prml

miaoqi shen

Published
with GitBook

目錄

前言

最近在学习机器学习试着去翻译一下PRML 希望能坚持下来

从数据中寻找模式是一个基础问题，它有着很长的成功史。例如：16世纪，Tycho Brahe（第谷·布拉赫）的大量天文观测使得Johannes Kepler（开普勒）发现了行星运动的经验规律。这反过来为经典力学的发展提供的跳板。同样的，原子光谱规律的发现在20世纪初的量子力学的发展和验证起着主要作用。模式识别领域主要关注利用计算机算法来自动发现数据中的规律，以及利用这些规律对数据进行分类等操作。

考虑对 图1.1 中的手写数字进行识别的问题。每一个数可以由28*28个像素的图像也就是784个实数向量来表示。我们的目的是建立一个以这样的向量作为输入并能识别0...9这些数字作为输出的机器。手写的多样性导致这不是一个简单的问题。这个问题可以用人工编写规则或根据数字的笔迹来区分它们，但是实际上这样的方法会导致规则激增及不符合规则的例外等问题，导致得到并不理想的结果。

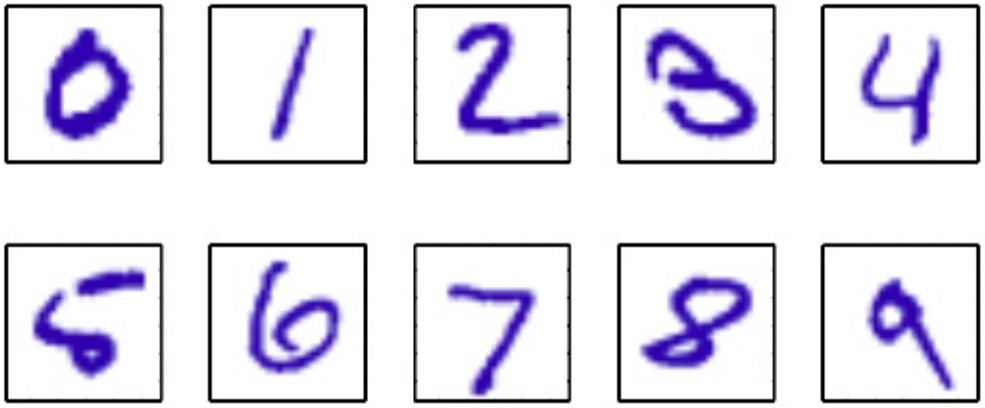

图 1.1: 来自美国邮政编码的手写数字的例子

一种使用 由N 个数字 $\{x_1, x_2, \ldots x_N\}$ 组成的大集合作为训练集（training set）来调节所采用模型的参数的机器学习方法会得到好得多的结果。训练集中的结果通常被我们独立考察、人工标注而事先知道。我们可以通过目标向量（target vector） t 表示并区分这些数字。使用向量来表示类别的技术我们将在后面介绍。注意对每个数字图像 x 只有一个目标向量 t 与之对应。

这种机器算法的结果可以由以新的数字图像 x 作为输入，并输出与目标向量形式相同的向量 y 的函数 $y(x)$ 来表示。$y(x)$ 的精确度是由训练阶段（training phase 也叫学习阶段 learning phase）决定的。一旦模型训练好我们就可以识别测试集合（test set)中新的数字图像。识别与训练集不同的新样本的精确度叫做泛化（generalization）。在实际情况中我们的训练集只是所有可能输入的一小部分，所以泛化是模式识别的中心问题。

在大部分实际应用中，为了能更容易的处理模式识别问题我们会对原始数据做预处理（perprocessed）把它们转换到新的空间。例如：在数字识别问题中，对数字的图像进行变换和伸缩，使得每个数字可以用大小固定的盒子来表达。现在所有数字的位置和大小都相同，这极大地减少了每个数字类的变化性，使后续区别不同类别的模式识别算法更加容易。这样的预处理我们有时也把它称为特征抽取（feature extraction）。注意新的测试数据必须和训练数据做一样的预处理。

有时我们为了加快计算速度也会进行预处理。例如：我们需要实时的从高清视频流中识别出人脸，计算机每秒钟需要处理大量的像素，将这些像素直接传给复杂的模式识别算法在计算上是不可行。因此我们需要找到可以快速计算的有效特征，这些特征保留区别人脸和其他的信息。这些特征被当作模式识别算法的输入。例如：一块矩形子域的图像灰度的平均值是可以很快计算出来的（Viola and Jones, 2004），而且这样的特征被证明在快速人脸识别中是很有效的。因为这样的特征数量是小于像素数的，所以这样的预处理代表了一种维数降低形式。大多数预处理都会导致信息丢失，所以我们必须要格外注意。如果对于问题处理很重要的信息丢失的话会导致系统精度大幅下降。

训练样本同时包含输入向量以及对应的目标向量的应用被称为监督学习（supervised learning）问题。数字识别就是这样的一个问题，像这样把输入向量分配到有限的离散类别中的一个的被称为分类（classification）问题。如果期望的输出是一个或多个连续变量，那么我们就把它称为回归（regression）问题。化工生产过程中的产量的预测就是这样的一个问题，它的输入由反应物、温度、压力组成。

在其他的模式识别问题中,训练数据只由一组输入向量 x 组成,没有任何对应的目标值。这就是无监督学习（unsupervised learning）。它的目标可能是在数据中发现相同的样本分组，我们把这叫做聚类（clustering），或者是把高维空间投影到2或3维空间中，这就叫数据可视化（visualization）。

最后，反馈学习(reinforcement learning)(Sutton and Barto, 1998)技术关注的问题是在给定的条件下,找到合适的动作,使得奖励达到最大值。它不像监督学习一样给定最优输出用例，而是需要在一系列的实验和错误中发现。通常学习算法有一套和环境交互的状态和动作序列。大多数情况下当前的动作同时影响当前和所有后续的奖励。例如：通过合适的反馈学习技术，一个神经网络可以成为西洋双陆棋（Backgammon）的高手(Tesauro, 1994)。这里神经网络必须学习把一大组 位置信息、骰子投掷的结果作为输入,产生一个移动的方式作为输出。这可以通过神经网络自己和自己玩数百万局来完成。主要的挑战来自于奖励是在经过大量的移动使游戏结束后以胜利的形式给出。奖励必须被合理地分配给所有引起胜利的移动步骤。这些移动中,有些移动很好,其他的移动不是那么好。这是信用分配(credit assignment)问题的一个例子。反馈学习的一个重要特征是在探索(exploration)和利用(exploitation)间做一个权衡。“探索”是指系统尝试新类型的动作,“利用”是指系统使用已知能产生较高奖励的动作。过分地集中于探索或者利用都会产生较差的结果。反馈学习一直是机器学 习研究中得一个活跃的领域。然而,详细讨论反馈学习不在本书的范围内。

尽管这样的任务都有自己的工具和技术，但是支撑它们的大部分关键思想还是想通的。本章的主要目标是以一种相对非正式的形式介绍最重要的概念,并且使用简单的例子来说明。在之后，我们将会看到同样的思想以 更加复杂的 能应用于真实世界的模式识别应用 中的模型形式重新出现。本章也将介绍将自始至终在本书中使用的三个重要 工具：概率论、决策论、信息论。虽然这些东西听起来让人感觉害怕,但是实际上它们非常直观。并且,如果想让机器学习技术在实际应用中发挥最大作用的话,必须要清楚地理解它们。

我们以一个贯穿本章用来阐述很多关键思想的简单的回归问题作为开始。假设我们观察到一个实输入变量 x 并利用这个变量来预测我们的实目标变量 t 。对于这个目的，一个很好的方法是考虑使用人工合成的数据，因为这样我们就精确的知道数据的生成过程，从而能够与学习到的模型作比较。这个例子的数据是由函数 $sin(2\pi x)$ 目标变量带有随机噪音。详细的描述见附录A。

现在我们考虑一个由 N 个观测量 x ，被记作 $X \equiv (x_1, \ldots, x_N)^T$,和对应的目标变量t，被记作 $T \equiv (t_1, \ldots, t_N)^T$ 。图1.2展示了由 $N = 10$ 个数据点组成的图像。图1.2的输入集合 X 是通过选择 $x_n (n = 1, \ldots, N)$ 来生成的。x_n 均匀的分布在区间 $[0, 1]$。对应的目标向量t_n是由函数 $sin(2\pi x)$ 加上一个小的符合高斯 分布的随机噪声（高斯分布在1.2.4节中说明）。通过这种方法我们获取到一个拥有潜在的规律性的大量真实数据集，我们希望通过学习获得这种规律性。这个数据是受随机噪声干扰的。噪声可能由本质随机的过程产生，如：放射性衰变。但是更典型的情况是由于存在没有被观察到的变化性的噪声源。

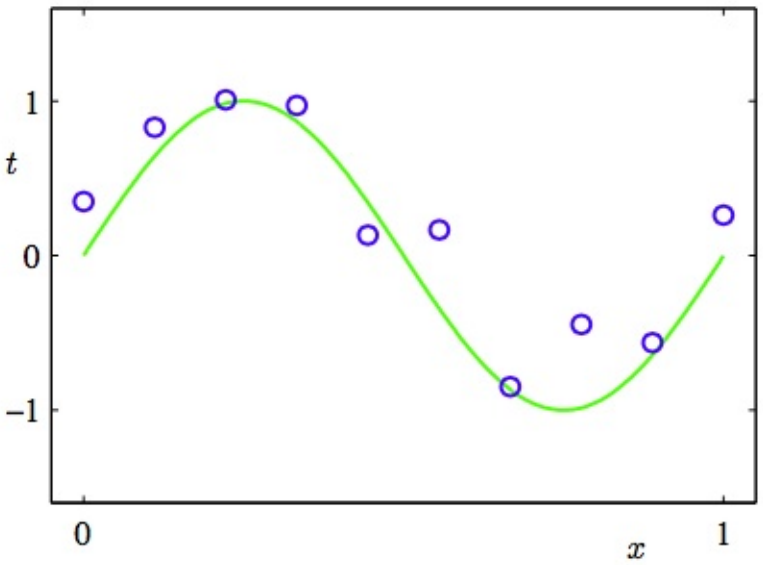

图 1.2: 由 $N = 10$ 个数据点组成的训练集的图像,用蓝色圆圈标记。每个数据点由输入变量x的观测以及 对应的目标变量t组成。绿色曲线给出了用来生成数据的$sin(2\pi x)$函数。我们的目标是对于某些新的x值, 预测t的值,而无需知道绿色曲线。

我们的目标是利用这个训练集，对新的输入变量$\hat{x}$预测出目标变量$\hat{t}$ 。和我们将要看到的一样，这涉及到隐式地发现基础函数$sin(2\pi x)$。这个问题本质上是比较困难的，因为我们需要从有限的数据集中生成。而且我们的数据是受噪声干扰的，这导致对于$\hat{x}$对应的 $\hat{t}$ 具有不确定性。概率论(我们在1.2节中讨论)提供了一个用来以精确的数学的形式描述这种不确定性的框架。决策论（在1.5节中讨论）让我们可以利用这种概率的表示，以合适的标准进行最优的预测。

但是现在,我们用一种简单的曲线拟合来考虑和处理这样的信息。我们将使用下面形式的多项式函数来进行数据拟合。

$$y(x, w) = w_0 + w_1 x + w_2 x^2 + \ldots + w_m x^m = \sum_{j=0}^{m} w_j x^j \tag{1.1}$$

其中m是这个多项式的阶数，x^j 表示x的j次幂。系数$w_0, \ldots w_m$整体记作向量w。 注意,尽管多项式函数$y(x, w)$是一个关于x的非线性函数，但他时关于系数w的线性函数。我们把这样的具有关于未知参数是线性的这一重要性质的多项式称为线性模型。将在第三、四章进行详细讨论。

