模式识别与机器学习

回归的线性模型

张晓平

武汉大学数学与统计学院

Table of contents

1. 线性基函数模型
   极大似然和最小二乘
   随机梯度下降
   正则化方法
2. 贝叶斯回归
   预测分布
3. 证据近似 (evidence approximation)
   计算证据函数
   最大化证据函数

线性基函数模型

线性基函数模型

$$y(\boldsymbol{x}, \boldsymbol{w}) = \sum_{j=0}^{M-1} w_j \phi_j(\boldsymbol{x}) = \boldsymbol{w}^T \boldsymbol{\phi}(\boldsymbol{x})$$

其中

- $\boldsymbol{w} = (w_0, \cdots, w_{M-1})^T, \boldsymbol{\phi} = (\phi_0, \cdots, \phi_{M-1})^T$, 这里 $\phi_0(\boldsymbol{x}) = 1$, w_0 为偏置参数。
- 一般地，$\boldsymbol{x} \in \mathbb{R}^D$，但为讨论方便，我们设 $x \in \mathbb{R}$.
- 设有观测数据集 $\mathcal{D} = \{\boldsymbol{x}_1, \cdots, \boldsymbol{x}_N\}$，对应的目标变量为 $\mathbf{t} = \{t_1, \cdots, t_N\}$.

线性基函数模型

基函数的选取有

- 多项式

$$\phi_j(x) = x^j$$

- 高斯函数

$$\phi_j(x) = \exp\left(-\frac{(x-\mu_j)}{2s^2}\right)$$

- Sigmoidal 函数

$$\phi_j(x) = \sigma\left(\frac{x-\mu_j}{s}\right)$$

  其中

$$\sigma(a) = \frac{1}{1+e^{-a}}$$

- 样条函数 (splines)、傅里叶函数 (fourier)、小波函数 (wavelet) 等

线性基函数模型

极大似然和最小二乘

极大似然和最小二乘

$$t = y(\boldsymbol{x}, \boldsymbol{w}) + \epsilon$$

其中 ϵ 为高斯噪声，即一个均值为 0、精度（即方差的倒数）为 β 的高斯随机变量。于是有

$$p(t \mid \boldsymbol{x}, \boldsymbol{w}, \beta) = \mathcal{N}(t \mid y(\boldsymbol{x}, \boldsymbol{w}), \beta)$$

设样本集 $\mathcal{D}$ 中的数据独立同分布于上述分布，则其似然函数为

$$p(\mathbf{t} \mid \mathcal{D}, \boldsymbol{w}, \beta) = \prod_{n=1}^{N} \mathcal{N}(t_n \mid y(\boldsymbol{x}_n, \boldsymbol{w}), \beta) = \prod_{n=1}^{N} \mathcal{N}(t_n \mid \phi(\boldsymbol{x}_n)^T \boldsymbol{w}, \beta^{-1})$$

极大似然和最小二乘

对数似然函数为

$$\begin{aligned} \ln p(\mathbf{t} \mid \boldsymbol{w}, \beta) &= \sum_{n=1}^{N} \ln \mathcal{N}(t_n \mid \phi(\boldsymbol{x}_n)^T \boldsymbol{w}, \beta^{-1}) && (1) \\ &= N\left(\frac{1}{2}\ln\beta - \frac{1}{2}\ln(2\pi)\right) - \beta E_{\mathcal{D}}(\boldsymbol{w}), && (2) \end{aligned}$$

其中

$$E_{\mathcal{D}}(\boldsymbol{w}) = \frac{1}{2}\sum_{n=1}^{N}\left[\phi(\boldsymbol{x}_n)^T \boldsymbol{w} - t_n\right]^2, \tag{3}$$

极大似然和最小二乘

首先，我们来计算对数似然函数的梯度

$$\frac{\partial}{\partial \boldsymbol{w}} \ln p(\mathbf{t} \mid \boldsymbol{w}, \beta) = \beta \sum_{n=1}^{N} \phi(\boldsymbol{x}_n) \left[\phi(\boldsymbol{x}_n)^T \boldsymbol{w} - t_n \right]$$

令其为 0 即得

$$\sum_{n=1}^{N} t_n \phi(\boldsymbol{x}_n) = \sum_{n=1}^{N} \phi(\boldsymbol{x}_n) \phi(\boldsymbol{x}_n)^T \boldsymbol{w}$$

极大似然和最小二乘

令

$$\mathbf{\Phi} = \begin{pmatrix} \phi(\boldsymbol{x}_1)^T \\ \vdots \\ \phi(\boldsymbol{x}_N)^T \end{pmatrix} = \begin{pmatrix} \phi_0(\boldsymbol{x}_1) & \cdots & \phi_{M-1}(\boldsymbol{x}_1) \\ \vdots & \ddots & \vdots \\ \phi_0(\boldsymbol{x}_N) & \cdots & \phi_{M-1}(\boldsymbol{x}_N) \end{pmatrix} \in \mathbb{R}^{N\times M},$$

则上式可改写为

$$\mathbf{\Phi}^T\mathbf{\Phi}\boldsymbol{w} = \mathbf{\Phi}^T\mathbf{t}$$

由此可求得 $\boldsymbol{w}$ 的极大似然解为

$$\boldsymbol{w}_{ML} = \left(\mathbf{\Phi}^T\mathbf{\Phi}\right)^{-1}\mathbf{\Phi}^T\mathbf{t} \tag{4}$$

记

$$\mathbf{\Phi}^\dagger = \left(\mathbf{\Phi}^T\mathbf{\Phi}\right)^{-1}\mathbf{\Phi}^T \tag{5}$$

则

$$\boldsymbol{w}_{ML} = \mathbf{\Phi}^\dagger\mathbf{t}. \tag{6}$$

这里，我们称 $\mathbf{\Phi}^\dagger$ 为 $\mathbf{\Phi}$ 的 Moore-Penrose 伪逆矩阵（pseudo-inverse matrix)。

极大似然和最小二乘

然后，我们来计算对数似然函数关于 β 的偏导数：

$$\frac{\partial}{\partial\beta}\ln p(\mathbf{t} \mid \boldsymbol{w}, \beta) = \frac{N}{2}\beta^{-1} - \frac{1}{2}\sum_{n=1}^{N}\left[\phi(\boldsymbol{x}_n)^T\boldsymbol{w} - t_n\right]^2 \tag{7}$$

置其为零，则得 β 的极大似然解为

$$\beta_{ML}^{-1} = \frac{1}{N}\sum_{n=1}^{N}\left[\phi(\boldsymbol{x}_n)^T\boldsymbol{w} - t_n\right]^2. \tag{8}$$

线性基函数模型

随机梯度下降

随机梯度下降

梯度下降法作为机器学习中经常使用的优化算法，其有着三种不同的形式：

- 批量梯度下降（Batch Gradient Descent, BGD）
- 随机梯度下降（Stochastic Gradient Descent, SGD）
- 小批量梯度下降（Mini-Batch Gradient Descent, MBGD）。

随机梯度下降

梯度下降法作为机器学习中经常使用的优化算法，其有着三种不同的形式：

- 批量梯度下降（Batch Gradient Descent, BGD）
- 随机梯度下降（Stochastic Gradient Descent, SGD）
- 小批量梯度下降（Mini-Batch Gradient Descent, MBGD）。

记

$$E_n(\boldsymbol{w}) = \frac{1}{2}\left[\phi(\boldsymbol{x}_n)^T\boldsymbol{w} - t_n\right]^2. \tag{9}$$

故

$$E_{\mathcal{D}}(\boldsymbol{w}) = \frac{1}{2}\sum_{n=1}^{N}\left(\phi(\boldsymbol{x}_n)^T\boldsymbol{w} - t_n\right)^2 = \sum_{n=1}^{N} E_n(\boldsymbol{w}). \tag{10}$$

