Probabilistic Semi-supervised Clustering with Constraints

Sugato Basu  
Mikhail Bilenko  
Raymond J. Mooney  
Department of Computer Sciences  
University of Texas at Austin  
Austin, TX 78705, USA

Abstract
Unsupervised clustering can be significantly improved using supervision in the form of pairwise constraints, i.e., pairs of instances labeled as belonging to same or different clusters. In recent years, a number of algorithms have been proposed for enhancing clustering quality by employing such supervision. Such methods use the constraints to either modify the objective function, or to learn the distance measure. We propose a probabilistic generative model for semi-supervised clustering based on Hidden Markov Random Fields (HMRFs) that provides a principled framework for incorporating supervision into prototype-based clustering. The model allows the use of a broad range of clustering distortion measures, including certain Bregman divergences (e.g., squared Euclidean distance and KL divergence) and directional distances (e.g., cosine distance). We present an algorithm that performs semi-supervised clustering of data by minimizing an objective function derived from the joint probability defined over the HMRF model. Experimental results on several datasets demonstrate the advantages of the proposed framework.

Keywords: clustering, semi-supervised learning, probabilistic generative models, metric learning, learning with pairwise constraints

1. Introduction
Large amounts of unlabeled data are available in many real-life machine learning problems, e.g., uncategorized messages in an automatic email classification system, genes of unknown functions for aiding gene function prediction, etc. Labeled data is often limited and expensive to generate, since labeling typically requires human expertise. Consequently, semi-supervised learning, which uses both labeled and unlabeled data, has become a topic of significant recent interest (Blum and Mitchell, 1998; Joachims, 1999; Nigam et al., 2000). In this paper, we focus on semi-supervised clustering, where the performance of unsupervised clustering algorithms is improved with limited amounts of supervision in the form of constraints on the data.

Earlier research on semi-supervised clustering has considered supervision in the form of labeled points (Demiriz et al., 1999; Sinkkonen and Kaski, 2000; Basu et al., 2002) or constraints (Wagstaff et al., 2001; Klein et al., 2002; Xing et al., 2003; Bar-Hillel et al., 2003; Bilenko et al., 2004; Basu et al., 2004a,b). In this paper, we consider supervision provided
in the form of must-link and cannot-link constraints on pairs of points, where a must-link constraint indicates that the points should be placed in the same cluster, while a cannot-link constraint requires the two points to be placed in different clusters. In certain applications, supervision in the form of class labels may be unavailable, while pairwise constraints are easily obtained, creating the need for methods that exploit such supervision. For example, complete class labels may be unknown in the context of clustering for speaker identification in a conversation (Bar-Hillel et al., 2003), or clustering GPS data for lane-finding (Wagstaff et al., 2001). In some domains, pairwise constraints occur naturally, e.g., the Database of Interacting Proteins (DIP) data set in biology contains information about proteins co-occurring in processes (Xenarios et al., 2001), which can be viewed as must-link constraints during clustering. Constraint-based supervision is also more general than class labels: any set of classified points implies an equivalent set of pairwise constraints, but not vice versa.

Existing methods for semi-supervised clustering fall into two general categories which we call constraint-based and distance-based. Constraint-based methods use the provided labels or constraints to guide the algorithm towards a more appropriate data partitioning. This is done by modifying the objective function for evaluating clusterings so that it includes satisfying constraints (Demiriz et al., 1999), enforcing constraints during the clustering process (Wagstaff et al., 2001), or initializing and constraining the clustering based on labeled examples (Basu et al., 2002). In distance-based approaches, an existing clustering algorithm that uses a particular distance to measure clustering distortion is employed; however, the distance measure is trained to satisfy the labels or constraints in the supervised data. Several adaptive distance measures have been used for semi-supervised clustering, including string-edit distance trained using Expectation Maximization (EM) (Bilenko and Mooney, 2003), KL divergence trained using gradient descent (Cohn et al., 2003), Euclidean distance modified by a shortest-path algorithm (Klein et al., 2002), or Mahalanobis distances trained using convex optimization (Xing et al., 2003).

We propose a probabilistic model based on Hidden Markov Random Fields (HMRFs) for semi-supervised clustering that combines the constraint-based and distance-based approaches in a unified probabilistic model. We motivate an objective function for semi-supervised clustering derived from the joint probability of the HMRF model, and propose a EM-based partitional clustering algorithm, HMRF-KMEANS, that finds a local minimum of this objective function. HMRF-KMEANS can be used to perform semi-supervised clustering with different distortion measures, namely certain Bregman divergences (Banerjee et al., 2004), which includes KL divergence, squared Euclidean distance, and I divergence. In a number of applications, such as text-clustering using a vector-space model, a directional distance measure based on the angle between vectors is more appropriate (Baeza-Yates and Ribeiro-Neto, 1999). Consequently, clustering algorithms that utilize distortion measures appropriate for directional data have recently been developed (Dhillon and Modha, 2001; Banerjee et al., 2003), and the HMRF-KMEANS framework naturally extends them. Additionally, incorporating the training of distortion measure parameters in the probabilistic formulation allows learning cluster-specific distortion measures, effectively permitting clusters to lie in different subspaces.

The proposed HMRF-based approach aids unsupervised clustering by incorporating supervision provided as pairwise constraints in the following ways:
Probabilistic Semi-supervised Clustering with Constraints

- Improved initialization, where initial cluster centroids are estimated from the neighborhoods inferred from the constraints;

- Constraint-sensitive assignment of instances to clusters, where points are assigned to clusters so that the overall distortion of the points from the cluster centroids is minimized, while a minimum number of must-link and cannot-link constraints are violated;

- Iterative distance learning, where the distortion measure is re-estimated during clustering, thereby changing the input data space to respect the specified constraints as well as to incorporate data variance.

In the remainder of the paper, we describe the HMRF-KMeans framework and present experimental results on clustering data sets from different domains (including hand-written digit recognition, text documents, and gene microarray data) that demonstrate the advantages of our approach.

2. The Model

In the partitional prototype-based clustering setting that we will be considering, a set of data points is separated into a pre-specified number of clusters, where each cluster has a representative (or “prototype”), so that a well-defined cost function, involving a distortion measure between the points and the cluster representatives, is minimized. A well-known unsupervised clustering algorithm that follows this framework is K-Means (MacQueen, 1967).

Our semi-supervised clustering model considers a sample of \(n\) data points \(X = (x_1, \ldots, x_n)\), each \(x_i \in \mathbb{R}^d\) being a \(d\)-dimensional vector, with \(x_{im}\) representing its \(m\)-th component. The model relies on a distortion measure \(D(\cdot, \cdot)\) used to compute distance between points: \(D : \mathbb{R}^d \times \mathbb{R}^d \rightarrow \mathbb{R}\), where \(A\) is the set of distortion measure parameters. Supervision is provided as two sets of pairwise constraints: must-link constraints \(C_{ML} = \{(x_i, x_j)\}\) and cannot-link constraints \(C_{CL} = \{(x_i, x_j)\}\), where \((x_i, x_j) \in C_{ML}\) implies that \(x_i\) and \(x_j\) are labeled as belonging to the same cluster, while \((x_i, x_j) \in C_{CL}\) implies that \(x_i\) and \(x_j\) are labeled as belonging to different clusters. The constraints may be accompanied by associated violation costs \(W\), where \(w_{ij}\) represents the cost of violating the constraint between points \(x_i\) and \(x_j\) if such a constraint exists, that is, either \((x_i, x_j) \in C_{ML}\) or \((x_i, x_j) \in C_{CL}\).

The task is to partition the datapoints \(X\) into \(K\) disjoint clusters \((X_1, \ldots, X_K)\) so that the total distortion between the points and the corresponding cluster representatives is minimized according to the given distortion measure \(D\), while constraint violations are kept to a minimum.

2.1 HMRF Model Components

The Hidden Markov Random Field (HMRF) probabilistic framework (Zhang et al., 2001) for semi-supervised constrained clustering consists of the following components:

- An observable set \(X = (x_1, \ldots, x_n)\) of random variables, corresponding to the given data points \(X\). Note that we overload notation and use \(X\) to refer to both the given set of data points and their corresponding random variables.
An unobservable (hidden) set \( Y = (y_1, \ldots, y_n) \) of random variables, corresponding to cluster assignments of points in \( X \). Each hidden variable \( y_i \) encodes the cluster label of the point \( x_i \) and takes values from the set of cluster indices \( (1, \ldots, K) \).

An unobservable (hidden) set of generative model parameters \( \Theta \), which consists of distortion measure parameters \( A \) and cluster representatives \( M = (\mu_1, \ldots, \mu_K) \): \( \Theta = \{A, M\} \). Note that distortion measure parameters may be individual for each cluster, in which case \( A = (A_1, \ldots, A_K) \).

An observable set of constraint variables \( C = (c_{12}, c_{13}, \ldots, c_{n-1,n}) \). Each \( c_{ij} \) is a tertiary variable taking on a value from the set \((-1, 0, 1)\), where \( c_{ij} = 1 \) indicates that \((x_i, x_j) \in C_{ML}\), \( c_{ij} = -1 \) indicates that \((x_i, x_j) \in C_{CL}\), and \( c_{ij} = 0 \) corresponds to pairs \((x_i, x_j)\) that are not constrained.

Since constraints are fully observed and the described model does not attempt to model them generatively, the joint probability of \( X \), \( Y \), and \( \Theta \) is conditioned on the constraints encoded by \( C \).

