链滴

FuzzyC-Means 算法

作者：Lonery
原文链接：https://ld246.com/article/1615351839585
来源网站：链滴

FuzzyC-Means

<p style="text-indent:2em">
模糊c-均值聚类算法 fuzzy c-means algorithm(FCMA)或(FCM)。模糊c均值聚类算法，是当前模糊统里表现比较好的算法之一 其特征与k-means相似，也是基于距离来判断分类。模糊c均值需要用户供除数据之外至少一个参数，而这个参数与k-means中的k类似。
</p>

<p1 style="text-indent:2em">

模糊c-均值聚类算法意在求解一个最小化问题即：

</p1>

$$\min J=\sum_{i=1}^{c}\sum_{j=1}^{n}u_{ij}^{m}\left(x_j-c_i\right)^2$$

$$\text{s.t.}\quad \sum_{i=1}^{c}u_{ij}^{m}=1,\quad \forall j=1,\ldots\ldots,n$$

<p2 style="text-indent:2em">

其中，uij为样本xj属于第i类的隶属度，故u是一个c×n的矩阵，xj就是第j个样本数据，ci为第i个聚类心一共有c个，m则是一个大于1的加权常数一般取2，可由用户定义。 在这里每个样本对于不同类的属度之和被限制为1，但一般情况下模糊集合的隶属度通常加起来不为1。接下来我们需要把这个条件值问题转化为无条件的极值问题， 这里用到的方法就是拉格朗日乘子法：

</p2>

$$\min J'=\sum_{i=1}^{c}\sum_{j=1}^{n}u_{ij}^{m}\left(x_i-c_i\right)^2+\sum_{j=1}^{n}\lambda_i\left(\sum_{i=1}^{c}u_{ij}^{m}-1\right)$$

<p3 style="text-indent:2em">

然后需要对各个变量求导，以求得使得原式最小的变量值。对聚类的中心C求导：

</p3>

$$\frac{\partial J'}{\partial c_k}=\sum_{i=1}^{c}\sum_{j=1}^{n}\frac{\partial u_{ij}^{m}\left(x_j-c_i\right)^2}{\partial c_k}-\frac{\partial}{\partial c_k}\sum_{j=1}^{n}\lambda_j\left(\sum_{i=1}^{c}u_{ij}-1\right)$$

其中，

$$\frac{\partial}{\partial c_k}\sum_{j=1}^{n}\lambda_j\left(\sum_{i=1}^{c}u_{ij}-1\right)=0$$

原文链接：FuzzyC-Means 算法

故:

$$\frac{\partial J}{\partial c_k}=\sum_{i=1}^{c}\sum_{j=1}^{n}\frac{\partial u_{ij}^{m}\left(x_j-c_i\right)^2}{\partial c_k}$$

$$=\sum_{i=1}^{c}\sum_{j=1}^{n}u_{ij}^{m}\frac{\partial}{\partial c_k}\left(x_j-c_i\right)^2$$

$$=\sum_{j=1}^{n}u_{kj}^{m}\frac{\partial}{\partial c_k}\left(x_j-c_k\right)^2$$

$$=-2\sum_{j=1}^{n}u_{kj}^{m}\left(x_j-c_k\right)=0$$

然后,

$$\Rightarrow\sum_{j=1}^{n}u_{k}^{m}c_k\quad-\sum_{j=1}^{n}u_{k}^{m}x_j=0$$

最后,

$$\boxed{\Rightarrow c_k=\frac{\sum_{j=1}^{n}u_{kj}^{m}x_j}{\sum_{j=1}^{n}u_{kj}^{m}}}$$

<p4 style="text-indent:2em">

至此聚类中心C的迭代公式已确认，接下来就是模糊矩阵u的迭代推导。

</p4>

第一部分，先对条件式的前半部分对u进行求导。

$$\frac{\partial J'}{\partial u_{ij}}=\frac{\partial}{\partial u_{ij}}\sum_{j=1}^{n}\sum_{i=1}^{c}u_{ij}^{m}\boxed{\left(x_j-c_i\right)^2}$$

$$\text{令}\,d_{ij}=\boxed{\left(x_j-c_i\right)^2}$$

$$\frac{\partial J'}{\partial u_{ij}}=\frac{\partial}{\partial u_{ij}}\sum_{j=1}^{n}\sum_{i=1}^{c}u_{ij}^{m}d_{ij}=\sum_{j=1}^{n}\sum_{i=1}^{c}\frac{\partial}{\partial u_{ij}}u_{ij}^{m}d_{ij}=\sum_{j=1}^{n}\sum_{i=1}^{c}mu_{ij}^{m-1}d_{ij}$$

