Hierarchical Heterogeneous Analysis of High-Dimensional Data

Yan Ren

2021103739

2022 年 11 月 2 日

目录

1 Model

- $X = (X_1, ..., X_n)^T \in \mathbb{R}^{n\times p}$, type I features (main G) for all observations, where $X_i \in \mathbb{R}^p$;
- $Z = (Z_1, ..., Z_n)^T \in \mathbb{R}^{n\times q}$, type II features (main E) for all observations, where $Z_i \in \mathbb{R}^q$;
- $y = (y_1, ..., y_n)^T \in \mathbb{R}^n$, the response vector.

1.1 Basic Model

简化后的异质性模型为：

$$y_i = X_i^T \beta_i' + Z_i^T \alpha_i' + \epsilon_i \tag{1.1}$$

其中 $\epsilon_i \sim (iid)N(0, \sigma^2)$

假设所有观测大组可根据基因（X）进行划分，大组中根据环境/图片特征变量（Z）进行进一步的划分。假设有 K 个亚组，则相同亚组中的样有相同参数。

1.2 Object Function

基于高斯混合分布，重参数化后最小化目标函数为（负对数似然+惩罚）：

$$\begin{aligned}
\mathcal{L}(\beta,\alpha,\pi|X,y) &= -\frac{1}{n}\sum_{i=1}^{n}\log p_Y(y) + \text{pen}(\beta',\alpha') \\
&= -\frac{1}{n}\sum_{i=1}^{n}\log\big(\sum_{k=1}^{K}\pi_k f_k(y_i|X_i,Z_i)\big) + \text{pen}(\beta',\alpha') \\
&= -\frac{1}{n}\sum_{i=1}^{n}\log\left\{\sum_{k=1}^{K}\pi_k\frac{1}{\sqrt{2\pi}\sigma_k}\exp\left(-\frac{1}{2\sigma_k^2}\left(y_i - X_i^T\beta_k' - Z_i^T\alpha_k'\right)^2\right)\right\} + \text{pen}(\beta',\alpha') \\
&= -\frac{1}{n}\sum_{i=1}^{n}\log\left\{\sum_{k=1}^{K}\pi_k\frac{\rho_k}{\sqrt{2\pi}}\exp\left(-\frac{1}{2}\left(y_i\rho_k - X_i^T\beta_k - Z_i^T\alpha_k\right)^2\right)\right\} + \text{pen}(\beta,\alpha)
\end{aligned} \tag{1.2}$$

其中重参数化为 $\beta_k = \beta_k'/\sigma_k, \rho_k = 1/\sigma_k$，惩罚函数为

$$
\begin{aligned}
\operatorname{pen}(\beta, \alpha) = & \sum_{k=1}^{K} \sum_{j=1}^{p} \operatorname{pen}\left(\left|\beta_{kj}\right|, \lambda_1\right) + \sum_{k=1}^{K} \sum_{l=1}^{q} \operatorname{pen}\left(\left|\alpha_{kl}\right|, \lambda_1\right) \\
& + \sum_{k<k'} \operatorname{pen}\left(\sqrt{\left\|\beta_k - \beta_{k'}\right\|_2^2 + \left\|\alpha_k - \alpha_{k'}\right\|_2^2}, \lambda_2\right) \\
& + \sum_{k<k'} \operatorname{pen}\left(\left\|\beta_k - \beta_{k'}\right\|_2, \lambda_3\right)
\end{aligned} \tag{1.3}
$$

第一项惩罚保证基因相关系数的稀疏性；第二，第三项惩罚共同作用以致模型呈现层级结构（Hierarchy）.

