

1 统计学习问题分类

监督学习：回归、分类；无监督学习：聚类、变换（降维/投影、嵌入）

2 一元统计分析

极大似然 似然函数： $L(\theta) \prod_i p(x^{(i)}|\theta)$ ，最大似然函数使得参数的极大似然估计 $\hat{\theta} \in ML$ ，常用方法是对似然函数取负对数，再寻找负对数的最小值

贝叶斯法

$$P(\Theta|X^{(1)}, \dots, X^{(n)}) = \frac{P(X^{(1)}, \dots, X^{(n)}|\Theta)P(\Theta)}{P(X^{(1)}, \dots, X^{(n)})}$$

先验和后验属于同一类分布的情况在贝叶斯方法中称为共轭先验，是贝叶斯方法中避免先验的一种常见做法

评价准则 统计量：样本的统计数

一致性：如果随着样本数量的增大，估计量依概率收敛于真实值，即 $\forall \epsilon > 0, \lim_{n \rightarrow \infty} P(|\hat{\theta}_n - \theta| < \epsilon) = 1$ 则称估计量是（弱）一致的。

无偏性：如果估计量的期望等于真实值，即对任意 θ 有 $E[\hat{\theta}] = \theta$ ，则称估计量是无偏的，否则就是有偏的。将 $E[\hat{\theta}] - \theta$ 称作估计量的偏差 (bias)。

有效性：估计量的方差应尽可能小。如果有两个无偏估计量 $\hat{\theta}_1$ 和 $\hat{\theta}_2$ ，且对任意 θ 有 $V[\hat{\theta}_1] < V[\hat{\theta}_2]$ ，则称 $\hat{\theta}_1$ 比 $\hat{\theta}_2$ 更有效。

充分统计量：设总体 X 的概率函数带有未知参数 θ ，统计量 $\hat{\theta} = f(X^{(1)}, \dots, X^{(n)})$ ，其中 f 是预定义函数，如果条件概率函数 $\mathbb{P}(X^{(1)}, \dots, X^{(n)}|\hat{\theta})$ 与 θ 无关，则称 $\hat{\theta}$ 是 θ 的充分统计量。

因子分解定理 若样本的概率函数能够分解为 $\mathbb{P}(X^{(1)}, \dots, X^{(n)}) = g(x^{(1)}, \dots, x^{(n)})h(f(x^{(1)}, \dots, x^{(n)}); \theta)$ ，则 $\hat{\theta} = f(X^{(1)}, \dots, X^{(n)})$ 是 θ 的充分统计量

指数族分布的充分统计量 若总体服从指数族分布 $P(X) = g(x)h(\theta) \exp(\theta^T \Phi(x))$ ，其中 $\Phi(x)$ 称为特征基函数，则 $\sum_{i=1}^n \Phi(x^{(i)})$ 是 θ 的充分统计量。高斯分布 $\phi(x) = (x^2, x)^T$

最小均方差估计与风险分解 $E[(\hat{\theta} - \theta)^2] = (E[\hat{\theta} - \theta])^2 + V[\hat{\theta} - \theta] = (E[\hat{\theta}] - \theta)^2 + V[\hat{\theta}]$ 第一项是估计量的偏差的平方，第二项是估计量的方差。偏差和方差分别反映了估计量的系统误差和随机误差，均方误差最小化同时考虑了系统误差和随机误差。如果将估计量限定为无偏的，则最小均方误差估计量就是一致最小方差无偏估计 (UMVUE)。如果允许估计量有偏，则最小均方误差估计可以不同于 UMVUE。

非参数统计分析 经验分布函数 (EDF): $\hat{F}(x) = \frac{1}{n} \sum_{i=1}^n I(x^{(i)} \leq x)$

平滑: $\tilde{F}(x) \triangleq \int_t \hat{F}(t)k(x-t)dt$, k 称为平滑核函数或简称核函数，核密度函数估计: $\hat{f}(x) = \frac{1}{n} \sum_{i=1}^n k(x - x^{(i)})$

k 近邻 (k-NN) 法: 在 x 数据上的每个点 x ，调书区间可度量 $h(x)$ 使得区间 $[x-h(x), x+h(x)]$ 中包含有 k 个数据，则可用 $\hat{f}(x) = \frac{1}{2nh(x)}$ 来估计密度

3 线性回归

最小二乘法

$$\min_{w,b} \mathcal{E}(w,b) = \min_{w,b} \sum_{i=1}^n (y^{(i)} - (wx^{(i)} + b))^2 \quad (2)$$

$$w_{LS} = \frac{\sum_{i=1}^n (x^{(i)} - \bar{x})(y^{(i)} - \bar{y})}{\sum_{i=1}^n (x^{(i)} - \bar{x})^2}, b_{LS} = \bar{y} - w_{LS} \bar{x} \quad (3)$$

