The information bottleneck and geometric clustering

DJ Strouse* & David J. Schwab†

December 29, 2017

Abstract

The information bottleneck (IB) approach to clustering takes a joint distribution \( P(X, Y) \) and maps the data \( X \) to cluster labels \( T \) which retain maximal information about \( Y \) (Tishby et al., 1999). This objective results in an algorithm that clusters data points based upon the similarity of their conditional distributions \( P(Y | X) \). This is in contrast to classic “geometric clustering” algorithms such as \( k \)-means and gaussian mixture models (GMMs) which take a set of observed data points \( \{x_i\}_{i=1:N} \) and cluster them based upon their geometric (typically Euclidean) distance from one another. Here, we show how to use the deterministic information bottleneck (DIB) (Strouse and Schwab, 2017), a variant of IB, to perform geometric clustering, by choosing cluster labels that preserve information about data point location on a smoothed dataset. We also introduce a novel intuitive method to choose the number of clusters, via kinks in the information curve. We apply this approach to a variety of simple clustering problems, showing that DIB with our model selection procedure recovers the generative cluster labels. We also show that, for one simple case, DIB interpolates between the cluster boundaries of GMMs and \( k \)-means in the large data limit. Thus, our IB approach to clustering also provides an information-theoretic perspective on these classic algorithms.

1 Introduction

Unsupervised learning is a crucial component of building intelligent systems (LeCun, 2016), since such systems need to be able to leverage experience to improve performance even in the absence of feedback. One aspect of doing so is discovering discrete structure in data, a problem known as clustering (MacKay, 2002). In the typical setup, one is handed a set of data points \( \{x_i\}_{i=1}^N \), and asked to return a mapping from data point label \( i \) to a finite set of cluster labels \( c \). The most basic approaches include \( k \)-means and gaussian mixture models (GMMs). GMMs cluster data based on maximum likelihood fitting of a probabilistic generative model. \( k \)-means can either be thought of as directly clustering data based on geometric (often Euclidean) distances between data points, or as a special case of GMMs with the assumptions of evenly sampled, symmetric, equal variance components.

*Department of Physics, Princeton University
†Initiative for the Theoretical Sciences, CUNY Graduate Center
The information bottleneck (IB) is an information-theoretic approach to clustering data \( X \) that optimizes cluster labels \( T \) to preserve information about a third “target variable” of interest \( Y \). The resulting (soft) clustering groups data points based on the similarity in their conditional distributions over the target variable through the KL divergence, \( \text{KL}[p(y \mid x_i) \mid p(y \mid x_j)] \). An IB clustering problem is fully specified by the joint distribution \( P(X,Y) \) and the tradeoff parameter \( \beta \) quantifying the relative preference for fewer clusters and more informative ones.

At first glance, it is not obvious how to use this approach to cluster geometric data, where the input is a set of data point locations \( \{x_i\}_{i=1}^{N} \). For example, what is the target variable \( Y \) that our clusters should retain information about? What should \( P(X,Y) \) be? And how should one choose the tradeoff parameter \( \beta \)?

Still et al. (2003) were the first to attempt to do geometric clustering with IB, and claimed an equivalence (in the large data limit) between IB and \( k \)-means. Unfortunately, while much of their approach is correct, it contained some fundamental errors that nullify the main results. In the next section, we describe those errors and how to correct them. Essentially, their approach did not properly translate geometric information into a form that could be used correctly by an information-theoretic algorithm.

In addition to fixing this issue, we also choose to use a recently introduced variant of the information bottleneck called the deterministic information bottleneck (DIB) (Strouse and Schwab, 2017). We make this choice due to the different way in which IB and DIB use the number of clusters provided to them. IB is known to use all of the clusters it has access to, and thus clustering with IB requires a search both over the number of clusters \( N_c \) as well as the the parsimony-informativeness tradeoff parameter \( \beta \) (Slonim et al., 2005). DIB on the other hand has a built-in preference for using as few clusters as it can, and thus only requires a parameter search over \( \beta \). Moreover, DIB’s ability to select the number of clusters to use for a given \( \beta \) leads to an intuitive model selection heuristic based on the robustness of a clustering result across \( \beta \) that we show can recover the generative number of clusters in many cases.