批量梯度下降

批量梯度下降法是最原始的形式，它是指在每一次迭代时使用所有样本来进行梯度的更新。其步骤为

1. 对目标函数(10)求梯度：

$$\frac{\partial E_{\mathcal{D}}(\mathbf{w})}{\partial \mathbf{w}} = \sum_{n=1}^{N} \phi(\mathbf{x}_n) \left[\phi(\mathbf{x}_n)^T \mathbf{w} - t_n \right]$$

2. 更新参数

$$\begin{aligned} \mathbf{w} &:= \mathbf{w} - \alpha \frac{\partial E_{\mathcal{D}}(\mathbf{w})}{\partial \mathbf{w}} \\ &= \mathbf{w} - \alpha \sum_{n=1}^{N} \phi(\mathbf{x}_n) \left[\phi(\mathbf{x}_n)^T \mathbf{w} - t_n \right] \end{aligned}$$

   其中 α 称为学习率 (learning rate)。

注意：在更新参数时，有一个求和函数，表示对所有样本进行处理。

BGD

- 优点
  - 一次迭代是对所有样本进行计算，此时利用矩阵进行操作，实现了并行。
  - 由全数据集确定的方向能够更好地代表样本总体，从而更准确地朝向极值所在的方向。当目标函数为凸函数时，BGD 一定能够得到全局最优。
- 缺点
  - 当样本数目 N 很大时，每迭代一步都需要对所有样本计算，训练过程会很慢。

BGD

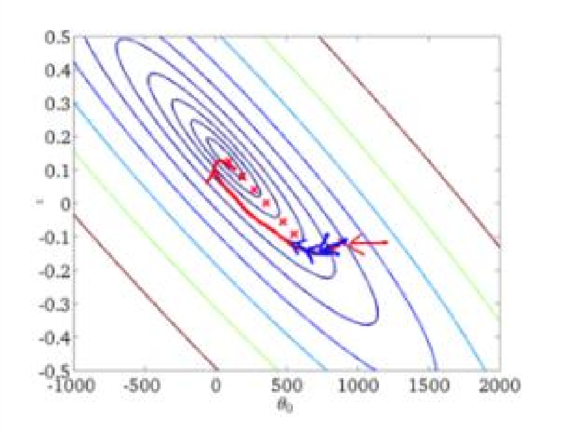

图 1: BGD 的收敛曲线示意图：从迭代的次数来看，BGD 迭代的次数相对较少

SGD

不同于批量梯度下降，随机梯度下降每次迭代使用一个样本来对参数进行更新，从而使得训练速度加快。其步骤为：

1. 对目标函数(9)求梯度

$$\frac{\partial E_n(\boldsymbol{w})}{\partial \boldsymbol{w}} = \phi(\boldsymbol{x}_n)\left[\phi(\boldsymbol{x}_n)^T\boldsymbol{w} - t_n\right]$$

2. 更新参数

$$\begin{aligned}\boldsymbol{w} &:= \boldsymbol{w} - \alpha\frac{\partial E_n(\boldsymbol{w})}{\partial \boldsymbol{w}} \\ &= \boldsymbol{w} - \alpha\phi(\boldsymbol{x}_n)\left[\phi(\boldsymbol{x}_n)^T\boldsymbol{w} - t_n\right]\end{aligned}$$

SGD

- 优点:
  - 由于不是在全部训练数据上的损失函数，而是在每轮迭代中，随机优化某一条训练数据上的损失函数，这样每一轮参数的更新速度大大加快。
- 缺点:
  - 准确度下降。由于即使在目标函数为凸函数的情况下，SGD 仍旧无法做到线性收敛。
  - 可能会收敛到局部最优，由于单个样本并不能代表全体样本的趋势。
  - 不易于并行实现。

SGD

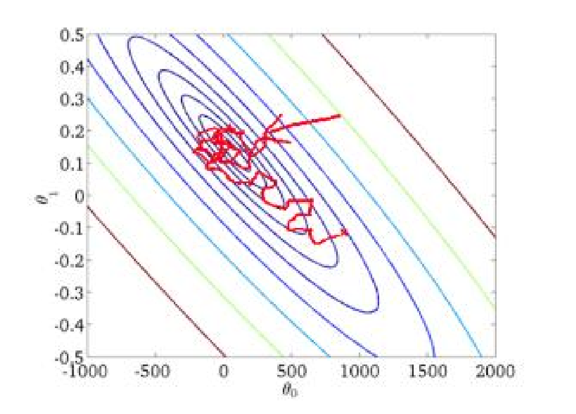

图 2: SGD 的收敛曲线示意图：迭代的次数上来看，SGD 迭代的次数较多，在解空间的搜索过程看起来很盲目

小批量梯度下降

小批量梯度下降是对批量梯度下降以及随机梯度下降的一个折中办法，其思想是：每次迭代使用 $batch\_size$ 个样本来对参数进行更新。设样本数为 N，每次迭代时参与的样本的个数 $batch\_size$，则在训练过程中，可将样本集 $\mathcal{D}$ 随机划分为 $\mathcal{D}_1, \cdots, \mathcal{D}_{N/batch\_size}$（无放回），其指标集分别记为 $\mathcal{I}_1, \cdots, \mathcal{I}_{N/batch\_size}$。其步骤为：

1. 对目标函数

$$E_{\mathcal{D}_i} = \frac{1}{2} \sum_{n \in \mathcal{I}_i} \left(t_n - \phi(\mathbf{x}_n)^T \mathbf{w} \right)^2$$

求梯度：

$$\frac{\partial E_{\mathcal{D}_i}(\mathbf{w})}{\partial \mathbf{w}} = \phi(\mathbf{x}_n) \sum_{n \in \mathcal{I}_i} \left[\phi(\mathbf{x}_n)^T \mathbf{w} - t_n \right]$$

2. 更新参数

$$\begin{aligned} \mathbf{w} &:= \mathbf{w} - \alpha \frac{\partial E_n(\mathbf{w})}{\partial \mathbf{w}} \\ &= \mathbf{w} - \alpha \sum_{n \in \mathcal{I}_i} \phi(\mathbf{x}_n) \left[\phi(\mathbf{x}_n)^T \mathbf{w} - t_n \right]. \end{aligned}$$

MBGD

- 优点：
  - 通过矩阵运算，每次在一个 batch 上优化神经网络参数并不会比单个数据慢太多。
  - 每次使用一个 batch 可以大大减小收敛所需要的迭代次数，同时可以使收敛到的结果更加接近梯度下降的效果。
  - 可实现并行化。
- 缺点：
  - *batch_size* 的选择不当可能会带来一些问题。

MBGD

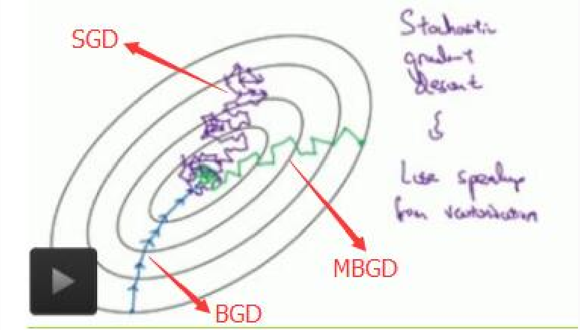

图 3: 三种梯度下降算法的收敛曲线示意图

随机梯度下降

几个重要概念：

- epoch：训练数据集中所有的样本数据被计算一次，称为一个 epoch
- batch size：每次更新参数所使用的样本的个数
- iteration：参数更新一次，就是一次 iteration

随机梯度下降

几个重要概念:

- epoch: 训练数据集中所有的样本数据被计算一次，称为一个 epoch
- batch size: 每次更新参数所使用的样本的个数
- iteration: 参数更新一次，就是一次 iteration

batch_size	梯度下降法	iteration
100	BGD	1
50	BSGD	2
10	BSGD	10
1	SGD	100

表 1: 设训练数据集中样本个数为 $N = 100$，考虑一个 epoch

线性基函数模型

正则化方法

正则化方法

第一章已经介绍过，可为误差函数添加正则化项来控制最小二乘法的过拟合，即

$$E_{\mathcal{D}}(\boldsymbol{w}) + \lambda E_{\mathcal{R}}(\boldsymbol{w}), \tag{11}$$

