ISSN 1004-9037,CODEN SCYCE4
Journal of Data Acquisition and Processing Vol. 39,No. 2,Mar. 2024,pp. 376－384
DOI:10. 16337/j. 1004-9037. 2024. 02. 010

http://sjcj. nuaa. edu. cn
E-mail:sjcj@nuaa. edu. cn
Tel/Fax:＋86-025-84892742

分布式稀疏软大间隔聚类

谢云轩，陈松灿

(南京航空航天大学计算机科学与技术学院,南京 211106)

摘　要：虽然软大间隔聚类(Soft large margin clustering,SLMC)相比其他诸如K-Means等算法具有更优的聚类性能与某种程度的可解释性,然而当面对大规模分布存储数据时,均遭遇了同样的可扩展瓶颈,其涉及的核矩阵计算需要高昂的时间代价。消减此代价的有效策略之一是采用随机Fourier特征变换逼近核函数,而逼近精度所依赖的特征维度常常过高,隐含着可能过拟合的风险。本文将稀疏性嵌入核SLMC,结合交替方向乘子法(Alternating direction method of multipliers, ADMM),给出了一个分布式稀疏软大间隔聚类算法(Distributed sparse SLMC, DS-SLMC)来克服可扩展问题,同时通过稀疏化获得更好的可解释性。

关键词：交替方向乘子法;软大间隔聚类;分布式机器学习;核近似

中图分类号：TP391　　**文献标志码**:A

Distributed Sparse Soft Large Margin Clustering

XIE Yunxuan, CHEN Songcan

(College of Computer Science and Technology, Nanjing University of Aeronautics & Astronautics, Nanjing 211106, China)

Abstract: Soft large margin clustering (SLMC) has been proved to achieve better clustering performance and interpretability than other algorithms, such as K-Means. However, when facing large scale distributed data storage, computing involved kernel matrix requires large time cost. One of the effective strategies to reduce this time cost is to use random Fourier feature transform to approximate the kernel function, and the feature dimension on which approximating accuracy depends is often too high, which implies the risk of overfitting. This paper embeds the sparsity into kernel SLMC and combines the alternating direction method of multipliers (ADMM) with SLMC. Finally, we propose a distributed sparse soft large margin clustering algorithm (DS-SLMC) to overcome scalability problem and achieve better interpretability through sparsity.

Key words: alternating direction method of multipliers (ADMM); soft large margin clustering (SLMC); distributed machine learning; kernel approximation

引　　言

如今,AI技术几乎在每个行业都显示出其独特的优势,优势之一便来自大数据驱动,如何应用大数据进行训练则成为了一个关键主题,其涉及的挑战主要在于训练模型的传统方法不再适用。因为传统

收稿日期:2022-08-31;**修订日期**:2023-01-19

的机器学习需读取全部的训练集进行模型拟合,当训练集因物理容量有限而无法处理时,这种策略自然失效,尤其在一般的台式机上执行时。为了克服这个问题,选择之一是对数据集进行精炼,其通过数据挖掘实现。聚类[1-2]是实现数据挖掘的流行手段,属于无监督学习,它将数据集划分为多个由相似对象组成的类或簇,使得同一簇或聚类内的任何两个示例的相似性大于不在同一簇中任两个示例的相似性。然而,对大数据或分布式存储数据的聚类,一般的单机聚类算法无法满足训练需求,即使可行,其时间代价也令人无法承受[3]。因而对单机聚类算法进行分布式改造减少时间成本,是提高聚类算法伸缩性的有效途径。尽管学界已提出了极其丰硕的聚类算法,产生了若干性能优越的聚类算法,包括DeepClustering[4]等,但对大量现存单机聚类算法不作选择地改造不切实际。因此本文限定在相对简单但又具代表性的聚类算法进行分布式改造,目的不是为了提出新的单机聚类算法,而是示范如何将那些性能相对优的简单算法适应到分布式学习环境,以重用这些算法使之发挥更大的潜能。

现有的性能相对认可且比较简单的聚类算法有K均值聚类(K-Means)[5]及其软化版模糊K-Means(Fuzzy C-Means, FCM)[6]与软大间隔聚类(Soft large margin clustering, SLMC)[7]。上述算法属于不同的实现方式,前一类是在数据空间中聚类数据,而后一类则是直接将数据映射到预定的C个输出标记实现聚类。本文在此两类中分别选取其中之一实现分布式改造,意在验证单机上性能有效的算法改造后其相对优势是否仍能继续保持。考虑到软聚类一般优于其硬聚类性能(由于软聚类表达出更多的信息,如数据点隶属聚类的表示不是0或1,而是[0,1]间的一个数),因此分别选用FCM和SLMC,但作为比较,也实现了FCM的分布式版。选择FCM是因为它流行和有效,而选择SLMC,除了不同于FCM在数据空间内的聚类之外,其作为判别型算法在标记空间进行聚类产生的良好性能和一定程度的可解释性,并受到后续关注[8-9]。当面对庞大的数据集时,SLMC与K-Means和FCM等一样,遭遇了类似的瓶颈:首先是过高的时间代价,尤其在计算它们核版本中的核矩阵时需要2倍于数据规模的时间代价。因此在数据集过大的情况下,高计算复杂度导致单机训练消耗无法忍受的复杂度。此外,作为单机算法,也无法适应数据分布存储场景。因此引入分布式改造是一个自然选择。然而,改造的挑战一是如何在保持原有核聚类算法优势的同时,降低其时间复杂度;二是如何使其通过不同节点间的数据通讯来实现分布式聚类。考虑到时间加速,如果简单地分割数据集进行单机运算,则在迭代更新时需保证本地参数的一致性,否则将无法分布式计算。

