Semi-supervised Clustering with Active Query

Yibing Luo 912491862
Che Xiao 997604551
Xi Chen 912507387

Abstract

Existing semi-supervised clustering is achieved by casting clustering into a matrix completion problem, and solves it efficiently by incorporating the available side information that is usually in the form of pairwise constraints. The main drawback of this kind of approach is that, the number of required pairwise constraints is too large. We hope to use active matrix completion to reduce the number of given constraints.

1 Introduction

Clustering is typically an unsupervised learning method**.** We want to find the best data partition based on the data pattern. In practice, sometimes we have additional side information for a small amount of data points. We want to incorporate the side information into traditional clustering. Since the process falls between unsupervised learning and supervised learning, it is called semi-supervised clustering.

One group of semi-supervised clustering directly incorporates the pairwise constraints into traditional unsupervised clustering. These algorithms either discard the cluster assignment inconsistent with given constraints (hard constrained clustering), or penalize the clustering partition which do not agree with constraints (soft constrained clustering).

Another group of semi-supervised clustering is based on distance matric learning. Yi *et al.* [1]proposed an algorithm based on input pattern assisted matrix. This algorithm first presents a matrix completion based framework, which is partitioning all existing data points with the pairwise constraints. Then it completes the matrix by solving the correlation between input pattern and cluster assignments.

We are willing to incorporate active learning algorithm into matrix completion step. Suppose we are allowed to query side information for certain points. We want to query the most informative unknown positions and add them to the existing training set. The additional side information will lead to a better final result. In this case, we want to query certain unknown constraints from the ground truth, and get a better matrix completion result.

People proposed several different ways to select query points. Just to name a few. Karimi *et al.* [2] query the positions with the highest expected change to current models. Jin *et al.* [3] use the Bayesian posterior distribution of the model for inference. Rish *et al.* [4] query the positions with the smallest margin, that is, the points that the model are most uncertain about.

2 Methods

2.1 matrix completion based framework

Let $D = \{O_1, \dots, O_n\}$ be the set of n objects to be clustered, and $X = \{x_1, \dots, x_n\}$ be their feature representation. Let M denote the set of must-link constraints. $(i,j) \in \mathcal{M}$ implies x_i and x_j are in the same cluster, while C denote the set of cannot-link constraints. $(i,j) \in$ C implies x_i and x_j are in different clusters. Let $\Omega = M \cup C$ include all the pairwise constraints.

Yi *et al.* [1]proposed the input pattern assisted pairwise similarity matrix completion to perform the semi-supervised clustering.

Firstly, a similarity matrix S ∈ {-1, +1} is defined as follows.

$$S_{ij} = \begin{cases} 1, & x_i \text{ and } x_j \text{ are assigned to the same cluster} \\ -1, & x_i \text{ and } x_j \text{ are not assigned to the same cluster} \end{cases}$$

Clearly, $S_{i,j} = 1$ if (i, j) ∈ M and $S_{i,j} = -1$ if (i, j) ∈ C

Based on partially observed constraints in M and C set, we have partially observed entries in the binary similarity matrix S. Filling out the missing value in S will prepare us to find the best data partition. That is to say, we convert the clustering problem to a matrix completion problem.

Secondly, we assume the membership vectors $\{u_i\}_{i=1}^{r}$ can be well approximated by $\{z_i\}_{i=1}^{k}$. Here, $Z=(\mathbf{z_1},\dots,\mathbf{z_k})$ includes the first k left singular vectors of X, which is the feature representation. With input pattern assisted, we can approximate S by ZMZ^T, $M \in \mathbb{R}^{k\times k}$. The optimal M will be obtained by solving following optimization problem:

$$\min_{M \in R^{k\times k}} |M|_{tr} + \frac{C}{2} ||R_\Omega(ZMZ^T) - R_\Omega(S)||_F^2$$

$$[R_\Omega(S)]_{i.j} = \begin{cases} S_{i,j}\ (i,j), & (i,j) \in \Omega \\ 0, & (i,j) \notin \Omega \end{cases}$$

Gradient descent or efficient stochastic subgradient descent can be applied to solve the nuclear norm regularization problem and find the optimal $\widehat{M}$. The estimated binary similarity matrix is given by $\hat{S} = Z\widehat{M}Z^T$.

Based on $\hat{S}$, we can find r partition of the data by applying k-means algorithm over top r eigenvectors of $\hat{S}$.

