CS229 课程笔记

伍炜臻
华南师范大学 广州
510631 867036276@qq.com

摘要

这是我在学习斯坦福大学机器学习 CS229 课程过程中所记录的笔记. 除了参考 CS229 提供的讲义, 还参考了 PRML[1], ESL[2], 机器学习 [3] , 统计学习方法 [4] 等资料. 目前这篇笔记尚未完成.

A note on CS229 course

Weizhen Wu
SCNU Guangzhou
510631 867036276@qq.com

目录

1 Linear regression

在线性模型当中, 一个输入 $\boldsymbol{x} = (x_1, \ldots, x_N)^T \in \mathbb{R}^N$ 的预测值由一个线性函数给出

$$\begin{aligned} h_{\boldsymbol{\theta}}(\boldsymbol{x}) &= \theta_0 + \theta_1 x_1 + \ldots \theta_N x_N \\ &= \sum_{n=0}^{N} \theta_n x_i \\ &= \boldsymbol{\theta}^T \boldsymbol{x} \end{aligned} \tag{1}$$

其中固定设置 $x_0 = 1$.

1.1 Least squares

对于 M 个样例的训练数据 $D = \{(\boldsymbol{x_1}, y_1), \ldots, (\boldsymbol{x_M}, y_M)\}$, 平方和误差函数 E 为

$$E(\boldsymbol{\theta}; D) = \frac{1}{2} \sum_{m=1}^{M} (h_\theta(\boldsymbol{x_m}) - y_m)^2. \tag{2}$$

线性回归任务就是要最小化误差函数

$$\hat{\boldsymbol{\theta}} = \arg \min_{\boldsymbol{\theta} \in \mathbb{R}^{N+1}} E(\boldsymbol{\theta}; D). \tag{3}$$

这被成为线性回归的最小二乘. 为了求解 $\boldsymbol{\theta}$, 我们需要求损失函数 E 关于参数 $\boldsymbol{\theta}$ 每一个分量 $\theta^{(n)}$ 的偏导数

$$\begin{aligned} \frac{\partial E}{\partial \theta^{(n)}} &= \frac{\partial}{\partial \theta^{(n)}} \frac{1}{2} \sum_{m=1}^{M} (h_\theta(\boldsymbol{x_n}) - y_n)^2 \\ &= \frac{1}{2} \sum_{m=1}^{M} \frac{\partial}{\partial \theta^{(n)}} (\boldsymbol{\theta}^T \boldsymbol{x_m} - y_m)^2 \\ &= \frac{1}{2} \sum_{m=1}^{M} \frac{\partial (\boldsymbol{\theta}^T \boldsymbol{x_m} - y_m)^2}{\partial (\boldsymbol{\theta}^T \boldsymbol{x_m} - y_m)} \frac{\partial (\boldsymbol{\theta}^T \boldsymbol{x_m} - y_m)}{\partial \theta^{(n)}} \\ &= \frac{1}{2} \sum_{m=1}^{M} 2(\boldsymbol{\theta}^T \boldsymbol{x_m} - y_m) \frac{\partial \boldsymbol{\theta}^T \boldsymbol{x_m}}{\partial \theta^{(n)}} \\ &= \sum_{m=1}^{M} (\boldsymbol{\theta}^T \boldsymbol{x_m} - y_m) x_m^{(n)}. \end{aligned} \tag{4}$$

则损失函数 E 关于 $\boldsymbol{\theta}$ 的梯度为

$$\nabla_{\boldsymbol{\theta}} E = \sum_{m=1}^{M} (\boldsymbol{\theta}^T \boldsymbol{x_m} - y_m) \boldsymbol{x_m} \tag{5}$$

其中 $(\nabla_{\boldsymbol{\theta}} E)^{(n)} = \partial E / \partial \theta^{(n)}$. 又引入记号

$$\boldsymbol{X} = \begin{pmatrix} \boldsymbol{x_1}^T \\ \vdots \\ \boldsymbol{x_M}^T \end{pmatrix}, \quad \boldsymbol{y} = \begin{pmatrix} y_1 \\ \vdots \\ y_M \end{pmatrix} \tag{6}$$

则损失函数的梯度可以重新写为

$$\nabla_{\boldsymbol{\theta}} E = \boldsymbol{X}^T(\boldsymbol{X}\boldsymbol{\theta} - \boldsymbol{y}). \tag{7}$$

这样, 最小化损失函数 E 的任务可以通过梯度方法来迭代求解

$$\boldsymbol{\theta} := \boldsymbol{\theta} - \alpha \nabla_{\boldsymbol{\theta}} E \tag{8}$$

也可以直接令 $\nabla_{\boldsymbol{\theta}} E = \boldsymbol{0}$ 来解析地求出 $\boldsymbol{\theta}$

$$\begin{aligned} \boldsymbol{X}^T\boldsymbol{X}\boldsymbol{\theta} &= \boldsymbol{X}^T\boldsymbol{y} \\ \boldsymbol{\theta} &= (\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{X}^T\boldsymbol{y} \end{aligned} \tag{9}$$

当 $\boldsymbol{X}^T\boldsymbol{X}$ 不可逆时, 可以用广义逆来代替.

1.2 Probabilistic model

假设响应变量 y 和输入变量 $\boldsymbol{x}$ 是的线性相关的

$$y = \boldsymbol{\theta}^T\boldsymbol{x} + \varepsilon \tag{10}$$

其中 $\varepsilon \sim \mathcal{N}(0, \sigma^2)$ 为独立同分布的噪声项. 有

$$\begin{aligned} p(y|\boldsymbol{x};\boldsymbol{\theta}) &= \mathcal{N}(y|\boldsymbol{\theta}^T\boldsymbol{x}, \sigma^2) \\ &= \frac{1}{\sqrt{2\pi}\sigma} \exp(-\frac{(y - \boldsymbol{\theta}^T\boldsymbol{x})^2}{2\sigma^2}) \end{aligned} \tag{11}$$

对于观测数据 $D = \{(\boldsymbol{x_1}, y_1), \ldots, (\boldsymbol{x_M}, y_M)\}$, 似然函数为

$$\begin{aligned} \mathcal{L}(\boldsymbol{\theta}|D) &= \prod_{m=1}^{M} p(y_m|\boldsymbol{x_M};\boldsymbol{\theta}) \\ &= \prod_{m=1}^{M} \frac{1}{\sqrt{2\pi}\sigma} \exp(-\frac{(y_m - \boldsymbol{\theta}^T\boldsymbol{x_m})^2}{2\sigma^2}) \end{aligned} \tag{12}$$

为了对参数 $\boldsymbol{\theta}$ 进行最大似然估计

$$\hat{\boldsymbol{\theta}} = \arg\max_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\theta}|D) \tag{13}$$

我们可以最小化负对数似然函数

$$-\log \mathcal{L}(\boldsymbol{\theta}|D) = \log \frac{1}{\sqrt{2\pi}\sigma} \sum_{m=1}^{M} \frac{(y_m - \boldsymbol{\theta}^T\boldsymbol{x_m})^2}{2\sigma^2} \tag{14}$$

优化问题 (13) 等价于

$$\hat{\boldsymbol{\theta}} = \arg\min_{\boldsymbol{\theta}} \sum_{m=1}^{M} (y_m - \boldsymbol{\theta}^T\boldsymbol{x_m})^2 \tag{15}$$

注意到, 这等价于上面的最小二乘的优化问题 (3).

1.3 Locally weighted linear regression

局部加权线性回归引入了距离加权的特性. 对于某个待预测的点 $\boldsymbol{x}$, 较近的样本点 $\boldsymbol{x_m}$ 将具有更大的距离加权. 预测函数定义为

$$h_{\boldsymbol{\theta}}(\boldsymbol{x}) = \boldsymbol{\theta}_{\boldsymbol{x}}^T \boldsymbol{x} \tag{16}$$

在这里, $\boldsymbol{\theta_x}$ 是 $\boldsymbol{x}$ 的函数. 局部加权的误差函数为

$$E(\boldsymbol{\theta}; \boldsymbol{x}, D) = \sum_{m=1}^{M} k_\sigma(\boldsymbol{x}, \boldsymbol{x_m})(\boldsymbol{\theta}^T \boldsymbol{x_m} - y_m)^2 \tag{17}$$

其中 $k_\sigma(\boldsymbol{x}, \boldsymbol{x_m})$ 为局部加权值, 定义为

$$k_\sigma(\boldsymbol{x_1}, \boldsymbol{x_2}) = \exp(-\frac{(\boldsymbol{x_1} - \boldsymbol{x_2})^2}{2\sigma^2}) \tag{18}$$

超参数 σ 控制了权值对距离 $||\boldsymbol{x_1} - \boldsymbol{x_2}||$ 的敏感程度. 局部加权的参数 $\boldsymbol{\theta_x}$ 为

$$\boldsymbol{\theta_x} = \arg\min_{\boldsymbol{\theta}} E(\boldsymbol{\theta}; \boldsymbol{x}, D) \tag{19}$$

为了求解 $\boldsymbol{\theta_x}$, 我们对 E 求关于 $\boldsymbol{\theta}$ 各个分量的偏导数

$$\frac{\partial E}{\partial \theta^{(n)}} = \sum_{m=1}^{M} k(\boldsymbol{x}, \boldsymbol{x_m})(\boldsymbol{\theta}^T \boldsymbol{x_m} - y_m) x_m^{(n)} \tag{20}$$

则 E 关于 $\boldsymbol{\theta}$ 的梯度为

$$\nabla_{\boldsymbol{\theta}} E = \boldsymbol{X}^T \boldsymbol{R}(\boldsymbol{X}\boldsymbol{\theta} - \boldsymbol{y}) \tag{21}$$

其中 $\boldsymbol{R}$ 为对角矩阵, 定义为

$$\boldsymbol{R} = diag(k_\sigma(\boldsymbol{x}, \boldsymbol{x_1}), \ldots, k_\sigma(\boldsymbol{x}, \boldsymbol{x_M})). \tag{22}$$

令 $\nabla_{\boldsymbol{\theta}} E = 0$ 得

$$\boldsymbol{\theta_x} = (\boldsymbol{X}^T \boldsymbol{R} \boldsymbol{X})^{-1} \boldsymbol{X}^T \boldsymbol{R} \boldsymbol{y} \tag{23}$$

需要注意的是, 对于不同的输入值 $\boldsymbol{x}$, 为了预测 y, 都需要计算相应 $\boldsymbol{\theta_x}$. 实际上, 这是并不是一个线性模型.