In the next section, we more formally define the geometric clustering problem, the IB approach of Still et al. (2003), and our own DIB approach. In section 3, we show that our DIB approach to geometric clustering behaves intuitively and is able to recover the generative number of clusters with only a single free parameter (the data smoothing scale \( s \)). In section 4, we discuss the relationship between our approach and GMMs and \( k \)-means, proving that at least in one simple case, DIB interpolates between GMM and \( k \)-means cluster boundaries by varying the data smoothing scale \( s \). Our approach thus provides a novel information-theoretic approach to geometric clustering, as well as an information-theoretic perspective on these classic clustering methods.

## 2 Geometric clustering with the (deterministic) information bottleneck

In a geometric clustering problem, we are given a set of \( N \) observed data points \( \{x_i\}_{i=1}^{N} \) and asked to provide a weighting \( q(c \mid i) \) that categorizes data points into (possibly multiple) clusters such that data points “near” one another are in the same cluster. The definition of
“near” varies by algorithm: for k-means, for example, points in a cluster are closer to their own cluster mean than to any other cluster mean.

In an information bottleneck (IB) problem, we are given a joint distribution \( P(X,Y) \) and asked to provide a mapping \( q(t \mid x) \) such that \( T \) contains the “relevant” information in \( X \) for predicting \( Y \). This goal is embodied by the information-theoretic optimization problem:

\[
q^*_\text{IB}(t \mid x) = \arg\min_{q(t \mid x)} I(X,T) - \beta I(T,Y), \tag{1}
\]

subject to the Markov constraint \( T \leftrightarrow X \leftrightarrow Y \). \( \beta \) is a free parameter that allows for setting the desired balance between the compression encouraged by the first term and the relevance encouraged by the second; at small values, we throw away most of \( X \) in favor of a succinct representation for \( T \), while for large values of \( \beta \), we retain nearly all the information that \( X \) has about \( Y \).

This approach of squeezing information through a latent variable bottleneck might remind some readers of a variational autoencoder (VAE) (Kingma and Welling, 2013), and indeed IB has a close relationship with VAEs. As pointed out by Alemi et al. (2016), a variational version of IB can essentially be seen as the supervised generalization of a VAE, which is typically an unsupervised algorithm.

We are interested in performing geometric clustering with the information bottleneck. For the purposes of this paper, we will focus on a recent alternative formulation of the IB, called the deterministic information bottleneck (DIB) (Strouse and Schwab, 2017). We do this because the DIB’s cost function more directly encourages the use of as few clusters as possible, so initialized with \( n^{\text{max}}_c \) clusters, it will typically converge to a solution with far fewer. Thus, it has a form of model selection built in that will prove useful for geometric clustering (Strouse and Schwab, 2017). IB, on the other hand, will tend to use all \( n^{\text{max}}_c \) clusters, and thus requires an additional search over this parameter (Slonim et al., 2005). DIB also differs from IB in that it leads to a hard clustering instead of a soft clustering.

Formally, the DIB setup is identical to that of IB except that the mutual information term \( I(X;T) \) in the cost functional is replaced with the entropy \( H(T) \):

\[
q^*_\text{DIB}(t \mid x) = \arg\min_{q(t \mid x)} H(T) - \beta I(T,Y). \tag{2}
\]

This change to the cost functional leads to a hard clustering with the form (Strouse and Schwab, 2017):

\[
q^*_\text{DIB}(t \mid x) = \delta(t - t^*(x)) \tag{3}
\]

\[
t^* = \arg\max_t \log q(t) - \beta d(x,t) \tag{4}
\]

\[
d(x,t) \equiv \text{KL}[p(y \mid x) \mid q(y \mid t)] \tag{5}
\]

\[
q(t) = \sum_x q(t \mid x)p(x) \tag{6}
\]

\[
q(y \mid t) = \frac{1}{q(t)}\sum_x q(t \mid x)p(x)p(y \mid x), \tag{7}
\]
where the above equations are to be iterated to convergence from some initialization. The IB solution (Tishby et al., 1999) simply replaces the first two equations with:

$$q_{IB}^*(t \mid x) = \frac{q(t)}{Z(x, \beta)} e^{-\beta d(x,t)}, \quad (8)$$

which can be seen as replacing the argmax in DIB with an exponential and a soft max.

