数据挖掘与应用

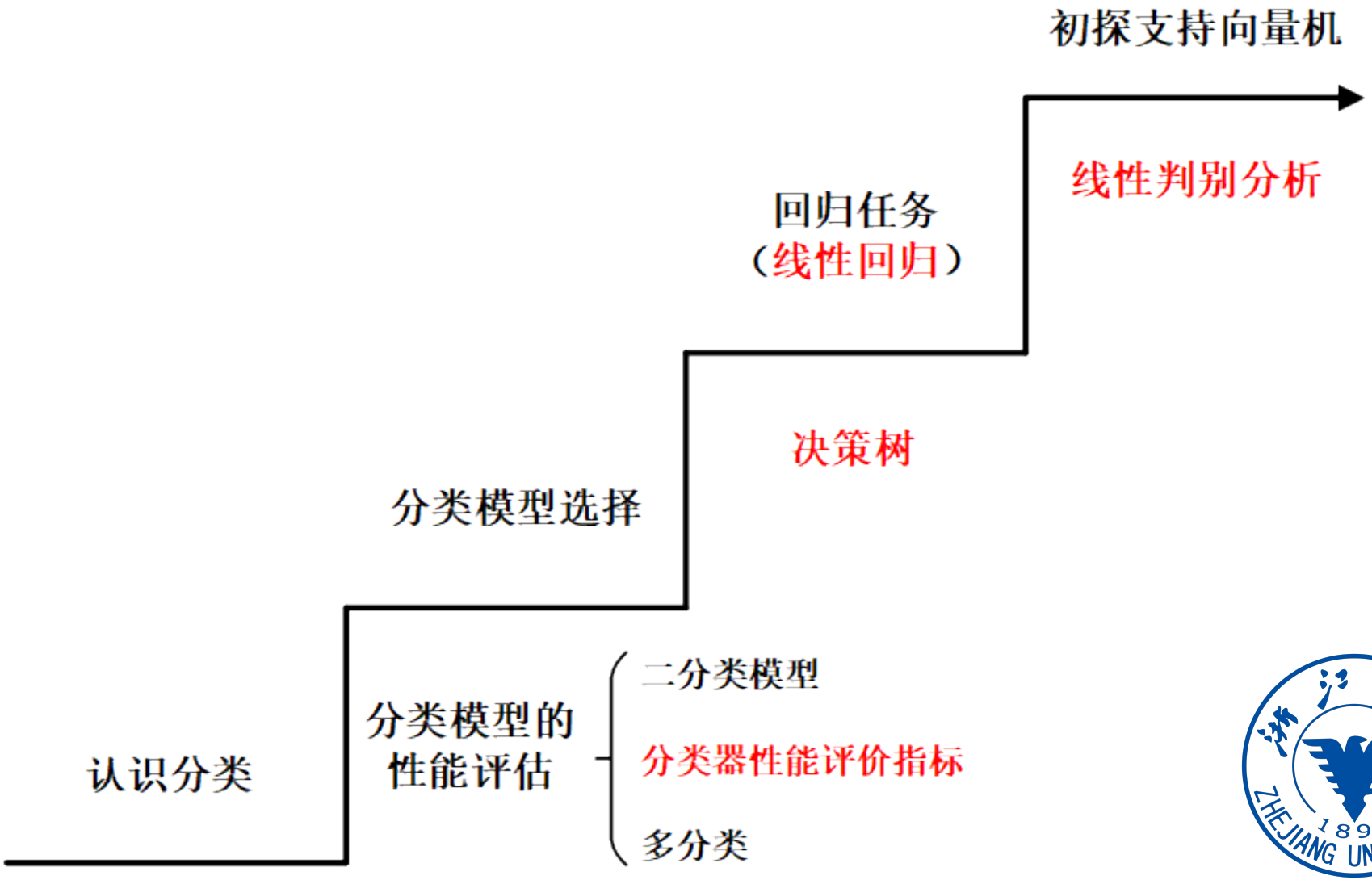
分类

授课教师：周晟

浙江大学　软件学院

2021.09

上节课回顾


初探支持向量机
线性判别分析
回归任务
（线性回归）
决策树
分类模型选择
二分类模型
分类器性能评价指标
多分类
分类模型的
性能评估
认识分类

数据挖掘十大算法

1. C4.5(决策树)
2. SVM
3. Naive Bayes
4. EM
5. Apriori（频繁项挖掘）
6. k-Means
7. PageRank
8. AdaBoost
9. kNN
10. CART （分类回归树）

1

[1] https://wizardforcel.gitbooks.io/dm-algo-top10/content/index.html

课程内容

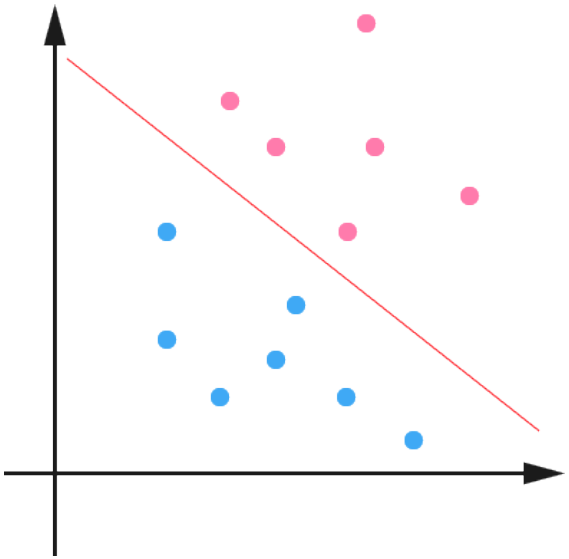
1. 深入理解 SVM
   - SVM 内容回顾
   - 支持向量
   - 拉格朗日乘子与优化
2. 贝叶斯分类器
   - 贝叶斯
   - 生成式贝叶斯分类器
   - 极大似然估计
   - 拉普拉斯修正
   - EM 算法