系数的值通过对训练集关于多项式拟合确定。这可以采用最小化误差函数（error function）的方法来取得。误差函数是对于确定的w值所得到的$y(x, w)$与训练集之间的差别。一种广泛使用的简单的选择是把每一个数据x_n对应的目标值t_n与它

的预测值$y(x_n, w)$之间的平方差的和作为误差函数。所以我们需要最小化：

$$E(w) = \frac{1}{2}\sum_{n=1}^{N} y(x_n, w) - t_n{}^2 \tag{1.2}$$

这个$\frac{1}{2}$因子是为了后面计算方便。在这章的后面环节我们将讨论选择这个误差函数的动机。现在，简单的提示：这是一个非负的量，当且仅当函数$y(x, w)$完全穿过训练集中的每一个点。图1.3阐释了平方和误差函数的几何意义。

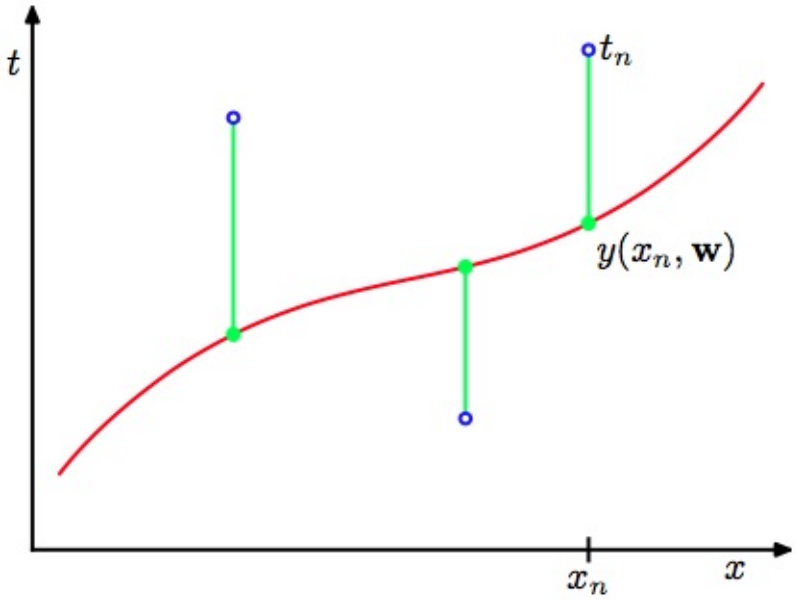

图 1.3: 误差函数(1.2)对应于每个数据点与函数$y(x, w)$之间位移(绿色垂直线)的平方和

我们可以通过选择w的值使得$E(w)$尽可能的小来解曲线拟合问题。因为误差函数是关于系数w的二次函数，所以它的导数是关于系数的线性函数。因此我们可以得到最小化误差函数的唯一解析解，记作w^*。最终的多项式函数由$y(x, w^*)$给出。

那么剩下的问题就是如何选择多项式的阶数m，正如我们将要看到的，这是一个被称为模型对比（model comparison）或模型选择（model selection）的重要概念的一个例子。在图1.4中展示了以图1.2中数据分别使阶数$m = 0, 1, 3, 9$做多项式拟合的例子。

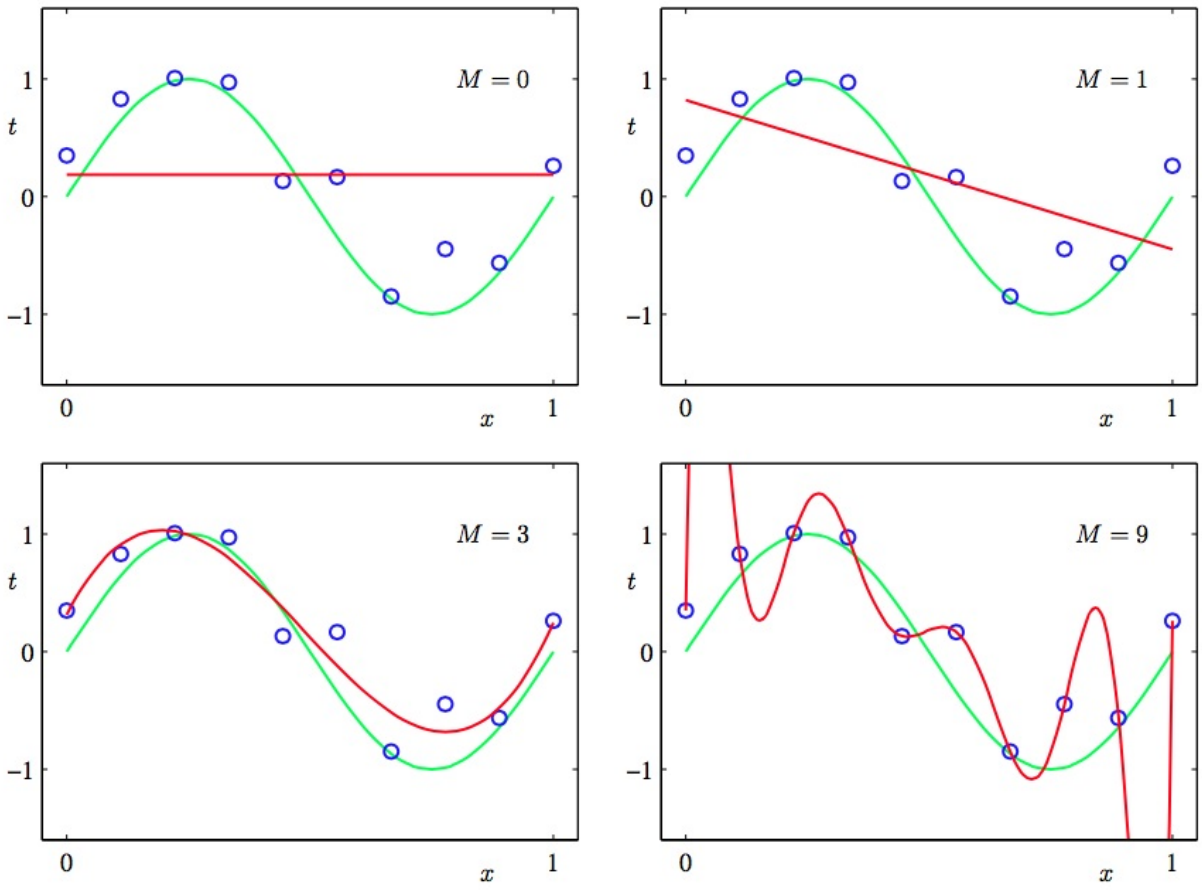

图 1.4: 不同阶数的多项式曲线,用红色曲线表示,拟合了图1.2中的数据集。

我们看到常数($m = 0$)和一阶($m = 1$)多项式对于数据的拟合效果相当差,很难代表函数$sin(2\pi x)$。图1.4中的三阶($m = 3$)多项式似乎对函数$sin(2\pi x)$作了很好的拟合。当使用足够高的阶数（$m = 9$）时，多项式与训练数据完全匹配。实际上多项式函数完全穿过每一个数据点，即$E(w^*) = 0$。然而拟合曲线出现了强烈的震荡，与生成函数$sin(2\pi x)$相差比较大。这就叫做过拟（overfitting）。

正如之前提到的那样，我们的目标是得到对新数据做很好的预测的泛化性。我们可以通过对有100个数据点组成的单独的测试集来定量的分析泛化性与m之间的关系。这100个数据点的生成方式与训练集的生成方式完全相同（当然目标值的随机噪声的取值是随机的）。对于每个m我们可以计算针对训练数据的残差$E(w^*)$，同样的我们也可以计算测试数据的残差$E(w^*)$。有时使用均方根(RMS)误差更方便，它由公式：

$$E_{RMS} = \sqrt{2E(w^*)/N} \tag{1.3}$$

定义。其中的N让我们可以平等的去比较不同大小的数据集。平方根确保了E_{RMS}和目标变量t具有相同的规模（和单位）。图1.5展示的不同的m对应的训练集与测试集的RMS误差。

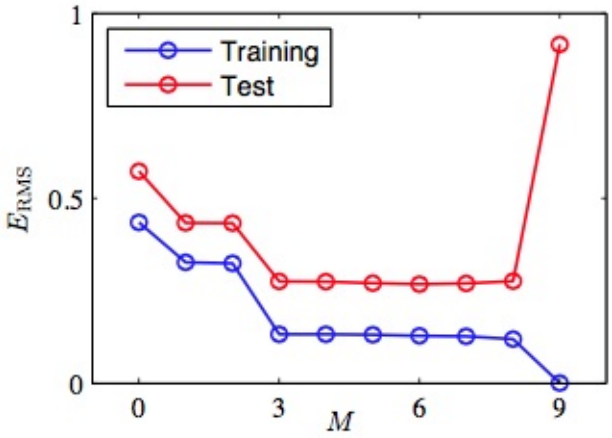

图 1.5: 公式(1.3)定义的根均方误差,在训练数据集上和独立的测试数据集上对于不同的m进行了计算。

测试集误差表示了我们对新的观测值x预测的目标变量t的准确程度。根据 图1.5,得到小的m值会造成较大的测试集误差,这可以归因于对应的多项式函数相当不灵活,不能够反映出$sin(2\pi x)$的震荡。当m的取值为$3 \leq m \leq 8$时,测试误差较小,对于合理的表达生成函数$sin(2\pi x)$。当$m = 9$时，和我们预期的一样训练误差降低到了0，因为此时的多项式的自由度为10，对应10个参数$w_0, \ldots w_9$所有可以调节这些参数使它精确的匹配训练集中的10个数据点。但是正如图1.4中显示的结果函数$y(x, w^*)$图像那样，它产生了强烈的震荡，这使得产生的测试误差非常大。

这可能看起来很矛盾,因为给定阶数的多项式以特殊的形式包含了所有低阶的多项式函数。因此$m = 9$的多项式至少能产生与$m = 3$的多项式一样好的结果。并且可以假设它能更好的预测由函数$sin(2\pi x)$生成的新数据（稍后将会看到确实是这样）。我们知道函数$sin(2\pi x)$的幂级数展开包含所有阶数的项,所以我们会认为结果会随着m的增大而单调地变好。

通过考察不同阶数多项式的系数w^*的值，我们更深入的考察问题。如表1.1所示那样：

	$M = 0$	$M = 1$	$M = 6$	$M = 9$
$w_0^\star$	0.19	0.82	0.31	0.35
$w_1^\star$		-1.27	7.99	232.37
$w_2^\star$			-25.43	-5321.83
$w_3^\star$			17.37	48568.31
$w_4^\star$				-231639.30
$w_5^\star$				640042.26
$w_6^\star$				-1061800.52
$w_7^\star$				1042400.18
$w_8^\star$				-557682.99
$w_9^\star$				125201.43

表 1.1: 不同阶数的多项式的系数w^*的值。观察随着多项式阶数的增加,系数的大小是如何剧烈增大的。

> || || *m = 0* || *m = 1* || *m = 6* || *m = 9* || || w_0^* || 0.19 || 0.82 || 0.31 || 0.35 || || w_1^* || || -1.27 || 7.99 || 232.37 || || w_1^* || || || -25.43 || -5321.83 || || w_1^* || || || 17.37 || 48568.31 || || w_1^* || || || || -231639.30 || || w_1^* || || || || 640042.26 || || w_1^* || || || || -1061800.52 || || w_1^* || || || || 1042400.18 || || w_1^* || || || || -557682.99 || || w_1^* || || || || 125201.43 ||

当m变大时系数的大小（标量）通常也随着变大。特别的，当$m = 9$时为了更好的与训练集数据匹配,系 数取了相当大的正数或者负数。但是没有考虑数据点之间的数据（特别是临近区间端点处的点），如图1.4所示的那样函数出现了强烈的震

荡。直觉的讲，这是由于有着大的m值的更灵活的多项式，针对目标值上的随机噪声过分地调参。
考察给定模型的行为随着数据集规模的变化情况也很有趣,如图1.6所示:

