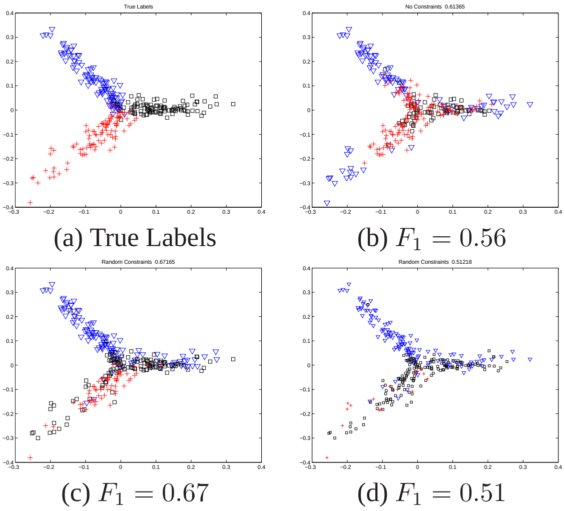
Active Query Selection for Semi-supervised Clustering

Pavan Kumar Mallapragada, Rong Jin and Anil K. Jain
Department of Computer Science and Engineering
Michigan State University, East Lansing, MI 48823
{pavanm,rongjin,jain}@cse.msu.edu

Abstract

Semi-supervised clustering allows a user to specify available prior knowledge about the data to improve the clustering performance. A common way to express this information is in the form of pair-wise constraints. A number of studies have shown that, in general, these constraints improve the resulting data partition. However, the choice of constraints is critical since improperly chosen constraints might actually degrade the clustering performance. We focus on constraint (also known as query) selection for improving the performance of semi-supervised clustering algorithms. We present an active query selection mechanism, where the queries are selected using a min-max criterion. Experimental results on a variety of datasets, using MPCK-means as the underlying semi-clustering algorithm, demonstrate the superior performance of the proposed query selection procedure.

1 Introduction



(a) True Labels (b) $F_1 = 0.56$

(c) $F_1 = 0.67$ (d) $F_1 = 0.51$

Figure 1. Illustration of constraint-based clustering. (a) 2-D projection of the Diff-300 dataset [2] using PCA, with true labels (b) K-means Clustering ($K = 3$) without any constraints, (c) & (d) two different clusterings with 100 pairwise constraints selected randomly. F_1 statistic [1] indicates the clustering quality.

The goal of clustering or unsupervised learning is to partition n objects represented as points in d dimensions. It is well-known that this problem is very difficult and considered to be ill-posed [7]. Any additional user-specified information should help in guiding the clustering algorithm towards a better solution. Semi-supervised clustering allows incorporation of "side-information" into the clustering algorithm, which is usually specified as constraints[1] of the form: should the u-th and the v-th objects in the data be put in the same cluster? The answer to this query can either be "yes" (a must-link query) or "no" (a must-not link query). Fig. 1 shows how introducing 100 randomly selected pairwise constraints improves the performance (Fig. 1(c)) of K-means clustering (Fig. 1(b)). Semi-supervised clustering algorithms focus on how to utilize the constraints effectively to infer the cluster labels. However, the constraints may not be always available a priori, but an oracle (user) may be available to provide the constraints, as needed by the algorithm. This scenario, where the system queries the oracle to obtain information relevant to learning is called active learning [5]. In an active learning framework, one aims to obtain a better partition of the data with minimal number of queries. Davidson et al. [4] and Wagstaff [10] show that when queries are not selected properly, the semi-supervised learning

[1] The constraints are referred to as the queries in active learning terminology.

This research was partially supported by ONR Grant No. N000140710225

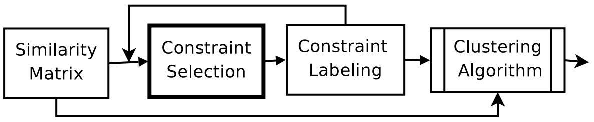


Figure 2. Active query selection.

degrades the clustering performance. Fig. 1(d) shows an example where the addition of 100 randomly chosen constraints actually degrades the performance of K-means (Fig. 1(b)). Thus, query selection is an important problem in semi-supervised clustering.