分类与超平面

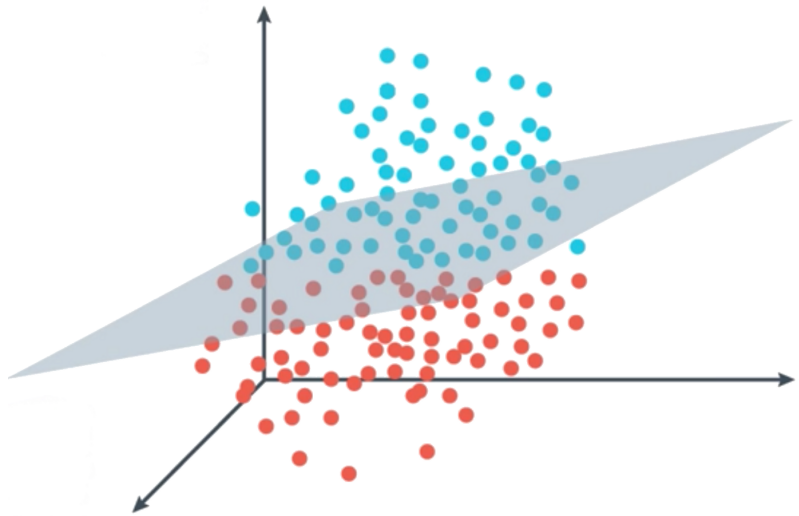
一个 n 维特征空间中的线性分类器就是要在特征空间中找到一个超平面，其方程可以表示为：

$$w^T x + b = 0$$

理想的超平面是在特征空间中将两类数据分隔开，即两类数据分别分布在超平面的两侧（虽然这种条件不一定可以满足）。



二维空间超平面



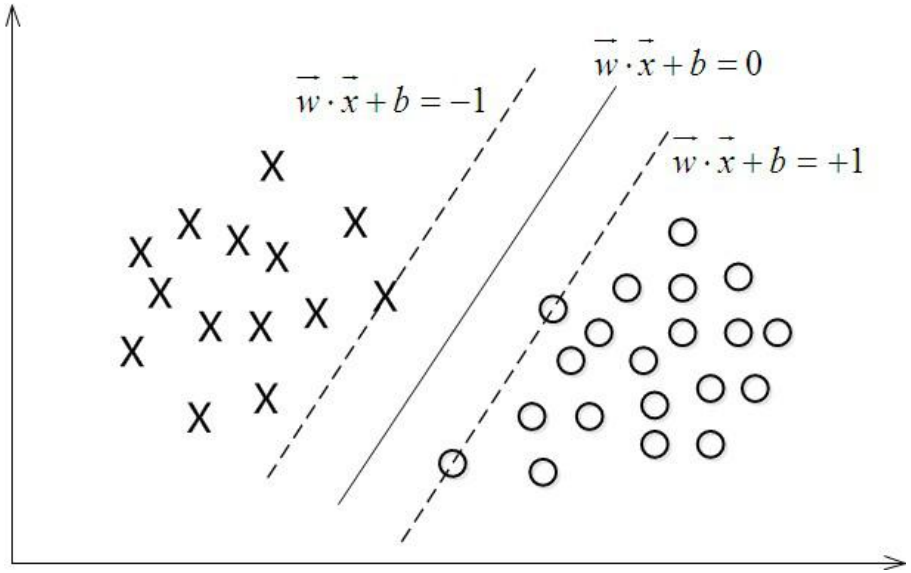
三维空间超平面

分类与超平面

为了计算方便，首先将类别信息数值化。

$$f(x) = w^T x + b \begin{cases} > 0 & y = 1 \\ = 0 & \text{超平面} \\ < 0 & y = -1 \end{cases}$$

这是一种理想情况，真实的分类器往往难以找到这种超平面。



分类的超平面

对于一个数据点 x 进行分类，实际上是通过把 x 带入到 $f(x) = wTx + b$ 算出结果然后根据其正负号来进行类别划分的。

函数间隔与几何间隔

函数间隔

函数间隔（functional margin）定义为：

$$\hat{\gamma} = y\left(w^T x + b\right) = yf(x)$$

因为负类的标签 $y = -1$，因此函数间隔具有非负性。

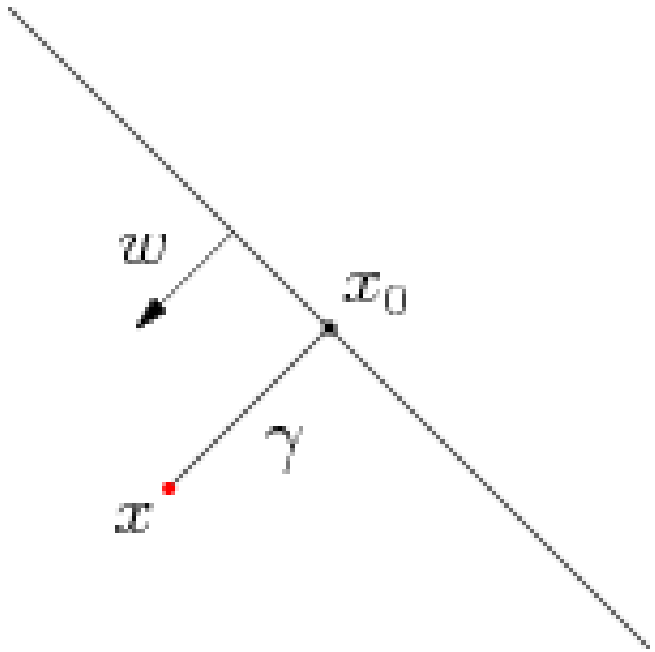
几何间隔

几何间隔定义为点到超平面的距离。

函数间隔与几何间隔

定义样本 x 在超平面上的投影为 x_0，w 是垂直于超平面的向量，易得：

$$x = x_0 + \gamma \frac{w}{\|w\|}$$



由于 x_0 是超平面上的点，满足 $f(x_0) = 0$，易得：

$$\gamma = \frac{w^T x + b}{\|w\|} = \frac{f(x)}{\|w\|}$$

因此，函数间隔与几何间隔满足关系：

$$\tilde{\gamma} = y\gamma = \frac{\hat{\gamma}}{\|w\|}$$

函数间隔与几何间隔

分类的最优间隔

对一个数据点进行分类，当它的间隔越大的时候，分类的置信度越大。对于一个包含 n 个点的数据集，我们可以很自然地定义它的间隔为所有这点的间隔值中最小的那个。为了使得分类的置信度高，我们希望所选择的超平面能够最大化这个间隔值。