然后再求后半部分。

$$\frac{\partial J'}{\partial u_{ij}} = \frac{\partial}{\partial u_{ij}} \sum_{j=1}^{n} \lambda_j \left(\sum_{i=1}^{c} \boxed{u_{ij}} - 1 \right)$$

注意：是uij。

$$= \frac{\partial}{\partial u_{ij}} \sum_{j=1}^{n} \left(\sum_{i=1}^{c} \lambda_j u_{ij} - \lambda_j \right)$$

$$= \sum_{j=1}^{n} \sum_{i=1}^{c} \frac{\partial}{\partial u_{ij}} \lambda_j u_{ij}$$

$$= \sum_{j=1}^{n} \sum_{i=1}^{c} \lambda_j$$

$$= \sum_{j=1}^{n} c\lambda_j$$

将两部分叠加。

$$\frac{\partial J'}{\partial u_{ij}} = \sum_{j=1}^{n} \sum_{i=1}^{c} m u_{ij}^{m-1} d_{ij} + \sum_{j=1}^{n} c\lambda_j$$

$$= \sum_{j=1}^{n} \left(\sum_{i=1}^{c} m u_{ij}^{m-1} d_{ij} + c\lambda_j \right) = 0$$

$$\Rightarrow \left(\sum_{i=1}^{c} m u_{ij}^{m-1} d_{ij} + c\lambda_j \right) = 0$$

$$\Rightarrow \left(\sum_{i=1}^{c} m u_{ij}^{m-1} d_{ij} + c\lambda_j \right) = \left(\sum_{i=1}^{c} m u_{ij}^{m-1} d_{ij} + \sum_{i=1}^{c} \lambda_j \right) = \sum_{i=1}^{c} \left(m u_{ij}^{m-1} d_{ij} + \lambda_j \right) = 0$$

$$\because m u_{ij}^{m-1} d_{ij} + \lambda_j \geq 0$$

$$\therefore m u_{ij}^{m-1} d_{ij} + \lambda_j = 0, \forall i = 1 \ldots\ldots c$$

$$\Rightarrow u_{ij} = \left(\frac{-\lambda_j}{m d_{ij}} \right)^{\frac{1}{m-1}}$$

原文链接：FuzzyC-Means 算法

$$又 \because \sum_{k=1}^{c} u_{kj} = 1 \quad \forall j = 1,\ldots,n$$

$$\therefore 有\sum_{k=1}^{c}\left(\frac{-\lambda_j}{md_{kj}}\right)^{\frac{1}{m-1}} = \left(\frac{-\lambda_j}{m}\right)^{\frac{1}{m-1}}\sum_{k=1}^{C}\left(\frac{1}{d_{kj}}\right)^{\frac{1}{m-1}} = 1$$

$$\Rightarrow \left(\frac{-\lambda_j}{m}\right)^{\frac{1}{m-1}} = \frac{1}{\sum_{k=1}^{C}\left(\frac{1}{d_{kj}}\right)^{\frac{1}{m-1}}}$$

代回uij中

最后u的迭代公式：

$$\therefore u_{ij} = \frac{1}{\sum_{k=1}^{C}\left(\frac{1}{d_{kj}}\right)^{\frac{1}{m-1}}} \cdot \left(\frac{1}{d_{ij}}\right)^{\frac{1}{m-1}}$$

$$\boxed{\Rightarrow u_{ij} = \frac{1}{\sum_{k=1}^{C}\left(\frac{d_{ij}}{d_{kj}}\right)^{\frac{1}{m-1}}}}$$

(以上参考至：https://www.cnblogs.com/wxl845235800/p/11053261.html)

<p7 style="text-indent:2em">

接下来我们就可以根据两个迭代公式将算法轻松地编程。初始模糊矩阵取随机数，采用的分类数据为尾花数据集，m设置为2，截至条件为u的二范数变化小于1e-4.

