基于云计算平台的仿生优化聚类数据挖掘算法

申燕萍[1] 顾苏杭[2] 郑丽霞[3]

(常州工业职业技术学院信息工程学院 江苏 常州 213164)[1]

(江南大学数字媒体学院 江苏 无锡 214122)[2] (东南大学微电子学院 南京 210096)[3]

摘 要 为了提高云计算平台数据挖掘的有效性以及数据聚类的性能,采用仿生优化算法与相似聚类相结合的方法来实现云计算平台数据聚类。在相似聚类的优化函数求解过程中,采用狼群优化算法,以头狼的位置来确定聚类中心点,从而实现类别中心点的优化与更新。文中分别采用 PBM 和 DB 聚类效果评价方法来对聚类效果进行检验,在满足预设评价标准的情况下,不断进行狼群优化和相似聚类计算,直到达到聚类指标要求为止。经过实验证明,相比一般聚类算法,狼群优化的聚类算法对数据量大且数据维度高的云计算平台数据聚类效果更好,收敛速度更快。

关键词 云计算平台,仿生优化,狼群算法,聚类,评价指标

中图法分类号 TP311 **文献标识码** A **DOI** 10.11896/jsjkx.190800042

Bionic Optimized Clustering Data Mining Algorithm Based on Cloud Computing Platform

SHEN Yan-ping[1] GU Su-hang[2] ZHENG Li-xia[3]

(School of Information Engineering, Changzhou Institute of Industry Technology, Changzhou, Jiangsu 213164, China)[1]

(School of Digital Media, Jiangnan University, Wuxi, Jiangsu 214122, China)[2]

(Microelectronics College, Southeast University, Nanjing 210096, China)[3]

Abstract In order to improve the validity of cloud computing platform data mining and the performance of data clustering, this paper combined bionic optimization algorithm with similar clustering to achieve cloud computing platform data clustering. In the process of solving the optimization function of similar clustering, wolf swarm optimization algorithm is used to locate the head wolf position to determine the cluster centers, so as to optimize and update the category centers. PBM and DB clustering effect evaluation methods were used to test the clustering effect, and wolf swarm optimization and similar clustering calculation were carried out continuously until the requirements of clustering index are met. Experiments results show that, compared with general clustering algorithms, wolf swarm optimization clustering algorithm has better clustering effect and faster convergence speed for cloud computing platform with large data volume and high data dimension.

Keywords Cloud computing platform, Bionic optimization, Wolf swarm algorithm, Clustering, Evaluation index

云计算作为一种数据服务模式,逐渐被各行业所接受。在各种差异化结构数据进入云计算平台后,如何从这些大规模数据中提取有价值的信息,成为了云计算深化应用的关键[1],是云计算平台在各领域继续发展的基础。例如:通过云计算电商平台,根据用户及商品数据,可以分析潜在用户的消费习惯及不同层次商品的受欢迎程度;通过云计算旅游平台,可以根据用户搜索习惯及体验反馈,进一步优化旅游路线和旅游服务;通过云计算社交平台,可以根据用户访问情况,为用户推荐更多的好友及定制服务。这些基于云计算平台的数据分析离不开大规模的数据挖掘[2],而云计算平台数据挖掘技术发展的关键点则是聚类[3]。

云计算平台数据量大,而且数据种类差异较大,一般聚类方法在数量和维度较小的情况下性能表现良好[4],但是在数据量较大时则会暴露出一些问题,特别是对于高维数据,聚类效果并不理想。因此,本文借助仿生优化算法处理复杂数据的优势,将其运用于云计算平台的数据挖掘过程中,以更好地在海量的云计算数据中提取更有价值的数据。仿生优化算法的种类较多,本文推荐狼群算法,因为它能在处理大数据量时快速收敛并得到较好的最优解。

1 狼群优化算法

设数据集合为 $M'=[m_{11}m_{12}\cdots m_{1M}m_{21}\cdots m_{IM}]$,将该集合

到稿日期:2019-08-09 返修日期:2019-09-24 本文受国家自然科学基金(青年基金)(61805036),国家自然科学基金面上项目(61376029)资助。

