Learning Nearest-Neighbor Quantizers from Labeled Data by Information Loss Minimization

Svetlana Lazebnik and Maxim Raginsky
Beckman Institute, University of Illinois
405 N. Mathews Ave., Urbana, IL 61801
{slazebni,maxim}@uiuc.edu

Abstract

This paper proposes a technique for jointly quantizing continuous features and the posterior distributions of their class labels based on minimizing empirical information loss, such that the index K of the quantizer region to which a given feature X is assigned approximates a sufficient statistic for its class label Y. We derive an alternating minimization procedure for simultaneously learning codebooks in the Euclidean feature space and in the simplex of posterior class distributions. The resulting quantizer can be used to encode unlabeled points outside the training set and to predict their posterior class distributions, and has an elegant interpretation in terms of universal lossless coding. The promise of our method is demonstrated for the application of learning discriminative visual vocabularies for bag-of-features image classification.

1 INTRODUCTION

Many statistical learning approaches that deal with continuous signals such as speech or images rely on forming discrete intermediate representations. For example, *bag-of-features* image classification methods (Csurka et al. 2004) quantize high-dimensional descriptors of local image patches to form a discrete *visual vocabulary* and then represent images by frequency counts of the *visual words* contained in them. A good quantizer for continuous features that are also supplied with class labels should fulfill the following two objectives: first, the quantized representation of the data should retain as much information as possible about its class; and second, the rule for encoding the data should be as simple as possible, and it should not depend on the posterior class distribution so it can be applied to test samples with unknown labels. This paper presents a method that meets these objectives by learning a nearest-neighbor quantizer based on a finite set of prototypes in the feature space, such that the index of the partition cell to which a given point is assigned approximates a sufficient statistic for its class label.

The problem of using supervisory information to produce more discriminative compressed representations of data is of great practical importance and has been studied for several decades in statistical learning and vector quantization literature — see (Kohonen 1990; Oehler and Gray 1995; Rao et al. 1996) for just a few examples. Several promising recent approaches are based on the notions of sufficient statistics and mutual information (Dhillon et al. 2003; Slonim et al. 2005; Tishby et al. 1999). However, these information-theoretic approaches are tailored primarily towards clustering discrete entities such as words or genes. They produce partitions that are defined only for the training set and cannot be used for encoding points with unknown class labels. By contrast, our approach is designed specifically for continuous signal spaces and has a simple encoding rule that does not depend on the class label and can be naturally extended outside the training set. Its practical utility is demonstrated in Section 4.2 for producing effective codebooks for bag-of-features image classification.

2 THE APPROACH

2.1 MINIMIZING EMPIRICAL INFORMATION LOSS

Consider a pair (X, Y) of jointly distributed random variables, where $X \in \mathcal{X}$ is a continuous *feature* and $Y \in \mathcal{Y}$ is a discrete *class label*. In the classification setting, given a training sequence $\{(X_i, Y_i)\}_{i=1}^N$ of i.i.d. samples drawn from $P(x, y)$, one typically seeks to minimize the probability of classification error $\Pr[\hat{Y}(X) \neq Y]$. However, in order to compute this probability, we must restrict ourselves to some specific, possibly suboptimal, classification procedure. A more general approach is based on the notion of *sufficient statistics*. Informally, a sufficient statistic of X for Y contains as much information about Y as X itself. Hence an optimal hypothesis testing procedure operating

on the sufficient statistic will perform as well as an optimal predictor of Y directly from X. This framework allows us to learn compressed representations of X that retain as much discriminative power as possible without having to commit to any particular classifier.

We seek a partitioning of $\mathcal{X}$ into C disjoint regions, such that the random variable $K \in \{1, \ldots, C\}$ giving the index of the partition cell of X would be a sufficient statistic of X for Y. This condition can be expressed in terms of *mutual information* as $I(K;Y) = I(X;Y)$ (Cover and Thomas 1991; Kullback 1968). Let P_x denote $P(y|X = x)$, μ the distribution of X, and $P = \int_{\mathcal{X}} P_x d\mu(x)$ the distribution of Y. Then

$$I(X;Y) = \int_{\mathcal{X}} D(P_x \| P) d\mu(x),$$

where $D(\cdot\|\cdot)$ is the relative entropy or the *Kullback–Leibler divergence* (Kullback 1968) defined as

$$D(P_x\|P) = \sum_{y\in\mathcal{Y}} P_x(y) \log \frac{P_x(y)}{P(y)}, \quad \text{and}$$

$$\begin{aligned} I(K;Y) &= \sum_{y\in\mathcal{Y}} \sum_{k=1}^{C} P(y,k) \log \frac{P(y,k)}{P(y)P(k)} \\ &= \sum_{k=1}^{C} P(k) D(P_k\|P)\,. \end{aligned}$$

