make this problem tractable, we are going to define a family of cost functions of which the IB and DIB cost functions are limiting cases. That family, indexed by α, is defined as:[6]

$$L_\alpha \equiv H(T) - \alpha H(T \mid X) - \beta I(T;Y)\,. \tag{11}$$

Clearly, $L_{\text{IB}} = L_1$. However, instead of looking at L_{DIB} as the $\alpha = 0$ case, we'll define the DIB solution $q_{\text{DIB}}(t \mid x)$ as the $\alpha \to 0$ limit of the solution to the generalized problem $q_\alpha(t \mid x)$:[7]

$$q_{\text{DIB}}(t \mid x) \equiv \lim_{\alpha \to 0} q_\alpha(t \mid x)\,. \tag{12}$$

Taking the variational calculus approach to minimizing L_α (under the Markov constraint), we get the following solution for the encoding distribution (see Section 7 for the derivation and explicit form of the normalization factor $Z(x, a, \beta)$):

$$d_\alpha(x,t) \equiv D_{\text{KL}}[p(y \mid x) \mid q_\alpha(y \mid t)] \tag{13}$$

$$\ell_{\alpha,\beta}(x,t) \equiv \log q_\alpha(t) - \beta d_\alpha(x,t) \tag{14}$$

$$q_\alpha(t \mid x) = \frac{1}{Z(x,\alpha,\beta)} \exp\left[\frac{1}{\alpha}\ell_{\alpha,\beta}(x,t)\right] \tag{15}$$

$$q_\alpha(y \mid t) = \frac{1}{q_\alpha(t)} \sum_x q_\alpha(t \mid x)\, p(x,y)\,. \tag{16}$$

Note that the last equation is just eqn 3, since this just follows from the Markov constraint. With $\alpha = 1$, we can see that the other three equations just become the IB solution from eqn 2, as should be the case.

Before we take the $\alpha \to 0$ limit, note that we can now write a generalized IB iterative algorithm (indexed by α) which includes the original as a special case ($\alpha = 1$):

$$d_\alpha^{(n-1)}(x,t) \equiv D_{\text{KL}}\left[p(y \mid x) \mid q_\alpha^{(n-1)}(y \mid t)\right] \tag{17}$$

$$\ell_{\alpha,\beta}^{(n-1)}(x,t) \equiv \log q_\alpha^{(n-1)}(t) - \beta d_\alpha^{(n-1)}(x,t) \tag{18}$$

$$q_\alpha^{(n)}(t \mid x) = \frac{1}{Z(x,\alpha,\beta)} \exp\left[\frac{1}{\alpha}\ell_{\alpha,\beta}^{(n-1)}(x,t)\right] \tag{19}$$

$$q_\alpha^{(n)}(t) = \sum_x p(x)\, q_\alpha^{(n)}(t \mid x) \tag{20}$$

$$q_\alpha^{(n)}(y \mid t) = \frac{1}{q_\alpha^{(n)}(t)} \sum_x q_\alpha^{(n)}(t \mid x)\, p(x,y)\,. \tag{21}$$

This generalized algorithm can be used in its own right, however we will not discuss that option further here.

For now, we take the limit $\alpha \to 0$ and see that something interesting happens with $q_\alpha(t \mid x)$ - the argument of the exponential begins to blow up. For a fixed x, this means that $q(t \mid x)$ will collapse into a delta function at the value of t which maximizes $\log q(t) - \beta D_{\text{KL}}[p(y \mid x) \mid q(y \mid t)]$. That is:

$$\lim_{\alpha \to 0} q_\alpha(t \mid x) = f : X \to T, \tag{22}$$

where:

$$f(x) \equiv t^* = \operatorname*{argmax}_t \ell(x,t) \tag{23}$$

$$\ell(x,t) \equiv \log q(t) - \beta D_{\text{KL}}[p(y \mid x) \mid q(y \mid t)]\,. \tag{24}$$

So, as anticipated, the solution to the DIB problem is a deterministic encoding distribution. The $\log q(t)$ above encourages that we use as few values of t as possible, via a "rich-get-richer" scheme that assigns each x preferentially to a t already with many xs assigned to it. The KL divergence term, as in the original IB problem, just makes sure we pick ts which retain as much information from x about y as possible. The parameter β, as in the original problem, controls the tradeoff between how much we value compression and prediction.

