Dong Wang

现代机器学习技术导论

2017 年 8 月 13 日

Springer

Contents

Chapter 1
机器学习概述

Chapter 2
线性模型

线性模型是机器学习中最简单、也是最常用的模型。所谓线性，在不同领域和不同应用场景下有不同的含义。我们所讨论的线性，是指变量间具有如下简单形式：

$$t = Wx, \tag{2.1}$$

其中，x和t表示两个多维变量，W 为模型参数。线性模型虽然简单，但在机器学习中有重要意义。首先，线性模型简单高效，容易实现；其次，很多实际问题都具有粗略的线性性，特别是在选择了合理的特征提取方式之后，这一线性性会更为明显，使得线性模型已经足以胜任工作；最后，线性模型参数较少，适应性强，具有很强的泛化能力。因此，当我们面对一个机器学习问题的时候，首先要考虑的就是线性模型。

本章我们将讨论几种简单的线性模型：一种是线性预测模型，包括线性回归和Logistic回归，在讨论它们概率意义的基础上引入贝叶斯方法；另一种是线性概率模型，基于隐变量和观察变量间的线性假设来推理数据的内在结构，如PCA、LDA、PLDA等。前者是有监督学习，后者一般用于无监督学习。

2.1 线性预测模型

如果x和t都是可见变量，则公式 2.1表示一种x对t的预测模型。这一模型称为“线性预测模型”。在有些问题中，x和t之间不存在直接线性关系，但可以通过某种变换$\phi(\cdot)$建立这种关系，即：

$$t = W\phi(x), \tag{2.2}$$

其中$\phi(\cdot)$通常是非线性的。从模型角度看，不论是基于原始数据还是基于变换后的数据，模型的线性属性和优化方法没有区别，但变换ϕ 的引入确实具有重要意义，它使得很多在原始变量空间里无法用线性模型解决的问题在变换空间中得以合理解决。下面我们从简单的多项式拟合问题开始讨论。

2.1.1 从多项式拟合说起

假设一个包括N个样本的数据集$D = \{(x^{(n)}, t^{(n)}) : n = 1,2,...,N\}$，其中$x^{(n)}$和$t^{(n)}$都是一维的，且$t^{(n)}$为$x^{(n)}$对应的目标值。我们的任务是学习一个预测函数$y = f(x)$，使其对数据集$D$中任一样本$x^{(n)}$的预测结果尽可能接近$t^{(n)}$。如果限定该预测函数为$M$次多项式，则得到预测公式为：

$$y(x;w) = w_0 + w_1 x + w_2 x^2 + \cdots + w_M x^M = \sum_{j=0}^{M} w_j x^j, \tag{2.3}$$

其中$w = [w_0, w_1, \ldots, w_M]^T$是每一阶多项式对应的预测系数。图 2.1给出一个$M = 1$的预测函数，该函数是一条以$w_0$为截距，以$w_1$为斜率的直线。

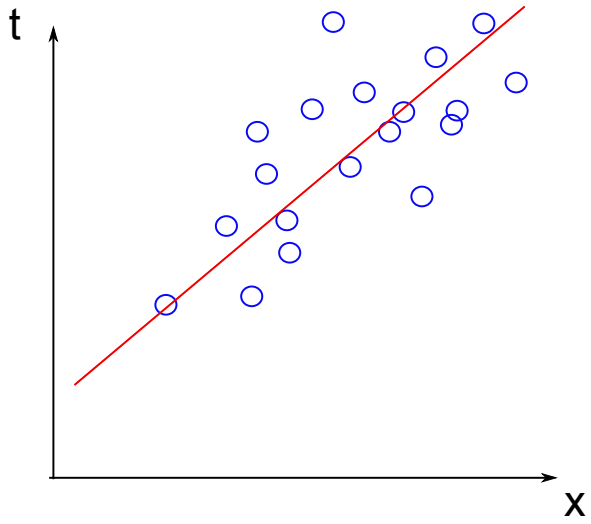

Fig. 2.1 多项式拟合得到的线性预测函数$y(x;w) = w_0 + w_1 x$。

给定预测模型形式（多项式最大阶数M确定），需要对预测系数w 进行优化。为此，需定义误差函数并通过修改参数w对该误差函数进行优化。一般定义训练集D上的平方误差为误差函数，公式如下：

$$E(w) = \frac{1}{2}\sum_{n=1}^{N}\{y(x^{(n)};w) - t^{(n)}\}^2. \tag{2.4}$$

注意该误差函数是w的函数。将式 2.3代入上式，有:

$$E(w) = \frac{1}{2}\sum_{n=1}^{N}(\sum_{j=0}^{M} w_j[x^{(n)}]^j - t^{(n)})^2.$$

对每个参数w_k求偏导数并令其等于零，可得：

$$\sum_{n=1}^{N}[x^{(n)}]^k \sum_{j=0}^{M} w_j[x^{(n)}]^j = \sum_{n=1}^{N} t^{(n)}[x^{(n)}]^k \quad k = 0,1,2,...,M$$

整理得：

$$\sum_{j=0}^{M} w_j \sum_{n=1}^{N}[x^{(n)}]^{k+j} = \sum_{n=1}^{N} t^{(n)}[x^{(n)}]^k \quad k = 0,1,2,...,M$$

展开写成：

$$\begin{bmatrix} w_0\sum_{n=1}^{N}[x^{(n)}]^0 + & w_1\sum_{n=1}^{N}[x^{(n)}]^1 & + ... + & w_M\sum_{n=1}^{N}[x^{(n)}]^M & = \sum_{n=1}^{N} t^{(n)}[x^{(n)}]^0 \\ w_0\sum_{n=1}^{N}[x^{(n)}]^1 + & w_1\sum_{n=1}^{N}[x^{(n)}]^2 & + ... + & w_M\sum_{n=1}^{N}[x^{(n)}]^{M+1} & = \sum_{n=1}^{N} t^{(n)}[x^{(n)}]^1 \\ \quad ... & \quad ... & ... & ... & ... \\ w_0\sum_{n=1}^{N}[x^{(n)}]^M + & w_1\sum_{n=1}^{N}[x^{(n)}]^{M+1} & + ... + & w_M\sum_{n=1}^{N}[x^{(n)}]^{2M} & = \sum_{n=1}^{N} t^{(n)}[x^{(n)}]^M \end{bmatrix}.$$

写成矩阵格式有：

$$\begin{bmatrix} \sum_{n=1}^{N}[x^{(n)}]^0 & \sum_{n=1}^{N}[x^{(n)}]^1 & ... & \sum_{n=1}^{N}[x^{(n)}]^M \\ \sum_{n=1}^{N}[x^{(n)}]^1 & \sum_{n=1}^{N}[x^{(n)}]^2 & ... & \sum_{n=1}^{N}[x^{(n)}]^{M+1} \\ \quad ... & \quad ... & ... & ... \\ \sum_{n=1}^{N}[x^{(n)}]^M & \sum_{n=1}^{N}[x^{(n)}]^{M+1} & ... & \sum_{n=1}^{N}[x^{(n)}]^{2M} \end{bmatrix} \begin{bmatrix} w_0 \\ w_1 \\ ... \\ w_M \end{bmatrix} = \begin{bmatrix} \sum_{n=1}^{N} t^{(n)}[x^{(n)}]^0 \\ \sum_{n=1}^{N} t^{(n)}[x^{(n)}]^1 \\ ... \\ \sum_{n=1}^{N} t^{(n)}[x^{(n)}]^M \end{bmatrix}.$$

可以证明，上式中左边矩阵是可逆的，因此w 有唯一解，即为最优预测系数。上述以多项式作为函数形式对$t = f(x)$进行拟合的方法称为多项式拟合。

如果将多项式拟合中的每一阶x^j看作一个非线性映射$\phi_j(x)=x^j$，并将自变量x扩展到多维变量，则上述多项式拟合可以扩展为一般线性拟合方法。设非线性映射个数为M，将映射后的变量写成向量格式:

$$\phi(x)=[\phi_0(x),\phi_1(x),...\phi_M(x)]^T.$$

同样采用平方误差作为误差函数：

$$E(w)=\frac{1}{2}\sum_{n=1}^{N}\{w^T\phi(x^{(n)})-t^{(n)}\}^2. \tag{2.5}$$

对上式取w的导数并使之为零，即有：

$$\begin{aligned}\nabla E(w) &= \sum_{n=1}^{N}\{w^T\phi(x^{(n)})-t^{(n)}\}\phi(x^{(n)})^T \\ &= w^T\sum_{n=1}^{N}\phi(x^{(n)})\phi(x^{(n)})^T-\sum_{n=1}^{N}t^{(n)}\phi(x^{(n)})^T \\ &= \mathbf{0}.\end{aligned}$$

写成矩阵形式，有：

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

其中$\boldsymbol{\Phi}$为数据矩阵，每一行代表一个样本，每一列代表一个非线性映射，即：

$$\boldsymbol{\Phi}=\begin{pmatrix}\phi_0(x^{(1)}) & \phi_1(x^{(1)}) & ... & \phi_M(x^{(1)}) \\ \phi_0(x^{(2)}) & \phi_1(x^{(2)}) & ... & \phi_M(x^{(2)}) \\ ... & ... & ... & ... \\ \phi_0(x^{(N)}) & \phi_1(x^{(N)}) & ... & \phi_M(x^{(N)})\end{pmatrix},$$