本文为了提高SLMC的可扩展性,对SLMC进行分布式改造。具体来说,本文将其与交替方向乘子法(Alternating direction method of multipliers, ADMM)结合,提出了SLMC的核近似分布式优化,降低时间代价。同时为了防止过拟合,提高算法的可解释性,对系数矩阵做了稀疏化处理。

1 相关工作

在现有的分布式优化方式中,ADMM是一种比较流行的方法。ADMM通过分解-协调过程,将全局问题分解为局部子问题,通过协调子问题的解最终得到全局问题的最终解[10-11]。ADMM对输入不敏感,还具有方法简单、可解释性高与收敛性易证明的优点。因而ADMM成为了分布式学习的主流方法。如DSPL(Distributed self-paced learning method)[12]使用ADMM与自步学习结合,IPA(Incremental plug-and-play ADMM)[13]将ADMM与可扩展即插即用算法结合,DP-ADMM(Differentially private ADMM)[14]将ADMM与差分隐私结合。这些分布式机器学习算法结合了ADMM之后,显著缩短训练时间的同时,达到高精度的性能,提高了原始算法对于大规模数据集的适用性。

ADMM引入辅助变量和等式约束$\boldsymbol{A}\boldsymbol{x}+\boldsymbol{B}\boldsymbol{y}=\boldsymbol{C}$,将原始目标函数$f(\boldsymbol{x})=\frac{1}{n}\sum_{i=1}^{n}f_i(\boldsymbol{x})$转化为

$$\begin{cases}\arg\min\limits_{x,y} f(\boldsymbol{x})+g(\boldsymbol{y})\\ \text{s.t.}\quad \boldsymbol{A}\boldsymbol{x}+\boldsymbol{B}\boldsymbol{y}=\boldsymbol{C}\end{cases} \tag{1}$$

式中:$f(\boldsymbol{x})$为关于$\boldsymbol{x}$的目标函数;$g(\boldsymbol{y})$为关于$\boldsymbol{y}$的目标函数;$\boldsymbol{A}\boldsymbol{x}+\boldsymbol{B}\boldsymbol{y}=\boldsymbol{C}$为约束条件。通过优化这个目标函数,模型最终会收敛到某一个全局性能指标的最优化或近似最优结果,从而解决了分布式中的一致性优化问题。

SLMC[7]由 Chen 于 2013 年提出。该模型将大间隔聚类(Maximum margin clustering, MMC)[15]与软聚类结合,在输出空间中寻找决策函数实现软聚类,同时将簇中心固定在预定的输出标记上。它兼有 MMC 方法和软聚类方法的优点,一方面具有簇间最大间隔的决策函数,另一方面又具有较强的聚类能力。它将每个示例的软赋值分配给各个簇(由标签表示),以学习内在的的数据结构。

该算法在输出空间进行聚类,而不是在数据空间中寻找一组聚类中心。具体来说,SLMC 将簇中心锚定为 C 个输出标记的预定义编码,计算决策函数和输出空间中数据的软隶属度。除此之外,其他一些编码策略,如正则单纯形顶点编码[16],也可以用于设计每个簇中心。另外,与 MMC 不同的是,该算法通过引入软学习原理,使每个实例都属于具有相应软隶属度的多个簇/聚类,比 MMC 更适合处理模糊的聚类赋值,并可通过软隶属度反映属于单个簇的程度。

设有数据集 $\boldsymbol{X}=\{\boldsymbol{x}_i\}_{i=1}^n$,$\boldsymbol{x}\in\mathbf{R}^d$,令 $f(\boldsymbol{x})=\boldsymbol{W}^{\mathrm{T}}\boldsymbol{\phi}(\boldsymbol{x})$ 表示 C 个聚类的决策函数,$\boldsymbol{W}\in\mathbf{R}^{d*C}$ 为权值矩阵。$\boldsymbol{U}=[u_{ci}]_{C*n}$ 表示软隶属度矩阵,$u_{ci}\in[0,1]$ 表示软隶属度。$\{l_1,l_2,\cdots,l_C\}$ 对应 C 簇的标记编码,其中 $\boldsymbol{l}_c\in\mathbf{R}^C$,第 c 个分量为 1,其余分量为 0。则相应的目标函数为

$$\begin{cases}\min\limits_{u_{ci},W}\dfrac{1}{2}\|\boldsymbol{W}\|^2+\dfrac{\lambda}{2}\sum\limits_{c=1}^{C}\sum\limits_{i=1}^{n}u_{ci}^2\|\boldsymbol{W}^{\mathrm{T}}\boldsymbol{\phi}(\boldsymbol{x}_i)-\boldsymbol{l}_c\|^2\\ \text{s.t.}\quad \sum\limits_{c=1}^{C}u_{ci}=1\\ \qquad 0\leqslant u_{ci}\leqslant 1\ \forall c=1,2,\cdots,C;\ i=1,2,\cdots,n\end{cases} \tag{2}$$