2 Logistic regression

二分类问题中, 假定一个给定的点 $\boldsymbol{x}$ 的类别为 $y = 1$ 和 $y = 0$ 的后验概率分别为

$$\begin{aligned} p(y = 1|\boldsymbol{x}; \boldsymbol{\theta}) &= h_{\boldsymbol{\theta}}(\boldsymbol{x}) \\ &= g(\boldsymbol{\theta}^T \boldsymbol{x}) \\ &= \frac{1}{1 + \exp(-\boldsymbol{\theta}^T \boldsymbol{x})} \end{aligned} \tag{24}$$

$$\begin{aligned} p(y=0|\boldsymbol{x};\boldsymbol{\theta}) &= 1-p(y=1|\boldsymbol{x};\boldsymbol{\theta}) \\ &= \frac{\exp(-\boldsymbol{\theta}^T\boldsymbol{x})}{1+\exp(-\boldsymbol{\theta}^T\boldsymbol{x})}, \end{aligned} \tag{25}$$

其中

$$g(z) = \frac{1}{1+\exp(-z)}, \tag{26}$$

它被称为 Logistic 函数或 Sigmoid 函数, 其导数为

$$\frac{dg(z)}{dz} = g(z)(1-g(z)). \tag{27}$$

式 24,25可以合并写成

$$p(y|\boldsymbol{x};\boldsymbol{\theta}) = h_{\boldsymbol{\theta}}(\boldsymbol{x})^y(1-h_{\boldsymbol{\theta}}(\boldsymbol{x}))^{1-y}, \quad y\in\{0,1\} \tag{28}$$

对于一个训练数据集 $D=\{(\boldsymbol{x_1},y_1),\ldots,(\boldsymbol{x_M},y_M)\}$ 和模型参数 $\boldsymbol{\theta}$, 似然函数为

$$\begin{aligned} \mathcal{L}(\boldsymbol{\theta}|\boldsymbol{X},\boldsymbol{y}) &= p(\boldsymbol{y}|\boldsymbol{X};\boldsymbol{\theta}) \\ &= \prod_{m=1}^{M} h_{\boldsymbol{\theta}}(\boldsymbol{x_m})^{y_m}(1-h_{\boldsymbol{\theta}}(\boldsymbol{x_m}))^{1-y_m} \end{aligned} \tag{29}$$

我们取损失函数为负对数似然函数

$$\begin{aligned} E(\boldsymbol{\theta}) &= -\log\mathcal{L}(\boldsymbol{\theta}|\boldsymbol{X},\boldsymbol{y}) \\ &= -\sum_{m=1}^{M}(y_m\log h_{\boldsymbol{\theta}}(\boldsymbol{x_m}) + (1-y_m)\log(1-h_{\boldsymbol{\theta}}(\boldsymbol{x_m}))) \end{aligned} \tag{30}$$

参数 $\boldsymbol{\theta}$ 的最大似然估计为

$$\hat{\boldsymbol{\theta}} = \arg\min_{\boldsymbol{\theta}} E(\boldsymbol{\theta}) \tag{31}$$

E 关于参数 $\boldsymbol{\theta}$ 的每一个分量的偏导数为

$$\begin{aligned}
\frac{\partial E}{\partial \theta^{(n)}} &= -\frac{\partial}{\partial \theta^{(n)}} \sum_{m=1}^{M} (y_m \log h_{\boldsymbol{\theta}}(\boldsymbol{x_m}) + (1-y_m)\log(1-h_{\boldsymbol{\theta}}(\boldsymbol{x_m}))) \\
&= -\sum_{m=1}^{M} (y_m \frac{\partial \log h_{\boldsymbol{\theta}}(\boldsymbol{x_m})}{\partial \theta^{(n)}} + (1-y_m)\frac{\partial \log(1-h_{\boldsymbol{\theta}}(\boldsymbol{x_m}))}{\partial \theta^{(n)}}) \\
&= -\sum_{m=1}^{M} (y_m \frac{\partial \log g(\boldsymbol{\theta}^T \boldsymbol{x_m})}{\partial g(\boldsymbol{\theta}^T \boldsymbol{x_m})} \frac{\partial g(\boldsymbol{\theta}^T \boldsymbol{x_m})}{\partial \boldsymbol{\theta}^T \boldsymbol{x_m}} \frac{\partial \boldsymbol{\theta}^T \boldsymbol{x_m}}{\partial \theta^{(n)}} + \\
&\qquad (1-y_m)\frac{\partial \log(1-g(\boldsymbol{\theta}^T \boldsymbol{x_m}))}{\partial (1-g(\boldsymbol{\theta}^T \boldsymbol{x_m})} \frac{\partial (1-g(\boldsymbol{\theta}^T \boldsymbol{x_m})}{\partial \boldsymbol{\theta}^T \boldsymbol{x_m}} \frac{\partial \boldsymbol{\theta}^T \boldsymbol{x_m}}{\partial \theta^{(n)}}) \\
&= -\sum_{m=1}^{M} (\frac{y_m}{g(\boldsymbol{\theta}^T \boldsymbol{x_m})} g(\boldsymbol{\theta}^T \boldsymbol{x_m})(1-g(\boldsymbol{\theta}^T \boldsymbol{x_m}))x_m^{(n)} + \frac{1-y_m}{1-g(\boldsymbol{\theta}^T \boldsymbol{x_m})}(-g(\boldsymbol{\theta}^T \boldsymbol{x_m}))(1-g(\boldsymbol{\theta}^T \boldsymbol{x_m}))x_m^{(n)}) \\
&= -\sum_{m=1}^{M} (y_m(1-g(\boldsymbol{\theta}^T)\boldsymbol{x_m})x_m^{(n)} - (1-y_m)g(\boldsymbol{\theta}^T \boldsymbol{x_m})x_m^{(n)}) \\
&= -\sum_{m=1}^{M} (y_m - g(\boldsymbol{\theta}^T \boldsymbol{x_m}))x_m^{(n)} \\
&= -\sum_{m=1}^{M} (y_m - h_{\boldsymbol{\theta}}(\boldsymbol{x_m}))x_m^{(n)}
\end{aligned} \tag{32}$$

则 E 关于参数 $\boldsymbol{\theta}$ 的梯度为

$$\begin{aligned}
\nabla_{\boldsymbol{\theta}} E &= -\sum_{m=1}^{M} (y_m - h_{\boldsymbol{\theta}}(\boldsymbol{x}))\boldsymbol{x_m} \\
&= \boldsymbol{X}^T(\tilde{\boldsymbol{y}} - \boldsymbol{y})
\end{aligned} \tag{33}$$

其中

$$\tilde{\boldsymbol{y}} = \begin{pmatrix} h_{\boldsymbol{\theta}}(\boldsymbol{x_1}) \\ \vdots \\ h_{\boldsymbol{\theta}}(\boldsymbol{x_M}) \end{pmatrix} \tag{34}$$

得到梯度之后, 我们可以使用梯度方法来进行迭代求解最优值.

此外, 当样本数量 M 比较小的时候可以通过二阶收敛的牛顿方法来进行迭代.E 的二阶偏导数为

$$\begin{aligned}\frac{\partial^2 E}{\partial\theta^{(n)}\partial\theta^{(j)}} &= \frac{\partial}{\partial\theta^{(j)}}\sum_{m=1}^{M}(h_{\boldsymbol{\theta}}(\boldsymbol{x_m})-y_m)x_m^{(n)}\\ &= \sum_{m=1}^{M}\frac{\partial h_{\boldsymbol{\theta}}(\boldsymbol{x_m})x_m^{(n)}}{\partial\theta^{(j)}}\\ &= \sum_{m=1}^{M}x_m^{(n)}\frac{\partial g(\boldsymbol{\theta}^T\boldsymbol{x_m})}{\partial\boldsymbol{\theta}^T\boldsymbol{x_m}}\frac{\partial\boldsymbol{\theta}^T\boldsymbol{x_m}}{\partial\theta^{(j)}}\\ &= \sum_{m=1}^{M}x_m^{(n)}g(\boldsymbol{\theta}^T\boldsymbol{x_m})(1-g(\boldsymbol{\theta}^T\boldsymbol{x_m}))x_m^{(j)}\\ &= \sum_{m=1}^{M}x_m^{(n)}h_{\boldsymbol{\theta}}(\boldsymbol{x_m})(1-h_{\boldsymbol{\theta}}(\boldsymbol{x_m}))x_m^{(j)}\end{aligned} \tag{35}$$

由 E 的所有二阶偏导数构成的 Hessian 矩阵为

$$\begin{aligned}\boldsymbol{H} &= \nabla_{\boldsymbol{\theta}}^2 E\\ &= \boldsymbol{X}^T\boldsymbol{R}\boldsymbol{X}\end{aligned} \tag{36}$$

其中

$$(\boldsymbol{H})_{i,j} = \frac{\partial^2 E}{\partial\theta^{(i)}\partial\theta^{(j)}},\quad \boldsymbol{R} = diag(h_{\boldsymbol{\theta}}(\boldsymbol{x_1}),\ldots,h_{\boldsymbol{\theta}}(\boldsymbol{x_M})) \tag{37}$$

牛顿法的迭代更新公式为

$$\boldsymbol{\theta} := \boldsymbol{\theta} - \boldsymbol{H}^{-1}\nabla_{\boldsymbol{\theta}}E. \tag{38}$$

2.1 Softmax regression

Softmax 回归是 Logistic 回归的多分类推广版本. 在 K 类的分类问题中, 这里先引入"1-of-K"或"one-hot" 的标签向量 $\boldsymbol{y}$. 它是一个 K 维的布尔值列向量, 并且第 k 类的标签向量满足

$$y^{(j)} = \begin{cases}1, & j=k\\ 0, & j\neq k\end{cases},\quad j=1,\ldots,K. \tag{39}$$

对于一个给定的点 $\boldsymbol{x}$, 其类别的后验概率为

$$p(\boldsymbol{y}|\boldsymbol{x},\boldsymbol{\Theta}) = \prod_{k=1}^{K}(\phi^{(k)})^{y^{(k)}} \tag{40}$$

其中

$$\begin{aligned}\phi^{(k)} &= \frac{\exp(z^{(j)})}{\sum_{j=1}^{K}\exp(z^{(j)})}\\ z^{(j)} &= \sum_{n=0}^{N}\theta_j^{(n)}x^{(n)}\\ &= \boldsymbol{\theta_j}^T\boldsymbol{x},\quad j=1,\ldots,K\end{aligned} \tag{41}$$

对于一个训练数据集 $D=\{(\boldsymbol{x_m},\boldsymbol{y_m}):m=1,\ldots,M\}$ 和模型参数 $\boldsymbol{\theta_k},b_k,k=1,\ldots,K$, 将训练数据的输入集和标签集分别记为矩阵 $\boldsymbol{X},\boldsymbol{Y}$, 并将所有参数记为 $\boldsymbol{\Theta}$, 似然函数为

$$L(\boldsymbol{\Theta})=p(\boldsymbol{Y}|\boldsymbol{X};\boldsymbol{\Theta})=\prod_{m=1}^{M}\prod_{k=1}^{K}(\phi_m^{(k)})^{y_m^{(k)}} \tag{42}$$

为了最大化似然函数, 我们取损失函数为负对数似然函数

$$\begin{aligned}E(\boldsymbol{\Theta})&=-\log L(\boldsymbol{\Theta})\\&=-\sum_{m=1}^{M}\sum_{k=1}^{K}y_m^{(k)}\log\phi_m^{(k)}\end{aligned} \tag{43}$$