$\mathbf{t}$ 为训练集中所有训练样本的目标值组成的向量，即：

$$\mathbf{t}=[t^{(1)},t^{(2)},...,t^{(N)}]^T.$$

由此可得线性拟合的最优预测参数为：

$$w=(\boldsymbol{\Phi}^T\boldsymbol{\Phi})^{-1}\boldsymbol{\Phi}^T\mathbf{t}.$$

上面推导中假设t是一维变量，这一推导很容易扩展到t是多维变量的情况，即式（2.1）所示的一般形式。此时模型参数不再是一个向量w，而是一个矩阵W。

到目前为止，我们似乎已经完美解决了线性拟合问题，连包含非线性变换的线性拟合问题也得到了确定解。然而，一个简单的问题是：为什么要使用平方误差作为误差函数而不用其它函数，比如绝对值误差？下一节我们将引入概率这一工具来解释这些选择。我们会发现，平方误差事实上对应的是训练数据中的一种高斯不确定性。

2.1.2 线性回归

几乎所有实际问题中都存在随机性或不确定性。这些不确定性可能来自于观测手段的不精确，但更多来自我们对数据本身理解上的局限性。例如当我们考察一架从北京飞往上海的飞机时，可以看到其飞行轨迹总体上是一条平稳的曲线，但细致观察，就会发现很多不确定性。这些不确定性有些来自驾驶员的自主操作，有些来自发动机转动时的不稳定，有些来自飞行过程中遇到的气流、云朵等的影响，还有的来自机舱内乘客的活动，等等。假设我们能考虑到所有这些细节，依动力学列出一个庞大的方程组，则飞行过程中的绝大部分不确定性是可以解释的。然而，在实际应用中，考虑到问题的复杂度，不可能对这些细节一一建模；即便我们想这么做，也不可能穷尽所有影响因素，总会有些因素超出我们的考虑范围和理解能力。因此，在处理一个任务时，我们不得不忽略很多细节；一旦我们忽略这些细节，就产生了不确定性。这意味着不确定性在所有实际问题中都是不可避免的。

这一节我们将引入概率模型来帮助我们描述这些不确定性，并由此得到基于概率的最优解。概率模型的引入是机器学习历史上里程碑式的事件，本书中几乎所有章节都围绕概率展开。就本章而言，概率方法可让我们对线性模型有更深刻的理解。这一理解有助于我们将线性模型和未来要讨论的复杂模型联系起来，并基于此对现有模型和算法进行系统化的优化和扩展。

回到之前的线性拟合任务$y = w^T \phi(x)$。我们希望拟合得到的预测值y和目标变量t越相似越好，因此提出了基于平方误差的优化准则。现在我们假设：y与t之所以存在差别，是因为观察值t由于种种原因是随机的，不确定的。不论产生这种随机性的原因是什么，我们假设这一随机性符合一个以0为均值，以β^{-1}为方差的高斯分布。引入一个随机变量ε 来表示这一随机性，则有：

$$\begin{aligned} t &= y(x;w)+\varepsilon \\ &= w^T\phi(x)+\varepsilon, \end{aligned} \tag{2.6}$$

其中$\varepsilon \sim N(0,\beta^{-1})$。

公式 2.6构造了一个由x 到观察值t的生成模型：首先，由输入变量x经过非线性映射生成特征向量ϕ，再经过线性映射生成预测y，最后加入一个高斯噪声得到目标的观察值t。这一模型称为线性回归模型（Linear Regression)。所谓回归，是指对输入变量x和目标变量t之间相互依赖关系的统计分析。图2.2给出了在一维输入变量情况下，线性回归模型生成目标变量的示意图。

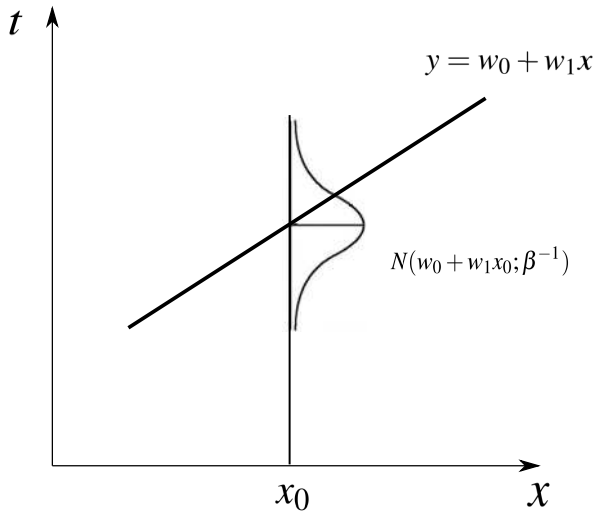

Fig. 2.2 设生成函数为$y(x;w)=w_0+w_1x$，目标值t服从以$y(x;w)$ 为中心的高斯分布。

给定一个输入变量x，可以基于上述线性回归模型计算对应的目标观察值t的生成概率：

$$p(t|x;w,\beta)=N(t|y(x);w,\beta^{-1}).$$

如果我们将(x,t)作为一个整体，则上式也是该二元组在这一模型下的生成概率。对给定的训练集$D=\{(x^{(n)},t^{(n)}):n=1,2,...,N\}$，该模型生成这一数据集的总概率为：

$$p(D;w,\beta)=\prod_{n=1}^{N}N(t^{(n)}|w^T\phi(x^{(n)}),\beta^{-1}). \tag{2.7}$$

上式是w和β的函数，一般称为似然函数。显然，模型对某一数据集的描述能力越强，则该模型生成这一数据集的概率越大，似然函数的值也越大。因此，如果我们能找到一组参数使得似然函数的值最大化，则可实现该模型在生成概率意义上的最优化。这一优化准则称为最大似然（Maximum Likelihood, ML）准则，相应的优化方法称为最大似然估计。最大似然估计可形式化如下：

$$\{w_{ML}, \beta_{ML}\} = \arg\max_{w,\beta} p(D; w, \beta).$$

对线性回归模型的最大似然估计可通过对该似然函数取偏导数的零点得到。为计算方便，将似然函数取对数，并带入高斯分布的概率公式，有：

$$\begin{aligned}\ln p(D; w, \beta) &= \sum_{n=1}^{N} \ln\{N(t^{(n)}|w^T\phi(x^{(n)}), \beta^{-1})\} \\ &= \frac{N}{2}\ln\beta - \frac{N}{2}\ln(2\pi) - \beta E(w),\end{aligned}$$

其中前两项与w无关，第三项为:

$$E(w) = \frac{1}{2}\sum_{n=1}^{N}\{t^{(n)} - w^T\phi(x^{(n)})\}^2.$$

因此，对w的最大似然估计等价于对$E(w)$的最小化。仔细观察发现，$E(w)$正是式（2.5）所定义的平方误差。这意味着最大似然估计事实上等价于线性拟合。

仔细观察似然函数$p(D; w, \beta)$，可以看到$E(w)$项来自高斯分布中的指数项$e^{\frac{\beta}{2}(t-w^T\phi(x))^2}$，这说明线性拟合中的平方误差事实上假设了目标观察值中的噪声符合高斯分布。一方面，这说明平方误差是合理的，因为基于中心极限定理，高斯噪声是最简单也是最现实的噪声；另一方面，这也说明概率模型是一种更有效的建模方式。在没有引入概率模型时，我们并不清楚平方误差对应的数据分布情况，因此对该误差的合理性不好判断。一旦我们引入了概率工具，将这一误差和高斯假设联系起来，即可更深刻理解这一误差的合理性与局限性。由此，当数据噪声不符合高斯分布时，基于概率模型可以依具体情况设计更合理的误差函数。我们在下面几节会反复看到类似情况：当引入概率模型后，我们才对很多传统方法有更深刻的理解，认识到这些经验设计背后的假设和含义，明确每一种方法的适用范围和缺陷，进而提出改进和扩展的方法。

现在让我们完成线性回归模型的讨论。优化w的过程和线性回归方法完全相同，我们仅写出其推导形式。首先计算上述平方误差的梯度：

$$\nabla_w \ln p(D|w,\beta) = \sum_{n=1}^{N} \{w^T \phi(x^{(n)}) - t^{(n)}\} \phi(x^{(n)}). \tag{2.8}$$

上式说明该误差函数对w的梯度由预测值和观察值之间的差异导致，以这些差异为权重对输入变量取均值即得到模型在当前参数下的梯度。取该梯度为零，解得w如下：

$$w_{ML} = (\boldsymbol{\Phi}^T \boldsymbol{\Phi})^{-1} \boldsymbol{\Phi}^T \mathbf{t}.$$

其中下标ML表明该解基于最大似然准则得到。同理，对β求导并令导数为零，有：

$$\nabla_\beta \ln p(D|w,\beta) = \frac{N}{2\beta} - E(w) = 0.$$

代入w的最大似然估计w_{ML}，得到：

$$\frac{1}{\beta_{ML}} = \frac{1}{N} \sum_{n=1}^{N} \{t^{(n)} - w_{ML}^T \phi(x^{(n)})\}^2. \tag{2.9}$$