Also like in the original problem, the solution above is only a formal solution, since eqn 15 depends on eqn 16 and vice versa. So we will again need to take an iterative approach; in analogy to the IB case, we repeat the following updates to convergence (from some initialization):[8]

[5] When you take the variational derivative of L_{DIB} + Lagrange multiplier term with respect to $q(t \mid x)$ and set it to zero, you get no explicit $q(t \mid x)$ term, and it is therefore not obvious how to solve these equations. We cannot rule that that approach is possible, of course; we have just here taken a different route.

[6] Note that for $\alpha < 1$, we cannot allow T to be continuous since $H(T)$ can become infinitely negative, and the optimal solution in that case will trivially be a delta function over a single value of T for all X, across all values of β. This is in constrast to the IB, which can handle continuous T. In any case, we continue to assume discrete X, Y, and T for convenience.

[7] Note a subtlety here that we cannot claim that the q_{DIB} is the solution to L_{DIB}, for although $L_{\text{DIB}} = \lim_{\alpha\to 0} L_\alpha$ and $q_{\text{DIB}} = \lim_{\alpha\to 0} q_\alpha$, the solution of the limit need not be equal to the limit of the solution. It would, however, be surprising if that were not the case.

[8] Note that, if at step m no xs are assigned to a particular $t = t^*$, then $q_m(t^*) = 0$ and for all future steps $n > m$, no xs will

$$d^{(n-1)}(x,t) \equiv D_{\mathrm{KL}}\Big[p(y \mid x) \mid q^{(n-1)}(y \mid t)\Big] \tag{25}$$

$$\ell_\beta^{(n-1)}(x,t) \equiv \log q(t) - \beta d^{(n-1)}(x,t) \tag{26}$$

$$f^{(n)}(x) = \operatorname*{argmax}_t \ell_\beta^{(n-1)}(x,t) \tag{27}$$

$$q^{(n)}(t \mid x) = \delta\Big(t - f^{(n)}(x)\Big) \tag{28}$$

$$q^{(n)}(t) = \sum_x q^{(n)}(t \mid x)\, p(x) \tag{29}$$

$$= \sum_{x:f^{(n)}(x)=t} p(x) \tag{30}$$

$$q^{(n)}(y \mid t) = \frac{1}{q^{(n)}(t)} \sum_x q^{(n)}(t \mid x)\, p(x,y) \tag{31}$$

$$= \frac{\sum_{x:f^{(n)}(x)=t} p(x,y)}{\sum_{x:f^{(n)}(x)=t} p(x)}. \tag{32}$$

This process is summarized in algorithm 2.

Like with the IB, the DIB solutions can be plotted as a function of β. However, in this case, it is more natural to plot $I(T;Y)$ as a function of $H(T)$, rather than $I(X;T)$. That said, in order to compare the IB and DIB, they need to be plotted in the same plane. When plotting in the $I(X;T)$ plane, the DIB curve will of course lie below the IB curve, since in this plane, the IB curve is optimal; the opposite will be true when plotting in the $H(T)$ plane. Comparisons with experimental data can be performed in either plane.

Algorithm 2 - The deterministic information bottleneck (DIB) method.

Given $p(x,y)$, $\beta \geq 0$
Initialize $f^{(0)}(x)$ and set $n = 0$
$q^{(0)}(t) = \sum_{x:f^{(0)}(x)=t} p(x)$
$q^{(0)}(y \mid t) = \frac{\sum_{x:f^{(0)}(x)=t} p(x,y)}{\sum_{x:f^{(0)}(x)=t} p(x)}$
while not converged **do**
    $n = n + 1$
    $d^{(n-1)}(x,t) \equiv D_{\mathrm{KL}}\big[p(y \mid x) \mid q^{(n-1)}(y \mid t)\big]$
    $\ell_\beta^{(n-1)}(x,t) \equiv \log q(t) - \beta d^{(n-1)}(x,t)$
    $f^{(n)}(x) = \operatorname*{argmax}_t \ell_\beta^{(n-1)}(x,t)$
    $q^{(n)}(t) = \sum_{x:f^{(n)}(x)=t} p(x)$
    $q^{(n)}(y \mid t) = \frac{\sum_{x:f^{(n)}(x)=t} p(x,y)}{\sum_{x:f^{(n)}(x)=t} p(x)}$
end while

ever again be assigned to t^* since $\log q_n(t^*) = -\infty$. In other words, the number of ts "in use" can only decrease during the iterative algorithm above (or remain constant). Thus, it seems plausible that our solution will not depend on the cardinality of T, provided it is chosen to be large enough.