该算法通过最小化目标函数的第 2 项来最大化输出空间中簇间间隔[17]。此外,该算法通过相应的模糊隶属关系,将输出空间中给定示例与聚类或簇中心之间的距离之和(或者更具体地说,给定示例的分类是经过决策函数计算后的结果与簇编码间的距离)最小化,从而将输出空间中簇内的模糊散度最小化。

注意到 $\boldsymbol{l}_k$ 作为聚类中心独立于数据,因而更适合使用 ADMM 进行分布式优化。上面定义的问题在如下诸方面非常具有挑战性。首先,输入数据集可能太大,无法同时全部处理,因此需要对大型数据集使用可扩展的算法。其次,如果简单地分割数据集进行单机运算,则在迭代更新时也需要保证每个本地参数 $\boldsymbol{W}$ 的一致性,否则将无法进行训练。

在 SLMC 进行优化时,其判别函数可等价为 $f(\boldsymbol{x})=\sum\limits_{i=1}^{n}\hat{\boldsymbol{\alpha}}_i K(\boldsymbol{x}_i,\boldsymbol{x})$[18],其中 $K(\cdot)$ 为核函数,如径向基核(Radial basis function, RBF)。这种方法的主要缺点是随着数据集规模的增加,$\hat{\boldsymbol{\alpha}}$ 的数量也会增加,这会带来高额的计算代价。因此本文通过引入 RBF 核的显式特征映射近似来解决此问题,即用傅里叶变换的蒙特卡罗逼近 RBF 核的特征映射。

近似关系式可写为向量形式 $K(\boldsymbol{x}_i,\boldsymbol{x}_j)=R^{\mathrm{T}}(\boldsymbol{x}_i)R(\boldsymbol{x}_j)$,其中 $R(\boldsymbol{x})$ 为

$$R(\boldsymbol{x})=\frac{1}{\sqrt{D}}\left[\sin(\boldsymbol{v}_1^{\mathrm{T}}\boldsymbol{x}),\sin(\boldsymbol{v}_2^{\mathrm{T}}\boldsymbol{x}),\cdots,\sin(\boldsymbol{v}_D^{\mathrm{T}}\boldsymbol{x}),\cos(\boldsymbol{v}_1^{\mathrm{T}}\boldsymbol{x}),\cos(\boldsymbol{v}_2^{\mathrm{T}}\boldsymbol{x}),\cdots,\cos(\boldsymbol{v}_D^{\mathrm{T}}\boldsymbol{x})\right] \tag{3}$$

式中 $\boldsymbol{v}$ 为 RBF 核对应高斯分布中随机抽样的向量。显然,这种近似的精度会随着样本数量的增加而增

加。在实验中,将通过考虑精度与复杂度的权衡来选择一个适当的D。因此,判别函数就可近似为

$$\hat{f}(\boldsymbol{x})=\sum_{i=1}^{n}\boldsymbol{\alpha}_i R^{\mathrm{T}}(\boldsymbol{x}_i)R(\boldsymbol{x})=\hat{\boldsymbol{\theta}}^{\mathrm{T}}R(\boldsymbol{x}) \tag{4}$$

式中$\boldsymbol{\alpha}_i(1\leqslant i\leqslant n)$为系数。

优化变量的维数从N降至$2D$,减少了大量的计算复杂度。除此之外,D的选取与数据集大小无关,进一步方便了在实际中的应用。但注意到,傅里叶变换在不同采样下的映射不同,理论上可能会导致聚类性能的变化,但根据后面的实验表明,这种变化对聚类结果产生的影响可忽略,因而比较稳定。同时由于逼近精度所依赖的特征维度常常过高,为防止过拟合,还需要对系数矩阵添加稀疏化约束。

2 分布式稀疏软大间隔聚类

在式(2)中定义的问题不能分布式解决,因为每个节点无法共享模型参数$\hat{\boldsymbol{\theta}}_m$,所以需要解耦所有批次之间的关系。具体来说,为每一个终端引入不同的模型参数,并使用一个辅助变量$\boldsymbol{Z}$来确保所有模型参数的一致性,并对$\boldsymbol{Z}$施加$l_{2,1}$范数约束实现稀疏化,缓和过拟合。现在,式(2)可重新表述为

$$\begin{cases}\underset{U_m,\hat{\boldsymbol{\theta}}_m,\boldsymbol{Z}}{\arg\min}\sum\limits_{m=1}^{M}\sum\limits_{i=1}^{n_m}\sum\limits_{c=1}^{C}u_{ci,m}^2\|\hat{\boldsymbol{\theta}}_m^{\mathrm{T}}R(\boldsymbol{x}_{i,m})-\boldsymbol{l}_c\|^2+\lambda\|\boldsymbol{Z}\|_{2,1}\\ \text{s.t.}\sum\limits_{c=1}^{C}u_{ci,m}=1\\ \quad\hat{\boldsymbol{\theta}}_m-\boldsymbol{Z}=0\quad\forall m=1,2,\cdots,M\\ \quad 0\leqslant u_{ci,m}\leqslant 1\quad\forall c=1,2,\cdots,C;\ i=1,2,\cdots,n_m;\ m=1,2,\cdots,M\end{cases} \tag{5}$$