An active query selection algorithm using the “farthest-first” strategy, was proposed by Basu et al. [1]. We refer to this algorithm as the Farthest First Query Selection (FFQS) algorithm. The FFQS algorithm has two phases: `Explore` and `Consolidate`. The `Explore` phase selects must-not link constraints such that they result in at least one seed point per cluster. We call this set of points as *skeleton* of the clusters. A preference to must-not link queries is given by selecting the farthest point from the existing skeleton. The `Explore` phase continues until K points are found such that there is a must-not link query between any pair from the K points, which is then followed by the `Consolidate` phase. The `Consolidate` phase randomly selects the points not included in the skeleton (non-skeletal points), and queries them against each point in the skeleton, until a must-link query is obtained.

In this paper, we propose an algorithm for active query selection based on the min-max criterion, which significantly improves the `Consolidate` phase of the FFQS algorithm. The block diagram of the proposed approach is shown in Fig. 1. Given any semi-supervised clustering algorithm, the proposed query selection scheme utilizes the pairwise similarity to determine an optimal set of queries. The cluster labels obtained at each iteration of active query selection may also be used in selecting the queries. Using the well known MPCK-means (Metric Pairwise Constrained K-means) semi-supervised learning algorithm [3], we show that the proposed approach is better than random query selection and the FFQS algorithm.

2 Min-max strategy for query selection

Let $\mathcal{X} = \{\mathbf{x}_i\}_{i=1}^n$ denote n objects to be clustered into K clusters. Let $S = [s_{ij}]$ be the $n \times n$ real symmetric similarity matrix, where $s_{ij} \geq 0$ denotes the similarity between $\mathbf{x}_i$ and $\mathbf{x}_j$. We adopt the general framework of `Explore-Consolidate` from the FFQS algorithm. Given the identified skeleton points, the key question that we address is *how to actively identify query points during the consolidate phase?* Using the `Explore` algorithm [1], we first find a set of points $\mathcal{X}_s \subset \mathcal{X}$ such that it contains at least one point from each of the K clusters. We refer to $\mathcal{X}_s$ as the skeleton of the data clusters.

- For all $\mathbf{x}_i \in (\mathcal{X} - \mathcal{X}_s)$, compute their largest similarities to skeleton using Eq (1).
- Select $\mathbf{q}$, the most uncertain $\mathbf{x}_i$ according to the min-max criterion in Eq (2).
- Select one representative point per cluster $\mathbf{u}_k$, $k = 1, \cdots, K$, in the skeleton such that $\mathbf{u}_k$ is closest to $\mathbf{q}$ in cluster k.
- Sort $\mathbf{u}_k, k = 1, \cdots, K$ in descending order of similarities to $\mathbf{q}$.
- For each $\mathbf{u}_k$
    - Seek answer to the query ($\mathbf{q}$, u_k).
    - If the query is must-link, go to step 1. Otherwise, continue to next $\mathbf{u}_k$.
- Update the skeleton by including $\mathbf{q}$ in it.

Figure 3. The proposed min-max algorithm for query selection.