损失函数 E 关于 $\theta_j^{(n)}$ 的偏导数为

$$\begin{aligned}\frac{\partial E}{\partial\theta_j^{(n)}}&=-\sum_{m=1}^{M}\sum_{k=1}^{K}\frac{\partial y_m^{(k)}\log\phi_m^{(k)}}{\partial\theta_j^{(n)}}\\&=-\sum_{m=1}^{M}\sum_{k=1}^{K}y_m^{(k)}\frac{\partial\log\phi_m^{(k)}}{\partial\phi_m^{(k)}}\frac{\partial\phi_m^{(k)}}{\partial\theta_j^{(n)}}\\&=-\sum_{m=1}^{M}\sum_{k=1}^{K}y_m^{(k)}\frac{\partial\log\phi_m^{(k)}}{\partial\phi_m^{(k)}}\sum_{l=1}^{K}\frac{\partial\phi_m^{(k)}}{\partial\exp(z_m^{(l)})}\frac{\partial\exp(z_m^{(l)})}{\partial z_m^{(l)}}\frac{\partial z_m^{(l)}}{\partial\theta_j^{(n)}}\\&=-\sum_{m=1}^{M}\sum_{k=1}^{K}\frac{y_m^{(k)}}{\phi_m^{(k)}}\sum_{l=1}^{K}\frac{\partial\phi_m^{(k)}}{\partial\exp(z_m^{(l)})}\frac{\partial\exp(z_m^{(l)})}{\partial z_m^{(l)}}\frac{\partial\sum_{i=1}^{N}\theta_l^{(i)}x_m^{(i)}}{\partial\theta_j^{(n)}}\\&=-\sum_{m=1}^{M}\sum_{k=1}^{K}\frac{y_m^{(k)}}{\phi_m^{(k)}}(\frac{1\{j=k\}}{\sum_{h=1}^{K}\exp(z_m^{(h)})}-\frac{\exp(z_m^{(k)})}{(\sum_{h=1}^{K}\exp(z_m^{(h)}))^2})\exp(z_m^{(j)})x_m^{(n)}\\&=-\sum_{m=1}^{M}\sum_{k=1}^{K}\frac{y_m^{(k)}}{\phi_m^{(k)}}\frac{1\{j=k\}\exp(z_m^{(j)})\sum_{h=1}^{K}\exp(z_m^{(h)})-\exp(z_m^{(j)})\exp(z_m^{(k)})}{(\sum_{h=1}^{K}\exp(z_m^{(h)}))^2}x_m^{(n)}\\&=-\sum_{m=1}^{M}\sum_{k=1}^{K}\frac{y_m^{(k)}}{\phi_m^{(k)}}\frac{\exp(z_m^{(k)})}{\sum_{h=1}^{K}\exp(z_m^{(h)})}\frac{1\{j=k\}\sum_{h=1}^{K}\exp(z_m^{(h)})-\exp(z_m^{(k)})}{\sum_{h=1}^{K}\exp(z_m^{(h)})}x_m^{(n)}\\&=-\sum_{m=1}^{M}\sum_{k=1}^{K}\frac{y_m^{(k)}}{\phi_m^{(k)}}\phi_m^{(k)}(1\{j=k\}-\phi_m^{(j)})x_m^{(n)}\\&=-\sum_{m=1}^{M}x_m^{(n)}\sum_{k=1}^{K}y_m^{(k)}(1\{j=k\}-\phi_m^{(j)})\\&=-\sum_{m=1}^{M}x_m^{(n)}(\sum_{k=1}^{K}y_m^{(k)}1\{j=k\}-\phi_m^{(j)}\sum_{k=1}^{K}y_m^{(k)})\\&=-\sum_{m=1}^{M}x_m^{(n)}(y_m^{(j)}-\phi_m^{(j)})\\&=\sum_{m=1}^{M}(\phi_m^{(j)}-y_m^{(j)})x_m^{(n)}\end{aligned} \tag{44}$$

从而 E 关于 $\boldsymbol{\theta_j}$ 的梯度为

$$\nabla_{\boldsymbol{\theta_j}}E=\boldsymbol{X}^T(\boldsymbol{\Phi_{\cdot j}}-\boldsymbol{Y_{\cdot j}}) \tag{45}$$

其中

$$\boldsymbol{X}=\left(\boldsymbol{x_1},\ldots,\boldsymbol{x_M}\right)^T,\quad \boldsymbol{Y}=\left(\boldsymbol{y_1},\ldots,\boldsymbol{y_M}\right)^T,\quad \boldsymbol{\Phi}=\left(\boldsymbol{\phi_1},\ldots,\boldsymbol{\phi_M^T}\right) \tag{46}$$

$\boldsymbol{\Phi}_{\boldsymbol{\cdot j}},\boldsymbol{Y}_{\boldsymbol{\cdot,j}}$ 分别表示 $\boldsymbol{\Phi},\boldsymbol{Y}$ 的第 j 列.

进一步地, 二阶偏导数为

$$\begin{aligned}
\frac{\partial^2 E}{\partial\theta_j^{(n)}\partial\theta_i^{(k)}} &= \frac{\partial}{\partial\theta_i^{(k)}}\sum_{m=1}^{M}(\phi_m^{(j)}-y_m^{(j)})x_m^{(n)}\\
&=\sum_{m=1}^{M}\frac{\partial\phi_m^{(j)}x_m^{(n)}}{\partial\theta_i^{(k)}}\\
&=\sum_{m=1}^{M}x_m^{(n)}\sum_{h=1}^{K}\frac{\partial\phi_m^{(j)}}{\partial\exp(\boldsymbol{\theta_h^T x_m})}\frac{\partial\exp(\boldsymbol{\theta_h^T x_m})}{\partial\boldsymbol{\theta_h^T x_m}}\frac{\partial\boldsymbol{\theta_h^T x_m}}{\partial\theta_i^{(k)}}\\
&=\sum_{m=1}^{M}x_m^{(n)}\frac{\partial\phi_m^{(j)}}{\partial\exp(\boldsymbol{\theta_i^T x_m})}\frac{\partial\exp(\boldsymbol{\theta_i^T x_m})}{\partial\boldsymbol{\theta_i^T x_m}}x_m^{(k)}\\
&=\sum_{m=1}^{M}x_m^{(n)}\left(\frac{1\{j=i\}}{\sum_{h=1}^{K}\exp(\boldsymbol{\theta_h^T x_m})}-\frac{\exp(\boldsymbol{\theta_j^T x_m})}{(\sum_{h=1}^{K}\exp(\boldsymbol{\theta_h^T x_m}))^2}\right)\exp(\boldsymbol{\theta_i^T x_m})x_m^{(k)}\\
&=\sum_{m=1}^{M}x_m^{(n)}\left(\frac{1\{j=i\}\exp(\boldsymbol{\theta_i^T x_m})}{\sum_{h=1}^{K}\exp(\boldsymbol{\theta_h^T x_m})}-\frac{\exp(\boldsymbol{\theta_j^T x_m})\exp(\boldsymbol{\theta_i^T x_m})}{(\sum_{h=1}^{K}\exp(\boldsymbol{\theta_h^T x_m}))^2}\right)x_m^{(k)}\\
&=\sum_{m=1}^{M}(1\{j=i\}-\phi_m^{(j)})\phi_m^{(i)}x_m^{(n)}x_m^{(k)}\\
&=\boldsymbol{X_{\cdot n}^T R_{ji} X_{\cdot k}}
\end{aligned}$$

其中 $\boldsymbol{X_{\cdot n}},\boldsymbol{X_{\cdot k}}$ 分别表示 $\boldsymbol{X}$ 的第 n 列和第 k 列.$\boldsymbol{R_{ji}}$ 为对角矩阵

$$\boldsymbol{R_{ji}}=\mathrm{diag}((1\{j=i\}-\phi_1^j)\phi_1^i,\ldots,(1\{j=i\}-\phi_M^j)\phi_M^i) \tag{47}$$

3 Gaussian discriminant analysis

现在我们不仅仅考虑 y 在给定 $\boldsymbol{x}$ 下的后验分布. 假设我们获得的数据点 $\boldsymbol{x}$ 是由 $y\in\{0,1\}$ 两个不同的高斯类生成的, 满足

$$\begin{aligned}
y&\sim\mathrm{Bernoulli}(\phi)\\
\boldsymbol{x}|y&\sim\mathcal{N}(\boldsymbol{\mu_y},\boldsymbol{\Sigma})
\end{aligned}$$

那么对于给定的一点 $\boldsymbol{x}$, 根据贝叶斯公式 y 的后验分布为

$$\begin{aligned}
p(y|\boldsymbol{x};\phi,\boldsymbol{\mu_0},\boldsymbol{\mu_1},\boldsymbol{\Sigma}) &= \frac{p(\boldsymbol{x}|y;\boldsymbol{\mu_y},\boldsymbol{\Sigma})p(y|\phi)}{p(\boldsymbol{x})}\\
&=\frac{p(\boldsymbol{x}|y;\boldsymbol{\mu_y},\boldsymbol{\Sigma})p(y|\phi)}{\sum_{y=0}^{1}p(\boldsymbol{x}|y;\boldsymbol{\mu_y},\boldsymbol{\Sigma})p(y|\phi)}
\end{aligned} \tag{48}$$

分布参数可以通过最大似然的方式来估计

$$\begin{aligned}\phi &= \frac{1}{M}\sum_{m=1}^{M} 1\{y_i = 1\} \\ \boldsymbol{\mu_0} &= \frac{\sum_{m=1}^{M} 1\{y_i = 0\}\boldsymbol{x_i}}{\sum_{m=1}^{M} 1\{y_i = 0\}} \\ \boldsymbol{\mu_1} &= \frac{\sum_{m=1}^{M} 1\{y_i = 1\}\boldsymbol{x_i}}{\sum_{m=1}^{M} 1\{y_i = 1\}} \\ \boldsymbol{\Sigma} &= \frac{1}{M}\sum_{m=1}^{M}(\boldsymbol{x_i} - \boldsymbol{\mu_{y_i}})(\boldsymbol{x_i} - \boldsymbol{\mu_{y_i}})^T\end{aligned}$$

这样, 对于一个新的输入值 $\boldsymbol{x}$, 我们可以根据后验分布来估计它所属的高斯类 y

$$\begin{aligned}\hat{y} &= arg\max_y \frac{p(\boldsymbol{x}|y)p(y)}{p(\boldsymbol{x})} \\ &= arg\max_y p(\boldsymbol{x}|y)p(y)\end{aligned} \tag{49}$$

3.1 Generalization

我们可以将高斯判别分析推广到多类的情况, 也可以让每一类具有不同的协方差 $\boldsymbol{\Sigma_k}$

$$\begin{aligned}p(y) &= \prod_{k=1}^{K} \phi_k^{y^{(k)}} \\ (\boldsymbol{x}|y = k) &\sim \mathcal{N}(\boldsymbol{\mu_k}, \boldsymbol{\Sigma_k}), \quad k = 1, \ldots, K\end{aligned}$$

4 Navie Bayes

考虑一个离散的输入空间 $\mathcal{X}$, 其中的点 $\boldsymbol{x} = (x_1, \ldots, x_N)^T$ 的分量 x_j 有 S_j 个取值. 设输出空间为 $\mathcal{Y} = \{c_1, \ldots, c_K\}$, 即有 K 类. 假设类别 y 的先验分布

$$p(y = k; \boldsymbol{\phi}) = \phi_k \tag{50}$$

由于在给定 y 下,$\boldsymbol{x}$ 的条件分布

$$p(\boldsymbol{x}|y) = p(x_1, \ldots, x_N|y) \tag{51}$$

是一个所有分量 $x_j, (j = 1, \ldots, N)$ 的联合分布, 我们很难利用有限的数据完成对 $p(\boldsymbol{x}|y)$ 模型的建立. 我们可以做出一个很强的假设, 在给定 y 下 $\boldsymbol{x}$ 的各个分量相互独立. 这就是朴素贝叶斯假设, 我们得到的是一个朴素贝叶斯分类器. 由于 $x_1, \ldots, x_N$ 相互独立, 有

$$\begin{aligned}p(\boldsymbol{x}|y) &= p(x_1, \ldots, x_N|y) \\ &= p(x_1|y)\ldots p(x_N|y) \\ &= \prod_{n=1}^{N} p(x_n|y)\end{aligned} \tag{52}$$

x_n 在 y 下的条件分布为

$$p(x_n = a_{nj}|y) = \varphi_{nj|y} \tag{53}$$

对 $\boldsymbol{\phi}$ 进行最大似然估计, 有

$$\phi_k = \frac{\sum_{m=1}^{M} 1\{y_m = c_k\}}{M}, \quad k = 1, \ldots, K \tag{54}$$

对 $\boldsymbol{\varphi}$ 进行最大似然估计, 有

$$\begin{gathered} \varphi_{nj|c_k} = \frac{\sum_{m=1}^{M} 1\{x_{mn} = a_{nj}, y_m = c_k\}}{\sum_{m=1}^{M} 1\{y_m = c_k\}} \\ m = 1, \ldots, M, \quad n = 1, \ldots, N, \quad j = 1, \ldots, S_j \end{gathered} \tag{55}$$