Fig. 1 shows a simple example of an HMRF. \( X \) consists of five datapoints with corresponding variables \((x_1, \ldots, x_5)\) that have cluster labels \( Y = (y_1, \ldots, y_5) \), which may each take on values \((1, 2, 3)\) denoting the three clusters. Three pairwise constraints are provided: two must-link constraints \((x_1, x_2)\) and \((x_1, x_4)\), and one cannot-link constraint \((x_2, x_3)\). Corresponding constraint variables are \( c_{12} = 1 \), \( c_{14} = 1 \), and \( c_{23} = -1 \); all other variables in \( C \) are set to zero. The task is to partition the five points into three clusters. Fig. 1 demonstrates one possible clustering configuration which does not violate any constraints. The must-linked points \( x_1, x_2 \) and \( x_4 \) belong to cluster 1; the point \( x_3 \), which is cannot-linked with \( x_2 \), is assigned to cluster 2; \( x_5 \), which is not involved in any constraints, belongs to cluster 3.
2.2 Markov Random Field over Labels

Each hidden random variable $y_i \in Y$ representing the cluster label of $x_i \in X$ is associated with a set of neighbors $N_i$. The set of neighbors is defined as all points to which $x_i$ is must-linked or cannot-linked: $N_i = \{y_j | (x_i, x_j) \in C_{ML} \text{ or } (x_i, x_j) \in C_{CL}\}$. The resulting random field defined over the hidden variables $Y$ is a Markov Random Field (MRF), where the conditional probability distribution over the hidden variables obeys the Markov property:

$$\forall i, \ P(y_i | Y - \{y_i\}, \Theta, C) = P(y_i | \{y_j : y_j \in N_i\}, \Theta, C).$$  (1)

Thus the conditional probability of $y_i$ for each $x_i$, given the model parameters and the set of constraints, depends only on the cluster labels of the observed variables that are must-linked or cannot-linked to $x_i$. Then, by the Hammersley-Clifford theorem (Hammersley and Clifford, 1971), the prior probability of a particular label configuration $Y$ can be expressed as a Gibbs distribution (Geman and Geman, 1984), so that

$$P(Y | \Theta, C) = \frac{1}{Z} \exp(-v(Y)) = \frac{1}{Z} \exp(-\sum_{N_i \in N} v_{N_i}(Y)),$$  (2)

where $N$ is the set of all neighborhoods, $Z$ is the normalizing term, and $v(Y)$ is the overall label configuration potential function, which can be decomposed into a sum of functions $v_{N_i}(Y)$, each denoting the potential for every neighborhood $N_i$ in the label configuration $Y$. Since the potentials for every neighborhood are based on pairwise constraints in $C$ (and model parameters $\Theta$), the label configuration can be further decomposed as:

$$P(Y | \Theta, C) = \frac{1}{Z} \exp(-\sum_{i,j} v(i,j)),$$  (3)

where each constraint potential function $v(i,j)$ has the following form:

$$v(i,j) = \begin{cases} 
    w_{ij}f_{ML}(i,j) & \text{if } c_{ij} = 1 \text{ and } y_i \neq y_j \\
    w_{ij}f_{CL}(i,j) & \text{if } c_{ij} = -1 \text{ and } y_i = y_j \\
    0 & \text{otherwise}
\end{cases}$$  (4)

The penalty functions $f_{ML}$ and $f_{CL}$ encode the lowered probability of observing configurations of $Y$ where constraints encoded by $C$ are violated. To this end, function $f_{ML}$ penalizes violated must-link constraints and function $f_{CL}$ penalizes violated cannot-link constraints. These functions are chosen to correspond with the distortion measure by employing same model parameters $\Theta$, and will be described in detail in Section 3. Overall, this formulation for observing the label assignment $Y$ results in higher probabilities being assigned to configurations in which cluster assignments do not violate the provided constraints.
2.3 Joint Probability in HMRF

The joint probability of $X$, $Y$, and $\Theta$, given $C$, in the described HMRF model can be factorized as follows:

$$P(X, Y, \Theta|C) = P(\Theta|C) P(Y|\Theta, C) P(X|Y, \Theta, C)$$  \hspace{1cm} (5)

The graphical plate model (Buntine, 1994) of the dependence between the random variables in the HMRF is shown in Figure 2, where the clear nodes represent the hidden variables, the shaded nodes are the observed variables, the directed links show dependencies between the variables, while the lack of an edge between two variables implies conditional independence. The prior probability of $\Theta$ is assumed to be independent of $C$. The probability of observing the label configuration $Y$ depends on the constraints $C$ and current generative model parameters $\Theta$. Observed datapoints corresponding to variables $X$ are generated using the model parameters $\Theta$ based on cluster labels $Y$, independent of the constraints $C$. The variables $X$ are assumed to be mutually independent: each $x_i$ is generated individually from a conditional probability distribution $P(x_i|y_i, \Theta)$. Then, the conditional probability $P(X|Y, \Theta, C)$ can be written as:

$$P(X|Y, \Theta, C) = P(X|Y, \Theta) = \prod_{i=1}^{n} p(x_i|y_i, \Theta),$$  \hspace{1cm} (6)

where $p(\cdot|y_i, \Theta)$ is the parameterized probability density function for the $y_i$-th cluster, from which $x_i$ is generated. This probability density is related to the clustering distortion measure $D$, as described below in Section 2.4.

From Eqns. (3), (5), and (6), it follows that maximizing the joint probability on the HMRF is equivalent to maximizing:

$$P(X, Y, \Theta|C) = P(\Theta) \left( \frac{1}{Z} \exp\left( - \sum_{c_{ij} \in C} v(i, j) \right) \right) \left( \prod_{i=1}^{n} p(x_i|y_i, \Theta) \right)$$  \hspace{1cm} (7)

The joint probability in Eqn. (7) has 3 factors. The first factor describes a probability distribution over the model parameters preventing them from converging to degenerate
values, thereby providing regularization. The second factor is the conditional probability of observing a particular label configuration given the provided constraints, effectively assigning a higher probability to configurations where the cluster assignments do not violate the constraints. Finally, the third factor is the conditional probability of generating the observed data points given the labels and the parameters: if maximum likelihood (ML) estimation was performed on the HMRF, the goal would have been to maximize this term in isolation.

Overall, maximizing the joint HMRF probability in Eq. (7) is equivalent to jointly maximizing the likelihood of generating datapoints from the model and the probability of label assignments that respect the constraints, while regularizing the model parameters.

### 2.4 Semi-supervised Clustering Objective Function on HMRF

Eqn. (7) suggests a general framework for incorporating constraints into clustering. A particular choice of the conditional probability \( p(\cdot|y, \Theta) \) is directly connected to the choice of the distortion measure appropriate for the clustering task.

When considering the conditional probability \( p(\cdot|y, \Theta) \) – the probability of generating a datapoint from the \( y \)-th cluster – our attention is restricted to probability densities from the exponential family, where the expectation parameter corresponding to the \( h \)-th cluster is \( \mu_h \), the mean of the points of that cluster. Using this assumption and the bijection between regular exponential distributions and regular Bregman divergences (Banerjee et al., 2004), the conditional density for observed data can be represented as:

\[
p(x_i|y_i, \Theta) = \frac{1}{Z_{\Theta}} \exp(-D(x_i, \mu_h)),
\]

where \( D(x_i, \mu_h) \) is the Bregman divergence between \( x_i \) and \( \mu_h \), corresponding to the exponential density \( p \), and \( Z_{\Theta} \) is the normalizer. Different clustering models fall into this exponential form:

- If \( x_i \) and \( \mu_h \) are vectors in Euclidean space, and \( D \) is the square of the \( L_2 \) distance parameterized by a positive semidefinite weight matrix \( A \) (\( D(x_i, \mu_h) = \|x_i - \mu_h\|_A^2 \)), then the cluster conditional probability is a Gaussian with covariance encoded by \( A \) (Kearns et al., 1997);

- If \( x_i \) and \( \mu_h \) are probability distributions and \( D \) is the KL divergence (\( d(x_i, \mu_h) = \sum_{m=1}^{d} x_{im} \log \frac{x_{im}}{\mu_{hm}} \)), then the cluster conditional probability is a multinomial distribution (Dhillon and Guan, 2003).

The relation in Eqn. (8) holds even if \( D \) is not a Bregman divergence but a directional distance measure like cosine distance. For example, if \( x_i \) and \( \mu_h \) are vectors of unit length and \( D \) is one minus the dot-product of the vectors (\( D(x_i, \mu_h) = 1 - \sum_{m=1}^{d} \frac{x_{im} \mu_{hm}}{\|x_i\| \|\mu_h\|} \)), then the cluster conditional probability is a von-Mises Fisher (vMF) distribution with unit concentration parameter (Banerjee et al., 2003), which is essentially the spherical analog of a Gaussian. The connection between specific distortion measures studied in this paper and their corresponding cluster conditional probabilities is discussed in more detail in Section 2.6.
Putting Eqn. (8) into Eqn. (7) and taking logarithms gives the following cluster objective function, minimizing which is equivalent to maximizing the joint probability over the HMRF in Eqn. (7):

$$J_{\text{obj}} = \sum_{x_i \in X} D(x_i, \mu_{y_i}) + \sum_{c_{ij} \in C} v(i, j) - \log P(\Theta) + \log Z + \log Z_{\Theta}$$

(9)

Thus, the task is to minimize $J_{\text{obj}}$ over the hidden variables $Y$ and $\Theta$ (note that given $Y$, the means $M = (\mu_1, \ldots, \mu_K)$ are uniquely determined).