注意到$\frac{1}{\beta}$事实上是高斯分布的方差，上式说明，对该方差的最大似然估计等于回归模型对训练数据进行预测得到的预测残差。

假设我们已经通过上述最大似然估计得到了一个线性回归模型如下：

$$t = w^T \phi(x) + \varepsilon,$$

如何基于该模型预测一个新的输入变量x' 的目标t' 呢？首先注意到我们得到的模型包含一个随机变量ε，因此是一个随机生成模型，该模型对一个确定的输入x'并没有一个确定的预测t'，而是给出t的概率$p(t|x)$。后面我们会看到，这种基于概率的预测对完整描述一个随机系统具有重要意义，但现在我们只希望像通常的预测模型一样有个确定的预测值。一种方法是求预测的期望，可计算如下：

$$\mathbb{E}[t'|x'] = \int t\, p(t'|x') \mathrm{d}t = y(x'; w_{ML}) = w_{ML}^T \phi(x').$$

可见，该预测和线性拟合的预测结果是一致的，同时也是基于该线性回归模型得到的概率最大的预测值。

一个问题是，高斯分布是最优选择吗？这可能需要结合数据的实际分布情况。比如当数据具有很强的长尾特性时，就可能要考虑像Student-t分布或拉普拉斯分布。一般来说，如果我们对数据的特性了解得并不清楚，高斯分布通常是合理选择。然而，有一种情况需要我们必须考虑非高斯分布：当目标t是离散的，则高斯分布显然是不适合的，这时需要用离散分布来描述数据中的噪声。下节要讨论的Logistic回归模型即是这种处理离散噪声的模型。

上述讨论中我们假设目标变量t是单变量。同样的方法很容易扩展到多变量情况；同时，ε可以认为是变量x 的一部分，因此线性回归模型事实上是标准线性模型$t = W\phi(x)$的特殊形式，其中x一部分是可见变量，另一部分则是不可见的（即ε），称为隐变量（Latent Variable）。在线性概率模型一节中我们会看到x全是隐变量的情况。

2.1.3 Fisher准则与线性分类

在分类问题中，给定输入向量x的特征$\phi = \phi(x)$，我们希望分类器能预测该输入所属的类别t。总体来说，分类问题有三种可能的求解方法：

- 区分函数法：设计一个区分函数$f(\phi; w)$，基于某种准则对该函数进行优化，进而得到分类面。代表性方法如Fisher 线性分类函数。与线性拟合类似，这一方法基于人为定义的准则，没有考虑概率意义，但直观简洁。
- 生成性概率模型法：对每个类C_k建立一个统计模型$p(\phi|C_k; w)$，在分类时考察测试样本在每个模型上的概率，再基于贝叶斯公式得到属于某一类的后验概率$P(C_k|\phi)$。这一方法依赖模型假设与实际数据的契合程度，假设越合理，分类性能越好。
- 区分性概率模型法：直接对后验概率$P(C_k|\phi; w)$建模。这一方法不对数据分布做显式假设，只关注分类面，在分类面比较复杂的任务中更有优势。

本节我们从Fisher区分函数开始讨论 [5]。为简便，我们只讨论二分类问题，但相关结论可以扩展到多分类问题中。设N个训练样本$\{(\phi^{(n)}, t^{(n)}) : n = 1, 2, ..., N\}$，其中$t^{(n)} \in \{C_1, C_2\}$，$C_1$和$C_2$为两类。这些点通过一个线性映射投影到一维空间$y$:

$$y = w^T \phi, \tag{2.10}$$

其中w是映射参数。如果基于该训练数据能学习一个优化的w，使得不同类的训练样本在映射空间里的区分性最大，则基于式 2.11即可得到一个简单的分类函数。Fisher准则定义了如下区分性度量：

$$J(w) = \frac{m_2 - m_1}{s_1^2 + s_2^2}, \tag{2.11}$$

其中m_1, m_2是C_1和C_2的样本点在映射空间里的均值，s_1和s_2是相应的方差。式2.11表明类间距离越大，类内的分散程度越小，则依Fisher准则这两类的区分性越强。这显然是符合直觉的。代入公式 2.10，有：

$$J(w) = \frac{w^T S_B w}{w^T S_W w}, \tag{2.12}$$

其中S_W是类间协方差矩阵：

$$S_W = \sum_{\phi^{(n)} \in C_1} (\phi^{(n)} - \mu_1)(\phi^{(n)} - \mu_1)^T + \sum_{\phi^{(n)} \in C_2} (\phi^{(n)} - \mu_2)(\phi^{(n)} - \mu_2)^T,$$

其中μ_1和μ_2是两类样本在原数据空间中的均值。S_B是类内协方差矩阵，定义如下：

$$S_B = (\mu_2 - \mu_1)(\mu_2 - \mu_1)^T.$$

基于式（2.12）对w进行优化，使得$\nabla_w J(w) = 0$，可推出:

$$(w^T S_B w) S_W w = (w^T S_W w) S_B w.$$

注意到$(w^T S_B w)$和$(w^T S_W w)$都是标量，且:

$$\begin{aligned} S_B w &= (\mu_2 - \mu_1)\{(\mu_2 - \mu_1)^T w\} \\ &\propto \mu_2 - \mu_1, \end{aligned} \tag{2.13}$$

则有：

$$S_W w \propto \mu_2 - \mu_1.$$

如果S_w满秩，则有

$$w \propto S_w^{-1}(\mu_2 - \mu_1). \tag{2.14}$$

上式意味着最有区分性的方向w应该是这两类采样点中心连心的方向，但该方向应基于类内方差矩阵进行调整。上述基于Fisher准则的线性模型也被称为线性判别分析（Linear Discriminate Analysis, LDA）。通过设计合理的Fisher准则，LDA很容易扩展到多分类问题 [4]。

Fisher准则的合理性是显然的，但我们碰到了与线性拟合同样的问题：为什么要选择这一准则，其它准则不好吗？为回答这一问题，让我们换一种思路，用线性拟合来求解区分函数。

设类C_1中的样本数为N_1，C_2中的样本数为N_2，二者加起来一共N个训练样本点。对C_1中的样本点，设其目标$t = N/N_1$，对C_2中的样本点，目标为$t = -N/N_2$。则线性拟合的误差函数为：

$$E(w) = \frac{1}{2}\sum_{\phi^{(n)}\in C_1}(w^T\phi^{(n)} - N/N_1)^2 + \frac{1}{2}\sum_{\phi^{(n)}\in C_2}(w^T\phi^{(n)} + N/N_2)^2.$$

取$\nabla_w E(w) = \mathbf{0}$，有:

$$\sum_{\phi^{(n)}\in C_1}(w^T\phi^{(n)} - N/N_1)\phi^{(n)} + \sum_{\phi^{(n)}\in C_2}(w^T\phi^{(n)} + N/N_2)\phi^{(n)} = \mathbf{0}.$$

整理可得：

$$(S_W + \frac{N_1N_2}{N}S_B)w = N(\mu_1 - \mu_2).$$

由式 2.13可知，S_Bw 与$\mu_2 - \mu_1$ 同向，这说明：

$$w \propto S_w^{-1}(\mu_2 - \mu_1),$$

而这正是公式 2.14所示的Fisher区分函数的解。

让我们梳理一下上面的推理逻辑：我们希望得到一个分类函数，使得不同类之间的区分性最大，为此我们定义了一个以w为参数的线性映射，并定义了Fisher准则来优化w。在这一过程中，我们并没有过多考虑为什么选择Fisher 准则，只是直觉上觉得这一准则定义了一个合理的区分性标准。然而，经过推导，我们发现在上述二分类问题中，依Fisher准则得到的映射函数和基于线性拟合得到的映射函数是等价的。由上节讨论可知，线性拟合等价于$p(t|\phi)$ 为高斯分布的线性回归，这说明Fisher方法事实上假设了分类任务中的类别标记是高斯的。这显然是不太合理的。

这一高斯分布假设带来分类面上的偏差。如图 2.3所示，其中两类数据点分别标为1和0，均值分别在μ_1和μ_2。正常情况下，每一类在各自均值附近分布。依线性拟合（等价于Fisher准则）优化拟合函数时，学习目标是各样本点到拟合直线的距离平方和最小（该距离如图中虚线所示），得到的分类面如绿线所示。当出现一些奇异点时（标为红色），为了照顾这些分类点，