这样, 我们得到了先验分布 $p(y)$ 和条件分布 $p(\boldsymbol{y})$. 对于一个新的点 $\boldsymbol{x}$, 我们希望对它的类别 y 进行预测, 只需要求出它的后验分布 $y|\boldsymbol{x}$. 根据贝叶斯公式

$$\begin{aligned} p(y|\boldsymbol{x}) &= \frac{p(\boldsymbol{x}|y)p(y)}{\sum_{k=1}^{K} p(\boldsymbol{x}|y = c_k)p(y = c_k)} \\ &= \frac{p(y)\prod_{n=1}^{N} p(x_n|y)}{\sum_{k=1}^{K} p(y = c_k)\prod_{n=1}^{N} p(x_n|y = c_k)} \end{aligned} \tag{56}$$

得到后验分布之后, 我们可以直接预测

$$\hat{y} = \arg\max_{y} p(y|\boldsymbol{x}) \tag{57}$$

4.1 Laplacian smoothing

在最大似然估计 (54),(55) 中, 可能会出现

$$\sum_{m=1}^{M} 1\{y_m = c_k\} = 0 \tag{58}$$

$$\sum_{m=1}^{M} 1\{x_{mn} = a_{nj}, y_m = c_k\} = 0 \tag{59}$$

的情况, 导致得到极端或者没有意义的参数. 我们可以使用拉普拉斯平滑 (Laplacian smoothing) 来进行修正

$$\phi_k = \frac{\sum_{m=1}^{M} 1\{y_m = c_k\} + 1}{M + K} \tag{60}$$

$$\varphi_{nj|c_k} = \frac{\sum_{m=1}^{M} 1\{x_{mn} = a_{nj}, y_m = c_k\} + 1}{\sum_{m=1}^{M} 1\{y_m = c_k\} + S_j} \tag{61}$$

5 Bayesian statistics and regularization

5.1 Maximum a posteriori

对于一组观测数据 D, 记其中的输入集为 $\boldsymbol{X}$, 输出集为 $\boldsymbol{y}$. 在频率学派的观点中, 模型参数 $\boldsymbol{\theta}$ 被视作一个未知常量. 对于一个给定的输入 $\boldsymbol{x}$, 输出值 y 的条件分布为

$$p(y|\boldsymbol{x}; \boldsymbol{\theta}) \tag{62}$$

而最大化似然函数

$$\hat{\boldsymbol{\theta}} = \arg\max_{\boldsymbol{\theta}} p(\boldsymbol{y}|\boldsymbol{X};\boldsymbol{\theta}) \tag{63}$$

就是用来估计这个参数.

现在我们考虑贝叶斯学派的观点, 我们把 $\boldsymbol{\theta}$ 视作一个随机变量, 并为它设定一个先验分布 $p(\boldsymbol{\theta})$. 对于给定的 $\boldsymbol{x}$, 输出值 y 的条件分布为

$$p(y|\boldsymbol{x}) = \int_{\boldsymbol{\theta}} p(y|\boldsymbol{x},\boldsymbol{\theta})p(\boldsymbol{\theta})d\boldsymbol{\theta} \tag{64}$$

我们希望在观测数据 $\boldsymbol{X},\boldsymbol{y}$ 以及输入值 $\boldsymbol{x}$ 的条件下对输出值 y 进行预测. 根据贝叶斯公式

$$p(\boldsymbol{y},\boldsymbol{\theta}|\boldsymbol{X}) = p(\boldsymbol{\theta}|\boldsymbol{X},\boldsymbol{y})p(\boldsymbol{y}|\boldsymbol{X}) \tag{65}$$

$$\begin{aligned} p(\boldsymbol{\theta}|\boldsymbol{X},\boldsymbol{y}) &= \frac{p(\boldsymbol{y},\boldsymbol{\theta}|\boldsymbol{X})}{p(\boldsymbol{y}|\boldsymbol{X})} \\ &= \frac{p(\boldsymbol{y}|\boldsymbol{X},\boldsymbol{\theta})p(\boldsymbol{\theta})}{\int_{\boldsymbol{\theta}} p(\boldsymbol{y}|\boldsymbol{X},\boldsymbol{\theta})p(\boldsymbol{\theta})d\boldsymbol{\theta}} \end{aligned} \tag{66}$$

我们得到了参数 $\boldsymbol{\theta}$ 后验分布的表达式. 这样我们就能写出 y 的条件分布的表达式

$$p(y|\boldsymbol{x},\boldsymbol{X},\boldsymbol{y}) = \int_{\boldsymbol{\theta}} p(y|\boldsymbol{x},\boldsymbol{\theta})p(\boldsymbol{\theta}|\boldsymbol{X},\boldsymbol{y})d\boldsymbol{\theta} \tag{67}$$

进一步, 我们可以写出 y 的条件期望, 并用来作为预测值

$$\hat{y} = \mathbb{E}(y|\boldsymbol{x},\boldsymbol{X},\boldsymbol{y}) = \int_y yp(\boldsymbol{y}|\boldsymbol{x},\boldsymbol{X},\boldsymbol{y})dy \tag{68}$$

遗憾的是, 参数的后验分布 (66) 通常难以计算, 并且没有解析解. 因此我们无法通过简单的方法来求解式 (68) 得到 y 的预测值. 在这里, 我们使用最大化后验概率 (maximum a posterior) 来估计 $\boldsymbol{\theta}$

$$\boldsymbol{\theta_{MAP}} = \arg\max_{\boldsymbol{\theta}} p(\boldsymbol{\theta}|\boldsymbol{X},\boldsymbol{y}) \tag{69}$$

根据式 (66), 我们有

$$p(\boldsymbol{\theta}|\boldsymbol{X},\boldsymbol{y}) = \frac{p(\boldsymbol{y}|\boldsymbol{X},\boldsymbol{\theta})p(\boldsymbol{\theta})}{p(\boldsymbol{y}|\boldsymbol{X})} \tag{70}$$

又由于 $p(\boldsymbol{y}|\boldsymbol{X})$ 和 $\boldsymbol{\theta}$ 无关, 式 (69) 可以写为

$$\boldsymbol{\theta_{MAP}} = \arg\max_{\boldsymbol{\theta}} P(\boldsymbol{y}|\boldsymbol{X},\boldsymbol{\theta})p(\boldsymbol{\theta}). \tag{71}$$

注意到, 这和之前的最大似然估计 (13) 仅相差一个先验分布 $p(\boldsymbol{\theta})$.

5.2 Regularization

给参数 $\boldsymbol{\theta}$ 设置先验分布的一个作用是限制模型的复杂度, 抑制过拟合. 如果我们设置 $\boldsymbol{\theta}$ 的每个分量相互独立且服从相同的零均值高斯分布

$$\theta_i \sim \mathcal{N}(0,\alpha) \tag{72}$$

即

$$\begin{aligned} p(\boldsymbol{\theta}) &= \mathcal{N}(\boldsymbol{\theta}|\boldsymbol{0},\alpha\boldsymbol{I}) \\ &= \frac{1}{(2\pi)^{k/2}|\boldsymbol{\Sigma}|^{1/2}} \exp(-\frac{1}{2}\boldsymbol{\theta}^T(\alpha\boldsymbol{I})\boldsymbol{\theta}) \end{aligned} \tag{73}$$

那么

$$p(\boldsymbol{y}|\boldsymbol{X},\boldsymbol{\theta}) \propto p(\boldsymbol{y}|\boldsymbol{X},\boldsymbol{\theta}) \exp(-\alpha \boldsymbol{\theta}^T \boldsymbol{\theta}) \tag{74}$$

取负对数, 有

$$\boldsymbol{\theta_{MAP}} = \arg\min_{\boldsymbol{\theta}} (\log p(\boldsymbol{y}|\boldsymbol{X},\boldsymbol{\theta}) + \alpha \boldsymbol{\theta}^T \boldsymbol{\theta}) \tag{75}$$

这等价于在最小化负对数似然添加一个 $L2$ 正则化项, 也被称为权重衰减项.

6 Model selection

当我们选定某一类模型, 有时我们仍需要指定一些控制模型的参数. 例如当使用多项式进行线性回归的时候, 我们需要确定使用多少阶的多项式. 显然, 阶数越高, 损失函数 E 最小值降得越低. 但不意味着模型对新数据具有很好的预测能力. 因为复杂的模型会带来严重的过拟合.

我们需要在一类模型中, 选择泛化能力较好的一个. 常用的方法是交叉验证. 把数据划分为不相交的两部分, 分别是训练集和验证集. 用训练集来训练模型的参数, 然后用验证集来评估模型的泛化能力. 对比不同的模型, 从而选择最优的.

- **简单交叉验证**, 将数据的 70% 作为训练集,30% 作为验证集, 对多个模型进行训练和评估. 最后选择误差最小的模型.
- **K-折交叉验证**, 将数据划分称 K 个大小相同的子集. 重复 K 次用不同的 $K-1$ 个子集的并来训练, 用余下的一个子集来验证. 最后选择 K 次的平均误差最小的模型.
- **留一法**, 当数据量 M 非常小的时候, 可以考虑使用留一法的交叉验证. 每次使用不同的 $M-1$ 个数据进行训练, 用余下的一个数据进行验证, 最后选择平均误差最小的模型.

除了交叉验证, 还可以使用自助法 (bootstrap) 进行重抽样. 对含有 M 个数据的数据集进行 M 次有放回均匀抽样. 用抽样得到的数据来训练, 用余下的数据来验证. 我们可以通过多次进行重抽样, 得到多组不同的训练集和验证集.

7 Lagrange duality

7.1 Primal optimization problem

对于原始优化问题

$$\begin{aligned} &\min_{\boldsymbol{\omega}} f(\boldsymbol{\omega}) \\ s.t. \quad &g_i(\boldsymbol{\omega}) \le 0, \quad i = 1, \dots, k, \\ &h_i(\boldsymbol{\omega}) = 0, \quad i = q, \dots, l. \end{aligned} \tag{76}$$

引入广义拉格朗日函数为

$$\mathcal{L}(\boldsymbol{\omega}, \boldsymbol{\alpha}, \boldsymbol{\beta}) = f(\boldsymbol{\omega}) + \sum_{i=1}^{k} \alpha_i g_i(\boldsymbol{\omega}) + \sum_{i=1}^{l} \beta_i h_i(\boldsymbol{\omega}) \tag{77}$$

这里,α_i, β_j 为拉格朗日乘子,$\alpha \geq 0$. 考虑 $\boldsymbol{x}$ 的函数

$$\theta_{\mathcal{P}}(\boldsymbol{\omega}) = \max_{\boldsymbol{\alpha},\boldsymbol{\beta}:\alpha_i>0} \mathcal{L}(\boldsymbol{\omega}, \boldsymbol{\alpha}, \boldsymbol{\beta}) \tag{78}$$

这里, 下标 $\mathcal{P}$ 表示原始问题. 容易发现

$$\theta_{\mathcal{P}} = \begin{cases} f(\boldsymbol{\omega}) & ,\boldsymbol{\omega} \text{ 满足问题 (76) 的约束条件} \\ +\infty & ,\text{其他情况} \end{cases} \tag{79}$$

现在考虑优化问题

$$\min_{\boldsymbol{\omega}} \theta_{\mathcal{P}}(\boldsymbol{\omega}) = \min_{\boldsymbol{\omega}} \max_{\boldsymbol{\alpha},\boldsymbol{\beta}:\alpha_i\geq 0} \mathcal{L}(\boldsymbol{\omega}, \boldsymbol{\alpha}, \boldsymbol{\beta}), \tag{80}$$

它与原始优化问题是等价的, 有相同的解. 这样就把原始问题表示为广义拉格朗日函数的极小极大问题. 为了方便, 记原始优化问题的最优值为

$$p^* = \min_{\boldsymbol{\omega}} \theta_{\mathcal{P}}(\boldsymbol{\omega}) \tag{81}$$