函数间隔的缺陷

函数间隔可以在超平面不变的情况下被取得任意大，而几何间隔因为有对 $||w||$ 的缩放则没有这个问题，因此分类问题转化为最大化几何间隔：

$$\max \tilde{\gamma}$$

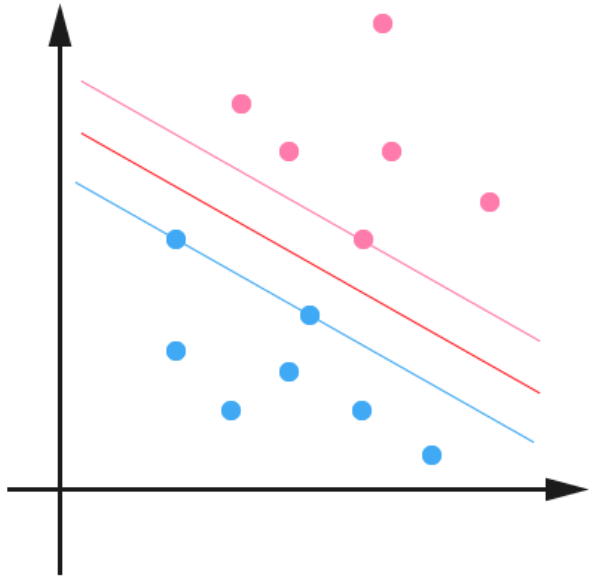
$$y_i \left(w^T x_i + b\right) = \hat{\gamma}_i \geq \hat{\gamma}, \quad i = 1, \ldots, n$$

最优超平面

为了计算方便，令函数间隔为1，目标函数转变为：

$$\max \frac{1}{\|w\|}$$

$$\text{s.t. } ,y_i\left(w^T x_i + b\right) \geq 1, i = 1, \ldots, i$$



支持向量

支持向量

给定特征空间中的几何间隔定义，最优超平面由距离超平面最近的若干个样本决定。因此支持向量定义为距离分类超平面最近的若干个点。根据函数间隔的特性，支持向量满足：

$$y(w^T x + b) = 1$$

而特征空间中的其他点则有：

$$y(w^T x + b) > 1$$

这些点对最有超平面的学习没有实质贡献，因此可以最大程度地提升存储和计算的效率。

支持向量机的优化

原优化目标：

$$\max \frac{1}{\|w\|} \quad \text{s.t. } , y_i \left(w^T x_i + b\right) \geq 1, i = 1, \ldots, n$$

为了计算方便，对优化目标进行等价变换：

$$\min \frac{1}{2}\|w\|^2 \quad \text{s.t. } , y_i \left(w^T x_i + b\right) \geq 1, i = 1, \ldots, n$$

该优化目标函数是二次的，约束条件是线性的，可以使用经典的二次优化工具进行优化。

支持向量机的优化

经典二次优化工具的缺陷：

1. 求解效率不高
2. 难以推广到非线性分类问题

拉格朗日对偶性

在带约束的优化问题中，常常利用拉格朗日对偶性 (Lagrange duality) 将原始问题转为对偶问题，通过解决对偶问题而得到原始问题的解。

对偶问题的优势有[2]：

1. 对偶问题的对偶是原问题；
2. 无论原始问题是否是凸的，对偶问题都是凸优化问题；
3. 对偶问题可以给出原始问题一个下界；
4. 当满足一定条件时，原始问题与对偶问题的解是完全等价的；

[2] https://zhuanlan.zhihu.com/p/38182879

等式约束优化问题

等式优化问题

$$\begin{aligned} \min \quad & f(\mathbf{x}) \\ \text{s.t.} \quad & g(\mathbf{x}) = 0 \end{aligned}$$

通过引入拉格朗日乘子可将等式约束优化问题转化为无约束优化问题

$$\min_{\mathbf{x}} L(\mathbf{x}) = \min_{\mathbf{x}} f(\mathbf{x}) + \lambda g(\mathbf{x})$$

通过对此式求导即可得最优解的必要条件：

$$\nabla_{\mathbf{x}} L = \frac{\partial L}{\partial \mathbf{x}} = \nabla f + \lambda \nabla g = \mathbf{0}$$

联立方程即可找到最优解

不等式约束优化问题

不等式优化问题

$$\begin{aligned} \min \quad & f(\mathbf{x}) \\ \text{s.t.} \quad & g(\mathbf{x}) \leq 0 \end{aligned}$$

此时最优解分为两种情况：

1. 内部解（interior solution），满足 $g(x^*) < 0$，此时约束条件无效
2. 边界解（boundary solution），满足 $g(x^*) = 0$，此时约束条件有效

互补松弛性

互补松弛性 (complementary slackness)

在不等式约束优化问题中，内部解和边界解分别满足如下条件：

1. 内部解：约束条件无效，内部解满足条件 $\partial f = 0$ 和 $\lambda = 0$
2. 边界解：与等式约束一致。由于最小化 f，梯度应指向 $g(x) < 0$ 的方向，而 g 的梯度则指向 $g(x) > 0$ 的情况。因此需满足 $\lambda \geq 0$，也称为对偶可行性 (dual feasibility)。

KKT 条件

KKT 条件

最佳解的必要条件包括 Lagrangian 函数的定常方程式、原始可行性、对偶可行性，以及互补松弛性：

$$\begin{aligned} \nabla_{\mathbf{x}} L &= \nabla f + \lambda \nabla g = \mathbf{0} \\ g(\mathbf{x}) &\leq 0 \\ \lambda &\geq 0 \\ \lambda g(\mathbf{x}) &= 0 \end{aligned}$$

拉格朗日函数的通用形式

带约束优化问题

$$\begin{aligned} &\min_{x \in R^n} && f(x) \\ &s.t. && c_i(x) \le 0, \quad i = 1, 2, \cdots, k \\ &&& h_j(x) = 0, \quad j = 1, 2, \cdots, l \end{aligned}$$

其对应的拉格朗日函数一般形式为：

$$L(x, \alpha, \beta) = f(x) + \sum_{i=1}^{k} \alpha_i c_i(x) + \sum_{j=1}^{l} \beta_j h_j(x)$$

其中 $\alpha_i \ge 0, \beta$ 为拉格朗日乘子。

拉格朗日函数等价性

定义最大化拉格朗日函数的目标函数：

$$\theta_P(x) = \max_{\alpha,\beta;\alpha_i \geq 0} L(x, \alpha, \beta)$$

重要结论

上述定义的最大化拉格朗日函数等价于原始函数的目标，即

$$\min f(x) = \min \max L(x, \alpha, \beta)$$

这也是 SVM 使用拉格朗日乘子法的理论基础。

拉格朗日函数等价性证明

$$\theta_P(x) = \max_{\alpha,\beta;\alpha_i \geq 0} L(x,\alpha,\beta) = \max f(x) + \sum_{i=1}^{k} \alpha_i c_i(x) + \sum_{j=1}^{l} \beta_j h_j(x)$$