4 COMPARISON OF IB AND DIB

To get an idea of how the IB and DIB solutions differ in practice, we generated a series of random joint distributions $p(x,y)$, solved for the IB and DIB solutions for each, and compared them in both the IB and DIB plane. To generate the $p(x,y)$, we first sampled $p(x)$ from a symmetric Dirichlet distribution with concentration parameter α_x (so $p(x) \sim \mathrm{Dir}[\alpha_x]$), and then sampled each row of $p(y \mid x)$ from another symmetric Dirichlet distribution with concentration parameter α_y (so $p(y \mid x) \sim \mathrm{Dir}[\alpha_y]\,, \forall x$). Since the number of clusters in use for both IB and DIB can only decrease from iteration to iteration (see footnote 8), we always initialized $|T| = |X|$.[9] For the DIB, we initialized the cluster assignments to be as even across the cluster as possible, i.e. each data points belonged to its own cluster. For IB, we initalized the cluster assignments to a normalized draw of a uniform random vector.

An illustrative pair of solutions is shown in figure 2. The key feature to note is that, while performance of the IB and DIB solutions are very similar in the IB plane, their behavior differs drastically in the DIB plane.

Perhaps most unintuitive is the behavior of the IB solution in the DIB plane. To understand this behavior, recall that the IB's compression term is the mutual information $I(X,T) = H(T) - H(T \mid X)$. This term is minimized by any $q(t \mid x)$ that maps ts independently of xs. Consider two extremes of such mappings. One is to map all values of X to a single value of T; this leads to $H(T) = H(T \mid X) = I(X,T) = 0$. The other is to map each value of X uniformly across all values of T; this leads to $H(T) = H(T \mid X) = \log |T|$ and $I(X,T) = 0$. In our initial studies, the IB consistently took the latter approach.[10] Since the DIB cost function favors the former approach (and indeed the DIB solution follows this approach), the IB consistently performs poorly by the DIB's standards. This difference is especially apparent at small β, where the compression term matters most, and as β increases, the DIB and IB solutions converge in the DIB plane.

To encourage the IB into more DIB-like behavior, we next altered our initialization scheme of $q(t \mid x)$. Originally, we used a normalized vector of uniform random numbers for each x. Next, we tried a series of delta-like functions, for which $q(t \mid x) = p_0$ for all x and one t, and the rest of the entries were uniform with a small amount of noise to break symmetry. The intended effect was to start the IB closer to solutions in which all data points were mapped to a single cluster. Results are shown in figure 3. While the different initialization schemes didn't change behavior in

[9] An even more efficient setting might be to set the cardinality of T based on the entropy of X, say $|T| = \mathrm{ceiling}(\exp(H(X)))$, but we didn't experiment with this.

[10] Intuitively, this approach is "more random" and is therefore easier to stumble upon during optimization.

the IB plane, we can see a gradual shift of the IB towards DIB-like behavior in the DIB plane as $p_0 \to 1$, i.e. the initialization scheme approaches a true delta. However, the IB still fails to reach the level of performance of the DIB, especially for large β, where the effect of the initialization washes out completely.

To summarize, the IB and DIB perform similarly by the IB standards, but the DIB tends to outperform the IB dramatically by the DIB's standards. Careful initialization of the IB can make up some of the difference, but not all.

It is also worth noting that, across all the datasets we tested, the DIB took 1-2 orders of magnitude fewer steps and time to converge, as illustrated in figure 4. About half of IB fits took at least an hour, while nearly a quarter took at least five hours. Contrast this with about half of DIB fits taking only five minutes, and more than 80% finishing within ten minutes. Put another way, about half of all DIB fits finished ten times faster than their IB counterpart, while about a quarter finished fifty times faster.

Note that the computational advantage of the DIB over the IB may vary by dataset and stopping criteria. In our case, we defined convergence for both algorithms as a change in cost function of less than 10^8 from one step to the next.