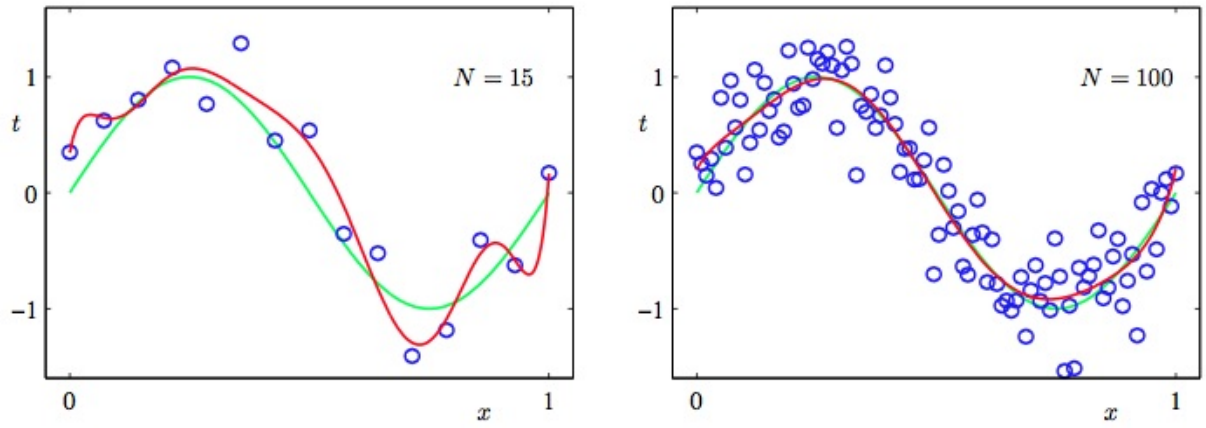

图 1.6: 使用$m = 9$的多项式, $N = 15$个数据点(左图)和$N = 100$个数据点(右图)通过最小化平方和 误差函数的方法得到的解。我们看到增大数据集的规模会减小过拟合问题。

我们可以看到, 对已一个给定的模型复杂度,当数据集的规模增加时,过拟问题变得不那么严重。即：数据集规模越大,能够用来拟合的数据模型就越复杂(即越灵活)。大致的概括，数据点的数量不应该小于模型的可调节参数的数量的若干倍(比如5或10)。然而, 正如在第3章所说的那样,参数的数量对于合理的模型复杂度来说是必要的。

根据训练集的大小来限制参数的数量是令人不满意的。我们更应该根据待解决问题的复杂程度来选择模型的复制度。我们将会看到通过最小二乘来寻找模型参数方法是最大似然（maximum likehood）（在1.2.5节中讨论）的一种特殊形式，过拟问题可以作为最大似然的一个通用属性来理解。采用贝叶斯方法可以避免过拟问题。从贝叶斯的观点来看,模型参数的数量超过数据点数量并不是一个难解的情形。实际上,在贝叶斯模型中,参数的有效(effective)数会自动根据数据集的规模调节。

但现在,继续使用当前的方法，考虑在实际中我们如何应用有限规模的数据集得到相对复杂且灵活的模型，还是很有用的。 正则化(regularization)是一种经常用来控制过拟现象的技术。这种技术涉及到给误差函数(1.2)增加一个惩罚项,使得系数不会取很大的值。采用所有系数的平方和作为惩罚项是一种最简单的形式。推导出了误差函数经简化后得到如下公式：

$$\widetilde{E}(w) = \frac{1}{2}\sum_{n=1}^{N} y(x_n, w) - t_n{}^2 + \frac{\lambda}{2}||w||^2 \qquad (1.4)$$

其中$||w||^2 \equiv w^T w = w_0^2 + w_1^2 + \ldots + w_m^2$，系数$\lambda$是用来控制平方和误差中比较重要的正则化项的。注意,通常系数w_0从正则化项中省略,因为包含w_0会使得结果依赖于目标变量原点的选择（Hastie et al., 2001）。如果要把它包含在正则化项中，它就需要有自己的正则化系数（在5.5.1节详细讨论这个问题）。同样的，公式(1.4)中的误差函数也可以用解析的形式求出最小值。因为这种方法减小了系数的值，意思在统计学文献中它被称为收缩(shrinkage)方法。二次正则项的一种特殊情况被称为山脊回归(ridge regression)(Hoerl and Kennard, 1970)。在神经网络中,这种方法被叫做权值衰减(weight decay)。

图1.7展示了在$m = 9$的情况下,使用的公式(1.4)的正则化误差函数,用与之前相同的数据做多项式拟合的结果。

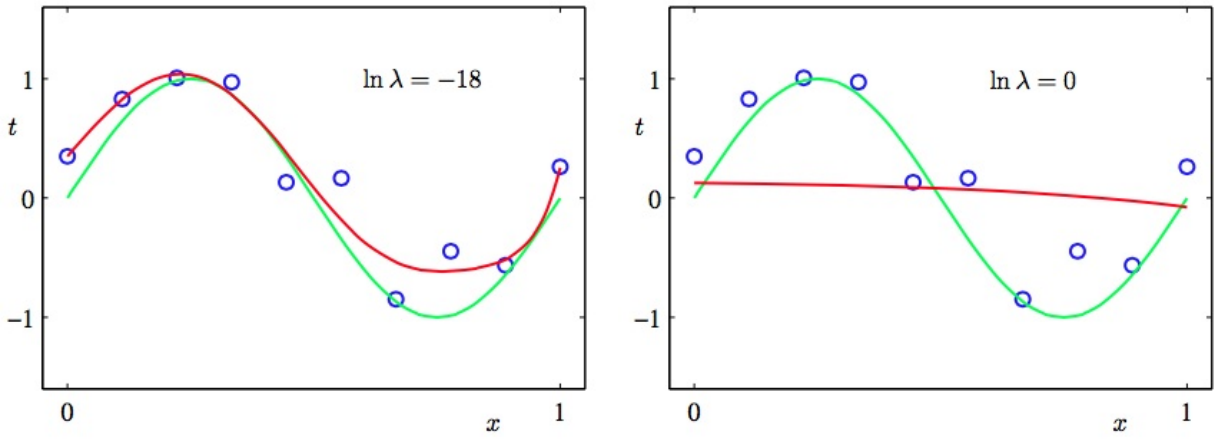

图 1.7: 使用正则化的误差函数(1.4)

我们看到,对于$ln\lambda = -18$,过拟合现象被压制,我们可以很好的模拟生成函数 $sin(2\pi x)$。当 λ的值取太大的时候，我们的拟合又变差的（如图1.7中$ln\lambda = 0$的图所示）。表1.2中给出了拟合的多项式的系数，表明正则化在减小系数的值方

	$\ln\lambda = -\infty$	$\ln\lambda = -18$	$\ln\lambda = 0$
$w_0^\star$	0.35	0.35	0.13
$w_1^\star$	232.37	4.74	-0.05
$w_2^\star$	-5321.83	-0.77	-0.06
$w_3^\star$	48568.31	-31.97	-0.05
$w_4^\star$	-231639.30	-3.89	-0.03
$w_5^\star$	640042.26	55.28	-0.02
$w_6^\star$	-1061800.52	41.32	-0.01
$w_7^\star$	1042400.18	-45.95	-0.00
$w_8^\star$	-557682.99	-91.53	0.00
$w_9^\star$	125201.43	72.68	0.01

面产生了预期的效果。
表1.2

图1.8展示了关于$ln\lambda$的训练集和测试集的RMS误差，用来表达正则化项对于泛化错误的影响。我们看到,在效果上,λ控制了模型的复杂性,因此决定了过拟合的程度。

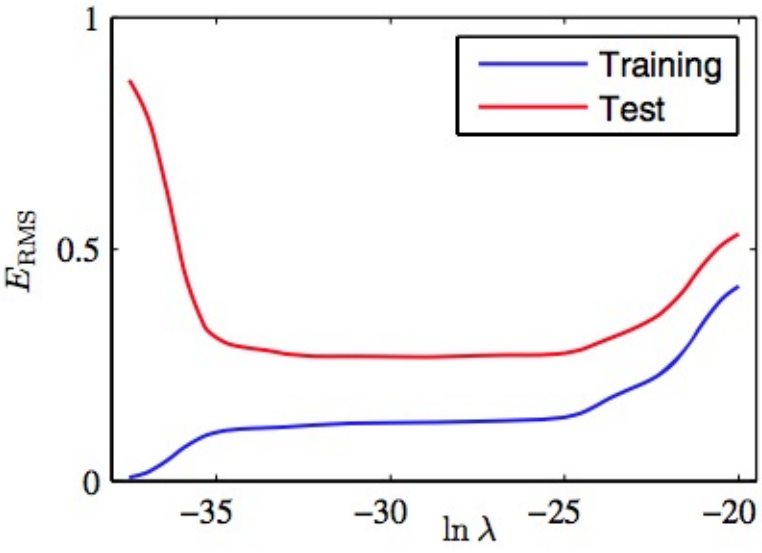

图1.8: 均方根误差(1.3)与$ln\lambda$的关系。

模型复杂度是一个重要的话题,将在1.3节详细讨论。这里我们简单地说一下,如果我们试着用最小化误差函数的方法解决一个实际的应用问题,我们就需要有一种能确定合适的模型复杂度的方法。上面的结果给出了一种达成这一目标的简单方式，即通过把现有的数据分割成一个用来确定系数w的训练集和分离出来的一个用来最优化模型的复杂度(m或λ)的验证集（也被称为拿出集（hold-out set））。但是在许多情况下,这太浪费有价值的训练数据了,我们不得不寻找更高级的方法。

目前我们关于多项式拟合的讨论大量地依赖于直觉。现在我们需要使用概率论的方法，来更加原则化的解决模式识别中的问题。概率论不仅提供了本书后续几乎所有章节的基础,也能让我们更深刻地理解本章中多项式拟合的问题以及引出的重要概念,使得我们可以把这些概念扩展到更复杂的情况。

模式识别的一个核心概念是不确定性。这是由测量时的噪声以及有限的数据集造成的。概率论提供了一个量化，控制这样的不确定性的一致性框架，是模式识别的核心基础。当于1.5节中讨论的决策论相结合时，我们可以从有限的信息中做出最优的决定，尽管这些信息是不完整的、有歧义的。

我们以一个简单的例子来介绍概率论的基本概念。想象一下，我们有两个盒子，一个红的一个蓝的，红的盒子中有2个苹果6个橘子，蓝色盒子中有3个苹果1个橘子，就像图1.9中展示的那样。

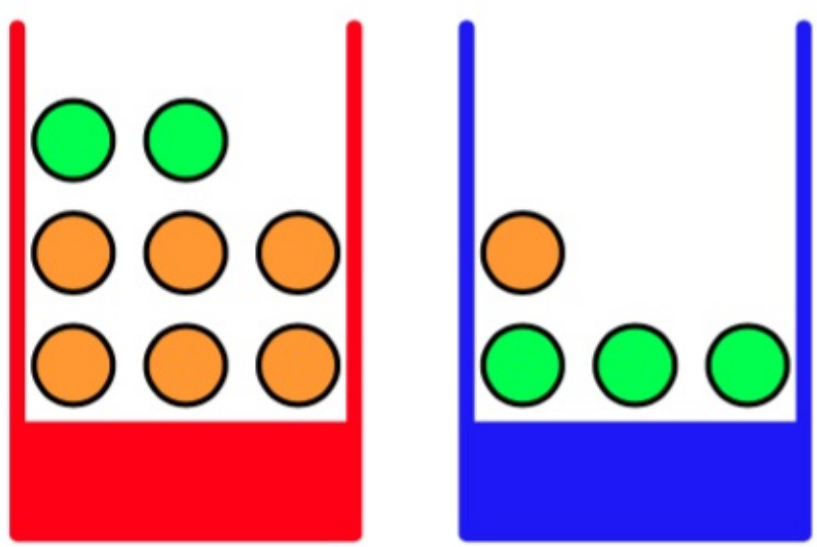

图 1.9 两个不同颜色的盒子,每个盒子中都有水 果,苹果用绿色表示,橘子用橙色表示。

假设我们随机的挑选一个盒子，并从中随机的挑选一个水果，观察一下我们选择的水果种类，然后把它放回到原来的盒子中。假设重复这个过程很多次。假设我们在40%的情况中选择了红盒子，在60%的情况中选择了蓝盒子。并且我们选择盒子中的水果时是等可能的。