The (D)IB is referred to as a “distributional clustering” algorithm (Slonim and Tishby, 2001) due to the KL divergence term $d(x, t) = KL[p(y \mid x) \mid q(y \mid t)]$, which can be seen as measuring how similar the data point conditional distribution $p(y \mid x)$ is to the cluster conditional, or mixture of data point conditionals, $q(y \mid t) = \sum_x q(x \mid t) p(y \mid x)$. That is, a candidate point $x'$ will be assigned to a cluster based upon how similar its conditional $p(y \mid x')$ is to the conditionals $p(y \mid x)$ for the data points $x$ that make up that cluster. Thus, both DIB and IB cluster data points based upon the conditionals $p(y \mid x)$.

To apply (D)IB to a geometric clustering problem, we must choose how to map the geometric clustering dataset $\{x_i\}_{i=1:N}$ to an appropriate IB dataset $P(X,Y)$. First, what should $X$ and $Y$ be? Since $X$ is the data being clustered by IB, we’ll choose that to be the data point index $i$. As for the target variable $Y$ that we wish to maintain information about, it seems reasonable to choose the data point location $x$ (though we will discuss alternative choices later). Thus, we want to cluster data indices $i$ into cluster indices $c$ in a way that maintains as much possible info about the location $x$ as possible (Still et al., 2003).

Now, how should we choose the joint distribution $p(i, x) = p(x \mid i) p(i)$? At first glance, one might choose $p(x \mid i) = \delta_{x_i}$, since data point $i$ was observed at location $x_i$. The reason not to do this lies with the fact that (D)IB is a distributional clustering algorithm, as discussed two paragraphs above. Data points are compared to one another through their conditionals $p(x \mid i)$, and with the choice of a delta function, there will be no overlap unless two data points are on top of one another. That is, choosing $p(x \mid i) = \delta_{x_i}$ leads to a KL divergence that is either infinite for data points at different locations, or zero for data points that lie exactly on top of one another, i.e. $KL[p(x \mid i) \mid p(x \mid j)] = \delta_{x_i,x_j}$. Trivially, the resulting clustering would assign each data point to its own cluster, grouping only data points that are identical. Put another way, all relational information in an IB problem lies in the joint distribution $P(X,Y)$. If one wants to perform geometric clustering with an IB approach, then geometric information must somehow be injected into that joint distribution, and a series of delta functions does not do that. A previous attempt at linking IB and $k$-means made this mistake (Still et al., 2003). Subsequent algebraic errors were tantamount to incorrectly introducing geometric information into IB, precisely in the way that such geometric information appears in $k$-means, and resulting in an algorithm that is not IB. We describe these errors in more detail in an appendix (section 6).

Based on the problems identified with using delta functions, a better choice for the conditionals is something spatially extended, such as:

$$p(x \mid i) \sim \exp \left[ -\frac{1}{2s^2} d(x, x_i) \right], \quad (9)$$

where $s$ sets the geometric scale or units of distance, and $d$ is a distance metric, such as the Euclidean distance $d(x, x_i) = ||x - x_i||^2$. If we indeed use the Euclidean distance, then...
Figure 1: Illustration of data smoothing procedure. Example dataset with one symmetric and one skew cluster. Top row: scatterplot of data points with smoothed probability distribution overlaid. Bottom row: heat map of the joint distribution $P(i, x)$ that is fed into DIB. The two spatial dimensions in the top row are binned and concatenated into a single dimension (on the horizontal axis) in the bottom row, which is the source of the “striations.”

$p(x \mid i)$ will be (symmetric) gaussian (with variance $s^2$), and this corresponds to gaussian smoothing our data. In any case, the obvious choice for the marginal is $p(i) = \frac{1}{N}$, where $N$ is the number of data points, unless one has a reason a priori to favor certain data points over others. These choices for $p(i)$ and $p(x \mid i)$ determine completely our dataset $p(i, x) = p(x \mid i) p(i)$. Figure 1 contains an illustration of this data smoothing procedure. We will explore the effect of the choice of smoothing scale $s$ throughout this paper.