1.3 Penalty Function

MCP 惩罚函数

$$
pen_{(\lambda,a)}(\beta) = \begin{cases} \lambda|\beta| - \dfrac{\beta^2}{2a}, & |\beta| \leq a\lambda \\ \dfrac{a}{2}\lambda^2, & |\beta| > a\lambda \end{cases} \tag{1.4}
$$

MCP 惩罚函数对参数求偏导得

$$
pen'_{(\lambda,a)}(\beta) = \begin{cases} \operatorname{sgn}(\beta)\left(\lambda - \frac{|\beta|}{a}\right) & |\beta| \leq a\lambda \\ 0 & \text{otherwise} \end{cases} \tag{1.5}
$$

超参数 a 常取为大于 1 的数。

2 Computation

2.1 Transformation

采用 EM 算法。E 步更新 π，M 步更新参数 β, α。引入变量 C_i 用于指示每个样本属于哪个类别，有后验概率如 2.1 所示

$$
q_{C_i}(c_i) = \frac{\pi_{c_i} f_{c_i}(y_i|X_i, Z_i)}{\sum_{k=1}^{K} \pi_k f_k(y_i|X_i, Z_i)} \tag{2.1}
$$

目标函数 1.2 中对数似然部分可以放缩如下

$$\begin{aligned}\log p_Y(y) &= log\sum_{k=1}^{K} p_{Y,C}(y,c) \\ &= logE_{q_C}\left(\frac{p_{Y,C}(y,c)}{q_C(c)}\right) \\ &\geq E_{q_C}\left(\frac{p_{Y,C}(y,c)}{q_C(c)}\right) \\ &= E_{q_C}\left(logp_{Y,C}(y,c)\right) - E_{q_C}\left(logq_C(c)\right)\end{aligned} \tag{2.2}$$

综上，目标函数可以放缩（根据 Gaussian mixture models and the EM algorithm）为

$$\begin{aligned}\mathcal{L}(\beta,\alpha,\pi|X,y) &= -\sum_{i=1}^{n}\log p_Y(y) + \mathrm{pen}(\beta,\alpha) \\ &\leq -E_{q_C}\left(logp_{Y,C}(y,c)\right) + E_{q_C}\left(logq_C(c)\right) + \mathrm{pen}(\beta,\alpha)\end{aligned} \tag{2.3}$$

在 M 步更新参数 β,α 式 2.3 中 $E_{q_C}(logq_C(c))$ 与参数无关。而 $E_{q_C}\left(logp_{Y,C}(y,c)\right)$ 可以进一步变形

$$\begin{aligned}E_{q_C}logp_{Y,C}(y,c) &= E_{q_C}log\left(p_C(c)p_{Y|C}(y)\right) \\ &= E_{q_C}log\left(\prod_{i=1}^{n} p_{C_i}(c_i)p_{Y_i|C_i}(y_i)\right) \\ &= E_{q_C}log\left(\prod_{i=1}^{n}\prod_{k=1}^{K}\pi_{c_i}f_{Y_i|C_i}(y_i)\right)^{I(C_i=k)} \\ &= E_{q_C}\sum_{i=1}^{n}\sum_{k=1}^{K} I(C_i=k)\left(log\pi_{c_i} + logf_{Y_i|C_i}(y_i)\right) \\ &= \sum_{i=1}^{n}\sum_{k=1}^{K} E_{q_C}I(C_i=k)\left(log\pi_{c_i} + logf_{Y_i|C_i}(y_i)\right) \\ &= \sum_{i=1}^{n}\sum_{k=1}^{K} q_{C_i}(k)\left(log\pi_{c_i} + logf_{Y_i|C_i}(y_i)\right) \\ &= \sum_{i=1}^{n}\sum_{k=1}^{K} q_{C_i}(k)\left(log\pi_{c_i} + log\rho_k - \frac{1}{2}log2\pi - \frac{1}{2}(y_i\rho_k - X_i^T\beta_k - Z_i^T\alpha_k)^2\right) \\ &= \sum_{i=1}^{n}\sum_{k=1}^{K} q_{C_i}(k)\left(C - \frac{1}{2}(y_i\rho_k - X_i^T\beta_k - Z_i^T\alpha_k)^2\right)\end{aligned} \tag{2.4}$$