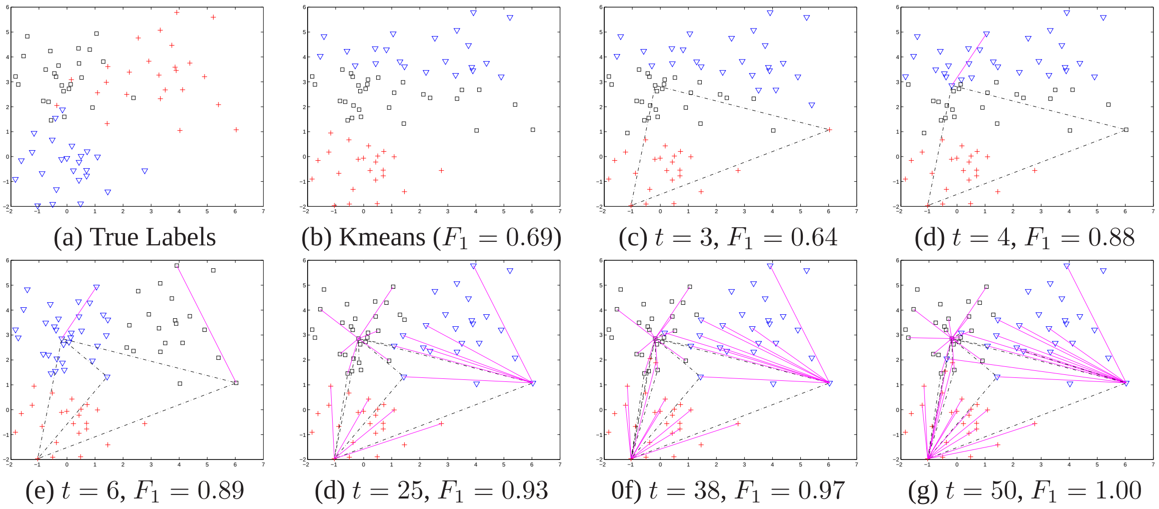
We assume that similarity measurement can be used to estimate the probability of two points to be in the same cluster. Hence, analogous to the nearest-neighbor method, the cluster label for a given data point $\mathbf{x}_i$ is decided mainly by the cluster label of the closest skeleton point. Let $P(\mathcal{X}_s, \mathbf{x}_i)$ denote the largest similarity of the non-skeletal point $\mathbf{x}_i$ to points in the skeleton $\mathcal{X}_s$. Then we have

$$P(\mathcal{X}_s, \mathbf{x}_i) = \max_{\mathbf{x}_j \in \mathcal{X}_s} s_{ij}. \tag{1}$$

In our experiments, we define s_{ij} as the Gaussian Kernel, i.e., $s_{ij} = \exp(-||\mathbf{x}_i - \mathbf{x}_j||^2/2\sigma^2)$, where σ is the kernel width parameter.

For each data point, its largest similarity to the skeleton points can be used to measure the uncertainty in deciding the cluster membership. In particular, the uncertainty is expected to be inversely related to the value of $P(\mathcal{X}_s, \mathbf{x}_i)$, i.e., the larger the largest similarity of $\mathbf{x}_i$ to the skeleton, the smaller is the uncertainty in deciding its cluster membership. Following the principle of active learning [9], we choose the data point with the largest uncertainty in deciding cluster membership, which leads to the proposed min-max approach, namely selecting the data point whose largest similarity to the skeleton points is the smallest. Hence, the most uncertain point $\mathbf{q}$ can be chosen as

$$\mathbf{q} = \arg\min_i P(\mathcal{X}_s, \mathbf{x}_i) = \arg\min_i \max_{\mathbf{x}_j \in \mathcal{X}_s} s_{ij}. \tag{2}$$



(a) True Labels (b) Kmeans ($F_1 = 0.69$) (c) $t = 3$, $F_1 = 0.64$ (d) $t = 4$, $F_1 = 0.88$

(e) $t = 6$, $F_1 = 0.89$ (d) $t = 25$, $F_1 = 0.93$ 0f) $t = 38$, $F_1 = 0.97$ (g) $t = 50$, $F_1 = 1.00$

Figure 4. Illustration of the proposed min-max query selection. (a) A 2-D Dataset with 75 samples generated from a mixture of 3 Gaussians. (b) K-means clustering, (c) skeleton from three must-not link queries, (d) first must-link constraint increases the performance significantly, (e) two must-link constraints obtained, (f) & (g) clustering with 25 and 38 constraints, (h) perfect clustering with 50 constraints. Solid lines represent must-link queries and broken lines represent must-not link queries. F_1 statistic denotes clustering quality and t indicates the number of constraints.