拟合直线必须非常接近水平（红线），这使得预测对数据变动不敏感，导致类间区分性下降。

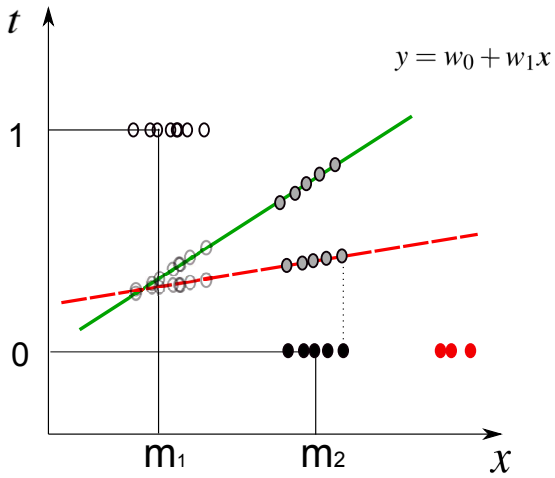

Fig. 2.3 基于线性拟合的分类方法。图中白点和黑点分属两类，数据为一维，两类数据分别标记为0和1。如果没有奇异数据，拟合直线为绿线，如果加入红色奇异数据（同属黑点类），则拟合直线偏向红点，导致类间区分性下降。注意图中拟合直线并非分类面，分类面需依数据点在拟合直线上投影的分布确定，如图中直线上的浅色点所示。

2.1.4 Logistic 回归

基于线性回归的分类方法之所以对奇异点（如图 2.3中的红点）敏感，一个直观的原因是奇异点离拟合直线太远，基于最小平方误差准则使得这些奇异点的影响过大。一个可能的解决方法是利用非线性函数对距离进行压缩，让过远的点产生的影响下降。另一方面，样本点的目标值只能取离散值，基于高斯分布假设的最小平方误差函数显然不合理。Logistic 回归从这两方面对线性回归进行修正，使之适应分类问题。

同样以二分类问题展开讨论。给定一个包括N个样本的训练集$D = \{(\phi^{(n)}, t^{(n)}); \phi^{(n)} = \phi(x^{(n)}), t^{(n)} \in \{0,1\}$，其中$t^{(n)}$取不同值代表不同类。不失一般性，可假设1代表$C_1$ 类、0 代表C_2 类。Logistic 回归假设t符合如下伯努利分布：

$$P(t|\phi;w) = y(\phi;w)^t(1-y(\phi;w))^{1-t}, \tag{2.15}$$

其中$y(\phi;w)$是ϕ属于C_1的预测函数，定义为：

$$y(\phi;w) = p(C_1|\phi;w) = \sigma(w^T\phi), \tag{2.16}$$

其中$\sigma(\cdot)$称为Logistic 函数，定义为：

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

注意$\sigma(\cdot)$将实数域映射到开区间$(0,1)$，可起到非线性压缩作用。Logistic 函数在机器学习里应用非常广泛，我们在后续章节会陆续看到。注意预测函数2.16非常接近线性模型，只不过在线性预测结果后加入了一个非线性压缩。基于Logistic 函数的简单性，我们可以“近似”认为这一模型依然是一种线性模型，或称为“近线性模型”。

和线性回归一样，公式 2.15和式 2.16定义了一个生成过程：首先对输入x经过一个非线性映射$\phi(\cdot)$生成特征，再经由一个线性映射$w^T\phi$投影到一个标量空间，再经过$\sigma(\cdot)$压缩到$(0,1)$之间，最后把该压缩值作为伯努利分布的参数生成目标t。这一模型称为Logistic回归模型。比较Logistic模型和线性回归模型，可见二者具有相似性，差别只是一个非线性映射函数$\sigma(\cdot)$和伯努利分布假设。

基于上述回归模型，可以用最大似然估计来优化模型参数w。首先，定义y_n为：

$$y^{(n)} = \sigma(w^T\phi^{(n)}) = p(C_1|\phi^{(n)}).$$

依公式 2.15，一个样本点$(x^{(n)},t^{(n)})$ 的概率可形式化写成如下形式：

$$p(t^{(n)}|\phi^{(n)},w) = \{y^{(n)}\}^{t^{(n)}}\{1-y^{(n)}\}^{1-t^{(n)}}.$$

则在数据集D上的似然函数可以表示为：

$$p(D;w) = \prod_{n=1}^{N}\{y^{(n)}\}^{t^{(n)}}\{1-y^{(n)}\}^{1-t^{(n)}}.$$

为计算方便，取上述似然函数的负对数作为优化目标：

$$L(w) = -\ln p(D;w) = -\sum_{n=1}^{N}\{t^{(n)}\ln y^{(n)} + (1-t^{(n)})\ln(1-y^{(n)})\}.$$

这一目标函数称为交叉熵。对该函数取对w的梯度，并利用关系$\sigma'(a) = \sigma(a)(1-\sigma(a))$，整理后可得:

$$\nabla_w E(w) = \sum_{n=1}^{N} (y^{(n)} - t^{(n)})\phi^{(n)}. \tag{2.17}$$

这意味着交叉熵函数对w的梯度取决于预测值和目标值之间的误差。类似形式在线性回归里也出现过，见式 2.8。注意的是，将上述梯度取零并不能直接得到w，因为其中$y^{(n)}$是w的非线性函数。但式 2.17中给出的梯度计算方法已经足够我们采用梯度下降法来对w逐步求精了。

梯度下降法（Gradient Descend, GD）是一种通用的函数优化方法。设有函数$f(w)$，优化的目标是找到一个w^*使得该函数取值最小。梯度下降法从一个随机的x开始进行迭代优化，每一步t选择一个使$f(w)$下降最大方向，并往该方向前进步长η_t。因为使$f(w^t)$下降最大的方向即是$f(w)$在w^t 点的梯度方向，因此该方法称为梯度下降法。如果步长η_t选的合理，梯度下降法可以保证收敛到局部最优。

利用GD对Logistic回归中的交叉熵函数$E(w)$进行优化，每一步迭代的参数更新公式如下：

$$w^{t+1} = w^t + \eta_t \nabla_w E(w^t).$$

一般来说，η的取值最初比较大，随着迭代的进行可以逐渐减小，直到某一小值后保持固定。

图 2.5给出基于线性回归（Fisher 区分函数）和Logistic 回归的在两组二维数据上生成的分类面。当数据分布比较正常，没有奇异值时，可以看到这两种分类方法效果很相似，但当某一类中出现奇异数据时，Logistic 回归受到的影响要小的多。

2.1.5 Softmax回归

同样的方法可应用到多分类问题上，称为Softmax回归。基于多分类问题的性质，Softmax回归将目标t由二分类问题中的伯努利分布扩展到多项分布（Multinomial Distribution），相应的表示方法也由0-1表示扩展为One-hot表示。设类别为K，则将ϕ的目标写成K 维二值向量t，其中只有ϕ所属类别对应的维度k为1，其余维度为0。基于这一表示，(ϕ, t)的概率可写成：

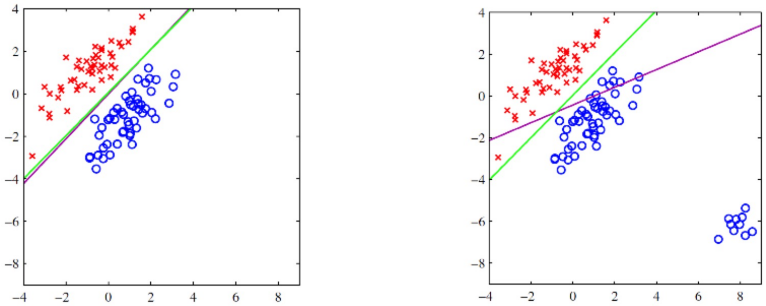

Fig. 2.4 绿色直线和紫色直线分别是Logistic回归和线性回归模型对两类二维数据生成的分类面。左图中的两组数据分布都比较集中，这两个模型给出的结果相差不大；右图中出现了一些离分类面较远的奇异样本，此时线性回归模型的表现明显变差，而Logistic回归模型受到的影响要小的多。

$$p(\phi,t)=\prod_{k=1}^{K}\{y_k\}^{t_k}, \tag{2.18}$$

其中$y_k=y_k(\phi;w)$是ϕ属于第k类的概率，t_k是t在第k维的取值。注意上式是以$(y_1,y_2,...,y_k)$为参数的多项分布。Softmax 回归定义一个近似线性的概率预测模型如下：

$$y_k(\phi;w)=p(C_k|\phi;w)=\frac{e^{w_k^T\phi}}{\sum_j e^{w_j^T\phi}} \quad k=1,2,...,K$$

其中w_k是对第k类数据进行预测的参数。上式最右侧公式称为Softmax函数。和Logistic函数类似，Softmax函数也是一种非线性归一函数，可将多个类的线性预测输出归一化到$(0,1)$之间，因而适合描述概率值。

下面我们求参数w的最大似然估计。由单样本概率公式2.18可知对w的似然函数如下：

$$p(D;w)=\prod_{n=1}^{N}\prod_{k=1}^{K}p(C_k|\phi^{(n)})^{t_k^{(n)}}=\prod_{n=1}^{N}\prod_{k=1}^{K}[y_k^{(n)}]^{t_k^{(n)}},$$

其中$y_k^{(n)}=y_k(\phi^{(n)})$。对上式取负对数得到多分类问题的交叉熵误差函数：

$$E(w)=-\ln p(D;w)=-\sum_{n=1}^{N}\sum_{k=1}^{K}t_k^{(n)}\ln y_k^{(n)}. \tag{2.19}$$