5 RELATED WORK

The DIB is not the first hard clustering version of IB.[11] Indeed, the agglomerative information bottleneck (AIB) [Slonim & Tishby (1999)] also produces hard clustering and was introduced soon after the IB. Thus, it is important to distinguish between the two approaches. AIB is a bottom-up, greedy method which starts with all data points belonging to their own clusters and iteratively merges clusters in a way which maximizes the gain in relevant information. It was explicitly designed to produce a hard clustering. DIB is a top-down method derived from a cost function that was not designed to produce a hard clustering. Our starting point was to alter the IB cost function to match the source coding notion of compression. The emergence of hard clustering in DIB is itself a result. Thus, while AIB does provide a hard clustering version of IB, DIB contributes the following in addition: 1) Our study emphasizes why a stochastic encoder is optimal for IB, namely due to the noise entropy term. The optimality of a stochastic encoder has been, for many, neither obvious nor necessarily desirable. 2) Our study provides a principled, top-down derivation of a hard clustering version of IB, based upon an intuitive change to the cost function. 3) Our non-trivial derivation also provides a cost-function and solution which interpolates between DIB and IB, by adding back the noise entropy continuously, i.e. with $0 < \alpha < 1$. This interpolation may be viewed as adding a regularization term to DIB. We are in fact currently exploring whether this type of regularization may be useful in dealing with finitely sampled data. Another interpretation of the cost function with intermediate α is as a penalty on *both* the mutual information between X and T as well as the entropy of the compression, $H(T)$. 4) It is likely that DIB offers a computational advantage to AIB. In the AIB paper, the authors say, "The main disadvantage of this method is computational, since it starts from the limit of a cluster per each member of the set X." In our experiments, we find that DIB is much more efficient than IB. Therefore, we expect that DIB will offer a considerable advantage in efficiency to AIB. However, we have not yet tested this.

The original IB also provides a deterministic encoding upon taking the limit $\beta \to \infty$ that corresponds to the causal-state partition of histories [Still et al (2010)]. However, this is the limit of no compression, whereas our approach allows for an entire family of deterministic encoders with varying degrees of compression.

[11] In fact, even the IB itself produces a hard clustering in the large β limit. However, it trivially assigns all data points to their own clusters.

6 DISCUSSION

Here we have introduced the deterministic information bottleneck (DIB) as an alternative to the information bottleneck (IB) for compression and clustering. We have argued that the DIB cost function better embodies the goal of lossy compression of relevant information, and shown that it leads to a non-trivial deterministic version of the IB. We have compared the DIB and IB solutions on synthetic data and found that, in our experiments, the DIB performs nearly identically to the IB in terms of the IB cost function, but far superior in terms of its own cost function. We also noted that the DIB achieved this performance at a computational efficiency 1-2 orders of magnitude better than the IB.

Of course, in addition to the studies with synthetic data here, it is important to compare the DIB and IB on real world datasets as well to see whether the DIB's apparent advantages hold. The linearity of the IB and DIB curves displayed above are indeed a signature of relatively simple data with not particularly complicated tradeoffs between compression and relevance.

One particular application of interest is maximally informative clustering. Previous work has, for example, offered a principled way of choosing the number of clusters based on the finiteness of the data [Still & Bialek (2004)]. Similarly interesting results may exist for the DIB, as well as relationships to other popular clustering algorithms such as k-means. More generally, there are learning theory results showing generalization bounds on IB for which an analog on DIB would be interesting as well [Shamir et al (2010)].

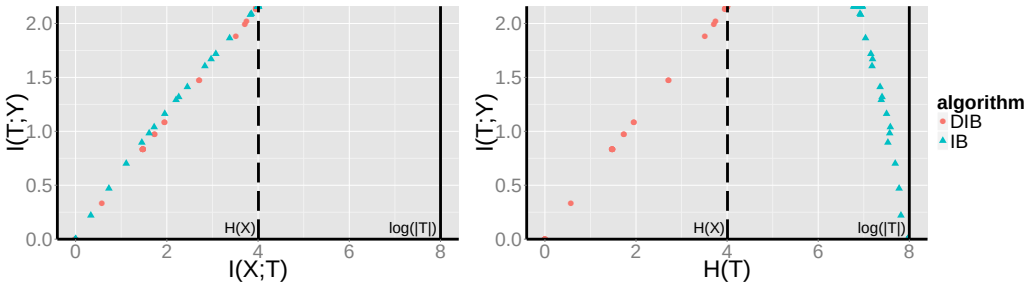

Figure 2: **Example IB and DIB solutions.** *Left*: IB plane. *Right*: DIB plane. Upper limit of the y-axes is $I(X, Y)$, since this is the maximal possible value of $I(T; Y)$. Solid vertical line marks $\log(|T|)$, since this is the maximal possible value of $H(T)$ and $I(X, T)$ (the latter being true since $I(X, T)$ is bounded above by both $H(T)$ and $H(X)$, and $|T| < |X|$). The dashed vertical line marks $H(X)$, which is both an upper bound for $I(X, T)$ and a natural comparison for $H(T)$ (since to place each data point in its own cluster, we need $H(T) = H(X)$). Here, $|X| = |Y| = 1024$ and $|T| = 256$.