With the above choices, we have a fully specified DIB formulation of a geometric clustering problem. Using our above notational choices, the equations for the $n^{th}$ step in the iterative DIB solution is (Strouse and Schwab, 2017):

\[ c^{(n)}(i) = \arg\max_c \log q^{(n-1)}(c) - \beta d_n(i, c) \]  
\[ d_n(i, c) \equiv \text{KL}[p(x \mid i) \mid q^{(n)}(x \mid c)] \]  
\[ q^{(n)}(c \mid i) = \delta(c - c^{(n)}(i)) \]  
\[ q^{(n)}(c) = \frac{n_c^{(n)}}{N} \]  
\[ q^{(n)}(x \mid c) = \sum_i q^{(n)}(i \mid c) p(x \mid i) \]  
\[ = \frac{1}{n_c^{(n)}} \sum_{i: c^{(n)}(i) = c} p(x \mid i) \],

where $n_c^{(n)}$ the number of data points assigned to cluster $c$ at step $n$, $n_c^{(n)} \equiv Nq^{(n)}(c) =$
\[ \sum_i q^{(n)}(c \mid i) = |i : c^{(n)}(i) = c|. \]

Note that this solution contains \( \beta \) as a free parameter. As discussed above, it allows us to set our preference between solutions with fewer clusters and those that retain more spatial information. It is common in the IB literature to run the algorithm for multiple values of \( \beta \) and to plot the collection of solutions in the “information plane” with the relevance term \( I(Y; T) \) on the \( y \)-axis and the compression term \( I(X; T) \) on the \( x \)-axis (Palmer et al., 2015; Creutzig et al., 2009; Chechik et al., 2005; Slonim et al., 2005; Still and Bialek, 2004; Tishby and Zaslavsky, 2015; Rubin et al., 2016; Strouse and Schwab, 2017; Ravid and Tishby, 2017). The natural such plane for the DIB is with the relevance term \( I(Y; T) \) on the \( y \)-axis and its compression term \( H(T) \) on the \( x \)-axis (Strouse and Schwab, 2017). The curve drawn out by (D)IB solutions in the information plane can be viewed as a Pareto-optimal boundary of how much relevant information can be extracted about \( Y \) given a fixed amount of information about \( X \) (IB) or representational capacity by \( T \) (DIB) (Strouse and Schwab, 2017). Solutions lying below this curve are of course suboptimal, but a priori, the (D)IB formalism doesn’t tell us how to select a single solution from the family of solutions lying on the (D)IB boundary. Intuitively however, when faced with a boundary of Pareto-optimality, if we must pick one solution, its best to choose one at the “knee” of the curve. Quantitatively, the “knee” of the curve is the point where the curve has its maximum magnitude second derivative. In the most extreme case, the second derivative is infinite when there is a “kink” in the curve, and thus the largest kinks might correspond to solutions of particular interest. In our case, since the slope of the (D)IB curve at any given solution is \( \beta^{-1} \) (which can be read off from the cost functionals), kinks indicate solutions that are valid over a wide range of \( \beta \). So large kinks also correspond to robust solutions, in the sense that they optimize a wide range of (D)IB tradeoffs. Quantitatively, we can measure the size of a kink by the angle \( \theta \) of the discontinuity it causes in the slope of the curve; see figure 2 for details. We will show in the next section that searches for solutions with large \( \theta \) result in recovering the generative cluster labels for geometric data, including the correct number of clusters.

Note that this model selection procedure would not be possible if we had chosen to use IB instead of DIB. IB uses all the clusters available to it, regardless of the choice of \( \beta \). Thus, all solutions on the curve would have the same number of clusters anyway, so any knees or kinks cannot be used to select the number of clusters.
Figure 2: “Kinks” in DIB information curve as model selection. $\beta_{\text{min}}$ and $\beta_{\text{max}}$ are the smallest and largest $\beta$ at which the solution at the kink is valid. Thus, $\beta_{\text{min}}^{-1}$ and $\beta_{\text{max}}^{-1}$ are the slopes of upper and lower dotted lines. The “kink angle” is then $\theta = \frac{\pi}{2} - \arctan(\beta_{\text{min}}) - \arctan(\beta_{\text{max}}^{-1})$. It is a measure of how robust a solution is to the choice of $\beta$; thus high values of $\theta$ indicate solutions of particular interest.