反证法

违反原始约束条件的情况有

1. $c_i(x) > 0$: **令** $\alpha_i \to +\infty$，**此时** $\theta_P(x) \to +\infty$
2. $h_j(x) \neq 0$: **令** $\beta_j h_j(x) \to +\infty$, **此时** $\theta_P(x) \to +\infty$

若数据符合约束条件，则易得：

$$\theta_P(x) = \max_{\alpha,\beta;\alpha_i \geq 0} \left[f(x) + \sum_{i=1}^{k} \alpha_i c_i(x) + \sum_{j=1}^{l} \beta_j h_j(x) \right] = f(x)$$

拉格朗日函数等价性证明

最大化拉格朗日函数的目标，满足如下性质：

$$\theta_P(x) = \begin{cases} f(x), x \text{ 满足原始问题约束} \\ +\infty, \text{ 否则} \end{cases}$$

因此最小化该函数与最小化原始函数等价：

$$p^* = \min_x \theta_P(x) = \min_x \max_{\alpha,\beta;\alpha_i \geq 0} L(x, \alpha, \beta)$$

也称为广义拉格朗日函数的极小极大问题

对偶问题

定义关于拉格朗日乘子 α, β 的函数：

$$\theta_D(\alpha, \beta) = \min_x L(x, \alpha, \beta)$$

极大化该函数：

$$\max_{\alpha,\beta;\alpha_i \geq 0} \theta_D(\alpha, \beta) = \max_{\alpha,\beta;\alpha_i \geq 0} \min_x L(x, \alpha, \beta)$$

称为广义拉格朗日函数的极大极小问题。

强对耦性和弱对耦性

给定拉格朗日的极小极大问题：

$$p^* = \min_x \theta_P(x) = \min_x \max_{\alpha,\beta;\alpha_i \geq 0} L(x, \alpha, \beta)$$

以及拉格朗日的极大极小问题：

$$\max_{\alpha,\beta;\alpha_i \geq 0} \theta_D(\alpha, \beta) = \max_{\alpha,\beta;\alpha_i \geq 0} \min_x L(x, \alpha, \beta)$$

两者满足如下关系：

$$\max_{\alpha,\beta;\alpha_i \geq 0} \min_x L(x, \alpha, \beta) \leq \min_x \max_{\alpha,\beta;\alpha_i \geq 0} L(x, \alpha, \beta)$$

这种关系称为弱对耦性（weak duality）
若等号满足，则称为强对耦性（strong duality）。

强对耦性

若原优化问题与对偶问题均存在最优解，则弱对偶性一定满足。而强对偶性则需要满足 Slater 条件与 KKT 条件。

强对耦性的优势

给定一个带约束优化问题，若优化目标满足强对偶性，则可以通过求对偶问题的最优解得到原目标的最优解。

使用拉格朗日乘子优化 SVM

利用拉格朗日乘子法，将带约束的优化问题转化为不带约束的优化问题：

$$\mathcal{L}(\boldsymbol{w}, b, \alpha) = \frac{1}{2}\|\boldsymbol{w}\|^2 + \sum_{i=1}^{m} \alpha_i \left(1 - y_i \left(\boldsymbol{w}^{\mathrm{T}} \boldsymbol{x}_i + b\right)\right)$$

或者

$$\mathcal{L}(\boldsymbol{w}, b, \alpha) = \frac{1}{2}\|w\|^2 - \sum_{i=1}^{n} \alpha_i \left(y_i \left(\boldsymbol{w}^{T} x_i + b\right) - 1\right)$$

令

$$\theta(w) = \max_{\alpha_i \geq 0} \mathcal{L}(w, b, \alpha)$$

易证带约束的最小化 $\frac{1}{2}\|\boldsymbol{w}\|^2$ 等价于最小化 $\theta(w)$

使用拉格朗日乘子优化 SVM

证明

$$\theta(w) = \max_{\alpha_i \geq 0} \mathcal{L}(w, b, \alpha) = \max \frac{1}{2}\|w\|^2 - \sum_{i=1}^{n} \alpha_i \left(y_i \left(\boldsymbol{w}^T x_i + b\right) - 1\right)$$

与 SVM 的优化目标一致

若样本不满足约束条件:

$$y_i(\boldsymbol{w}^T x_i + b) < 1$$

则 $\theta(w) = +\infty$
若样本满足约束条件，则 $\theta(w) = \frac{1}{2}\|w\|^2$

使用拉格朗日乘子优化 SVM

利用拉格朗日乘子，SVM 的优化目标转化为：

SVM 的优化目标

$$\min_{w,b} \theta(w) = \min_{w,b} \max_{\alpha_i \geq 0} \mathcal{L}(w, b, \alpha) = p^*$$

对应的对偶问题为：

$$\max_{\alpha_i \geq 0} \min_{w,b} \mathcal{L}(w, b, \alpha) = d^*$$

由于 SVM 优化目标满足 KKT 条件，因此将 SVM 的优化进一步转化为拉格朗日乘子的对偶问题。

SVM 的拉格朗日对偶问题求解

$$\frac{\partial \mathcal{L}}{\partial w} = 0 \Rightarrow w = \sum_{i=1}^{n} \alpha_i y_i x_i$$

$$\frac{\partial \mathcal{L}}{\partial b} = 0 \Rightarrow \sum_{i=1}^{n} \alpha_i y_i = 0$$

代入可得

$$\begin{aligned}\mathcal{L}(w, b, \alpha) &= \frac{1}{2} \sum_{i,j=1}^{n} \alpha_i \alpha_j y_i y_j x_i^T x_j - \sum_{i,j=1}^{n} \alpha_i \alpha_j y_i y_j x_i^T x_j - b \sum_{i=1}^{n} \alpha_i y_i + \sum_{i=1}^{n} \alpha_i \\ &= \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{n} \alpha_i \alpha_j y_i y_j x_i^T x_j\end{aligned}$$

SVM 的拉格朗日对偶问题求解

$$\begin{aligned}
&\max_{\alpha} \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{n} \alpha_i \alpha_j y_i y_j x_i^T x_j \\
&\text{s.t. } , \alpha_i \geq 0, i = 1, \ldots, n \\
&\sum_{i=1}^{n} \alpha_i y_i = 0
\end{aligned}$$