7.2 Dual optimization problem

现在考虑 $\boldsymbol{\alpha}, \boldsymbol{\beta}$ 的函数

$$\theta_{\mathcal{D}}(\boldsymbol{\alpha}, \boldsymbol{\beta}) = \min_{\boldsymbol{\omega}} \mathcal{L}(\boldsymbol{\omega}, \boldsymbol{\alpha}, \boldsymbol{\beta}) \tag{82}$$

将其极大化

$$\max_{\boldsymbol{\alpha},\boldsymbol{\beta}:\alpha_i\geq 0} \theta_{\mathcal{D}}(\boldsymbol{\alpha}, \boldsymbol{\beta}) = \max_{\boldsymbol{\alpha},\boldsymbol{\beta}:\alpha_i\geq 0} \min_{\boldsymbol{\omega}} \mathcal{L}(\boldsymbol{\omega}, \boldsymbol{\alpha}, \boldsymbol{\beta}) \tag{83}$$

称为原始问题的对偶问题. 记对偶问题的最优值为

$$d^* = \max_{\boldsymbol{\alpha},\boldsymbol{\beta}:\alpha_i\geq 0} \theta_{\mathcal{D}}(\boldsymbol{\alpha}, \boldsymbol{\beta}) \tag{84}$$

7.3 The relationship between the primal and the dual problems

定理 1 若原始问题和对偶问题都有最优值, 则

$$d^* = \max_{\boldsymbol{\alpha},\boldsymbol{\beta}:\alpha_i\geq 0} \min_{\boldsymbol{\omega}} \mathcal{L}(\boldsymbol{\omega}, \boldsymbol{\alpha}, \boldsymbol{\beta}) \leq \min_{\boldsymbol{\omega}} \max_{\boldsymbol{\alpha},\boldsymbol{\beta}:\alpha_i\geq 0} = p^* \tag{85}$$

定理 2 如果 f 和 g_i 是凸函数,h_j 是仿射函数; 并且假设不等式约束 g_i 是严格可行的, 即存在 $\boldsymbol{\omega}$ 使 $g_i(\boldsymbol{\omega}) < 0, \forall i = 1, \ldots, k$, 则存在 $\boldsymbol{\omega}^*, \boldsymbol{\alpha}^*, \boldsymbol{\beta}^*$ 使 $\boldsymbol{\omega}^*$ 是原始问题的解,$\boldsymbol{\alpha}^*, \boldsymbol{\beta}^*$ 是对偶问题的解, 并且

$$p^* = d^* = \mathcal{L}(\boldsymbol{\omega}^*, \boldsymbol{\alpha}^*, \boldsymbol{\beta}^*) \tag{86}$$

定理 3 如果 f 和 g_i 是凸函数,h_j 是仿射函数; 并且假设不等式约束 g_i 是严格可行的, 则 $\boldsymbol{\omega}^*$ 和 $\boldsymbol{\alpha}^*, \boldsymbol{\beta}^*$ 分别是原始问题和对偶问题的解的充分必要条件是 $\boldsymbol{\omega}^*, \boldsymbol{\alpha}^*, \boldsymbol{\beta}^*$ 满足下面的 Karush-Kuhn-Tucker(KKT)

条件

$$\begin{aligned}
&\nabla_{\boldsymbol{\omega}}\mathcal{L}(\boldsymbol{\omega}^*,\boldsymbol{\alpha}^*,\boldsymbol{\beta}^*)=0\\
&\nabla_{\boldsymbol{\alpha}}\mathcal{L}(\boldsymbol{\omega}^*,\boldsymbol{\alpha}^*,\boldsymbol{\beta}^*)=0\\
&\nabla_{\boldsymbol{\beta}}\mathcal{L}(\boldsymbol{\omega}^*,\boldsymbol{\alpha}^*,\boldsymbol{\beta}^*)=0\\
&\alpha_i^* g_i(\boldsymbol{\omega}^*)=0,\quad i=1,\ldots,k\\
&g_i(\boldsymbol{\omega}^*)\le 0,\quad i=1,\ldots,k\\
&\alpha_i\ge 0,\quad i=1,\ldots,k\\
&h_i(\boldsymbol{\omega}^*)=0,\quad i=1,\ldots,l
\end{aligned}\tag{87}$$

8 Support Vector Machine

8.1 Optimal margin classifier

设二分类问题中有一组线性可分的训练数据 $D=\{(\boldsymbol{x_m},y_m)\}:m=1,\ldots,M\}$, 其中 $\boldsymbol{x}\in\mathbb{R}^N, y\in\{-1,1\}$. 线性分类器为

$$h_{\boldsymbol{\omega},b}(\boldsymbol{x})=sign(\boldsymbol{\omega}^T\boldsymbol{x}+b)\tag{88}$$

从直观上来看, 对于某个样本点 $\boldsymbol{x_i}$, 若 $\boldsymbol{\omega}^T\boldsymbol{x}+b$ 越大, 则分类器对 $\boldsymbol{x_i}$ 分类的信心越大. 因此需要找到一组参数 $\boldsymbol{\omega},b$ 使分类器的决策边界将训练数据的正负样本完全分隔开, 满足

$$y_m(\boldsymbol{\omega}^T\boldsymbol{x_m}+b)>0,\quad m=1,\ldots,M,\tag{89}$$

并且希望分类器对训练数据分类的信心尽可能大, 即希望

$$y_m(\boldsymbol{\omega}^T\boldsymbol{x_m}+b)\gg 0,\quad m=1,\ldots,M.\tag{90}$$

$\hat{\gamma_m}=y(\boldsymbol{\omega_m}^T\boldsymbol{x_m}+b)$ 被称为第 m 个样本的函数间隔 (functional margin). 事实上, 按相同的比例对 $\boldsymbol{\omega},b$ 进行缩放, 可以对应地缩放 $\hat{\gamma}$ 的大小, 而不会改变分类器 $h_{\boldsymbol{\omega},b}$ 的决策边界. 因此无法通过最大化函数间隔来确定一个最优的决策边界.

现在需要引入几何间隔 (geometric margin) 的概念, 几何间隔 γ_m 定义为点 $\boldsymbol{x_m}$ 到决策边界 $W:\boldsymbol{\omega}^T\boldsymbol{x}+b=0$ 的欧拉距离. 已知决策边界是一个 N 维空间上的超平面, 其单位法向量为 $\frac{\boldsymbol{\omega}}{||\boldsymbol{\omega}||}$. 并且 $(\boldsymbol{x_m}-y_m\gamma_m\frac{\boldsymbol{\omega}}{||\boldsymbol{\omega}||})$ 是超平面上的一点, 于是有

$$\boldsymbol{\omega}^T(\boldsymbol{x_m}-y_m\gamma_m\frac{\boldsymbol{\omega}}{||\boldsymbol{\omega}||})+b=0\tag{91}$$

从而解得第 m 个样本的几何间隔

$$\begin{aligned}
\gamma_m&=\frac{y_m(\boldsymbol{\omega}^T\boldsymbol{x_m}+b)}{||\boldsymbol{\omega}||}\\
&=y_m((\frac{\boldsymbol{\omega}}{||\boldsymbol{\omega}||})^T\boldsymbol{x_m}+\frac{b}{||\boldsymbol{\omega}||})\\
&=\frac{\hat{\gamma_m}}{||\boldsymbol{\omega}||}
\end{aligned}\tag{92}$$

训练数据中每个样本点都有一个函数间隔和几何间隔, 我们用其中的最小值的来定义决策超平面的函数间隔和几何间隔

$$\begin{aligned}\hat{\gamma} &= \min_m \hat{\gamma_m},\\ \gamma &= \min_m \gamma_m.\end{aligned} \tag{93}$$

这样, 我们就可以选择最大间隔的决策超超平面来构建分类器. 对应的优化问题为

$$\begin{aligned}&\max_{\boldsymbol{\omega},b} \gamma\\ s.t.\quad &\gamma \le y_m((\frac{\boldsymbol{\omega}}{||\boldsymbol{\omega}||})^T \boldsymbol{x_m} + \frac{b}{||\boldsymbol{\omega}||}), \quad m = 1, \dots, M\end{aligned} \tag{94}$$

优化问题可以重新表达为

$$\begin{aligned}&\max_{\boldsymbol{\omega},b} \frac{\hat{\gamma}}{||\boldsymbol{\omega}||}\\ s.t.\quad &\hat{\gamma} \le y_m(\boldsymbol{\omega}^T \boldsymbol{x_m} + b), \quad m = 1, \dots, M.\end{aligned} \tag{95}$$

注意到 $\hat{\gamma}$ 在这里是个自由参数, 不影响优化问题的解, 因此可以令 $\hat{\gamma} = 1$, 从而优化问题为

$$\min_{\boldsymbol{\omega},b} \quad ||\boldsymbol{\omega}||^2 \tag{96}$$

$$s.t. \quad 1 \le y_m(\boldsymbol{\omega}^T \boldsymbol{x} + b), \quad m = 1, \dots, M \tag{97}$$

这是一个凸二次规划问题, 可以使用现有的算法库来直接求解.

8.2 Dual problem

约束条件 (97) 可以表达为

$$g_i(\boldsymbol{\omega}, b) = 1 - y_m(\boldsymbol{\omega}^T \boldsymbol{x_m} + b) \le 0, \quad m = 1, \dots, M \tag{98}$$

注意到这里只有不等式约束, 构建拉格朗日函数为

$$\mathcal{L}(\boldsymbol{\omega}, b, \boldsymbol{\alpha}) = \frac{1}{2}||\boldsymbol{\omega}||^2 + \sum_{m=1}^{M} \alpha_m (1 - y_m(\boldsymbol{\omega}^T \boldsymbol{x_m} + b)) \tag{99}$$

根据拉格朗日对偶性, 原始问题的对偶问题为

$$\max_{\boldsymbol{\alpha}} \min_{\boldsymbol{\omega},b} \mathcal{L}(\boldsymbol{\omega}, b, \boldsymbol{\alpha}) \tag{100}$$

为了求解对偶问题, 需要先对 $\mathcal{L}(\boldsymbol{\omega}, b, \boldsymbol{\alpha})$ 关于 $\boldsymbol{\omega}, b$ 求极小, 再对 $\boldsymbol{\alpha}$ 求极大.