2.2 Active query

We want to reduce the sample complexity of existing algorithm by active learning technique. Our goal is to successfully recover the similarity matrix with fewer constraints. Specifically, we want to start with a small number of constraints and actively query additional constraints in order to complete similarity matrix S. In practice, we want to first recover the similarity matrix S by p constraints. Then we examine the estimated similarity matrix to figure out the q most uncertain positons, i.e. the q most uncertain (i,j) pairwise constraints in $\hat{S}$. We query those q entries from the ground truth and reconstruct S by ($p+q$) queries. We apply spectral clustering over the newly estimated similarity matrix $\hat{S}$, and find the best data partition.

Now the question is how to wisely select q entries to query. The q positions should be the most uncertain entries in $\hat{S}$ after first round of matrix completion. Recall that S is defined as: $S_{i,j} = 1$ if x_i and x_j are assigned to the same cluster, while $S_{i,j} = -1$ if x_i and x_j are assigned to different clusters. Then if $S_{i,j} = 0$, we are not sure whether the x_i and x_j belong to the same cluster or not. So we assume the closer to 0 $S_{i,j}$ is, the more uncertain the position (i, j) is. Based on the assumption, we select q pairs of (i, j) which are most close to 0 in estimated similarity matrix $\hat{S}$, and query those q constraints from the ground truth.

3 Experiments

We evaluate our algorithm over several benchmark datasets. Table 1 summarizes the datasets we use in the following experiments.

Table 1. Description of datasets used in this study

	# of instances	**# of features**	**# of clusters**
Segment	2310	19	7
dna	2000	180	3
diabetes	768	8	2

We want to compare our semi-clustering with active queries to the original one without queries. We want to see how the relative error decreases as the number of queries increase.

Here we wisely selected q positions to query from the ground truth. We expect to see the prediction error is smaller than the situation where we randomly query q positions from the ground truth. We first sample 0.1% of total constraints, and then subsequently query 20, 40, 60, 80, 200, 400, 600, 800, 2000, 4000, 6000, 8000 additional constraints from ground truth, by active query and random query as a control. We apply the similar comparison for active query and random query at base sample rate at 0.001%, 0.0005% and 0.0001%. Each experiment is repeated 5 times.