取上述误差函数对w的梯度。注意到Softmax函数的导数有如下形式：

$$\frac{\partial y_k}{\partial a_j} = \begin{cases} y_k(1-y_j) & j=k \\ -y_k y_j & j \neq k \end{cases}$$

其中$a_j = w_j^T \phi$。由此可得到$E(w)$ 对w的梯度如下：

$$\nabla_{w_j} E(w) = \sum_{n=1}^{N} (y_j^{(n)} - t_j^{(n)}) \phi^{(n)}.$$

这一梯度公式和线性回归及Logistic回归的梯度公式具有同一格式，都意味着这样一个事实：误差函数的梯度之所以产生，是因为训练样本的预测值和目标值之间存在误差，并且这些误差不能通过数据平均完全抵消。和Logistic回归一样，我们要依赖梯度下降算法实现对交叉熵误差函数的最小化。

至此我们已经介绍了最基本的线性模型。这些模型虽然简单，却是分析更复杂模型的基础。我们特别强调模型的概率意义。传统基于朴素优化准则（如平方误差和Fisher准则）的模型方法是直观的，但对模型本身的理解并不深入。引入概率意义以后，我们才发现隐藏在各种优化准则背后的分布假设。认识这些假设对我们理解某种方法的优点和不足有重要意义。例如在图 2.5所示的分类例子中，Fisher准则之所以在奇异数据上性能下降，是因为这种方法对数据做了高斯假设，这种假设与实际数据偏离的越远，模型就越不适合。当我们将高斯假设换成伯努利假设后，这种偏差就被纠正了。如果我们对Fisher准则背后的假设不了解，则很难发现这一问题并对其进行修正。从另一个角度看，这也提示我们在学习某种模型和算法时，理解其基础假设要比推导优化公式更重要。

我们还要特别强调一下本节中介绍的最大似然（ML）准则。上面我们谈到各种对应关系、数据分布假设等，到目前为止都是基于最大似然准则，即模型对训练数据的概率最大化。最大似然显然是一种理性选择，但并不完美。例如，我们经常希望将知识和数据结合在一起，这时就不仅要考虑训练数据概率最大化，还要考虑先验知识在目标函数中的比重，这时的准则就不再是最大似然，而是最大后验（Maximum A Posterior，MAP）。事实上，并没有哪种学习准则是绝对最优的，重要的是面对不同任务时选择恰当的准则，并根据数据的分布情况选择恰当的模型。

2.2 线性概率模型

前一节讨论的线性预测模型假设输入变量和输出变量是可见的，基于输入变量对目标变量进行预测。这事实上是建立一个条件概率模型$p(t|x)$。现在我们考虑一个反向问题：如果给我们一个t，是否可以推理出x呢？假设x和t都是可见变量，则可基于前一节所述的线性预测模型对$p(x|t)$建模。然而，在多数推理任务中x是不可见的，即隐变量（Latent Variable），这时无法显式建立一个t对x的预测模型，因而无法对x 直接预测。为解决这类推理问题，我们通常对$p(x)$和$p(t|x)$的分布形式进行假设，训练时基于最大似然准则求解这些分布的参数，推理时利用贝叶斯公式计算后验概率$p(x|t)$来预测x。

可以将上述过程形式化。首先将一维观察变量t扩展到多维变量，并定义x和t之间具有如下线性关系：

$$t = Wx + \varepsilon,$$

其中W是参数矩阵,上式中t,x,ε都是随机变量，分别代表观察量、隐变量和随机噪声。这里我们没有考虑x上的变换$\phi(\cdot)$，因为x是被推理的隐含变量，加入变换将改变模型的性质。在t上进行变换是可能的，相当于特征提取过程。可以将ε和x写在一起，则得到更简洁的线性形式：

$$t = Wx.$$

这一模型称为线性概率模型。

注意线性概率模型和线性预测模型具有类似形式，区别在于线性概率模型中x是不可见的随机变量，而在线性预测模型中x是可见的确定输入。这一区别很重要：当x变成隐变量以后，我们能观察到的数据只有t，因此学习方式由监督学习变成了无监督学习，推理过程也由前向预测变成了反向推理。

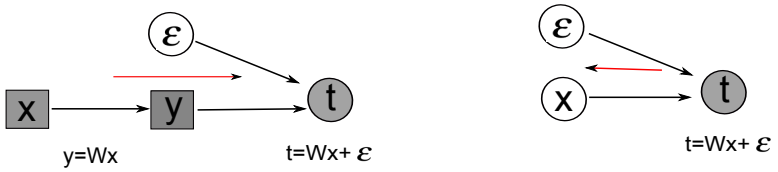

Fig. 2.5 线性预测模型（左）和线性概率模型（右）的区别。图中方框表示确定值，圆圈表示随机变量，白色表示隐含变量，灰色表示可见值或变量。如图中红线所示，线性预测模型的推理过程从左到右，线性概率模型的推理过程由右向左。

线性概率模型建立了一种对观察变量t的概率描述框架，在这一框架下，每当观察到一个t时，可以依定义好的线性关系去推理数据背后的隐藏原因x。因此，该模型被广泛用于因子分析和特征提取等任务中。在后面章节我们会看到，线性概率模型是概率图模型的简单形式。我们将这一模型放到本章讨论，目的是要强调不同模型之间的相关性，即使是线性拟合和因子分析这两种很不相同的方法也具有很强的内在联系。

2.2.1 主成分分析

我们从主成分分析（Principal Component Analysis，PCA）开始讨论。PCA希望通过找到若干相互正交的方向，使得观察数据在这些方向上的映射最大可能地代表原数据的分布性质。每个方向称为一个主成分（Principle Component，PC）。依对原数据的代表能力排序，称为第一主成分，第二主成分等。一般来说，我们希望找到最具有代表性的几个主成分来代表数据，主成分的个数一般远小于数据维度，因此PCA可用在数据降维中。

考虑如下两种准则来评价主成分对数据的代表性：一是数据在主成分代表的映射空间里方差最大，二是由映射恢复成原始数据的损失最小。这两种准则事实上是等价的。下面我们从映射空间方差最大这一准则推导出PCA基本公式，该准则意味着希望在主成分空间的投影尽量保持原数据的分散性。

我们从寻找第一主成分开始。设包括N个样本的训练数据集$D = \{t^{(n)} \in R^D : n = 1,2,...,N\}$，现在的任务是要寻找一个方向$v_1 \in R^D$，使得数据集$D$在这一方向上的投影方差最大。不失一般性，令$v_1$为单位向量，即$v_1^T v_1 = 1$。数据集在投影空间的方差计算如下：

$$\begin{aligned} Var(v_1) &= \frac{1}{N}\sum_{n=1}^{N}(v_1^T t^{(n)} - v_1^T \bar{t})^2 \\ &= \frac{1}{N}\sum_{n=1}^{N} v_1^T (t^{(n)} - \bar{t})(t^{(n)} - \bar{t})^T v_1 \\ &= v_1^T \{\frac{1}{N}\sum_{n=1}^{N}(t^{(n)} - \bar{t})(t^{(n)} - \bar{t})^T\} v_1 \\ &= v_1^T S v_1, \end{aligned} \tag{2.20}$$

其中$\bar{t}$为训练样本集的均值，S是协方差矩阵。求使$Var(v_1)$最小化的v_1 且满足约束条件$v_1^T v_1 = 1$。基于拉格朗日乘子法，上述带约束的优化任务等价于对如下目标函数的无约束优化：

$$\Psi(v_1) = Var(v_1) + \lambda_1(1 - v_1^T v_1) = v_1^T S v_1 + \lambda_1(1 - v_1^T v_1). \tag{2.21}$$

求上式对v_1的梯度：

$$\nabla\Psi(v_1) = 2Sv_1 - 2\lambda_1 v_1 = 0.$$

整理可得下式:

$$Sv_1 = \lambda_1 v_1.$$

上式说明满足优化条件的主成分方向v_1必然是协方差矩阵S的特征向量。将上式代入优化目标函数，有：

$$Var(v_1) = v_1^T S v_1 = \lambda_1.$$

上式说明数据集在v_1上投影的方差等于v_1 对应的特征向量λ的大小。因此，为取使$L(v_1)$最大的主成分，只需取协方差矩阵的最大特征值所对应的特征向量即可。

取得第一主成分后，依类似步骤可得到后续各主成分。设已经得到前$k-1$个主成分，现在寻找第k个主成分v_k，使得数据在v_k上的投影方差最大，且v_k与前$k-1$个主成分正交。基于这些目标和条件，得到类似式 2.21的目标函数：

$$L(v_k) = v_k^T S v_k + \lambda_k(1 - v_k^T v_k) + \sum_{i=1}^{k-1} \lambda_i v_k^T v_i.$$

取对v_k的梯度：

$$\nabla L(v_k) = 2Sv_k - 2\lambda_k v_k + \sum_{i=1}^{k-1} \lambda_i v_i = 0,$$

将上式两端分别左乘$v_i^T(\forall i < k)$，依正交性可得$\lambda_i = 0(\forall i < k)$，因此得到与求第一主成分一样的形式，即$v_k$为协方差矩阵$S$的特征向量，对应的特征向量为$\lambda_k$。由此可知，所有主成分都是协方差的特征向量，且加入每一个主成分