式中:m为节点数量;n_m为每个节点数据集的大小;$u_{ci,m}$为软隶属度;$\hat{\boldsymbol{\theta}}_m^{\mathrm{T}}R(\boldsymbol{x}_{i,m})$为近似后的目标函数;$\{l_1,l_2,\cdots,l_C\}$对应$C$簇的标记编码;$\lambda$为模型参数。

不同于原始问题,每个节点都有自己的模型参数$\hat{\boldsymbol{\theta}}_m$和$\sum\limits_{c=1}^{C}u_{ci,m}=1$的约束,同时还需确保模型参数$\hat{\boldsymbol{\theta}}_m$具有与辅助变量$\boldsymbol{Z}$相同的值。这样经过重新表述之后,每个终端都可以分布式优化模型参数。当约束得到严格满足时,新问题等价于原始问题。由此一来,对于$\boldsymbol{Z}$的稀疏化约束等价于对每个设备本地模型参数$\hat{\boldsymbol{\theta}}_m$的稀疏化约束,提高了模型的可解释性,就可以使用ADMM优化求解这个问题。

该目标函数的增广拉格朗日优化格式为

$$\begin{cases}\underset{U_m,\hat{\boldsymbol{\theta}}_m,\boldsymbol{Z}}{\arg\min}\sum\limits_{m=1}^{M}\sum\limits_{i=1}^{n_m}\sum\limits_{c=1}^{C}u_{ci,m}^2\|\hat{\boldsymbol{\theta}}_m^{\mathrm{T}}R(\boldsymbol{x}_{i,m})-\boldsymbol{l}_c\|^2+\lambda\|\boldsymbol{Z}\|_{2,1}+\sum\limits_{m=1}^{M}\mathrm{tr}(\boldsymbol{\alpha}_m^{\mathrm{T}}(\hat{\boldsymbol{\theta}}_m-\boldsymbol{Z}))+\dfrac{\rho}{2}\sum\limits_{m=1}^{M}\|\hat{\boldsymbol{\theta}}_m-\boldsymbol{Z}\|_{\mathrm{F}}^2\\ \text{s.t.}\sum\limits_{c=1}^{C}u_{ci,m}=1\\ \quad 0\leqslant u_{ci,m}\leqslant 1\quad\forall c=1,2,\cdots,C;\ i=1,2,\cdots,n_m;\ m=1,2,\cdots,M\end{cases} \tag{6}$$

式中:$\boldsymbol{\alpha}_m$为拉格朗日乘子;ρ为ADMM迭代的步长;$\mathrm{tr}(\cdot)$为矩阵的秩。

对于联合$(\hat{\boldsymbol{\theta}},\boldsymbol{U})$,该目标函数的优化问题非凸,不易求得全局最优解。其算法实现采用交替迭代策略,将其分解为两个子问题。子问题都只涉及单变量,从而可以优化求解。理论上可以保证整个迭代过程的收敛[19]。当固定$\boldsymbol{U}$矩阵即数据与类别之间的软隶属度矩阵之后,原本的非凸问题转变为凸问题,所以可通过计算得到一个闭式解。经过固定$\boldsymbol{U}$之后的目标函数为

$$\underset{\hat{\boldsymbol{\theta}}_m}{\arg\min}\sum_{m=1}^{M}\sum_{i=1}^{n_m}\sum_{c=1}^{C}u_{ci,m}^2\|\hat{\boldsymbol{\theta}}_m^{\mathrm{T}}R(\boldsymbol{x}_{i,m})-\boldsymbol{l}_c\|^2+\sum_{m=1}^{M}\mathrm{tr}(\boldsymbol{\alpha}_m^{\mathrm{T}}(\hat{\boldsymbol{\theta}}_m-\boldsymbol{Z}))+\frac{\rho}{2}\sum_{m=1}^{M}\|\hat{\boldsymbol{\theta}}_m-\boldsymbol{Z}\|_{\mathrm{F}}^2 \tag{7}$$

对 $\hat{\boldsymbol{\theta}}_m$ 求导，同时令求导结果为0。可得

$$\frac{\partial J_1}{\partial \hat{\boldsymbol{\theta}}_m}=2\sum_{i=1}^{n_m}\sum_{c=1}^{C}u_{ci,m}^2\left(R(\boldsymbol{x}_{i,m})R^{\mathrm{T}}(\boldsymbol{x}_{i,m})\hat{\boldsymbol{\theta}}_m-R(\boldsymbol{x}_{i,m})\boldsymbol{l}_c^{\mathrm{T}}\right)+\boldsymbol{\alpha}_m+\rho(\hat{\boldsymbol{\theta}}_m-\boldsymbol{Z})=0 \tag{8}$$