申燕萍(1979—),硕士,讲师,CCF 会员,主要研究方向为软件技术、数据挖掘,E-mail:59682819@qq.com(通信作者);**顾苏杭**(1989—),博士,工程师,主要研究方向为机器学习、人工智能;**郑丽霞**(1979—),博士,讲师,主要研究方向为微电子。

记为狼群中头领狼的坐标位置[5-6]，设狼群总量为 N，其他 $N-1$ 只狼的坐标位置描述为 $Q_l=[q_{l1}\ q_{l2}\cdots q_{lk}\cdots q_{lJ}]$，其中 $J=I\times M$，$k=1,2,\cdots,J$，$l=1,2,\cdots,N-1$。

将数据集合中的部分数据作为狼群游走的方向狼，它们的作用是探索游走的方向，然后进行模拟训练。设部分数据集合的总量为 T_{mm}，T_{mm} 个数据集合游走的步长为：

$$Step_Y(k)=|\max_k-\min_k|/S \tag{1}$$

其中，S 为移动步长权重。

位置更新计算的方法为[7]：

$$q_{i,k}^g=q_{i,k}+\sin(2\pi\times g/h)\times Step_Y(k) \tag{2}$$

其中，$i=1,2,\cdots,T_{mm}$，h 表示移动方向个数，$g=1,2,\cdots,h$。

除了方向狼之外，其他狼的运动步长和更新方式分别为[8]：

$$Step_B(k)=2\times|\max_k-\min_k|/S \tag{3}$$

$$q_{i,k}^{z+1}=q_{i,k}^z+Step_B(k)\cdot[d_k-q_{i,k}^z]/|d_k-q_{i,k}^z| \tag{4}$$

其中，$i=1,2,\cdots,N-T_{mm}-1$，$d_k(d_k\in(0,D_k))$为某一非方向狼和头领狼之间的距离。

$$D_k=\frac{1}{J\times\omega}\times\sum_{k=1}^{J}|\max_k-\min_k| \tag{5}$$

其中，ω 为距离权重，它是不同数据集合之间进行分类的重要依据。

当狼群到达目的地后，听从头领狼号令进行围攻，此时的运动步长和位置更新计算方式为[9]：

$$Step_W(k)=|\max_k-\min_k|/(2\times S) \tag{6}$$

$$q_{i,k}^{z+1}=q_{i,k}^z+\lambda\cdot Step_W(k)\cdot|d_k-q_{i,k}| \tag{7}$$

其中，$z=1,2,\cdots,N-1$，$\lambda\in[-1,1]$为随机数。当执行到预设迭代次数时算法停止，$q_{i,k}^{z+1}$ 为此时头狼的位置。

2 聚类方法

2.1 相似聚类

一般根据中心点与其他点的距离来对数据类别进行相似性判断。将两点距离与预设阈值进行对比，判断该点是否与中心点属于同一类别[10]，在实际操作过程中并不是直接计算两点之间的距离，而是引入高斯函数辅助计算相似度，这样能够根据高斯分布函数图更直观地判断类别的聚合程度及同类别数据点的分布情况。设某中心点 x_i，其他点与中心点的相似性关系的计算方法为[11]：

$$S_{ij}=\begin{cases}\exp[-\dfrac{\|x_i-x_j\|^2}{2\sigma^2}], & i\neq j\\ 0, & i=j\end{cases} \tag{8}$$

在聚类过程中，为了对数据集中所有的点都进行有效分类，并且使分类后所有的数据点到各类中心点的距离最小，建立如下优化函数：

$$\varepsilon=\sum_i\|x_i-\sum_{j,x_j\in\hat{N}(x_i)}S_{ij}x_j\|^2 \tag{9}$$

其中，$x_j\in\hat{N}(x_i)$表示 x_j 为 N 个样本点中除了 x_i 的所有节点，且有：

$$\sum_{j,x_j\in N(x_i)}S_{ij}x_j=1,S_{ij}\geqslant0 \tag{10}$$

$$\begin{aligned}\varepsilon&=\sum_i\|x_i-\sum_{j,x_j\in N(x_i)}S_{ij}x_j\|^2\\&=\|\sum_{j,x_j\in N(x_i)}S_{ij}(x_i-x_j)\|^2\\&=\sum_{j,x_j\in N(x_i)}S_{ij}S_{ik}(x_i-x_j)^{\mathrm{T}}(x_i-x_k)\end{aligned} \tag{11}$$

