Learning Kernels for Semi-Supervised Clustering

Bojun Yan
George Mason University, USA

Carlotta Domeniconi
George Mason University, USA

INTRODUCTION

As a recent emerging technique, semi-supervised clustering has attracted significant research interest. Compared to traditional clustering algorithms, which only use unlabeled data, semi-supervised clustering employs both unlabeled and supervised data to obtain a partitioning that conforms more closely to the user's preferences. Several recent papers have discussed this problem (Cohn, Caruana, & McCallum, 2003; Bar-Hillel, Hertz, Shental, & Weinshall, 2003; Xing, Ng, Jordan, & Russell, 2003; Basu, Bilenko, & Mooney, 2004; Kulis, Dhillon, & Mooney, 2005).

In semi-supervised clustering, limited supervision is provided as input. The supervision can have the form of labeled data or pairwise constraints. In many applications it is natural to assume that pairwise constraints are available (Bar-Hillel, Hertz, Shental, & Weinshall, 2003; Wagstaff, Cardie, Rogers, & Schroedl, 2001). For example, in protein interaction and gene expression data (Segal, Wang, & Koller, 2003), pairwise constraints can be derived from the background domain knowledge. Similarly, in information and image retrieval, it is easy for the user to provide feedback concerning a qualitative measure of similarity or dissimilarity between pairs of objects. Thus, in these cases, although class labels may be unknown, a user can still specify whether pairs of points belong to the same cluster (Must-Link) or to different ones (Cannot-Link). Furthermore, a set of classified points implies an equivalent set of pairwise constraints, but not vice versa. Recently, a kernel method for semi-supervised clustering has been introduced (Kulis, Dhillon, & Mooney, 2005). This technique extends semi-supervised clustering to a kernel space, thus enabling the discovery of clusters with non-linear boundaries in input space. While a powerful technique, the applicability of a kernel-based semi-supervised clustering approach is limited in practice, due to the critical settings of kernel's parameters. In fact, the chosen parameter values can largely affect the quality of the results. While solutions have been proposed in supervised learning to estimate the optimal kernel's parameters, the problem presents open challenges when no labeled data are provided, and all we have available is a set of pairwise constraints.

BACKGROUND

In the context of supervised learning, the work in (Chapelle & Vapnik) considers the problem of automatically tuning multiple parameters for a support vector machine. This is achieved by minimizing the estimated generalization error achieved by means of a gradient descent approach over the set of parameters. In (Wang, Xu, Lu, & Zhang, 2002), a Fisher discriminant rule is used to estimate the optimal spread parameter of a Gaussian kernel. The authors in (Huang, Yuen, Chen & Lai, 2004) propose a new criterion to address the selection of kernel's parameters within a kernel Fisher discriminant analysis framework for face recognition. A new formulation is derived to optimize the parameters of a Gaussian kernel based on a gradient descent algorithm. This research makes use of labeled data to address classification problems. In contrast, the approach we discuss in this chapter optimizes kernel's parameters based on unlabeled data and pairwise constraints, and aims at solving clustering problems. In the context of semi-supervised clustering, (Cohn, Caruana, & McCallum, 2003) uses gradient descent combined with a weighted Jensen-Shannon divergence for EM clustering. (Bar-Hillel, Hertz, Shental, & Weinshall, 2003) proposes a Redundant Component Analysis (RCA) algorithm that uses only must-link constraints to learn a Mahalanobis distance. (Xing, Ng, Jordan, & Russell, 2003) utilizes both must-link and cannot-link constraints

to formulate a convex optimization problem which is local-minima-free. (Segal, Wang, & Koller, 2003) proposes a unified Markov network with constraints. (Basu, Bilenko, & Mooney, 2004) introduces a more general HMRF framework, that works with different clustering distortion measures, including Bregman divergences and directional similarity measures. All these techniques use the given constraints and an underlying (linear) distance metric for clustering points in input space. (Kulis, Dhillon, & Mooney, 2005) extends the semi-supervised clustering framework to a non-linear kernel space. However, the setting of the kernel's parameter is left to manual tuning, and the chosen value can largely affect the results. The selection of kernel's parameters is a critical and open problem, which has been the driving force behind our research work.