正则化 带约束优化问题：

$$\min_w \sum (y^{(i)} - wx^{(i)})^2, s.t. w^2 \leq c \quad (4)$$

$$L(w, \lambda) = \sum (y^{(i)} - wx^{(i)})^2 + \lambda(w^2 - c) \quad (5)$$

$$w_{reg} = \frac{\sum (x^{(i)})^2 + \lambda}{\sum (x^{(i)})^2 + \lambda} \quad (6)$$

正则化是在求最小二乘解的约束限制参数 w 的取值范围，正则化权重越大，取值范围限定得越小。

偏差-方差均衡： $\sum (x^{(i)})^2$

$$E[\hat{w}] = \frac{\sum (x^{(i)})^2 + w}{\sum (x^{(i)})^2 + \epsilon} \quad (7)$$

ϵ 越大， $E[\hat{w}]$ 偏差 w 越多，偏差平方越大； $V[\hat{w}]$ 越小，方差越小。综合考虑偏差和方差，则可以找到一个合适的 ϵ ，使得两项之和达到最小。

贝叶斯: $W \sim N(0, \sigma_w^2), P(Y^{(i)}|W) \sim N(wx^{(i)}(i), \sigma^2)$ ，则 $\epsilon = \sigma^2/\sigma_w^2$ ，正则化项实质上对应于后验分布中由先验分布引入的项

最小二乘回归

$$X = \begin{bmatrix} x^{(1)} \\ \vdots \\ x^{(n)} \end{bmatrix} \quad (8)$$

$$w_{LS} = \arg \min_w (y - Xw)^T (y - Xw) \quad (9)$$

$$X^T y = X^T X w \quad (10)$$

$$\Phi \triangleq (f_1, \dots, f_p)^T, y = w^T \Phi(x) \quad (11)$$

$$\Phi^T \Phi w_{LS} = \Phi^T y \quad (12)$$

$$\min_w (y - Xw)^T (y - Xw) + \lambda w^T w \quad (13)$$

$$w_{ridge} = (X^T X + \lambda I)^{-1} X^T y \quad (14)$$

贝叶斯线性回归 假设 $P(W) = N(m_0, S_0)$ ，样本条件分布如下，可得

$$P(Y^{(1)}, \dots, Y^{(n)}|W) = \prod N(w^T x^{(i)}, \sigma^2) \quad (15)$$

$$P(W|Y^{(1)}, \dots, Y^{(n)}) = N(m_n, S_n) \quad (16)$$

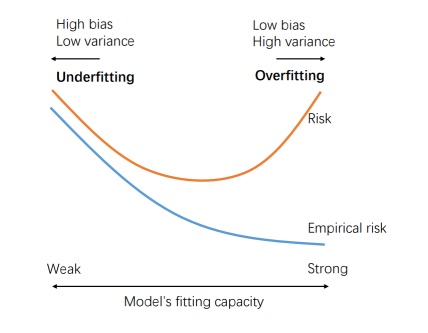
$$m_n = S_n(S_0^{-1}m_0 + \frac{1}{\sigma^2}X^T y) \quad (17)$$

$$S_n = (S_0^{-1} + \frac{1}{\sigma^2}X^T X) \quad (18)$$

序贯学习：

$$P(Y^{(1)}, \dots, Y^{(n+1)}|W) = \frac{P(Y^{(1)}, \dots, Y^{(n)}|W)P(Y^{(n+1)}|W)}{P(Y^{(n+1)}|Y^{(1)}, \dots, Y^{(n)})} \quad (19)$$

模型评价与选择 一般来说，如果模型中可学习参数太少，模型的拟合能力很弱，在训练数据上的误差风险很大，风险也很大，这种现象称为欠拟合。经验风险降低但风险反而升高的现象在统计学习中称为过拟合



赤池信息量准则 $AIC = 2NLL + 2p$ ，其中， NLL 是训练数据上估计的负对数似然，是参数的个数。AIC 值越小，模型越好。当 p 一定时，负对数似然越小的回归函数越好，也就是极大似然估计是最好的，代入得 $AIC = n \ln(\mathcal{E}(w_{LS})) + 2p$

贝叶斯模型评价 对参数（随机变量）求期望，称为模型证据

$$LASSO \min_w (y - Xw)^T (y - Xw) + \lambda \|w\|_1 \quad (20)$$

lasso 回归虽然具有前述选择等优点，但它一般情况下没有闭式解

$$\hat{f}(x) \triangleq \sum_{i=1}^n y^{(i)} k(x, x^{(i)}) \quad (21)$$

Mercer 条件: 令矩阵 K ，其中 $K_{ij} = k(x^{(i)}, x^{(j)})$ ，要求对任意 x ， K 半正定。有 $k(x, y) = k(y, x)$ ， $k(x, y) = (\Phi(x))^T \Phi(y)$

$$\hat{f}(x) \triangleq \sum_{i=1}^n y^{(i)} \Phi(x^{(i)})^T \Phi(x) = y^T \Phi \Phi(x) = w^T \Phi(x) \quad (22)$$