其中 λ 为正则化系数，用于控制数据项 $\mathcal{E}_{\mathcal{D}}(\boldsymbol{w})$ 和正则项 $\mathcal{E}_{\mathcal{R}}(\boldsymbol{w})$ 的相对重要性。考虑

$$E_{\mathcal{D}}(\boldsymbol{w}) = \frac{1}{2}\sum_{n=1}^{N}\left[\phi(\boldsymbol{x}_n)^T\boldsymbol{w} - t_n\right]^2,$$

正则化项 $\mathcal{E}_{\mathcal{R}}(\boldsymbol{w})$ 的常见选择有

- Ridge 回归

$$E_{\mathcal{R}}(\boldsymbol{w}) = \frac{1}{2}\|\boldsymbol{w}\|_2^2 \tag{12}$$

- Lasso 回归

$$E_{\mathcal{R}}(\boldsymbol{w}) = \|\boldsymbol{w}\|_1 \tag{13}$$

Ridge 回归

误差函数为

$$E(\boldsymbol{w}) = \frac{1}{2}\sum_{n=1}^{N}\left[\phi(\boldsymbol{x}_n)^T\boldsymbol{w} - t_n\right]^2 + \frac{\lambda}{2}\|\boldsymbol{w}\|_2^2 \tag{14}$$

Ridge 回归：法方程组

对应的极小值问题有解析解，可通过如下步骤求得

- 计算梯度

$$\frac{\partial E(\boldsymbol{w})}{\partial \boldsymbol{w}} = \sum_{n=1}^{N} \phi(\boldsymbol{x}_n) \left[\phi(\boldsymbol{x}_n)^T \boldsymbol{w} - t_n\right] + \lambda \boldsymbol{w}$$

- 置其为零可得

$$\left[\sum_{n=1}^{N} \phi(\boldsymbol{x}_n)\phi(\boldsymbol{x}_n)^T + \lambda \boldsymbol{I}\right] \boldsymbol{w} = \sum_{n=1}^{N} t_n \phi(\boldsymbol{x}_n)^T$$

即

$$\left(\boldsymbol{\Phi}^T \boldsymbol{\Phi} + \lambda \boldsymbol{I}\right) \boldsymbol{w} = \boldsymbol{\Phi}^T \mathbf{t} \tag{15}$$

于是

$$\boldsymbol{w} = \left(\boldsymbol{\Phi}^T \boldsymbol{\Phi} + \lambda \boldsymbol{I}\right)^{-1} \boldsymbol{\Phi}^T \mathbf{t}$$

Ridge 回归：梯度下降法

这里只介绍 BGD，其步骤为

- 计算梯度

$$\frac{\partial E(\boldsymbol{w})}{\partial \boldsymbol{w}} = \sum_{n=1}^{N} \phi(\boldsymbol{x}_n) \left(\phi(\boldsymbol{x}_n)^T \boldsymbol{w} - t_n \right) + \lambda \boldsymbol{w}$$

- 更新参数

$$\begin{aligned} \boldsymbol{w} &:= \boldsymbol{w} - \alpha \left\{ \sum_{n=1}^{N} \phi(\boldsymbol{x}_n) \left[\phi(\boldsymbol{x}_n)^T \boldsymbol{w} - t_n \right] + \lambda \boldsymbol{w} \right\} \\ &= (1 - \alpha\lambda)\, \boldsymbol{w} - \alpha \sum_{n=1}^{N} \phi(\boldsymbol{x}_n) \left[\phi(\boldsymbol{x}_n)^T \boldsymbol{w} - t_n \right] \end{aligned}$$

Lasso 回归

误差函数为

$$E(\boldsymbol{w}) = \frac{1}{2}\sum_{n=1}^{N}\left(\phi(\boldsymbol{x}_n)^T\boldsymbol{w} - t_n\right)^2 + \lambda\|\boldsymbol{w}\|_1 \tag{16}$$

由于正则化项使用的是 L_1 范数，导致 Lasso 回归的目标函数不是连续可导的，也就是说，最小二乘法、梯度下降法、牛顿法、拟牛顿法等都不能用。L_1 正则化问题可用近端梯度下降法 (Proximal Gradient Descent，PGD)。

Lasso 回归

考虑优化问题

$$\min_{\boldsymbol{x}} f(\boldsymbol{x}) + \lambda \|\boldsymbol{x}\|_1 \tag{17}$$

其中 $f(\boldsymbol{x})$ 可导，梯度 $\nabla f(\boldsymbol{x})$ 满足 Lipschitz 条件，即存在常数 $L > 0$ 使得

$$\frac{\|\nabla f(\boldsymbol{x}') - \nabla f(\boldsymbol{x})\|}{\|\boldsymbol{x}' - \boldsymbol{x}\|} \leq L, \quad \forall \boldsymbol{x}, \boldsymbol{x}'$$

将 $f(\boldsymbol{x})$ 在 $\boldsymbol{x} = \boldsymbol{x}_k$ 处做二阶泰勒展开，得

$$f(\boldsymbol{x}) = f(\boldsymbol{x}_k) + \nabla f(\boldsymbol{x}_k)^T(\boldsymbol{x} - \boldsymbol{x}_k) + \frac{1}{2}(\boldsymbol{x} - \boldsymbol{x}_k)^T \boldsymbol{H}(\boldsymbol{x} - \boldsymbol{x}_k) \tag{18}$$

其中 $\boldsymbol{H}$ 为 $f(\boldsymbol{x})$ 的 Hessian 矩阵。

Lasso 回归

用 $L\boldsymbol{I}$ 近似 Hessian 矩阵 $\boldsymbol{H}$，得

$$
\begin{aligned}
f(\boldsymbol{x}) &\approx f(\boldsymbol{x}_k) + \nabla f(\boldsymbol{x}_k)^T(\boldsymbol{x}-\boldsymbol{x}_k) + \frac{L}{2}(\boldsymbol{x}-\boldsymbol{x}_k)^T(\boldsymbol{x}-\boldsymbol{x}_k) \\
&= \frac{L}{2}\left[\|\boldsymbol{x}-\boldsymbol{x}_k\|^2 + \frac{2}{L}\nabla f(\boldsymbol{x}_k)^T(\boldsymbol{x}-\boldsymbol{x}_k) + \frac{1}{L^2}\|\nabla f(\boldsymbol{x}_k)\|_2^2\right] \\
&\qquad\qquad + f(\boldsymbol{x}_k) - \frac{1}{2L}\|\nabla f(\boldsymbol{x}_k)\|_2^2 \\
&= \frac{L}{2}\left\|\boldsymbol{x} - \left(\boldsymbol{x}_k - \frac{1}{L}\nabla f(\boldsymbol{x}_k)\right)\right\|^2 + \varphi(\boldsymbol{x}_k)
\end{aligned}
\tag{19}
$$

其中 $\varphi(\boldsymbol{x}_k) = \varphi(\boldsymbol{x}_k) - \frac{1}{2L}\|\nabla f(\boldsymbol{x}_k)\|_2^2$ 与 $\boldsymbol{x}$ 无关。

Lasso 回归

因此，

$$\boldsymbol{x}_{k+1} = \arg\min_{\boldsymbol{x}} \left\{ \frac{L}{2} \left\| \boldsymbol{x} - \left(\boldsymbol{x}_k - \frac{1}{L} \nabla f(\boldsymbol{x}_k) \right) \right\|^2 + \lambda \|\boldsymbol{x}\|_1 \right\}.$$

上述优化问题可分为两步

- 计算

$$z = \boldsymbol{x}_k - \frac{1}{L} \nabla f(\boldsymbol{x}_k)$$

- 求解优化问题

$$\boldsymbol{x}_{k+1} = \arg\min_{\boldsymbol{x}} \left\{ \frac{L}{2} \|\boldsymbol{x} - \boldsymbol{z}\|^2 + \lambda \|\boldsymbol{x}\|_1 \right\}. \tag{20}$$