In practice, going from the continuous data X to a quantized version K is bound to lose some discriminative information and so K cannot a sufficient statistic of X in the strict mathematical sense. Instead, we would like to minimize the *information loss* $I(X;Y) - I(K;Y)$ over all partitions of $\mathcal{X}$. Since the true distribution of (X,Y) is unknown, we have to minimize an empirical version of the loss. To that end, we can use the training sequence to approximate μ by the empirical distribution $\mu_N = \frac{1}{N}\sum_{i=1}^{N} \delta_{X_i}$ and to obtain estimates of P_{X_i} for each $i = 1, 2, \ldots, N$ either by point masses, i.e., $P_{X_i} = \delta_{Y_i}$, or by using a consistent nonparametric density estimator such as Parzen windows. Then it can be shown (Banerjee et al. 2005; Dhillon et al. 2003) that minimizing the empirical information loss is equivalent to partitioning the training sequence $\{X_i\}_{i=1}^{N}$ into C disjoint sets $\mathcal{R}_1, \ldots, \mathcal{R}_C$ and associating with these sets C probability distributions $\pi_1, \ldots, \pi_C$ in the interior of the probability simplex $P(\mathcal{Y})$[1] that jointly minimize the following objective function:

$$\sum_{k=1}^{C} \sum_{X_i\in\mathcal{R}_k} D(P_{X_i}\|\pi_k). \tag{1}$$

[1]The requirement that π_k lie in the interior of $P(\mathcal{Y})$ is a technical condition dictated by the property of $D(\cdot\|\cdot)$ as a Bregman divergence (Banerjee et al. 2005). It is not a restrictive condition in practice, since one can always perturb π_k by a small amount to force it into the interior.

The optimization can be performed by an iterative descent algorithm initialized by some choice of $\{\pi_k\} \subset \mathrm{Int}(\mathcal{P}(\mathcal{Y}))$, where each X_i is assigned to $\mathcal{R}_k$ with the smallest $D(P_{X_i}\|\pi_k)$, and the class distribution centroids π_k are then recomputed by averaging the P_{X_i}'s over each $\mathcal{R}_k$:

$$\pi_k = \frac{1}{|\mathcal{R}_k|} \sum_{X_i\in\mathcal{R}_k} P_{X_i}\,. \tag{2}$$

The above approach has been used by (Dhillon et al. 2003) to cluster words for text document classification. However, (1) is not suited for continuous data because it produces an arbitrary partition of the discrete input set without any regard to spatial coherence or complexity of the resulting regions. It is also unsuited for classifying or predicting class labels for points $X \in \mathcal{X}$ outside the training set, because extending the mapping K to these points requires knowledge of the joint distribution of X and Y, which is what we are trying to estimate in the first place. As discussed next, we propose to resolve these difficulties by placing structural constraints on the partitions and the encoding rule.

2.2 CONSTRAINING THE ENCODER

In the setting of this paper, we assume that the data X comes from a compact subset $\mathcal{X}$ of a Euclidean space $\mathbb{R}^d$. Because information loss minimization over arbitrary partitions of $\mathcal{X}$ is intractable, we propose instead to search only the space of *Voronoi partitions* of $\mathcal{X}$ with respect to C centers or prototypes $m_1, \ldots, m_C \in \mathcal{X}$. In other words, we seek the set of centers $\mathcal{M}^* = \{m_1^*, \ldots, m_C^*\} \subset \mathcal{X}$ and their associated posterior class distributions $\Pi^* = \{\pi_1^*, \ldots, \pi_C^*\} \subset \mathrm{Int}(\mathcal{P}(\mathcal{Y}))$ that jointly minimize

$$\sum_{k=1}^{C} \sum_{X_i\in\mathcal{R}(m_k)} D(P_{X_i}\|\pi_k), \tag{3}$$

where $\mathcal{R}(m_k) \triangleq \{x \in \mathcal{X} : \|x - m_k\| \le \|x - m_j\|, \forall j \neq k\}$ is the Voronoi cell of m_k. In doing so, we sacrifice some optimality with respect to information loss, but gain by having a simple encoding rule. Namely, the partition (quantizer) index of a point $X \in \mathcal{X}$ is simply the index of its nearest prototype $m_k \in \mathcal{M}^*$. Note that this rule does not involve the (possibly unknown) label of X, and is thus suitable for encoding unlabeled data. Moreover, nearest-prototype quantization can naturally lead to classification: having mapped X onto its partition index k, we can predict the label $\hat{Y}$ of X by the maximum a posteriori probability (MAP) criterion:

$$\hat{Y} = \arg\max_{y\in\mathcal{Y}} \pi_k(y)\,. \tag{4}$$

In the subsequent experiments, we choose this MAP classification procedure primarily for its simplicity. However, as discussed earlier, the loss minimization framework is generic, so in principle we can learn any other classifier

based on the quantized representation of X and the estimated posterior distributions π_k.