K 近邻回归

$$\hat{f}(x) \triangleq \frac{1}{k} \sum_{j=1}^k y^{(n_j)}, s.t. x^{(n_j)} \in N_k(x) \quad (23)$$

4 线性分类

$$y = \text{sign}(w_{LS} x + b_{LS}) \quad (24)$$

有 train-test mismatch 问题

zero-one loss/0-1 loss 考虑到 y 函数，平方损失函数等价于

$$L(w, b, x, y) \triangleq I(y \neq \text{sign}(w^T x + b)) \quad (25)$$

Fisher 投影 (LDA) 对于 +1 类， v_+ 为均值， $m_+ = w^T v_+$ 为投影后均值， S_+ 为协方差矩阵， $S_+ = w^T S_+ w$ 为投影后方差。-1 类同理。类内方差 $S_w = S_+ + S_-$ 类间方差 $(m_+ - m_-)^2$ ，求：

$$\min_w w^T (S_+ + S_-) w, s.t. w^T (v_+ - v_-)(v_+ - v_-)^T w > c \quad (26)$$

$$w \propto (S_+ + S_-)^{-1} (v_+ - v_-) \quad (27)$$

感知机 Perceptron 使用代理损失函数：

$$L(w, b, x, y) \triangleq \max(0, -y(w^T x + b)) \quad (28)$$

使用随机梯度下降 SGD

$$i f y_i (w^T x_i + b) < 0 : w = w + \eta y_i x_i, b = b + \eta y_i \quad (29)$$

$$\text{否则: } w = w + \alpha_i y_i x_i, b = \sum \alpha_i y_i, \alpha_i \geq 0, \text{ 直接 } \alpha_i \geq 0, \text{ 直接 } \alpha_i \geq 0 \quad (30)$$

$$i f y_i (\sum \alpha_j y_j x_j^T x_i + \sum \alpha_j y_j) < 0 : \alpha_i = \alpha_i + \eta \quad (31)$$

$$\text{最终分类器为 } \text{sign}(\sum \alpha_i y_i (x_i^T x + 1)) \quad (32)$$

带基函数的感知机

$$i f y_i ((\sum \alpha_j y_j \Phi(x_j))^T \Phi(x_i)) < 0 : \alpha_i = \alpha_i + \eta \quad (33)$$

$$\text{最终分类器为 } \text{sign}((\sum \alpha_i y_i \Phi(x_i))^T \Phi(x)) \quad (34)$$

带核函数的感知机

$$i f \sum \alpha_j y_j y_j k(x_i, x_j) < 0 : \alpha_i = \alpha_i + \eta \quad (35)$$

$$\text{最终分类器为 } \text{sign}(\sum \alpha_i y_i k(x_i, x)) \quad (36)$$

交叉熵损失

$$CE(B(p), B(q)) = -p \ln q - (1-p) \ln(1-q) \quad (37)$$

逻辑回归 样本 $(x_i, y_i), y_i \sim B(q_i), q_i = f(x_i; w, b)$ ，令

$$q_i = \frac{1}{1 + \exp(-w^T x_i - b)} \triangleq \sigma(w^T x_i + b) \quad (38)$$

$$\sigma(x) = \frac{1}{1 + \exp(-x)} + \text{交叉熵损失} = \text{逻辑回归} \quad (39)$$

$$\text{广义逻辑回归 GLR} \min_w \sum (-y_i \ln q_i - (1 - y_i) \ln(1 - q_i)) \quad (40)$$

$$\text{其中 } q_i = \frac{1}{1 + \exp(-w^T \Phi(x_i))} \quad (41)$$

分类器为 $y = \text{sign}(-w^T \Phi(x))$

GLR 的解

$$L(w) = \sum (-y_i \ln q_i - (1 - y_i) \ln(1 - q_i)) \quad (42)$$

$$\nabla L(w) = \sum (q_i - y_i) \Phi(x_i) = \Phi^T (q - y) \quad (43)$$

$$\nabla^2 L(w) = \sum q_i (1 - q_i) \Phi(x_i) \Phi(x_i)^T = \Phi^T R \Phi \quad (44)$$

$$\text{其中 } R = \text{diag}(q_i (1 - q_i)), \text{ 根据牛顿迭代法} \quad (45)$$

$$w^{new} = w^{old} - (\Phi^T R \Phi)^{-1} \Phi^T (q - y) \quad (46)$$

$$\text{令 } z \triangleq \Phi w^{old} - R^{-1} (q - y), \text{ 有} \quad (47)$$

$$w^{new} \text{ 可视化为加权最小二乘问题的解} \quad (48)$$

$$\min_z \frac{1}{2} (z - \Phi w)^T R (z - \Phi w) \quad (49)$$

当 q_i 远离 0.5 时权重变低

朴素贝叶斯

$$P(Y = i|X) = \frac{P(Y = i)P(X|Y = i)}{\sum_{j=1}^n P(Y = j)P(X|Y = j)} \quad (50)$$

