1 核方法: 近邻法与支持向量机

1.1 密度估计的非参数法

对于随机变量 $\boldsymbol{x}$, 如果知道其概率密度函数 $p(\boldsymbol{x})$, 就能够计算出其在某一区域 R 内取值的概率. 现在, 我们需要根据 $\boldsymbol{x}$ 的一组采样点 $\mathcal{D}:\{\boldsymbol{x}_n\}_{n=1}^N$ 估计概率密度函数 $p(\boldsymbol{x})$. 这就是**密度估计 (density estimation)** 问题.

定义 1.1 密度估计 给定随机变量 $\boldsymbol{x}$ 的一组采样点 $\mathcal{D}:\{\boldsymbol{x}_n\}_{n=1}^N$, 密度估计是根据 $\mathcal{D}$ 估计随机变量 $\boldsymbol{x}$ 的概率密度函数 $p(\boldsymbol{x})$ 的过程.

密度估计有两类办法:

(1) 参数法: 假定 $p(\boldsymbol{x})$ 具有已知的带参数形式, 那么问题转化为应用最大似然法确定参数的值.

(2) 非参数法: 不对 $p(\boldsymbol{x})$ 作任何假设, 而是直接根据样本 $\mathcal{D}$ 估计 $p(\boldsymbol{x})$.

本节介绍的直方图方法, 核密度估计法和近邻法都属于非参数法.

1.1.1 直方图方法

直方图是我们熟知的表示随机变量分布的方法. 将数据空间划分为若干个小区域, 称为**区间 (bins)** 或**箱 (buckets)**, 然后统计每个区间内的样本点数目, 最后通过归一化得到概率密度函数的估计. 具体地, 设数据空间被划分为 M 个区间 $\{\mathcal{R}_j\}_{j=1}^M$, 每个区间的体积为 V, 则在区间 $\mathcal{R}_j$ 内的概率密度函数估计为

$$p_{\mathcal{R}_j}(\boldsymbol{x})=\frac{1}{NV}\sum_{n=1}^{N}\mathbb{I}(x_n\in\mathcal{R}_j)=\frac{n_j}{NV}$$

其中 n_j 即 $\mathcal{R}_j$ 中的样本点的数目.

直方图的数学原理可以推导如下.

证明. 根据概率密度函数的定义, 随机变量 $\boldsymbol{x}$ 落在某一区域 $\mathcal{R}$ 内的概率为

$$P=\int_{\mathcal{R}}p(\boldsymbol{x})\mathrm{d}\boldsymbol{x}$$

对 $\boldsymbol{x}$ 随机采样 N 次, 有 K 个点落入 $\mathcal{R}$ 的概率服从二项分布:

$$p\left(K|N,P\right)=C_N^KP^K(1-P)^{N-K}$$

如果 N 和 K 都很大, 那么上述二项分布是一个窄峰, 我们就可以近似地认为 $P\approx\dfrac{K}{N}$. 另一方面, 如果区域 $\mathcal{R}$ 足够小, 那么可以认为在该区域内 $p(\boldsymbol{x})$ 近似为常数, 即 $P=p(\boldsymbol{x})V$. 结合上述两式即可得

$$p(\boldsymbol{x})=\frac{K}{NV}$$

这就说明了直方图方法的合理性. □

直方图方法的优点是简单直观, 但缺点也很明显: 结果曲线不光滑, 并且高维空间下将因维度灾难而效果变差.

1.1.2 核密度估计法

直方图对 $p(\boldsymbol{x}) = \dfrac{K}{NV}$ 的处理方法是固定区域 $\mathcal{R}$ 的体积 V, 然后从已知数据中估计 K. 类似地, 为了计算某一 $\boldsymbol{x}$ 处的概率密度 $p(\boldsymbol{x})$, 我们可以将 $\mathcal{R}$ 选择为中心位于 $\boldsymbol{x}$, 边长为 h 的小立方体, 然后定义如下核函数

$$k\left(\frac{\boldsymbol{x}-\boldsymbol{x}_n}{h}\right) = \begin{cases} 1, & \text{if } \forall 1 \leqslant i \leqslant D, \left|\dfrac{x_i - x_{n,i}}{h}\right| < \dfrac{1}{2} \\ 0, & \text{otherwise} \end{cases}$$

其中 x_i 是 $\boldsymbol{x}$ 的第 i 个分量. 则

$$K = \sum_n k\left(\frac{\boldsymbol{x}-\boldsymbol{x}_n}{h}\right)$$

将其代入概率密度函数的估计公式就有

$$p(\boldsymbol{x}) = \frac{1}{N}\sum_n \frac{1}{h^D} k\left(\frac{\boldsymbol{x}-\boldsymbol{x}_n}{h}\right)$$

这样的方法被称作**核密度估计法 (Kernal Density Estimation)**.