Unfortunately, the objective function defined by (3) is still unsatisfactory: while the optimal choice of $\Pi = \{\pi_k\}$ for a given $\mathcal{M} = \{m_k\}$ is given by (2), optimizing the codebook $\mathcal{M}$ for a given Π is a difficult combinatorial problem. Therefore we opt for a suboptimal design procedure suitable for designing vector quantizers with structurally constrained encoders (Gersho and Gray 1992; Rao et al. 1996). Namely, we make the objective function differentiable by allowing "soft" partitions of the feature space. Let $w_k(x)$ denote the "weight" of assignment of a point $x \in \mathcal{X}$ to $\mathcal{R}(m_k)$, with $\sum_{k=1}^{C} w_k(x) = 1$. As suggested by Rao et al. (Rao et al. 1996), a natural choice for these weights is the Gibbs distribution

$$w_k(x) = \frac{e^{-\beta\|x-m_k\|^2/2}}{\sum_j e^{-\beta\|x-m_j\|^2/2}}, \tag{5}$$

where $\beta > 0$ is the parameter that controls the "fuzziness" of the assignments, such that smaller β's correspond to softer cluster assignments, and the limit of infinite β yields hard clustering. While in principle it is possible to use annealing techniques to pass to the limit of infinite β (Rose 1998), we have found that a fixed value of β works well in practice (our method for selecting this value in the experiments will be discussed in Section 4). Note also that we deliberately avoid any probabilistic interpretation of (5) even though it has the form of the posterior probability for a Gaussian distribution with $\beta = \frac{1}{\sigma^2}$. As will be further discussed in Section 4.1, ours is not a generative approach and does not assume any specific probabilistic model underlying the data.

We are now ready to write down the final form of our objective function:

$$E(\mathcal{M}, \Pi) = \sum_{i=1}^{N}\sum_{k=1}^{C} w_k(X_i) D(P_{X_i}\|\pi_k)\,. \tag{6}$$

A local optimum of this function can be found via alternating minimization, where we first hold Π fixed and update $\mathcal{M}$ to reduce the objective function, and then hold $\mathcal{M}$ fixed and update Π. For a fixed $\Pi = \{\pi_k\}$, $E(\mathcal{M}, \Pi)$ is reduced by gradient descent over the m_k's. The update for each m_k has the form

$$\begin{aligned} m_k^{(t+1)} &= m_k^{(t)} - \alpha \frac{\partial E^{(t)}}{\partial m_k^{(t)}} \\ &= m_k^{(t)} - \alpha \sum_{i=1}^{N}\sum_{j=1}^{C} D(P_{X_i}\|\pi_j^{(t)}) \frac{\partial w_j^{(t)}(X_i)}{\partial m_k^{(t)}}, \end{aligned} \tag{7}$$

where $\alpha > 0$ is the *learning rate* shared by all the centers and found using line search (Bishop 1995), and

$$\frac{\partial w_j(x)}{\partial m_k} = \beta[\delta_{jk} w_k(x) - w_k(x) w_j(x)](x - m_k)\,. \tag{8}$$

For a fixed codebook $\mathcal{M}$, the minimization over Π is accomplished in closed form by setting the derivatives of the Lagrangian $E(\mathcal{M}, \Pi) + \sum_k \lambda_k \sum_y \pi_k(y)$ w.r.t. $\pi_k(y)$ to zero for all k and all $y \in \mathcal{Y}$ and solving for $\pi_k(y)$ and for the Lagrange multipliers λ_k. The resulting update is

$$\pi_k^{(t+1)}(y) = \frac{\sum_{i=1}^{N} w_k^{(t+1)}(X_i) P_{X_i}(y)}{\sum_{y'}\sum_{i=1}^{N} w_k^{(t+1)}(X_i) P_{X_i}(y')}, \qquad \forall y \in \mathcal{Y}. \tag{9}$$

The two updates (7) and (9) are alternated for a fixed number of iterations or until the reduction in the value of the objective function falls below a specified threshold (this is guaranteed to happen in a finite number of iterations, because the sequence of objective function values produced by the updates is monotonically decreasing and bounded from below by 0, and therefore has a limit).

2.3 TRADING OFF INFORMATION LOSS AND DISTORTION

The loss minimization approach presented in the previous section does not pay any attention to the distortion $\|X - m_k\|^2$ incurred by encoding some data point to its nearest centroid. In practice, the regions produced by the above optimization procedure may be arbitrarily large or elongated, as some centroids either come too closely together or migrate far outside the support of the training set. To combat this effect, we propose in this section an optional variant of our basic objective function (6) to trade off information loss and mean squared distortion in a Lagrangian formulation:

$$\tilde{E}(\mathcal{M}, \Pi) = E(\mathcal{M}, \Pi) + \lambda F(\mathcal{M}, \Pi)\,, \tag{10}$$

where λ is a tradeoff parameter, and

$$F(\mathcal{M}, \Pi) = \sum_{i=1}^{N}\sum_{k=1}^{C} w_k(X_i)\|X_i - m_k\|^2 \tag{11}$$

is the standard distortion function for soft clustering (Gersho and Gray 1992). An analogous Lagrangian approach has been used by (Oehler and Gray 1995) for joint compression and classification of images, where the objective function is a sum of a Bayes weighted risk term and a mean squared error term. The updates for the m_k are once again given by $m_k^{(t+1)} = m_k^{(t)} - \alpha \frac{\partial \tilde{E}^{(t)}}{\partial m_k^{(t)}}$ (the full expression is omitted for lack of space), and the updates for π_k are given by (9).

The behavior of the modified objective function (10) is demonstrated experimentally in Figure 3 in Section 4.1; in all the other experiments we stick with the original objective function (6).