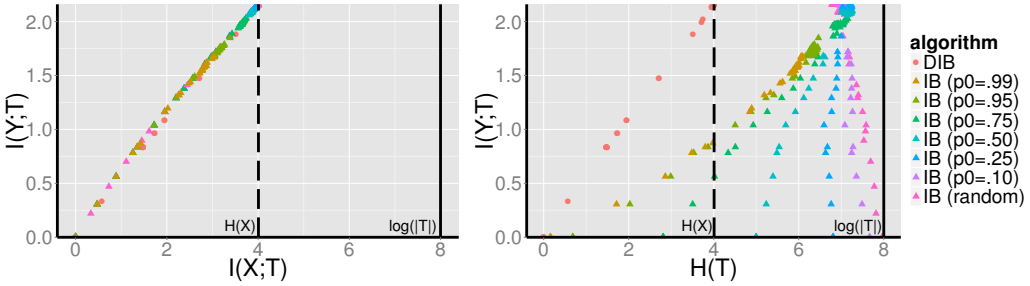

Figure 3: **Example IB and DIB solutions across different IB initializations.** Details identical to figure 2, except colors represent different initializations for the IB, as described in the text. "IB (random)" denotes the original initialization scheme of a normalized vector of uniform random numbers.

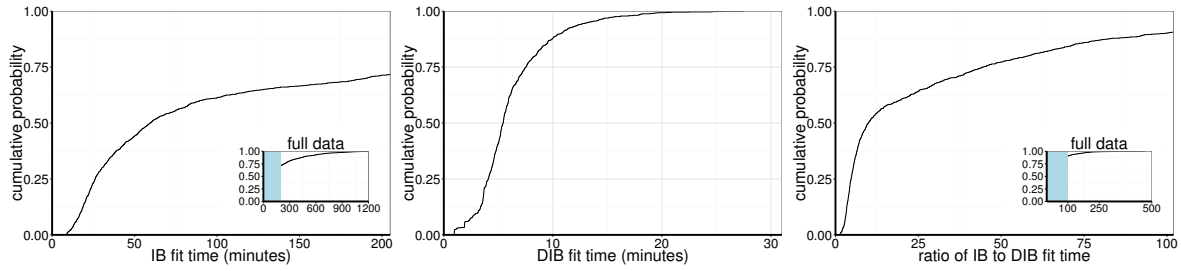

Figure 4: **Fit times for IB and DIB, as well as their ratios.** *Left*: cumulative distribution of IB fit times. Data shown here are for the original initialization of IB, though the delta-like initializations lead to nearly identical results. Mean fit time was 171 minutes. *Center*: cumulative distribution of DIB fit times. Mean fit time was 6 minutes. *Right*: cumultative distribution of ratios of IB to DIB fit times. Ratios are for the b^{th} value of β for DIB and the b^{th} value of β for IB, though those values of β are not necessarily the same. Both algorithms were fit to the same data. The IB fits are those resulting from the original random intialization, though the delta-like initializations lead to nearly identical results.

Another potential area of application is modeling the extraction of predictive information in the brain (which is one particular example in a long line of work on the exploitation of environmental statistics by the brain [Barlow (1981), Barlow (2001), Barlow (2001), Atick & Redlich (1992), Olshausen & Field (1996), Olshausen & Field (1997), Simoncelli & Olshausen (2001), Olshausen & Field (2004)]). There, X would be the stimulus at time t, Y the stimulus a short time in the future $t+\tau$, and T the activity of a population of sensory neurons. One could even consider neurons deeper in the brain by allowing X and Y to correspond not to an external stimulus, but to the activity of upstream neurons. An analysis of this nature using retinal data was recently performed with the IB [Palmer et al (2015)]. It would be interesting to see if the same data corresponds better to the behavior of the DIB, particularly in the DIB plane where the IB and DIB differ dramatically.