其中 $P(X|Y = i) = \prod_{j=1}^n P(X_j|Y = i)$

K-NN

$$\hat{f}(x) \triangleq \text{sign}(\frac{1}{k} \sum_{j=1}^k y^{(n_j)}, s.t. x^{(n_j)} \in N_k(x) \quad (51)$$

稀疏表示

$$x \approx \sum_{i=1}^n \alpha_i x^{(i)}, s.t. \sum_{i=1}^n I(\alpha_i \neq 0) = k \quad (52)$$

$$\alpha = (\alpha_1, \dots, \alpha_n)^T \text{ 称为 } K\text{-稀疏向量, 分类器为} \quad (53)$$

$$\hat{f}(x) \triangleq \text{sign}(\frac{1}{k} \sum_{j=1}^k y^{(n_j)}, s.t. \alpha_{n_j} \neq 0 \quad (54)$$

解 α ：

$$\min (x - X\alpha)^T (x - X\alpha) + \lambda \|\alpha\|_1 \quad (55)$$

其中 $X = (x^{(1)}, \dots, x^{(n)})$ ，把 2-范数换成 1-范数也行

硬边界 SVM 假设数据线性可分，分类器无差，分类边界 $w^T x_i + b = 0$ ，

则 $y_i = \text{sign}(w^T x_i + b)$ ，距离可写为

$$d_i = \frac{|w^T x_i + b|}{\|w\|_2} = \frac{y_i (w^T x_i + b)}{\|w\|_2} \quad (56)$$

最大间隔问题为 $\max_w w, b \min_i d_i$ ，记为

$$\max_{w,b} \gamma(w, b), \gamma(w, b) \triangleq \min_{i=1, \dots, n} \frac{y_i (w^T x_i + b)}{\|w\|_2} \quad (57)$$

考虑到 $\forall \alpha > 0, \gamma(\alpha w, \alpha b) = \gamma(w, b)$ ，问题化为

$$\max_{w,b} \frac{1}{\|w\|_2}, \min_i y_i (w^T x_i + b) = 1 \quad (58)$$

进一步松弛为

$$\min_{w,b} \frac{1}{\|w\|_2^2}, s.t. \forall i, y_i (w^T x_i + b) \geq 1 \quad (59)$$

拉格朗日乘子 $\alpha = (\alpha_1, \dots, \alpha_n)^T$ ，拉格朗日函数为

$$L(w, b, \alpha) = \frac{1}{2} \|w\|_2^2 + \sum_{i=1}^n \alpha_i (1 - y_i (w^T x_i + b)) \quad (60)$$

KKT 条件为

$$\nabla_w L(w, b, \alpha) = 0 \Rightarrow w = \sum_{i=1}^n \alpha_i y_i x_i \quad (61)$$

$$\frac{\partial}{\partial b} L(w, b, \alpha) = 0 \Rightarrow \sum_{i=1}^n \alpha_i y_i = 0 \quad (62)$$

$$\alpha_i \geq 0, i = 1, \dots, n \quad (63)$$

$$1 - y_i (w^T x_i + b) \leq 0, i = 1, \dots, n \quad (64)$$

$$\alpha_i (1 - y_i (w^T x_i + b)) = 0, i = 1, \dots, n \quad (65)$$

当 $\alpha_i > 0$ 时，距离满足 $y_i (w^T x_i + b) = 0$ ，只有这些点才对计算 w 有贡献，称为支持向量

软边界 SVM 数据通常不线性可分；有时可分，但为扩大间隔，去掉一些。

$$\min_{w,b} \frac{1}{2} \|w\|_2^2, s.t. \sum_{i=1}^n I(y_i (w^T x_i + b) < 1) \leq c \quad (66)$$

$$\text{用代理损失函数 } \max(1 - y_i (w^T x_i + b), 0), \text{ 用拉格朗日法} \quad (67)$$

$$\min_{w,b} \frac{1}{2} \|w\|_2^2 + \lambda \sum_{i=1}^n \max(1 - y_i (w^T x_i + b), 0) \quad (68)$$

$$\text{令 } \xi_i = \max(0, 1 - y_i (w^T x_i + b)), \text{ 再松弛为} \quad (69)$$

$$\min_{w,b} \frac{1}{2} \|w\|_2^2 + \lambda \sum_{i=1}^n \xi_i, s.t. \forall i, \xi_i \geq 1 - y_i (w^T x_i + b), \quad (70)$$