Lasso 回归

由于

$$
\begin{aligned}
\min_{\boldsymbol{x}} \left\{ \frac{L}{2} \left\|\boldsymbol{x} - \boldsymbol{z}\right\|^2 + \lambda \|\boldsymbol{x}\|_1 \right\} &= \min_{x_1} \left\{ \frac{L}{2} \left(x^1 - z^1\right)^2 + \lambda |x^1| \right\} \\
&+ \cdots \\
&+ \min_{x_D} \left\{ \frac{L}{2} \left(x^D - z^D\right)^2 + \lambda |x^D| \right\}
\end{aligned}
$$

从而优化问题(20)可转换为求解 D 个独立的单变量优化问题。

Lasso 回归

定理

优化问题

$$\min_{x} \ \frac{1}{2}(x-z)^2 + \lambda|x|$$

的解为

$$x^* = \mathrm{soft}(z,\lambda) = \begin{cases} z+\lambda, & z < -\lambda \\ 0 & |z| \leq \lambda \\ z-\lambda, & z > \lambda, \end{cases}, \tag{21}$$

其中 $\mathrm{soft}(z,\lambda)$ 称为软阈值函数 (soft thresholding)。

证明: 令 $f(x) = \frac{1}{2}(x - z)^2 + \lambda|x|$，求导数得

$$f'(x) = (x - z) + \lambda \mathrm{sign}(x), \quad \forall x \neq 0$$

分三种情况讨论:

- $z < -\lambda$
- $z > \lambda$
- $-\lambda \leq z \leq \lambda$

Lasso 回归

情形 1：$z < -\lambda$

- 当 $x > 0$ 时，$\mathrm{sign}(x) = 1$，$f'(x) = x - (z - \lambda) > 0$，故 f 在 $(0, \infty)$ 单调递增，不可能取得极小值。
- 当 $x < 0$ 时，$\mathrm{sign}(x) = -1$，$f'(x) = x - (z + \lambda), f'' = 1 > 0$。因此，$f$ 的极小值在 $x = z + \lambda$ 处取得。

Lasso 回归

情形 2：$z > \lambda$

- 当 $x < 0$ 时，$\mathrm{sign}(x) = -1$，$f'(x) = x - (z + \lambda) < 0$，故 f 在 $(-\infty, 0)$ 单调递减，不可能取得极小值。
- 当 $x > 0$ 时，$\mathrm{sign}(x) = 1$，$f'(x) = x - (z - \lambda), f'' = 1 > 0$。因此，$f$ 的极小值在 $x = z - \lambda$ 处取得。

情形 3：$-\lambda \le z \le \lambda$

- 当 $x < 0$ 时，$\mathrm{sign}(x) = -1$，$f'(x) = x - (z + \lambda) < 0$，故 f 在 $(-\infty, 0)$ 单调递减，不可能取得极小值。
- 当 $x > 0$ 时，$\mathrm{sign}(x) = 1$，$f'(x) = x - (z - \lambda) > 0$，故 f 在 $(-\infty, 0)$ 单调递增，不可能取得极小值。

那么 $\boldsymbol{x} = 0$ 是否为 $f(\boldsymbol{x})$ 的极值点呢？

Lasso 回归

$\forall \Delta x \neq 0$,

$$f(\Delta x) = \frac{1}{2}(\Delta x - z)^2 + \lambda|\Delta x| = \frac{1}{2}(\Delta x)^2 - \Delta x z + \lambda|\Delta x| + f(0)$$

- 当 $\Delta x > 0$ 时，由 $z < \lambda$ 知

$$f(\Delta x) > \frac{1}{2}(\Delta x)^2 - \lambda\Delta x + \lambda\Delta x + f(0) > f(0),$$

- 当 $\Delta x < 0$ 时，由 $z > -\lambda$ 知

$$f(\Delta x) > \frac{1}{2}(\Delta x)^2 + 2\lambda|\Delta x| + f(0) > f(0),$$

因此，$f(x)$ 在 $x = 0$ 处取得极小值。

$\forall \Delta x \neq 0$,

$$f(\Delta x) = \frac{1}{2}(\Delta x - z)^2 + \lambda|\Delta x| = \frac{1}{2}(\Delta x)^2 - \Delta x z + \lambda|\Delta x| + f(0)$$

- 当 $\Delta x > 0$ 时，由 $z < \lambda$ 知

$$f(\Delta x) > \frac{1}{2}(\Delta x)^2 - \lambda\Delta x + \lambda\Delta x + f(0) > f(0),$$

- 当 $\Delta x < 0$ 时，由 $z > -\lambda$ 知

$$f(\Delta x) > \frac{1}{2}(\Delta x)^2 + 2\lambda|\Delta x| + f(0) > f(0),$$

因此，$f(x)$ 在 $x = 0$ 处取得极小值。

综合以上三种情况知，(21)成立。

再回到优化问题(20)，由上述定理可知，它的解为

$$x_{k+1}^i = \begin{cases} z^i + \frac{\lambda}{L}, & z^i < -\frac{\lambda}{L}, \\ 0, & -\frac{\lambda}{L} < z^i < \frac{\lambda}{L}, \\ z^i - \frac{\lambda}{L}, & z^i > \frac{\lambda}{L}. \end{cases}$$

贝叶斯回归

贝叶斯回归

引入先验 (prior) 的回归/分类，或者说 MAP estimator(最大后验估计) 不能算是贝叶斯方法。完整的贝叶斯方法并不止于算出后验分布 (posterior distribution) 的均值，而时利用整个后验分布对预测过程进行平滑。

贝叶斯回归

预测分布

预测分布

定义：预测分布

$$p(t \mid \boldsymbol{x}, \mathbf{t}, \alpha, \beta) = \int p(t \mid \boldsymbol{x}, \boldsymbol{w}, \beta) \cdot p(\boldsymbol{w} \mid \mathbf{t}, \alpha, \beta)\, d\boldsymbol{w} \tag{22}$$

其中

- 条件概率 $p(t \mid \boldsymbol{x}, \boldsymbol{w}, \beta)$ 为

$$p(t \mid \boldsymbol{x}, \boldsymbol{w}, \beta) = \frac{\beta^{1/2}}{(2\pi)^{1/2}} \exp\left\{-\frac{\beta}{2}\left[t - \phi^T \boldsymbol{w}\right]^2\right\}$$

- $p(\boldsymbol{w} \mid \mathbf{t}, \alpha, \beta)$ 为后验分布，即

$$p(\boldsymbol{w} \mid \mathbf{t}, \alpha, \beta) = C \exp\left\{-\frac{1}{2}(\boldsymbol{w} - \boldsymbol{m}_N)^T \boldsymbol{S}_N^{-1}(\boldsymbol{w} - \boldsymbol{m}_N)\right\}.$$

$$C = \frac{1}{(2\pi)^{M/2}|\boldsymbol{S}_N|^{1/2}}, \quad \boldsymbol{S}_N^{-1} = \alpha \boldsymbol{I} + \beta \boldsymbol{\Phi}^T \boldsymbol{\Phi}, \quad \boldsymbol{m}_N = \beta \boldsymbol{S}_N \boldsymbol{\Phi}^T \mathbf{t}$$

预测分布

如何理解预测分布

$$p(t \mid \mathbf{x}, \mathbf{t}, \alpha, \beta) = \int p(t \mid \mathbf{x}, \mathbf{w}, \beta) \cdot p(\mathbf{w} \mid \mathbf{t}, \alpha, \beta) \, d\mathbf{w}$$

预测分布

如何理解预测分布

$$p(t \mid \mathbf{x}, \mathbf{t}, \alpha, \beta) = \int p(t \mid \mathbf{x}, \mathbf{w}, \beta) \cdot p(\mathbf{w} \mid \mathbf{t}, \alpha, \beta)\, d\mathbf{w}$$

可理解为将不同 $\mathbf{w}$ 对应的预测结果组合起来，形成最终的预测结果，而组合的权重由 $\mathbf{w}$ 的后验分布决定。由于 $\mathbf{w}$ 是一个连续的随机变量，故这个组合就是一个积分。