7 APPENDIX: DERIVATION OF GENERALIZED IB SOLUTION

Given $p(x, y)$ and subject to the Markov constraint $T \leftrightarrow X \leftrightarrow Y$, the generalized IB problem is:

$$\min_{q(t|x)} L[q(t \mid x)] = H(T) - \alpha H(T \mid X) - \beta I(T;Y) - \sum_{x,t} \lambda(x)\, q(t \mid x)\,, \tag{33}$$

where we have now included the Lagrange multiplier term (which enforces normalization of $q(t \mid x)$) explicitly. The Markov constraint implies the following factorizations:

$$q(t \mid y) = \sum_x q(t \mid x)\, p(x \mid y) \tag{34}$$

$$q(t) = \sum_x q(t \mid x)\, p(x)\,, \tag{35}$$

which give us the following useful derivatives:

$$\frac{\delta q(t \mid y)}{\delta q(t \mid x)} = p(x \mid y) \tag{36}$$

$$\frac{\delta q(t)}{\delta q(t \mid x)} = p(x)\,. \tag{37}$$

Now taking the derivative of the cost function with respect to the encoding distribution, we get:

$$\begin{aligned}
\frac{\delta L}{\delta q(t \mid x)} &= -\frac{\delta}{\delta q(t \mid x)} \sum_t q(t) \log q(t) && (38)\\
&\quad - \frac{\delta}{\delta q(t \mid x)} \sum_{x,t} \lambda(x)\, q(t \mid x) \\
&\quad + \alpha \frac{\delta}{\delta q(t \mid x)} \sum_{x,t} q(t \mid x)\, p(x) \log q(t \mid x) \\
&\quad - \beta \frac{\delta}{\delta q(t \mid x)} \sum_{y,t} q(t \mid y)\, p(y) \log \left[\frac{q(t \mid y)}{q(t)}\right] \\
&= -\log q(t)\, \frac{\delta q(t)}{\delta q(t \mid x)} - q(t)\, \frac{\delta \log q(t)}{\delta q(t \mid x)} && (39)\\
&\quad - \lambda(x)\, \frac{\delta q(t \mid x)}{\delta q(t \mid x)} \\
&\quad + \alpha \left[p(x) \log q(t \mid x)\, \frac{\delta q(t \mid x)}{\delta q(t \mid x)}\right. \\
&\quad \left. + q(t \mid x)\, p(x)\, \frac{\delta \log q(t \mid x)}{\delta q(t \mid x)}\right] \\
&\quad - \beta \sum_y \left[p(y) \log \left[\frac{q(t \mid y)}{q(t)}\right] \frac{\delta q(t \mid y)}{\delta q(t \mid x)}\right] \\
&\quad + \beta \sum_y \left[q(t \mid y)\, p(y)\, \frac{\delta \log q(t \mid y)}{\delta q(t \mid x)}\right. \\
&\quad \left. + q(t \mid y)\, p(y)\, \frac{\delta \log q(t)}{\delta q(t \mid x)}\right] \\
&= -p(x) \log q(t) - p(x) - \lambda(x) && (40)\\
&\quad + \alpha \left[p(x) \log q(t \mid x) + p(x)\right] \\
&\quad - \beta \sum_y \left[p(y) \log \left[\frac{q(t \mid y)}{q(t)}\right] p(x \mid y)\right. \\
&\quad \left. + p(y)\, p(x \mid y) - q(t \mid y)\, p(y)\, \frac{p(x)}{q(t)}\right] \\
&= -p(x) \log q(t) - p(x) - \lambda(x) && (41)\\
&\quad + \alpha \left[p(x) \log q(t \mid x) + p(x)\right] \\
&\quad - \beta p(x) \left[\sum_y p(y \mid x) \log \left[\frac{q(t \mid y)}{q(t)}\right]\right. \\
&\quad \left. + \sum_y p(y \mid x) - \sum_y q(y \mid t)\right] \\
&= p(x) \left[-1 - \log q(t) - \frac{\lambda(x)}{p(x)}\right. && (42)\\
&\quad + \alpha \log q(t \mid x) + \alpha \\
&\quad \left. - \beta \left[\sum_y p(y \mid x) \log \left[\frac{q(t \mid y)}{q(t)}\right]\right]\right]\,.
\end{aligned}$$