### 2.5 Parameter Priors

Following the definition of $\Theta$ in Sec. 2.1, the prior term $\log P(\Theta)$ in Eqn. (9) and the subsequent equations can be factored as follows:

$$\log P(\Theta) = \log \left( P(A) P(M) \right) = \log P(A) + \log P(M)$$

where the distortion parameters $A$ are assumed to be independent of the cluster centroids $M = (\mu_1, \ldots, \mu_K)$, and uniform priors are considered over the cluster centroids (leading to the constant term $\log P(M)$). For different distortion measures, parameter values may exist that lead to degenerate solutions of the optimization problem. For example, for squared Euclidean distance, the zero matrix $A = \mathbf{0}$ is one such solution. Employing a prior distribution $P(A)$ on distortion measure parameters provides us with a regularization mechanism for preventing degenerate solutions.

The standard Gaussian prior is unsuitable for the parameters of the distortion function, since it would permit the parameters to take on zero and negative values. An alternative prior distribution that constrains the parameter values to be non-negative is the Rayleigh distribution (Papoulis and Pillai, 2001). Assuming independence of the distortion measure parameters $a_{ij} \in A$, the prior term based on the Rayleigh distribution and the corresponding $\log P(A)$ terms are the following:

$$P(A) = \prod_{a_{ij} \in A} \frac{a_{ij} \exp \left( -\frac{a_{ij}^2}{2s^2} \right)}{s^2}$$

(10)

$$\log P(A) = \sum_{a_{ij} \in A} \left( \log a_{ij} - \frac{a_{ij}}{2s^2} - 2 \log s \right)$$

(11)

where $s$ is the width parameter of the Rayleigh distribution.

### 2.6 Adaptive Distortion Measures

Selecting an appropriate distortion measure $D$ for a clustering task typically involves knowledge about properties of the particular domain and dataset. For example, squared Euclidean distance is most appropriate for low-dimensional data with distribution close to Gaussian, while cosine distance best captures distance between data described by vectors in high-dimensional space where differences in angles are important but vector lengths are not.
Distortion measures from two families are considered in this chapter: Bregman divergences (Banerjee et al., 2004), which include parameterized squared Euclidean distance and Kullback-Leibler divergence, and distortion measures based on directional similarity functions, which include cosine similarity and Pearson’s correlation (Mardia and Jupp, 2000). The distortion measure in these cases is chosen to be the directional similarity measure subtracted from a constant sufficiently large so that the resulting value is non-negative. For both Bregman divergences and cosine distance, there exist efficient K-Means-type iterative relocation algorithms that minimize the corresponding clustering objective (Banerjee et al., 2003, 2004), which the HMRF-KMEANS naturally extends to incorporate pairwise supervision.

For many realistic datasets, off-the-shelf distortion measures may fail to capture the correct notion of similarity in a clustering setting. While some unsupervised measures like Mahalanobis distance and Pearson’s distance attempt to correct distortion estimates using the global mean and variance of the dataset, these measures may still fail to estimate distances accurately if the attributes’ true contributions to the distance is not correlated with their variance. Several semi-supervised clustering approaches exist that incorporate adaptive distortion measures, including parameterizations of Jensen-Shannon divergence (Cohn et al., 2003) and squared Euclidean distance (Bar-Hillel et al., 2003; Xing et al., 2003). However, these techniques use only constraints to learn the distortion measure parameters and exclude unlabeled data from the parameter learning step, as well as separate the parameter learning step from the clustering process.

Going a step further, the HMRF model provides an integrated framework which incorporates both learning the distortion measure parameters and constraint-sensitive cluster assignments. In HMRF-KMEANS, the parameters of the distortion measure are learned iteratively as the clustering progresses, utilizing both unlabeled data and pairwise constraints. The parameters are modified to decrease the parameterized distance between violated must-linked constraints and increase it between violated cannot-link constraints, while allowing constraint violations if this allows significantly more cohesive clustering.

This section presents three examples of distortion functions and their parameterizations for use with HMRF-KMEANS: squared Euclidean distance, cosine distance and KL divergence. Through parameterization, each of these functions becomes adaptive in a semi-supervised clustering setting. Additionally, we explain how cluster-specific distortion measures can be used where each cluster is associated with individual distortion measure parameters. This modification permits clusters to have different shapes which may be appropriate for some datasets.

Once a distortion measure is chosen for a given domain, the functions \( f_{ML} \) and \( f_{CL} \), introduced in Sec. 2.2 for penalizing must-link and cannot-link constraint violations respectively, must be defined. These functions typically follow a functional form identical or similar to the corresponding distortion measure, and are chosen as follows:

\[
\begin{align*}
    f_{ML}(i,j) &= \varphi(i,j) \\
    f_{CL}(i,j) &= \varphi_{\text{max}} - \varphi(i,j)
\end{align*}
\]

where \( \varphi : X \times X \to \mathbb{R}^+ \) is a non-negative function that penalizes constraint violation, and \( \varphi_{\text{max}} \) is the maximum value of \( \varphi \) over any pair of points in the dataset. The function \( \varphi \) is chosen to correlate with the distortion measure, assigning higher penalties to violations of
must-link constraints between points that are distant with respect to the current parameter values of the distortion measure. Conversely, penalties for violated cannot-link constraints are higher for points that have low distance between them. With this formulation of the penalty functions, constraint violations lead to changes in the distortion measure parameters that attempt to mend the violations. The \( \varphi \) function for different clustering distortion measures is discussed in the following sections.

Accordingly, the potential function \( v(i, j) \) in Eq. (4) becomes:

\[
v(i, j) = \begin{cases} 
  w_{ij} \varphi(x_i, x_j) & \text{if } c_{ij} = 1 \text{ and } y_i \neq y_j \\
  0 & \text{if } c_{ij} = -1 \text{ and } y_i = y_j \\
  \end{cases}
\]

and the objective function for semi-supervised clustering in Eqn. (9) can be expressed as:

\[
J_{obj} = \sum_{x_i \in X} D(x_i, \mu_l) + \sum_{(x_i, x_j) \in C_{ML}} w_{ij} \varphi(x_i, x_j) \\
+ \sum_{(x_i, x_j) \in C_{CL}} w_{ij} \left( \varphi_{max} - \varphi(x_i, x_j) \right) - \log P(A) + \log \det(A) + \log Z + \log Z_0
\]

(15)

2.6.1 Parameterized Squared Euclidean Distance

Squared Euclidean distance is parameterized using a symmetric positive-definite matrix \( A \) as follows:

\[
D_{\text{euca}}(x_i, x_j) = \|x_i - x_j\|_A^2 = (x_i - x_j)^T A (x_i - x_j).
\]

(16)

This form of the parameterized squared Euclidean distance is equivalent to Mahalanobis distance with an arbitrary positive semidefinite weight matrix \( A \) in place of the covariance matrix, and it was previously used for semi-supervised clustering by Xing et al. (2003) and Bar-Hillel et al. (2003). Such a formulation can also be viewed as a projection of every instance \( x \) onto a space spanned by \( A^{1/2} \): \( x \rightarrow A^{1/2} x \).

To use parameterized squared Euclidean distance as the adaptive distortion measure for clustering, the \( \varphi \) function is defined as: \( \varphi(x_i, x_j) = D_{\text{euca}}(x_i, x_j) \). Using this definition along with Eqn. (15), the following objective function is obtained for semi-supervised clustering with adaptive squared Euclidean distance:

\[
J_{\text{euca}} = \sum_{x_i \in X} D_{\text{euca}}(x_i, \mu_l) + \sum_{(x_i, x_j) \in C_{ML}} w_{ij} D_{\text{euca}}(x_i, x_j) \\
+ \sum_{(x_i, x_j) \in C_{CL}} w_{ij} \left( D_{\text{euca}}^{\text{max}} - D_{\text{euca}}(x_i, x_j) \right) - \log P(A) + \log \det(A) + \log Z
\]

(17)

Note that the \( \log \det(A) \) term in Eqn. (17) corresponds to the \( \log Z_0 \) term: it is the closed-form normalizer for a Gaussian distribution with covariance matrix \( A \), which is the underlying cluster conditional probability distribution for parameterized squared Euclidean distance.
2.6.2 Parameterized Cosine Distance

Cosine distance can be parameterized using a symmetric positive-definite matrix $A$, which leads to the following distortion measure:

$$D_{\cos A}(x_i, x_j) = 1 - \frac{x_i^T A x_j}{\|x_i\|_A \|x_j\|_A}. \quad (18)$$

Because for realistic high-dimensional domains computing the full matrix $A$ would be computationally expensive, a diagonal matrix is considered in this case, such that $A = \text{diag}(a)$ is a vector of positive weights. Each weight can be viewed as the degree of relative importance of the corresponding attribute, similar to the inverse document frequency (IDF) weighting used for representing textual documents in vector space Baeza-Yates and Ribeiro-Neto (1999).