3 Results: geometric clustering with DIB

We ran the DIB as described above on four geometric clustering datasets, varying the smoothing width $s$ (see eqn 9) and tradeoff parameter $\beta$, and measured for each solution the fraction of spatial information extracted $\tilde{I}(c;x) = \frac{I(c;x)}{I(i;x)}$ and the number of clusters used $n_c$, as well as the kink angle $\theta$. We iterated the DIB equations above just as in Strouse and Schwab (2017) with one difference. Iterating greedily from some initialization can lead to local minima (the DIB optimization problem is non-convex). To help overcome suboptimal solutions, upon convergence, we checked whether merging any two clusters would improve the value $L$ of the cost functional in eqn 2. If so, we chose the merging with the highest such reduction, and began the iterative equations again. We repeated this procedure until the algorithm converged and no merging reduced the value of $L$. We found that these “non-local” steps worked well in combination with the greedy “local” improvements of the DIB iterative equations. While not essential to the function of DIB, this improvement in performance produced cleaner information curves with less “noise” caused by convergence to local minima.

Results are shown in figure 3. Each large row represents a different dataset. The left column shows fractional spatial information $\tilde{I}(c;x)$ versus number of clusters used $n_c$, stacked by smoothing width $s$.

Note that $I(i;x)$ is an upper bound on $I(c;x)$ due to the data processing inequality, (Cover and Thomas, 2006) so $\tilde{I}(c;x)$ is indeed the fraction of potential geometric information extracted from the smoothed $P(i;x)$. Not all values of $n_c$ are present because while varying the implicit parameter $\beta$, DIB will not

\footnote{Note that $I(i;x)$ is an upper bound on $I(c;x)$ due to the data processing inequality, (Cover and Thomas, 2006) so $\tilde{I}(c;x)$ is indeed the fraction of potential geometric information extracted from the smoothed $P(i;x)$.}

\footnote{Note that this is not the same as the information plane curve from figure 2. While the $y$-axes are the same (up to the normalization), the $x$-axes are different.}
Figure 3: Results: model selection and clustering with DIB. Results for four datasets. Each row represents a different dataset. \textit{Left column:} fraction of spatial information extracted, $I(c; x) = \frac{f(c|x)}{f(x)}$, versus number of clusters used, $n_c$, across a variety of smoothing scales, $s$. \textit{Center column:} kink angle $\theta$ (of the $I(c; x)$ vs $H(c)$ curve) versus number of clusters used, $n_c$, across a variety of smoothing scales, $s$. \textit{Right column:} example resulting clusters.
necessarily “choose” to use all possible cluster numbers. For example, for small smoothing width \( s \), most points won’t have enough overlap in \( p(x | i) \) with their neighbors to support solutions with few clusters, and for large smoothing width \( s \), local spatial information is thrown out and only solutions with few clusters are possible. More interestingly, DIB may retain or drop solutions based on how well they match the structure of the data, as we will discuss for each dataset below. Additionally, solutions that match well the structure in the data (for example, ones with \( n_c \) matched to the generative parameters) tend to be especially robust to \( \beta \), that is they have a large kink angle \( \theta \). Thus, \( \theta \) can be used to perform model selection. For datasets with structure at multiple scales, the kink angle \( \theta \) will select different solutions for different values of the smoothing width \( s \). This allows us to investigate structure in a dataset at a particular scale of our choosing. We now turn to the individual datasets.

The first dataset (top large row) consists of 3 equally spaced, equally sampled symmetric gaussian clusters (see solutions in right column). We see that the 3-cluster solution stands out in several ways. First, it is robust to spatial scale \( s \). Second, the 3-cluster solution extract nearly all of the available spatial information; solutions with \( n_c \geq 4 \) extract little extra \( \tilde{I}(c; x) \). Third and perhaps most salient, the 3-cluster solution has by far the largest value of kink angle \( \theta \) across a wide range of smoothing scales. In the right column, we show examples of 3 and 4-cluster solutions. Note that while all 3-cluster solutions look exactly like this one, the 4-cluster solutions vary in how they chop one true cluster into two.