然后通过化简，即得

$$\hat{\boldsymbol{\theta}}_m^{k+1}=\left(2\sum_{i=1}^{n_m}\sum_{c=1}^{C}u_{ci,m}^{2,k}R(\boldsymbol{x}_{i,m})R^{\mathrm{T}}(\boldsymbol{x}_{i,m})+\rho\boldsymbol{I}\right)^{-1}\left(\rho\boldsymbol{Z}^k-\boldsymbol{\alpha}_m^k+2\sum_{i=1}^{n_m}\sum_{c=1}^{C}u_{ci,m}^{2,k}R(\boldsymbol{x}_{i,m})\boldsymbol{l}_c^{\mathrm{T}}\right) \tag{9}$$

固定 $\hat{\boldsymbol{\theta}}_m$，该目标函数的优化问题就改变为

$$\begin{cases}\underset{U_m}{\arg\min}\sum_{m=1}^{M}\sum_{i=1}^{n_m}\sum_{c=1}^{C}u_{ci,m}^2\|\hat{\boldsymbol{\theta}}_m^{\mathrm{T}}R(\boldsymbol{x}_{i,m})-\boldsymbol{l}_c\|^2\\ \text{s.t.}\sum_{c=1}^{C}u_{ci,m}=1\\ \quad 0\leqslant u_{ci,m}\leqslant 1\quad \forall c=1,2,\cdots,C;\ i=1,2,\cdots,n_m;\ m=1,2,\cdots,M\end{cases} \tag{10}$$

然后对转变后的函数采用拉格朗日乘子法，将约束条件函数与原函数联立，从而求出使原函数取得极值的 $\boldsymbol{U}$ 的闭式解。将其转化为

$$J_2(u_{ci,m})=\sum_{i=1}^{n_m}\sum_{c=1}^{C}u_{ci,m}^2\|\hat{\boldsymbol{\theta}}_m^{\mathrm{T}}R(\boldsymbol{x}_{i,m})-\boldsymbol{l}_c\|^2-\sum_{i=1}^{n}\gamma_i\left(\sum_{c=1}^{C}u_{ci,m}-1\right) \tag{11}$$

令 $J_2(\boldsymbol{U})$ 对 u_{ki} 求导，并令结果为0，得到

$$\frac{\partial J_2}{\partial u_{ci,m}}=2\|\hat{\boldsymbol{\theta}}_m^{\mathrm{T}}R(\boldsymbol{x}_{i,m})-\boldsymbol{l}_c\|^2u_{ci,m}-\gamma_i=0 \tag{12}$$

通过化简，能够求出闭式解为

$$u_{ci,m}=\gamma_i/2\|\hat{\boldsymbol{\theta}}_m^{\mathrm{T}}R(\boldsymbol{x}_{i,m})-\boldsymbol{l}_c\|^2 \tag{13}$$

但仍旧有一个约束条件，因此需要将约束条件加入，使其成立

$$u_{ci,m}=\frac{1/\|\hat{\boldsymbol{\theta}}_m^{\mathrm{T}}R(\boldsymbol{x}_{i,m})-\boldsymbol{l}_c\|^2}{\sum_{c=1}^{C}1/\|\hat{\boldsymbol{\theta}}_m^{\mathrm{T}}R(\boldsymbol{x}_{i,m})-\boldsymbol{l}_c\|^2} \tag{14}$$

对于辅助变量 $\boldsymbol{Z}$ 而言，有

$$\frac{\partial L}{\partial \boldsymbol{Z}}=2\lambda\boldsymbol{E}\boldsymbol{Z}-\sum_{m=1}^{M}\boldsymbol{\alpha}_m-\rho\left(\sum_{m=1}^{M}\hat{\boldsymbol{\theta}}_m-m\boldsymbol{Z}\right)=0 \tag{15}$$

式中 $\boldsymbol{E}$ 为 $\boldsymbol{E}_{(i,i)}=\dfrac{1}{2(\|\boldsymbol{Z}_{i*}\|_2)}$，$i=1,2,\cdots,C$。为了防止分母为0，在 $\boldsymbol{E}$ 分母处添加非负小量数。

因而 $\boldsymbol{E}_{(i,i)}=\dfrac{1}{2(\|\boldsymbol{Z}_{i*}\|_2)+\epsilon}$，$i=1,2,\cdots,C$。

化简可得

$$\boldsymbol{Z}^{k+1}=(2\lambda\boldsymbol{E}+\rho m\boldsymbol{I})^{-1}\left(\rho\sum_{m=1}^{M}\hat{\boldsymbol{\theta}}_m^{k+1}+\sum_{m=1}^{M}\boldsymbol{\alpha}_m^k\right) \tag{16}$$

拉格朗日乘子 $\boldsymbol{\alpha}_m$ 迭代更新式为

$$\boldsymbol{\alpha}_m^{k+1}=\boldsymbol{\alpha}_m^k+\rho(\hat{\boldsymbol{\theta}}_m^{k+1}-\boldsymbol{Z}^{k+1}) \tag{17}$$