由于在梯度为 0 时，需满足

$$w = \sum_{i=1}^{n} \alpha_i y_i x_i$$

因此对于未知标签的节点，可得：

$$\begin{aligned}
f(x) &= \left(\sum_{i=1}^{n} \alpha_i y_i x_i \right)^T x + b \\
&= \sum_{i=1}^{n} \alpha_i y_i \langle x_i, x \rangle + b
\end{aligned}$$

SVM 求解过程

1. 带约束优化问题利用拉格朗日乘子转化为无约束优化
2. 因为 SVM 满足 KKT 条件，转而优化对偶问题
3. 优化对偶问题，得到最终解。SVM 对应的 KKT 条件为：

$$\begin{cases} \alpha_i \geqslant 0 \\ y_i f\left(\boldsymbol{x}_i\right) - 1 \geqslant 0 \\ \alpha_i \left(y_i f\left(\boldsymbol{x}_i\right) - 1\right) = 0 \end{cases}$$

贝叶斯决策论

贝叶斯决策论

对于分类任务而言，在所有相关概率都已知的理想情形下，贝叶斯决策论考虑如何基于这些概率和误判所造成的损失来选择最优的类别标签。

以多分类为例，假设有 K 种可能的类别标签，即 $Y = \{c_1, c_2, ..., c_K\}$，$\lambda_{i,j}$ 是将一个真实标签为 c_j 的样本误分类为 c_i 所产生的损失。

基于后验概率 $P(c_i|x)$，可以获得将样本 x 分类为 c_i 所产生的期望损失，即在样本 x 上的“条件风险”：

$$R(c_i|x) = \sum_{j=1}^{K} \lambda_{ij} P(c_j|x)$$

贝叶斯决策论

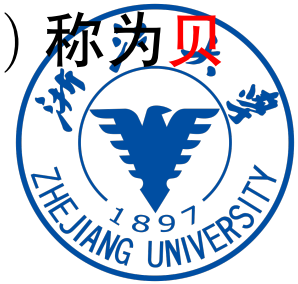
贝叶斯分类器的任务是寻找一个判定准则 $h : X \to Y$，以最小化总体风险。

$$R(h) = \mathbb{E}_x \left[R(h(x)|x) \right]$$

由此就产生了贝叶斯判定准则：想要最小化总体风险，只需要对每个样本都选择那个能使条件风险 $R(c|x)$ 最小的类别标签，即

$$h^*(x) = \arg\min_{c \in Y} R(c|x)$$

此时，h^* 称为贝叶斯最优分类器，与其对应的总体风险 $R(h^*)$ 称为贝叶斯风险。



贝叶斯决策论

优化目标是最小化分类错误率，则误判损失 λ_{ij} 可以写做：

$$\lambda_{ij} = \begin{cases} 0, & if \quad i = j; \\ 1, & otherwise, \end{cases}$$

此时条件风险为：

$$R(c|x) = 1 - P(c|x) \qquad (P(c_j|x)\text{为 0 或 1})$$

因此，最小化分类错误率的贝叶斯最优分类器为：

$$h^*(x) = \underset{c \in Y}{\arg\max}\, P(c|x)$$

即对于每个样本 x，选择能使其后验概率 $P(c|x)$ 最大的类别标签

后验概率

最小化分类错误率的贝叶斯分类器可通过最大化后验概率决定。然而，在现实任务中这通常难以直接获得。
贝叶斯分类起的目标是基于有限的训练样本集来尽可能准确地估计出后验概率 $P(c|x)$

大体来说，主要有两种策略：

- 给定 x，可通过直接建模 $P(c|x)$ 来预测 c，这样得到的是“判别式模型”
- 先对联合概率分布 $P(x,c)$ 建模，然后再由此获得 $P(c|x)$，这样得到的是“生成式模型”

生成式贝叶斯分类器

前面介绍的决策树、支持向量机等都可归入判别式模型的范畴。而对于生成式模型来说，贝叶斯公式如下：

$$P(c|x) = \frac{P(x,c)}{P(x)} = \frac{P(c)P(x|c)}{P(x)}$$

类先验概率 似然 证据因子

其中，$P(c)$ 是类“先验”概率；$P(x|c)$ 是样本 x 相对于类标记 c 的类条件概率，或称为“似然”；$P(x)$ 是用于归一化的“证据”因子。

贝叶斯决策论

$$P(c|x) = \frac{P(c)P(x|c)}{P(x)}$$

对于给定样本 x，证据因子$P(x)$ 与类标签无关，因此估计 $P(c|x)$ 的问题就转化为如何基于训练数据 D 来估计先验$P(c)$ 和似然$P(x|c)$

类先验概率$P(c)$ 表达了样本空间中各类样本所占的比例。根据大数定律，当训练集包含充足的独立同分布样本时，$P(c)$ 可通过各类样本出现的概率进行估计。

对于类条件概率（似然）$P(x|c)$ 来说，直接用频率来估计是不可行的，因为“未被观测到”与“出现概率为零”通常是不同的。

极大似然估计

估计类条件概率（似然）的一种常用策略是先假定其具有某种确定的概率分布形式，再基于训练样本对概率分布的参数进行估计。

事实上，概率模型的训练过程就是参数估计过程。统计学界的两个学派分别提供了不同的解决方案：

- 频率主义学派认为参数虽然未知，但却是客观存在的固定值。因此，可通过优化似然函数等准则来确定参数值
- 贝叶斯学派则认为参数是未观察到的随机变量，其本身也可有分布。因此，可假定参数服从一个先验分布，然后基于观测到的数据来计算参数的后验分布。

极大似然估计源自频率主义学派，是根据数据采样来估计概率分布参数的经典方法。

极大似然估计

假设 $P(x|c)$ 具有确定的形式并被参数向量 θ_c 唯一确定，将 $P(x|c)$ 记为 $P(x|\theta_c)$。令 D_c 表示训练集 D 中第 c 类样本组成的集合，且假设这些样本是独立同分布的。

极大似然估计

参数 θ_c 对于数据集 D_c 的似然是：

$$P(D_c|\theta_c) = \prod_{x \in D_c} P(x|\theta_c)$$

对 θ_c 进行极大似然估计，就是去寻找能最大化似然 $P(D_c|\theta_c)$ 的参数值 $\hat{\theta}_c$。

极大似然估计

上式中的连乘操作易造成下溢，通常使用对数似然（log-likehood）：

$$\begin{aligned} LL(\theta_c) &= \log P(D_c|\theta_c) \\ &= \sum_{x \in D_c} \log P(x|\theta_c) \end{aligned}$$