其中，x_k 表示第 k 个节点。为了简化式(11)，令：

$$G_{ij}=(x_i-x_j)^{\mathrm{T}}(x_i-x_k) \tag{12}$$

将式(12)代入式(11)得：

$$\varepsilon=\sum_{j,x_j\in N(x_i)}S_{ij}G_{ij}S_{ik} \tag{13}$$

因此，关于聚类优化的数学描述为求解式(14)：

$$\begin{aligned}&\min\ S_{ij}\sum_{j,k,x_j,x_k\in N(x_i)}S_{ij}G_{ij}S_{ik}\\&\text{s.t.}\ \sum_jS_{ij}=1,S_{ij}\geqslant0\end{aligned} \tag{14}$$

求解出 S_{ij} 后，可以得到相似度矩阵，用以判定 x_i 与 x_j 是否属于同类。

2.2 相似聚类与狼群算法的结合

在相似聚类的过程中，主要求解式(9)的最优解。考虑到云计算平台数据量大且维度高的特点，在求解所有样本点与中心点的距离的过程中，在每次中心点更新之后需要重新执行式(9)，此时在预设迭代次数条件下不容易得到全局最优解，因此在聚类过程中需引入高效算法来快速得到全局最优解。

相似聚类与狼群算法相结合的步骤为：

1)在初始化 K 个中心点后，建立聚类模型；

2)根据式(8)和式(9)，得到 K 个中心点的优化函数；

3)在式(9)中，根据狼群算法进行优化，算法迭代停止时头狼的位置即为聚类的中心点，从而可以确定 K 个中心点的位置；

4)根据 K 个中心点求解式(14)的最优 S_{ij}；

5)将 S_{ij} 代入式(9)后，继续采用狼群算法更新中心点位置；

6)不断重复更新中心点位置，直到达到聚类要求为止。

2.3 聚类结果评价

聚类结果评价指标的选择会影响聚类算法的适用度[12]。在云计算平台下，数据差异性大，维度高，使得数据挖掘的难度有所提升，而聚类结果评价会直接影响数据挖掘的效果，在不同的应用需求下，聚类结果评价的侧重点也有差别。目前，常用的聚类评价方法主要有 PBM-index 和 DB-index[13-14]，本文主要使用这两种评价方法。

PBM-index 评价方法的数学描述为：设样本被分为了 K 个类别，共有 K 个中心点，第 k 个中心点为 c_k，μ_{kj} 表示 k，j 两个点的关联因子，数学表示为$[\mu_{kj}]_{K\times n}$。则有：

$$E_K=\sum_{k=1}^{K}\sum_{j=1}^{n}\mu_{kj}\|x_j-c_k\| \tag{15}$$

$$D_K=\max_{i,j=1}^{K}\|c_i-c_j\| \tag{16}$$

其中，E_K 表示所有节点 x_j 距离第 k 个中心点的距离，D_K 为所有 K 个中心点中彼此间距离最大的两个中心点的距离。

根据式(15)和式(16)可得聚类评价方法为：

$$PBM(K)=(\frac{1}{K}\times\frac{E_1}{E_K}\times D_K)^{\sigma} \tag{17}$$

其中，E_1 表示所有节点 x_j 距离第 1 个中心点的距离；σ 表示评价因子，取值范围为 $\sigma\geqslant1$。

DB-index 评价方法的数学描述为：设样本被分为了 K 个类别，共有 K 个中心点，其中第 k 类的中心点为 c_k，第 k' 类的中心点为 $c_{k'}$，则评价计算公式为：

$$DB(K)=\frac{1}{K}\times\sum_{k}^{K}\max_{k\neq k'}\{\frac{\sum_{x\in c_k}\|x-c_k\|+\sum_{x\in c_{k'}}\|x-c_{k'}\|}{\|c_k-c_{k'}\|}\} \tag{18}$$

聚类优势与 PBM 值成正比，而与 DB 值成反比。

3　实例仿真

为了验证仿生优化聚类算法的性能，将其与常用的 K-means 算法、Mean-shift 算法和 DBSCAN 算法进行性能对比[15-16]。实验采用 Matlab 进行实例仿真。