将 $\mathcal{L}(\boldsymbol{\omega}, b, \boldsymbol{\alpha})$ 分别关于 $\boldsymbol{\omega}, b$ 求偏导数

$$\nabla_{\boldsymbol{\omega}} \mathcal{L}(\boldsymbol{\omega}, b, \boldsymbol{\alpha}) = \boldsymbol{\omega} - \sum_{m=1}^{M} \alpha_m y_m \boldsymbol{x_m}, \tag{101}$$

$$\nabla_b \mathcal{L}(\boldsymbol{\omega}, b, \boldsymbol{\alpha}) = \sum_{m=1}^{M} \alpha_m y_m, \tag{102}$$

令其等于 0 可以得到

$$\boldsymbol{\omega} = \sum_{m=1}^{M} \alpha_m y_m \boldsymbol{x_m}, \tag{103}$$

$$\sum_{m=1}^{M} \alpha_m y_m = 0. \tag{104}$$

将它们代入到式 99可以得到

$$\begin{aligned} \min_{\boldsymbol{\omega},b} \mathcal{L}(\boldsymbol{\omega}, b, \boldsymbol{\alpha}) &= \frac{1}{2}\boldsymbol{\omega}^T\boldsymbol{\omega} + \sum_{m=1}^{M}\alpha_m - \sum_{m=1}^{M}\boldsymbol{\omega}^T(\alpha_m y_m \boldsymbol{x_m}) - \sum_{m=1}^{M}\alpha_m y_m b \\ &= -\frac{1}{2}(\sum_{m=1}^{M}\alpha_m y_m \boldsymbol{x_m})^T(\sum_{m=1}^{M}\alpha_m y_m \boldsymbol{x_m}) + \sum_{m=1}^{M}\alpha_m \\ &= -\frac{1}{2}\sum_{i=1}^{M}\sum_{j=1}^{M}\alpha_i\alpha_j y_i y_j \boldsymbol{x_i}^T\boldsymbol{x_j} + \sum_{m=1}^{M}\alpha_m \end{aligned} \tag{105}$$

再对 $\min_{\boldsymbol{\omega},b}\mathcal{L}(\boldsymbol{\omega}, b, \boldsymbol{\alpha})$ 求极大

$$\begin{aligned} \max_{\boldsymbol{\alpha}} \quad & -\frac{1}{2}\sum_{i=1}^{M}\sum_{j=1}^{M}\alpha_i\alpha_j y_i y_j \boldsymbol{x_i}^T\boldsymbol{x_j} + \sum_{m=1}^{M}\alpha_m \\ s.t. \quad & \sum_{m=1}^{M}\alpha_m y_m = 0 \\ & \alpha_m \geq 0, \quad m = 1, \ldots, M \end{aligned} \tag{106}$$

设该优化问题的解为 $\boldsymbol{\alpha}^* = (\alpha_1^*, \ldots, \alpha_N^*)$. 可以验证原始问题和对偶问题的 KKT 条件是成立的

$$\alpha_m^* \geq 0, \tag{107}$$

$$y_m(\boldsymbol{\omega}^*\boldsymbol{x_m} + b^* - 1) = 0, \tag{108}$$

$$\alpha_m^*(y_m(\boldsymbol{\omega}^*\boldsymbol{x_m} + b*) - 1) = 0, \tag{109}$$

因此

$$\boldsymbol{\omega}^* = \sum_{m=1}^{M}\alpha_m y_m \boldsymbol{x_m} \tag{110}$$

是原始问题的解.

对于训练数据中的每个点, 要么 $\alpha_m = 0$, 要么 $y_m(\boldsymbol{\omega}^{*T}\boldsymbol{x_m} + b^*) = 1$. 其中 $\alpha_m \geq 0$ 的点则被成为支持向量 (Support vector).

对于任一支持向量 j, 满足 $y_j(\boldsymbol{\omega}^*\boldsymbol{x_j} + b^*) = 1$, 并且 $y_j^2 = 1$

$$y_j(\boldsymbol{\omega}^{*T}\boldsymbol{x_j} + b^*) = 1 \tag{111}$$

$$y_j^2(\boldsymbol{\omega}^{*T}\boldsymbol{x_j} + b^*) = y_j \tag{112}$$

从而解得

$$b^* = y_j - \sum_{m=1}^{M}\alpha_m y_m \boldsymbol{x_m}^T\boldsymbol{x_j} \tag{113}$$

虽然只需要任意一个支持向量就能解的 b^*, 但是为了更好的数值稳定性, 可以分别利用所有支持向量解出 b_j^*, 然后再取它们的平均值.

对于一个新的点 $\boldsymbol{x}$, 它的预测值为

$$\begin{aligned} h(\boldsymbol{x}) &= sign(\boldsymbol{\omega}^{*T}\boldsymbol{x} + b^*) \\ &= sign(\sum_{m=1}^{M} \alpha_m^* y_m \boldsymbol{x_m}^T \boldsymbol{x} + b^*) \end{aligned} \tag{114}$$

注意到, 只有支持向量对预测函数有影响.

8.3 Kernel trick

定义设 $\mathbb{R}^n$ 是输入空间,$\mathcal{H}$ 是特征空间. 对于函数

$$K : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}, \tag{115}$$

如果存在一个从输入空间到特征空间的映射

$$\phi : \mathbb{R}^n \to \mathcal{H}, \tag{116}$$

满足

$$K(\boldsymbol{x}, \boldsymbol{y}) = \phi(\boldsymbol{x})^T \phi(\boldsymbol{y}), \tag{117}$$

对任意 $\boldsymbol{x}, \boldsymbol{y} \in \mathcal{X}$ 成立, 则称 $K(\boldsymbol{x}, \boldsymbol{y})$ 为核函数.

核函数的思想是不直接定义特征映射 ϕ, 而是使用核函数 $K(\boldsymbol{x}, \boldsymbol{y})$ 来直接计算特征空间上的内积 $\phi(\boldsymbol{x})^T\phi(\boldsymbol{y})$. 通常计算核函数 $K(\boldsymbol{x}, \boldsymbol{y})$ 的代价是远低于计算 $\phi(\boldsymbol{x})^T\phi(\boldsymbol{y})$ 的.

定理设对称函数 $K : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}$, 如果对任意 $\boldsymbol{x_m}$, $m = 1, \ldots, M$, 函数 K 对应的 Gram 矩阵 K

$$K = \Big(K(\boldsymbol{x_i}, \boldsymbol{x_j})\Big)_{M \times M}, \tag{118}$$

半正定矩阵, 则称 $K(\boldsymbol{x}, \boldsymbol{y})$ 是正定核. 即可以用作核函数.

当分类问题是非线性的时候, 需要使用非线性映射 ϕ 将数据从原始输入空间 $\mathbb{R}^M$ 映射到特征空间 $\mathcal{H}$, 从而在特征空间上进行学习. 利用核技巧, 可以使算法间接地在特征空间进行求解. 在支持向量机上使用核函数可以得到非线性支持向量机, 只需要将对偶问题 (106) 中输入空间上数据点内积 $\boldsymbol{x_i}^T\boldsymbol{x_j}$ 替换为核函数 $K(\boldsymbol{x_i}, \boldsymbol{x_j})$, 得到优化问题

$$\min_{\boldsymbol{\alpha}} \quad -\frac{1}{2}\sum_{i=1}^{M}\sum_{j=1}^{M} \alpha_i \alpha_j y_i y_j K(\boldsymbol{x_i}, \boldsymbol{x_j}) + \sum_{m=1}^{M} \alpha_m \tag{119}$$

$$s.t. \quad \sum_{m=1}^{M} \alpha_m y_m = 0 \tag{120}$$

$$\alpha_m \geq 0, \quad m = 1, \ldots, M \tag{121}$$

求得最优解 $\boldsymbol{\alpha}^* = (\alpha_1^*, \ldots, \alpha_M^*)^T$. 选择一个支持向量 j, 计算

$$b^* = y_j - \sum_{m=1}^{M} \alpha_m^* y_m K(\boldsymbol{x_m}, \boldsymbol{x_j}) \tag{122}$$

或者对所有支持向量计算 b_j* 再取平均值作为 b^*. 最后构建分线性支持向量机决策函数

$$h(\boldsymbol{x}) = sign(\sum_{m=1}^{M} \alpha_m^* y_m K(\boldsymbol{x_m}, \boldsymbol{x}) + b^*) \tag{123}$$

事实上在使用决策函数对新数据点进行预测的时候, 只需要使用支持向量进行计算.

8.4 Soft margin

以上推导的支持向量机是基于训练数据是线性可分, 或者是在特征空间上线性可分的. 除此之外, 支持向量机的决策边界对边界附近的数据是敏感的, 特别是在有异常数据的时候. 为了解决线性不可分的问题以及增加支持向量机的稳定性, 我们需要引进软间隔的学习策略. 允许数据有少量误差, 对每个数据点 $(\boldsymbol{x_m}, y_m)$ 引入松弛变量 $\xi_m \geq 0$, 约束条件 (97) 变为

$$y_m(\boldsymbol{\omega}^T \boldsymbol{x_m} + b) \geq 1 - \xi_m \tag{124}$$

使用预先设置的参数 C 来控制支持向量机对误差 ξ_m 的敏感程度, 优化问题 (96) ~ (97) 变为

$$\min_{\boldsymbol{\omega}, b, \boldsymbol{\xi}} \quad \frac{1}{2}||\boldsymbol{\omega}||^2 + C\sum_{m=1}^{M} \xi_m \tag{125}$$

$$s.t. \quad y_m(\boldsymbol{\omega}^T \boldsymbol{x_m} + b) \geq 1 - \xi_m, \quad m = 1, \ldots, M \tag{126}$$

$$\xi_m \geq 0, \quad m = 1, \ldots, M \tag{127}$$

拉格朗日函数为

$$\mathcal{L}(\boldsymbol{\omega}, b, \boldsymbol{\xi}, \boldsymbol{\alpha}, \boldsymbol{r}) = \frac{1}{2}\boldsymbol{\omega}^T\boldsymbol{\omega} + C\sum_{m=1}^{M} \xi_m - \sum_{m=1}^{M} \alpha_m(y_m(\boldsymbol{\omega}^T\boldsymbol{x} + b) - 1 + \xi_m) - \sum_{m=1}^{M} r_m\xi_m \tag{128}$$

优化问题 (125) ~ (127) 的对偶问题为

$$\min_{\alpha} \quad \frac{1}{2}\sum_{i=1}^{M}\sum_{j=1}^{M} \alpha_i\alpha_j y_i y_j \boldsymbol{x_i}^T\boldsymbol{x_j} - \sum_{m=1}^{M} \alpha_m \tag{129}$$

$$s.t. \quad \sum_{m=1}^{M} \alpha_m y_m = 0 \tag{130}$$

$$0 \leq \alpha_m \leq C, \quad m = 1, \ldots, M \tag{131}$$

求得最优解 $\boldsymbol{\alpha}^* = (\alpha_1^*, \ldots, \alpha_M^*)^T$. 有

$$\boldsymbol{\omega}^* = \sum_{m=1}^{M} \alpha_m y_m \boldsymbol{x_m} \tag{132}$$

利用 $\alpha_j > 0$ 的支持向量 j 计算

$$b^* = y_j - \sum_{m=1}^{M} \alpha_m^* y_m \boldsymbol{x_m}^T\boldsymbol{x_j} \tag{133}$$

从而可以构建决策函数

$$h(\boldsymbol{x}) = \text{sign}(\sum_{m=1}^{M} \alpha_m y_m \boldsymbol{x_m}^T\boldsymbol{x} + b^*) \tag{134}$$

同样的, 可以将非线性的核技巧应用在软间隔的支持向量机上.

8.5 Sequential minimal optimization

9 K-means clustering

K 均值是一种无监督学习的聚类算法. 假设我们有一组数据 $D = \{\boldsymbol{x_1}, \ldots, \boldsymbol{x_M}\}$, 我们的目标是将数据中划分成 K 个不相交的子集 $\mathcal{C} = \{C_1, \ldots, C_k\}$. 每个团簇 c_k 都有一个中心 $\boldsymbol{\mu_k}$. 我们使用一个损失函数来衡量团簇的紧密程度

$$E(\mathcal{C}, \boldsymbol{\mu}) = \sum_{k=1}^{k} \sum_{m=1}^{M} 1\{\boldsymbol{x_m} \in C_k\} ||\boldsymbol{x_m} - \boldsymbol{\mu_k}||^2 \tag{135}$$

我们希望找到一个团簇的划分 $\mathcal{C}$ 使得损失函数最小

1. 首先随机初始化点 $\boldsymbol{x_m}$ 所属的类别,$m = 1, \ldots, M$
2. 重复一下步骤, 直至 $\mathcal{C}$ 的状态不再发生改变
   (a) 计算类别 C_k 的中心 $\boldsymbol{\mu_k} = \frac{\sum_{m=1}^{M} 1\{\boldsymbol{x_m} \in C_k\} \boldsymbol{x_m}}{\sum_{m=1}^{M} 1\{\boldsymbol{x_m} \in C_k\}}, k = 1, \ldots, K$
   (b) 将 x_m 划分到 j 类, 满足 $j = \arg\min_k ||\boldsymbol{x_m} - \boldsymbol{\mu_k}||$, $m = 1, \ldots, M$.