预测分布

回过来看 MAP，它能降低过拟合，但不能避免过拟合，因为 MAP 假定参数只会取一个固定的值，而不是一个分布，这是过度自信的表现。更具体来说，MAP 将预测分布

$$p(t \mid \boldsymbol{x}, \mathbf{t}, \alpha, \beta) = \int p(t \mid \boldsymbol{x}, \boldsymbol{w}, \beta) \cdot p(\boldsymbol{w} \mid \mathbf{t}, \alpha, \beta)\, d\boldsymbol{w}$$

中的后验分布 $p(\boldsymbol{w} \mid \mathbf{t}, \alpha, \beta)$ 近似为一个 delta 函数 $\delta(\boldsymbol{w} - \hat{\boldsymbol{w}})$，从而忽略了 $\boldsymbol{w}$ 的不确定性。

预测分布

下面来推导预测分布

$$p(t \mid \boldsymbol{x}, \mathbf{t}, \alpha, \beta) = \int p(t \mid \boldsymbol{x}, \boldsymbol{w}, \beta) \cdot p(\boldsymbol{w} \mid \mathbf{t}, \alpha, \beta) \, d\boldsymbol{w}$$

的具体形式。

预测分布

下面来推导预测分布

$$p(t \mid \boldsymbol{x}, \mathbf{t}, \alpha, \beta) = \int p(t \mid \boldsymbol{x}, \boldsymbol{w}, \beta) \cdot p(\boldsymbol{w} \mid \mathbf{t}, \alpha, \beta)\, d\boldsymbol{w}$$

的具体形式。

$$\begin{aligned}
& p(t \mid \boldsymbol{x}, \boldsymbol{w}, \beta) \cdot p(\boldsymbol{w} \mid \mathbf{t}, \alpha, \beta) \\
\propto \quad & \exp\left\{-\frac{\beta}{2}\left[t^2 - 2t\phi^T\boldsymbol{w} + \boldsymbol{w}^T\phi\phi^T\boldsymbol{w}\right]\right\} \\
& \qquad \cdot \exp\left\{-\frac{1}{2}\left[\boldsymbol{w}^T\boldsymbol{S}_N^{-1}\boldsymbol{w} - 2\boldsymbol{w}^T\boldsymbol{S}_N^{-1}\boldsymbol{m}_N + \boldsymbol{m}_N\boldsymbol{S}_N^{-1}\boldsymbol{m}_N\right]\right\} \\
\propto \quad & \exp\left\{-\frac{1}{2}\left[\boldsymbol{w}^T\underbrace{\left(\boldsymbol{S}_N^{-1} + \beta\phi\phi^T\right)}_{\boldsymbol{S}^{-1}}\boldsymbol{w} - 2\boldsymbol{w}^T\underbrace{\left(\boldsymbol{S}_N^{-1}\boldsymbol{m}_N + \beta\phi t\right)}_{\boldsymbol{m}}\right] - \frac{\beta}{2}t^2\right\}
\end{aligned}$$

预测分布

$$\boldsymbol{S}^{-1} = \boldsymbol{S}_N^{-1} + \beta\boldsymbol{\phi}\boldsymbol{\phi}^T, \quad \boldsymbol{m} = \boldsymbol{S}_N^{-1}\boldsymbol{m}_N + \beta\boldsymbol{\phi}t \tag{23}$$

预测分布

$$\boldsymbol{S}^{-1} = \boldsymbol{S}_N^{-1} + \beta\boldsymbol{\phi}\boldsymbol{\phi}^T, \quad \boldsymbol{m} = \boldsymbol{S}_N^{-1}\boldsymbol{m}_N + \beta\boldsymbol{\phi}t \tag{23}$$

于是，

$$\begin{aligned}
& p(t \mid \boldsymbol{x}, \boldsymbol{w}, \beta) \cdot p(\boldsymbol{w} \mid \mathbf{t}, \alpha, \beta) \\
\propto \quad & \exp\left\{-\frac{1}{2}\left[\boldsymbol{w}^T\boldsymbol{S}^{-1}\boldsymbol{w} - 2\boldsymbol{w}^T\boldsymbol{m}\right] - \frac{\beta}{2}t^2\right\} \\
\propto \quad & \exp\left\{-\frac{1}{2}(\boldsymbol{w} - \boldsymbol{S}\boldsymbol{m})^T\boldsymbol{S}^{-1}(\boldsymbol{w} - \boldsymbol{S}\boldsymbol{m}) + \frac{1}{2}\boldsymbol{m}^T\boldsymbol{S}\boldsymbol{m} - \frac{\beta}{2}t^2\right\}
\end{aligned}$$

预测分布

而

$$\int \exp\left\{-\frac{1}{2}(\boldsymbol{w}-\boldsymbol{Sm})^T\boldsymbol{S}^{-1}(\boldsymbol{w}-\boldsymbol{Sm})\right\}\,d\boldsymbol{w} = (2\pi)^{M/2}|\boldsymbol{S}|^{1/2}$$

故

$$\begin{aligned}
& \int p(t \mid \boldsymbol{x}, \boldsymbol{w}, \beta)\cdot p(\boldsymbol{w} \mid \mathbf{t}, \alpha, \beta)\,d\boldsymbol{w} \\
\propto \quad & \exp\left\{\frac{1}{2}\boldsymbol{m}^T\boldsymbol{S}\boldsymbol{m} - \frac{\beta}{2}t^2\right\} \\
\propto \quad & \exp\left\{\frac{1}{2}\left(\boldsymbol{S}_N^{-1}\boldsymbol{m}_N + \beta\phi t\right)^T \boldsymbol{S}\left(\boldsymbol{S}_N^{-1}\boldsymbol{m}_N + \beta\phi t\right) - \frac{\beta}{2}t^2\right\} \\
\propto \quad & \exp\left\{-\frac{\beta}{2}\left[\left(1-\beta\phi^T\boldsymbol{S}\phi\right)t^2 - 2\phi^T\boldsymbol{S}\boldsymbol{S}_N^{-1}\boldsymbol{m}_N t\right]\right\}
\end{aligned}$$

预测分布

- 对 $\boldsymbol{S}^{-1} = \boldsymbol{S}_N^{-1} + \beta\boldsymbol{\phi}\boldsymbol{\phi}^T$ 两端左乘 $\phi^T\boldsymbol{S}$，可得

$$\phi^T = \phi^T\boldsymbol{S}\boldsymbol{S}_N^{-1} + \beta\phi^T\boldsymbol{S}\phi\phi^T$$

预测分布

- 对 $\boldsymbol{S}^{-1} = \boldsymbol{S}_N^{-1} + \beta\boldsymbol{\phi}\boldsymbol{\phi}^T$ 两端左乘 $\phi^T\boldsymbol{S}$，可得

$$\phi^T = \phi^T\boldsymbol{S}\boldsymbol{S}_N^{-1} + \beta\phi^T\boldsymbol{S}\phi\phi^T \tag{24}$$

亦即

$$\phi^T\boldsymbol{S}\boldsymbol{S}_N^{-1} = \left(1 - \beta\phi^T\boldsymbol{S}\phi\right)\phi^T$$

预测分布

- 对 $\boldsymbol{S}^{-1} = \boldsymbol{S}_N^{-1} + \beta\boldsymbol{\phi}\boldsymbol{\phi}^T$ 两端左乘 $\phi^T\boldsymbol{S}$，可得

$$\phi^T = \phi^T\boldsymbol{S}\boldsymbol{S}_N^{-1} + \beta\phi^T\boldsymbol{S}\phi\phi^T \tag{24}$$

亦即

$$\phi^T\boldsymbol{S}\boldsymbol{S}_N^{-1} = \left(1 - \beta\phi^T\boldsymbol{S}\phi\right)\phi^T$$

- 对(24)两端同时右乘 $\boldsymbol{S}_N\phi$，可得

$$\phi^T\boldsymbol{S}_N\phi = \phi^T\boldsymbol{S}\phi + \beta\cdot\phi^T\boldsymbol{S}\phi\cdot\phi^T\boldsymbol{S}_N\phi$$