在这个例子中,我要选择的盒子的颜色是一个随机变量,记作B。这个随机变量可以取两个值中的一个,即r(对应红盒子)或b(对应蓝盒子)。同样的水果的选择也是一个随机变量，记作F。它可以取a(苹果)或者o(橘子)。

开始阶段,我们把一个事件的概率定义为在总试验次数趋于无穷的情况下事件发生的次数与试验总数的比值。因此选择红盒子的概率为$\frac{4}{10}$，选择蓝色盒子的概率是$\frac{6}{10}$。我们把这个概率记作$p(B=r)=\frac{4}{10}, p(B=b)=\frac{6}{10}$。注意，根据定义概率一定位于区间[0, 1]内。另外，包含所有可能的结果，并且它们之间是互斥的（如在前面的例子中我们必须选择红色或蓝色盒子），那么他们的概率的和就等于1。

现在我们可以问这样的问题:选择到苹果的整体概率是多少?或者,假设我们选择了橘子, 那么选择的盒子是蓝色的概率是多少？一旦我们通过掌握概率论的两个基本规则：加法规则（sum rule）和乘积规则（product rule），来回答这样的问题，事实上我们也就可以回答与模式识 别相关的比这些复杂得多的问题。掌握这些规则之后,我们将重新回到我们的水果盒子的例子。

为了推导概率的规则，考虑图1.10所示的更一般的情形。如图1.10所示，它涉及到两个随机变量X, Y（例如可以是上面例子中“盒子”和“水果”的随机变量）。

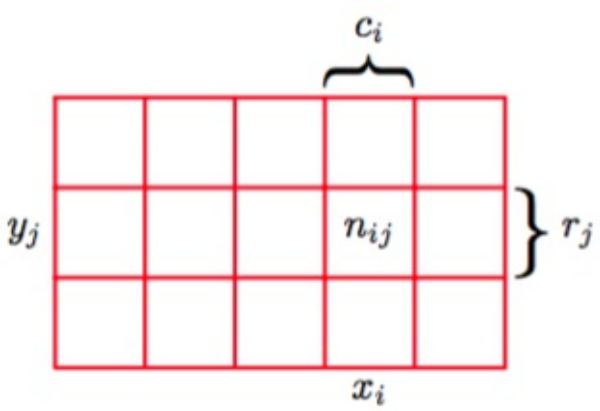

图 1.10:两个随机变量

假设X可以任意的取x_i，其中$i = 1, \ldots, m$。设Y可以任意的取y_j，其中$j = 1, \ldots, l$。现在我们做N次同时对X和Y进行取样的实验，把结果为$X = x_i, Y = y_j$的试验的数量记作n_{ij}。并且,把X的结果是x_i(与Y的取值无关)的试验的数量记作c_i,类似的,把Y的结果是x_i(与Y的取值无关)的试验的数量记作r_j。

X取x_i且Y取y_j的概率被记作$p(X = x_i, Y = y_j)$,被称为$X = x_i$和$Y = y_j$的联合概率(joint probability)。它的计算方法是单元格i, j中的数量与总数的比值，即：

$$p(X = x_i, Y = y_i) = \frac{n_{ij}}{N} \tag{1.5}$$

这里有一个隐式的规则：$N \to \infty$。类似地,X取x_i（与Y取值无关）的概率被记作$p(X = x_i)$,计算方法是列i的数量与总数的比值,即:

$$p(X = x_i) = \frac{c_i}{N} \tag{1.6}$$

由于在图1.10中列i的数量是列中每一个单元格的数量的和（$c_i = \sum_j n_{ij}$），所以根据公式(1.5)和(1.6)我们就得到：

$$p(X = x_i) = \sum_{j=1}^{l} p(X = x_i, Y = y_j) \tag{1.7}$$

这就是加法规则。注意,$p(X = xi)$有时被称为边缘概率(marginal probability),因为它是通过边缘化或加和其他变量来得到的（本例中为Y）。

如果我们只考虑那些$X = x_i$的情况，那么其中$Y = y_j$所占的比例被记作$p(Y = y_j|X = x_i)$，被称为给定$X = x_i$的$Y = y_j$的条件概率(conditional probability)。它是由单元格i, j中的数量与列i的总数的比值，即：

$$p(Y = y_j|X = x_i) = \frac{n_{ij}}{c_i} \tag{1.8}$$

由公式(1.5),(1.6)和(1.8)我们可以得到：

$$p(X = x_i, Y = y_j) = \frac{n_{ij}}{N} = \frac{n_{ij}}{ci} \cdot \frac{c_i}{N} = p(Y = y_j|X = x_i)p(X = x_i) \tag{1.9}$$

这就是概率的乘法规则。

现在我们可以很清楚的区分随机变量（水果例子中的盒子变量B）和随机变量的取值（盒子是红色时取值r）。当B取值为r时的概率我们记作$p(B = r)$。虽然这种记法消除了歧义，但是它相当笨拙，在很多情况下也没有必要。所以，在上下文没有奇异的情况下，我们简单地用$p(B)$表示随机变量B的分布，用$p(r)$表示对于特定的值r的分布估计。

使用这种简洁的记法,可以用下面的形式表示概率论的两条基本规则:

加法规则

$$p(X) = \sum_{Y} p(X, Y) \tag{1.10}$$

乘法规则

$$p(X, Y) = p(Y|X)p(X) \tag{1.11}$$

这里$p(X, Y)$是联合概率,可以表述为“X且Y的概率”。同样的，$p(Y|X)$是条件概率,可以表 述为“给定X的条件下Y的概率”，$p(X)$是边缘概率,可以简单地表述为“X的概率”。这两个简单 的规则是我们全书中使用的全部概率推导的基础。

根据乘法规则以及对称性$p(X, Y) = p(Y, X)$我们就可以得到两个条件概率之间的关系：

$$p(Y|X) = \frac{p(X|Y)p(Y)}{p(X)} \tag{1.12}$$

这就是贝叶斯定理,在模式识别和机器学习领域扮演着核心角色。使用加法规则，贝叶斯定理中的分母可以用分子中的项表示为：

$$p(X) = \sum_{Y} p(X|Y)p(Y) \tag{1.13}$$

我们可以把贝叶斯定理中的分母看作为了保证左边的条件概率对于所有的Y取值的和为1。

在图1.11中我们展示了两个变量的联合分布的简单例子，用来说明边缘和条件概率的概念。

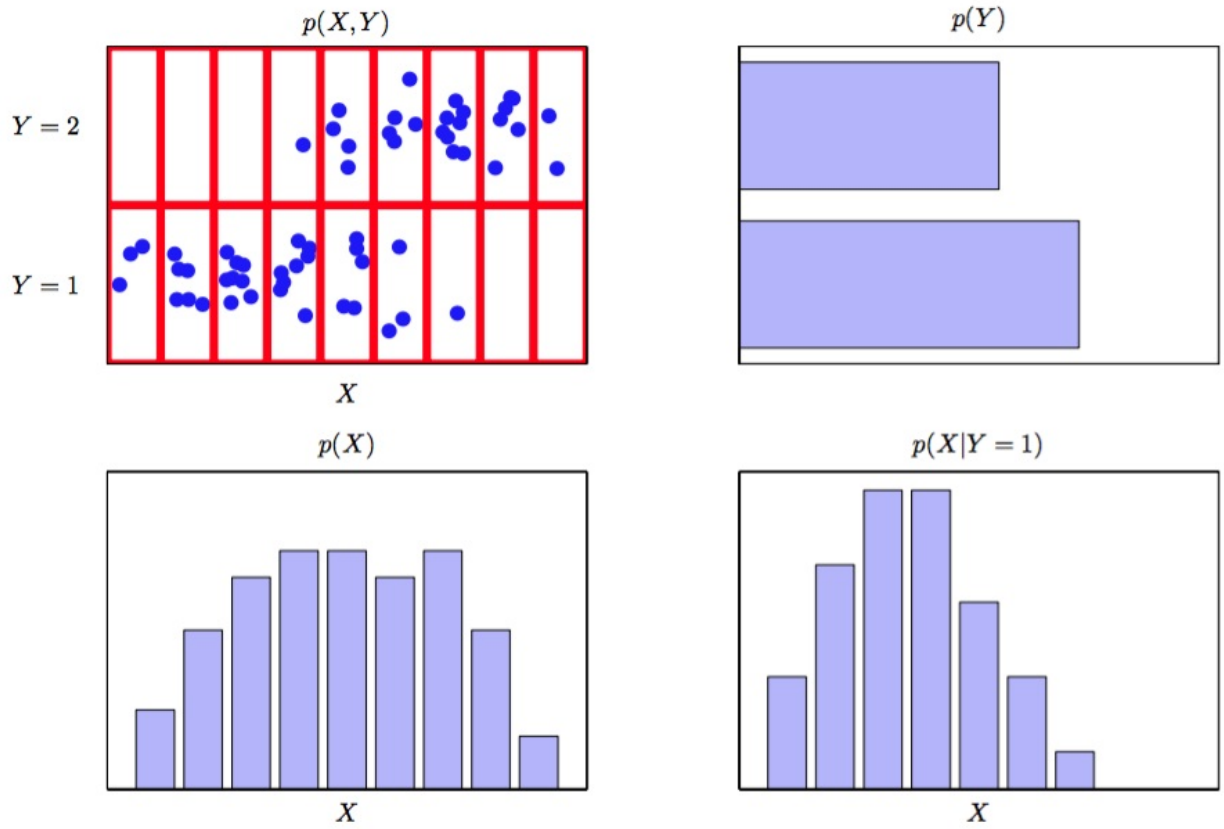

图 1.11: 两个变量X和Y 上的概率分布的一个例子。

这里只有 $N = 60$ 的有限个样本，它们的联合分布展示在左上。右上方是 Y 取两种值的比例的直方图。根据概率的定义，这些比例在 $N \to \infty$ 时会等于相应的概率 $p(Y)$。我们可以吧直方图看作一种在给定有限的数据点的情形下，对概率分布建模的简单的方式。使用数据对概率分布建模是统计模式识别的核心，在本书中将会详细介绍。图1.11中剩下的两张图分别给出了 $p(X)$ 和 $p(X|Y = 1)$ 的估计的直方图。

现在，让我们回到那个水果的例子中。再一次强调随机变量和它的实例之间的区别。选择红盒子或者蓝盒子的概率分别由下式给出:

$$p(B = r) = \frac{4}{10} \tag{1.14}$$

$$p(B = b) = \frac{6}{10} \tag{1.15}$$

注意：$p(B = r) + p(B = b) = 1$。

现在，假设我们随机的挑选一个盒子，并选到了蓝色。那么选到苹果的概率就是蓝盒子中苹果的比例：$\frac{3}{4}$，所以 $p(F = a|B = b) = \frac{3}{4}$。实际上,我们可以写出给定盒子的条件下水果种类的全部四个概率：

$$p(F = a|B = r) = \frac{1}{4} \tag{1.16}$$

$$p(F = o|B = r) = \frac{3}{4} \tag{1.17}$$

$$p(F = a|B = b) = \frac{3}{4} \tag{1.18}$$

$$p(F = o|B = b) = \frac{1}{4} \tag{1.19}$$

注意：这些概率是标准化的，所以：

$$p(F = a|B = r) + p(F = o|B = r) = 1 \tag{1.20}$$
$$p(F = a|B = b) + p(F = o|B = b) = 1 \tag{1.21}$$