3 DISCUSSION

3.1 UNIVERSAL LOSSLESS CODING INTERPRETATION

Our approach has an interpretation in terms of *universal lossless coding* (Rissanen 1984) of class labels Y. Based on the standard correspondence between discrete probability distributions and lossless codes (Cover and Thomas 1991), knowing P_x is equivalent to knowing the optimal lossless code for Y given $X = x$. This code encodes each $Y = y$ with a codeword of length $\ell_x(y) = -\log P_x(y)$ and has average codeword length equal to $H(P_x)$, the entropy of P_x. However, in our setting, we are constrained to having only C possible codes for Y associated with a partition of the feature space into C cells $\mathcal{R}_1, \ldots, \mathcal{R}_C$ and the corresponding probability distributions $\{\pi_k\}_{k=1}^C$. Upon observing a pair $(X = x, Y = y)$, we compute the index k of the cell containing x and encode y with the lossless code matched to π_k. This code produces a binary codeword of the length $\ell_k(y) = -\log \pi_k(y)$. When Y is distributed according to P_x, the excess average codeword length or *redundancy* of this code relative to the optimal code for P_x is given by

$$\mathrm{E}_{P_x}[\ell_k(Y)] - H(P_x) = D(P_x \| \pi_k)\,. \tag{12}$$

In this scenario, the optimal set of codes (or, equivalently, the distributions π_k) is the one with minimal average redundancy

$$\sum_{k=1}^{C} \int_{\mathcal{R}_k} D(P_x \| \pi_k) d\mu(x)\,. \tag{13}$$

When we restrict the quantizers to nearest-neighbor ones and when we do not possess full knowledge of the distribution of (X, Y), but instead have access to a training sequence $\{(X_i, Y_i)\}_{i=1}^N$, this problem reduces to minimizing the objective function in (3). Finally, when the probabilities P_{X_i} are estimated by point masses δ_{Y_i}, the objective function simplifies to

$$-\sum_{k=1}^{C} \sum_{X_i \in \mathcal{R}_k} \log \pi_k(Y_i) = \sum_{k=1}^{C} \sum_{X_i \in \mathcal{R}_k} \ell_k(Y_i)\,. \tag{14}$$

This has the interpretation of minimizing the sum of total description lengths of the labels Y_i corresponding to the X_i's in each partition cell $\mathcal{R}_k$.

3.2 COMPARISON TO OTHER INFORMATION-THEORETIC CLUSTERING METHODS

The *information bottleneck* (IB) method (Tishby et al. 1999) seeks a compressed representation $\hat{X}$ of the data by minimizing $I(X; \hat{X}) - \beta I(\hat{X}; Y)$ over all conditional probability distributions $P(\hat{x}|x)$, where $\beta > 0$ is a variational parameter. Besides being based on a different objective function, our method differs from IB in other important respects. In IB, in order to compute $P(\hat{x}|x)$, and thus the encoding of X to $\hat{X}$, one must have full knowledge of $P(y|x)$. Thus, IB is fundamentally an unsupervised procedure for grouping objects from the training set based on the side information provided by Y. By contrast, our encoding rule does not involve $P(y|x)$, and, in fact, can be used to predict this distribution for points outside the training set. Second, IB does not impose any continuous structure on $\mathcal{X}$ and can in principle result in highly complex partitions, while we admit only Voronoi partitions in order to make the operation of our classifier outside the training data as simple as possible.

Information-based clustering (IBC) (Slonim et al. 2005) is a recent technique aimed at finding data partitions that simultaneously satisfy the objectives of coherence and informativeness. In IBC, one seeks a soft assignment of objects $x \in \mathcal{X}$ to C clusters $\mathcal{R}_1, \ldots, \mathcal{R}_C$ to maximize an objective function of the form $\beta \sum_{k=1}^{C} s(\mathcal{R}_k) - I(X; K)$, where β is a variational constant and $s(\mathcal{R}_k)$ is a measure of "similarity" between the objects in $\mathcal{R}_k$ based on the mutual information between some domain-specific variable Y correlated with the observed variable X. For example, X and Y may be genes and their expression data, or stocks and their prices. Just as with IB, cluster assignment is well-defined only for objects in the training sequence and encoding an observation $X \in \mathcal{X}$ requires the knowledge of the joint distribution of X and Y. Thus, IBC is also unsuitable for our application of simultaneously quantizing continuous data and predicting its class distribution.

4 EXPERIMENTAL EVALUATION

This section presents an experimental evaluation on several synthetic and real datasets. The two main implementation issues for our method are estimation of posterior probabilities P_{X_i} and the choice of the "softness" constant β. For the results reported below, we estimate P_{X_i} by averaging the point masses associated with the labels of the ten nearest neighbors of X_i and its own label Y_i, but we have also found the point mass estimate $P_{X_i} = \delta_{Y_i}$ to produce very similar performance. We set β to $\frac{d}{\hat{\sigma}^2}$, where d is the dimensionality of the data and $\hat{\sigma}^2$ is the mean squared error of the k-means clustering that we use to initialize the loss minimization procedure.