The cluster membership of the selected point $\mathbf{q}$ is determined by formulating pairwise queries of the form $(\mathbf{q}, \mathbf{u}_k)$, where $u_k \in \mathcal{X}_s$, and u_k belongs to cluster k, $k = 1, \cdots, K$, until a must-link query is obtained.

The skeleton found by the initial explore phase may not be robust as it involves very few points from the dataset. After soliciting the cluster membership of query point q, we can add it to the existing skeleton, i.e., $X_s \rightarrow X_s \cup q$ The proposed algorithm using the min-max approach is summarized in Figure 2.

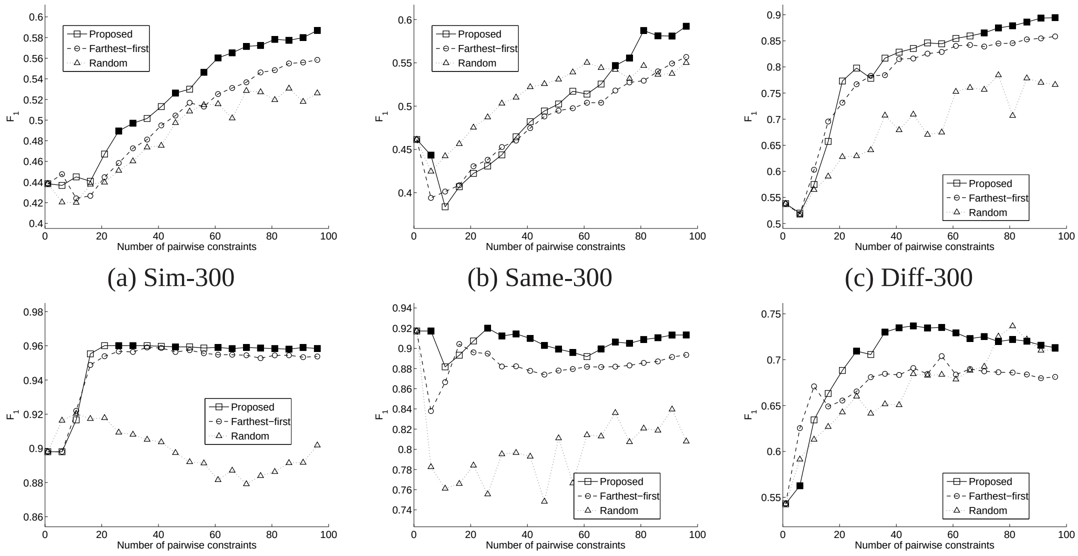
In addition to the constraints generated above, we can infer additional constraints using the transitive closure of the set of constraints [1]. Given three data points u, v and w, let $+(u, v)$ denote a must-link constraint between u and v, and $-(u, v)$ denote a must-not link constraint. We now have, (i) $+(u, v) \wedge +(v, w) \Rightarrow +(u, w)$, (ii) $+(u, v) \wedge -(v, w) \Rightarrow -(u, w)$, and (iii) for a two cluster case, $-(u, v) \wedge -(v, w) \Rightarrow +(u, w)$.

3 Experiments and Results

We present the performance of the proposed query selection algorithm on six datasets, including three used to evaluate FFQS in [1], 2 UCI datasets and one real face dataset. The text datasets Sim-300 and Same-300 datasets span around 3,000 dimensions, and Diff-300 spans around 1,000 dimensions. However, since there are only 300 samples per dataset, we followed the commonly used practice of Latent Semantic Indexing (LSI) [8] to reduce the dimension to 20. The iris dataset has 150 samples in 4 dimensions. The Digits-389 [3] dataset has 317 examples in 16 dimensions. The ethnicity dataset is a collection of real face images [6] with 2630 face images reduced using PCA to 30 dimensions. The ethnicity dataset is a two class dataset, while the rest of them have three classes. The results on several other datasets that we tested are similar and we do not present them here due to limited space.