The second dataset (second row) consists of 3 more equally sampled symmetric gaussian clusters, but this time not equally spaced; two are much closer to one another than the third. This is a dataset with multiple scales present, thus we should expect that the number of clusters picked out by any model selection procedure, e.g. kink angle, should depend on the spatial scale of interest. Indeed, we see that to be true. The 3-cluster solution is present for all smoothing widths shown, but is only selected out as the best solution by kink angle for intermediate smoothing widths (\( s = 2 \)). For large smoothing widths (\( s = 8 \)), we see that the 2-cluster solution is chosen as best. For smoothing widths in between (\( s = 4 \)), the 2 and 3-cluster solutions are roughly equally valid. In terms of spatial information, the 2 and 3-cluster solutions are also prominent, with both transitions from \( n_c = 1 \rightarrow 2 \) and \( n_c = 2 \rightarrow 3 \) providing significant improvement in \( \tilde{I}(c; x) \) (but little improvement for more fine-grained clusterings).

The third dataset (third row) features even more multi-scale structure, with 5 symmetric, equally sampled gaussians, again with unequal spacing. Sensible solutions exist for \( n_c = 2−5 \), and this can be seen by the more gradual rise of the fractional spatial information \( \tilde{I}(c; x) \) with \( n_c \) in that regime. We also again see a transition in the model selection by \( \theta \) from the 5-cluster solution at small smoothing widths (\( s = 1, 2 \)) and the 2-cluster solution at larger smoothing widths (\( s = 8 \)), with intermediate \( n_c \) favoring those and intermediate solutions. Example clusters for \( n_c = 2−5 \) are shown at right.

Finally, we wanted to ensure that DIB and our model selection procedure would not hallucinate structure where there is none, so we applied it to a single gaussian blob, with the hope that no solution with \( n_c > 1 \) would stand out and prove robust to \( \beta \). As can be seen in the fourth row of figure 3, that is indeed true. No solution at any smoothing width had particularly high kink angle \( \theta \), and no solution remained at the “knee” of the \( \tilde{I}(c; x) \) versus \( n_c \) curve across a wide range of smoothing widths.

Overall, these results suggest that DIB on smoothed data is able to recover generative
geometric structure at multiple scales, using built-in model selection procedures based on identifying robust, spatially informative solutions.

4 Results: DIB vs GMMs & $k$-means

Here we show that in the limit of infinite data and small smoothing scale $s$, the behavior of (D)IB is intimately related to the hard cluster boundaries implied by GMMs. We assume we have one gaussian cluster centered at $\mu_1 = (0, 0)$ with covariance $\Sigma_1 = \text{diag}(\sigma_1, \sigma_2)$, and a second gaussian cluster centered at $\mu_2 = (L, 0)$ with covariance $\Sigma_2 = \text{diag}(\sigma, \sigma)$. If we have a mixture model with weights $w_1$ and $w_2$, then the hard maximum likelihood boundary between these two clusters in the $(x_1, x_2)$ plane is given by:

$$T_1 \equiv \frac{x_1^2}{2\sigma_1^2} + \frac{x_2^2}{2\sigma_2^2} + \log \sigma_1 \sigma_2$$

$$T_2 \equiv \frac{1}{2\sigma_2} \left( x_1^2 + x_2^2 + L^2 - 2Lx_1 \right)$$

$$\log w_1 - T_1 = \log w_2 - T_2.$$

On the other hand, the (D)IB algorithm would classify a new test point at location $(x_1, x_2)$ gaussian smoothed by $s$ based on the KL divergence between its smoothed distribution and the two cluster gaussians:

$$\text{KL}_1 = s^2 \left( \frac{\sigma_1^2 + \sigma_2^2}{2\sigma_1^2 \sigma_2^2} \right) + \frac{x_1^2}{\sigma_1^2} + \frac{x_2^2}{\sigma_2^2} - \frac{k}{2} + \log \frac{\sigma_1 \sigma_2}{s^2}$$  \hfill (16)

$$\text{KL}_2 = s^2 + \frac{1}{2\sigma_2^2} \left( (x_1 - L)^2 + x_2^2 \right) - \frac{k}{2} + \log \frac{\sigma^2}{s^2},$$  \hfill (17)

where $k = 2$ is the number of dimensions. The boundary implied by DIB is found by setting:

$$\log w_1 - \beta \text{KL}_1 = \log w_2 - \beta \text{KL}_2.$$  \hfill (18)

Notice that for small smoothing width $s \to 0$ and either $\beta = 1$ or evenly sampled clusters $w_1 = w_2$, this is identical to the hard boundary implied by the GMM. For $\beta > 1$, $w_1 \neq w_2$, and small smoothing width, we see that, compared with a GMM, DIB encourages capturing more information about spatial location at the expense of using clusters more equally. Put another way, the effect of the cluster prior term $\log \frac{w_1}{w_2}$ is reduced by pulling it closer to zero, i.e. replacing it with $\log \left( \frac{w_1}{w_2} \right)^{1/\beta}$. This provides an interesting information theoretic interpretation of GMMs and also shows the manner in which clustering with DIB is a generalization.