我们既可以将上式理解为有多少数据点 $\boldsymbol{x}_n$ 落到以 $\boldsymbol{x}$ 为中心的立方体内, 也可以理解为 $\boldsymbol{x}$ 落到多少个以 $\boldsymbol{x}_n$ 为中心的小立方体里.

与直方图类似, 采用上述核函数给出的 $p(\boldsymbol{x})$ 是不光滑的. 我们可以用光滑的核函数 $k(\boldsymbol{x}, \boldsymbol{x}_n)$, 例如高斯核函数, 来解决这一问题:

$$k(\boldsymbol{x}, \boldsymbol{x}_n) = \frac{1}{(2\pi h^2)^{D/2}} \exp\left(-\frac{|\boldsymbol{x}-\boldsymbol{x}_n|^2}{2h^2}\right)$$

此时估计的 $p(\boldsymbol{x})$ 可以写为

$$p(\boldsymbol{x}) = \frac{1}{N}\sum_n k(\boldsymbol{x}, \boldsymbol{x}_n)$$

1.1.3 近邻法

在数据分布不均匀时, 固定体积 V 的做法可能导致性能下降. 因此, 我们可以采用固定 K 而改变 V 的方法. 这就需要用到近邻法.

尽管近邻法可以用于密度估计, 这时属于无监督学习, 但更多的时候近邻法被用于分类的监督学习, 即 **K-近邻算法 (K-Nearest Neighbor, KNN)**.

K-近邻算法的原理基于 Bayes 公式. 记需要预测类别的点为 $\boldsymbol{x}$, 对于 $\boldsymbol{x}$ 的 K 个近邻数据点 (来自训练集, 因此类别已知), 属于第 k 类的数目记为 K_k, 则第 k 类在 $\boldsymbol{x}$ 处的分布的概率密度为

$$p(\boldsymbol{x}|\mathcal{C}_k) = \frac{K_k}{N_k V}$$

其中 N_k 是训练集中属于第 k 类的数据总数. 属于第 k 类的先验概率为

$$p(\mathcal{C}_k) = \frac{N_k}{N}$$

利用 Bayes 公式得到后验概率

$$p(\mathcal{C}_k|\boldsymbol{x}) = \frac{p(\boldsymbol{x}|\mathcal{C}_k)p(\mathcal{C}_k)}{p(\boldsymbol{x})} = \frac{\frac{K_k}{N_k V}\cdot\frac{N_k}{N}}{\frac{K}{NV}} = \frac{K_k}{K}$$

对于此式的直观理解即在 $\boldsymbol{x}$ 的 K 个临近的样本中, 如果有 K_k 个属于第 k 类, 那么 $\boldsymbol{x}$ 属于第 k 类的概率自然为 $\frac{K_k}{K}$.

K-近邻算法只有一个超参数 K. K 越大, 偏差越大, 方差越小. 它还可以用于回归, 简单的做法是将 $\boldsymbol{x}$ 的 K 个近邻的某一属性的平均值 (或者使用核函数进行加权) 作为对 $\boldsymbol{x}$ 的属性的预测.

1.1.4 非参数法的优缺点

非参数法的最大优点是它不需要假设分布函数的形式, 而是通过数据推断分布函数并进行预测, 保证了估计的无偏性和一致性. 然而, 非参数法也需要大量的数据才能保证良好的效果.

1.2 核方法的主要思想

1.3 支持向量机

支持向量机通过核方法进行非线性分类. 它的主要想法是: 在所有可能的分类超平面中, 选择一个使得分类间隔最大的超平面作为最终的分类超平面.