从而有

$$\phi^T\boldsymbol{S}\phi = \frac{\phi^T\boldsymbol{S}_N\phi}{1 + \beta\phi^T\boldsymbol{S}_N\phi}$$

预测分布

- 对 $\boldsymbol{S}^{-1} = \boldsymbol{S}_N^{-1} + \beta\boldsymbol{\phi}\boldsymbol{\phi}^T$ 两端左乘 $\phi^T\boldsymbol{S}$，可得

$$\phi^T = \phi^T\boldsymbol{S}\boldsymbol{S}_N^{-1} + \beta\phi^T\boldsymbol{S}\phi\phi^T \tag{24}$$

亦即

$$\phi^T\boldsymbol{S}\boldsymbol{S}_N^{-1} = \left(1 - \beta\phi^T\boldsymbol{S}\phi\right)\phi^T$$

- 对(24)两端同时右乘 $\boldsymbol{S}_N\phi$，可得

$$\phi^T\boldsymbol{S}_N\phi = \phi^T\boldsymbol{S}\phi + \beta\cdot\phi^T\boldsymbol{S}\phi\cdot\phi^T\boldsymbol{S}_N\phi$$

从而有

$$\phi^T\boldsymbol{S}\phi = \frac{\phi^T\boldsymbol{S}_N\phi}{1+\beta\phi^T\boldsymbol{S}_N\phi} \quad \Rightarrow \quad 1 - \beta\phi^T\boldsymbol{S}\phi = \frac{1}{1+\beta\phi^T\boldsymbol{S}_N\phi}$$

预测分布

因此

$$
\begin{aligned}
& \int p(t \mid \boldsymbol{x}, \boldsymbol{w}, \beta) \cdot p(\boldsymbol{w} \mid \mathbf{t}, \alpha, \beta) \, d\boldsymbol{w} \\
\propto \quad & \exp\left\{-\frac{\beta}{2}\left[\left(1 - \beta\boldsymbol{\phi}^T\boldsymbol{S}\boldsymbol{\phi}\right)t^2 - 2\left(1 - \beta\boldsymbol{\phi}^T\boldsymbol{S}\boldsymbol{\phi}\right)\boldsymbol{\phi}^T\boldsymbol{m}_N t\right]\right\} \\
\propto \quad & \exp\left\{-\frac{\beta\left(1 - \beta\boldsymbol{\phi}^T\boldsymbol{S}\boldsymbol{\phi}\right)}{2}\left(t^2 - \boldsymbol{\phi}^T\boldsymbol{m}_N\right)^2\right\} \\
\propto \quad & \exp\left\{-\frac{1}{2(\beta^{-1} + \boldsymbol{\phi}^T\boldsymbol{S}_N\boldsymbol{\phi})}\left(t^2 - \boldsymbol{\phi}^T\boldsymbol{m}_N\right)^2\right\}
\end{aligned}
$$

预测分布

因此

$$
\begin{aligned}
& \int p(t \mid \boldsymbol{x}, \boldsymbol{w}, \beta) \cdot p(\boldsymbol{w} \mid \mathbf{t}, \alpha, \beta)\, d\boldsymbol{w} \\
\propto \quad & \exp\left\{-\frac{\beta}{2}\left[\left(1-\beta\boldsymbol{\phi}^T\boldsymbol{S}\boldsymbol{\phi}\right)t^2 - 2\left(1-\beta\boldsymbol{\phi}^T\boldsymbol{S}\boldsymbol{\phi}\right)\boldsymbol{\phi}^T\boldsymbol{m}_N t\right]\right\} \\
\propto \quad & \exp\left\{-\frac{\beta\left(1-\beta\boldsymbol{\phi}^T\boldsymbol{S}\boldsymbol{\phi}\right)}{2}\left(t^2-\boldsymbol{\phi}^T\boldsymbol{m}_N\right)^2\right\} \\
\propto \quad & \exp\left\{-\frac{1}{2(\beta^{-1}+\boldsymbol{\phi}^T\boldsymbol{S}_N\boldsymbol{\phi})}\left(t^2-\boldsymbol{\phi}^T\boldsymbol{m}_N\right)^2\right\}
\end{aligned}
$$

记

$$
m_N = \boldsymbol{m}_N^T\phi(x), \quad \sigma_N^2 = \beta^{-1} + \boldsymbol{\phi}^T\boldsymbol{S}_N\boldsymbol{\phi},
$$

归一化后即得预测分布:

$$
p(t \mid \boldsymbol{x}, \mathbf{t}, \alpha, \beta) = \mathcal{N}(t \mid m_N, \sigma_N^2), \tag{25}
$$

证据近似 (evidence approximation)

证据近似 (evidence approximation)

- 贝叶斯线性回归需要知道参数 $\boldsymbol{w}$ 的先验分布，其中包含超参数
- 若把超参数也看做是随机变量，直接对超参数积分是难以得到解析形式的
- 可使用近似方法进行处理
- 证据近似也称为
  - 经验贝叶斯 (Empirical Bayes)
  - Type 2 Maximum Likelihood
  - Generalized Maximum Likelihood

证据近似 (evidence approximation)

- 目标变量 $t \in \mathbb{R}$ 由决策函数 $y(\boldsymbol{x}, \boldsymbol{w})$ 加上高斯噪声给定，即

$$t = y(\boldsymbol{x}, \boldsymbol{w}) + \epsilon$$

其中

$$\epsilon \sim \mathcal{N}(0, \beta^{-1}).$$

- 线性回归模型

$$y(\boldsymbol{x}, \boldsymbol{w}) = \sum_{j=0}^{M-1} w_j \phi_j(\boldsymbol{x}) = \boldsymbol{\phi}(\boldsymbol{x})^T \boldsymbol{w}$$

其中

$$\boldsymbol{w} = (w_0, \cdots, w_{M-1})^T \in \mathbb{R}^M, \quad \boldsymbol{\phi}(\boldsymbol{x}) = (\phi_0(\boldsymbol{x}), \cdots, \phi_{M-1}(\boldsymbol{x}))^T \in \mathbb{R}^M$$

证据近似 (evidence approximation)

贝叶斯回归

- 引入参数 $\boldsymbol{w}$ 的先验分布 $p(\boldsymbol{w})$
- 由于似然函数 $p(t \mid \boldsymbol{w})$ 为指数形式，故 $\boldsymbol{w}$ 的共轭先验为高斯分布，设为

$$p(\boldsymbol{w}) = (\boldsymbol{w} \mid 0, \alpha^{-1}\boldsymbol{I})$$

- 后验估计仍为高斯分布：

$$p(\boldsymbol{w} \mid \mathbf{t}) = (\boldsymbol{w} \mid \boldsymbol{m}_N, \boldsymbol{S}_N)$$

其中

$$\boldsymbol{S}_N^{-1} = \boldsymbol{S}_0^{-1} + \beta\boldsymbol{\Phi}^T\boldsymbol{\Phi}, \quad \boldsymbol{S}_N^{-1}\boldsymbol{m}_N = \beta\boldsymbol{\Phi}^T\mathbf{t}$$

$$\boldsymbol{\Phi} = \begin{bmatrix} \phi(\boldsymbol{x}_1)^T \\ \vdots \\ \phi(\boldsymbol{x}_N)^T \end{bmatrix} \in \mathbb{R}^{N\times M}$$

证据近似 (evidence approximation)

贝叶斯回归的参数与超参数

- 参数为 $\boldsymbol{w} \in \mathbb{R}^M$，其中 M 表示模型的复杂度
- 两个超参数
  - α 为参数精度
  $$p(\boldsymbol{w} \mid \alpha) = \mathcal{N}(\boldsymbol{w} \mid 0, \alpha^{-1}\boldsymbol{I})$$
  - β 为噪声精度
  $$p(t \mid \boldsymbol{x}, \boldsymbol{w}, \beta) = \mathcal{N}(\boldsymbol{w} \mid y(\boldsymbol{x}, \boldsymbol{w}), \beta^{-1})$$