由于损失函数在迭代过程中单调递减, 而代价函数 E 常常是是非凸的, 因此算法必将经过有限次迭代后收敛到一个局部极小值点. 可以通过多次不同的初始化, 寻找更好的聚类结果. 算法也可以改为在开始的时候随机初始化各类别的中心 $\boldsymbol{\mu_k}$, 只需对后续步骤稍作修改.

此外, 由于数据点 $\boldsymbol{x}$ 的各个分量的量纲可能不同, 因而不同分量的数值具有不同的分布尺度. 我们需要对数据进行标准化处理.

10 Mixtures of Gaussians

在无监督学习问题中, 假设观测数据中的点 $\boldsymbol{x}$ 是由 k 个不同的高斯分布 $z_i (i = 1, \ldots, k)$ 混合的模型所生成的.

$$p(\boldsymbol{x}; \boldsymbol{\mu}, \boldsymbol{\Sigma}, \boldsymbol{\phi}) = \sum_{i=1}^{k} p(\boldsymbol{x}|z_i; \boldsymbol{\mu_i}, \boldsymbol{\Sigma_i}) p(z_i; \boldsymbol{\phi}), \tag{136}$$

其中

$$p(z_i; \boldsymbol{\phi}) = \phi_i, \tag{137}$$

$$p(\boldsymbol{x}|z_i) = \mathcal{N}(\boldsymbol{x}; \boldsymbol{\mu_i}, \boldsymbol{\Sigma_i}). \tag{138}$$

在这里,$\sum_{i=1}^{k} \phi_i = 1$.

如果我们知道模型的参数 $\boldsymbol{\mu}, \boldsymbol{\Sigma}, \boldsymbol{\phi}$, 那么对于一个点 $\boldsymbol{x_m}$, 我们可以求得它由高斯 z_j 产生的后验概率 γ_{mj}

$$\begin{aligned}\gamma_{mj} &= p(z_j|\boldsymbol{x_m};\boldsymbol{\mu},\boldsymbol{\Sigma},\boldsymbol{\phi}) \\ &= \frac{p(\boldsymbol{x_m}, z_j;\boldsymbol{\mu},\boldsymbol{\Sigma},\boldsymbol{\phi})}{p(\boldsymbol{x_m};\boldsymbol{\mu},\boldsymbol{\Sigma},\boldsymbol{\phi})} \\ &= \frac{p(\boldsymbol{x_m}|z_j;\boldsymbol{\mu},\boldsymbol{\Sigma})p(z_j;\boldsymbol{\phi})}{\sum_{i=1}^{k} p(\boldsymbol{x_m}|z_i;\boldsymbol{\mu},\boldsymbol{\Sigma})p(z_i;\boldsymbol{\phi})} \\ &= \frac{\phi_j \mathcal{N}(\boldsymbol{x_m};\boldsymbol{\mu_j},\boldsymbol{\Sigma_j})}{\sum_{i=1}^{k} \phi_i \mathcal{N}(\boldsymbol{x_m};\boldsymbol{\mu_i},\boldsymbol{\Sigma_i})}\end{aligned} \tag{139}$$

对于已知的训练数据 $D = \{\boldsymbol{x_1}, \ldots, \boldsymbol{x_M}\}$, 高斯混合模型的似然函数为

$$\begin{aligned}\mathcal{L}(\boldsymbol{\mu},\boldsymbol{\Sigma},\boldsymbol{\phi}) &= P(D;\boldsymbol{\mu},\boldsymbol{\Sigma},\boldsymbol{\phi}) \\ &= \prod_{m=1}^{M} p(\boldsymbol{x_m};\boldsymbol{\mu},\boldsymbol{\Sigma},\boldsymbol{\phi}) \\ &= \prod_{m=1}^{M} \sum_{i=1}^{k} \phi_i \mathcal{N}(\boldsymbol{x_m};\boldsymbol{\mu_i},\boldsymbol{\Sigma_i})\end{aligned} \tag{140}$$

取对数, 得到对数似然函数

$$\log \mathcal{L}(\boldsymbol{\mu},\boldsymbol{\Sigma},\boldsymbol{\phi}) = \sum_{m=1}^{M} \log \sum_{i=1}^{k} \phi_i \mathcal{N}(\boldsymbol{x_m};\boldsymbol{\mu_i},\boldsymbol{\Sigma_i}) \tag{141}$$

在这里, 我们无法通过令对数似然函数导数为 0 的方式来求参数极大似然估计的解析解. 现在我们考虑使用期望最大化算法 (expectation-maximization), 或者 EM 算法.

首先求 $\log \mathcal{L}$ 关于第 j 类均值 $\boldsymbol{\mu_j}$ 的梯度, 先求关于 $\boldsymbol{\mu_j}$ 第 l 个分量的偏导数

$$\begin{aligned}\frac{\partial \log \mathcal{L}}{\partial \mu_j^{(l)}} &= \frac{\partial \sum_{m=1}^{M} \log \sum_{i=1}^{k} \phi_i \mathcal{N}(\boldsymbol{x_m};\boldsymbol{\mu_k},\boldsymbol{\Sigma_k})}{\partial \mu_j^{(l)}} \\ &= \sum_{m=1}^{M} \frac{1}{\sum_{i=1}^{k} \phi_i \mathcal{N}(\boldsymbol{x_m};\boldsymbol{\mu_i},\boldsymbol{\Sigma_i})} \frac{\partial \sum_{i=1}^{k} \phi_i \mathcal{N}(\boldsymbol{x_m};\boldsymbol{\mu_i},\boldsymbol{\Sigma_i})}{\partial \mu_j^{(i)}} \\ &= \sum_{m=1}^{M} \frac{\phi_j}{\sum_{i=1}^{k} \phi_i \mathcal{N}(\boldsymbol{x_m};\boldsymbol{\mu_i},\boldsymbol{\Sigma_i})} \frac{\partial \mathcal{N}(\boldsymbol{x_m};\boldsymbol{\mu_j},\boldsymbol{\Sigma_j})}{\partial \mu_j^{(i)}} \\ &= \sum_{m=1}^{M} \frac{\phi_j}{\sum_{i=1}^{k} \phi_i \mathcal{N}(\boldsymbol{x_m};\boldsymbol{\mu_i},\boldsymbol{\Sigma_i})} \frac{1}{(2\pi)^{D/2}|\boldsymbol{\Sigma_j}|^{1/2}} \frac{\partial \exp(-1/2(\boldsymbol{x_m}-\boldsymbol{\mu_j})^T \boldsymbol{\Sigma}^{-1}(\boldsymbol{x_m}-\boldsymbol{\mu_j}))}{\partial \mu_j^{(i)}} \\ &= \sum_{m=1}^{M} \frac{\phi_j \exp(-1/2(\boldsymbol{x_m}-\boldsymbol{\mu_j})^T \boldsymbol{\Sigma}^{-1}(\boldsymbol{x_m}-\boldsymbol{\mu_j}))}{\sum_{i=1}^{k} \phi_i \mathcal{N}(\boldsymbol{x_m};\boldsymbol{\mu_i},\boldsymbol{\Sigma_i})} \frac{1}{(2\pi)^{D/2}|\boldsymbol{\Sigma_j}|^{1/2}} \frac{\partial -1/2(\boldsymbol{x_m}-\boldsymbol{\mu_j})\boldsymbol{\Sigma_j}^{-1}(\boldsymbol{x_m}-\boldsymbol{\mu_j})}{\partial \mu_j^{(i)}} \\ &= \sum_{m=1}^{M} \frac{\phi_j \mathcal{N}(\boldsymbol{x_m};\boldsymbol{\mu_j},\boldsymbol{\Sigma_j})}{\sum_{i=1}^{k} \phi_i \mathcal{N}(\boldsymbol{x_m};\boldsymbol{\mu_i},\boldsymbol{\Sigma_i})} \frac{\partial -1/2 \sum_{r=1}^{D} \sum_{c=1}^{D} (x_m^{(r)} - \mu_j^{(r)})(\boldsymbol{\Sigma_j}^{-1})_{rc}(x_m^{(c)} - \mu_j^{(c)})}{\partial \mu_j^{(l)}} \\ &= \sum_{m=1}^{M} \frac{\phi_j \mathcal{N}(\boldsymbol{x_m};\boldsymbol{\mu_j},\boldsymbol{\Sigma_j})}{\sum_{i=1}^{k} \phi_i \mathcal{N}(\boldsymbol{x_m};\boldsymbol{\mu_i},\boldsymbol{\Sigma_i})} (\boldsymbol{\Sigma_j}^{-1}(\boldsymbol{x_m}-\boldsymbol{\mu_k}))^{(j)}\end{aligned} \tag{142}$$

从而 $\log \mathcal{L}$ 关于 $\boldsymbol{\mu_j}$ 的梯度为

$$\nabla_{\boldsymbol{\mu_j}} \log \mathcal{L} = \sum_{m=1}^{M} \frac{\phi_j \mathcal{N}(\boldsymbol{x_m}; \boldsymbol{\mu_j}, \boldsymbol{\Sigma_j})}{\sum_{i=1}^{k} \phi_i \mathcal{N}(\boldsymbol{x_m}; \boldsymbol{\mu_i}, \boldsymbol{\Sigma_i})} \boldsymbol{\Sigma_j^{-1}} (\boldsymbol{x_m} - \boldsymbol{\mu_k}) \tag{143}$$

令其为 0, 得到

$$\boldsymbol{\mu_j} = \frac{1}{M_j} \sum_{m=1}^{M} \gamma_{mj} \boldsymbol{x_m} \tag{144}$$

其中

$$M_j = \sum_{m=1}^{M} \gamma_{mj} \tag{145}$$

同样的, 令 $\log \mathcal{L}$ 关于 $\boldsymbol{\Sigma_j}$ 的梯度为 0, 我们可以得到

$$\boldsymbol{\Sigma_j} = \frac{1}{M_j} \sum_{m=1}^{M} \gamma_{mj} (\boldsymbol{x_m} - \boldsymbol{\mu_j})(\boldsymbol{x_m} - \boldsymbol{\mu_j})^T \tag{146}$$

最后, 我们关于 ϕ_j 来最大化 $\log \mathcal{L}$. 考虑限制条件 $\sum_{i=1}^{k} \phi_i = 1$, 我们最大化拉格朗日函数

$$\log \mathcal{L} + \lambda (\sum_{i=1}^{k} \alpha_i - 1) \tag{147}$$

它关于 ϕ_j 的偏导数为 0

$$\sum_{m=1}^{M} \frac{\mathcal{N}(\boldsymbol{x_m}, \boldsymbol{\mu_j}, \boldsymbol{\Sigma_j})}{\sum_{i=1}^{k} \phi_i \mathcal{N}(\boldsymbol{x_m}; \boldsymbol{\mu_i}, \boldsymbol{\Sigma_i})} + \lambda = 0 \tag{148}$$

等号两边同时乘以 ϕ_j 得到

$$\sum_{m=1}^{M} \gamma_{mj} + \phi_j \lambda = 0 \tag{149}$$

将等号两端关于所有 j 求和, 并注意到 $\sum_{i=1}^{k} \phi_i = 1, \sum_{i=1}^{k} \gamma_{mi} = 1$, 于是有

$$M + \lambda = 0 \tag{150}$$

代入到式 (149) 得到

$$\phi_j = \frac{1}{M} \sum_{m=1}^{M} \gamma_{mj} \tag{151}$$

通过以上步骤, 我们获得了 $\boldsymbol{\mu_j}, \boldsymbol{\Sigma_j}, \phi_j$ 的表达式 (143),(146) 和 (151). 但事实上, 我们它们的值都依赖于各个数据点 $\boldsymbol{x_m}$ 由高斯 z_j 产生的后验概率 γ_{mj}. 而由式 (139) 我们知道 γ_{mj} 是依赖于 $\boldsymbol{\mu_j}, \boldsymbol{\Sigma_j}, \phi_j$ 的. 因此不能解析地求出 $\boldsymbol{\mu_j}, \boldsymbol{\Sigma_j}, \phi_j$ 的极大似然估计值.