现在我们可以用加法和乘法规则来计算选到苹果的总的概率。

$$\begin{aligned} p(F = a) &= p(F = a|B = r)p(B = r) + p(F = a|B = b)p(B = b) \\ &= \tfrac{1}{4} \times \tfrac{4}{10} + \tfrac{3}{4} \times \tfrac{6}{10} = \tfrac{11}{20} \end{aligned} \tag{1.22}$$

然后我们根据加法规则就可以得到：$p(F = o) = 1 - \frac{11}{20} = \frac{9}{20}$。

反过来，假设我们知道选到是说过是橘子，我们想知道它是从那个盒子中来的。这需要我们在 给定水果种类的条件下估计盒子的概率分布,而公式(1.16)至(1.19)给出的是在已知盒子颜色的情形下水果的概率分布。我们可以通过贝叶斯定理来解决这样的反转的条件概率：

$$p(B = r|F = o) = \frac{p(F = o|B = r)p(B = r)}{p(F = o)} = \frac{3}{4} \times \frac{4}{10} \times \frac{20}{9} = \frac{2}{3} \tag{1.23}$$

根据加法规则我们得到：$p(B = b|F = o) = 1 - \frac{2}{3} = \frac{1}{3}$。

我们可以按下面的方式来解释贝叶斯定理。如果我们在不知道水果种类的情况下，被问及会选择哪个盒子，这是我们得到的最完整的信息只有概率$p(B)$。因为这个概率是在我们知道水果种类之前就能得到的，所以我们把这叫做先验概率（prior probability）。一旦我们被告知我们选择的水果是橘子的时候，就可以使用贝叶斯定理来计算概率$p(B|F)$。由于这个概率是我们在观察到F之后获得的，所以我们把这叫做后验概率（posterior probability）。注意，在这个例子中，选择红盒子的先验概 率是$\frac{4}{10}$，所以我们更可能选择蓝盒子。但是，一旦我们观测到选择的是橘子的时候，我们得到选择红盒子的后验概率是$\frac{2}{3}$，更可能选择的时候红盒子。这个结果与我们的直觉相符，因为红盒子中橘子的比例比蓝盒子中的高得多。因此选择了橘子这个事实提供了利于选择红盒子的有效证据。事实上,这个超过了先验的假设的证据相当有效,使 得红盒子被选择的可能性大于蓝盒子。

最后，我们强调一下，如果两个变量的联合分布可以分解成它们的边缘分布的乘积，也就是$p(X, Y) = p(X)p(Y)$，那么变量X, Y是相互独立的。根据乘法规则，我们得到$p(Y|X) = p(Y)$，也就是对于对定的X的条件下Y的条件分布是独立于X的。举个例子，在水果的例子中，如果每个盒子包含同样比例的苹果和橘子，那么$p(F|B) = P(F)$，从而选择哪个苹果与选择哪个盒子无关。

除了考虑离散事件的概率外，我们还希望考虑连续变量的概率。我们会把的讨论限制在一个相对非正式的形式上。如果一个实值变量x落在区间$(x, x + \delta x)$的概率由$p(x)\delta x$给出，其中$\delta x \to 0$，那么我们就把$p(x)$称作x的概率密度（probability density）。图1.12阐释了这个概念。x位于区间(a, b)的概率由下式给出:

$$p(x \in (a, b)) = \int_a^b p(x)dx \tag{1.24}$$

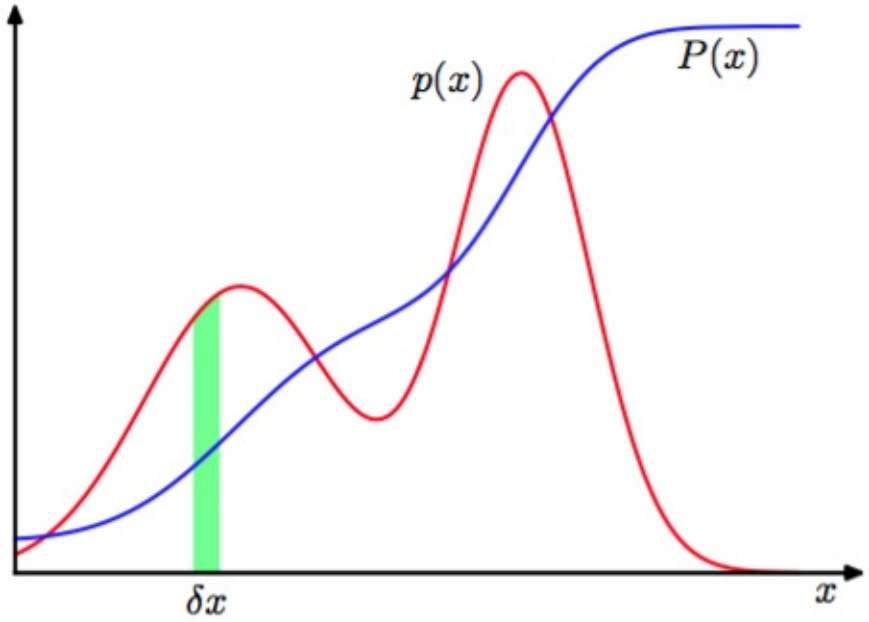

图 1.12: 连续变量的概率密度函数

因为概率是非负的，并且x的值必须在实轴上，所以概率密度$p(x)$必须满足这两个条件：

$$p(x) \leq 0 \tag{1.25}$$

$$\int_{-\infty}^{\infty} p(x)dx = 1 \tag{1.26}$$

在变量的非线性变化下，概率密度由一个简单的函数通过Jacobian因子变换得到。例如：一个变量$x = g(y)$，那么函数$f(x)$就变成$\tilde{f}(y) = f(g(y))$。现在，考虑概率密度$p_x(x)$，与它对应的关于新变量y的密度$p_y(y)$，其中不同的下标表示$p_x(x), p_y(y)$是不同的两个密度函数。观测区间$(x, x + \delta x)$变换为区间$(y, y + \delta y)$，当δx很小时，我们有$p_x(x)\delta x \simeq p_y(y)\delta y$即：

$$p_y(y) = p_x(x)|\frac{dx}{dy}| = p_x(g(y))|g'(y)| \tag{1.27}$$

这个性质的一个结果就是：概率密度的最大值取决于变量的选择。

x位于区间$(-\infty, z)$的概率是由累计分布函数（cumulative distribution function）给出的：

$$P(z) = \int_{-infty}^{z} p(x)dx \tag{1.28}$$

它就像图1.12那样满足$P'(x) = p(x)$。

果我们有几个连续变量$x_1, \ldots, x_D$，一起被记作向量x，那么我们就定义：联合概率密 度

$p(x) = p(x_1, \ldots, x_D)$是使得落在包含点$x$的无穷小体积$\delta x$的点的概率等于$p(x)\delta x$。多变量概率密度必须满足

$$p(x) \leq 0 \tag{1.29}$$

$$\int p(x)dx = 1 \tag{1.30}$$

其中积分必须包含整个x空间。这也适用于离散变量和连续变量相结合的联合概率分布。

注意：如果x是离散变量，那么$p(x)$就叫做概率质量函数（probability mass function），因为它可以被看做在合法的x值上的“概率质量”的集合。

概率的加法，乘法规则以及贝叶斯定理，都适用于概率密度或离散变量与连续变量相结合的情形下。例如：x, y是两个实值变量，它们的加法，乘法规则可以表示为如下形式：

$$p(x) = \int p(x, y)dy \tag{1.31}$$

$$p(x, y) = p(y|x)p(x) \tag{1.32}$$

形式化地证明连续变量的加法，乘法规则（Feller, 1966）需要一个被叫做测度论（measure theory ）的数学分支，这超出的本书的范围。不过，它的正确性在直觉下是显然的。我们把实值变量分割为宽度为Δ，然后考虑这些离散的区间上的概率分布。当$\Delta \to 0$时，把求和转换为积分就得到希望的结果了。

概率中的一个重要操作是找到加权平均值。概率分布$p(x)$的函数$f(x)$的平均值被称为$f(x)$的期望，记作

$$\mathbb{E}[f] = \sum_x p(x)f(x) \tag{1.33}$$

所以平均值是有不同的x的概率进行加权的。在连续变量的情形下,期望由对应的概率密度的积 分的形式表示:

$$\mathbb{E}[f] = \int p(x)f(x)dx \tag{1.34}$$

两种情形下,如果我们给定有限的N个点，这些点满足某个概率分布或概率密度函数, 那么期望可以通过求和的方式估计：

$$\mathbb{E}[f] \simeq \frac{1}{N}\sum_{n=1}^{N} f(x_n)$$

在第11章中讨论取样方法时，我们会广泛使用这个方法。当$N \to \infty$时，公式(1.35)的估计就精确了。
有时,我们会考虑多变量函数的期望。这种情形下,我们可以使用下标来表明根据哪个变量进行的平均,例如：

$$\mathbb{E}_x[f(x, y)] \tag{1.36}$$

表示函数$f(x, y)$关于x的分布的平均，注意$\mathbb{E}_x[f(x, y)]$是关于y的函数。

我们同样可以得到关于条件分布的条件期望（conditional expectation）:

$$\mathbb{E}[f|y] = \sum_x p(x|y)f(y) \tag{1.37}$$

连续变量的定义于此类似。

$f(x)$方差的定义如下：

$$var[f] = \mathbb{E}[(f(x) - \mathbb{E}[f(x)])^2] \tag{1.38}$$

它度量了$f(x)$与均值$\mathbb{E}[f(x)]$之间的变异性的程度。把平方展开，方差可以写成$f(x)$与$f(x)^2$的期望的形式：

$$var[f] = \mathbb{E}[f(x)^2] - \mathbb{E}[f(x)]^2 \tag{1.39}$$

特别的，变量x自身的方差可以表示为：

$$var[x] = \mathbb{E}[x^2] - \mathbb{E}[x]^2 \tag{1.40}$$

对于两个变量x, y，他们的协方差（corvariance）定义为：

$$\begin{aligned} cov[x, y] &= \mathbb{E}_{x,y}[\{x - \mathbb{E}[x]\}\{y - \mathbb{E}[y]\}] \\ &= \mathbb{E}_{x,y}[xy] - \mathbb{E}[x]\mathbb{E}[y] \end{aligned} \tag{1.41}$$

它表示x, y在多大程度上协同变化。如果x, y相互独立，那么它们之间的协方差为0。

如果x, y是两个随机变量的向量，那么他们的协方差是一个矩阵。

$$\begin{aligned} cov[x, y] &= \mathbb{E}_{x,y}[\{x - \mathbb{E}[x]\}\{y^T - \mathbb{E}[y^T]\}] \\ &= \mathbb{E}_{x,y}[xy^T] - \mathbb{E}[x]\mathbb{E}[y^T] \end{aligned} \tag{1.42}$$

如果我们考虑向量x各分量之间的协方差，可以稍微简化下我们的记法：$cov[x] \equiv cov[x, x]$

目前为止，我们通过重复随机事件的发生的频率来考察概率。我们把这叫做经典（classical）或频率论（frequentist）的概率解释。现在我们转向更加通用的贝叶斯（Bayesian）观点，它使频率提供了不确定性的量化描述。

考虑一个不确定事件，如：月球是否曾经自己的轨道上围绕太阳旋转，或者本世纪末北极冰盖是否会消失。这种事件不能像盒子中的水果一样重复做很多次来定义它们的概率。但是我们通常会想一些办法，如：极地冰块融化的速度。如果我们得到了新的证据，如：从人造卫星获得的新的诊断信息，我们可能会修正关于冰川融化速度的观点。我们对冰川融化速度的评估会影响我们采取的行动，例如在何种程度上的减少温室气体的排放。在这样的情况下，我们定量的描述不确定性，当有少量新的证据是对其进行精确的修正，从而修正接下来所要采取的最优行动或决策。这些都可以通 过优雅的通用的贝叶斯概率观点来实现。