Section 4.1 uses classification as an example task to demonstrate the effectiveness of our information loss minimization strategy. However, it is important to emphasize that learning stand-alone classifiers is *not* the primary goal of our method. Instead, our target application is producing maximally informative quantized representations of continuous data that can be incorporated into more complex statistical models. Such models may not even be aimed at classifying the individual features directly, but at combining them into higher-level representations, e.g., combining multiple phonemes to form an utterance or multiple local

Dataset		# classes	# samples	dim.	10NN rate	Bayes rate
Concentric[1]	(synthetic)	2	2,500	2	98.01 ± 0.44	100
Nonlinear	(synthetic)	2	10,000	2	95.65 ± 0.19	96.32
Clouds[1]	(synthetic)	2	5,000	2	88.32 ± 0.43	90.33
Texture[1]	(real)	11	5,500	40	97.35 ± 0.27	-
Satimage[2]	(real)	6	6,435	36	89.18 ± 0.45	-
USPS[3]	(real)	10	9,298	256	94.46 ± 0.29	-

Table 1: Summary of the datasets used in our experiments. The *nonlinear* dataset was generated by us, and the rest were downloaded from the URLs listed in the footnotes. *Texture* contains features of small image patches taken from 11 classes from the Brodatz album. *Satimage* is Landsat satellite measurements for 6 classes of soil. The features in *USPS* are grayscale pixel values for 16×16 images of 10 digits from postal envelopes.

image patches to form a global image model. Accordingly, we will demonstrate in Section 4.2 the use of our method to build effective discrete visual vocabularies for image classification.

4.1 SYNTHETIC AND REAL DATA

Table 1 is a summary of the datasets used in the experiments of this section. For each dataset, the table lists the average performance of a ten-nearest-neighbor (10NN) classifier trained on random subsets consisting of half the samples. This is the effective upper bound on the MAP classification performance of our "info-loss" method, because the info-loss method makes its decisions based on the quantized version of the nearest-neighbor estimate of the posterior class distribution. For the three synthetic datasets, the table also lists the theoretically computed optimal Bayes upper bound. Note that for these datasets, the 10NN performance comes quite close to the Bayes bound.

A good "floor" or a baseline for our method is provided by standard k-means quantization, where the data centers $m_1, \ldots m_C$ are learned without taking class labels into account, and the posterior distributions $P(y|k) = \pi_k$ are obtained afterwards by the averaging rule (2). As an alternative baseline that does take advantage of class information for learning the data centers, but does not directly minimize information loss, we chose a generative framework where each class conditional density $P(x|y)$ is modeled as a mixture of C Gaussians, and mixture components are shared between all the classes:

$$P(x|y) = \sum_{k=1}^{C} P(x|k)\, P(k|y)\,. \tag{15}$$

$P(x|k)$ is a Gaussian with mean m_k and a spherical covariance matrix σ^2I, $\sigma^2 = \frac{1}{\beta}$. The parameters of this model, i.e., the means m_k and the class-specific mixture weights $P(k|y)$, are learned using the EM algorithm (Bishop 1995). [Alternatively, one could use GMVQ, a hard clustering algorithm for Gauss mixture modeling (Aiyer et al. 2005).]

[1]http://www.dice.ucl.ac.be/neural-nets/Research/Projects/ELENA/databases/

[2]http://www.ics.uci.edu/~mlearn/MLSummary.html

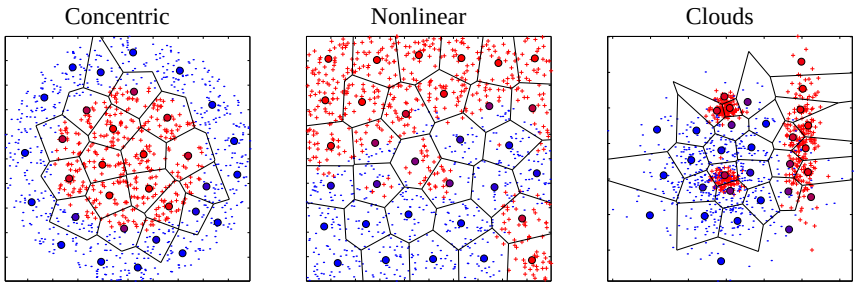
[3]ftp://ftp.kyb.tuebingen.mpg.de/pub/bs/data/

Instead of fixing a global value of σ^2, we also experimented with including the variances σ_k^2 as parameters in the optimization, but this had little effect on classification performance, or even resulted in overfitting for the more high-dimensional datasets.