To use parameterized cosine distance as the adaptive distortion measure for clustering, the $\varphi$ function is defined as: $\varphi(x_i, x_j) = D_{\cos A}(x_i, x_j)$. Using this definition along with Eqn. (15), the following objective function is obtained for semi-supervised clustering with adaptive cosine distance:

$$J_{\cos A} = \sum_{x_i \in X} D_{\cos A}(x_i, \mu_l) + \sum_{(x_i, x_j) \in C_{ML}} w_{ij} D_{\cos A}(x_i, x_j) + \sum_{(x_i, x_j) \in C_{CL}} w_{ij} (1 - D_{\cos A}(x_i, x_j)) - \log P(A) + \log Z + \log Z_\Theta \quad (19)$$

2.6.3 Parameterized KL Divergence

In certain domains, data is described by probability distributions, e.g. text documents can be represented as probability distributions over words generated by a multinomial model (Pereira et al., 1993). KL divergence is a widely used distance measure for such data: $D_{KL}(x_i, x_j) = \sum_{m=1}^d x_{im} \log \frac{x_{im}}{x_{jm}}$, where $x_i$ and $x_j$ are probability distributions over $d$ events: $\sum_{m=1}^d x_{im} = \sum_{m=1}^d x_{jm} = 1$. In previous work, Cohn et al. (2003) parameterized KL divergence multiplying $m$-th component by a weight $\gamma_m$: $D_{KL}'(x_i, x_j) = \sum_{m=1}^d \gamma_m x_{im} \log \frac{x_{im}}{x_{jm}}$.

In our framework, KL distance is parameterized using a diagonal matrix $A$, where $A = \text{diag}(a)$ is a vector of positive weights. This parameterization of KL by $A$ converts it to I-divergence, a function that also belongs to the class of Bregman divergences (Banerjee et al., 2004). I-divergence has the form: $D_I(x_i, x_j) = \sum_{m=1}^d x_{im} \log \frac{x_{im}}{x_{jm}} - \sum_{m=1}^d (x_{im} - x_{jm})$, where $x_i$ and $x_j$ no longer need to be probability distributions but can be any non-negative vectors (for probability distributions, I-divergence and KL divergence are equivalent). The following parameterization of KL divergence is used:

$$D_{IA}(x_i, x_j) = \sum_{m=1}^d a_m x_{im} \log \frac{x_{im}}{x_{jm}} - \sum_{m=1}^d a_m (x_{im} - x_{jm}), \quad (20)$$
which can be interpreted as scaling every component of the original probability distribution by a weight contained in the corresponding component of \( A \), and then taking I-divergence between the transformed distributions.

For every distortion measure, the clustering framework described in Section 2.4 requires defining an appropriate constraint potential function that is symmetric, since the constraint pairs are unordered. To meet this requirement, a sum of weighted I-divergences from \( x_i \) and \( x_j \) to the mean vector \( \frac{x_i + x_j}{2} \) is used. This parameterized divergence to the mean, \( D_{IMA} \), is equivalent to Jensen-Shannon divergence (Cover and Thomas, 1991), the symmetric KL divergence to the mean, and is defined as follows:

\[
D_{IMA}(x_i, x_j) = \sum_{m=1}^{d} a_m (x_{im} \log \frac{2x_{im}}{x_{im} + x_{jm}} + x_{jm} \log \frac{2x_{jm}}{x_{im} + x_{jm}}).
\]

(21)

To use parameterized KL divergence as the adaptive distortion measure for clustering, the \( \varphi \) function is defined as:

\[
\varphi(x_i, x_j) = D_{IMA}(x_i, x_j).
\]

Using this definition along with Eqn. (15), the following objective function is obtained for semi-supervised clustering with adaptive KL distance:

\[
J_{IA} = \sum_{x_i \in X} D_{IA}(x_i, \mu_l) + \sum_{(x_i, x_j) \in C_{ML}} w_{ij} D_{IMA}(x_i, x_j) + \sum_{(x_i, x_j) \in C_{CL}} w_{ij} (D_{IMA}^{max} - D_{IMA}(x_i, x_j)) - \log P(A) + \log Z + \log Z_{\Theta}
\]

(22)

2.6.4 Cluster-specific Distortion Measures

While previous work on semi-supervised clustering has been constrained to learning a single distortion measure for the entire dataset, the HMRF-KMEANS model allows using multiple distortion measures with corresponding parameter sets \((A_1, \ldots, A_K)\) instead of a single measure with parameters \( A \). In the probabilistic framework described in Section 2.3, this corresponds to an individual generative model with a parameterized probability density \( p(\cdot | y_i, \Theta_i) \) used for \( y_i \)-th cluster. To incorporate this modification into Eqns. 15-22, each centroid distortion term \( D(x_i, \mu_l) \) must be replaced with a cluster-specific distortion term \( D_{Al}(x_i, \mu_l) \), and the prior term \( \log P(A) \) must be replaced with \( \sum_{i=1}^{K} \log P(A_i) \).

To incorporate individual distortion measures into the constraint violation terms, the penalty functions \( f_{ML}(\cdot, \cdot) \) and \( f_{CL}(\cdot, \cdot) \) must also become cluster-specific by relying on the parameters of individual clusters involved in the violations. Since a violated must-link constraint involves two clusters into which the constraint’s points are placed, the penalty must be calculated using both distortion measures. This can be achieved by taking the average of the penalties calculated for each cluster’s penalty function: \( f_{ML}(i, j) = \frac{1}{2} \varphi_{Al}(i, j) + \frac{1}{2} \varphi_{Aj}(i, j) \), where each \( \varphi_{Al}(i, j) \) is the penalty function for \( l_i \)-th cluster based on that cluster’s distortion measure \( D_{Al}(\cdot, \cdot) \). On the other hand, violated cannot-link constraints involve only a single cluster (one to which both points are assigned), and therefore the cannot-
link penalty function relies on the cluster’s distortion measure: \( f_{CL}(i, j) = \varphi_{A_{ij}}^{\text{max}} - \varphi_{A_{ij}}(i, j) \). Here \( \varphi_{A_{ij}}^{\text{max}} \) remains the maximum value of the penalty function over the entire dataset.

For those datasets where clusters are likely to have different shapes, using cluster-specific distortion measures provides a way to achieve this, while a single distortion measure suffices for other datasets where clusters have similar shape.

3. The HMRF-KMEANS Algorithm

Since the cluster assignments and the generative model parameters are unknown in a clustering setting, minimizing Eqn. (9) specifies an “incomplete-data problem”. A popular solution technique for such problems the is Expectation Maximization (EM) algorithm (Dempster et al., 1977). The K-Means algorithm (MacQueen, 1967) is known to be equivalent to the EM algorithm with hard clustering assignments, under certain assumptions (Kearns et al., 1997; Basu et al., 2002; Banerjee et al., 2004). This section describes a K-Means-type hard partitional clustering algorithm, HMRF-KMEANS, that finds a local minimum of the semi-supervised clustering objective function \( J_{\text{obj}} \) in Eqn. (9).

3.1 Normalizer Estimation

Before describing the details of the clustering algorithm, it is important to consider the normalizer components: the MRF normalizer \( \log Z \) and the distortion function normalizer \( \log Z_\Theta \) in Eqn. (9). It is computationally intractible to estimate the MRF normalizer in closed form in any non-trivial MRF, and approximate inference methods must be employed for computing it (Wainwright and Jordan, 2003).

Estimation of the distortion normalizer \( \log Z_\Theta \) depends on the distortion measure \( D \) used by the model. This chapter considers three parameterized distortion measures: parameterized squared Euclidean distance, parameterized cosine distance, and parameterized Kullback-Leibler (KL) divergence. For Euclidean distance, \( Z_\Theta \) can be estimated in closed form, and this estimation is performed while minimizing the clustering objective function \( J_{\text{obj}} \) in Eqn. (9). For the other distortion measures, estimating the distortion normalizer \( Z_\Theta \) cannot be performed in closed form, and approximate inference must be again used (Banerjee et al., 2003).

Since approximate methods can be very expensive computationally, two simplifying assumptions can be made: the MRF normalizer may be considered to be constant in the clustering process, and the distortion normalizer may be assumed constant for all distortion measures that do not provide its closed-form estimate. With these assumptions, the objective function \( J_{\text{obj}} \) in Eqn. (9) no longer exactly corresponds to a joint probability on a HMRF. However, minimizing this simplified objective has been shown to work well empirically (Bilenko et al., 2004; Basu et al., 2004b). However, if in some application it is important to preserve the semantics of the underlying joint probability model, then the normalizers \( Z \) and \( Z_\Theta \) must be estimated by approximation methods.

3.2 EM Framework

As discussed in Section 2.1, \( J_{\text{obj}} \) can be minimized by a K-Means-type iterative algorithm HMRF-KMEANS. The outline of the algorithm is presented in Figure 3. The basic idea
of HMRF-KMeans is as follows: the constraints are used to obtain a good initialization of the clustering. Then in the E-step, given the current cluster representatives, every data point is re-assigned to the cluster that minimizes its contribution to \( \mathcal{J}_{\text{obj}} \). In the M-step, the cluster representatives \( M = (\mu_1, \ldots, \mu_K) \) are re-estimated from the cluster assignments to minimize \( \mathcal{J}_{\text{obj}} \) for the current assignment. The clustering distortion measure, \( D \), is subsequently updated in the M-step to reduce the objective function by modifying the parameters of the distortion measure.