其中 C 为与更新参数无关的常数项。此时最小化目标为原目标函数的上界，表达式为

$$
\begin{aligned}
\mathcal{L}(\beta,\alpha|\pi,X,y) =& -\frac{1}{n}\sum_{i=1}^{n}\log p_Y(y)+\operatorname{pen}(\beta,\alpha)\\
\le& -\frac{1}{n}E_{q_C}\left(logp_{Y,C}(y,c)\right)+\operatorname{pen}(\beta,\alpha)\\
=& -\frac{1}{n}\sum_{i=1}^{n}\sum_{k=1}^{K}q_{C_i}(k)\left(C-\frac{1}{2}(y_i\rho_k-X_i^T\beta_k-Z_i^T\alpha_k)^2\right)\\
&+\sum_{k=1}^{K}\operatorname{pen}\left(\|\beta_k\|_2,\lambda_1\right)+\sum_{k=1}^{K}\operatorname{pen}\left(\|\alpha_k\|_2,\lambda_1\right)\\
&+\sum_{k<k'}\operatorname{pen}\left(\sqrt{\|\beta_k-\beta_{k'}\|_2^2+\|\alpha_k-\alpha_{k'}\|_2^2},\lambda_2\right)\\
&+\sum_{k<k'}\operatorname{pen}\left(\left\|\beta_k-\beta_{k'}\right\|_2,\lambda_3\right)
\end{aligned}
\tag{2.5}
$$

2.2 MCP

β, z 均为 p 维向量，现假设目标函数有

$$
\mathcal{L}(x;\lambda)=\frac{\rho}{2}\|x-z\|_2^2+\operatorname{pen}(\|x\|_2;\lambda)
\tag{2.6}
$$

则 x 对应解为

$$
x=\begin{cases}\frac{S\left(z,\frac{\lambda}{\rho}\right)}{1-\frac{1}{a\rho}} & \text{if } |z|\le a\lambda\\ z & \text{if } |z|> a\lambda\end{cases}
\tag{2.7}
$$

其中 $S(.,.)$ 为 soft thresholding operator，定义如下

$$
S(x,t)=\operatorname{sign}(x)(|x|-t)_+=x\left(1-\frac{t}{|x|}\right)_+
\tag{2.8}
$$

故 x 解也可写为

$$
x=\begin{cases}\frac{z\left(1-\frac{\lambda}{|z|\rho}\right)_+}{1-\frac{1}{a\rho}} & \text{if } |z|\le a\lambda\\ z & \text{if } |z|> a\lambda\end{cases}
\tag{2.9}
$$

3 Model Preparation

3.1 Version 1

为了尝试推导以及检验代码正确性，先尝试构建较为简单的模型进行实验。考虑以下目标函数

$$\mathcal{L}^{(1)}(\beta,\alpha)=-\frac{1}{n}\sum_{i=1}^{n}\sum_{k=1}^{K}q_{C_i}(k)\left(C-\frac{1}{2}(y_i\rho_k-X_i^T\beta_k-Z_i^T\alpha_k)^2\right)+\sum_{k=1}^{K}\sum_{j=1}^{p}\text{pen}\left(|\beta_{kj}|,\lambda_1\right) \tag{3.1}$$

对于类别 k 目标函数为

$$\mathcal{L}_k^{(1)}(\beta_k,\alpha_k)=-\frac{1}{n}\sum_{i=1}^{n}q_{C_i}(k)\left(C-\frac{1}{2}(y_i\rho_k-X_i^T\beta_k-Z_i^T\alpha_k)^2\right)+\sum_{j=1}^{p}\text{pen}\left(|\beta_{kj}|,\lambda_1\right) \tag{3.2}$$