这里可以通过 EM 算法来迭代求解. 首先

1. 按为各类的 $\boldsymbol{\mu_j}, \boldsymbol{\Sigma}, \phi_j$ 设置一个初始值.

2. 重复一下步骤, 直至对数似然函数收敛或者迭代次数达到上限

(a) E 步骤 (expectation step), 计算在当前参数下的各数据点 $\boldsymbol{x_m}$ 由各高斯类 z_j 生成的后验概率 γ_{mj}

$$\gamma_{mj} := \frac{\phi_j \mathcal{N}(\boldsymbol{x_m}; \boldsymbol{\mu_j}, \boldsymbol{\Sigma_j})}{\sum_{i=1}^{k} \phi_i \mathcal{N}(\boldsymbol{x_m}; \boldsymbol{\mu_i}, \boldsymbol{\Sigma_j})} \tag{152}$$

(b) M 步骤 (maximization step) 中, 利用 γ_{mj} 重新估计 $\boldsymbol{\mu_j}, \boldsymbol{\Sigma_j}, \phi_j$

$$\boldsymbol{\mu_j} := \frac{1}{M_j} \sum_{m=1}^{M} \gamma_{mj} \boldsymbol{x_m} \tag{153}$$

$$\boldsymbol{\Sigma_j} := \frac{1}{M_j} \sum_{m=1}^{M} \gamma_{mj} (\boldsymbol{x_m} - \boldsymbol{\mu_j})(\boldsymbol{x_m} - \boldsymbol{\mu_j})^T \tag{154}$$

$$\phi_j := \frac{M_j}{M} \tag{155}$$

其中

$$M_j := \sum_{m=1}^{M} \gamma_{mj} \tag{156}$$

(c) 计算对数似然函数

$$\log \mathcal{L}(\boldsymbol{\mu}, \boldsymbol{\Sigma}, \boldsymbol{\phi}) = \sum_{m=1}^{M} \log \sum_{i=1}^{k} \phi_i \mathcal{N}(\boldsymbol{x_m}; \boldsymbol{\mu_i}, \boldsymbol{\Sigma_j}) \tag{157}$$

在初始化步骤中, 可以先使用 K 均值算法对数据进行聚类, 然后计算每类的均值和协方差矩阵来设置 $\boldsymbol{\mu_j}, \boldsymbol{\Sigma_j}$, 用各类别中数据点占比例来设置 ϕ_j.

最大似然的方法可能会带来奇异性, 即某个高斯退化到一个具体的数据点上. 这种情况需要避免. 另外, 对数似然函数通常是非凸的, 并且在 EM 算法的迭代过程中, 对数似然函数是单调递增的, 收敛到某个局部极大值点.EM 算法不能保证找到全局极大值点.

11 Factor analysis

对于一组产生于混合高斯分布的数据 $\{\boldsymbol{x_m} : m = 1, \ldots, M\}, \boldsymbol{x_m} \in \mathbb{R}^n$, 我们可以使用 EM 算法来拟合这个混合的模型. 如果数据的维数 n 大于样本数量 M, 我们便不能用来估计 n 维的高斯, 即使仅仅估计一个高斯.

虽然我们可以通过约束高斯分布的协方差矩阵为对角矩阵从而减小自由度, 实现估计高斯分布, 但是这样的模型无法估计变量之间的相关性.

11.1 Marginals and conditions of Gaussians

在引入因子分析前, 我们先介绍联合多变量高斯分布的条件分布和边缘分布. 假设我们有随机变量

$$\boldsymbol{x} = \begin{pmatrix} \boldsymbol{x_1} \\ \boldsymbol{x_2} \end{pmatrix} \tag{158}$$

其中 $\boldsymbol{x_1} \in \mathbb{R}^r, \boldsymbol{x_2} \in \mathbb{R}^s$, 那么有 $\boldsymbol{x} \in \mathbb{R}^{r+s}$. 并且假设 $\boldsymbol{x} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$

$$\boldsymbol{\mu} = \begin{pmatrix} \boldsymbol{\mu_1} \\ \boldsymbol{\mu_2} \end{pmatrix}, \quad \boldsymbol{\Sigma} = \begin{pmatrix} \boldsymbol{\Sigma_{11}} & \boldsymbol{\Sigma_{12}} \\ \boldsymbol{\Sigma_{21}} & \boldsymbol{\Sigma_{22}} \end{pmatrix} \tag{159}$$

注意到, 协方差矩阵是对称矩阵, 我们有 $\boldsymbol{\Sigma_{21}^T = \Sigma_{12}}$.

在 $\boldsymbol{x_2}$ 的条件下,$\boldsymbol{x_1}$ 的条件分布为 $\boldsymbol{x_1} \sim \mathcal{N}(\boldsymbol{\mu_{1|2}}, \boldsymbol{\Sigma_{1|2}})$, 其中

$$\boldsymbol{\mu_{1|2} = \mu_1 + \Sigma_{12}\Sigma_{22}^{-1}(x_2 - \mu_2)} \tag{160}$$

$$\boldsymbol{\Sigma_{1|2} = \Sigma_{11} - \Sigma_{12}\Sigma_{22}^{-1}\Sigma_{21}} \tag{161}$$

11.2 The Factor analysis model

12 Principal components analysis

主成分分析是一种常用的数据降维方法. 对于一组数据 $D = \{\boldsymbol{x_1}, \ldots, \boldsymbol{x_M}\}$, 为了方便, 我们一般需要对数据进行中心化处理

$$\boldsymbol{\mu} = \frac{1}{M}\sum_{m=1}^{M}\boldsymbol{x_m} \tag{162}$$

$$\boldsymbol{\tilde{x}_m = x_m - \mu} \tag{163}$$

如果数据中各个维度的尺度差异较大, 我们还需要进行方差标准化处理.

假设数据是 $\mathbb{R}^N$ 中的点, 我们考虑将它们投影到一维子空间上

$$y = \boldsymbol{w}^T\tilde{\boldsymbol{x}} \tag{164}$$

并且希望数据点的投影 $y_1, \ldots, y_M$ 尽可能分散. 我们可以用 y 的方差来衡量投影后的分散程度

$$\begin{aligned}\frac{1}{M}\sum_{m=1}^{M}y_m^2 &= \frac{1}{M}\sum_{m=1}^{M}\boldsymbol{w}^T\boldsymbol{x_m}\boldsymbol{x_m}^T\boldsymbol{w} \\ &= \frac{1}{M}\boldsymbol{w}^T\boldsymbol{X}^T\boldsymbol{X}\boldsymbol{w}\end{aligned} \tag{165}$$

我们可以通过最大化 y 的方差来确定投影的子空间 $\boldsymbol{w}$. 由于我们只关心 $\boldsymbol{w}$ 的方向, 而长度 $||\boldsymbol{w}||$ 是无关的, 我们需要添加约束条件 $\boldsymbol{w}^T\boldsymbol{w} = 1$ 来防止 $||\boldsymbol{w}||$ 趋于无穷大. 于是得到优化问题

$$\boldsymbol{w} = \arg\max_{\boldsymbol{w}}\boldsymbol{w}^T\boldsymbol{X}^T\boldsymbol{X}\boldsymbol{w} \tag{166}$$

$$s.t. \quad \boldsymbol{w}^T\boldsymbol{w} = 1 \tag{167}$$

构造拉格朗日函数

$$\mathcal{L}(\boldsymbol{w}, \alpha) = \boldsymbol{w}^T\boldsymbol{X}^T\boldsymbol{X}\boldsymbol{w} + \alpha(\boldsymbol{w}^T\boldsymbol{w} - 1) \tag{168}$$

令它关于 $\boldsymbol{w}$ 的梯度为 0

$$\nabla_{\boldsymbol{w}}\mathcal{L}(\boldsymbol{w}, \alpha) = 2\boldsymbol{X}^T\boldsymbol{X}\boldsymbol{w} - 2\alpha\boldsymbol{w} = 0 \tag{169}$$

$$\boldsymbol{X}^T\boldsymbol{X}\boldsymbol{w} = \alpha\boldsymbol{w} \tag{170}$$

因此 $\boldsymbol{w}$ 是矩阵 $\boldsymbol{X}^T\boldsymbol{X}$ 的特征值 α 的特征向量. 将该式左乘 $\boldsymbol{w}$, 得到

$$\begin{aligned}\boldsymbol{w}^T\boldsymbol{X}^T\boldsymbol{X}\boldsymbol{w} &= \alpha\boldsymbol{w}^T\boldsymbol{w} \\ &= \alpha\end{aligned} \tag{171}$$

为了最大化 $\boldsymbol{w}^T\boldsymbol{X}^T\boldsymbol{X}\boldsymbol{w}$, 我们取 $\boldsymbol{w}$ 为 $\boldsymbol{X}^T\boldsymbol{X}$ 最大的特征值的特征向量.

现在, 我们考虑把数据点投影到 d 维的子空间中, 其中 $d < N$. 我们需要选取一组基 $\boldsymbol{w_1}, \ldots, \boldsymbol{w_d}$, 满足 $\boldsymbol{w_i}^T\boldsymbol{w} = 1$ 并且 $\boldsymbol{w_i}^T\boldsymbol{w_j} = 0, i \neq j$. 记

$$\boldsymbol{W} = \begin{pmatrix} \boldsymbol{w_1}^T \\ \vdots \\ \boldsymbol{w_d}^T \end{pmatrix}, \quad \boldsymbol{X} = \begin{pmatrix} \boldsymbol{x_1}^T \\ \vdots \\ \boldsymbol{x_M}^T \end{pmatrix} \tag{172}$$

数据的投影可以表示为

$$\boldsymbol{Y} = \begin{pmatrix} \boldsymbol{y_1}^T \\ \vdots \\ \boldsymbol{y_M}^T \end{pmatrix} = \boldsymbol{W}\boldsymbol{X}^T \tag{173}$$

为了最大化投影后样本点的方差, 在约束条件下的优化问题为

$$\boldsymbol{W} = \arg\max_{\boldsymbol{W}} \operatorname{tr}(\boldsymbol{W}\boldsymbol{X}^T\boldsymbol{X}\boldsymbol{W}^T) \tag{174}$$

$$s.t. \quad \boldsymbol{W}^T\boldsymbol{W} = \boldsymbol{I} \tag{175}$$

构造拉格朗日函数

$$\mathcal{L}(\boldsymbol{W}, \alpha) = \boldsymbol{W}\boldsymbol{X}^T\boldsymbol{X}\boldsymbol{W}^T + \alpha(\boldsymbol{W}^T\boldsymbol{W} - \boldsymbol{I}) \tag{176}$$

求得

$$\boldsymbol{X}^T\boldsymbol{X}\boldsymbol{W}^T = \alpha\boldsymbol{W}^T \tag{177}$$

对矩阵 $\boldsymbol{X}^T\boldsymbol{X}$ 进行特征值分解, 得到特征值 $\alpha_1 \geq \cdots \geq \alpha_N$ 取最大的 d 个特征值对应的单位特征向量 $\boldsymbol{w_1}, \ldots, \boldsymbol{w_d}$ 作为投影变换的基. 变换的矩阵为 $\boldsymbol{W} = (\boldsymbol{w_1}, \ldots, \boldsymbol{w_d})^T$. 最后根据式 (173), 我们可以得到主成分分析的结果 $\boldsymbol{Y}$.

13 Independent Components analysis

参考文献

[1] Christopher Michael Bishop. *Pattern Recognition And Machine Learning Bishop*. 2006.

[2] David Ruppert, Trevor Hastie, Robert Tibshirani, Jerome Friedman New, and York Springer. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. 2003.

[3] 周志华. 机器学习. 2016.

[4] 李航. 统计学习方法. 2012.