如果我们想要尊重常识,来做出合理的推断，用概率来表达不确定性不是可选的，而是不可避免的。如：Cox (1946)证明的，如果用数值来表示置信的程度，那么用这种置信度的常识属性的公理集合推导出的一组用来来处理置信的程度的规则，等价于概率的加法规则和乘积规则。这首次粗略的证明了概率论可以被当作布尔逻辑的扩展来处理不确定性(Jaynes, 2003)。大量其他的学着也发表了不同的满足这样的不确定性度量的属性的性质集合或者公理集合（Ramsey, 1931; Good, 1950; Savage, 1961; deFinetti, 1970; Lindley, 1982）。在这些情况下，结果的数值完全符合概率的规则。因此把这些看成（贝叶斯观点）概率就很自然了。

在模式识别领域,对概率有一个更加通用的观点同样是很有帮助的。考虑1.1节中讨论的多项式曲线拟合，以频率论的观点去考察观察到的随机变量t_n似乎是合理的。然而我们想强调并量化模型参数w的不确定性。我们将会看到，从贝叶斯的观点来说，我们能够使用概率论的机制来描述模型参数w或模型选择的不确定性。

现在，贝叶斯定理有了新的意义。回忆那个盒子中的水果的例子，水果种类的确定，为选择红盒的概率提供了相关的信息。在这个例子中，贝叶斯定理通过观测到的数据提供的证据，把先验概率转化为了后验概率。和将要看到的细节一样，当我们进行数量的推断（如：多项式曲线拟合中的参数w），我们可以采用同样的方法。在观测数据之前，我们以先验概率$p(w)$的形式给出了，一些关于参数w的假设。观测到的数据$D = t_1, \dots, t_n$的影响，是通过条件概率$p(D|w)$来表达的，这个在1.2.5节中显示的表达出来。贝叶斯定理的公式：

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

我们可以根据观测到D后的后验概率$p(w|D)$来估计w的不确定性。

贝叶斯定理右侧的量$p(D|w)$由观测到的数据集D来估计，可以被看成参数向量w的似然函数（likelihood function）。不同的参数向量w的情况下，观测到的数据集的可能性。注意似然不表示它是w的概率分布，它关于w的积分也不（一定）等于1。

给似然函数这个定义之后，我们可以用自然语言来描述贝叶斯定理：

$$\text{posterior} \propto \text{likelihood} \times \text{prior} \tag{1.44}$$

其中所有的量都是关于w的函数。公式（1.43）中的分母是一个标准化的常数，用来保证左边的后验分布是一个合法的概率密度且积分为1。实际上，对公式（1.43）两边同时对w进行积分，我们可以用先验分布和似然函数来表示贝叶斯定理中的分母：

$$p(D) = \int p(D|w)P(w)dw \tag{1.45}$$

在贝叶斯和频率论观点中，似然函数$p(D|w)$都起着重要作用。然而，在这两种观点中它的使用方式有着本质的不同。在频率论的观点中，w被当作固定的参数，它的值是由某种形式的估计来确定的，这个估计误差是由可能的数据集D分布来确定的。与之相比，在贝叶斯观点下，只有一个数据集D(即实际观测到的数据集)，参数的不确定性是通过w的概率分布来表示的。

最大似然（maximum likelihood）是频率论广泛使用的一种估计，其中w取使似然函数$p(D|w)$达到最大值的值，也就是

使w的值等于使观察到的数据集出现的概率最大的值。在机器学习的文献中，似然函数的负对数别称为误差函数（error function）。因为负对数是一个单调递减的函数，最大化似然函数也就是最小化误差。

自助法（bootstrap）是频率论中一种决定误差的方法(Efron, 1979; Hastie et al., 2001)。这种方法中,使用下面的方式创造多个数据集：假设我们的原始数据集包含N个数据点$X = x_1, \ldots, x_N$。我们可以通过随机的从X中取N个数据来创建数据集X_B。选取是可以重复的，所以有些X中的点可能在X_B中出现多次，而有些可能不出现。这样的过程可以重复L次，得到L个大小为N的通过对原数据集X采样得到的数据集。参数估计的统计精确度就可以通过考察不同的自助数据集之间的预测变异性来进行评估。

贝叶斯观点的一个优点是：很自然的包含了先验知识。例如：假设，掷一枚普通的硬币3次,每次都是正面朝上。经典的最大似然估计硬币正面朝上的概率时，结果会是1，表示将来所有的投掷都会是正面朝上，与之相对的，带有任意合理的先验条件的贝叶斯方法都不会得出这么极端的结论。

关于频率论和贝叶斯的观点之间的优缺点已经有很多争论，事实上，并没有纯粹的频率论或贝叶斯观点。举个例子，针对贝叶斯方法的一种广泛的批评就是先验概率的选择通常是为了计算的方便而不是为了反映出任何先验的知识。一些观点甚至认为，贝叶斯观点中的结论对先验选择的依赖性是困难的来源。减少对于先验的依赖性是所谓无信息（noninformative）先验的一个研究动机。但是这导致了对比不同模型间的困难，并且当模型选择不好的时候极有可能导致不好的结果。频率论的估计方法在一定的程度上避免了这一问题，并且交叉验证的技术在模型比较等方面也很有用。

过去几年贝叶斯方法在实际应用中的重要性的逐渐增长，所以本书重点强调贝叶斯观点，在必要的时候讨论一些有用频率学概念。虽然贝叶斯的框架起源于18世纪，但它的实际应用在很长时间内都被执行完整的贝叶斯步骤的困难性所限制，尤其在做预测或比较不同的模型时，需要边缘化（求和或积分）整个参数空间。取样方法的发展,如马尔可夫链（Markov chain）蒙特卡罗（Monte Carlo）（将在11章中讨论）以及计算机的速度和内存容量的极大提高，打开了在一系列令人映像深刻的问题领域中实际使用贝叶斯方法的大门。其中蒙特卡罗方法非常灵活，可以应用于许多种模型。然而，它在计算上的复使得它主要应用于小规模问题。

最近，许多高效的判别式方法被提出来，例如变种贝叶斯（variational Bayes），期望传播（expectation propagation ）（在第10章中讨论）。它们提供了抽样方法的一种补充替代方法，使得贝叶斯技术能应用在大规模的应用中（Blei et al., 2003）。

我们将会用整个第二章来学习不同的概率分布及他们的核心属性。但是，我们先介绍在连续变量中最重要的一个分布：正态或高斯分布（Normal 或 Gaussian）分布。我们在本章接下来的部分和本书剩余的部分中大量的使用这个分布。

对于一元实值变量x,高斯分布被定义为：

$$\mathcal{N}(x|\mu,\sigma) = \frac{1}{(2\pi\sigma^2)^{\frac{1}{2}}} exp\{-\frac{1}{2\sigma^2}(x-\mu)^2\} \tag{1.46}$$

它是由均值（mean）μ和方差（variance）σ^2控制的。方差的平方根，也就是σ被称为标准差（standard deviation）。方差的导数写作：$\beta = 1/\sigma^2$被称为精度（precision）。我们很快就能看到使用这些项的动机。图1.13展示了高斯分布的图像：

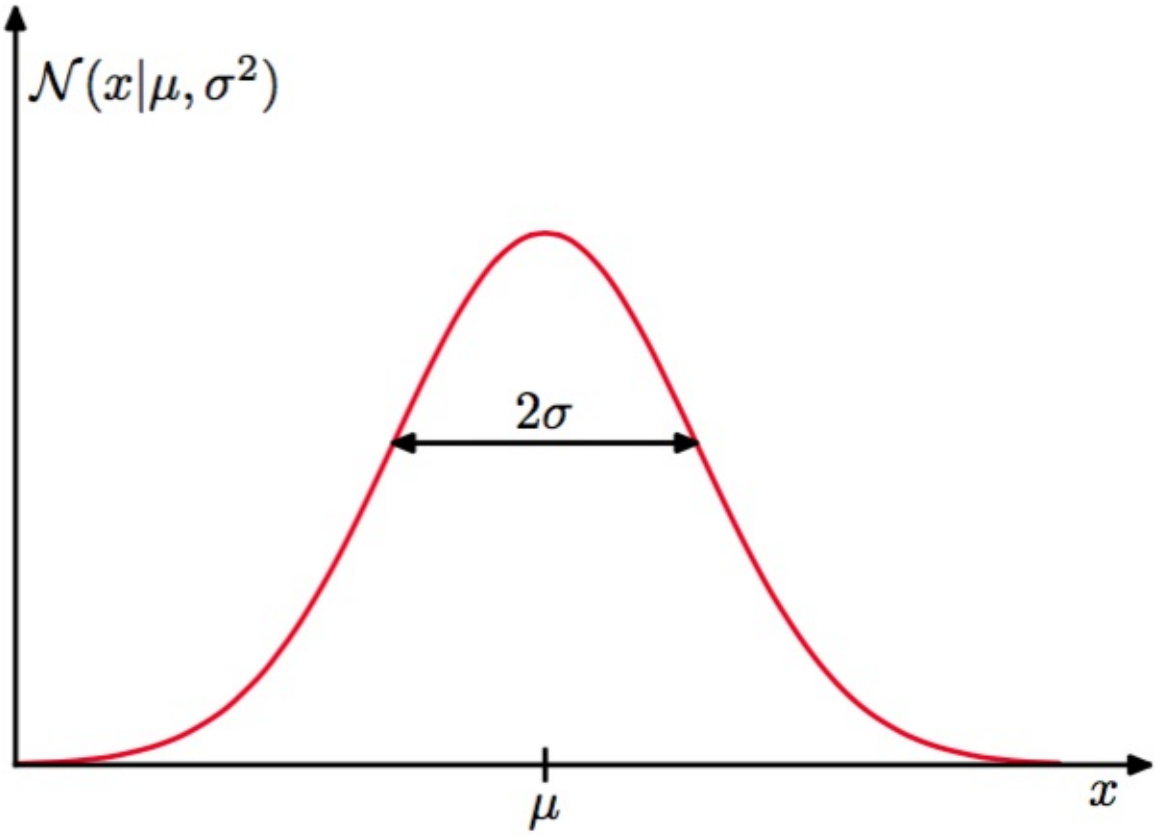

图 1.13: 高斯分布

从公式（1.46）可以得到高斯分布满足：

$$\mathcal{N}(x|\mu,\sigma^2) > 0 \tag{1.47}$$

再者，很容易就证明高斯是标准化的：

$$\int_{-\infty}^{\infty} \mathcal{N}(x|\mu,\sigma^2)dx = 1 \tag{1.48}$$

因此公式(1.46)满足有效的概率密度的两个条件。
我们已经能够找到关于x的函数在高斯分布下的期望，特别的x的均值：

$$\mathbb{E}[x] = \int_{-\infty}^{\infty} \mathcal{N}(x|\mu,\sigma^2)xdx = \mu \tag{1.49}$$

由于参数μ表示在分布下的x的平均值，它通常被叫做均值。类似地,二阶矩为：

$$\mathbb{E}[x^2] = \int_{-\infty}^{\infty} \mathcal{N}(x|\mu, \sigma^2)x^2 dx = \mu^2 + \sigma^2 \tag{1.50}$$

从公式（1.49）和（1.50），可以得出x的方差是：

$$var[x] = \mathbb{E}[x^2] - \mathbb{E}[x]^2 = \sigma^2 \tag{1.51}$$

因此，σ^2也被称为方差参数。分布中出现最多的被称为众数。在高斯分布中众数正好与均值重合。

我们也对D维连续变量的向量x的高斯分布也感兴趣。定义为：

$$\mathcal{N}(x|\mu, \Sigma) = \frac{1}{2\pi}^{D/2} \frac{1}{|\Sigma|^{1/2}} exp\{-\frac{1}{2}(x-\mu)^T \Sigma^{-1}(x-\mu)\} \tag{1.52}$$