1.3.1 支持向量机的数学原理

我们用更严谨的语言描述上述问题. 考虑线性分类模型

$$y(\boldsymbol{x}) = \boldsymbol{w}\cdot\boldsymbol{x} + b$$

或使用基函数的模型

$$y(\boldsymbol{x}) = \boldsymbol{w}^{\mathrm{t}}\boldsymbol{\phi}(\boldsymbol{x}) + b$$

决策面为 $y(\boldsymbol{x}) = 0$. 对于前一种情况, 可以看出 $\boldsymbol{w}$ 是垂直于决策面的向量, 单位化后即为 $\hat{\boldsymbol{w}} = \frac{\boldsymbol{w}}{||\boldsymbol{w}||}$. 任意一点 $\boldsymbol{x}$ 在 $\hat{\boldsymbol{w}}$ 上的投影长度为 $\hat{\boldsymbol{w}}\cdot\boldsymbol{x}$. 于是 $\boldsymbol{x}$ 与决策面的距离为

$$d(\boldsymbol{x}) = \hat{\boldsymbol{w}}\cdot\boldsymbol{x} - d_0$$

其中 d_0 是决策面上一点 $\boldsymbol{x}$ 对应的 $d(\boldsymbol{x})$ 值, 也即原点到决策面的距离. 于是有

$$d(\boldsymbol{x}) = \frac{\boldsymbol{w}\cdot\boldsymbol{x}}{||\boldsymbol{w}||} + \frac{b}{||\boldsymbol{w}||} = \frac{y(\boldsymbol{x})}{||w||}$$

记对象的类别标签为 $t_n \in \{-1, 1\}$, 则对于训练样本 $\{\boldsymbol{x}_n, t_n\}_{n=1}^N$, 我们希望所有样本点都被正确分类, 即满足

$$\frac{t_n y(\boldsymbol{x})}{||\boldsymbol{w}||} > 0, \quad n = 1, 2, \ldots, N$$

使用基函数也是类似. 对于线性可分体系, 训练集中的一类数据全部在决策面的一边, 而另一类数据全部在决策面的另一边. 因此对于训练集中任一数据 $\boldsymbol{x}_n$, 其与决策面的距离可以写成

$$\frac{t_n y(\boldsymbol{x})}{||\boldsymbol{w}||} = \frac{t_n \left[\boldsymbol{w}^{\mathrm{t}}\boldsymbol{\phi}(\boldsymbol{x}_n) + b\right]}{||\boldsymbol{w}||}$$

训练集中所有数据点离决策面的最小距离由下式给出:

$$\frac{1}{||\boldsymbol{w}||} \min_n \{t_n[\boldsymbol{w}^{\mathrm{t}}\boldsymbol{\phi}(\boldsymbol{x}_n) + b]\}$$

支持向量机的目标是使得这一间隔最大化, 即

$$\arg\max_{\boldsymbol{w},b} \left\{ \frac{1}{||\boldsymbol{w}||} \min_n \{t_n[\boldsymbol{w}^{\mathrm{t}}\boldsymbol{\phi}(\boldsymbol{x}_n) + b]\} \right\}$$

上式既要求最小值又要求最大值, 比较复杂, 因此要进一步化简. 注意到当 $\boldsymbol{w}$ 和 b 成比例改变时, 距离公式 $\frac{t_n \left[\boldsymbol{w}^{\mathrm{t}}\boldsymbol{\phi}(\boldsymbol{x}_n) + b\right]}{||\boldsymbol{w}||}$ 不会改变, 因此不失一般性地总是可以选取 $\boldsymbol{w}, b$ 使得距离决策面最近的数据点 n 满足 $t_n[\boldsymbol{w}^{\mathrm{t}}\boldsymbol{\phi}(\boldsymbol{x}_n) + b] = 1$, 于是对于任何数据点都有

$$t_n[\boldsymbol{w}^{\mathrm{t}}\boldsymbol{\phi}(\boldsymbol{x}_n) + b] \geqslant 1$$

这被称作决策平面的**规范表示 (Canonical Representation)**. 此时目标简化为

$$\arg\max_{\boldsymbol{w},b} \left\{ \frac{1}{||\boldsymbol{w}||} \right\}$$

或者写成更方便的形式:

$$\arg\min_{\boldsymbol{w},b} \left\{ \frac{1}{2}||\boldsymbol{w}||^2 \right\}$$

这样, 支持向量机的目标就是在 $t_n[\boldsymbol{w}^{\mathrm{t}}\boldsymbol{\phi}(\boldsymbol{x}_n) + b] \geqslant 1$ 的约束下求解上述最小化问题. 这里约束条件是线性的, 目标函数是二次函数, 因此在数学上属于凸优化问题, 具有良好的性质.