To see the effect of larger smoothing widths, we compared the numerically calculated DIB, GMM, and $k$-means cluster boundaries for the “true” assignments with $n_c = 2$ over a range of smoothing widths (see figure 4). The data consisted of 1000 points sampled equally ($w_1 = w_2$) from one isotropic and one skew gaussian as shown. We can see that for small smoothing widths, the DIB boundary indeed approaches that of the GMM. For larger
smoothing widths, the effect of the “shape” of the clusters is muted and the DIB boundary approaches \( k \)-means. Note that this is just one particular example however and DIB need not approach \( k \)-means in the large \( s \) limit in general.

![Figure 4: Cluster boundaries for different algorithms.](image)

**Figure 4:** Cluster boundaries for different algorithms. Colored lines show boundaries separating two clusters for 3 different algorithms: \( k \)-means, GMMs, and DIB with 3 different levels of smoothing. Dataset was 1000 points drawn equally from a single symmetric gaussian and a single skew gaussian. Black points show data.

5 Discussion

Here, we have shown how to use the formalism of the information bottleneck to perform geometric clustering. A previous paper (Still et al., 2003) claimed to contribute similarly, however for the reasons discussed in sections 2 and 6, their approach contained fundamental flaws. We amend and improve upon that paper in four ways. First, we show to fix the errors they made in their problem setup (with the data preparation). Second, we argue for using DIB over IB in this setting for its preference for using as few clusters as it can. Third, we introduce a novel form of model selection for the number of clusters based on discontinuities (or “kinks”) in the slope of the DIB curve, which indicate solutions that are robust across the DIB tradeoff parameter \( \beta \). We show that this information-based model selection criterion allows us to correctly recover generative structure in the data at multiple spatial scales. Finally, we compare the resulting clustering algorithm to \( k \)-means and gaussian mixture models (GMMs). We found that for large smoothing width, \( s \), the performance of the method seems to behave similarly to \( k \)-means. More interestingly, we found that for small smoothing width, the method behaves as a generalization of a GMM, with a tunable tradeoff between compression and fidelity of the representation.

We have introduced one way of doing geometric clustering with the information bottleneck, but we think it opens avenues for other ways as well. First, the uniform smoothing we perform above could be generalized in a number of ways to better exploit local geometry and better estimate the “true” generative distribution of the data. For example, one could do gaussian smoothing with mean centered on each data point but the covariance estimated by the sample covariance of neighboring data points around that mean. Indeed, our early experiments with this alternative suggest it may be useful for certain datasets. Second, while choosing spatial location as the relevant variable for DIB to preserve information about seems to be the obvious first choice to investigate, other options might prove interesting. For ex-
ample, preserving information about the identity of neighbors, if carefully formulated, might make fewer implicit assumptions about the shape of the generative distribution, and enable the extension of our approach to a wider range of datasets.

Scaling the approach introduced here to higher-dimensional datasets is non-trivial because the tabular representation used in the original IB (Tishby et al., 1999) and DIB (Strouse and Schwab, 2017) algorithms leads to an exponential scaling with the number of dimensions. Recently, however, Alemi et al. (2016) introduced a variational version of IB, in which one parameterizes the encoder $q(t \mid x)$ (and “decoder” $q(y \mid t)$) with a function approximator, e.g. a deep neural network. This has the advantage of allowing scaling to much larger datasets. Moreover, the choice of parameterization often implies a smoothness constraint on the data, relieving the problem encountered above of needing to smooth the data. It would be interesting to develop a variational version of DIB, which could then be used to perform information-theoretic clustering as we have done here, but on larger problems and perhaps with no need for data smoothing.