ADMM的停止条件由第 k 次迭代的原始残差和对偶残差的平方范数决定，有

$$\|\boldsymbol{r}^k\|_2^2=\sum_{m=1}^{M}\|\hat{\boldsymbol{\theta}}_m^k-\boldsymbol{Z}^k\|_{\mathrm{F}}^2 \tag{18}$$

$$\|\boldsymbol{s}^k\|_2^2=m\rho^2\|\boldsymbol{Z}^k-\boldsymbol{Z}^{k-1}\|_{\mathrm{F}}^2 \tag{19}$$

式中:$\boldsymbol{r}^k$为原始残差;$\boldsymbol{s}^k$为对偶残差。

算法伪代码如算法1所示,模型权重和拉格朗日乘数$\boldsymbol{\alpha}_m$都可以在每个节点迭代更新,通过数据通讯实现分布式计算。注意到模型权重$\hat{\boldsymbol{\theta}}_m$的更新是模型训练过程中耗时最长的计算,因此分布式计算$\hat{\boldsymbol{\theta}}_m$可以显著提高算法的效率。本文同时在迭代更新中引入了ρ的动态变化来加快模型的收敛速度。当$r>10$ s时,更新$\rho\leftarrow 2\rho$;当$r<10$ s时,更新$\rho\leftarrow\rho/2$。对数据的类别预测有两种:一是通过软隶属度矩阵;二是通过决策函数,两种方法预测的结果基本相同。

算法1 分布式软大间隔聚类算法

输入:λ、ρ,停止参数ε_s、ε_r,节点数M、映射特征维度D、输入数据X;

输出:软隶属度矩阵$\boldsymbol{U}_m$;

(1) Each learner constructs the random feature map $R_m(x)$ with the same seed

(2) Initialize $\boldsymbol{U}_m$ by FCM, $k=0$

(3) Update $\boldsymbol{Z}^{k+1}$ in the central server by Eq.(16)

(4) Update $\hat{\boldsymbol{\theta}}_m^{k+1}$ in parallel by Eq.(9)

(5) Update $\boldsymbol{U}_m^{k+1}$ in parallel by Eq.(14)

(6) Update $\boldsymbol{\alpha}_m^{k+1}$ in parallel by Eq.(17)

(7) Update primal and dual residuals $\boldsymbol{r}^{k+1}$ and $\boldsymbol{s}^{k+1}$ in the central server by Eq.(18) and Eq.(19)

(8) $k=k+1$

(9) If not $\|\boldsymbol{r}^{k+1}\|_2^2<\varepsilon_r$ and $\|\boldsymbol{s}^{k+1}\|_2^2<\varepsilon_s$ then go step 3

(10) Output $\boldsymbol{U}_m$ at each learner

3 相关数据集与实验

3.1 实验设置

本节同时对单机软大间隔聚类和分布式FCM算法进行对比分析。对FCM进行分布式优化的方法是简单地在每个迭代中并行化簇分配和簇平均计算。

分布式FCM聚类的优化问题表述为

$$\min\sum_{m=1}^{M}\sum_{i=1}^{n_m}\sum_{c=1}^{C}u_{ci,m}^2\|\boldsymbol{\phi}(\boldsymbol{x}_{i,m})-\boldsymbol{\phi}(\boldsymbol{v}_{c,m})\|^2 \tag{20}$$

式中:$\boldsymbol{\phi}(\cdot)$为核函数;$\boldsymbol{v}_{c,m}$为聚类中心。

本文通过对集合$\{0.1,0.5,1,5,10\}$和$\{10^{-2},10^{-1},1,10\}$的穷举搜索来设置正则化和核参数。本文对各算法执行了20次独立实验,在每次实验中使用最优的核与参数设置,并记录实验结果,最后取平均值。实验中所有核近似的参数D均设置为50,所有的实验都在1台具有英特尔(R)核心(TM)四核处理器(i7CPU@3.6 GHz)和16 GB内存的64位机器上进行。

3.2 数据集

对于时间占用对比,通过设置不同的计算节点数量,对比100万数据集的总共训练时间。数据集为USCensus1990即美国1990年的人口普查数据和PokerHandDataset。通过采样得到大小为100万的数据集,将其根据节点数量均分,同时保证每个子集样本的一致性分布,这些数据集的属性如表1所示。

对于其余的性能指标对比，一共有3个有代表性的小数据集，其属性如表2所示。

表1 大规模数据集属性

Table 1 Large-scale dataset attributes

数据集	数据个数	特征数量	目标聚类个数
USCensus1990	1 000 000	68	10
Pokerhand	1 000 000	10	10
Dry bean	13 611	17	7
HTRU2	17 898	9	2

表2 小规模数据集属性

Table 2 Small-scale dataset attributes

数据集	数据个数	特征数量	目标聚类个数
Arrhythmia	452	279	13
Ecoli	336	8	6
Image segmentation	2 310	19	7