$$\xi_i \geq 0 \quad (71)$$

$$\text{拉格朗日函数为} \quad (72)$$

$$L(w, b, \xi, \alpha, \beta) = \frac{1}{2} \|w\|_2^2 + \lambda \sum_{i=1}^n \xi_i - \sum_{i=1}^n \beta_i \xi_i \quad (73)$$

$$- \sum_{i=1}^n \alpha_i (y_i (w^T x_i + b) - 1 + \xi_i) \quad (74)$$

KKT 条件为

$$\nabla_w L(w, b, \xi, \alpha, \beta) = 0 \Rightarrow w = \sum_{i=1}^n \alpha_i y_i x_i \alpha_i y_i = 0 \quad (75)$$

$$\frac{\partial}{\partial b} L(w, b, \xi, \alpha, \beta) = 0 \Rightarrow \sum_{i=1}^n \alpha_i y_i = 0 \quad (76)$$

$$\frac{\partial}{\partial \xi} L(w, b, \xi, \alpha, \beta) = 0 \Rightarrow \alpha_i + \beta_i = \lambda \quad (77)$$

$$\alpha_i \geq 0, \beta_i \geq 0, i = 1, \dots, n \quad (78)$$

$$\xi_i \geq 1 - y_i (w^T x_i + b), \xi_i \geq 0, i = 1, \dots, n \quad (79)$$

$$\alpha_i (y_i (w^T x_i + b) - 1 + \xi_i) = 0, i = 1, \dots, n \quad (80)$$

$$\beta_i \xi_i = 0, i = 1, \dots, n \quad (81)$$

代替 β_i ，得

$$w = \sum_{i=1}^n \alpha_i y_i x_i \quad (82)$$

$$\sum_{i=1}^n \alpha_i y_i = 0 \quad (83)$$

$$0 \leq \alpha_i \leq \lambda, i = 1, \dots, n \quad (84)$$

$$\xi_i \geq 1 - y_i (w^T x_i + b), \xi_i \geq 0, i = 1, \dots, n \quad (85)$$

$$\alpha_i (y_i (w^T x_i + b) - 1 + \xi_i) = 0, i = 1, \dots, n \quad (86)$$

$$(\lambda - \alpha_i) \xi_i = 0, i = 1, \dots, n \quad (87)$$

所有距离可分为 3 类

$$y_i (w^T x_i + b) > 1 \Rightarrow \alpha_i = 0, \xi_i = 0 \quad (88)$$

$$y_i (w^T x_i + b) = 1 \Rightarrow 0 < \alpha_i < \lambda, \xi_i = 0 \quad (89)$$

$$y_i (w^T x_i + b) < 1 \Rightarrow \alpha_i = \lambda, \xi_i > 0 \quad (90)$$

当 $\alpha_i > 0$ 时，距离满足 $y_i (w^T x_i + b) \leq 1$ ，只有这些点对计算 w 有贡献，称为支持向量

对偶问题 $\max_{\alpha} \alpha \min_{w,b} \xi_i, L(w, b, \xi, \alpha)$ ，由 KKT 条件得

$$\max_{\alpha} - \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n \alpha_i \alpha_j y_i y_j x_i^T x_j + \sum_{i=1}^n \alpha_i \quad (91)$$

$$s.t. 0 \leq \alpha_i \leq \lambda, \sum_{i=1}^n \alpha_i y_i = 0 \quad (92)$$

分类器为 $y = \text{sign}(\sum_{i=1}^n \alpha_i y_i x_i^T x + b)$ ， b 由 $0 < \alpha_i < \lambda$ 的支持向量计算得出

带基函数 SVM & 核 SVM

$$\min_{w,b} \frac{1}{2} \|w\|_2^2 + \lambda \sum_{i=1}^n \xi_i, s.t. \forall i, \xi_i \geq 0, \quad (93)$$

$$\xi_i \geq 1 - y_i (w^T \Phi(x_i) + b)$$

$$P(X_i|Z_i) = \prod_{j=1}^k (N(X_i; \mu_j, \Sigma_j))^{I(z_i=j)} \tag{89}$$

$$P(X_i) = \sum_{j=1}^k w_j N(X_i; \mu_j, \Sigma_j) \tag{90}$$

$$\begin{aligned} X_i \text{ i.i.d.} \\ \text{期望最大化算法 (EM) for GMM} \quad &\text{已知参数 } w_j, \mu_j, \Sigma_j \\ P(Z_i = j|X_i) &= \frac{w_j N(x_i; \mu_j, \Sigma_j)}{\sum_{j=1}^k w_j N(x_i; \mu_j, \Sigma_j)} \triangleq \gamma_{ij} \tag{91} \\ w_j &= \frac{\sum_i \gamma_{ij}}{n}, \mu_j = \frac{\sum_i \gamma_{ij} x_i}{\sum_i \gamma_{ij}} \\ \Sigma_j &= \frac{\sum_i \gamma_{ij} (x_i - \mu_j)(x_i - \mu_j)^T}{\sum_i \gamma_{ij}} \tag{92} \end{aligned}$$