Setting this to zero implies that:

$$\alpha \log q(t \mid x) = 1 - \alpha + \log q(t) + \frac{\lambda(x)}{p(x)} + \beta \left[\sum_y p(y \mid x) \log \left[\frac{q(t \mid y)}{q(t)} \right] \right]. \tag{43}$$

We want to rewrite the β term as a KL divergence. First, we will need that $\log\left[\frac{q(t|y)}{q(t)}\right] = \log\left[\frac{q(t,y)}{q(t)p(y)}\right] = \log\left[\frac{q(y|t)}{p(y)}\right]$. Second, we will add and subtract $\beta \sum_y p(y \mid x) \log\left[\frac{p(y|x)}{p(y)}\right]$. This gives us:

$$\alpha \log q(t \mid x) = 1 - \alpha + \log q(t) + \frac{\lambda(x)}{p(x)} + \beta \sum_y p(y \mid x) \log \left[\frac{p(y \mid x)}{p(y)} \right] - \beta \left[\sum_y p(y \mid x) \log \left[\frac{p(y \mid x)}{q(y \mid t)} \right] \right]. \tag{44}$$

The second β term is now just $D_{\mathrm{KL}}[p(y \mid x) \mid q(y \mid t)]$. This leaves us with the equation:

$$\log q(t \mid x) = z(x, \alpha, \beta) + \frac{1}{\alpha} \log q(t) - \frac{\beta}{\alpha} D_{\mathrm{KL}}[p(y \mid x) \mid q(y \mid t)], \tag{45}$$

where we have divided both sides by α and absorbed all the terms that don't depend on t into the factor:

$$z(x, \alpha, \beta) \equiv \frac{1}{\alpha} - 1 + \frac{\lambda(x)}{\alpha p(x)} + \frac{\beta}{\alpha} \sum_y p(y \mid x) \log \left[\frac{p(y \mid x)}{p(y)} \right]. \tag{46}$$

Exponentiating both sides to solve for $q(t \mid x)$, we get:

$$d(x, t) \equiv D_{\mathrm{KL}}[p(y \mid x) \mid q(y \mid t)] \tag{47}$$

$$\ell_\beta(x, t) \equiv \log q(t) - \beta d(x, t) \tag{48}$$

$$q(t \mid x) = \frac{1}{Z} \exp\left[\frac{1}{\alpha} \ell_\beta(x, t) \right] \tag{49}$$

where:

$$Z(x, \alpha, \beta) \equiv \exp[-z] \tag{50}$$

is just a normalization factor. Now that we're done with the general derivation, let's add a subscript to the solution to distinguish it from the special cases of the IB and DIB.

$$q_\alpha(t \mid x) = \frac{1}{Z} \exp\left[\frac{1}{\alpha} \ell_\beta(x, t) \right]. \tag{51}$$

The IB solution is then:

$$q_{\mathrm{IB}}(t \mid x) = q_{\alpha=1}(t \mid x) \tag{52}$$

$$= \frac{q(t)}{Z} \exp[-\beta d(x, t)], \tag{53}$$

while the DIB solution is:

$$q_{\mathrm{DIB}}(t \mid x) = \lim_{\alpha \to 0} q_\alpha(t \mid x) \tag{54}$$

$$= \delta(t - t^*(x)), \tag{55}$$

with:

$$t^*(x) = \operatorname*{argmax}_t \ell_\beta(x, t). \tag{56}$$

Acknowledgements

The authors would like to thank Richard Turner, Bill Bialek, Stephanie Palmer, and Gordon Berman for helpful conversations, and the Hertz Foundation, DOE CSGF (DJ Strouse), and NIH grant K25 GM098875-06 (David Schwab) for funding.

References

Atick, J.J. & Redlich, A.N. (1992). *What Does the Retina Know about Natural Scenes?* Neur Comp, 4, 196-210. 6

Bach, F.R. & Jordan, M.I. (2005). *A probabilistic interpretation of canonical correlation analysis*. Tech Report. 2

Barlow, H. (1981). *Critical Limiting Factors in the Design of the Eye and Visual Cortex*. Proc of the Royal Society B: Biological Sciences, 212(1186), 1–34. 3, 6

Barlow, H. (2001). *Redundancy reduction revisited*. Network: Comp in Neural Systems, 12(3), 241–253. 3, 6