此时参数 θ_c 的极大似然估计 $\hat{\theta}_c$ 为：

$$\hat{\theta}_c = \arg\max_{\theta_c} LL(\theta_c)$$

极大似然估计

例如，在连续属性情形下，假设概率密度函数 $p(x|c) \sim \mathcal{N}(\mu_c, \sigma_c^2)$，则参数 μ_c 和 σ_c^2 的极大似然估计为：

$$\hat{\mu}_c = \frac{1}{|D_c|} \sum_{x \in D_c} x$$

$$\hat{\sigma}_c^2 = \frac{1}{|D_c|} \sum_{x \in D_c} (x - \hat{\mu}_c)(x - \hat{\mu}_c)^T$$

也就是说，通过极大似然法得到正态分布均值就是样本均值，方差就是 $(x - \hat{\mu}_c)(x - \hat{\mu}_c)^T$ 的均值，这显然是一个符合直觉的结果。

朴素贝叶斯分类器

最大似然估计的问题

基于贝叶斯公式来估计后验概率 $P(c|x)$ 的主要困难在于：类条件概率 $P(x|c)$ 是所有属性上的联合概率，难以从有限的训练样本直接估计而得到。

朴素贝叶斯分类器 (naive Bayes classifier) 采用了属性条件独立性假设：对已知类别，假设所有属性相互独立。即假设每个属性独立地对分类结果发生影响。

$$P(c|x) = \frac{P(c)P(x|c)}{P(x)} = \frac{P(c)}{P(x)} \prod_{i=1}^{d} P(x_i|c)$$

其中 d 为属性数目，x_i 为 x 在第 i 个属性上的取值。

朴素贝叶斯分类器

由于对于所有类别来说 $P(x)$ 相同，因此对应的贝叶斯判定准则为：

$$h_{nb}(x) = \arg\max_{c \in Y} P(c) \prod_{i=1}^{d} P(x_i|c)$$

这也就是朴素贝叶斯分类器的表达式。

由上式可知，朴素贝叶斯分类器的训练过程就是基于训练集 D 来估计类先验概率 $P(c)$，并为每个属性估计条件概率 $P(x_i|c)$。

朴素贝叶斯分类器

令 D_c 表示训练集 D 中第 c 类样本组成的集合，若有充足的独立同分布样本，则可容易地估计出类先验概率：

$$P(c) = \frac{|D_c|}{|D|}$$

对离散属性而言，令 D_{c,x_i} 表示 D_c 中在第 i 个属性上取值为 x_i 的样本组成的集合，则条件概率 $P(x_i|c)$ 可估计为：

$$P(x_i|c) = \frac{|D_{c,x_i}|}{|D_c|}$$

朴素贝叶斯分类器

而对于连续属性，可以考虑概率密度函数。假定 $p(x_i|c) \sim \mathcal{N}(\mu_{c,i}, \sigma_{c,i}^2)$，其中 $\mu_{c,i}$ 和 $\sigma_{c,i}^2$ 分别为第 c 类样本在第 i 个属性上取值的均值和方差，则有：

$$p(x_i|c) = \frac{1}{\sqrt{2\pi}\sigma_{c,i}} \exp\left(-\frac{(x_i - \mu_{c,i})^2}{2\sigma_{c,i}^2}\right)$$

朴素贝叶斯分类器——以西瓜数据集为例

编号	色泽	根蒂	敲声	纹理	脐部	触感	密度	含糖率	好瓜
1	青绿	蜷缩	浊响	清晰	凹陷	硬滑	0.697	0.460	是
2	乌黑	蜷缩	沉闷	清晰	凹陷	硬滑	0.774	0.376	是
3	乌黑	蜷缩	浊响	清晰	凹陷	硬滑	0.634	0.264	是
4	青绿	蜷缩	沉闷	清晰	凹陷	硬滑	0.608	0.318	是
5	浅白	蜷缩	浊响	清晰	凹陷	硬滑	0.556	0.215	是
6	青绿	稍蜷	浊响	清晰	稍凹	软粘	0.403	0.237	是
7	乌黑	稍蜷	浊响	稍糊	稍凹	软粘	0.481	0.149	是
8	乌黑	稍蜷	浊响	清晰	稍凹	硬滑	0.437	0.211	是
9	乌黑	稍蜷	沉闷	稍糊	稍凹	硬滑	0.666	0.091	否
10	青绿	硬挺	清脆	清晰	平坦	软粘	0.243	0.267	否
11	浅白	硬挺	清脆	模糊	平坦	硬滑	0.245	0.057	否
12	浅白	蜷缩	浊响	模糊	平坦	软粘	0.343	0.099	否
13	青绿	稍蜷	浊响	稍糊	凹陷	硬滑	0.639	0.161	否
14	浅白	稍蜷	沉闷	稍糊	凹陷	硬滑	0.657	0.198	否
15	乌黑	稍蜷	浊响	清晰	稍凹	软粘	0.360	0.370	否
16	浅白	蜷缩	浊响	模糊	平坦	硬滑	0.593	0.042	否
17	青绿	蜷缩	沉闷	稍糊	稍凹	硬滑	0.719	0.103	否

图：训练数据

朴素贝叶斯分类器——以西瓜数据集为例

编号	色泽	根蒂	敲声	纹理	脐部	触感	密度	含糖率	好瓜
测 1	青绿	蜷缩	浊响	清晰	凹陷	硬滑	0.697	0.460	?

图: 测试数据

首先，我们来估计类先验概率 $P(c)$，显然有:

$$P(\text{好瓜} = \text{是}) = \frac{8}{17} \approx 0.471$$

$$P(\text{好瓜} = \text{否}) = \frac{9}{17} \approx 0.529$$

朴素贝叶斯分类器——以西瓜数据集为例

接着，我们为每个属性估计条件概率 $P(x_i|c)$：

$$P_{\text{青绿}|\text{是}} = P(\text{色泽} = \text{青绿} \mid \text{好瓜} = \text{是}) = \frac{3}{8} = 0.375$$

$$P_{\text{青绿}|\text{否}} = P(\text{色泽} = \text{青绿} \mid \text{好瓜} = \text{否}) = \frac{3}{9} \approx 0.333$$

$$P_{\text{蜷缩}|\text{是}} = P(\text{根蒂} = \text{蜷缩} \mid \text{好瓜} = \text{是}) = \frac{5}{8} = 0.625$$

$$P_{\text{蜷缩}|\text{否}} = P(\text{根蒂} = \text{蜷缩} \mid \text{好瓜} = \text{否}) = \frac{3}{9} \approx 0.333$$

$$P_{\text{浊响}|\text{是}} = P(\text{敲声} = \text{浊响} \mid \text{好瓜} = \text{是}) = \frac{6}{8} = 0.750$$

$$P_{\text{浊响}|\text{否}} = P(\text{敲声} = \text{浊响} \mid \text{好瓜} = \text{否}) = \frac{4}{8} \approx 0.444$$