Note that this corresponds to the generalized EM algorithm (Neal and Hinton, 1998; Dempster et al., 1977), where the objective function is reduced but not necessarily minimized in the M-step. Effectively, the E-step minimizes \( \mathcal{J}_{\text{obj}} \) over cluster assignments \( Y \), the M-step \((A)\) minimizes \( \mathcal{J}_{\text{obj}} \) over cluster representatives \( M \), and the M-step \((B)\) reduces \( \mathcal{J}_{\text{obj}} \) over the parameters of the distortion measure \( D \). The E-step and the M-step are repeated till a specified convergence criterion is reached. The specific details of the E-step and M-step are discussed in the following sections.

**Algorithm: HMRF-KMeans**

**Input:** Set of data points \( X = \{x_i\}_{i=1}^N \), number of clusters \( K \), set of must-link constraints \( C_{\text{ML}} = \{(x_i, x_j)\} \), set of cannot-link constraints \( C_{\text{CL}} = \{(x_i, x_j)\} \), distortion measures \( \{D_h\}_{h=1}^K \), constraint violation costs \( W \).

**Output:** Disjoint \( K \)-partitioning \( \{X_h\}_{h=1}^K \) of \( X \) such that objective function \( \mathcal{J}_{\text{obj}} \) in Eqn.(9) is (locally) minimized.

**Method:**

1. Initialize the \( K \) clusters centroids \( \{\mu_h^{(0)}\}_{h=1}^K \), set \( t \leftarrow 0 \)
2. Repeat until convergence
   2a. **E-step:** Given \( \{\mu_h^{(t)}\}_{h=1}^K \), re-assign cluster labels \( \{y_i^{(t+1)}\}_{i=1}^N \) on the points \( \{x_i\}_{i=1}^N \) to minimize \( \mathcal{J}_{\text{obj}} \).
   2b. **M-step(A):** Given cluster labels \( \{y_i^{(t+1)}\}_{i=1}^N \), re-calculate cluster centroids \( \{\mu_h^{(t+1)}\}_{h=1}^K \) to minimize \( \mathcal{J}_{\text{obj}} \).
   2c. **M-step(B):** Re-estimate distortion measures \( \{D_h\}_{h=1}^K \) to reduce \( \mathcal{J}_{\text{obj}} \).
   2d. \( t \leftarrow t+1 \)

Figure 3: HMRF-KMeans algorithm

### 3.3 Initialization

Good initial centroids are essential for the success of partitional clustering algorithms such as K-Means. Good centroids are inferred from both the constraints and unlabeled data during initialization. For this, a two stage initialization process is used.

**Neighborhood inference:** At first, the transitive closure of the must-link constraints is taken to get connected components consisting of points connected by must-links. Let there be \( \lambda \) connected components, which are used to create \( \lambda \) neighborhoods. These correspond to the must-link neighborhoods in the MRF over the hidden cluster variables.

**Cluster selection:** The \( \lambda \) neighborhood sets produced in the first stage are used to initialize the HMRF-Means algorithm. If \( \lambda = K \), \( \lambda \) cluster centers are initialized with the centroids of all the \( \lambda \) neighborhood sets. If \( \lambda < K \), \( \lambda \) clusters are initialized from the neigh-
neighborhoods, and the remaining $K - \lambda$ clusters are initialized with points obtained by random perturbations of the global centroid of $X$. If $\lambda > K$, a weighted variant of farthest-first traversal (Hochbaum and Shmoys, 1985) is applied to the centroids of the $\lambda$ neighborhoods, where the weight of each centroid is proportional to the size of the corresponding neighborhood. In weighted farthest-first traversal, we maintain a set $S$ of traversed neighborhoods, and at every step pick the next neighborhood $N_x$ for which the centroid $x$ has the farthest weighted distance from the traversed set (where the weighted distance of a point $x$ distance from a set is $d(x, S) = \min_{y \in S} |N_x||S|D(x, y)$). The selected neighborhood is added to the traversed set, and selection is repeated until $K$ neighborhood centroids are selected. Weighted farthest-first traversal selects neighborhoods that are relatively far apart as well as large in size, and the chosen neighborhoods are set as the $K$ initial cluster centroids for HMRF-KMeans.

Overall, this two-stage initialization procedure is able to take into account both unlabeled and labeled data to obtain cluster representatives that provide a good initial partitioning of the dataset.

3.4 E-step

In the E-step, assignments of data points to clusters are updated using the current estimates of the cluster representatives. In the general unsupervised K-Means algorithm, there is no interaction between the cluster labels, and the E-step is a simple assignment of every point to the cluster representative that is nearest to it according to the clustering distortion measure. In contrast, the HMRF model incorporates interaction between the cluster labels defined by the random field over the hidden variables. As a result, computing the assignment of data points to cluster representatives to find the global minimum of the objective function, given the cluster centroids, is computationally intractable in any non-trivial HMRF model (Segal et al., 2003).

There exist several techniques for computing cluster assignments that approximate the optimal solution in this framework, e.g., iterated conditional modes (ICM) (Besag, 1986; Zhang et al., 2001), loopy belief propagation (Pearl, 1988; Segal et al., 2003), and linear programming relaxation (Kleinberg and Tardos, 1999). In this work, the ICM approach is followed, which is a greedy strategy to sequentially update the cluster assignment of each point, keeping the assignments for the other points fixed. The inexpensive greedy ICM algorithm has comparable performance to more expensive global approximation techniques like loopy belief propagation or linear programming relaxation, but is computationally more efficient (Bilenko and Basu, 2004). ICM performs sequential cluster assignment, in random order, for all the points. Each point $x_i$ is assigned to the cluster representative $\mu_h$ that minimizes the point’s contribution to the objective function $J_{obj}(x_i, \mu_h)$:

$$J_{obj}(x_i, \mu_h) = D(x_i, \mu_h) + \sum_{(x_i, x_j) \in C_M} w_{ij} \varphi(x_i, x_j)$$
$$+ \sum_{(x_i, x_j) \in C_L} w_{ij} (\varphi_{max} - \varphi(x_i, x_j)) - \log P(A),$$

(23)
where \( C_{im}^i \) and \( C_{cil}^i \) are the subsets of \( C_{ml} \) and \( C_{cl} \) respectively in which \( x_i \) appears in the constraints. The optimal assignment for every point minimizes the distortion between the point and its cluster representative (first term of \( J_{obj} \)) along with incurring a minimal penalty for constraint violations caused by this assignment (second and third terms of \( J_{obj} \)). After all points are assigned, they are randomly re-ordered, and the assignment process is repeated. This process proceeds until no point changes its cluster assignment between two successive iterations.

Overall, the assignment of points to clusters incorporates pairwise supervision by discouraging constraint violations proportionally to their severity, which guides the algorithm towards a desirable partitioning of the data.

### 3.5 M-step

The M-step of the algorithm consists of two parts: re-estimation of cluster centroids and an update of the distortion measure parameters.

#### 3.5.1 Centroid Re-estimation

In the first part of the M-step, the cluster centroids \( M \) are re-estimated from points currently assigned to them, to decrease the objective function \( J_{obj} \) in Eqn. (9). For Bregman divergences and cosine distance, the cluster representative calculated in the M-step of the EM algorithm is equivalent to the expectation value over the points in that cluster, which is equal to their arithmetic mean (Banerjee et al., 2004, 2003). Additionally, it has been experimentally demonstrated that for clustering with distribution-based measures, e.g., KL divergence, smoothing cluster representatives by a priori using a deterministic annealing schedule leads to considerable improvements (Dhillon and Guan, 2003). With smoothing controlled by a parameter \( \alpha \), each cluster representative \( \mu_h \) is estimated as follows when \( D_{IA} \) is the distortion measure:

\[
\mu_{h}^{(IA)} = \frac{1}{1 + \alpha} \left( \frac{1}{|X_h|} \sum_{x_i \in X_h} x_i + \alpha \frac{1}{n} \right) \quad (24)
\]

For directional measures, each cluster representative is the arithmetic mean projected onto unit sphere (Banerjee et al., 2003). Taking the distortion parameters into account, centroids are estimated as follows when \( D_{cosA} \) is the distortion measure:

\[
\frac{\mu_{h}^{(cosA)}}{\|\mu_{h}^{(cosA)}\|_A} = \frac{\sum_{x_i \in X_h} x_i}{\|\sum_{x_i \in X_h} x_i\|_A} \quad (25)
\]

#### 3.5.2 Update of Distortion Parameters

In the second part of the M-step, the parameters of the parameterized distortion measure are updated to decrease the objective function. In general, for parameterized Bregman divergences or directional distances with general parameter priors, it is difficult to attain a closed-form update for the parameters of the distortion measure that can minimize the ob-
Gradient descent provides an alternative avenue for learning the distortion measure parameters.