其中D为向量μ被称为均值，$D \times D$矩阵Σ被称为协方差，$|\Sigma|$表示Σ的行列式。本章中我们很少用到多变量高斯分布，详细的性质将在2.3节讨论。

现在，假设我们有观测值$X = (x_1, \ldots, x_n)^T$的数据集，用来表示N个标量x的观测值。注意, 我们使用了一个大写的X的来和使用x标记的向量变量$(x_1, \ldots, x_D)^T$作区分。假定，独立地从均值μ和方差σ^2未知的高斯分布中获取观测数据集合，我们想从这个数据集中获取这些参数。独立地从相同的分布中抽取的数据点被称为独立同分布（independent and identically distributed），通常缩写成i.i.d。我们已经知道两个独立事件的联合概率可以由各个事件的边缘概率的乘积得到。由于我们的数据集X是独立同分布的，我们可以用μ, σ^2给出数据集的概率：

$$p(X|\mu, \sigma^2) = \prod_{n=1}^{N} N(x_n|\mu, \sigma^2) \tag{1.53}$$

当我们把它看成μ, σ^2的函数时，这就是高斯分布的似然函数。就像图1.14展示的那样。

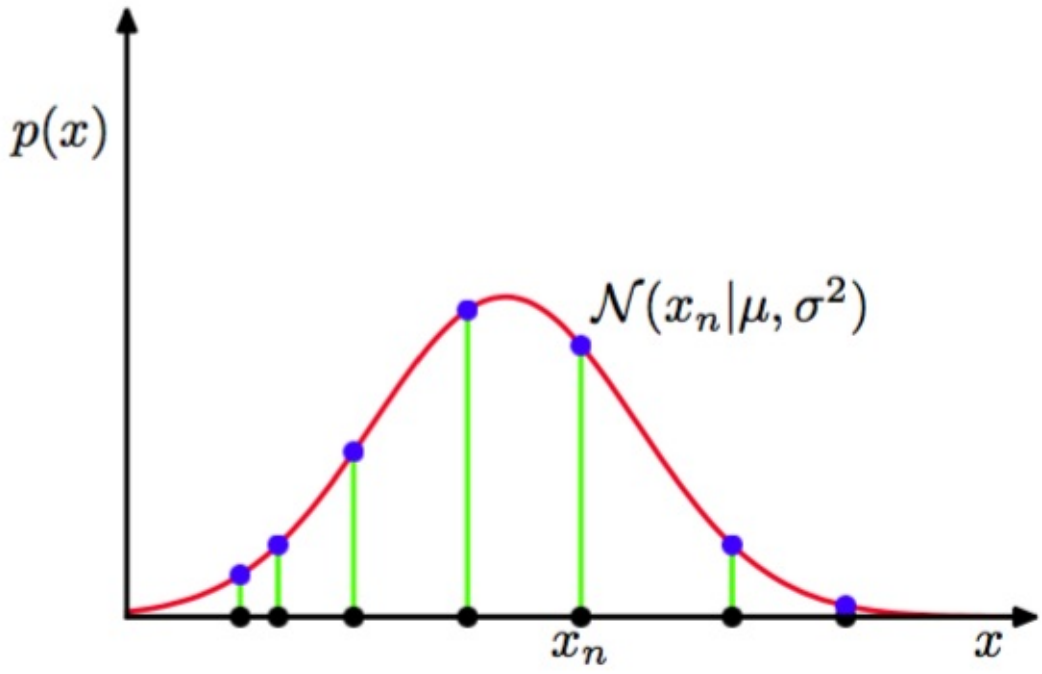

图 1.14: 高斯概率分布的似然函数，黑点表示数据集$\{x_n\}$的值，公式(1.53)给 出的似然函数对应于蓝色值的乘积。

使用一个观测数据集来决定概率分布的参数的一个通用的标准是寻找使似然函数取得最大值的参数值。这个观点看起来有点奇怪，因为在之前的概率论讨论中，似乎在给定数据集的情况下求最大化概率的参数，而不是给定参数的最大化数据ji出现的概率会更加自然。事实上,这两个标准是相关的。后面将使用曲线拟合的例子来说明这一点。

但是现在，我们需要通过最大化似然函数（1.53）来确定高斯分布中的μ, σ^2。实际上，最大化似然函数的对数会更加方便。因为对数是一个单调递增函数，最大化一个函数的对数等价于最大化这个函数。取对数后不仅简化了后续的数学分析，同样有助于数学计算。因为大量小概率的乘积很容易下溢，这可以通过去对数后的加法来解决。通过公式（1.46）和（1.53）似然函数可以写成：

$$\ln p(x|\mu, \sigma^2) = -\frac{1}{2\sigma^2}\sum_{n=1} N(x_n - \mu)^2 - \frac{N}{2}\ln \sigma^2 - \frac{N}{2}\ln(2\pi) \tag{1.54}$$

关于μ，最大化函数（1.54），我们可以得到最大似然解：

$$\mu_M L = \frac{1}{N}\sum_{n=1}^{N} x_n \tag{1.55}$$

这是样本均值（sample mean），即观测到样本的均值。类似地，关于σ^2的最大化函数（1.54）得到了方差的最大似然解：

$$\sigma^2_{ML} = \frac{1}{N}\sum_{n=1}^{N}(x_n - \mu_{ML})^2 \tag{1.56}$$

这是关于样本均值$\mu_M L$的样本方差，注意，我们要求关于μ, σ^2的联合最大化函数（1.54），但是在高斯分布中μ与σ^2是无关的，所以我们先求出（1.55）然后用这个结果来求（1.56）。

接下来，我们会强调最大化似然的一些限制，这里我们以使用最大化似然求解一元高斯分布的参数为例。实际情况下，最大似然方法会系统性的低估分布的方差。这一种被称为偏置（bias）的现象。它与多项式曲线拟合中的过拟问题有关。注意，最大似然的解：$\mu_M L, \sigma^2_{ML}$是关于数据集的值$x_1, \ldots, x_n$的函数。考虑这些量关于具有参数μ, σ^2的高斯分布的数据集的期望。很容易就能证明：

$$\mathbb{E}[\mu_{ML}] = \mu \tag{1.57}$$

$$\mathbb{E}[\sigma^2_{ML}] = (\frac{N-1}{N})\sigma^2 \tag{1.58}$$

所以一般来说，最大似然能对均值做出正确的估计，但是对方差低估了因子(N - 1) / N。背后的原因在图1.15中说明。

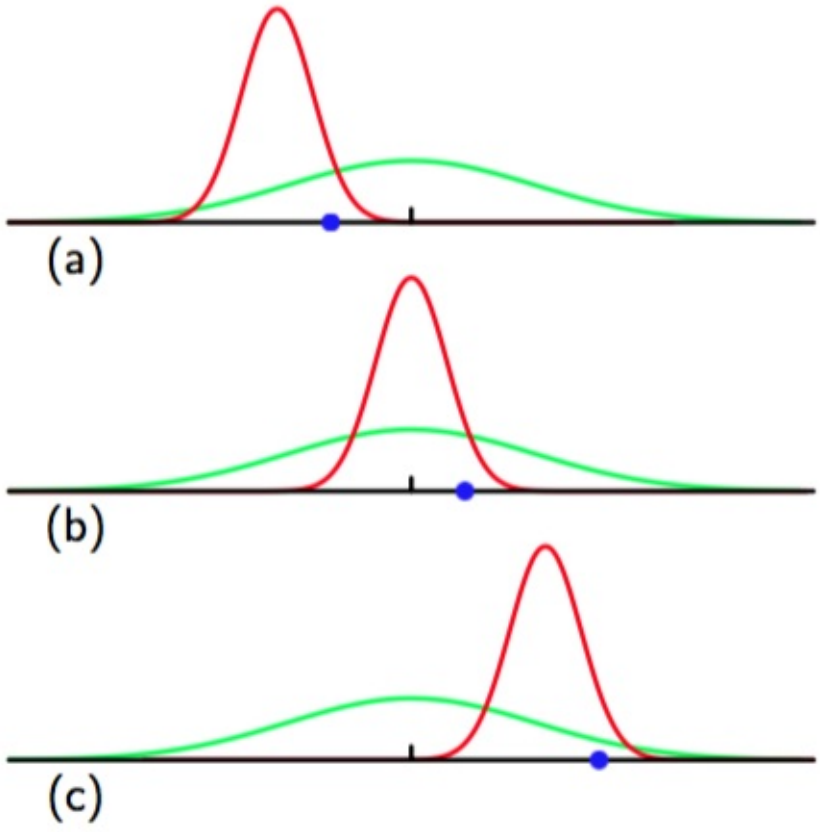

图 1.15: 最大似然方法确定高斯分布的方差时,偏移是如何产生的

根据公式（1.58）下面公式是无偏的：

$$\tilde{\sigma}^2 = \frac{N}{N-1}\sigma^2_{ML} = \frac{1}{N-1}\sum_{n=1}^{N}(x_n - \mu_{ML})^2 \tag{1.59}$$

在10.1.3节中，我们将会看到当我们采用贝叶斯方法是这个结果是如何自动出现的。

注意，当数据点的数量N增加时，最大似然的偏置变得越来越不重要，当$N \to \infty$时，方差的最大似然的解等于生成数据的分布的方差。实际上，N的值只要不是太小，偏置不会导致大问题。然而,在本书中，我们感兴趣的是带有很多参数的复杂模型，这时最大似然带来的偏置问题会严重的多。实际上,我们会看到,最大似然的偏置问题是我们在之前的多项式曲线拟合问题中遇到的过拟问题的核心。

我们已经知道怎么把多项式拟合问题表示为误差最小化问题。现在我们回到曲线拟合的例子，并以概率的角度来看待，以及让我们完全从贝叶斯的角度来看待这个问题，从而更深刻地认识误差函数和正则化。

曲线拟合问题的目标是能够根据N个输入$X = (x_1, \ldots, x_N)^T$组成的数据集和它们对应的目标值$T = (t_1, \ldots, t_N)^T$，在给出新的输入变量x的新值的情况下，预测目标变量t。我们可以用目标变量值的概率分布来表示我们的不确定性。为了这个目标，我们可以假设，对于给定的x的值，对应的目标变量t是具有与公式(1.1)给出的多项式曲线$y(x, w)$的值相等的均值的高斯分布，即：