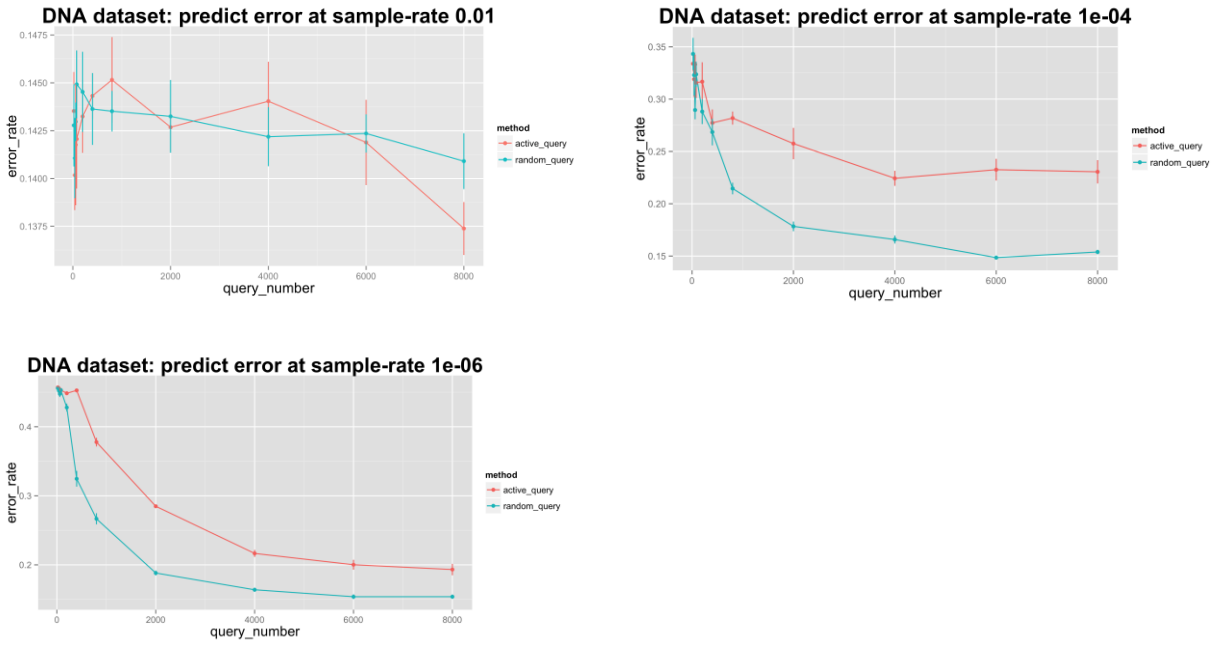

Figure 1. Experiments on DNA dataset

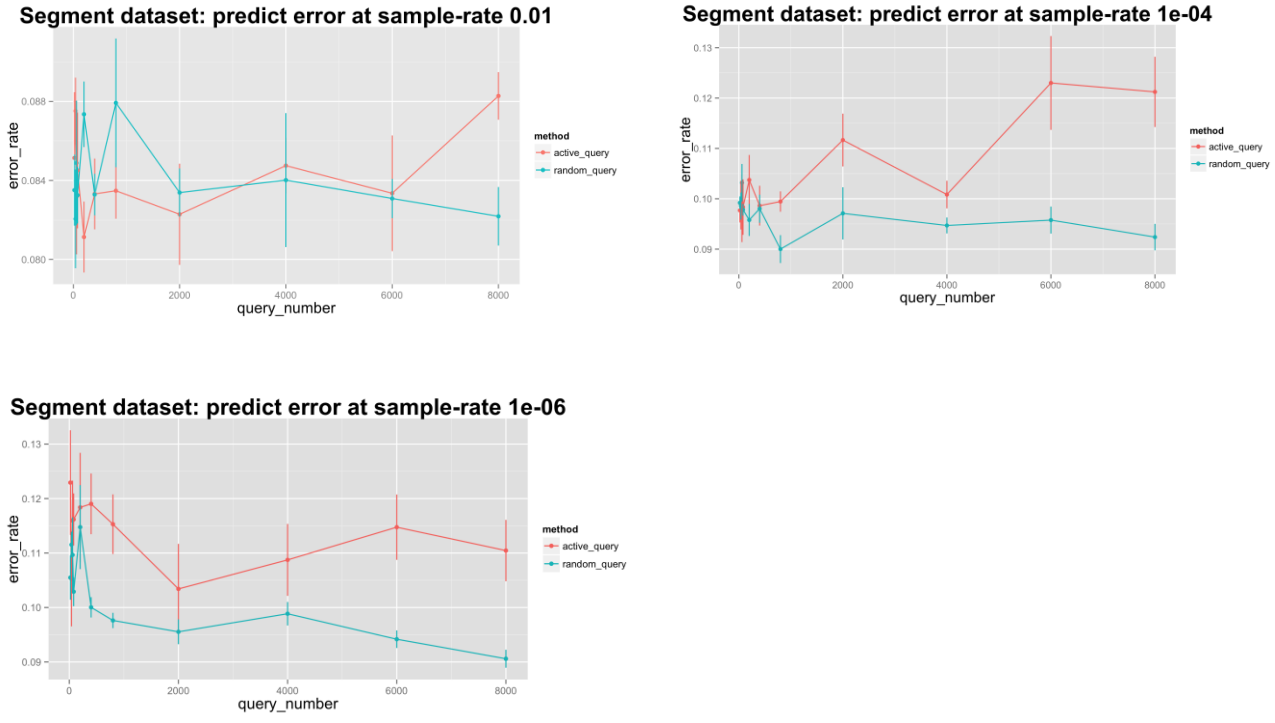

Figure 2. Experiments on Segment dataset

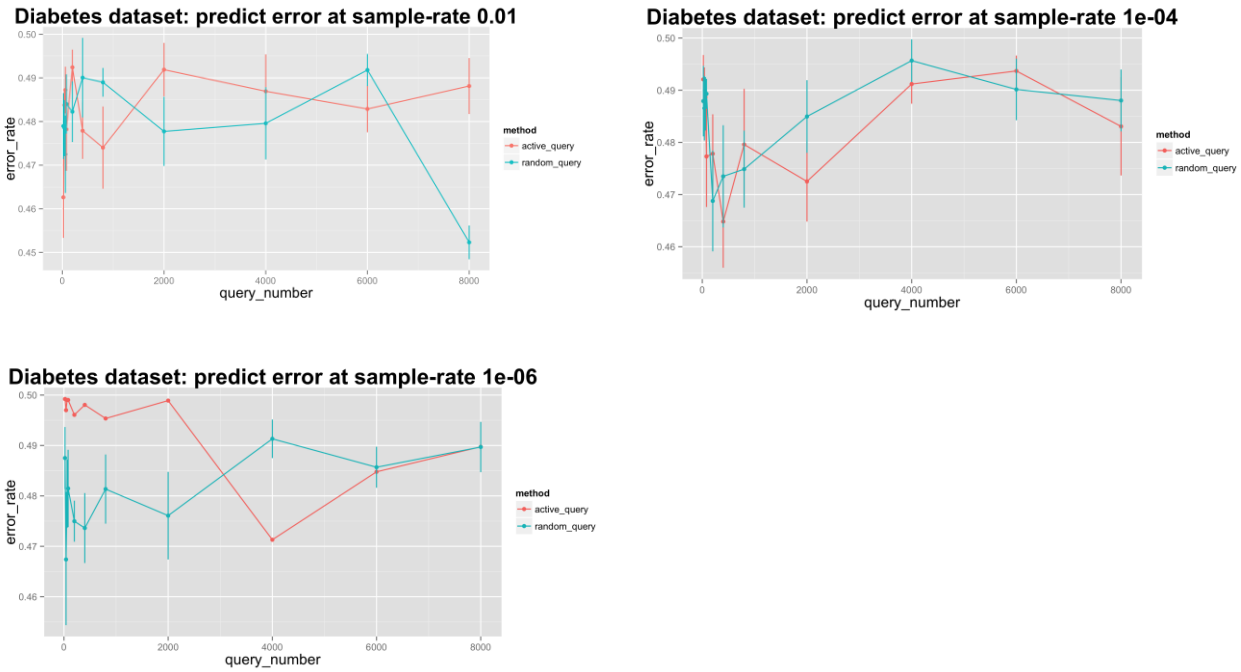

Figure 3. Experiments on Diabetes dataset

When the base sample rate is large (DNA dataset, base sample rate = 0.01; Segment dataset, base sample rate = 0.01,10^{-4},10^{-6}; Diabetes, rate = 0.01,10^{-4},10^{-6}), the error rate almost saturates. Querying more entries will not reduce the error much. When the base sample rate is not large enough (DNA dataset, base sample rate = 10^{-4},10^{-6}), the error rate in general decreases as the query number increases. It is interesting to notice that random query usually works better than active query.

4 Discussion

There is a lower bound for the error rate. After reaching certain error rate, querying more entries will not help to increase the prediction accuracy. If the lower bound is not reached, querying more entries will in general lead to the decrease in error rate.

We further examine the estimated similarity matrix before additional query. When the base sample rate is small (e.g. base query rate = 10^{-6}), the prediction error is large, and more entries in $\hat{S}$ center around 0. That is to say, a large number of entries are very close to 0, suggesting a large number of constraints are of similar degree of uncertainty. It is hard to differentiate which data points are the most uncertain ones. Our query selecting algorithm is biased to choose the constraints with small row index. On the contrary, the random query is an unbiased selection. That might be one of the reasons why the random query method behaves better than the active query method.