For squared Euclidean distance, a full parameter matrix $A$ is updated during gradient descent using the rule: $A = A + \eta \frac{\partial D_{\text{euc}}}{\partial A}$ (where $\eta$ is the learning rate). Using Eqn. (17), $\frac{\partial D_{\text{euc}}}{\partial A}$ can be expressed as:

$$
\frac{\partial D_{\text{euc}}}{\partial A} = \sum_{x_i \in X} \frac{\partial D_{\text{euc}}(x_i, \mu_{y_i})}{\partial A} + \sum_{(x_i, x_j) \in C_{\text{ML}} \text{ s.t. } y_i \neq y_j} w_{ij} \frac{\partial D_{\text{euc}}(x_i, x_j)}{\partial A} + \sum_{(x_i, x_j) \in C_{\text{CL}} \text{ s.t. } y_i = y_j} w_{ij} \left[ \frac{\partial \varphi_{\text{max}}}{\partial A} - \frac{\partial \varphi(x_i, x_j)}{\partial A} \right] - \frac{\partial \log P(A)}{\partial A} + \frac{\partial \log \det(A)}{\partial A}. 
$$

The gradient of the parameterized squared Euclidean distance is given by (Bilenko et al., 2004):

$$
\frac{\partial D_{\text{euc}}}{\partial A} = (x_i - x_j)(x_i - x_j)^T
$$

When Rayleigh priors are used for the set of parameters $A$, the partial derivative of the log-prior with respect to every individual parameter $a_m \in A$, $\frac{\partial \log P(A)}{\partial a_m}$, is given by:

$$
\frac{\partial \log P(A)}{\partial a_m} = \frac{1}{a_m} - \frac{a_m}{s^2}
$$

The gradient of the distortion normalizer $\log \det(A)$ term (Bilenko et al., 2004) is:

$$
\frac{\partial \log \det(A)}{\partial A} = 2A^{-1} - \text{diag}(A^{-1}).
$$

For parameterized cosine distance and KL divergence, a diagonal parameter matrix $A$ is considered, where $a = \text{diag}(A)$ is a vector of positive weights. During gradient descent, each weight $a_m$ is individually updated as: $a_m = a_m + \eta \frac{\partial \log P(A)}{\partial a_m}$ (where $\eta$ is the learning rate). Using Eqn. (15), $\frac{\partial \log P(A)}{\partial a_m}$ can be expressed as:

$$
\frac{\partial \log P(A)}{\partial a_m} = \sum_{x_i \in X} \frac{\partial \varphi(x_i, \mu_{y_i})}{\partial a_m} + \sum_{(x_i, x_j) \in C_{\text{ML}} \text{ s.t. } y_i \neq y_j} w_{ij} \frac{\partial \varphi_{\text{max}}}{\partial a_m} - \frac{\partial \varphi(x_i, x_j)}{\partial a_m} [\frac{\partial \varphi_{\text{max}}}{\partial a_m} - \frac{\partial \varphi(x_i, x_j)}{\partial a_m}] - \frac{\partial \log P(A)}{\partial a_m}
$$

1. For the specific case of parameterized squared Euclidean distance, a closed-form update of the parameters can be obtained (Bilenko et al., 2004).
Calculation of the gradient $\frac{\partial J_{\text{obj}}}{\partial a_m}$ for cosine distance and KL divergence, which are parameterized by a diagonal matrix $A$, needs the gradients of the corresponding distortion measures and constraint potential functions, which are:

$$
\frac{\partial D_{\text{cos}}(x_i, x_j)}{\partial a_m} = \frac{x_{im}x_{jm} \left\| x_i \right\| A \left\| x_j \right\| A - x_i^T A x_j x_{im}^2 x_{jm}^2 + x_{jm}^2 \left\| x_i \right\|^2 A}{\left\| x_i \right\|^2 A \left\| x_j \right\|^2 A},
$$

$$
\frac{\partial D_{\text{IA}}(x_i, x_j)}{\partial a_m} = x_{im} \log \frac{x_{im}}{x_{jm}} - (x_{im} - x_{jm}),
$$

$$
\frac{\partial D_{\text{IM}}(x_i, x_j)}{\partial a_m} = x_{im} \log \frac{2x_{im}}{x_{im} + x_{jm}} + x_{jm} \log \frac{2x_{jm}}{x_{im} + x_{jm}}.
$$

(30)

If cluster-specific distortion measures described in Section 2.6.4 are employed, updates (29)-(30) become individualized for each particular distortion metric $A_i$, and accordingly involve only points assigned to that cluster.

Overall, the distance learning step results in modifying the distortion measure so that data points that participate in violated must-link constraints are brought closer together, while points between which cannot-link constraints are not satisfied are pulled apart. This process leads to a transformed data space that facilitates partitioning of the unlabeled data, by attempting to mend the constraint violations as well as reflecting the natural variance in the data.

### 3.6 Convergence of HMRF-KMEANS

The HMRF-KMEANS algorithm alternates between updating the assignment of points to clusters, and updating the parameters. Since all updates ensure a decrease in the objective function, each iteration of HRMF-KMEANS monotonically decreases the objective function. Let us inspect each step in the update to ensure that this is indeed the case.

For analyzing the cluster assignment step, let us consider Eqn. (15). Each point $x_i$ moves to a new cluster $h$ only if the following component, contributed by the point $x_i$, is decreased with the move:

$$
D(x_i, \mu_h) + \sum_{(x_i, x_j) \in C_{ML}^{\text{ML}}} w_{ij} \varphi(x_i, x_j) + \sum_{(x_i, x_j) \in C_{CL}^{\text{CL}}} w_{ij} (\varphi^{\text{max}} - \varphi(x_i, x_j)) - \log P(A).
$$

Given a set of centroids and distortion parameters, the new cluster assignment of points will decrease $J_{\text{obj}}$ or keep it unchanged.

For analyzing the centroid re-estimation step, let us consider an equivalent form of Eqn. (15):
Each cluster centroid $\mu_h$ is re-estimated by taking the mean of the points in the partition $X_h$, which minimizes the component $\sum_{x_i \in X_h} D(x_i, \mu_h)$ of $J_{\text{obj}}$ in Eqn. (31) contributed by the partition $X_h$ (Banerjee et al., 2004, 2003). The constraint potential and the prior term in the objective function do not take a part in centroid re-estimation, because they are not explicit functions of the centroid. So, given the cluster assignments and the distortion parameters, $J_{\text{obj}}$ will decrease or remain the same in this step.

For the parameter estimation step, the gradient-descent updates of the parameters in Eqn. (29) decreases $J_{\text{obj}}$ or keeps it unchanged.

Hence the objective function decreases after every cluster assignment, centroid re-estimation and parameter re-estimation step. Now, note that the objective function is bounded below by a constant. Being the negative log-likelihood of a probabilistic model with the normalizer terms, $J_{\text{obj}}$ is bounded below by zero. Even without the normalizers, the objective function is bounded below by zero, since the distortion and potential terms are non-negative due to the fact that $A$ is positive definite. Since $J_{\text{obj}}$ is bounded below, and HMRF-KMEANS results in a decreasing sequence of objective function values, the value sequence must have an accumulation point. The accumulation point in this case will be a fixed point of $J_{\text{obj}}$ since neither updating the assignments or the parameters can further decrease the value of the objective function. As a result, the HMRF-KMEANS algorithm will converge to a fixed point of the objective. In practice, convergence can be determined if subsequent iterations of HMRF-KMEANS result in insignificant changes in $J_{\text{obj}}$.

4. Experiments

This section describes the experiments we performed to demonstrate the effectiveness of various aspects of HMRF-KMEANS.

4.1 Datasets

We ran experiments on both low-dimensional and high-dimensional datasets to evaluate the HMRF-KMEANS framework with different distortion measures. For the low-dimensional datasets, on which squared Euclidean distance was used as the distortion measure, we considered the following datasets:

- Three datasets from the UCI repository: Iris, Wine, and Ionosphere (Blake and Merz, 1998);
- The Protein dataset used by Xing et al. (2003) and Bar-Hillel et al. (2003);
- Randomly sampled subsets from the *Digits* and *Letters* handwritten character recognition datasets, also from the UCI repository. For *Digits* and *Letters*, we chose two sets of three classes: \{I, J, L\} from *Letters* and \{3, 8, 9\} from *Digits*, sampling 10% of the data points from the original datasets randomly. These classes were chosen since they represent difficult visual discrimination problems.

Table 1 summarizes the properties of the low-dimensional datasets: the number of instances, the number of dimensions, and the number of classes.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Iris</th>
<th>Wine</th>
<th>Ionosphere</th>
<th>Protein</th>
<th>Letters</th>
<th>Digits</th>
</tr>
</thead>
<tbody>
<tr>
<td>Instances</td>
<td>150</td>
<td>178</td>
<td>351</td>
<td>116</td>
<td>227</td>
<td>317</td>
</tr>
<tr>
<td>Dimensions</td>
<td>4</td>
<td>13</td>
<td>34</td>
<td>20</td>
<td>16</td>
<td>16</td>
</tr>
<tr>
<td>Classes</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>6</td>
<td>3</td>
<td>3</td>
</tr>
</tbody>
</table>

For the high-dimensional text data, we considered 3 datasets that have the characteristics of being sparse, high-dimensional, and having a small number of points compared to the dimensionality of the space. This is done for two reasons:

- When clustering sparse high-dimensional data, e.g., text documents represented using the vector space model, it is particularly difficult to cluster small datasets, as observed by Dhillon and Guan (2003). The purpose of performing experiments on these subsets is to scale down the sizes of the datasets for computational reasons but at the same time not scale down the difficulty of the tasks.

- Clustering small number of sparse high-dimensional data points is a likely scenario in realistic applications. For example, when clustering the search results in a web-search engine like Vivisimo\(^2\), typically the number of webpages that are being clustered is in the order of hundreds. However the dimensionality of the feature space, corresponding to the number of unique words in all the webpages, is in the order of thousands. Moreover, each webpage is sparse, since it contains only a small number of all the possible words. On such datasets, clustering algorithms can easily get stuck in local optima: in such cases it has been observed that there is little relocation of documents between clusters for most initializations, which leads to poor clustering quality after convergence of the algorithm (Dhillon and Guan, 2003). Supervision in the form of pairwise constraints is most beneficial in such cases and may significantly improve clustering quality.

