clustering uncertain data

Presented by Mohammad Hadi Salari

Simon Fraser University

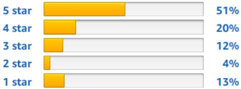
msalari@sfu.ca

November 20, 2019

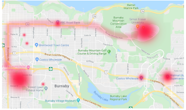
Overview

Some Definitions

- Uncertain Data: When each data point has a probability distribution over some space instead of being one certain point in the space.



(a) Amazon users ranking for one item



(b) The distribution of a person's location

- Clustering is the task of grouping a set of objects in such a way that objects in the same group (called a cluster) are more similar (in some sense or another) to each other than to those in other groups (clusters)[1]

Motivation

Applications of clustering uncertain points:

- Recommendation system
- Dimensionality reduction
- Summarization
- ...

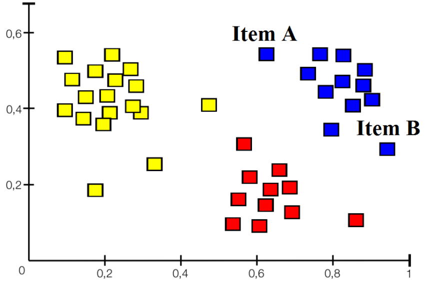


Figure: clusters of items of an online shop

Problem Formulation

- We have N points, Each comes from a distribution: $p_i \sim P_i$
- We want to find K clusters, whose centers comes from a distribution: $q_c \sim Q_c$
- Point i belongs to center c with the probability of γ_{ic}
  $\forall p_i \sum_{q_c} \gamma_{ic} = 1.0$
- We define the dissimilarity of point i to cluster c by Kullback–Leibler divergence of their distributions.
  Other options: $\frac{1}{2}(KL(P_i||Q_c) + KL(Q_c||P_i))$, Jensen–Shannon divergence

Objective Function

$$argmin \sum_{i=1}^{N} \sum_{c=1}^{K} \gamma_{ic} KL(P_i||Q_c)$$

Problem Formulation Example

- $p_1, p_2, p_3 \sim \mathcal{N}(0.0, 1.0)$ and $p_4, p_5 \sim \mathcal{N}(1.0, 1.0)$
- $q_1 \sim \mathcal{N}(0.0, 1.0)$ and $q_2 \sim \mathcal{N}(1.0, 1.0)$
- $\gamma_{11} = \gamma_{21} = \gamma_{31} = 1.0 and \gamma_{42} = \gamma_{52} = 1.0$
- Objective function $=$ $3KL(\mathcal{N}(0.0, 1.0)||\mathcal{N}(0.0, 1.0)) + 2KL(\mathcal{N}(1.0, 1.0)||\mathcal{N}(1.0, 1.0)) = 0$

Related Works

Current works can be separated to three groups:

- density-based algorithms: Put the dense region of point to a cluster. e.g. FDBSCAN[2]
- possible world-based algorithms: A set of possible worlds is sampled from an uncertain data. Aggregating the result of the clusters of the possible worlds. e.g. [3]
- partition-based algorithms: Try to minimize the expected distance of points to their cluster centers. e.g. [4]

Related Works Drawbacks

- density-based algorithms: They assume that pairwise distances between uncertain objects are mutually independent which may not be a reasonable assumption.

Independent Distance Assumption

$$P(A \leftrightarrow_\epsilon B, B \leftrightarrow_\epsilon C) = P(A \leftrightarrow_\epsilon B)P(B \leftrightarrow_\epsilon C)$$

($\leftrightarrow_\epsilon$ means that the distance is lower than an ϵ)

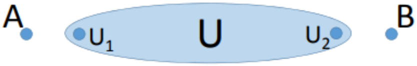
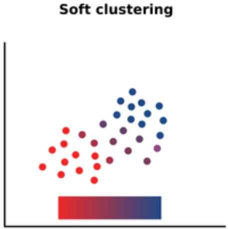


Figure: Distances of U to A and B are not independent

Related Works Drawbacks

- possible world-based algorithms: A sampled possible world does not consider the distribution of a data object. So The most probable clusters calculated using possible worlds may still carry a very low probability.
- partition-based algorithms: Current works assume hard clusters which means each data point either belongs to a cluster completely or not. Some of them also restrict cluster centers to be exactly one of the data points.



Our Solution

- We just have some samples from each P_i.
  In our method we use non-parametric Kernel density estimation with Gaussian kernels to estimate P_is.
- Use an EM scheme to solve the problem.

The steps of the EM scheme

- step 1: given the centers of the clusters (Q_c), we find the probability of each point belonging to each cluster (γ_{ic}).
- step 2: given the probability of each point belonging to each cluster (γ_{ic}) we find the Q_cs that minimize the objective function.