对应增广拉格朗日函数为

$$\begin{aligned}\mathcal{L}_{k,\rho}^{(1)}(\beta_k,\alpha_k,\tau)=&-\frac{1}{n}\sum_{i=1}^{n}q_{C_i}(k)\left(C-\frac{1}{2}(y_i\rho_k-X_i^T\beta_k-Z_i^T\alpha_k)^2\right)\\&+\sum_{j=1}^{p}\text{pen}\left(|\theta_{kj}|,\lambda_1\right)+\tau^T(\beta_k-\theta_k)+\frac{\rho}{2}\|\beta_k-\theta_k\|_2^2\end{aligned} \tag{3.3}$$

τ 为对应的拉格朗日乘子（向量），$\rho>0$ 为拉格朗日参数。给定 $(\beta_k^{(t)},\alpha_k^{(t)},\tau^{(t)})$ 则第 $t+1$ 步更新写为

$$\beta_k^{(t+1)}\in \underset{\beta\in R^p}{arg\,min}\ \mathcal{L}_{k,\rho}^{(1)}\left(\beta,\theta^{(t)},\tau^{(t)}\right) \tag{3.4}$$

$$\theta_k^{(t+1)}\in \underset{\theta\in R^p}{arg\,min}\ \mathcal{L}_{k,\rho}^{(1)}\left(\beta^{(t+1)},\theta,\tau^{(t)}\right) \tag{3.5}$$

$$\tau^{(t+1)}=\tau^{(t)}+\rho\left(\beta^{(t+1)}-\theta^{(t+1)}\right) \tag{3.6}$$

Update β_k 式 3.3 中与 β_k 相关的项为

$$\begin{aligned}\mathcal{L}_{k,\rho}^{(1)}(\beta_k)=&-\frac{1}{n}\sum_{i=1}^{n}q_{C_i}(k)\left(C-\frac{1}{2}(y_i\rho_k-X_i^T\beta_k-Z_i^T\alpha_k)^2\right)\\&+\tau^T(\beta_k-\theta_k)+\frac{\rho}{2}\|\beta_k-\theta_k\|_2^2\end{aligned} \tag{3.7}$$

对 β_k 求偏导，得

$$\beta_k = \left(n^{-1}X^TW_kX + \rho I_p\right)^{-1}\left(n^{-1}X^TW_k'y^{\beta_k} + \rho\theta_k - \tau\right) \tag{3.8}$$

其中 $y^{(\beta_k)} = y\rho_k - Z\alpha_k$，$W_k = \rho_k diag(q_{C_1}(k), ..., q_{C_n}(k))$，$I_p$ 为 p 维单位矩阵。

Update θ_k 式 3.3 中与 θ_k 相关的项为

$$\begin{aligned}\mathcal{L}^{(1)}_{k,\rho}(\beta_k) &= \sum_{j=1}^{p}\text{pen}\left(\theta_{kj}, \lambda_1\right) + \tau^T(\beta_k - \theta_k) + \frac{\rho}{2}\|\beta_k - \theta_k\|_2^2 \\ &= \frac{\rho}{2}\|\theta_k - z_k\|_2^2 + \sum_{j=1}^{p}\text{pen}\left(\theta_{kj}, \lambda_1\right)\end{aligned} \tag{3.9}$$

其中 $z_k = \beta_k + \frac{\tau}{\rho}$.

故 $\theta_k = (\theta_{k1}, ..., \theta_{kp})$ 的解为

$$\theta_{kj} = \begin{cases} \frac{S\left(z_{kj}, \frac{\lambda}{\rho}\right)}{1 - \frac{1}{a\rho}} & \text{if } |z_{kj}| \le a\lambda \\ z_{kj} & \text{if } |z_{kj}| > a\lambda \end{cases} \tag{3.10}$$