3.1　聚类性能的 PBM 和 DB 评价

仿真数据对象为 UCI 中的 Iris 数据集，选取 $N=500$ 个样本进行仿真，相似聚类初始值 K 设为 3，狼群算法初值 $\lambda=0.5$，$T_{mm}=100$。实验进行狼群相似聚类，聚类的可视化结果如图 1 所示。

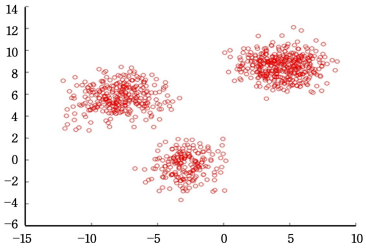

图 1　$K=3$ 时数据样本的聚类可视化

Fig. 1　Clustering visualization results of data sample when $K=3$

从图 1 可以看出，500 个样本被较好地分为了 3 类，虽然有 2 个类别的中心点距离较近，但是不影响 500 个样本点均有归属类，没有出现孤立样本点。

3.1.1　PBM 性能评价

虽然采用狼群优化完成了聚类，但是图 1 无法反映出聚类的数值指标，也不能反映出本文算法相比其他聚类算法的优势。下面以 PBM 评价指标为仿真对象，通过不同的聚类算法对不同样本容量及不同属性个数的数据样本进行仿真。仿真迭代结束后，根据式(17)分别对每种算法的每一类情况计算 PBM 的最大值(Max)、最小值(Min)和均值(Mean)，具体结果如表 1 所列，其中 N 表示样本集合，P 表示样本属性。

表 1　$P=10$ 时不同样本容量的 PBM 性能

Table 1　PMB performance of different sample sizes when $P=10$

	PBM	K-means	Mean-shift	DBSCAN	狼群聚类
$N=2000$	Max	0.07231	0.07296	0.07275	0.07368
	Mean	0.07043	0.07182	0.06912	0.07241
	Min	0.06801	0.06833	0.06611	0.07012
$N=10000$	Max	0.05171	0.05266	0.03933	0.05278
	Mean	0.04768	0.05185	0.03577	0.05103
	Min	0.03701	0.04778	0.03309	0.05011
$N=50000$	Max	0.02844	0.02853	0.02763	0.03172
	Mean	0.02817	0.02823	0.02623	0.02863
	Min	0.01762	0.01793	0.01543	0.01810
$N=100000$	Max	0.00944	0.00873	0.00763	0.01172
	Mean	0.00899	0.00823	0.00623	0.01012
	Min	0.00762	0.00793	0.00543	0.00877

选择 Vehicle 数据集中每个样本的 10 个属性进行聚类分析，从表 1 可以得出，随着训练样本数量的增加，训练样本的模型复杂度有所提高，所有聚类算法计算得到的 PBM 值均在下降。狼群算法在计算中心点位置时，受运算时间及预设迭代次数的影响，PBM 性能可能下降，但相比其他 3 种聚类算法，狼群聚类算法的 PBM 值更大，当样本数量达到 100000 时，只有本文算法的 PBM 均值大于 0.1，相较而言，狼群聚类算法在 PBM 的性能方面更优。

在样本容量固定为 50000 时，Vehicle 数据集中不同属性个数所表现的 PBM 性能的统计结果如表 2 所列。

表 2　$N=50000$ 时不同数据属性个数的 PBM 性能

Table 2　PBM performance of different numbers of data attributes when $N=50000$

	PBM	K-means	Mean-shift	DBSCAN	狼群聚类
$P=4$	Max	0.02835	0.02846	0.02757	0.03183
	Mean	0.02811	0.02817	0.02646	0.02869
	Min	0.01769	0.01779	0.01542	0.01812
$P=10$	Max	0.02833	0.02847	0.02757	0.03182
	Mean	0.02809	0.02818	0.02647	0.02868
	Min	0.01768	0.01789	0.01541	0.01811
$P=15$	Max	0.02827	0.02842	0.02751	0.03159
	Mean	0.02779	0.02805	0.02637	0.02856
	Min	0.01724	0.01721	0.01504	0.01799
$P=20$	Max	0.01737	0.01902	0.01716	0.02559
	Mean	0.01237	0.01305	0.01231	0.02056
	Min	0.00834	0.00921	0.00704	0.01684