3.3 评估指标

虽然本文提出的是无监督分布式聚类算法，但在训练时可通过带类别标签的数据集进行训练。具体来说，在训练时将类别信息剔除，然后将聚类的结果与真实标签相比较。因此可采用聚类准确度(Clustering accuracy, CA)[15,20]来对算法进行性能评价，有

$$\mathrm{CA}=\frac{1}{n}\sum_{c=1}^{C}\max_{t=1,2,\cdots,C}T(F_c,Y_t) \tag{21}$$

式中：F_c代表第c聚类标签，而Y_t代表真实标签。$T(F_c,Y_t)$则代表属于Y_t真实类别而被预测为F_c类别的数据个数。通过比较预测聚类的标签与数据集中的真实标签的方式来确定聚类的准确度。

此外，由于本文提出的是分布式聚类算法，因此也将时间占用对比纳入评估指标。具体来说，对于同等规模的数据集，设置不同的计算节点进行分布式学习，对比完成模型训练的时间。

3.4 实验分析

图1显示随着节点数的增加，在同等数据集大小的情况下，当节点数为1时(即单机情况下)，需要消耗大量的时间进行训练。但当节点逐渐增长时，训练时间急剧减少，这个结果符合直觉，即分布式学习能够大幅降低算法的时间成本。与此同时，DS-SLMC引入核近似方法，采用随机傅里叶特征变换逼近核函数，避免了计算核矩阵时2次于数据数的时间代价，从而进一步减少训练时间。这个实验证明DS-SLMC在保持核技巧优秀划分性能的前提下，可以同时降低计算核矩阵的时间代价，从而达到良好的加速比及扩展性能，避免了单机算法计算代价过大的缺点，适用于大数据或分布式存储数据的训练。

表3显示了聚类性能的对比。虽然DS-SLMC的聚类精度相比单机算法低，但依然在可接受的范围内。(1)因为ADMM策略，整个分布式系统虽然使用分割过的数据集进行学习，但通过不同节点间的数据通讯，每个节点都从其余节点习得足够的知识。(2)因为DS-SLMC在SLMC的基础上引入稀疏化，从而避免了算法的过拟合问题。与同样分布式化的经典聚类算法FCM相比，DS-SLMC都取得了更为优异的结果，因此也验证了单机上性能有效的算法改造后相对优势仍能获得保持。因为分布式聚类的最终性能取决于其基聚类性能，所以要选择性能优异的基聚类进行分布式优化，这也是本文着力于对SLMC进行分布式优化改进的原因之一。

由前两个实验可知，DS-SLMC尽管是针对大数据的分布式聚类算法，但不管在何种规模的数据集下依然可以完成训练，达到有竞争力的性能。即将那些性能相对优的已有简单算法适应到分布式学习环境，提高其伸缩性能，以重用这些算法使之发挥更大的潜能。

如上文所说，DS-SLMC引入的核近似包含了傅里叶变换，由于不同采样下的映射不同，理论上可

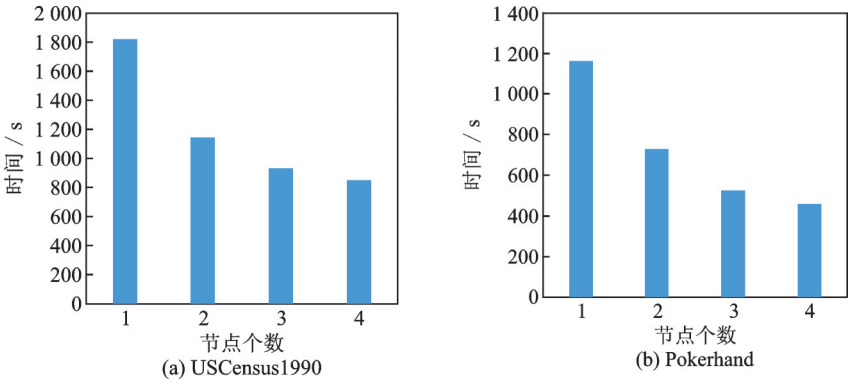

图1 不同节点数的时间占用

Fig.1 Time occupation of different number of nodes

表3 在不同数据集上的聚类准确度CA结果

Table 3 Clustering accuracy evaluation results on different datasets

数据集	FCM	Distributed FCM	SLMC	DS-SLMC
Pokerhand	0.421 5	0.414 8	0.737 8	0.729 1
Dry bean	0.847 9	0.839 1	0.972 2	0.971 9
HTRU2	0.927 7	0.923 6	0.929 4	0.929 6
Arrhythmia	0.369 0	0.360 9	0.546 5	0.548 6
Ecoli	0.591 7	0.583 1	0.796 4	0.785 7
Image segmentation	0.540 9	0.529 7	0.647 3	0.642 8

能会导致聚类性能的变化,因此本文设置了基于不同初始化变换种子的对比试验。实验随机采样出100个随机数种子,并对每个种子独立运行10次实验,取聚类准确度百分数的平均值作为该种子的聚类性能。对不同种子对比时,取方差作为对比的最终结果。表4结果表明,这种变化无法对聚类结果产生较大的影响。因而验证了该算法在不同采样下聚类效果依旧是稳定的。