后所增加的方差等于该主成分对应的特征值。因此，若要求前K 个主成分，只需选择特征值最大的K个特征向量即可。

2.2.2 概率主成分分析

PCA是经典的无监督学习方法，广泛应用于降维、正规化、流形学习等任务中。然而，PCA的优化函数（映射空间方差最大或恢复误差最小）和主成分之间的正交限制很大程度上是种人为定义，这使得PCA的适用性缺少明确解释。本节我们将寻求PCA的概率意义，用线性概率模型来解释PCA，这一方法称为概率主成分分析（Probabilistic PCA，PPCA） [11]。

我们考虑如下简单的线性概率模型：

$$t = \mu + Wx + \varepsilon, \tag{2.22}$$

其中$t \in R^D$是D维观察变量，$\mu \in R^D$是固定偏移量，W是模型参数。$x \in R^M$是符合正态分布的M维隐含变量:

$$p(x) = N(x|\mathbf{0}, \mathbf{I}),$$

$\varepsilon \in R^D$ 是D维高斯变量：

$$p(\varepsilon) = N(x|\mathbf{0}, \sigma^2\mathbf{I}).$$

上式定义了数据t的生成模型：首先基于先验概率$p(x)$ 生成隐变量的采样点x，再通过线性变换生成Wx，之后将变换后的采样点位移μ，最后加入一个高斯噪音ε。这一过程如图 2.6所示。值得一提的是，这一模型中所有变量都是高斯的：x是一个高斯变量，其线性变换$\mu + Wx$也是高斯的，再加入一个高斯噪音依然是高斯的。因此，实际观察到的数据t也符合高斯分布，如图 2.6所示。这种基于高斯分布的线性概率模型通常称为线性高斯模型。我们本节所讨论的模型都属于这一类型。

仔细观察PPCA的模型形式 2.22，可以看到这一方程和线性拟合十分相似，但x不论在训练还是推理时都是不可见的，我们所能知道的只是一个假设的先验概率$p(x)$。另外，注意μ是一个模型参数，而非随机变量。

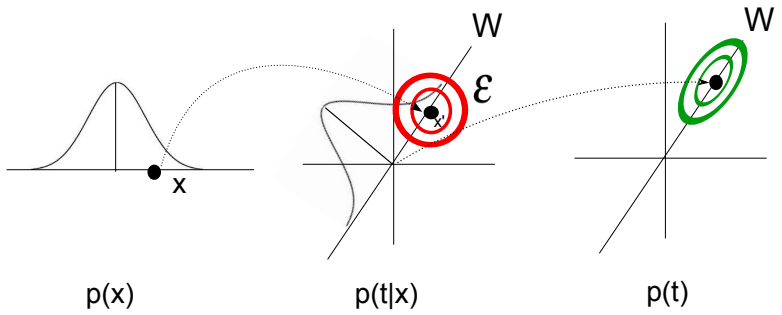

Fig. 2.6 PPCA模型，其中观察数据t 是二维变量，x是一维变量。首先根据隐变量x的先验分布$p(x)$得到一个样本点$\hat{x}$，之后对该样本点做线性变换$x' = Wx + \mu$，最后加入均值为$\mathbf{0}$、协方差为$\sigma^2\mathbf{I}$高斯噪声（图中的红色圆圈），得到观察值t的一个取值。绿色的椭圆表示边缘分布$p(t)$的密度轮廓线。图片参考 [2]的图12.9。

下面我们基于最大似然准则估计PPCA的模型参数W，μ和σ^2。基于PPCA的模型假设，$p(x)$和$p(t|x)$都服从高斯分布，因而边缘分布$p(t)$也服从高斯分布，其均值可由期望得到：

$$\mathbb{E}[t] = \mathbb{E}[\mu + Wx + \varepsilon] = \mu, \tag{2.23}$$

其中利用了x和ε的均值为零的事实。类似，方差可由t的协方差矩阵得到：

$$\begin{aligned} \mathrm{Cov}(t) &= \mathbb{E}[(Wx+\varepsilon)(Wx+\varepsilon)^T] \\ &= \mathbb{E}[Wxx^TW^T] + \mathbb{E}[\varepsilon\varepsilon^T]\,. \\ &= WW^T + \sigma^2\mathbf{I} \end{aligned} \tag{2.24}$$

由此可得：

$$p(t) = N(t|\mu, C) = N(t|\mu, WW^T + \sigma^2\mathbf{I}).$$

基于上述公式，基于最大似然准则可确定模型的参数。给定N个观测点的数据集$D = \{t^{(n)} : n = 1, 2, ..., N\}$，对应的对数似然函数为：

$$\begin{aligned} \ln p(D|\mu, W, \sigma^2) &= \sum_{n=1}^{N} \ln p(t^{(n)}|\mu, W, \sigma^2) \\ &= -\frac{ND}{2}\ln(2\pi) - \frac{N}{2}\ln(|C|) - \frac{1}{2}\sum_{n=1}^{N}(t^{(n)}-\mu)^T C^{-1}(t^{(n)}-\mu) \end{aligned}.$$

Tipping 和Bishop 证明 [11]，对上述似然函数进行优化可得如下最大似然估计：

$$\mu_{ML} = \bar{t}$$

$$W_{ML} = U_M(L_M - \sigma^2\mathbf{I})^{1/2}R$$

$$\sigma^2_{ML} = \frac{1}{D-M}\sum_{i=M+1}^{D}\lambda_i.$$

其中，$U_M \in R^{D\times M}$由原始数据协方差矩阵S的前M个最大特征值对应的特征向量排列而成，$L_M \in R^{M\times M}$是对应的特征值组成的对角矩阵，$R \in R^{M\times M}$是任意正交矩阵。注意正交矩阵R是任意的，不会对数据分布$p(t)$产生影响，这一特性来源于先验概率$p(x)$的各向同性。

基于上述生成模型，可以从观察变量t推理出隐变量x，即后验概率$p(x|t)$。由于先验概率$p(x)$和条件概率$p(t|x)$都是高斯的，可以确定该后验概率亦具有高斯分布形式。依贝叶斯定理，可得：

$$p(x|t) = \frac{p(t|x)p(x)}{p(t)} = N(x|M^{-1}W^T_{ML}(t-\mu_{ML}), \sigma^{-2}_{ML}M),$$

其中

$$M = W^T_{ML}W_{ML} + \sigma^2_{ML}\mathbf{I}.$$

基于该后验概率，对给定t，可基于最大后验(MAP)准则确定一个最能描述t的隐变量取值x。由于该后验概率是高斯的，MAP估计等同于均值，因而有：

$$x_{MAP}|t = M^{-1}W^T_{ML}(t-\mu_{ML}) = (W^T_{ML}W_{ML} + \sigma^2_{ML}\mathbf{I})^{-1}W^T_{ML}(t-\mu_{ML}).$$

当$\sigma_{ML} \to 0$时，可推导出x的MAP估计如下:

$$\begin{aligned} x_{MAP}|t &= (W^T_{ML}W_{ML})^{-1}W^T_{ML}(t-\mu_{ML}) \\ &= R^T L^{-1/2}_{ML}U^T_{ML}(t-\mu_{ML}). \end{aligned}$$

上式表明PPCA 在对原始数据t进行MAP估计时，首先将其映射到和传统PCA同样的子空间（U_{ML}由协方差矩阵的前M个特征向量组成），再通过一个对角阵$L^{-1/2}_{ML}$进行尺度调整以便得到的x 具有各向同性。这说

明PCA是PPCA在$\sigma_{ML} \to 0$时的特殊形式。从另一个角度看，这说明PCA事实上假设了一个线性高斯模型，基于这一模型，观察数据由一个简单的各向同性的正态分布经过一个线性变换得到，因此观察变量t也应该符合高斯分布。当这一条件不满足时，PCA的适用性将会降低。例如，当数据t明显不符合高斯分布时，PCA的结果可能会产生较大偏差。

2.2.3 概率线性判别分析

PCA描述数据的整体分布特性，不考虑不同类数据的不同分布。本节对该模型做简单扩展，使其可以对有类别标记的数据进行建模。为了表达的更清楚，我们将PPCA的模型公式 2.22重写如下：

$$t = \mu + Wx + \varepsilon$$

$$x \sim N(\mathbf{0}, \mathbf{I}), \ \ \varepsilon \sim N(\mathbf{0}, \sigma^2\mathbf{I}).$$

该模型描述了如下生成过程：首先从高斯分布中得到x，再经过一个线性映射$\mu + Wx$，最后再入高斯噪声ε。基于最大似然准则，可以对参数$\{\mu, W, \sigma^2\}$进行估计。

注意上述模型中的t没有区分类别，因此是一个标准描述型模型（对数据的分布属性进行描述）。现在考虑t属于不同类，每一类C_k的均值μ_k各不相同，则上述生成模型可写成：

$$t = \mu_k + Wx + \varepsilon \tag{2.25}$$

$$x \sim N(\mathbf{0}, \mathbf{I}), \ \ \varepsilon \sim N(\mathbf{0}, \sigma^2\mathbf{I}).$$