从表 2 可以看出，随着样本属性个数的增加，所有聚类算法的 PBM 性能均在下降。在相同实验条件下，狼群聚类算法的 PBM 值高于其他 3 种聚类算法得到的 PBM 值。在 P 不大于 15 的情况下，PBM 值变化很小；而当 $P=20$ 时，4 种聚类算法的 PBM 性能均有明显下降。因此，在实际应用中，可以考虑将样本属性进行优化，将属性个数控制在 15 以内，以便获得更好的 PBM 性能。

3.1.2　DB 性能评价

下面以 DB 评价指标为仿真对象，进行实例仿真。不同样本容量的 DB 性能的结果如表 3 所列。

表 3　$P=10$ 时不同样本容量的 DB 性能

Table 3　DB performance of different sample sizes when $P=10$

	DB	K-means	Mean-shift	DBSCAN	狼群聚类
$N=2000$	Max	0.7137	0.7078	0.6521	0.6392
	Mean	0.6322	0.6412	0.6158	0.6101
	Min	0.6041	0.5935	0.5764	0.5722
$N=10000$	Max	0.7008	0.6735	0.6321	0.6019
	Mean	0.6317	0.6234	0.6042	0.5812
	Min	0.6038	0.5895	0.5629	0.5561
$N=50000$	Max	0.6078	0.5877	0.5646	0.5521
	Mean	0.5234	0.5032	0.4837	0.4237
	Min	0.4706	0.4629	0.4293	0.3873
$N=100000$	Max	0.6079	0.5871	0.5639	0.5518
	Mean	0.5232	0.5033	0.4836	0.4239
	Min	0.4707	0.4631	0.4291	0.3872

从表 3 可以看出，随着样本数量的增加，4 种聚类算法的 DB 值均在下降，这表明 4 种算法的 DB 性能均在提升，但在相同的实验条件下，狼群优化算法的 DB 性能均优于其他 3 种聚类算法。对比发现，样本容量从 50000 增加至 100000 时，本文算法的 DB 性能变化不大，DB 值趋于稳定。

在样本容量固定为 50000 时，不同属性个数所表现的 DB 性能的统计结果如表 4 所列。

表 4 不同数据属性个数的 DB 性能

Table 4 DB Performance of Different Numbers of Data Attributes

	DB	K-means	Mean-shift	DBSCAN	狼群聚类
$P=5$	Max	0.6024	0.5857	0.5621	0.5517
	Mean	0.5215	0.5006	0.4817	0.4214
	Min	0.4671	0.4609	0.4276	0.3851
$P=10$	Max	0.6071	0.5875	0.5643	0.5529
	Mean	0.5231	0.5029	0.4832	0.4235
	Min	0.4711	0.4631	0.4291	0.3872
$P=15$	Max	0.7219	0.7017	0.6691	0.6221
	Mean	0.6487	0.6331	0.6041	0.5887
	Min	0.5932	0.5653	0.5316	0.5167
$P=15$	Max	1.0031	1.0007	0.9778	0.9303
	Mean	0.9077	0.8911	0.8861	0.8182
	Min	0.7992	0.7719	0.7623	0.7052

从表 4 可以看出，随着样本属性个数的增加，所有聚类算法的 DB 性能均在下降，特别是当属性个数大于 15 时，*DB* 值上升明显，数据维度对 DB 性能的影响较大。对比发现，在相同实验条件下，狼群优化算法表现出了更优的 DB 性能。

3.2 不同聚类算法的运算时间

从 3.1 节可以得到，当训练样本的属性个数为 $P=10$ 时，对于本文所用 UCI 数据集，4 种聚类算法表现出了较好的聚类性能。本节对 4 种不同聚类算法在 $P=10$，$PBM>0.2$，$DB<0.5$ 的情况下的运算时间进行仿真，统计结果如表 5 所列。

表 5 不同聚类算法的运算时间

Table 5 Running time of different clustering algorithms

（单位：s）

样本容量	K-means	Mean-shift	DBSCAN	狼群聚类
100	5.322	8.213	9.321	7.239
200	8.121	12.315	13.665	12.413
500	12.387	39.416	41.451	37.5841
50000	32.726	127.213	177.214	129.167
100000	181.714	623.537	933.317	626.479