</p7>

```
function [V,F]=FSC(X,C,m)
[n,D]=size(X);
if D>n
    X=X';
    [n,D]=size(X);
end
if nargin==2
    m=2;
```

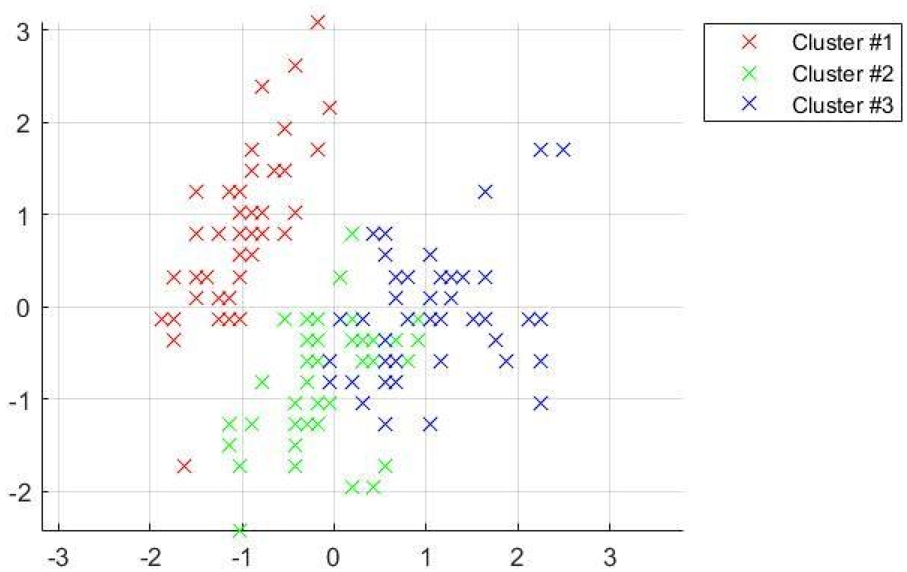
```
end
old=0;
A=rand(C,n);
A=oneness(A,n);
new=norm(A);
while abs(new-old)>1e-4
old=new;
V=updataV(A);
A=updataA(X,V);
new=norm(A);
[~,index]=max(A);
end
F=zeros(C,n);
for f=1:n
    F(index(f),f)=1;
end
if D>2
    X=zscore(X); %数据标准化
    [~,~,latent]=pca(X); %PCA降维
    [~,b]=sort(latent,'descend');
    X=X(:,b(1:2)); %取前两维绘图
    PlotClusterinResult(X,index);
end
if D==2
    PlotClusterinResult(X,index);
end
function A=oneness(A,n)
for i=1:n %模糊矩阵概率归一
    A(:,i)=A(:,i)/sum(A(:,i));
end
end

function V=updataV(A)
V=zeros(D,C);
for i=1:C
    V(:,i)=(A(i,:).^m*X)'/sum(A(i,:).^m);
end
end

function A=updataA(X,V)
A=zeros(C,n);
for i=1:C
    for j=1:n
        A(i,j)=1/sum((norm(X(j,:)'-V(:,i))./norm(X(j,:)'-V)).^(1/(m-1)));
    end
end
A=oneness(A,n);
end
end
function PlotClusterinResult(X, IDX)
k=max(IDX);
Colors=hsv(k);
Legends = {};
for i=0:k
```

```
    Xi=X(IDX==i,:);
    if i~=0
        Style = 'x';
        MarkerSize = 8;
        Color = Colors(i,:);
        Legends{end+1} = ['Cluster #' num2str(i)];
    else
        Style = 'o';
        MarkerSize = 6;
        Color = [0 0 0];
        if ~isempty(Xi)
            Legends{end+1} = 'Noise';
        end
    end
    if ~isempty(Xi)
        plot(Xi(:,1),Xi(:,2),Style,'MarkerSize',MarkerSize,'Color',Color);
    end
    hold on;
end
hold off;
axis equal;
grid on;
legend(Legends);
legend('Location', 'NorthEastOutside');
end
```

测试Matlab自带fisheriris数据集。

```
FSC(meas,3)
```

结果如下图：



原文链接：FuzzyC-Means 算法