Barlow, H. (2001). *The exploitation of regularities in the environment by the brain*. BBS 24, 602–607. 3, 6

Bekkerman, R., El-Yaniv, R., Tishby, N., & Winter, Y. (2001). *On feature distributional clustering for text categorization*. SIGIR. 2

Bekkerman, R., El-Yaniv, R., & Tishby, N. (2003). *Distributional word clusters vs. words for text categorization.* JMLR, 3, 1183-1208. 2

Chechik, G., Globerson, A., Tishby, N., & Weiss, Y. (2005). *Information Bottleneck for Gaussian Variables*. NIPS. 1, 2

Cover, T.M. & Thomas, J.A. (2006). *Elements of Information Theory*. John Wiley & Sons, Inc. 1

Creutzig, F., Globerson, A., & Tishby, N. (2009). *Past-future information bottleneck in dynamical systems*. Physical Review E, 79(4). 2

Hecht, R.M. & Tishby, N. (2005). *Extraction of relevant speech features using the information bottleneck method*. InterSpeech. 2

Hecht, R.M. & Tishby, N. (2007). *Extraction of relevant Information using the Information Bottleneck Method for Speaker Recognition*. InterSpeech. 2

Hecht, R.M., Noor, E., & Tishby, N. (2009). *Speaker recognition by Gaussian information bottleneck*. InterSpeech. 2

Kandel, E.R., Schwartz, J.H., Jessell, T.M., Siegelbaum, S.A., & Hudspeth, A.J. (2013). *Principles of Neural Science*. New York: McGraw-Hill. 1

Mackay, D. (2002). *Information Theory, Inference, & Learning Algorithms*. Cambridge University Press. 1

Olshausen, B.A. & Field, D.J. (1996). *Emergence of simple-cell receptive field properties by learning a sparse code for natural images*. Nature, 381(6583), 607–609. 6

Olshausen, B.A. & Field, D.J. (1997). *Sparse coding with an overcomplete basis set: A strategy employed by V1?* Vision Research. 6

Olshausen, B.A. & Field, D.J. (2004). *Sparse coding of sensory inputs*. Curr Op in Neurobio, 14(4), 481–487. 6

Palmer, S.E., Marre, O., Berry, M.J., & Bialek, W. (2015). *Predictive information in a sensory population*. PNAS, 112(22), 6908–6913. 1, 2, 6

Schneidman, E., Slonim, N., Tishby, N., deRuyter van Steveninck, R., & Bialek, W. (2002). *Analyzing neural codes using the information bottleneck method*. NIPS. 2

Shamir, O., Sabato, S., & Tishby, N. (2010). *Learning and Generalization with the Information Bottleneck*. Theoretical Comp Sci, Vol 411, Issu 29-30, Pgs 2696-2711. 6

Simoncelli, E. P., & Olshausen, B. A. (2001). *Natural image statistics and neural representation*. Annual Review of Neuroscience. 6

Slonim, N. & Tishby, N. (1999). *Agglomerative information bottleneck*. NIPS. 5

Slonim, N. & Tishby, N. (2000). *Document clustering using word clusters via the information bottleneck method*. SIGIR. 2

Slonim, N. & Tishby, N. (2001). *The Power of Word Clusters for Text Classification*. ECIR, 1–12. 2

Slonim, N. & Weiss, Y. (2002). *Maximum likelihood and the information bottleneck*. NIPS, 15, 335–342. 2

Still, S. & Bialek, W. (2004). *How many clusters? An information-theoretic perspective*. Neur Comp, 16(12), 2483–2506. 6

Still, S., Crutchfield, J.P., & Ellison, C.J. (2010). *Optimal causal inference: Estimating stored information and approximating causal architecture*. Chaos, 20(3), 037111. 5

Tishby, N., Pereira, F. & Bialek, W. (1999). *The Information Bottleneck Method*. Proc of The 37th Allerton Conf on Comm, Control, & Comp, Univ. of Illinois. 1, 2

Tishby, N. & Zaslavsky, N. (2015). *Deep Learning and the Information Bottleneck Principle*. arXiv.org. 2

Turner, R.E. & Sahani, M. (2007). *A maximum-likelihood interpretation for slow feature analysis*. Neur Comp, 19(4), 1022–1038. 2

Wallace, G.K. (1991). *The JPEG Still Picture Compression Standard*. Comm ACM, vol. 34, pp. 30-44.

1