Our Solution

given the centers of our clusters we consider γ_{ic} related to the KL-divergence of P_i and Q_c.

$$\gamma_{ic} = softmax(-KL(P_i||Q_c)) = \frac{e^{-KL(P_i||Q_c)}}{\sum_{c\prime} e^{-KL(P_i||Q_{c\prime})}}$$

Our Solution - First Try

We assume the centers are mixtures of Gaussians and use gradient descent to find their parameters that minimize the objective function.
$Q_c \sim \lambda_1 \mathcal{N}(\mu_1, \sigma_1) + \lambda_2 \mathcal{N}(\mu_2, \sigma_2) + ... + \lambda_k \mathcal{N}(\mu_k, \sigma_k)$

Gradient of objective function of the centers parameters

$$\frac{\nabla(L)}{\nabla(\mu_c)} = -\sum_{P_i} \int P_i(x) \frac{\frac{\nabla(Q_c(x))}{\nabla(\mu_c)}}{Q_c(x)} dx$$
$$\frac{\nabla(L)}{\nabla(\sigma_c)} = -\sum_{P_i} \int P_i(x) \frac{\frac{\nabla(Q_c(x))}{\nabla(\sigma_c)}}{Q_c(x)} dx$$
$$\frac{\nabla(L)}{\nabla(\lambda)} = -\sum_{P_i} \int P_i(x) \frac{\frac{\nabla(Q_c(x))}{\nabla(\lambda_c)}}{Q_c(x)} dx$$

Our Solution - First Try

Gradient of $Q_c(X)$ of its parameters

$$\frac{\nabla(Q_c(x))}{\nabla(\mu_c)} = \lambda_c \frac{1}{\sqrt{2\pi}\sigma_c} e^{\frac{-(x-\mu_c)^2}{2\sigma_c^2}} \frac{x-\mu_c}{\sigma_c^2}$$

$$\frac{\nabla(Q_c(x))}{\nabla(\sigma_c)} = \lambda_c \frac{1}{\sqrt{2\pi}\sigma_c^2} e^{\frac{-(x-\mu_c)^2}{2\sigma_c^2}} (-1 + \frac{(x-\mu_c)^2}{\sigma_c^2})$$

$$\frac{\nabla(Q_c(x))}{\nabla(\lambda_c)} = \frac{1}{\sqrt{2\pi}\sigma_c} e^{\frac{-(x-\mu_c)^2}{2\sigma_c^2}}$$

We estimate those integrals with Monte Carlo integration.

Why it doesn't work?!

- In each iteration, the gradient descent takes a lot of time.
- To have a low variance in the Monte Carlo estimation, we need to choose a large sample which makes the previous problem worse.
- We do not know how much complex the centers should be. So we to run the algorithm several times with different K and that makes the previous problems even worse!

Our Solution - Second Try

We tried to find a close form for the objective function minimization.

Minimizing objective function analytically

$$argmin \sum_{i=1}^{N} \sum_{c=1}^{K} \gamma_{ic} KL(P_i || Q_c) =$$
$$argmin \sum_{c=1}^{K} \sum_{i=1}^{N} \gamma_{ic} KL(P_i || Q_c) =$$
$$argmin \sum_{c=1}^{K} \sum_{i=1}^{N} \gamma_{ic} \int P_i(x) \log \frac{P_i(x)}{Q_c(x)} dx =$$
$$argmax \sum_{c=1}^{K} \sum_{i=1}^{N} \gamma_{ic} \int P_i(x) \log Q_c(x) dx =$$
$$argmax \sum_{c=1}^{K} \int \sum_{i=1}^{N} \gamma_{ic} P_i(x) \log Q_c(x) dx =$$
$$argmax \sum_{c=1}^{K} \int P'_c(x) \log Q_c(x) dx =$$
$$argmin \sum_{c=1}^{K} \int P'_c(x) \log \frac{P'_c(x)}{Q_c(x)} dx =$$
$$argmin \sum_{c=1}^{K} KL(P'_c || Q_c) \Rightarrow$$
$$\mathbf{Q_c \sim P'_c, \quad P'_c \sim \sum_{i=1}^{N} \frac{\gamma_{ic}}{\sum_{j=1}^{N} \gamma_{jc}} P_i}$$

When should we terminate?

We terminate our algorithm when the γ_{ic}s in EM-scheme have been converged.

Convergence Metric

Convergence $= \max_{p_i,q_c}$ the difference between γ_{ic} in the current and previous iteration.

So we terminate our iterations when the convergence metric is lower than a threshold or the iteration number is higher than Max Iteration.

Centers Initialization

- First, we pick the point that minimize the objective function if we assign all points to this cluster.
- Then, to find initial center t+1, for each remaining point (N-t) we first, compute γ_{ic}s if we add that point to our centers and then find the one that minimize our objective function.

Baseline

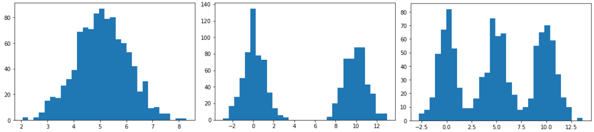
We compare our algorithm on the Movement data set with the state-of-the-art clustering algorithms for uncertain data:

- UK-means(UKM) [5]
- CK-means (CKM) [6]
- UK-medoids (UKMD) [7]
- MMVar (MMV) [8]
- UCPC [9]
- FDBSCAN (FDB) [2]
- FOPTICS (FOP) [10]
- RPC [3]

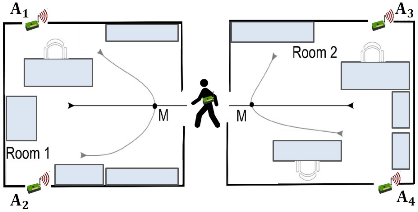
Because these are hard clustering algorithms, we need to convert the output of our algorithm to hard clusters. So we assign each point P_i to the cluster that has the maximum γ_{ic}.