We derived 3 datasets from the *20-Newsgroups* collection.\(^3\) This collection has messages harvested from 20 different Usenet newsgroups, 1000 messages from each newsgroup. From the original dataset, a reduced dataset was created by taking a random subsample of 100 documents from each of the 20 newsgroups. Three datasets were created by selecting 3

\(^2\) http://www.vivisimo.com
\(^3\) http://www.ai.mit.edu/people/jrennie/20Newsgroups
categories from the reduced collection. *News-Similar-3* consists of 3 newsgroups on similar topics (**comp.graphics**, **comp.os.ms-windows**, **comp.windows.x**) with significant overlap between clusters due to cross-posting. *News-Related-3* consists of 3 newsgroups on related topics (**talk.politics.misc**, **talk.politics.guns**, and **talk.politics.mideast**). *News-Different-3* consists of articles posted in 3 newsgroups that cover different topics (**alt.atheism**, **rec.sport.baseball**, **sci.space**) with well-separated clusters. All the text datasets were converted to the vector-space model by tokenization, stop-word removal, TF-IDF weighting, and removal of very high-frequency and low-frequency words, following the methodology of Dhillon and Modha (2001).

On the biological domain, we used a subset of the Eisen dataset (Eisen et al., 1998) **Eisen-4** having 258 genes from 4 underlying functional pathways: 48 from Chromatin Structure, 73 from Cytoskeleton, 38 from Glycolysis and 99 from Protein Degradation. Each gene is described by a vector of gene responses to particular experimental conditions. The dataset has been pre-processed so that every feature has zero mean and unit variance, making cosine distance equivalent to Pearson’s distance, the distortion measure traditionally used in this domain. Table 2 summarizes the properties of the high-dimensional datasets.

<table>
<thead>
<tr>
<th>Table 2: High-dimensional datasets used in experimental evaluation</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Instances</strong></td>
</tr>
<tr>
<td><strong>Dimensions</strong></td>
</tr>
<tr>
<td><strong>Classes</strong></td>
</tr>
</tbody>
</table>

### 4.2 Clustering Evaluation

We used normalized mutual information (NMI) as our clustering evaluation measure. NMI is an external clustering validation metric that estimates the quality of the clustering with respect to a given underlying class labeling of the data: it measures how closely the clustering algorithm could reconstruct the underlying label distribution in the data (Strehl et al., 2000). If $\hat{Y}$ is the random variable denoting the cluster assignments of the points and $Y$ is the random variable denoting the underlying class labels on the points (Banerjee et al., 2003), then the NMI measure is defined as:

$$NMI = \frac{I(Y; \hat{Y})}{(H(Y) + H(\hat{Y}))/2}$$

(32)

where $I(X;Y) = H(X) - H(X|Y)$ is the mutual information between the random variables $X$ and $Y$, $H(X)$ is the Shannon entropy of $X$, and $H(X|Y)$ is the conditional entropy of $X$ given $Y$ (Cover and Thomas, 1991). NMI effectively measures the amount of statistical information shared by the random variables representing the cluster assignments and the user-labeled class assignments of the data points. Though various clustering evaluation measures have been used in the literature, NMI and it’s variants have become popular lately among clustering practitioners (Dom, 2001; Fern and Brodley, 2003; Meila, 2003).
4.3 Methodology

We generated learning curves using two-fold cross-validation performed over 20 runs on each dataset. In every trial, 50% of the dataset was set aside as the training fold. Every point on the learning curve corresponds to the number of constraints on pairs of data points from the training fold. These constraints are obtained by randomly selecting pairs of points from the training fold and creating must-link or cannot-link constraints depending on whether the underlying classes of the two points are same or different. Unit constraint costs $W$ were used for all constraints (original and inferred), since the datasets did not provide individual weights for the constraints. We set the gradient step size $\eta$ for learning the distortion measure parameters and the Rayleigh prior width parameter $s$ based on pilot studies. The gradient step size was set to $\eta = 100.0$ for clustering with weighted cosine distance $D_{\cos_a}$ and $\eta = 0.08$ for weighted I divergence $D_{I_a}$. The Rayleigh prior width parameter was set to $s = 1$. In a real-life setting, the free parameters of the algorithm could be tuned using cross-validation with a hold-out set. The clustering algorithm was run on the whole dataset, but NMI was calculated using points in the test fold. We performed two sets of experiments with the HMRF-KMEANS algorithm, described in detail in the following subsections.

4.3.1 First experiment

We performed sensitivity studies with HMRF-KMEANS to study the effectiveness of each component of the algorithm. We compared the proposed HMRF-KMEANS algorithm with three ablations, as well as with unsupervised K-Means clustering. The following variants were compared for distortion measures $D_{\cos_a}$, $D_{I_a}$ and $D_{euc_A}$:

- KMEANS-C-D-R is the complete HMRF-KMEANS algorithm that incorporates constraints in cluster assignments (C) as described in Section 3.4, includes distance learning (D) as described in Section 3.5, and also performs regularization (R) using a Rayleigh prior as described in Section 2.5;
- KMEANS-C-D is the first ablation of HMRF-KMEANS that includes all components except for regularization of distortion measure parameters;
- KMEANS-C is an ablation of HMRF-KMEANS that uses pairwise supervision for initialization and cluster assignments, but does not perform distortion measure learning;
- KMEANS is the unsupervised K-Means algorithm.

The goal of these experiments was to evaluate the utility of each component of our framework and identify settings in which particular components are beneficial.

4.3.2 Second experiment

In the second experiment, we compared the performance of variations of the distortion measure parameterization. We compared three variants of the KMEANS-C-D ablation from the first experiment, all of them based on distortion measure $D_{euc_A}$:

- KMEANS-C-D, which is same as HMRF-KMEANS and involves both constraints in assignments and metric learning; a single set of distortion measure parameters corresponding to a diagonal matrix is used for all clusters;
• KMEANS-C-DM, which is HMRF-KMEANS with multiple metrics (M) for different clusters: cluster-specific distortion measure parameters effectively allow clusters of different shapes;

• KMEANS-C-DF, which corresponds to HMRF-KMEANS where a single metric parameterized by a full matrix (F) is used for all clusters: instead of feature weighting, distortion measure learning corresponds to feature extraction.

In these experiments, our goal was to evaluate whether employing cluster-specific distortion measure parameters or the full matrix parameterization leads to improvements in clustering quality.

4.4 Results and Discussion

4.4.1 First experiment

Low-dimensional datasets: Figures 4-9 show learning curves for the ablation experiments on the six low-dimensional datasets. Across all datasets, the overall HMRF-KMEANS approach without regularization (KMEANS-C-D) outperforms the constraints-only ablation (KMEANS-C) and unsupervised KMeans, which indicates that employing constraints in the cluster assignment step of the algorithm and for learning distortion measure parameters leads to significant improvements in clustering quality. Since the performance of KMEANS-C-D-R is not substantially different from KMEANS-C-D, we conclude that regularization does not lead to performance improvements on low-dimensional datasets. This can be explained by the fact that the number of distortion measure parameters is small for low-dimensional domains, while estimates obtained from data do not have high variance, and therefore incorporating a prior in the probabilistic model is not necessary.

For the Wine, Protein, and Digits-389 datasets, the difference between ablations that utilize metric learning (KMEANS-C-D-R and KMEANS-C-D) and those that do not (KMEANS-C and KMEANS) at the beginning of the learning curve indicates that even in the absence of constraints, weighting features by their variance (essentially using unsupervised Mahalanobis distance) improves clustering accuracy. For the Wine dataset, additional constraints provide an insubstantial improvement in cluster quality on this dataset, which shows that meaningful feature weights are obtained from scaling by variance using just the unlabeled data.

Some of the metric learning curves display a characteristic “dip”, where clustering accuracy decreases as a few initial constraints are provided, but after a certain point starts to increase and eventually rises above the initial point on the learning curve. We conjecture that this phenomenon is due to the fact that metric parameters learned using too few constraints are unreliable, and a significant number of constraints is required by the metric learning mechanism to estimate parameters accurately. Overall, we can see that when both constraints and distortion measure learning are utilized, the unified approach benefits from the individual strengths of the two methods, as can be seen from the KMEANS-C-D results.

High-dimensional datasets: Figures 10-14 present the results for the ablation experiments where weighted cosine similarity $D_{\cos}$ was used as the distortion measure, while Figures 11-15 summarize experiments where weighted I-divergence $D_{I_a}$ was used. Figure 16 contains the results on the Eisen-4 dataset.
Figure 4: Results for $D_{euclidean}$ on Iris

Figure 5: Results for $D_{euclidean}$ on Wine

Figure 6: Results for $D_{euclidean}$ on Protein

Figure 7: Results for $D_{euclidean}$ on Ionosphere

Figure 8: Results for $D_{euclidean}$ on Digits-389

Figure 9: Results for $D_{euclidean}$ on Letters-IJL
Figure 10: Results for $D_{\cos_a}$ on News-Different-3

Figure 11: Results for $D_{I_a}$ on News-Different-3

Figure 12: Results for $D_{\cos_a}$ on News-Related-3

Figure 13: Results for $D_{I_a}$ on News-Related-3

Figure 14: Results for $D_{\cos_a}$ on News-Similar-3

Figure 15: Results for $D_{I_a}$ on News-Similar-3
As the results demonstrate, the full HMRF-KMEANS algorithm with regularization (KMEANS-C-D-R) outperforms the unsupervised K-Means baseline as well as the ablated versions of the algorithm for both distortion measures $D_{\cos}$ and $D_{I_a}$. As can be seen from results for zero pairwise constraints in Figs. 10-16, distortion measure learning is beneficial even in the absence of any pairwise constraints, since it allows capturing the relative importance of the different attributes in the unsupervised data. In the absence of supervised data or when no constraints are violated, distance learning attempts to minimize the objective function by adjusting the weights given the distortion between the unsupervised datapoints and their corresponding cluster representatives.