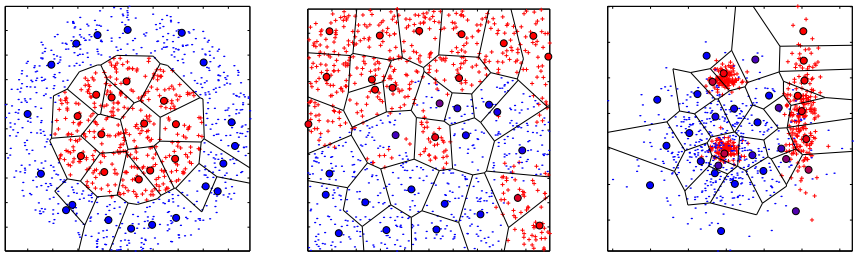
Figure 1 shows results on three two-dimensional two-class synthetic datasets. Part (a) shows the centers and partitions produced by k-means and used to initialize both EM and info-loss optimizations. Part (b) shows the resulting info-loss partitions. In all three cases, our method partitions the data space in such a way as to separate the two classes as much as possible. For example, the "concentric" dataset (left column) consists of uniformly sampled points, such that the "red" class is contained inside a circle and the "blue" class forms a ring around it. The regions produced by k-means do not respect the circular class boundary, whereas the regions produced by the info-loss method conform to it quite well. It is important to keep in mind, however, that separating classes is not the primary goal of information loss minimization. Instead, the criterion given by (6) is more general, seeking to partition the data into regions where the posterior distributions P_{X_i} of the individual data points are as homogeneous as possible, measured in terms of their similarity to the "prototype" distribution π_k. When the classes in the dataset are separable, this criterion naturally leads to regions whose prototype distributions are nearly "pure," i.e., dominated by a single class.

Figure 1 (c) compares the classification performance of the three clustering methods. For k-means and info-loss, MAP classification is performed using the rule (4) while for EM, it is derived from the probabilistic model (15). For the "concentric" dataset, the info-loss classification rate falls somewhat as the codebook size increases from 16 to 128. This is because the decision regions in this case are simple enough to be approximated well even with $C = 8$, and increasing C causes the method to overfit. Finally, Figure 1 (d) compares the performance of the three methods w.r.t. minimizing information loss or equivalently, maximizing the mutual information $I(K;Y)$ between the region index and the class label. Once again, info-loss outperforms both k-means and EM.

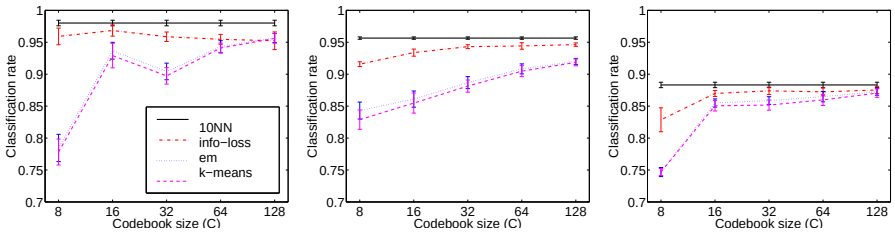
Figure 2 shows analogous results for the three real datasets



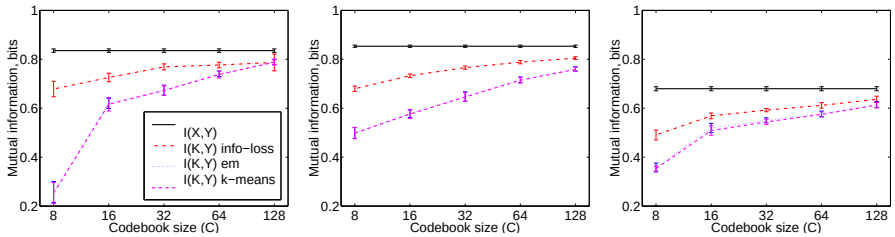
(a) Centers and partitions produced by k-means ($C = 32$)



(b) Centers and partitions produced by our info-loss method ($C = 32$)



(c) Classification rate vs. codebook size



(d) Mutual information vs. codebook size

Figure 1: Results on synthetic data (best viewed in color). For (c) and (d), we have performed 10 runs with different random subsets of half the samples used to train the models and the rest used as test data for reporting classification accuracy and mutual information. The height of the error bars is twice the standard deviation for each measurement. In (d), information loss is given by the vertical distance between $I(X;Y)$ and $I(K;Y)$.

in our study. As in Figure 1, info-loss outperforms the two baseline methods. Recall from Table 1 that these datasets have as many as 11 classes and 256 dimensions, so our method appears to scale quite well as the number of classes and the dimensionality of the feature space increase. It is worth noting that in all our experiments, EM achieves only a small improvement over k-means, so it seems to be relatively ineffective as a way of incorporating class information into clustering. This weakness may be due to the fact that the generative model (15) encodes a strong relationship between the density of the data and its class structure. By contrast, our info-loss framework is much more flexible, because it makes minimal assumptions about the data density, approximating it by the empirical distribution, and does not require any correspondence between the modes of this density and the posterior class distribution. As far as our method is concerned, the data can be generated using one process, such as a mixture of Gaussians, and the class distribution can be "painted on" by a completely different process.

Finally, Figure 3 demonstrates the tradeoff between quantizer distortion and information loss for the Lagrangian objective function (10) of Section 2.3. We can see that for the *texture* dataset, it is possible to achieve "the best of both worlds": for intermediate values of the tradeoff parameter (i.e., $\lambda = 1$), classification accuracy is not significantly affected, while the mean squared Euclidean distortion in the feature space is almost as low as for the pure k-means algorithm.