Data sets

- Synthetic data: We generated each cluster as a mixture of Gaussians with different number of Gaussians. We generated 10 points for each cluster.



- Real data: Indoor User Movement Prediction from RSS data Data Set, 13197 radio signal records about 314 temporal sequences from a wireless sensor network. According to user movement path, the data set is divided into six classes.



Evaluation

- If we consider G as ground truth clustering and C as the clustering obtained by our method.
- TP is the set of common pairs of objects in both C and G
- FP is the set of pairs of objects in C but not G
- FN is the set of pairs of objects in G but not C
- TN is the set of pairs of objects not in both G and C

Precision and Recall

$$\text{Precision(C)}=\frac{|TP|}{|TP|+|FP|}$$
$$\text{Recall(C)}=\frac{|TP|}{|TP|+|FN|}$$
$$\text{Accuracy(C)}=\frac{|TP|+|TN|}{|TP|+|FN|+|FP|+|TN|}$$

- With only 5 iteration we got the precision and recall of 1.0 on the synthetic data.
- With 14 iterations we got 0.719 accuracy, 0.36 precision and 0.34 recall on the Movement data set.

Dataset	Metric	UKM	CKM	UKMD	MMV	UCPC	FDB	FOP	PDB	SC	RPC
Movement	ACC	0.3490	0.3341	0.3478	0.3427	0.3494	0.2834	0.2643	0.3121	0.2548	**0.4315**

Figure: The precision of some state-of-the-art algorithms [3]

Experiment Figures

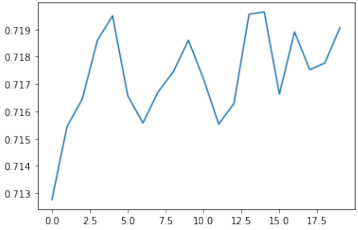


Figure: How accuracy change over iterations

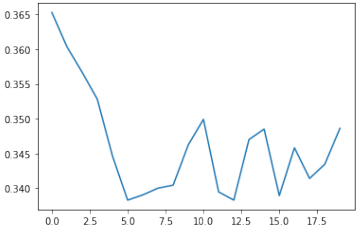


Figure: How recall change over iterations

Discussion and Conclusion

- We proposed an EM algorithm to find clusters for uncertain data using KL-divergence distance.
- We used a non-parametric kernel density estimation to estimate the density function of data points, but it works not very well when the dimension of our space is larger than 2. We can use generative models such as Normalizing Flows to generate new samples and measure the probability of samples that work better for large dimensions.
- The time complexity of our algorithm is $O(N^2 \times k^2 \times$ (sample size) $+$ (iteration number) $\times N \times k \times$ (sample size)$)$.

References I

- M. A. Deshmukh and R. A. Gulhane *Importance of Clustering in Data Mining*. International Journal of Scientific Engineering Research, Volume 7, Issue 2, February-2016 ISSN 2229-551
- Hans-Peter Kriegel and Martin Pfeifle *Density-based clustering of uncertain data*. KDD 2005
- Hongmei Liu and Xian-Chao Zhang and Xiaotong Zhang and Qimai Li and Xiao-ming Wu *Clustering Uncertain Data via Representative Possible Worlds with Consistency Learning*. ArXiv 2019
- Bin Jiang and Jian Pei and Yufei Tao and Xuemin Lin-ming Wu *Clustering Uncertain Data Based on Probability Distribution Similarity*. IEEE Transactions on Knowledge and Data Engineering 2013
- Michael Chau, Reynold Cheng, Ben Kao, and Jackey Ng. *Uncertain data mining: An example in clustering location data*. In PAKDD, pages 199–204, 2006.

References II

Sau Dan Lee, Ben Kao, and Reynold Cheng. *Reducing UK-means to K-means. In ICDM Workshops*. pages 483–488, 2007.

Francesco Gullo, Giovanni Ponti, and Andrea Tagarelli. *Clustering uncertain data via Kmedoids*. In SUM, pages 229–242, 2008.

Francesco Gullo, Giovanni Ponti, and Andrea Tagarelli. *Minimizing the variance of cluster mixture models for clustering uncertain objects*. In ICDM, pages 839–844, 2010.

Francesco Gullo and Andrea Tagarelli. *Uncertain centroid based partitional clustering of uncertain data*. In VLDB, pages 610–621, 2012.

Hans-Peter Kriegel and Martin Pfeifle. *Hierarchical density-based clustering of uncertain data*. In ICDM, pages 689–692, 2005.

Thank You!