证据近似 (evidence approximation)

若引入 α 与 β 的超先验分布，则预测分布可通过对 α, β 和 $\boldsymbol{w}$ 求积分 (边缘化, marginalizing) 得到：

$$p(t \mid \mathbf{x}, \mathbf{t}) = \int\int\int p(t \mid \mathbf{x}, \mathbf{w}, \beta) \cdot p(\mathbf{w} \mid \mathbf{t}, \alpha, \beta) \cdot p(\alpha, \beta \mid \mathbf{t})\, d\mathbf{w}\, d\alpha\, d\beta \quad (26)$$

其中

$$\begin{aligned} p(t \mid \mathbf{x}, \mathbf{w}, \beta) &= \mathcal{N}(t \mid \phi(\mathbf{x})^T \mathbf{w}, \beta^{-1}) \\ p(\mathbf{w} \mid \mathbf{t}, \alpha, \beta) &= \mathcal{N}(\mathbf{w} \mid \mathbf{m}_N, \mathbf{S}_N) \end{aligned}$$

注

积分(26)无法给出解析解，只能近似处理。

证据近似 (evidence approximation)

若存在 $\hat{\alpha}, \hat{\beta}$ 使得 $p(\alpha, \beta \mid \mathbf{t})$ 在 $\hat{\alpha}, \hat{\beta}$ 附近取得最大，则预测分布可近似为：

$$
\begin{aligned}
p(t \mid \boldsymbol{x}, \mathbf{t}) &\approx \int\int\int p(t \mid \boldsymbol{x}, \boldsymbol{w}, \hat{\beta}) \cdot p(\boldsymbol{w} \mid \mathbf{t}, \hat{\alpha}, \hat{\beta}) \cdot p(\alpha, \beta \mid \mathbf{t})\, d\boldsymbol{w}\, d\alpha\, d\beta \\
&= \int p(t \mid \boldsymbol{x}, \boldsymbol{w}, \hat{\beta}) \cdot p(\boldsymbol{w} \mid \mathbf{t}, \hat{\alpha}, \hat{\beta})\, d\boldsymbol{w} \\
&:= p(t \mid \boldsymbol{x}, \mathbf{t}, \hat{\alpha}, \hat{\beta})
\end{aligned}
$$

我们的目标就是要找到这样的 $\hat{\alpha}, \hat{\beta}$。

证据近似 (evidence approximation)

由贝叶斯公式知，

$$p(\alpha, \beta \mid \mathbf{t}) \propto p(\mathbf{t} \mid \alpha, \beta) \cdot p(\alpha, \beta).$$

如果先验分布 $p(\alpha, \beta)$ 相对比较平，那么在证据框架中，$\hat{\alpha}$ 和 $\hat{\beta}$ 可通过最大化边缘似然函数 $p(\mathbf{t} \mid \alpha, \beta)$ 来获得，这将使我们能够从训练数据本身确定这些超参数的值。

证据近似 (evidence approximation)

计算证据函数

计算证据函数

边缘似然函数 $p(\mathbf{t} \mid \alpha, \beta)$ 是通过对 $\boldsymbol{w}$ 求积分而得到的，即

$$p(\mathbf{t} \mid \alpha, \beta) = \int p(\mathbf{t} \mid \boldsymbol{w}, \beta) \cdot p(\boldsymbol{w} \mid \alpha) \, d\boldsymbol{w} \tag{27}$$

其中

$$p(\mathbf{t} \mid \boldsymbol{w}, \beta) = \prod_{i=1}^{N} \mathcal{N}(t_i \mid \phi(\boldsymbol{x}_n)^T \boldsymbol{w}, \beta^{-1}), \quad p(\boldsymbol{w} \mid \alpha) = \mathcal{N}(\boldsymbol{w} \mid 0, \alpha^{-1} \boldsymbol{I})$$

通过推导可知，

$$p(\mathbf{t} \mid \alpha, \beta) = \left(\frac{\beta}{2\pi}\right)^{N/2} \left(\frac{\alpha}{2\pi}\right)^{M/2} \int \exp\left\{-E(\boldsymbol{w})\right\} \, d\boldsymbol{w} \tag{28}$$

其中

$$\begin{aligned} E(\boldsymbol{w}) &= \beta E_{\mathcal{D}}(\boldsymbol{w}) + \alpha E_{\mathcal{R}}(\boldsymbol{w}) \\ &= \frac{\beta}{2} \|\boldsymbol{\Phi} \boldsymbol{w} - \mathbf{t}\|^2 + \frac{\alpha}{2} \|\boldsymbol{w}\|^2. \end{aligned} \tag{29}$$

计算证据函数

对 $\boldsymbol{w}$ 配方，可得

$$E(\boldsymbol{w}) = E(\boldsymbol{m}_N) + \frac{1}{2}(\boldsymbol{w} - \boldsymbol{m}_N)^T \boldsymbol{A}(\boldsymbol{w} - \boldsymbol{m}_N) \tag{30}$$

其中

$$\boldsymbol{A} = \boldsymbol{S}_N^{-1} = \alpha \boldsymbol{I} + \beta \boldsymbol{\Phi}^T \boldsymbol{\Phi} \tag{31}$$

$$\boldsymbol{S}_N^{-1} \boldsymbol{m}_N = \beta \boldsymbol{\Phi}^T \mathbf{t} \tag{32}$$

$$E(\boldsymbol{m}_N) = \frac{\beta}{2} \|\boldsymbol{\Phi} \boldsymbol{m}_N - \mathbf{t}\|^2 + \frac{\alpha}{2} \|\boldsymbol{m}_N\|^2 \tag{33}$$

注意 $\boldsymbol{A}$ 为 $E(\boldsymbol{w})$ 关于 $\boldsymbol{w}$ 的 Hessian 矩阵，即

$$\boldsymbol{A} = \nabla^2 E(\boldsymbol{w}).$$

计算证据函数

因

$$\int \exp\left\{-\frac{1}{2}(\boldsymbol{w}-\boldsymbol{m}_N)^T\boldsymbol{A}(\boldsymbol{w}-\boldsymbol{m}_N)\right\} d\boldsymbol{w} = (2\pi)^{M/2}|\boldsymbol{A}|^{-1/2}$$

故

$$\int \exp\{-E(\boldsymbol{w})\}\, d\boldsymbol{w} = (2\pi)^{M/2}|\boldsymbol{A}|^{-1/2}\exp\{-E(\boldsymbol{m}_N)\}\,. \tag{34}$$

从而，边缘似然函数的对数为

$$\ln p(\mathbf{t}\mid\alpha,\beta) = \frac{M}{2}\ln\alpha + \frac{N}{2}\ln\beta - E(\boldsymbol{m}_N) - \frac{1}{2}\ln|\boldsymbol{A}| - \frac{M}{2}\ln(2\pi) \tag{35}$$

此即证据函数。

证据近似 (evidence approximation)

最大化证据函数

最大化证据函数

本小节讨论如何确定 α 和 β 来最大化 $p(\mathbf{t} \mid \alpha, \beta)$。设 λ_i 为 $\mathbf{\Phi}^T\mathbf{\Phi}$ 的特征值，对应的特征向量为 $\boldsymbol{u}_i$，即满足

$$\left(\mathbf{\Phi}^T\mathbf{\Phi}\right)\boldsymbol{u}_i = \lambda_i \boldsymbol{u}_i. \tag{36}$$

以下分为三种情况来讨论 α 和 β 的确定。

- 已知 β，确定 α
- 已知 α，确定 β
- α, β 都需要确定

由 $\boldsymbol{A} = \alpha \boldsymbol{I} + \beta \boldsymbol{\Phi}^T \boldsymbol{\Phi}$ 可知，$\boldsymbol{A}$ 的特征值 $\beta\lambda_i + \alpha$。对(35)两端关于 α 求偏导可得

$$\frac{d}{d\alpha} \ln p(\mathbf{t} \mid \alpha, \beta) = \frac{M}{2\alpha} - \frac{1}{2} \boldsymbol{m}_N^T \boldsymbol{m}_N - \sum_i \frac{1}{\beta\lambda_i + \alpha} \tag{37}$$