MAIN FOCUS

In kernel-based learning algorithms, a kernel function $K(x_i, x_j)$ allows the calculation of dot products in feature space without knowing explicitly the mapping function. It is important that the kernel function in use conforms to the learning target. For classification, the distribution of data in feature space should be correlated to the label distribution. Similarly, in semi-supervised clustering, one wishes to learn a kernel that maps pairs of points subject to a must-link constraint close to each other in feature space, and maps points subject to a cannot-link constraint far apart in feature space. The authors in (Cristianini, Shawe-Taylor, & Elisseeff) introduce the concept of kernel alignment to measure the correlation between the groups of data in feature space and the labeling to be learned. In (Wang, Xu, Lu & Zhang), a Fisher discriminant rule is used to estimate the optimal spread parameter of a Gaussian kernel. The selection of kernel's parameters is indeed a critical problem. For example, empirical results in the literature have shown that the value of the spread parameter σ of a Gaussian kernel can strongly affect the generalization performance of an SVM. Values of σ which are too small or too large lead to poor generalization capabilities. When $\sigma \to 0$, the kernel matrix becomes the identity matrix. In this case, the resulting optimization problem gives Lagrangians which are all 1s, and therefore every point becomes a support vector. On the other hand, when $\sigma \to \infty$, the kernel matrix has entries all equal to 1, and thus each point in feature space is maximally similar to each other. In both cases, the machine will generalize very poorly. The problem of setting kernel's parameters, and of finding in general a proper mapping in feature space, is even more difficult when no labeled data are provided, and all we have available is a set of pairwise constraints. In our research we utilize the given constraints to derive an optimization criterion to automatically estimate the optimal kernel's parameters. Our approach integrates the constraints into the clustering objective function, and optimizes the kernel's parameters iteratively while discovering the clustering structure. Specifically, we steer the search for optimal parameter values by measuring the amount of must-link and cannot-link constraint violations in feature space. Following the method proposed in (Basu, Bilenko, & Mooney, 2004; Bilenko, Basu, & Mooney), we scale the penalty terms by the distances of points, that violate the constraints, in feature space. That is, for violation of a must-link constraint (x_i, x_j), the larger the distance between the two points x_i and x_j in feature space, the larger the penalty; for violation of a cannot-link constraint (x_i, x_j), the smaller the distance between the two points x_i and x_j in feature space, the larger the penalty. Considering the Gaussian kernel function

$$K(x_i, x_j) = \exp\left(-\left\|x_i - x_j\right\|/(2\sigma^2)\right)$$

and adding the constraint

$$\sum_{x_i \in X} \left\|\phi(x_i) - \phi(x_r)\right\|^2 \geq Const$$

(where x_r is a point randomly selected from data set X) to avoid the trivial minimization of the objective function $J_{kernel-obj}$, we obtain the following function:

$$J_{kernel-obj} = \sum_{c=1}^{k} \sum_{x_i, x_j \in \pi_c} \frac{1 - K(x_i, x_j)}{|\pi_c|} + \sum_{(x_i, x_j) \in ML, l_i \neq l_j} 2w_{ij}(1 - K(x_i, x_j)) + \sum_{(x_i, x_j) \in CL, l_i = l_j} 2\overline{w}_{ij}(K(x_i, x_j) - K(x', x'')) - \left(\sum_{x_i \in X} 2(1 - K(x_i, x_r)) - Const\right)$$

where LM is the set of must-link constraints, LC is the set of cannot-link constraints, π_c represents the C^{th} cluster, x' and x'' are the farthest points in feature space, w_{ij} and $\overline{w}_{ij}$ are the penalty costs for violating a must-link and a cannot-link constraints respectively, and l_i represents the cluster label of x_i. The resulting minimization problem is non-convex.