$j = 1, ..., p$，若给定每个样本类别，等价于再各个类别内部使用 ADMM 方法进行迭代求解系数；若未知样本类别，使用 EM 算法进行估计，算法流程

当前实验大体结果如下：

3.2 Version 2

依旧对目标函数

$$\mathcal{L}^{(1)}(\beta, \alpha) = -\frac{1}{n}\sum_{i=1}^{n}\sum_{k=1}^{K} q_{C_i}(k)\left(C - \frac{1}{2}(y_i\rho_k - X_i^T\beta_k - Z_i^T\alpha_k)^2\right) + \sum_{k=1}^{K}\sum_{j=1}^{p}\text{pen}\left(|\beta_{kj}|, \lambda_1\right) \tag{3.11}$$

不同于上一节，现在直接对于 β_{kj} 求偏导进行求解。记 $u_{kj} = (\rho_k y - X_{(-j)}\beta_{k(-j)})^T W_k \tilde{X}_j$，$l_{kj} = \tilde{X}_j^T W_k \tilde{X}_j$ 则

$$\frac{\partial \mathcal{L}}{\partial \beta_{kj}} = \begin{cases} 2l_{kj}\beta_{kj} - 2u_{kj}, & |\beta_{kj}| > a\lambda \\ 2l_{kj}\beta_{kj} - 2u_{kj} + sgn(\beta_{kj})(\lambda - \frac{|\beta_{kj}|}{a}), & |\beta_{kj}| \le a\lambda \end{cases} \tag{3.12}$$

其中 $\tilde{X}_j = (x_{1j}, ..., x_{nj})^T \in \mathbb{R}^{n\times 1}$，$\beta_{k(-j)}$ 表示 β_k 中除去第 j 维其他参数，$X_{(-j)} \in \mathbb{R}^{n\times(p-1))}$ 表示去除第 j 维的数据。$\frac{\partial \mathcal{L}}{\partial \beta_{kj}} = 0$ 时，解得

Algorithm 1:

Input : 数据 y, X，超参数 $n, p, K, a, \lambda, \rho$

Output: $\beta_k, k = 1, ..., K$ 的估计值

初始化 $q_{C_i}(k) = \frac{1}{K} + \mathrm{N}(0, 0.5^2), k = 1, ..., K, i = 1, ..., n$ 标准化;

repeat

  for $k\ in\ 1 : K$ **do**

    $W_k = diag(q_{C_1}(k), ..., q_{C_n}(k))$;

    $W'_k = diag(q_{C_1}(k), ..., q_{C_n}(k)) \cdot \rho_k$;

    更新 $\beta_k^{(iter)} = \left(n^{-1} X^T W_k X + \rho I_p\right)^{-1} \left(n^{-1} X^T W'_k y^{\beta_k} + \rho\theta_k - \tau\right)$;

    更新 $\theta_k^{(iter)} = \begin{cases} \frac{S\left(z_k, \frac{\lambda}{\rho}\right)}{1 - \frac{1}{a\rho}} & \text{if } \|z_k\|_2 \leq a\lambda \\ z & \text{if } \|z_k\|_2 > a\lambda \end{cases}$;

    更新 $\tau^{(iter)} = \tau^{(iter-1)} + \rho(\beta_k^{(iter)} - \theta_k^{(iter)})$.

    更新 $\rho_k^{(iter)} = \frac{-B - \sqrt{B^2 - 4AC}}{2A}$.

      where $A = -y^T W_k y; B = (X\beta_k)^T W_k y; C = \sum_{i=1}^n q_{C_i}(k)$.

  end for

  计算基于 $\beta_k^{(iter)}$ 得到的后验概率 q_C

    $q_{C_i}^{(iter)}(k) = \frac{\pi_k^{(iter)} f_k(y_i; \beta_k^{(iter)} / \rho_k^{(iter)})}{\sum_{k'=1}^K \pi_{k'}^{(iter)} f_{k'}(y_i; \beta_{k'}^{(iter)} / \rho_k^{(iter)})}, k = 1, ..., K$

