

Multi-Scale Fusion Subspace Clustering Using Similarity Constraint

1. 主要贡献:

- ① 将 DAE 的不同层输出结果融为一体
- ② 提出相似及约束模型, 稳定训练过程并监督①的损失
- ③ 在 4 个新的数据集上验证有效性.

2. Subspace Clustering 基本思路.

① 求解 affinity matrix

{ factorization based method
model based method
self-expression method

② 在 affinity matrix 上应用 spectral clustering

3. SC-MSFSC 模块.

3.1 Feature Extraction Module.

Auto-Encoder network.

Loss function:

$$\frac{1}{2} \|X - \hat{X}\|_F^2$$

(1).

其中, $X, \hat{X} \in \mathbb{R}^{N \times H \times W \times C}$

3.2 Self-Expression Module

Loss function:

$$\|C\|_p + \frac{1}{2} \|Z - CZ\|_F^2 \quad \text{s.t. } \text{diag}(C) = 0 \quad (2)$$

其中, $Z = [z_1, z_2, \dots, z_N]$, $z_i \in \mathbb{R}^d$, $i=1, \dots, N$

$$C \in \mathbb{R}^{N \times N}$$

3.3 Multi-Scale Fusion Module

符号标记: $Z_l \in \mathbb{R}^{N \times D_l}$: l -th Encoder Output

\hat{X}_l : l -th Decoder Output

(1) 式更新为:

$$\frac{1}{2} \sum_l \|X_l - \hat{X}_l\|_F^2 \quad (3)$$

同样, (2) 更新为:

$$\sum_l \|C_l\|_p + \frac{1}{2} \sum_l \|Z_l - CZ_l\|_F^2 \quad \text{s.t. } \text{diag}(C_l) = 0 \quad (4)$$

为融合各 C_l 矩阵, 使用如下方法:

① 将所有 C_l 拼接为 C_s , $C_s \in \mathbb{R}^{N \times N \times L}$

② 使用 **卷积核 k** 整合 C_s 的各通道, 得到 $C_f \in \mathbb{R}^{N \times N}$
(使不同层的 C_l 具有不同的权重)

$$C_f = k \otimes C_s$$

因此, (4)式增加一项:

$$\|C_F\|_F + \frac{1}{2} \|Z - C_F Z\|_F^2 \quad \text{s.t. } \text{diag}(C_F) = 0 \quad (5)$$

其中, Z 即为 Z_L

同理, (3)式中最后一项 $\frac{1}{2} \|X_L - \hat{X}_L\|_F^2$ 修正为:

$$\frac{1}{2} \|X_L - \hat{X}_F\|_F^2 \quad (6)$$

其中, $\hat{X}_F = C_F Z_L$

3.4 Similarity Constraint Module

used to supervise the fused coefficient matrix

符号表式: $\alpha \in (0, 1)$: 阈值, 将 C 中每一列元素和小于 α 的置 0

C_{s-de} : 经过阈值法处理后的 C_s 矩阵

C_{de} : similarity matrix, C_{s-de} 的均值

$$\begin{cases} C_{s-de} \in \mathbb{R}^{M \times N \times L} \\ C_{de} \in \mathbb{R}^{N \times N} \end{cases}$$

Similarity constraint loss:

$$\|C_{de} - C_F\|_F^2 \quad (7)$$

目的使 C_F 保留一些在 denoising 中丢失的信息

3.5 训练设置

$$L_{pre} = \frac{1}{2} \|X - \hat{X}\|_F^2$$

$$L_{fine} = \hat{L}_0 + \lambda_1 \hat{L}_1 + \lambda_2 \hat{L}_2 + \lambda_3 L_3 + \lambda_4 L_4$$