表4 不同种子的方差对比

Table 4 Comparison of variance between different seeds

数据集	方差
Arrhythmia	0.012 1
Ecoli	0.040 0
Image segmentation	0.181 3

4 结束语

本文提出了一种分布式软大间隔聚类算法,将单机聚类算法扩展到分布式版本。为了实现这个目标,本文将原始的聚类问题重新定义为分布式判别函数,采用随机傅里叶特征变换逼近核函数,同时将变换特征的稀疏性嵌入SLMC并基于ADMM优化解决了并行处理不同终端的问题。而实验结果也表明本文提出算法有较优的性能。具体来说,该算法在尽量降低误差的情况下,达到了期望的加速比。此外也通过实验验证了该算法的收敛性,以及在不同数据集上都拥有良好的聚类性能。经过分布式改造后,SLMC适应到大数据或分布式存储数据的环境,大幅加速了训练过程,大大降低模型训练的时间代价,提高了在大数据集上的适用性与可扩展性。

参考文献：

[1] HUANG P, ZHANG D. Locality sensitive C-means clustering algorithms[J]. Neurocomputing, 2010, 73(16/17/18): 2935-2943.

[2] NG A Y, JORDAN M I, WEISS Y. On spectral clustering: Analysis and an algorithm[C]//Proceedings of Advances in Neural Information Processing Systems. Cambridge, USA: MIT Press, 2001: 345-352.

[3] BALCAN M F, EHRLICH S, LIANG Y. Distributed k-means and k-median clustering on general topologies[C]//Proceedings of International Conference on Neural Information Processing Systems. [S.l.]:Curran Associates Inc., 2013: 1995-2003.

[4] HERSHEY J R, CHEN Z, LE ROUX J, et al. Deep clustering: Discriminative embeddings for segmentation and separation [C]//Proceedings of 2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). [S.l.]: IEEE, 2016: 31-35.

[5] MACQUEEN J. Some methods for classification and analysis of multivariate observations[C]//Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability. Berkley: [s.n.], 1967: 281-297.

[6] BEZDEK J C. Pattern recognition with fuzzy objective function algorithms[M]. NewYork: Plenum Press, 1981.

[7] WANG Y, CHEN S. Soft large margin clustering[J]. Information Sciences, 2013, 232: 116-129.

[8] WANG Y, NIE L, LI Y, et al. Soft large margin clustering for unsupervised domain adaptation[J]. Knowledge-Based Systems, 2020, 192: 105344.

[9] SALTOS R, WEBER R. A rough-fuzzy approach for support vector clustering[J]. Information Sciences, 2016, 339: 353-368.

[10] GLOWINSKI R, MARROCCO A. On the solution of a class of non linear Dirichlet problems by a penalty-duality method and finite elements of order one[C]//Proceedings of Optimization Techniques IFIP Technical Conference. Berlin, Heidelberg: Springer, 1975: 327-333.

[11] GABAY D, MERCIER B. A dual algorithm for the solution of nonlinear variational problems via finite element approximation [J]. Computers & Mathematics with Applications, 1976, 2(1): 17-40.

[12] ZHANG X, ZHAO L, CHEN Z, et al. Distributed self-paced learning in alternating direction method of multipliers[EB/OL]. (2018-07-06)[2022-7-18]. https://arXiv.org/pdf/1807.02234.

[13] SUN Y, WU Z, XU X, et al. Scalable plug-and-play ADMM with convergence guarantees[J]. IEEE Transactions on Computational Imaging, 2021, 7: 849-863.

[14] SHANG F, XU T, LIU Y, et al. Differentially private ADMM algorithms for machine learning[J]. IEEE Transactions on Information Forensics and Security, 2021, 16: 4733-4745.

[15] XU L, NEUFELD J, LARSON B, et al. Maximum margin clustering[J]. Advances in Neural Information Processing Systems, 2004, 17: 1537-1544.

[16] AN S, LIU W, VENKATESH S. Face recognition using kernel ridge regression[C]//Proceedings of 2007 IEEE Conference on Computer Vision and Pattern Recognition. [S.l.]: IEEE, 2007: 1-7.

[17] SUYKENS J A K, VANDEWALLE J. Least squares support vector machine classifiers[J]. Neural Processing Letters, 1999, 9(3): 293-300.

[18] BELKIN M, NIYOGI P, SINDHWANI V. Manifold regularization: A geometric framework for learning from labeled and unlabeled examples[J]. Journal of Machine Learning Research, 2006, 7(11): 2399-2434.

[19] GORSKI J, PFEUFFER F, KLAMROTH K. Biconvex sets and optimization with biconvex functions: A survey and extensions[J]. Mathematical Methods of Operations Research, 2007, 66(3): 373-407.

[20] ZHANG K, TSANG I W, KWOK J T. Maximum margin clustering made practical[J]. IEEE Transactions on Neural Networks, 2009, 20(4): 583-596.

作者简介：

谢云轩(1997-)，**通信作者**，男，硕士，研究方向：分布式机器学习、联邦学习，E-mail：rfnoah@nuaa.edu.cn。

陈松灿(1962-)，男，教授，博士生导师，研究方向：机器学习及其应用。

(编辑：刘彦东)