$$\begin{aligned} \text{迭代(91)和(92)直到收敛} \\ f(x_i) = \arg \max_j P(Z_i = j|X_i = x_i) = \arg \max_j \gamma_{ij} \tag{93} \end{aligned}$$

$$\begin{aligned} \text{K-MEANS 与 GMM} \quad &\text{K-MEANS 是 GMM 的特例, 认为 } w_j = 1/k, \Sigma_j = I, \gamma_{ij} \text{ 计算如下} \\ &\gamma_{i,j} = 1, \text{ if } P(Z_i = j|X_i) = \max_l P(Z_i = l|X_i) \\ &0, \text{ otherwise} \tag{94} \end{aligned}$$

k-means 使用硬分配 (hard assignment), GMM-EM 用软分配, 因此 k-means 对于具有不同大小、密度或不规则形状的簇存在局限性

EM 解决带隐变量的最大似然估计问题。观测变量 X , 隐变量 Z , 待估参数 θ

算法 1 EM 算法
<ol style="list-style-type: none">$t \leftarrow 0$, initialize θ^0 repeat <ol style="list-style-type: none">E-step: $Q(\theta) = \mathbb{E}_{Z \sim P(Z X=x, \theta^t)}[\log P(X, Z; \theta)]$ M-step: $\theta^{t+1} = \arg \max_{\theta} Q(\theta)$ until $\left\ \theta^{t+1} - \theta^t \right\ < \epsilon$ $\hat{\theta} = \theta^{t+1}$

$$\begin{aligned} \log P(X; \theta) &= (\sum_z P(Z = z|X; \theta^t)) \log P(X; \theta) \\ &= \sum_z P(Z = z|X; \theta^t) \log P(X, Z; \theta) \\ &= \sum_z P(Z = z|X; \theta^t) \log P(Z|X; \theta) \\ &\triangleq Q(\theta) + H(\theta) \end{aligned} \tag{95}$$

$H(\theta^t)$ 是 $P(Z|X; \theta^t)$ 和 $P(Z|X; \theta)$ 间的交叉熵, 有 $H(\theta) \geq H(\theta^t)$. 我们优化 $Q(\theta)$, 有 $Q(\theta^{t+1}) \geq Q(\theta^t)$. 因此有 $\log P(X; \theta^{t+1}) \geq \log P(X; \theta^t)$. EM 为贪心, 每一步 P 不减

非参数聚类

基于距离的聚类 凝聚聚类: 自顶向下, 合并相近数据簇; 分离聚类: 自顶向下, 通过切割将距离最长的边来分割子图

基于距离的聚类 Mean-shift: 用 Parzen 窗估计局部密度并计算局部均值, 将局部模式 (密度最高的点) 移到平均处; DBSCAN: 给定一个随机选择的数据, 找到它的最近邻居并估计局部密度; 如果密度足够高, 则将此数据及邻居设置为簇, 并尝试扩展, 直到到达低密度区域

嵌入 embedding 增加特征的维度, 或为对象构建高维特征向量

例: 评分预测. 设有 n 个电影和 m 个用户, 每部电影有 1 个嵌入向量 $m_i \in \mathbb{R}^p$, 每个用户有 1 个嵌入向量 $u_i \in \mathbb{R}^p$, 评分为 $r_{ij} = m_i^T u_j$. 评分矩阵为 $R = M^T U$, 为一个低秩矩阵. 若已知 R , 可用截断 SVD 得到 M, U 用于评分预测

例: 词嵌入, 可用 LDA 忽略 (减小类内方差, 增大类间差别)

6 基于树的模型与集成学习

回归树 regression stump

$$f(x) = (\beta_1 - \beta_0) \text{sign}(w^T x + b) + \beta_0 \tag{96}$$

模型组合 线性组合 (线性模型等价于另一线性模型. 一般不如此组合. 若基模型表现良好且有多样性 (well and diversely), 则组合模型一定有提升.

保证 well and diversely 的方法: 训练数据多样性 (数据分割, 特征分割, 不同核); 训练方法多样性

性能随多样性存在矛盾

模型组合方法: 简单相加/投票 (bagging, boosting); 训练组合 (stacking: 每个基模型学一个特征, 再训练一个总的模型进行组合); 局部自适应组合 (树模型: 将输入空间分为若干子域, 每个基模型处理一个)

bootstrap aggregating (bagging) 使用 bootstrap sampling (自助抽样) 得到数据多样性: 给定数据集 $x^{(i)}|_{i=1}^n$, 进行 n 次带放回的均匀采样.

一个数据抽不到的概率为 $(1 - \frac{1}{n})^n$, 当 $n \rightarrow +\infty$, 有 $1/e$ 的数据抽不到. 使用一次自助抽样训练一个基模型再组合起来.

