曲线拟合之最大斯然估计和最小二乘法

emacsun

目录

之前，我们讨论过关于曲线拟合的一些概念，提到了均方误差函数和贝叶斯估计，并在一篇博文中介绍了 matlab 实现。今天，我们讨论通过最小二乘法对最大斯然估计进行深入的挖掘。

1 问题模型

假设目标变量是 t，其可以通过如下模型生成：

$$t = y(\mathbf{x}, \mathbf{w}) + \epsilon \tag{1.1}$$

其中，ϵ 是高斯白噪，均值为 0，方差是 β。所以我们可以把 t 的概率密度函数表示为：

$$p(t|\mathbf{x}, \mathbf{w}, \beta) = \mathcal{N}(t|y(\mathbf{x}, \mathbf{w}), \beta^{-1}) \tag{1.2}$$

2 问题分析：最大化斯然函数和最小化平方和误差函数的等效性

如果我们的损失函数是均方误差，那么对于一个新的输入 $\mathbf{x}$，最优的预测是基于目标变量的条件均值。针对式 ~(1.2)，我们有：

$$\mathbb{E}[t|\mathbf{x}] = \int tp(t|\mathbf{x})\mathrm{d}t = y(\mathbf{x}, \mathbf{w}) \tag{2.1}$$

注意，高斯白噪的假设使得给定 $\mathbf{x}$ 的 t 的条件均值是各向一致的。

现在考虑输入的数据集合 $\mathbf{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$, 对应的目标变量是 $t_1, \ldots, t_N$。我们假设数据集合是从式 ~(1.2) 所示分布中采样得到的。所以，关于 $\mathbf{X}$ 的最大斯然估计为：

$$p(\mathbf{t}|\mathbf{X}, \mathbf{w}, \beta) = \prod_{n=1}^{N} \mathcal{N}(t_n|\mathbf{w}^T\phi(\mathbf{x}_n), \beta^{-1}) \tag{2.2}$$

注意，这里我们假定

$$y(\mathbf{x}, \mathbf{w}) = \mathbf{w}^T\phi_{x_n}$$

并且，我们不特别的约定基函数 $\phi(\mathbf{x})$ 的形式。在监督学习问题中（分类或者回归），我们的目标不是为输入变量建模。$\mathbf{x}$ 会一直待在条件变量中，所以从现在开始我们去掉 $p(\mathbf{t}|\mathbf{x}, \mathbf{w}, \beta)$ 中的 $\mathbf{x}$。对式 ~(2.2) 求对数，把乘法变成加法，我们有：

$$\ln p(\mathbf{t}|\mathbf{w}, \beta) = \sum_{n=1}^{N} \ln \mathcal{N}(t_n|\mathbf{w}^T\phi(\mathbf{x}_n), \beta^{-1}) \tag{2.3}$$

对式 ~(2.3) 稍作变形，有：

$$\ln p(\mathbf{t}|\mathbf{w}, \beta) = \frac{N}{2}\ln(\beta) - \frac{N}{2}\ln(2\pi) - \beta E_D(\mathbf{w}) \tag{2.4}$$

其中，

$$E_D(\mathbf{w}) = \frac{1}{2}\sum_{n=1}^{N}\{t_n - \mathbf{w}^T\phi(x_n)\}^2 \tag{2.5}$$

我们发现，优化基于高斯白噪的斯然函数和最小化平方和误差函数是等效的。通过对 ~(2.4) 进行求导，有：

$$\nabla \ln p(\mathbf{t}|\mathbf{w}, \beta) = \sum_{n=1}^{N}\{t_n - \mathbf{w}^T\phi(\mathbf{x}_n)\}\phi(\mathbf{x}_n)^T \tag{2.6}$$

令式 ~(2.6) 等于零，

$$0 = \sum_{n=1}^{N} t_n\phi(\mathbf{x}_n)^T - \mathbf{w}^T(\sum_{n=1}^{N}\phi(\mathbf{x}_n)\phi(\mathbf{x}_n)^T) \tag{2.7}$$