不等式约束条件下的函数极值求解可以利用拉格朗日方法和 KKT 条件. 为每个训练集数据点 $\boldsymbol{x}_n$ 的约束条件引入拉格朗日因子 λ_n, 定义如下函数

$$L(\boldsymbol{w}, b, \lambda) = \frac{1}{2}||\boldsymbol{w}||^2 - \sum_n \lambda_n \left\{ t_n[\boldsymbol{w}^{\mathrm{t}}\boldsymbol{\phi}(\boldsymbol{x}_n) + b] - 1 \right\}$$

则上述极值问题的 KKT 条件给出

$$\frac{\partial}{\partial(\boldsymbol{w}, b)} L(\boldsymbol{w}, b, \lambda) = 0, \quad \lambda_n \left\{ t_n[\boldsymbol{w}^{\mathrm{t}}\boldsymbol{\phi}(\boldsymbol{x}_n) + b] - 1 \right\} = 0$$

对 $\boldsymbol{w}$ 的偏导数得出

$$\boldsymbol{w} = \sum_n \lambda_n t_n \boldsymbol{\phi}(\boldsymbol{x}_n)$$

将它代回 $y(\boldsymbol{x}) = \boldsymbol{w}^{\mathrm{t}}\boldsymbol{\phi}(\boldsymbol{x}) + b$ 得到

$$y(vecx) = \sum_{n,i} \lambda_n t_n \phi_i(\boldsymbol{x}_n)\phi_i(\boldsymbol{x}) + b = \sum_n \lambda_n t_n k(\boldsymbol{x}, \boldsymbol{x}_n) + b$$

于是模型的结果与核函数 $k(\boldsymbol{x}, \boldsymbol{x}_n) = \sum_i \phi_i(\boldsymbol{x}_n)\phi_i(\boldsymbol{x}) = \boldsymbol{\phi}^{\mathrm{t}}(\boldsymbol{x}_n)\boldsymbol{\phi}(\boldsymbol{x})$ 有关. 可以看到, 不处于间隔边缘上的点总有 $\lambda_n = 0$, 也即只有间隔边缘的数据点 (被称作**支持向量**) 对 $y(\boldsymbol{x})$ 的计算有贡献.

现在计算 b. 间隔边缘 $\mathcal{S}$ 上的数据点 $\boldsymbol{x}_m$ 满足 $t_m y(\boldsymbol{x}_m) = 1$, 根据前面得出的 $y(\boldsymbol{x})$ 的表达式可得

$$t_m \left[\sum_{n \in \mathcal{S}} \lambda_n t_n k(\boldsymbol{x}_m, \boldsymbol{x}_n) + b \right] = 1$$

于是

$$b = \frac{1}{N_{\mathcal{S}}} \sum_{m \in \mathcal{S}} \left[t_m - \sum_{n \in \mathcal{S}} \lambda_n t_n k(\boldsymbol{x}_m, \boldsymbol{x}_n) \right]$$

其中 $N_{\mathcal{S}}$ 表示落在间隔边缘上的点的数目.

1.3.2 软间隔的使用

在线性不可分体系中, 严格的边界条件就不再适用了. 为此, 我们可以引入**松弛变量 (Slack Variables)**, 记作 $\xi_n \geqslant 0$, 使得约束条件放宽为

$$t_n y(\boldsymbol{x}_n) \geqslant 1 - \xi_n$$

ξ_n 可以看作是数据点越过间隔边界的距离, 即它造成了某种误差, 需要给予相应的惩罚. 由此, 模型的误差函数可以改写成

$$E(\boldsymbol{w}, b, \xi_n; C) = C \sum_n \xi_n + \frac{1}{2}||\boldsymbol{w}||^2$$

其中 C 是超参数, 用于调整对于越界的惩罚力度. 同样可以用拉格朗日方法和 KKT 条件求解上述问题.