  更新 $\pi_k^{(iter)} = \frac{1}{n} \sum_{i=1}^n q_{C_i}^{(iter)}(k)$

until 收敛（$\frac{\|\beta^{(iter)} - \beta^{(iter-1)}\|}{\|\beta^{(iter)}\|}$小于临界值）;

p	非零个数	epsilon	flexmix	自己迭代
8	3	0	收敛到真值	收敛到真值
8	3	$N(0, 0.5^2)$	收敛到接近真值	收敛到接近真值
40	3	0	全0估计	距离真值有差异，详情见表
40	3	$N(0, 0.5^2)$	全0估计	收敛到接近真值

$$
\beta_{kj}^{(iter)} = \begin{cases} \frac{u_{kj}}{l_{kj}} & \frac{u_{kj}}{l_{kj}} = 0,\ or\ |\frac{u_{kj}}{l_{kj}}| > a\lambda \\ \left(\frac{2u_{kj}-\lambda}{2l_{kj}-\frac{1}{a}}\right)_+, & 0 < \frac{u_{kj}}{L_{kj}} \leqslant a\lambda \\ \left(\frac{2u_{kj}+\lambda}{2l_{kj}-\frac{1}{a}}\right)_-, & -a\lambda \leqslant \frac{u_{kj}}{l_{kj}} < 0 \end{cases} \tag{3.13}
$$

算法流程如 2

Algorithm 2:

Input ：数据 y, X，超参数 $n, p, K, a, \lambda, \rho$

Output: $\beta_k, k = 1, ..., K$ 的估计值

初始化 $q_{C_i}(k) = \frac{1}{K} + \mathrm{N}(0, 0.5^2), k = 1, ..., K, i = 1, ..., n$ 标准化;

repeat

  for $k\ in\ 1 : K$ **do**

    $W_k = diag(q_{C_1}(k), ..., q_{C_n}(k))$;

    for $j\ in\ 1 : p$ **do**

      $u_{kj} = (\rho_k y - X_{(-j)}\beta_{k(-j)})^T W_k \tilde{X}_j$

      $l_{kj} = \tilde{X_j}^T W_k \tilde{X}_j$

      更新 $\beta_{kj}^{(iter)} = \begin{cases} \frac{u_{kj}}{l_{kj}} & \frac{u_{kj}}{l_{kj}} = 0,\ or\ |\frac{u_{kj}}{l_{kj}}| > a\lambda \\ \left(\frac{2u_{kj}-\lambda}{2l_{kj}-\frac{1}{a}}\right)_+, & 0 < \frac{u_{kj}}{L_{kj}} \leqslant a\lambda \\ \left(\frac{2u_{kj}+\lambda}{2l_{kj}-\frac{1}{a}}\right)_-, & -a\lambda \leqslant \frac{u_{kj}}{l_{kj}} < 0 \end{cases}$

    end for

    更新 $\rho_k^{(iter)} = \frac{-B-\sqrt{B^2-4AC}}{2A}$.

      where $A = -y^T W_k y; B = (X\beta_k)^T W_k y; C = \sum_{i=1}^n q_{C_i}(k)$.

  end for

  计算基于 $\beta_k^{(iter)}$ 得到的后验概率 q_C

    $q_{C_i}^{(iter)}(k) = \frac{\pi_k^{(iter)} f_k(y_i; \beta_k^{(iter)}/\rho_k^{(iter)})}{\sum_{k'=1}^K \pi_{k'}^{(iter)} f_{k'}(y_i; \beta_{k'}^{(iter)}/\rho_k^{(iter)})}, k = 1, ..., K$

  更新 $\pi_k^{(iter)} = \frac{1}{n}\sum_{i=1}^n q_{C_i}^{(iter)}(k)$

until 收敛（$\frac{\|\beta^{(iter)}-\beta^{(iter-1)}\|}{\|\beta^{(iter)}\|}$小于临界值）;

References

[1] Yueyong Shi, Yuling Jiao, Yongxiu Cao, and Yan Yan Liu. An alternating direction method of multipliers for mcp-penalized regression with high-dimensional data. *Acta Mathematica Sinica, English Series*, 34:1892–1906, 2018.