For high-dimensional datasets, regularization is clearly beneficial to performance, as can be seen from the improved performance of KMEANS-C-D-R over KMEANS-C-D on all datasets. This can be explained by the fact that the number of distortion measure parameters is large for high-dimensional datasets, and therefore algorithm-based estimates of parameters tend to be unreliable unless they incorporate a prior.

Overall, our results show that the HMRF-KMEANS algorithm effectively incorporates labeled and unlabeled data in all its stages, each of which improves the clustering quality.

4.4.2 Second experiment

Figs.17-22 show the results for the six low-dimensional datasets that compare performances of full vs. diagonal distortion measure parameterizations, as well as singular vs. cluster-specific parameterizations. As can be seen from results, both full matrix parameterization and individual metrics for each cluster can lead to significant improvements in clustering quality. However, the relative usefulness of these two techniques varies between the datasets, e.g., multiple metrics are particularly beneficial for Letters-IJL and Digits-389 datasets, while switching from a diagonal to a full weight matrix leads to large improvements on Wine, Ionosphere, and Letters-IJL. These results can be explained by the fact that the relative success of the two techniques depends on the properties of a particular dataset: using a full weight matrix helps when the attributes are highly correlated, while multiple
Probabilistic Semi-supervised Clustering with Constraints

Figure 17: *Iris*: distortion measure learning

Figure 18: *Wine*: distortion measure learning

Figure 19: *Protein*: distortion measure learning

Figure 20: *Ionosphere*: distortion measure learning

Figure 21: *Digits-389*: distortion measure learning

Figure 22: *Letters-IJL*: distortion measure learning
metrics lead to improvements when clusters in the dataset have different shapes or lie in different subspaces of the original space.

5. Related work

A related unified model for semi-supervised clustering with constraints was recently proposed by Segal et al. (2003). Their model is a unified Markov network that combines a binary Markov network derived from pairwise protein interaction data and a Naive Bayes Markov network modeling gene expression data. Our proposed HMRF framework is more general than this formulation, since it works with a broad class of clustering distortion measures, including Bregman divergences like KL divergence and squared Euclidean distance and directional similarity measures. In contrast, the formulation of Segal et al. considers only a Gaussian cluster conditional probability distribution, which corresponds to having Mahalanobis distance as the underlying clustering distance measure. Additionally, the HMRF-KMeans algorithm performs distance learning in the unified framework, which is not done in the Markov Network model.

The HMRF-KMeans algorithm proposed in this paper is related to the EM algorithm for HMRF model-fitting proposed by Zhang et al. (2001). However, HMRF-KMeans performs an additional step of distance learning in the M-step, which is not considered in the HMRF-EM algorithm. The discussion of the HMRF-EM algorithm was also restricted only to Gaussian conditional distributions, which has been generalized in our formulation.

There has been other research in semi-supervised clustering focusing individually on either constraint-based or distance-based semi-supervised clustering. COP-KMeans is a constraint-based clustering algorithm that has a heuristically motivated objective function (Wagstaff et al., 2001). Our method, on the other hand, has an underlying probabilistic model based on Hidden Markov Random Fields. Bansal et al. (2002), Blum et al. (2004) and Charikar et al. (2003) also propose frameworks for pairwise constrained clustering, but their models perform clustering using only the constraints; in comparison, HMRF-KMeans uses both constraints and an underlying distortion measure between the points for clustering. In settings where supervision is limited, the two-phase active learning algorithm for selecting informative pairwise constraints recently proposed by Basu et al. (2004a) can be incorporated in the HMRF-KMeans framework. Other recent research on constrained clustering includes variational techniques for constrained clustering using a graphical model (Hiu et al., 2005), model-level constraints to uncover multiple constraints in a dataset (Gondek et al., 2005), and feasibility studies for clustering under different types of constraints (Davidson and Ravi, 2005).

In recent work on distance-based semi-supervised clustering with pairwise constraints, Cohn et al. (2003) used gradient descent for weighted Jensen-Shannon divergence in the context of EM clustering. Xing et al. (2003) utilized a combination of gradient descent and iterative projections to learn a Mahalanobis distance for K-Means clustering. The Redundant Component Analysis (RCA) algorithm used only must-link constraints to learn a Mahalanobis distance using convex optimization (Bar-Hillel et al., 2003). Other methods include training a string-edit distance using Expectation Maximization (EM) (Bilenko and Mooney, 2003), modification of the squared Euclidean distance using the shortest-path algorithm (Klein et al., 2002), learning a margin-based clustering distortion measure using
boosting (Hertz et al., 2004), and learning a distance metric transformation that is globally linear but locally non-linear (Chang and Yeung, 2004). Spectral learning is another recent method that utilizes supervision to transform the clustering distance measure using spectral methods (Kamvar et al., 2003). All these distance learning techniques for clustering train the distance measure first using only supervised data, and then perform clustering on the unsupervised data. In contrast, our method integrates distance learning with the clustering process and utilizes both supervised and unsupervised data to learn the distortion measure.

6. Future Work

We have presented the general probabilistic framework for incorporating pairwise supervision into a prototype-based clustering algorithm, as well as two instantiations of that framework for particular distortion measures. There are several open issues that would be interesting to explore in future work.

Investigating alternative approaches to training distortion measures in the M-step of our algorithm may lead to improved performance of the algorithm. Our initial results as well as other recent work on distance learning for clustering (Klein et al., 2002; Bilenko et al., 2004; Bar-Hillel et al., 2003; Xing et al., 2003) suggest that transforming the data space can be highly beneficial for clustering quality. Therefore, we conjecture that developing alternative feature selection or feature extraction approaches, which perform other types of data space transformation using supervised data, is a promising direction for future work.

The weighted farthest-first algorithm for cluster initialization that we have described in Section 3.3 has proven itself very useful. We intend to explore theoretical implications of this initialization algorithm in the HMRF model, as well as develop alternative techniques that utilize both labeled and unlabeled data for initializing cluster representatives.

An important practical issue for applying HMRF-KMEANS in real domains is model selection. For semi-supervised clustering with constraints, the key model selection issue is one of choosing the right number of clusters. One can consider using a traditional model selection criterion suitable for the supervised setting, or perform model selection by cross-validation. An alternative is to perform model-selection using bounds on the test-set error-rate such that valuable supervision data is saved for learning. The PAC-MDL bounds provide such a tool that has been successfully applied to model selection for clustering (Banerjee et al., 2003), and can be readily extended to the semi-supervised clustering setting.

Clustering with HMRF-KMEANS produces linear cluster boundaries (created by the Voronoi tesselation of the space) in the input data space. Moreover, HMRF-KMEANS is applicable only to vector-based data, and cannot work with input provided in the form of a graph. In recent work, Kulis et al. (2005) proposed an approach for obtaining non-linear cluster boundaries by mapping the data using a non-linear transform to a kernel space and performing semi-supervised kernel-based clustering in that space. Their approach relies on using squared Euclidean distance as the clustering distortion measure, and can be used to perform constrained clustering for graph-based input data. An interesting area of future work would be to extend HMRF-KMEANS so that such semi-supervised graph-based clustering using the kernel approach can be applied to other clustering distortion measures apart from squared Euclidean distance.
A model has been recently proposed for overlapping clustering of data (Banerjee et al., 2005), where a data instance can belong simultaneously to multiple clusters, in contrast to HMRF-KMeans, where an instance is assigned only to one cluster. Such a model is very useful in domains like biology (where a gene can belong to multiple processes) or movie clustering for recommender systems (where a movie can belong to multiple genres). In such a model, it would be useful to be able to incorporate prior knowledge, e.g., the DIP data for overlapping gene clustering. We would like to work on a model for semi-supervised overlapping clustering in the future.

7. Conclusions

We have introduced a theoretically motivated framework for semi-supervised clustering that employs Hidden Random Markov Fields (HMRFs) to utilize both pairwise constraints and unlabeled data in the clustering process. The framework can be used with a number of distortion measures, including Bregman divergences and directional measures, and it accommodates trainable measures that can be adapted to specific datasets. We introduced the HMRF-KMeans algorithm that performs clustering in this framework and incorporates supervision in the form of pairwise constraints in all stages of the clustering algorithm: initialization, cluster assignment, and distortion measure parameter estimation. We presented three instantiations of the algorithm based on three particular popular distortion measures: Euclidean distance, KL divergence, and cosine similarity. Experimental evaluation has shown that the algorithm derived from the HMRF framework leads to improved cluster quality on realistic low- and high-dimensional datasets over unsupervised clustering and ablations of the proposed approach.

Acknowledgments

We would like to thank Srujana Merugu and Arindam Banerjee for insightful comments, and Joachim Buhmann for alerting us to the importance of considering parameter priors. This research was supported by the National Science Foundation under grants IIS-0117308, ITR: IIS-0325116, and CISE Research Infrastructure Grant EIA-0303609, and by Faculty Fellowships from IBM Corporation and Google Inc.

References