We use the MPCK-means algorithm [3], the state of the art scalable constraint based clustering algorithm, to test the utility of the constraints selected. As in previous studies, we assume that the number of clusters is known. We follow the experimental setup in [1] and report the mean performance over 20 runs of 5-fold cross validation for the first 100 queries. For each fold, queries are selected only from four out of the five subsets of the data. The clustering is performed on the complete data, and the performance is evaluated using the F_1 measure [1] on the fifth fold that is not included in the query selection. The kernel width parameter of the Gaussian kernel is set to the 20^{th} percentile of the distribution of pairwise Euclidean distances.

The performance of the three query selection algorithms is summarized in Fig. 5, which plots the F_1



(a) Sim-300 (b) Same-300 (c) Diff-300

(d) Iris (e) Digits-389 (f) Ethnicity

Figure 5. Performance of min-max (proposed), random and FFQS. Significant differences (measured using paired t-test at 95% confidence level) between the proposed and FFQS algorithms are shown with filled markers.

measure of cluster validity against the number of constraints. The proposed query selection algorithm outperforms the FFQS algorithm and random query selection on all the six datasets considered. Differences in performance that are significant at 95% confidence using a paired t-test are plotted using filled markers. For the Same-300 dataset, randomly selected queries perform better than any of the query selection methods for small number (< 100) of constraints. This may be because, for small number of queries the initial `Explore` phase may not be able to represent the cluster structure. However, as the number of queries increases, the proposed approach expands the skeleton and outperforms both the FFQS and random queries.

4 Conclusion

We have presented an active query selection algorithm for semi-supervised clustering, that generalizes the method in [1]. An implementation based on a special case of the min-max approach, using similarity between the pair of points as the confidence of must-link constraint is developed. The results on the datasets used in [1, 3] and UCI repository show improved performance of the proposed algorithm compared to the FFQS method and random query selection. The performance of the proposed query selection algorithm is as measured by the F_1 statistic, in general, better than the FFQS approach.

References

[1] S. Basu, A. Banerjee, and R. J. Mooney. Active semi-supervision for pairwise constrained clustering. In *Proc. SDM*, pages 333–344, 2004.

[2] S. Basu, M. Bilenko, and R. J. Mooney. A probabilistic framework for semi-supervised clustering. In *Knowledge Discovery in Databases*, pages 59–68, 2004.

[3] M. Bilenko, S. Basu, and R. J. Mooney. Integrating constraints and metric learning in semi-supervised clustering. In *Proc. 21st ICML*, pages 81–88, 2004.

[4] I. Davidson, K. Wagstaff, and S. Basu. Measuring constraint-set utility for partitional clustering algorithms. In *Proc. PKDD*, pages 115–126, 2006.

[5] R. O. Duda, P. E. Hart, and D. G. Stork. *Pattern Classification*. Wiley, 2000.

[6] A. Jain and X. Lu. Ethnicity identification from face images. In *Proc.SPIE*, volume 5404, pages 114–123, 2004.

[7] A. K. Jain and R. C. Dubes. *Algorithms for Clustering Data*. Prentice Hall, 1988.

[8] T. Landauer, P. W. Foltz, and D. Laham. Introduction to latent semantic analysis. *Discourse Processes*, 25:259–284, 1998.

[9] D. D. Lewis and J. Catlett. Heterogeneous uncertainty sampling for supervised learning. In *Proc. 11th ICML*, pages 148–156, 1994.

[10] K. L. Wagstaff. Value, cost, and sharing: Open issues in constrained clustering. In *Lecture Notes in Computer Science*, volume 4747, pages 1–10, 2007.