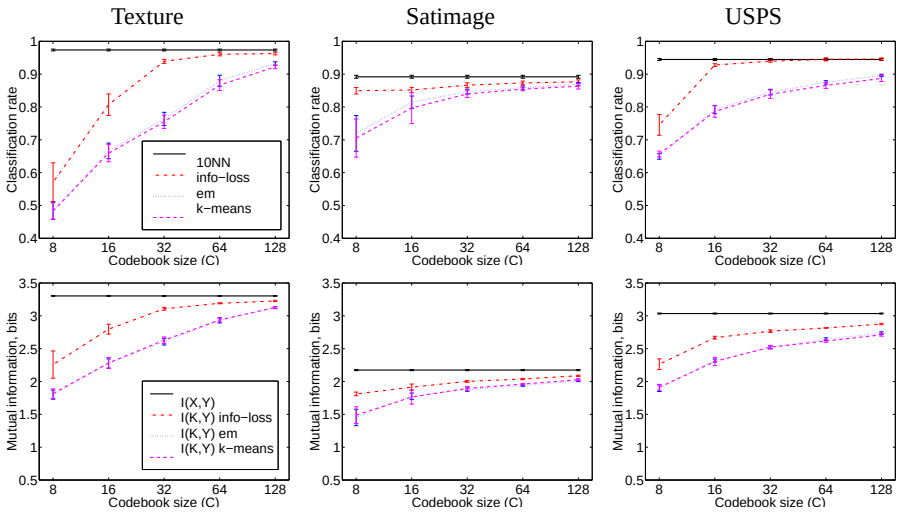


Figure 2: Results for three real image datasets. First row: classification rate vs. codebook size. Second row: mutual information vs. codebook size. As in Figure 1, means and standard deviations are reported over 10 runs with half the dataset randomly selected for training and half for testing.

λ	distortion	info. loss	class. rate
0	0.424 ± 0.04	0.282 ± 0.04	94.0 ± 1.1
0.1	0.386 ± 0.02	0.273 ± 0.03	94.5 ± 0.8
0.5	0.276 ± 0.02	0.329 ± 0.07	92.7 ± 2.6
1.0	0.247 ± 0.01	0.375 ± 0.04	90.7 ± 2.3
5.0	0.201 ± 0.01	0.479 ± 0.08	87.0 ± 3.1
10.0	0.192 ± 0.01	0.561 ± 0.06	84.2 ± 2.2
∞	0.184 ± 0.01	0.705 ± 0.05	75.6 ± 1.9

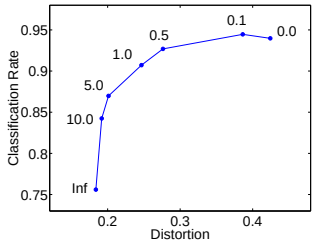


Figure 3: Trading off quantizer distortion and information loss for the *texture* dataset with $C = 32$. Left: mean squared distortion, information loss, and classification rate as a function of λ, where $\lambda = 0$ corresponds to pure info-loss clustering and $\lambda = \infty$ corresponds to k-means. Right: classification error plotted as a function of distortion. The values of λ corresponding to each data point are shown on the plot.

4.2 APPLICATION: CONSTRUCTING CODEBOOKS FOR BAG-OF-FEATURES IMAGE CLASSIFICATION

In this section, we consider the application of building visual vocabularies for *bag-of-features* image classification (Csurka et al. 2004). Analogously to *bag-of-words* document classification (see, e.g., McCallum and Nigam 1998), this framework represents images by histograms of discrete indices of the "visual words" contained in them. Despite the extreme simplicity of this model (in particular, its lack of information about the spatial layout of the patches), it has often outperformed more complex parts-and-relations models, and is currently considered to be a state-of-the-art approach to image classification. Despite the promise of bag-of-features methods, the issue of effective design of visual codebooks is currently not well understood, and remains an active research area.

Figure 4 demonstrates results of our quantization procedure applied to a fifteen-category scene dataset consisting of 4485 images (Lazebnik et al. 2006).[2] This dataset includes both indoor and outdoor categories and is quite challenging (for example, it is difficult to distinguish indoor categories such as bedroom and living room). For training sets consisting of 100 images per class, (Lazebnik et al. 2006) report a bag-of-features classification rate of 72.2% with a vocabulary of size 200 computed by k-means clustering. Following their procedure, we extract image features by computing 128-dimensional SIFT descriptors (Lowe 2004) of 16×16 patches sampled on a regular 8×8 grid. Next, we form a vocabulary by running either k-means or our info-loss algorithm on $22,500$ patches randomly sampled from all the classes in the training set. Finally, we encode the patches in each image I into the index of its closest codebook center or "vocabulary word," and represent the image as a vector of frequency counts $N_k(I)$ of each index k. Figure 4 shows results of classifying histograms based on the two types of codebooks. We use two different classifiers, Naive Bayes and support vector machines with a histogram intersection kernel. Overall, as seen from Figure 4 (a), codebooks produced by our method yield a statistically significant improvement of at least 2%. The improvement is higher for smaller vocabulary sizes and for Naive Bayes, which is a weaker classification method that relies more directly on the quality of the probability estimates output by the quantizer. Specifically, Naive Bayes performs maxi-