6 Appendix: errors in Still et al. (2003)

A previous attempt was made to draw a connection between IB and $k$-means (Still et al., 2003). Even before reviewing the algebraic errors that lead their result to break down, there are two intuitive reasons why such a claim is unlikely to be true. First, IB is a soft clustering algorithm, and $k$-means is a hard clustering algorithm. Second, the authors made the choice not to smooth the data and to set $p(x \mid i) = \delta_{xx_i}$. As discussed in section 2, (D)IB clusters data points based on these conditionals, and delta functions trivially only overlap when they are identical.

The primary algebraic mistake appears just after eqn 14, in the claim that $p_n(x \mid c) \propto p_{n-1}(x \mid c)^{1/\lambda}$. Combining the previous two claims in that proof, we obtain:

$$p_n(x \mid c) = \frac{1}{N} \sum_i \delta_{xx_i} \frac{1}{Z_n(i, \lambda)} p_{n-1}(x_i \mid c)^{1/\lambda}.$$  \hspace{1cm} (19)

Certainly, this does not imply that $p_n(x \mid c) \propto p_{n-1}(x \mid c)^{1/\lambda}$ everywhere, because of the $\delta_{xx_i}$ factor which picks out only a finite number of points.

One might wonder why with these mistakes, the authors still obtain an algorithm that looks and performs like $k$-means. The reason is because their sequence of mistakes leads to the result in eqn 15 that effectively assumes that IB has access to geometric information it should not, namely the cluster centers at step $n$. Since these are exactly what $k$-means uses to assign points to clusters, it is not surprising that the behavior then resembles $k$-means.

7 Acknowledgements

We would like to thank Léon Bottou for helpful discussions. We would also like to acknowledge financial support from NIH K25 GM098875 (Schwab) and the Hertz Foundation (Strouse).
References

Alemi, A. A., Fischer, I., Dillon, J. V., and Murphy, K. (2016). Deep Variational Information Bottleneck. arXiv e-prints. https://arxiv.org/abs/1612.00410.

Chechik, G., Globerson, A., Tishby, N., and Weiss, Y. (2005). Information Bottleneck for Gaussian Variables. Journal of Machine Learning Research, 6:165–188.

Cover, T. M. and Thomas, J. A. (2006). Elements of Information Theory. Wiley-Interscience.

Creutzig, F., Globerson, A., and Tishby, N. (2009). Past-future information bottleneck in dynamical systems. Physical Review E, 79(4):041925–5.

Kingma, D. P. and Welling, M. (2013). Auto-Encoding Variational Bayes. arXiv e-prints. https://arxiv.org/abs/1312.6114.

LeCun, Y. (2016). Predictive learning. NIPS.

MacKay, D. J. C. (2002). Information Theory, Inference, and Learning Algorithms. Cambridge University Press.

Palmer, S. E., Marre, O., Berry II, M. J., and Bialek, W. (2015). Predictive information in a sensory population. Proceedings of the National Academy of Sciences, 112(22):6908–6913.

Ravid, S. and Tishby, N. (2017). Opening the black box of deep neural networks via information. arXiv e-prints, cs.LG. https://arxiv.org/abs/1703.00810.

Rubin, J., Ulanovsky, N., Nelken, I., and Tishby, N. (2016). The representation of prediction error in auditory cortex. PLoS computational biology, 12(8):e1005058–28.

Slonim, N., Atwal, G., Tkacik, G., and Bialek, W. (2005). Information-based clustering. Proceedings of the National Academy of Sciences, 102(51):18297–18302.

Slonim, N. and Tishby, N. (2001). The Power of Word Clusters for Text Classification. In European Colloquium on Information Retrieval Research.

Still, S. and Bialek, W. (2004). How many clusters an information-theoretic perspective. Neural Computation, 16(12):2483–2506.

Still, S., Bialek, W., and Bottou, L. (2003). Geometric clustering using the information bottleneck method. NIPS.

Strouse, D. and Schwab, D. J. (2017). The Deterministic Information Bottleneck. Neural Computation.

Tishby, N., Pereira, F. C., and Bialek, W. (1999). The information bottleneck method. Proceedings of The 37th Allerton Conference on Communication, Control, and Computing, pages 368–377.

Tishby, N. and Zaslavsky, N. (2015). Deep Learning and the Information Bottleneck Principle. arXiv e-prints. https://arxiv.org/abs/1703.00810.