Boosting 多个基模型一个一个地训练, 组合起来的模型会一点一点变好.

Boosting 对 回归

$$F_P(x) = \sum_{j=1}^p \beta_j f_j(x) \tag{97}$$

其中 f_j 为基模型, F_P 为总的模型. Boosting 中模型一个一个地训, 可考虑

$$F_j(x) = F_{j-1}(x) + f_j(x) \tag{98}$$

使用平方损失函数

$$\begin{aligned} L(f_j) &= \sum_{i=1}^n (y_i - F_j(x_i))^2 \\ &= \sum_{i=1}^n (y_i - F_{j-1}(x_i) - f_j(x_i))^2 \tag{99} \end{aligned}$$

令 $r_i = y_i - F_{j-1}(x_i)$, 则 f_j 在回归 $(x^{(i)}, r^{(i)})$, 即除 f_1 外, f_i 在回归残差

算法 2 AdaBoost Algorithm
Require: $(x_i, y_i) _{i=1}^n, y_i \in \{+1, -1\}$ Ensure: $F_P(x) = \text{sign}(\sum_{j=1}^P \beta_j f_j(x))$ <ol style="list-style-type: none">for $j = 1, \dots, p$ do if $j=1$ then $w_{ij} = 1/n$ else $w_{ij} = w_{i,j-1} \exp(-y_i \beta_{j-1} f_{j-1}(x_i))$ $w_{ij} = \frac{w_{ij}}{\sum_i w_{ij}}$ 用 w_{ij} 给第 i 个数据加权, 训练分类器 f_j 计算加权错误率 $e_j = \sum_i w_{ij} I(y_i \neq f_j(x_i))$ $\beta_j = \frac{1}{2} \ln \frac{1-e_j}{e_j}$

AdaBoost 在加权数据上训练的两种方法

1. 对 loss 加权. 如 $\sum_{i=1}^n w_{ij} (-y_i \ln q_i - (1-y_i) \ln(1-q_i))$

2. 无显式 loss, 可将 w_{ij} 作为概率, 对数据重采样

若 f_j 对 x_i 分类正确, 则权重下降 ($\times \exp(-\beta_j)$), 反之上升 ($\times \exp(\beta_j)$). f_{j+1} 将更关注分类错误的数据.

实际上 AdaBoost 使用指数损失函数, 设分类器 $y = \text{sign}(x) \in \{+1, -1\}$

$$L(f; x, y) = \exp(-y f(x)) \tag{100}$$

最小化 e_j 实际上就是在加权数据上训练一个基分类器, 指数损失函数也是 0-loss 的一个上界

决策树 一棵树, 每个内部节点对应某些特征的条件, 每个叶节点表示一类 (分类) 或一个值 (回归)

算法 3 HA
Require: A set of training data $\mathcal{D} = \{x_i, y_i\}$ Ensure: A classification tree or regression tree T <ol style="list-style-type: none">function HA(\mathcal{D}) if \mathcal{D} 不用分裂 then return 叶节点 else 寻找一个条件来分裂 将 \mathcal{D} 按条件分割为 $\mathcal{D}_1, \mathcal{D}_2, \dots$ $T_1 = \text{HA}(\mathcal{D}_1), T_2 = \text{HA}(\mathcal{D}_2), \dots$ 建树, 条件为根, T_1, T_2, \dots 为子树 return 生成的树

Hunt's algorithm (HA)

分裂条件选取 贪心: 最小化当前经验风险

回归树: 若使用平方损失

$$\min_{j, t_j, \alpha_j, \beta_j} \sum_{i=1}^n (\alpha_j I(x_j^{(i)} < t_j) + \beta_j I(x_j^{(i)} > t_j) - y^{(i)})^2 \tag{101}$$

取 α_j 为 $\{y^{(i)} | x_j^{(i)} < t_j\}$ 的均值, β_j 同理

二元分类树: 设 p_0 为 0 的百分数, p_1 同理. 常用 3 个指标:

误分率: $\mathcal{E}(D) = \min(p_0, p_1)$

熵: $H(D) = -p_1 \log_2 p_1 - p_0 \log_2 p_0$

Gini 指数: $G(D) = 1 - p_1^2 - p_0^2$

使用熵或带信息量判定分裂几叉, 定义如下:

$$r \triangleq \frac{H(D) - \sum_i \frac{|D_i|}{|D|} H(D_i)}{-\sum_i \frac{|D_i|}{|D|} \log_2 \frac{|D_i|}{|D|}} \tag{102}$$