从表 5 可以看出，在聚类运算时间方面，K-means 算法的优势明显，DBSCAN 算法最耗时，狼群聚类和 Mean-shift 的运算时间居中。在实际应用过程中，可以根据实际需求动态调整每次训练的样本数量，以提高狼群聚类算法在具体应用场景的适用性。

结束语 本文采用狼群优化算法来实现云计算平台的数据聚类，将狼群算法运用于相似聚类中心点的优化与更新，进而提高中心点的计算效率。经过实验仿真证明，在云计算平台的数据聚类方面，本文算法与常用聚类算法相比，在 PBM 和 DB 上表现出了更优的性能。

参 考 文 献

[1] CAI Q Q, CUI H G, TANG H. Big data mining analysis method based on cloud computing[C]// Green Energy and Sustainable Development I: Proceedings of the International Conference on Green Energy and Sustainable Development (GESD 2017). 2017.

[2] ZHAO Y P, XU S C. Data Hierarchical Clustering Based on Cloud Computing and Nonnegative Matrix Decomposition [J]. Modern Electronic Technology, 2018, 41(5): 56-60. (in Chinese)
赵艳萍，徐胜超. 基于云计算与非负矩阵分解的数据分级聚类[J]. 现代电子技术，2018，41(5)：56-60.

[3] XIE F W. Optimal analysis of path mining for discontinuous data in cloud computing environment [J]. Computer Simulation, 2018, 35(6): 438-441. (in Chinese)
谢辅雯. 云计算环境下非连续数据路径挖掘优化分析[J]. 计算机仿真，2018，35(6)：438-441.

[4] KARIMI M B, ISAZADEH A, RAHMANI A M. QoS-aware service composition in cloud computing using data mining techniques and genetic algorithm [J]. Journal of Supercomputing, 2017, 73(4): 1-29.

[5] CAO S, AN J C. Otsu image segmentation method based on improved wolf swarm optimization algorithm [J]. Microelectronics and Computer, 2017, 34(10): 16-21. (in Chinese)
曹爽，安建成. 改进狼群优化算法的 Otsu 图像分割法[J]. 微电子学与计算机，2017，34(10)：16-21.

[6] GUPTA S, DEEP K. Random walk grey wolf optimizer for constrained engineering optimization problems [J]. Computational Intelligence, 2018, 34(4): 189-197.

[7] FARIS H, ALJARAH I, AL-BETAR M A, et al. Grey wolf optimizer: a review of recent variants and applications[J]. Neural Computing & Applications, 2018, 30(2): 413-435.

[8] WANG T, WANG Y, MENG L P. An improved wolf swarm search algorithm and its application in clustering [J]. Computer Applications and Software, 2016, 33(12): 257-263.

[9] WANG F Z. Automatic segmentation of FCM image based on wolf swarm optimization [J]. Control Engineering, 2018(9): 1727-1732.

[10] LIU X J, YUAN J B, FENG F P. K-means clustering algorithm for data distribution in cloud computing environment [J]. Minicomputer system, 2017, 38(4): 712-715.

[11] CHEN G L, YAN F L, PAN J H. Incremental Density Fast Clustering Algorithms for High Complexity Dynamic Data in Cloud Computing Environment [J]. Computer Science, 2018, 45(2): 287-290.

[12] FENG L W, CHANG D X, DENG Y, et al. The latest clustering performance evaluation index with the farthest score [J]. Journal of Intelligent Systems, 2017, 12(1): 67-74.

[13] KARUNAMBIGAI M G, AKRAM M, SIVASANKAR S, et al. Clustering Algorithm for Intuitionistic Fuzzy Graphs[J]. International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems, 2017, 25(3): 367-383.

[14] HAI M, ZHANG Y. Performance Comparison of Clustering Algorithms on Hai Mo, Zhang You. Spark Platform [J]. Computer Science, 2017, 44(s1): 414-418.

[15] GAO G S. Overview of unsupervised clustering methods for entity analysis [J]. Computer Engineering and Applications, 2018(7): 11-19.

[16] CHEN X Q, ZHOU L J, LIU Y Z. Overview of Clustering Algorithms [J]. Integrated Technology, 2017, 6(3): 41-49.