同样基于最大似然准则，可对参数$\{\mu_k\}, W, \sigma^2$进行估计。注意在上式中，不同类别的数据仅是均值不同，但方差共享。这一模型事实上等价于前一节讨论过的线性区分性分析（LDA） [8]。因此，LDA可以认为是一个多类生成模型，其中每一类数据用一个线性高斯模型表示，且所有类共享一个线性映射矩阵W，意味着所有类的协方差矩阵都是$W^TW + \sigma^2$。

上述模型中每一类的中心μ_k是模型参数，因此模型参数量随训练数据中的类别增加而增长，这种模型称为非参数模型（No-Parametric Model）。这一模型有两个严重问题：一是如果数据中包含的类很多，则不仅计算开销增加，对数据的利用也不够充分（例如对那些样本很小的类，基于该模型得到

的μ_k会产生偏差）；二是无法处理在测试时遇到的新类，因为这些类在训练数据中没有出现过，因此可能不适合当前模型。

概率LDA（Probabilistic LDA，PLDA）将μ_k看作一个随机变量（而非模型参数）解决了上述问题 [7, 9]。在PLDA中，首先由一个先验概率采样出某一类（设为C_k）的均值μ_k，再由一个线性高斯模$\mu_k + Wx + \varepsilon$生成该类的所有采样。在实际应用中，类中心μ_k通常代表某种物理性质（如人脸、声音等），因此通常限制在一个低维空间上。由此，PLDA可形式化成如下公式：

$$t = \mu + Fu_k + Wx_{kj} + \varepsilon \tag{2.26}$$

$$u_k \sim N(\mathbf{0}, \mathbf{I}),\ x \sim N(\mathbf{0}, \mathbf{I}),\ \varepsilon \sim N(\mathbf{0}, \sigma^2\mathbf{I}).$$

注意上式中的u和x是不同于低维空间的两个随机变量，其中u代表类间差异，x代表类内差异。基于该模型，μ_k不再是参数，而是隐变量，模型参数量不再和训练样本中的类别数相关，因此是一个严格的生成模型。给定一个测试数据，基于上述生成模型可推理得到后验概率$p(\mu|t)$，该后验概率代表t的类别属性，因而可用作分类任务中的特征向量。注意这一推理与数据的类别没有直接关系，因为在模型中没有和类别相关的参数，所有参数对所有类都是共享的。

图 2.7给出PLDA模型的生成过程：首先从一个一维正态分布采产生成一个类中心u_k，经过线性映射$u'_k = \mu + Fu_k$在二维空间上生成类k的中心变量u'_k。基于此中心，依PPCA 的生成过程得到该类的所有样本数据。具体来说，首先由一个高斯随机变量依一维正态分布生成随机数x，经过一个线性变换W映射到二维空间，得到Wx，加入高斯随机变量ε，得到$Wx + \varepsilon$，最后与类中心u'_k相加，即可得到该类的一个采样数据t。这一采样过程重复进行，即可得到该类的所有数据。注意上述对类k的数据生成基于一个固定的类中心u_k，基于该类中心，类内数据符合一个高斯分布。

上述PLDA模型的参数$\{\mu, F, W, \sigma\}$可基于最大似然准则进行估计。直接求似然函数需要对所有隐变量边缘化（Marginalize），计算上比较困难，可采用期望最大化（Expectation Maximization, EM）算法通过迭代方式求解。该方法是一个迭代优化过程，每一轮迭代分为期望计算（E）和期望优化（M）两步。在期望计算时，利用当前参数求u,x 的后验概率，并基于该后验概率计算似然函数的期望；在期望优化时，对前面得到的期望函数依模型参数进行优化，得到更新后的参数。上述E 步和M 步交替迭代进行，直到收敛。EM算法是解决包含隐变量的概率模型问题的基本方法，可以证明该算法总会收敛到局部最优 [3, 12]。

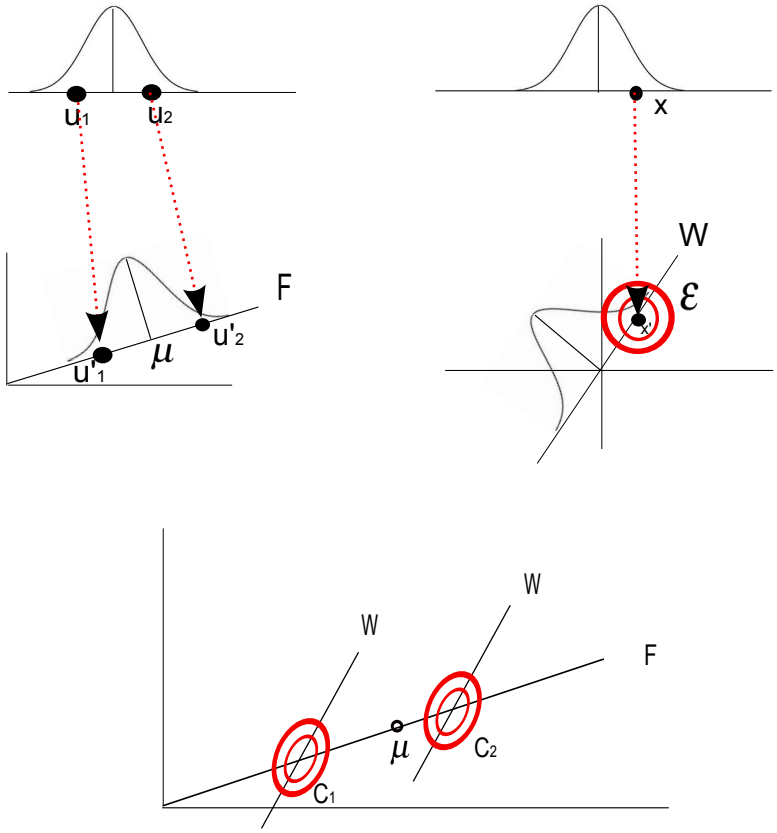

Fig. 2.7 PLDA模型，其中观察数据t 是二维，类中心隐变量u和类内隐变量x都是一维。首先根据隐变量u基于正态分布采样得到类i 的中心向量u_i，经过线性变换得到在数据空间中的中心向量u_i'。基于该类中心，对类中每一个数据点，依下述过程采样生成：首先根据正态分布采样得到一维变量x，经过线性变换得到数据空间中的采样x'作为相对类中心u_i'的偏移量，最后加入高斯噪声ε 后，即得到该类的一个采样点。

对比PLDA和PPCA，可以发现他们在形式上非常相似，都是线性高斯模型。事实上，对于一个特定的类C_k，PLDA事实上是一个对μ 进行随机化的PPCA，即在PPCA采样之前，先对μ_k进行一次采样。二者的优化方法是一样的，都基于最大似然准则。上述相似性让我们从另一个角度认识PCA和LDA这两种看似不同的机器学习工具：PCA用于描述任务，LDA用于分类任务，PCA用于非监督学习，LDA用于监督学习。虽然存在很大差异，但这两种模型都可以归结为相似的概率模型，存在非常密切的联系，具有类

似的优点和缺陷。例如，二者都是线性高斯模型，如果数据不符合高斯分布时，性能都会有显著下降。

2.3 贝叶斯方法

前面几节我们介绍了几种基础的线性模型，这些线性模型通过引入若干随机变量，不仅对数据的分布情况描述的更加准确，而且可以对若干重要模型（PCA,LDA）进行概率解释。不论哪种模型，一个基本假设是模型中的参数是确定的，因而可用各种优化方法进行求解。这种确定性参数的一个缺点是我们无法对它们引入有价值的先验知识。例如，如果我们已经知道某些参数的取值范围，在建模时考虑这一知识会降低模型过训练的风险。

本节将介绍贝叶斯方法，该方法将模型参数看作随机变量，将对参数取值范围的先验知识转化成参数的先验概率。引入随机参数不仅可以利用人们对模型的先验知识，更重要的是对模型本身的改变：对模型的优化不再是寻找一个最优参数（如最大似然估计），而是对参数的后验概率进行估计。因此，即使引入的先验是一个无信息先验（如大范围的平均分布），贝叶斯方法依然有重要价值。

令模型参数为w，先验概率为$p(w)$。基于某一观察数据集D，依贝叶斯定理可估计w的后验概率：

$$p(w|D) = \frac{p(D|w)p(w)}{p(D)}. \tag{2.27}$$

上式中对w先验概率（Prior）$p(w)$代表没有任何数据的情况下对w取值的人为假设；似然函数$p(D|w)$是经验知识，表示在给定一组参数w时，可以生成数据D的概率；后验概率$p(w|D)$ 是对参数w 的再估计，表示在观察到数据D后，对参数w的概率重估。$p(D)$是归一化因子，同时也是各种可能的w得到的似然函数的期望,这一期望称为某类模型的“Evidence”，经常用在模型结构选择中。

基于后验概率$p(w|D)$，可依MAP准则选择最优模型参数，即最大后验估计:

$$w_{MAP} = \arg\max_{w} p(w|D).$$

和最大似然估计相比，最大后验估计考虑了w的先验知识，因此当经验数据D较少时通常可获得更好的估计。当数据量增大时，先验知识占的比重越来越小，最大后验估计趋近于最大似然估计。