[2] http://www-cvr.ai.uiuc.edu/ponce_grp/data

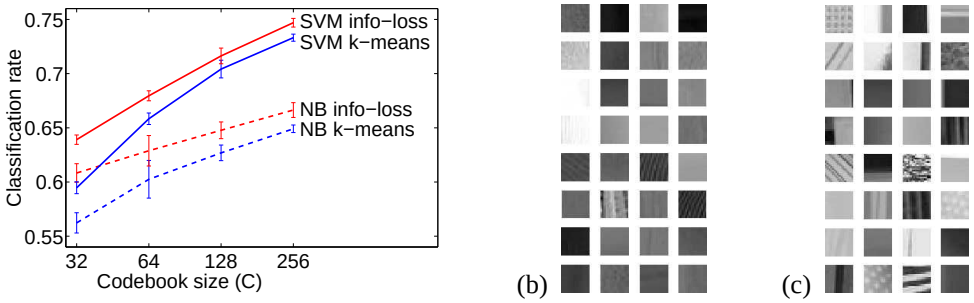


Figure 4: (a) Performance of bag-of-features classification for different dictionary sizes and different methods of dictionary formation. The results are averaged over five runs with different random training/test splits. Error bars show twice the standard deviations. (b) Dictionary of size 32 produced by k-means. (c) Dictionary produced by our info-loss method.

mum likelihood classification according to the *multinomial event model* (McCallum and Nigam 1998):

$$P(I|y) = \prod_k P(k|y)^{N_k(I)}\,, \tag{16}$$

where in the case of a codebook output by our method, $P(k|y)$ is obtained directly by Bayes rule from the centroid π_k. It is interesting to note the perceptual difference in the two types of codebooks, as seen in Figure 4 (b) and (c) for $C = 32$. The centers produced by our method are higher-contrast patches with salient edges or texture patterns. Intuitively, such patterns should be more informative about the image category than more generic, low-contrast patches that make up the standard k-means dictionary.

5 SUMMARY

This paper has presented a method for compressing continuous datasets while minimizing the loss of discriminative information. It works by simultaneously partitioning the feature space by the nearest-neighbor rule with respect to the Euclidean distance, and the simplex of probability distributions by the nearest-neighbor rule with respect to the KL divergence. Moreover, the encoding rule follows the Markov chain $X \to K \to Y$, so that assigning a point to its quantizer region in feature space immediately leads to an estimate of its posterior class distribution. This estimate can be used directly for MAP classification as in Section 4.1, or incorporated into a more complex statistical modeling framework, as demonstrated in Section 4.2 for bag-of-features image classification.

References

A. Aiyer, K. Pyun, Y. Huang, D.B. O'Brien and R.M. Gray. Lloyd clustering of Gauss mixture models for image compression and classification. *Signal Processing: Image Commun.* 20: 459–485 (2005).

A. Banerjee, S. Merugu, I.S. Dhillon and J. Ghosh. Clustering with Bregman divergences. *J. Mach. Learning Res.* 6:1705–1749 (2005).

C. Bishop. *Neural Networks for Pattern Recognition.* Clarendon Press, Oxford, 1995.

T.M. Cover and J.A. Thomas. *Elements of Information Theory*, Wiley, New York, 1991.

G. Csurka, C. Dance, L. Fan, J. Willamowski and C. Bray. Visual categorization with bags of keypoints. *ECCV Workshop on Statistical Learning in Computer Vision* (2004).

I. Dhillon, S. Mallela, and R. Kumar. A divisive information-theoretic feature clustering algorithm for text classification. *J. Mach. Learning Res.* 3:1265-1287 (2003).

A. Gersho and R.M. Gray. *Vector Quantization and Signal Compression*, Kluwer, Boston, 1992.

T. Kohonen. Improved versions of learning vector quantization. *Proc. IEEE Int. Joint Conf. on Neural Networks*, vol. I, pp. 545–550 (1990).

S. Kullback. *Information Theory and Statistics*, Dover, New York, 1968.

S. Lazebnik, C. Schmid and J. Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. *Proc. IEEE Conf. Computer Vision and Pattern Recognition* (2006).

D. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110 (2004).

A. McCallum and K. Nigam. A comparison of event models for naive Bayes text classification. *AAAI-98 Workshop on Learning for Text Categorization*, pp. 41–48 (1998).

K.L. Oehler and R.M. Gray. Combining image compression and classification using vector quantization. *IEEE Trans. Pattern Analysis Mach. Intel.*, 17:461–473 (1995).

A. Rao, D. Miller, K. Rose and A. Gersho. A generalized VQ method for combined compression and estimation. *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing*, pp. 2032–2035 (1996).

K. Rose. Deterministic annealing for clustering, compression, classification, regression, and related optimization problems. *Proc. IEEE*, 86(11):2210-2239 (1998).

J. Rissanen. Universal coding, information, prediction, and estimation. *IEEE Trans. Inform. Theory*, IT-30:629–636 (1984).

N. Slonim, G.S. Atwal, G. Tkačik and W. Bialek. Information-based clustering. *Proc. Nat'l Acad. Sci.*, 102:18297–18302 (2005).

N. Tishby, F.C. Pereira and W. Bialek. The information bottleneck method. *Proc. 37th Annual Allerton Conf. on Communication, Control and Computing*, pp. 368–377 (1999).