这里用到了

$$\frac{d}{d\alpha} \ln |\boldsymbol{A}| = \frac{d}{d\alpha} \ln \prod_{i=1}^{M} (\beta\lambda_i + \alpha) = \frac{d}{d\alpha} \sum_{i=1}^{M} \ln(\beta\lambda_i + \alpha) = \sum_{i=1}^{M} \frac{1}{\beta\lambda_i + \alpha}$$

已知 β，确定 α

由 $\boldsymbol{A} = \alpha \boldsymbol{I} + \beta \boldsymbol{\Phi}^T \boldsymbol{\Phi}$ 可知，$\boldsymbol{A}$ 的特征值 $\beta\lambda_i + \alpha$。对(35)两端关于 α 求偏导可得

$$\frac{d}{d\alpha} \ln p(\mathbf{t} \mid \alpha, \beta) = \frac{M}{2\alpha} - \frac{1}{2} \boldsymbol{m}_N^T \boldsymbol{m}_N - \sum_i \frac{1}{\beta\lambda_i + \alpha} \tag{37}$$

这里用到了

$$\frac{d}{d\alpha} \ln |\boldsymbol{A}| = \frac{d}{d\alpha} \ln \prod_{i=1}^{M} (\beta\lambda_i + \alpha) = \frac{d}{d\alpha} \sum_{i=1}^{M} \ln(\beta\lambda_i + \alpha) = \sum_{i=1}^{M} \frac{1}{\beta\lambda_i + \alpha}$$

由此可知，(35)关于 α 的驻点满足

$$\frac{M}{2\alpha} - \frac{1}{2} \boldsymbol{m}_N^T \boldsymbol{m}_N - \sum_i \frac{1}{\beta\lambda_i + \alpha} = 0 \tag{38}$$

亦即

$$\alpha \boldsymbol{m}_N^T \boldsymbol{m}_N = M - \alpha \sum_{i=1}^{M} \frac{1}{\beta\lambda_i + \alpha} = \sum_{i=1}^{M} \frac{\beta\lambda_i}{\beta\lambda_i + \alpha} := \gamma \tag{39}$$

已知 β，确定 α

于是，边缘似然函数(35)的极大解为

$$\alpha = \frac{\gamma}{\boldsymbol{m}_N^T \boldsymbol{m}_N} \tag{40}$$

已知 β，确定 α

于是，边缘似然函数(35)的极大解为

$$\alpha = \frac{\gamma}{\boldsymbol{m}_N^T \boldsymbol{m}_N} \tag{40}$$

注

由于 γ 和 $\boldsymbol{m}_N$ 都依赖于 α，故上式是 α 的一个隐式解。

已知 β，确定 α

α 的计算

- 初始化 α
- 重复以下步骤，直到收敛：
  - 利用 $\left(\alpha \boldsymbol{I} + \beta \boldsymbol{\Phi}^T \boldsymbol{\Phi}\right) \boldsymbol{m}_N = \beta \boldsymbol{\Phi}^T \mathbf{t}$ 求出 $\boldsymbol{m}_N$；
  - 利用 $\gamma = \sum_{i=1}^{M} \frac{\beta \lambda_i}{\beta \lambda_i + \alpha}$ 计算 γ；
  - 利用 $\alpha = \frac{\gamma}{\boldsymbol{m}_N^T \boldsymbol{m}_N}$ 重新计算 α。

已知 β，确定 α

注

在迭代过程中，$\mathbf{\Phi}^T\mathbf{\Phi} \in \mathbb{R}^{M \times M}$ 是不变的，故只需在迭代前计算一次 λ_i。

注

α 的值是纯粹通过训练数据集来确定的，这与极大似然法完全不同。

已知 α，确定 β

对(35)两端关于 β 求偏导可得

$$\frac{d}{d\beta}\ln p(\mathbf{t}\mid\alpha,\beta)=\frac{N}{2\beta}-\frac{1}{2}\|\boldsymbol{\Phi}\boldsymbol{m}_N-\mathbf{t}\|^2-\frac{\gamma}{2\beta} \tag{41}$$

这里用到了

$$\frac{d}{d\beta}\ln|\boldsymbol{A}|=\frac{d}{d\beta}\ln\prod_{i=1}^{M}(\beta\lambda_i+\alpha)=\frac{d}{d\beta}\sum_{i=1}^{M}\ln(\beta\lambda_i+\alpha)=\sum_{i=1}^{M}\frac{\lambda_i}{\beta\lambda_i+\alpha}=\frac{\gamma}{\beta}.$$

由此可知，(35)关于 β 的驻点满足

$$\frac{N}{2\beta}-\frac{1}{2}\|\boldsymbol{\Phi}\boldsymbol{m}_N-\mathbf{t}\|^2-\frac{\gamma}{2\beta}=0 \tag{42}$$

已知 α，确定 β

于是，边缘似然函数(35)的极大解为

$$\beta^{-1} = \frac{1}{N - \gamma} \|\boldsymbol{\Phi} \boldsymbol{m}_N - \mathbf{t}\|^2. \tag{43}$$

跟 α 一样，这是 β 的一个隐式解。

已知 α，确定 β

β 的计算

- 初始化 β
- 重复以下步骤，直到收敛：
  - 利用 $\left(\alpha \boldsymbol{I} + \beta \boldsymbol{\Phi}^T \boldsymbol{\Phi}\right) \boldsymbol{m}_N = \beta \boldsymbol{\Phi}^T \mathbf{t}$ 求出 $\boldsymbol{m}_N$；
  - 利用 $\gamma = \sum_{i=1}^{M} \frac{\beta \lambda_i}{\beta \lambda_i + \alpha}$ 计算 γ；
  - 利用 $\beta^{-1} = \frac{1}{N-\gamma} \|\boldsymbol{\Phi} \boldsymbol{m}_N - \mathbf{t}\|^2$ 重新计算 β。

α, β 均需确定

由以上分析可知

$$\nabla \ln p(\mathbf{t} \mid \alpha, \beta) = \begin{bmatrix} \frac{M}{2\alpha} - \frac{1}{2}\boldsymbol{m}_N^T\boldsymbol{m}_N - \sum_{i=1}^{M} \frac{1}{\beta\lambda_i + \alpha} \\ \frac{N}{2\beta} - \frac{1}{2}\|\boldsymbol{\Phi}\boldsymbol{m}_N - \mathbf{t}\|^2 - \frac{\gamma}{2\beta} \end{bmatrix} = 0 \tag{44}$$

即

$$\begin{cases} \alpha & = & \frac{\gamma}{\boldsymbol{m}_N^T\boldsymbol{m}_N} \\ \beta^{-1} & = & \frac{1}{N-\gamma}\|\boldsymbol{\Phi}\boldsymbol{m}_N - \mathbf{t}\|^2 \end{cases} \tag{45}$$

其中

$$\gamma = \sum_{i=1}^{M} \frac{\beta\lambda_i}{\beta\lambda_i + \alpha}$$

α, β 均需确定

α, β 的计算

- 初始化 α, β
- 重复以下步骤，直至收敛
  - 利用 $\left(\alpha \boldsymbol{I} + \beta \boldsymbol{\Phi}^T \boldsymbol{\Phi}\right) \boldsymbol{m}_N = \beta \boldsymbol{\Phi}^T \mathbf{t}$ 求出 $\boldsymbol{m}_N$；
  - 利用 $\gamma = \sum_{i=1}^{M} \frac{\beta \lambda_i}{\beta \lambda_i + \alpha}$ 更新 γ；
  - 利用

$$\begin{cases} \alpha & = & \dfrac{\gamma}{\boldsymbol{m}_N^T \boldsymbol{m}_N} \\ \beta^{-1} & = & \dfrac{1}{N - \gamma} \left\| \boldsymbol{\Phi} \boldsymbol{m}_N - \mathbf{t} \right\|^2 \end{cases}$$

    更新 α, β。