继而有：

$$\mathbf{w}_{ML} = (\ {}^T\ )^{-1}\ {}^T\mathbf{t} \tag{2.8}$$

这个解是最小二乘问题的解。这里 是 $N \times M$ 的矩阵：

$$= \begin{bmatrix} \phi_0(\mathbf{x}_1) & \phi_1(\mathbf{x}_1) & \ldots\phi_{M-1}(\mathbf{x}_1) \\ \phi_0(\mathbf{x}_2) & \phi_1(\mathbf{x}_2) & \ldots\phi_{M-1}(\mathbf{x}_2) \\ \vdots & \vdots & \ddots & \vdots \\ \phi_0(\mathbf{x}_N) & \phi_1(\mathbf{x}_N) & \ldots\phi_{M-1}(\mathbf{x}_N) \end{bmatrix} \tag{2.9}$$

其中，$(\ ^T \ )^{-1} \ ^T$ 是 的 Moore-Penrose 伪逆。这个伪逆是逆的推广。当 是方阵且可逆时，这个结果就直接等于 $^{-1}$

此刻，我们再分析 w_0。重写 $E_D(\mathbf{w})$:

$$E_D(\mathbf{w}) = \frac{1}{2}\sum_{n=1}^{N}\{t_n - w_0 - \sum_{j=1}^{M-1} w_j\phi_j(\mathbf{x}_n)\}^2 \tag{2.10}$$

对 w_0 求导，可得：

$$w_0 = \bar{t} - \sum_{j=1}^{M-1} w_j\bar{\phi_j} \tag{2.11}$$

其中：

$$\bar{t} = \frac{1}{N}\sum_{n=1}^{N} t_n, \qquad \bar{\phi_j} = \frac{1}{N}\phi_j(\mathbf{x}_n) \tag{2.12}$$

所以，w_0 补充了训练集合中目标值的均值与基函数之间的差值。

另外，我们可以对式 ~(2.4) 求 β 的导数，得 β:

$$\frac{1}{\beta_{ML}} = \frac{1}{N}\sum_{n=1}^{N}\{t_n - \mathbf{w}_{ML}^T\phi(\mathbf{x}_n)\}^2 \tag{2.13}$$

我们看到噪声精度的倒数是目标值在回归函数周围的方差。

3 最小二乘的几何意义

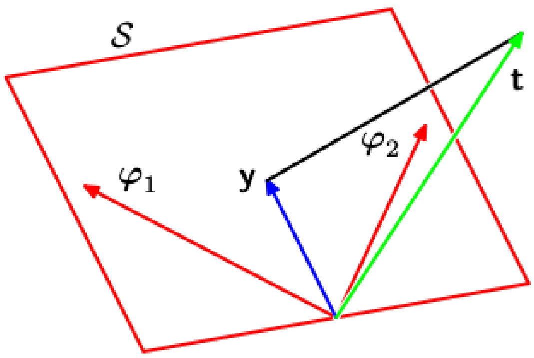

图 1: 最小二乘的几何意义

考虑 N 维空间，$\mathbf{t}$ 是其中一个矢量。每一个基函数 $\phi_j(\mathbf{x}_n)$ 取 N 个训练集合中的值也可以视作一个矢量，标记为 φ_j，如图 1所示。注意 φ_j 对应 的第 j 列。如果基函数的个数 M 小于训练集合的点数 N，那么 M 个矢量 $\phi_j(\mathbf{x}_n)$ 张成

一个 M 维的空间 $\mathcal{S}$。我们定义 $\mathbf{y}$ 是一个 N 维向量其第 n 个坐标为 $y(\mathbf{x}_n, \mathbf{w})$。因为 $\mathbf{y}$ 是 φ_j 的线性组合。所以，$\mathbf{y}$ 可以在 M 维空间 $\mathcal{S}$ 的任意位置。式 ~(2.5) 是 $\mathbf{y}$ 和 $\mathbf{t}$ 的欧几里得距离。所以 $\mathbf{w}$ 的最小二乘解对应着 $\mathcal{S}$ 中距离 $\mathbf{t}$ 最近的 $\mathbf{y}$。从图 1 可以看出这个解对应着 $\mathbf{t}$ 向 $\mathcal{S}$ 的各个坐标系投影。

在实际应用中，直接求解 ^{T} 的逆比较困难（因为，这个矩阵的维度比较大），所以一些数学技巧比如 SVD 分解经常会被用到。