The results we obtain are not ideal. For the future study, we could try alternative way to select the active query position. We can quantify the uncertainty of prediction of each missing entry in the

similarity matrix $\hat{S}$. Chakraborty *et al.* [5] assume that the set of missing entries conditioned on the observed entries follows a multivariate normal distribution. They compute the covariance matrix and query the top queries with largest values on the diagonal in the covariance matrix. We could also compute the posterior distribution of the model for inference.

Reference

[1] J. Yi, L. Zhang, R. Jin, Q. Qian, and A. K. Jain, "Semi-supervised Clustering by Input Pattern Assisted Pairwise Similarity Matrix Completion," *Proc. 30th Int. Conf. Mach. Learn.*, vol. 28, pp. 1400–1408, 2013.

[2] R. Karimi, C. Freudenthaler, A. Nanopoulos, and L. Schmidt-Thieme, "Non-myopic active learning for recommender systems based on matrix factorization," *Proc. 2011 IEEE Int. Conf. Inf. Reuse Integr.*, pp. 299–303, 2011.

[3] R. Jin and L. Si, "A bayesian approach toward active learning for collaborative filtering," *Proc. 20th Conf. Uncertain. Artif. Intell.*, pp. 278–285, 2004.

[4] I. Rish and G. Tesauro, "Active collaborative prediction with maximum margin matrix factorization," *Inform. Theory App. Work.*, 2007.

[5] S. Chakraborty, J. Zhou, V. Balasubramanian, S. Panchanathan, I. Davidson, and J. Ye, "Active matrix completion," *Proc. 2013 IEEE 13th Int. Conf. Data Min.*, no. November 2015, pp. 81–90, 2013.