防止过拟合 早停: 提前停止分裂, 即使不能分裂

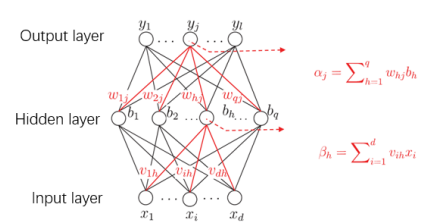
剪枝: 从训练好的树上移除枝节

一般来说剪枝优于早停, 剪枝考虑联合成本:

$$J(T) \triangleq \mathcal{E}(D, T) + \lambda |T| \tag{103}$$

\mathcal{E} 表示经验风险, $|T|$ 表示树的复杂度

7 图模型和深度学习



BP 网络模型 训练数据 $D = \{(x_k, y_k)\}, x_k \in \mathbb{R}^d, y_k \in \mathbb{R}^l$

待学习参数: 权重: v_{ih}, w_{hj} ; 偏置: γ_h, θ_j

梯度下降 GD 给定样本 (x^k, y^k) , 输出 \hat{y}^k

$$\begin{aligned} b_h &= f(\beta_h - \gamma_h), \beta_h = \sum_{i=1}^d v_{ih} x_i^k \\ \hat{y}_j^k &= f(\alpha_j - \theta_j), \alpha_j = \sum_{h=1}^q w_{hj} b_h \tag{104} \end{aligned}$$

$$E_k = \frac{1}{2} \sum_{j=1}^l (\hat{y}_j^k - y_j^k)^2$$

每个参数 ν 更新为

$$\nu \leftarrow \nu + \Delta \nu \quad \Delta \nu = -\eta \frac{\partial E_k}{\partial \nu} \tag{105}$$

$$\frac{\partial E_k}{\partial \hat{y}_j^k} = \hat{y}_j^k - y_j^k$$

$$g_j = \frac{\partial E_k}{\partial \alpha_j} = (\hat{y}_j^k - y_j^k) f'(\alpha_j - \theta_j)$$

$$\Delta w_{hj} = -\eta g_j b_h$$

$$\Delta \theta_j = \eta g_j$$

$$\frac{\partial E_k}{\partial b_h} = \sum_{j=1}^l \frac{\partial E_k}{\partial \alpha_j} \frac{\partial \alpha_j}{\partial b_h} = \sum_{j=1}^l g_j w_{hj}$$

$$e_h = \frac{\partial E_k}{\partial \beta_h} = (\sum_{j=1}^l g_j w_{hj}) f'(\beta_h - \gamma_h)$$

$$\Delta v_{ih} = -\eta e_h x_i^k$$

$$\Delta \gamma_h = \eta e_h$$

频繁更新参数, 不同样本更新可能不一致, 随机梯度下降 (SGD)

$$\Delta \nu = -\eta \frac{\partial E}{\partial \nu}, E = \sum_{k=1}^m E_k \tag{107}$$

标值梯度下降, 参数更新不频繁, 累计误差可能下降缓慢. 实际应用中使用小批量 (small batches of) 数据

$$\begin{aligned} \nu &\leftarrow \nu + \Delta \nu^{(t)} \\ \Delta \nu^{(t)} &= -\eta \frac{\partial E_k}{\partial \nu} + \alpha \Delta \nu^{(t-1)} \tag{108} \end{aligned}$$

使梯度下降更平滑

局部最小值 解决方法: 多次不同初始化; 模拟退火; 遗传算法

防止过拟合 使用验证集 (早停)

正则化 $E = \frac{1}{m} \sum_{k=1}^m E_k + \omega \sum_i \nu_i^2$

Dropout 和 DropConnect

8 统计学习理论

最小描述长度原理 MDL. Kolmogorov 复杂度: 一个对象的 Kolmogorov 复杂度是输出为该对象的计算机程序的最短长度

例: "ababababababababababababababab" 可用 print(' ab' *16). Kol-

mogorov 复杂度在 Python 中不超过 14

随机性: 一个字符串是随机的, 当且仅当每个产生该字符串的计算机程序至少与字符串本身一样长

最小描述长度原理 (MDL): 统计学习任务是找到数据的最短描述

对于一个数据集 D 和假设空间 H , 可表示为:

$$h^* = \arg \min_{h \in H} L(h) + L(D|h) \tag{109}$$

例: 二元分类数据集 $\{(x^{(i)}, y^{(i)})|_{i=1}^n\}$, 多个模型

h_1, \dots, h_m , 有如下几种编码方式: (1) 编码每个 $x^{(i)}$; (2) 编码一些 h_j ; (3) 计算 $\hat{y}^{(k)} = \text{sign}(h_j(x^{(i)}))$, 编码集合

$\{i|y^{(i)} \neq \hat{y}^{(k)}\}$. 最小编码长度的模型最优.

若所有 h_j 和 i 均用固定长度编码, MDL 相当于最小化 0-1-loss 的经验风险

若 h_j 定长编码, i 变长编码, MDL 相当于给每组数据加权

若 h_j 变长编码, i 定长编码, MDL 相当于对参数有偏好

MDL 无需概率论的解释, 更加灵活