朴素贝叶斯分类器——以西瓜数据集为例

$$P_{\text{清晰}\mid\text{是}} = P(\text{纹理} = \text{清晰} \mid \text{好瓜} = \text{是}) = \frac{7}{8} = 0.875$$

$$P_{\text{清晰}\mid\text{否}} = P(\text{纹理} = \text{清晰} \mid \text{好瓜} = \text{否}) = \frac{2}{9} \approx 0.222$$

$$P_{\text{凹陷}\mid\text{是}} = P(\text{脐部} = \text{凹陷} \mid \text{好瓜} = \text{是}) = \frac{6}{8} = 0.750$$

$$P_{\text{凹陷}\mid\text{否}} = P(\text{脐部} = \text{凹陷} \mid \text{好瓜} = \text{否}) = \frac{2}{9} \approx 0.222$$

$$P_{\text{硬滑}\mid\text{是}} = P(\text{触感} = \text{硬滑} \mid \text{好瓜} = \text{是}) = \frac{6}{8} = 0.750$$

$$P_{\text{硬滑}\mid\text{否}} = P(\text{触感} = \text{硬滑} \mid \text{好瓜} = \text{否}) = \frac{6}{8} \approx 0.667$$

朴素贝叶斯分类器——以西瓜数据集为例

$$
\begin{aligned}
p_{\text{密度:0.697}|\text{ 是}} &= p(\text{密度 } = 0.697 |\ \text{好瓜 } = \text{是}) \\
&= \frac{1}{\sqrt{2\pi}\cdot 0.129} exp\left(-\frac{(0.697-0.574)^2}{2\cdot 0.129^2}\right) \approx 1.959 \\
p_{\text{密度:0.697}|\text{ 否}} &= p(\text{密度 } = 0.697 |\ \text{好瓜 } = \text{否}) \\
&= \frac{1}{\sqrt{2\pi}\cdot 0.195} exp\left(-\frac{(0.697-0.496)^2}{2\cdot 0.195^2}\right) \approx 1.203 \\
p_{\text{含糖:0.460}|\text{ 是}} &= p(\text{含糖 } = 0.460 |\ \text{好瓜 } = \text{是}) \\
&= \frac{1}{\sqrt{2\pi}\cdot 0.101} exp\left(-\frac{(0.460-0.279)^2}{2\cdot 0.101^2}\right) \approx 0.788 \\
p_{\text{含糖:0.460}|\text{ 否}} &= p(\text{含糖 } = 0.460 |\ \text{好瓜 } = \text{否}) \\
&= \frac{1}{\sqrt{2\pi}\cdot 0.108} exp\left(-\frac{(0.460-0.154)^2}{2\cdot 0.108^2}\right) \approx 0.066
\end{aligned}
$$

朴素贝叶斯分类器——以西瓜数据集为例

于是，有

$$P(\text{好瓜} = \text{是}) \times P_{\text{青绿}\,|\,\text{是}} \times P_{\text{蜷缩}\,|\,\text{是}} \times P_{\text{浊响}\,|\,\text{是}} \times P_{\text{清晰}\,|\,\text{是}} \times P_{\text{凹陷}\,|\,\text{是}} \times P_{\text{硬滑}\,|\,\text{是}} \times p_{\text{密度:0.697}|\,\text{是}} \times p_{\text{含糖:0.460}|\,\text{是}} \approx 0.063$$

$$P(\text{好瓜} = \text{否}) \times P_{\text{青绿}\,|\,\text{否}} \times P_{\text{蜷缩}\,|\,\text{否}} \times P_{\text{浊响}\,|\,\text{否}} \times P_{\text{清晰}\,|\,\text{否}} \times P_{\text{凹陷}\,|\,\text{否}} \times P_{\text{硬滑}\,|\,\text{否}} \times p_{\text{密度:0.697}|\,\text{否}} \times p_{\text{含糖:0.460}|\,\text{否}} \approx 6.80 \times 10^{-5}$$

由于 $0.063 > 6.80 \times 10^{-5}$，因此，朴素贝叶斯分类器将测试样本“测 1”判别为“好瓜”。

朴素贝叶斯分类器——拉普拉斯修正

需注意，若某个属性值在训练集中没有与某个类同时出现过，则基于以上公式进行概率估计和判别将会出现问题。

例如，在使用西瓜数据集训练朴素贝叶斯分类器时，对一个“敲声 = 清脆”的测试用例，有

$$P_{\text{清脆}\mid\text{是}} = P(\text{敲声} = \text{清脆} \mid \text{好瓜} = \text{是}) = \frac{0}{8} = 0$$

因为朴素贝叶斯表达式的连乘结果为零，因此，无论该样本的其他属性是什么，哪怕在其他属性上明显像好瓜，分类的结果都将是“好瓜 = 否”，这显然不太合理。

朴素贝叶斯分类器——拉普拉斯修正

为了避免其他属性携带的信息被训练集中未出现的属性值“抹去”，在估计概率值时通常要进行“平滑”，常用“拉普拉斯修正”。

具体来说，令 K 表示训练集 D 中可能的类别数，N_i 表示第 i 个属性可能的取值数，则可将估计概率的公式修正为：

$$\hat{P}(c) = \frac{|D_c| + 1}{|D| + K}$$

$$\hat{P}(x_i|c) = \frac{|D_{c,x_i} + 1|}{|D_c| + N_i}$$

朴素贝叶斯分类器——拉普拉斯修正

例如，在上述例子中，类先验概率可估计为：

$$\hat{P}(\text{好瓜} = \text{是}) = \frac{8+1}{17+2} \approx 0.474, \quad \hat{P}(\text{好瓜} = \text{否}) = \frac{9+1}{17+2} \approx 0.526$$

类似的，$P_{\text{青绿}\mid\text{是}}$ 可估计为：

$$\hat{P}_{\text{青绿}\mid\text{是}} = \hat{P}(\text{色泽} = \text{青绿} \mid \text{好瓜} = \text{是}) = \frac{3+1}{8+3} \approx 0.364$$