最大后验估计依然是点估计，即选择某一确定的参数进行预测和推理。事实上，后验概率$p(w|D)$提供了一种更有价值的预测和推理方式，即在预测或推理时考虑所有可能的参数w，这样得到的结果会更加可靠。这一方法称为贝叶斯方法。以预测任务为例，贝叶斯方法可写成下式：

$$t = \int t p(t|w,x) p(w|D) \mathrm{d}w.$$

下面我们以第一节中介绍的线性回归模型为例来说明贝叶斯方法。与传统线性回归模型类似，定义输入x和输出t之间存在线性关系，不同的是对参数w定义高斯先验：

$$t = w^T x + \varepsilon,$$

$$w \sim N(\mathbf{0}, \alpha^{-1}\mathbf{I}); \quad \varepsilon \sim N(\mathbf{0}, \beta^{-1}\mathbf{I}).$$

上式等价于：

$$p(t|x,w,\beta) = N(t|w^T x, \beta^{-1}\mathbf{I}),$$

$$p(w|\alpha) = N(w|\mathbf{0}, \alpha^{-1}\mathbf{I}). \tag{2.28}$$

给定一个包含N个采样点的数据集$D = \{(x^{(n)}, t^{(n)}); n = 1,2,...,N\}$，由贝叶斯公式可知：

$$p(w|D) \propto p(D|w) p(w|\alpha),$$

由此得到w的似然函数如下:

$$\begin{aligned} \ln p(w|D) &= \ln \prod_{n=1}^{N} p(t^{(n)}|x^{(n)}, w) + \ln p(w|\alpha) + const \\ &= -\frac{\beta}{2} \sum_n \{t^{(n)} - w^T x^{(n)}\}^2 - \frac{1}{2} w^T w + const, \end{aligned}$$

其中$const$是与w无关的常量。最大后验估计即是取使上式最大的w。注意到上式是一个平方误差加上一个二阶量$w^T w$，其中平方误差是标准线性回归模型的目标函数，而二阶量来源于先验概率$p(w)$。如果将该二阶量看作原目标

函数的正则项（Regularization），则引入先验概率等价于在原目标函数上引入一个l_2范数约束，该约束使得优化目标倾向于取值更小的参数，而这正是先验概率$p(w)$取值最大的区域。不同的先验概率等价于不同的正则约束，这从概率模型角度解释了传统模式识别方法中正则约束的作用。

注意式2.28定义的先验概率包含一个参数α，这一参数是参数w的先验概率的参数，因此也称超参数。超参数可以依经验设定，也可以基于最大似然准则学习。注意到$p(D) = \int p(D|w)p(w;\alpha)\mathrm{d}w$是$\alpha$的函数，因此可以将$p(D)$作为似然函数优化$\alpha$。也可对$\alpha$引入一个先验概率，再基于最大后验准则求解$\alpha$，这种方法可称为层次性贝叶斯模型（Hierarchical Bayes Model）。

一般来说，先验概率的形式可以自由选择，但在实际建模时通常希望后验概率越简单越好。一种方法是选择一种先验概率，其与似然函数相乘后得到的后验概率具有和先验概率相同的形式。这一先验概率称为似然函数的共轭先验。共轭先验不仅极大简化了模型计算复杂度，而且提供了一个简单的在线学习框架。在这一框架中，依过去经验D得到的后验$p(w|D)$在下一步学习时成为新的先验，新的数据D'可依此先验进一步学习得到新的后验。这一过程迭代进行，可逐渐学习最新知识。注意这一学习方法的前提是先验概率与条件概率必须共轭的，否则后验概率无法得到与先验概率一致的形式，从而无法形成新的先验。

2.4 本章小结

线性模型是机器学习里最简单的模型，也是最重要的模型之一，是学习其它复杂模型的基础。本章讨论了两种线性模型：一种是输入变量可见的预测模型，一种是输入变量不可见的描述模型。预测模型多用于预测任务，描述模型多用于推理任务；预测模型多用监督学习，描述模型多用于非监督学习。不论哪种模型，都可表示成简单的线性形式$y = Wx$。这些模型对观察值或隐变量做出某种分布假设，并依最大似然准则估计模型参数。

在线性预测模型中，我们讨论了用于回归问题的线性回归模型和用于分类问题的Logistic回归模型。线性回归模型假设目标变量的条件分布$p(t|x)$是一个高斯分布，而Logistic回归模型假设$p(t|x)$是一个伯努利分布。通过讨论发现，线性回归模型通过最大似然估计得到的回归参数与线性拟合得到的拟合参数是一致的，表明这两种模型具有等价性，这为线性拟合找到了合理的概率解释。同时，我们发现传统基于Fisher准则的线性分析模型在二分类问题上也等价于线性回归模型，这相当于对数据的类别标签假设了高斯分布，

显然是不合理的。Logistic回归模型将高斯分布假设修正为伯努利分布假设，因此更适合分类问题。

在线性概率模型中，我们讨论了基于非监督学习的概率PCA（PPCA）方法和基于监督学习的概率LDA（PLDA）方法。类似线性拟合与线性回归模型的关系，PPCA是传统PCA方法的概率形式，而PLDA是传统LDA方法的概率形式。在PPCA方法中，观察数据是由一个正态分布的隐随机变量通过一个线性变换再加上一个高斯噪声生成，这意味着PPCA（及其传统形式，PCA）只适用于符合高斯分布的数据。PLDA 在PCA基础上考虑类间差异。这一模型假设每个类的中心向量由低维空间的一个正态分布经过线性变换得到；得到中心向量后，通过一个PPCA模型生成该类的所有数据。不同类数据共享同一个PPCA模型，因此该模型假设不同类的协方差矩阵是相同的，唯一差别是均值（中心向量）上的不同。和LDA相比，PLDA是将原来作为模型参数的类均值向量修正为随机变量。这一修正具有重要意义，它使得模型参数与数据无关，因此具有更强的泛化能力，可以处理训练数据中没有见过的新类别。

最后，我们讨论了线性模型的贝叶斯扩展，将原来确定性的参数扩展为随机变量，依据先验知识对这些变量设置先验概率。利用贝叶斯公式，这些先验知识可以和经验知识有效结合起来，得到一种更有效的参数估计方法：最大后验估计。更重要的是，贝叶斯方法改变了我们对模型参数的认识：模型参数不一定是一些确定的数值，还可以是一些随机变量。基于这些随机变量的后验概率，考虑各种可能的模型参数，由此可做出更合理的预测或推理。

2.5 相关资源

- 本章对线性预测部分的讨论参考了Bishop的《Pattern Recognition and Machine Learning》第三章、第四章对线性模型的描述。对PCA部分的讨论参考了该书的第十二章关于连续隐变量模型的讨论 [2]。
- 对PLDA的讨论参考了Prince等人的论文《Probabilistic Linear Discriminant Analysis for Inferences About Identity》 [9]。
- 线性模型在各种机器学习和模式识别经典教材中都是基础内容。如Hastie, Tibshirani 和Friedman的《The Elements of Statistical Learning》一书的第三、四章 [6]，周志华《机器学习》一书的第三章 [1]。
- 关于线性高斯模型，可参考Roweis等人在1999 年的综述文章 [10]。

Chapter 3
神经模型

Chapter 4
深度学习

Chapter 5
核方法

Chapter 6
图模型

Chapter 7
非监督学习

Chapter 8
非参数模型

Chapter 9
遗传学习

Chapter 10
强化学习

Chapter 11
优化方法

References

[1] 周志华(2016) 机器学习. 清华大学出版社

[2] Bishop CM (2006) Pattern recognition and machine Learning. Springer

[3] Dempster AP, Laird NM, Rubin DB (1977) Maximum likelihood from incomplete data via the em algorithm. Journal of the royal statistical society Series B (methodological) pp 1–38

[4] Duda R, Hart P (1973) pattern classification and scene analysis. Wiley

[5] Fisher RA (1936) The use of multiple measurements in taxonomic problems. Annals of human genetics 7(2):179–188

[6] Friedman J, Hastie T, Tibshirani R (2001) The elements of statistical learning, vol 1. Springer series in statistics Springer, Berlin

[7] Ioffe S (2006) Probabilistic linear discriminant analysis. Computer Vision–ECCV 2006 pp 531–542

[8] Kumar N, Andreou AG (1998) Heteroscedastic discriminant analysis and reduced rank hmms for improved speech recognition. Speech communication 26(4):283–297

[9] Prince SJ, Elder JH (2007) Probabilistic linear discriminant analysis for inferences about identity. In: Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on, IEEE, pp 1–8

[10] Roweis S, Ghahramani Z (1999) A unifying review of linear gaussian models. Neural computation 11(2):305–345

[11] Tipping ME, Bishop CM (1999) Probabilistic principal component analysis. Journal of the Royal Statistical Society: Series B (Statistical Methodology) 61(3):611–622

[12] Wu CJ (1983) On the convergence properties of the em algorithm. The Annals of statistics pp 95–103