同时，上文提到的概率 $P_{\text{清脆}\mid\text{是}}$ 可估计为：

$$\hat{P}_{\text{清脆}\mid\text{是}} = \hat{P}(\text{敲声} = \text{清脆} \mid \text{好瓜} = \text{是}) = \frac{0+1}{8+3} \approx 0.091$$

朴素贝叶斯分类器——拉普拉斯修正

显然，拉普拉斯修正避免了因训练集样本不充分而导致概率估值为零的问题，并且在训练集变大时，修正过程所引入先验的影响也会逐渐变得可忽略，使得估值渐趋向于实际概率值。

实质上，拉普拉斯修正假设了属性值与类别均匀分布，这是在朴素贝叶斯学习过程中额外引入的关于数据的先验。

EM 算法

在前面的课程中，我们一直假设训练样本所有属性变量的值都已被观测到，即训练样本是“完整”的。

但在现实应用中往往会遇到“不完整”的训练样本，例如由于西瓜的根蒂已经脱落，无法看出是“蜷缩”还是“硬挺”，即训练样本的“根蒂”属性变量值未知。

在这种存在“未观测”变量的情形下，是否仍能对模型参数进行估计呢？

EM 算法

未观测变量即“隐变量”。令 X 表示已观测变量集，Z 表示隐变量集，Θ 表示模型参数。欲对 Θ 做极大似然估计，则应最大化对数似然

$$LL(\Theta|X,Z) = \ln P(X,Z|\Theta)$$

然而，由于 Z 是隐变量，上式无法直接求解，此时我们可以通过对 Z 计算期望，来最大化已观测数据的对数“边际似然”

$$LL(\Theta|X) = \ln P(X|\Theta) = \ln \sum_Z P(X,Z|\Theta)$$

EM 算法

隐变量的分布假设

对于每个样本 i，用 $Q_i(z)$ 表示样本 i 隐变量 z 的某种分布，且 $Q_i(z)$ 满足

$$\sum_{z}^{Z} Q_i(z) = 1, \quad Q_i(z) \geq 0$$

原优化目标转化为：

$$\begin{aligned}
\sum_{i}^{n} \log p\left(x_i; \theta\right) &= \sum_{i}^{n} \log \sum_{z} p\left(x_i, z; \theta\right) \\
&= \sum_{i}^{n} \log \sum_{z}^{Z} Q_i(z) \frac{p\left(x_i, z; \theta\right)}{Q_i(z)} \\
&\geq \sum_{i}^{n} \sum_{z}^{Z} Q_i(z) \log \frac{p\left(x_i, z; \theta\right)}{Q_i(z)}
\end{aligned}$$

Jesen 不等式

如果 $f(x)$ 是凹函数，则满足

$$f(E(x)) \geq E(f(x))$$

$\log(x)$ 是常见的凹函数，因此满足：

$$\log \sum_j \lambda_j y_j \geq \sum_j \lambda_j \log y_j, \lambda_j \geq 0, \sum_j \lambda_j = 1$$

下述条件满足时，取到等号：

$$\frac{P\left(x^{(i)}, z^{(i)}; \theta\right)}{Q_i\left(z^{(i)}\right)} = c$$

EM 算法

由于已经对 Q 函数进行了分布的假设，因此满足：

$$Q_i\left(z^{(i)}\right) = \frac{P\left(x^{(i)}, z^{(i)}; \theta\right)}{\sum_z P\left(x^{(i)}, z^{(i)}; \theta\right)} = \frac{P\left(x^{(i)}, z^{(i)}; \theta\right)}{P\left(x^{(i)}; \theta\right)} = P\left(z^{(i)} \mid x^{(i)}; \theta\right)$$

拉升下界的 $Q_i(z^{(i)})$ 就是后验概率，由此知道了在给定 θ 的情况下如何选择最优的 Q(E-step)

在给定 $Q(z)$ 后调整 θ, 从而最大化 $L(\theta)$ 的下界（M-step）

EM 算法

EM（Expectation-Maximization）算法是常用的估计参数隐变量的利器。直译为“期望最大化算法”，通常直接称 EM 算法。

它是一种迭代式的方法，其基本想法是：若参数 Θ 已知，则可根据训练数据判断出最优隐变量 Z 的值（E 步）；反之，若 Z 的值已知，则可方便地对参数 Θ 做极大似然估计（M 步）。

贝叶斯分类器——EM 算法

以初始值 Θ^0 为起点，对于下式，可迭代执行以下步骤直至收敛：

- E 步（Expectation）：基于 Θ^t 推断隐变量 Z 的期望，记为 Z^t
- M 步（Maximization）：基于已观测变量 X 和 Z^t 对参数 Θ 做极大似然估计，记为 Θ^{t+1}

$$LL(\Theta|X) = \ln P(X|\Theta) = \ln \sum_Z P(X, Z|\Theta)$$

这就是 EM 算法的原型。进一步，若我们不是取 Z 的期望，而是基于 Θ^t 计算隐变量 Z 的概率分布 $P(Z|X, \Theta^t)$，则 EM 算法的两个步骤是：

- E 步：以当前参数 Θ^t 推断隐变量分布 $P(Z|X, \Theta^t)$，并计算对数似然 $LL(\Theta|X, Z)$ 关于 Z 的期望

$$Q(\Theta|\Theta^t) = \mathbb{E}_{Z|X,\Theta^t} LL(\Theta|X, Z)$$

- M 步：寻找参数最大化期望似然，即

$$\Theta^{t+1} = \arg\max_{\Theta} Q(\Theta|\Theta^t)$$

贝叶斯分类器——EM 算法

简要来说，EM 算法使用两个步骤交替计算：第一步是期望（E）步，利用当前估计的参数值来计算对数似然的期望值；第二步是最大化（M）步，寻找能使 E 步产生的似然期望最大化的参数值。然后，新得到的参数值重新被用于 E 步……直至收敛到局部最优解。

事实上，隐变量估计问题也可通过梯度下降等优化算法求解，但由于求和的项数将随着隐变量的数目以指数上升，会给梯度计算带来麻烦；而 EM 算法则可看作一种非梯度优化方法。

总结

1. 深入理解 SVM
   - SVM 内容回顾
   - 支持向量
   - 拉格朗日乘子与优化
2. 贝叶斯分类器
   - 贝叶斯
   - 生成式贝叶斯分类器
   - 极大似然估计
   - 拉普拉斯修正
   - EM 算法