$$p(t|x, w, \beta) = \mathcal{N}(t|y(x, w), \beta^{-1}) \tag{1.60}$$

其中，为了和后续章节记号的一致性，我们定义的分布的方差的导数为精度（precision）参数β。图1.16阐述了这种模式。

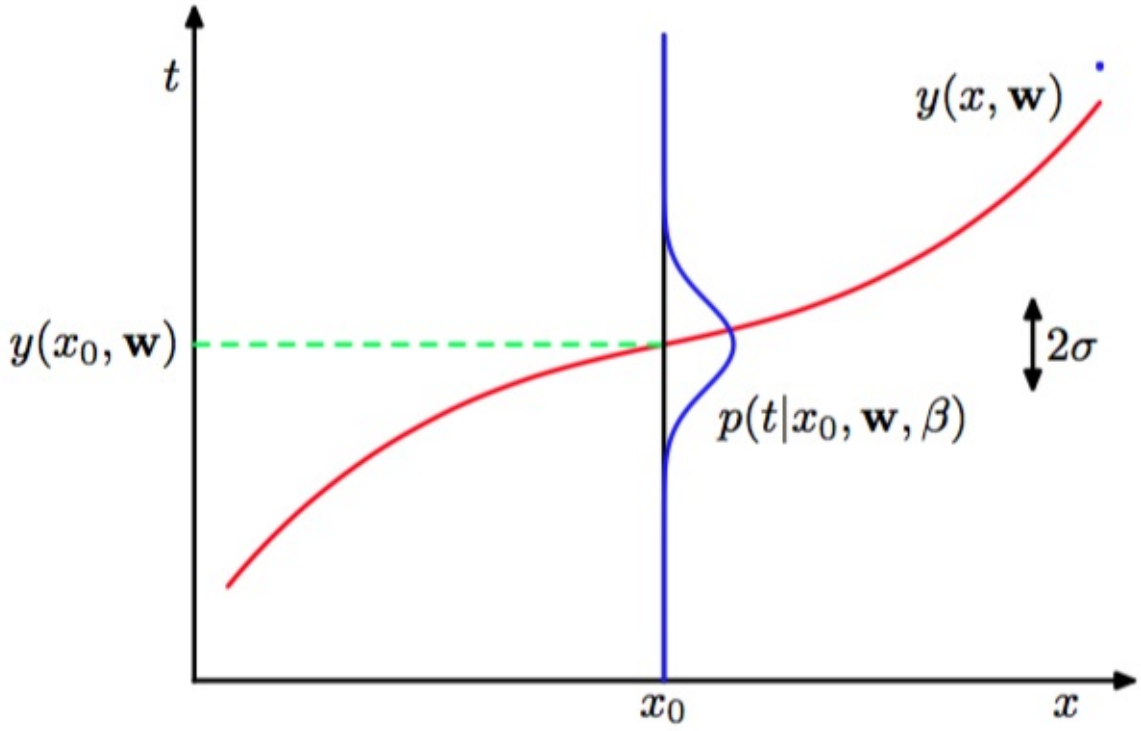

图 1.16: 目标值的高斯分布

现在，使用训练数据$\{X, T\}$，并通过最大似然来确定位置参数w, β。假定数据从（1.60）分布中独立的取出，那么似然函数就等于：

$$p(T|X, w, \beta) = \prod_{n=1}^{N} \mathcal{N}(t_n|y(x_n, w), \beta^{-1}) \tag{1.61}$$

与之前处理简单高斯分布时的做法一样，为了方便，把它转化为最大化似然函数的对数。代入（1.46）给出的高斯分布公式，可以得到似然函数的对数形式：

$$\ln p(T|X, w, \beta) = -\frac{\beta}{2} \sum_{n=1}^{N} \{y(x_n, w) - t_n\}^2 + \frac{N}{2} \ln \beta - \frac{N}{2} \ln (2\pi) \tag{1.62}$$

首先考虑确定多项式系数的最大似然解，记作w_{ML}。它们是由对（1.62）关于w的最大化来确定的。为了达到这个目的，可以先省略式（1.62）右手边的最后两项，因为它们与w无关。并且，使用一个正系数来缩放似然函数的对数并不会改变它关于w的最大值的位置，所以我们可以使用 1/2来代替$\beta/2$。最后，等价地去最小化似然函数的负对数，来替代最大化似然函数的对数。于是得到，对于确定w的最大化似然等价于（1.2）中给出的最小化平方和误差函数。所以，平方和误差函数是

采用高斯噪声的最大似然的自然结果。

同样，可以使用最大似然来确定高斯条件分布的精度参数β。关于β来最大化公式（1.62）得到：

$$\frac{1}{\beta_{ML}} = \frac{1}{N}\sum_{n=1}^{N}\{y(x_x, w_{ML}) - t_n\}^2 \tag{1.63}$$

再次提醒，和简单的高斯分布情况一样，首先确定控制均值的参数向量w_{ML}，然后使用这个结果来确定精度β_{ML}。

当确定好参数w, β后，就可以对新的值x做预测。由于现在有了概率模型，所以可以使用一种称为预测分布（predictive distribution）来表达t的概率分布，来代替一个简单的点估计。这是通过把最大似然参数代入式（1.60）得到的：

$$p(t|x, w_{ML}, \beta_{ML}) = \mathcal{N}(t|y(x, w_{ML}), \beta_{ML}^{-1}) \tag{1.64}$$

现在让我们朝着贝叶斯的方法前进一步，在多项式系数w上引入先验分布。简单起见，我们考虑高斯分布：

$$p(w|\alpha) = \mathcal{N}(w|0, \alpha^{-1}I) = (\frac{\alpha}{2\pi})^{(M+1)/2}exp-\frac{\alpha}{2}w^T w \tag{1.65}$$

其中α是分布的精度，$M+1$是M阶多项式的向量w中元素个数。像α这样的控制分布的模型参数被称为超参数（hyperparameters）。使用贝叶斯定理，w的后验分布，正比于先验分布和似然函数的乘积：

$$p(w|X, T, \alpha, \beta) \propto p(T|X, w, \beta)p(w|\alpha) \tag{1.66}$$

对于给定的数据集，可以通过找到最可能的w值来确定w，即最大化后验分布。这种技术叫做最大后验（maximum posterior）或简写为MAP。取公式(1.66)的负对数,结合公式(1.62)和 公式(1.65),我们可以看到,最大化后验概率就是最小化下式：

$$\frac{\beta}{2}\sum_{n=1}^{N}\{y(x_n, w) - t_n\}^2 + \frac{\alpha}{2}w^T w \tag{1.67}$$

因此，最大化后验概率等价于最小化正则化的平方和误差函数（之前在公式(1.4)中提到），正则化参数为$\lambda = \alpha/\beta$。

尽管包含了先验分布$p(w|\alpha)$，我们还是在对w做点估计，还没有以贝叶斯的观点来对待。在完全的贝叶斯的方法中，应该一致的使用概率的加法，乘法规则。我们稍后会看到，这需要对所有w值进行积分。这样的边缘化是使用贝叶斯方法的模式识别的核心。

在曲线拟合问题中，已经给定训练数据集X, T和新的测试数据x。我们的目标是预测t。因此，我们想估计预测分布$p(t|x, X, T)$。这里我们假设参数α, β是固定的，并且事先知道（在后续章节中我们将讨论如何使用贝叶斯方法从数据中推断出这样的参数）。

简单地说，贝叶斯方法就是一致的使用概率的加法，乘法规则。预测分布可以写成：

$$p(t|x, X, T) = \int p(t|x, w)p(w|X, T)dw \tag{1.68}$$

这里的$p(t|x, w)$是由公式（1.60）省略了参数α, β简化后得到，$p(w|X, T)$是参数的后验分布，可以由公式（1.66）的右手边标准化得到。在3.3节将看到, 对于曲线拟合这样的问题，这个后验分布是一个高斯分布，并可以得到解析解。同样的，公式（1.68）中的积分也可以解析的得到预测分布的高斯分布形式：

$$p(t|x, X, T) = \mathcal{N}(t|m(x), s^2(x)) \tag{1.69}$$

其中均值和方差是由下式给出的：

$$m(x) = \beta\Phi(x)^T S \sum_{n=1}^{N} \Phi(x_n)t_n \tag{1.70}$$

$$s^2(x) = \beta^{-1} + \Phi(x)^T S\Phi(x)t_n \tag{1.71}$$

其中矩阵S是由

$$S^{-1} = \alpha I + \beta \sum_{n=1}^{N} \Phi(x_n)\Phi(x)^T \tag{1.72}$$

其中I是单位矩阵，定义向量$\Phi(x)$为$\Phi_i(x) = x^i, i = 0, \ldots, M$。

我们看到公式（1.69）的预测分布的均值和方差是依赖x的。公式（1.71）的第一项表示由目标值上的噪声引起的预测值t的不确定性。这种不确定性在最大似然的预测分布（1.64）中由β_{ML}^{-1}表达的。然而，第二项是由使用贝叶斯方法导致的参数w的不确定性引起的。合成正弦回归问题的预测分布在图1.17展现。

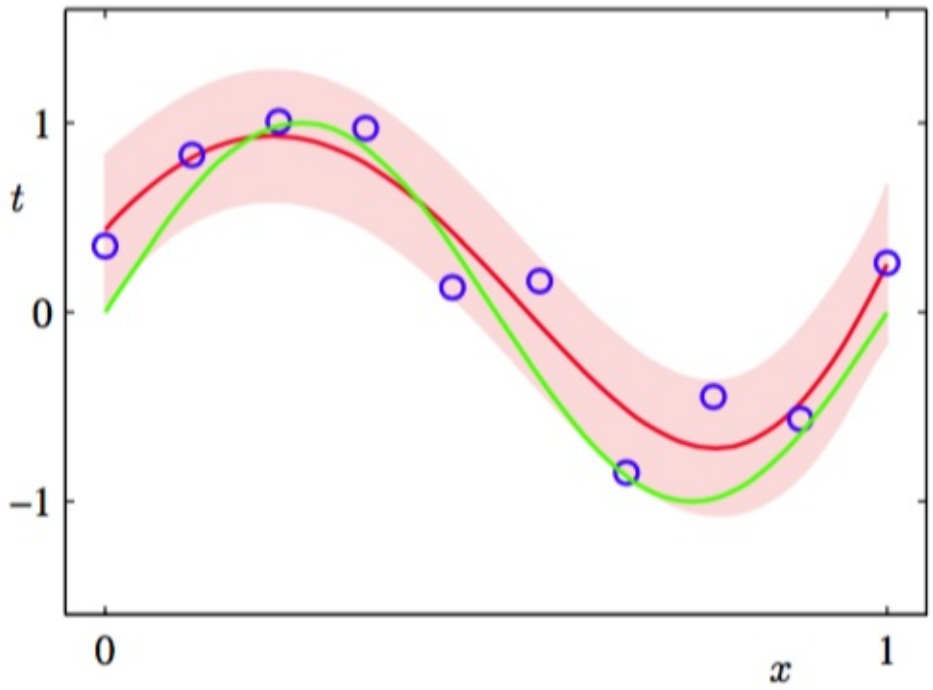

图 1.17: 用贝叶斯方法处理多项式曲线拟合问题得到的预测分布的结果。使用的多项式为$M = 9$，超参数被固定为$\alpha = 5 \times 10^{-3}, \beta = 11.1$(对应于已知的噪声方差)。其中，红色曲线表示预测概率分布的均值,红色区域对应于均值周围±1标准差的范围。

在我们使用最小二乘的多项式曲线拟合中，我们知道存在一个最优的阶数使它具有最好的泛化能力。多项式的阶数控制着模型的自由变量的数量，从而控制模型的复杂度。带有正则化的最小二乘，正则化系数λ同样控制着模型的有效复杂度。而对于更复杂的模型，如：混合分布和神经网络，它们有多个参数来控制复杂度。在实际应用中，我们需要确定这样的参数的值，这样做的主要目标通常是为了能更好的对新数据进行预测。此外，除了找到模型的合适的复杂度参数之外，可能还希望找到一个不同类型模型的范围，以便能够找到对于特定应用的最好的模型。

在最大似然方法中，我们已经看到，由于过拟问题，模型通过数据训练后，对未知的参数的预测表现的并不好。如果数据量丰富，一个简单的方法是使用其中的一部分可用数据，训练出一系列模型，或确定某个模型的一系列模型复杂度参数，然后使用独立数据，有时被称为验证集（validation set），比较各个模型的预测能力，选择最优的那个。如果模型设计使用有限的数据集迭代多次，那么对于验证数据会发生一定程度的过拟合，这时候就需要备用的第三个测试集（test set）来最总评估被选择模型的表现。

在很多应用中，能提供的训练和测试数据是有限的，为了更好的构建模型，我们希望尽可能的使用可用的数据来进行训练。然而，如果验证集比较小，它对预测表现的估计就会有一定的噪声。解决这个困境的一个方法是使用交叉验证（cross-validation），就像图1.18展示的那样。这种方法使用$(S-1)/S$的可用数据用来训练，同时使用所有的数据来评估表现。当数据相当稀疏时，使得$S=N$是比较合适的选择。其中N是数据集的量，这种技术就叫留一法。

图 1.18: 参数为 S 的交叉验证方法,这里说明了 S = 4 的情形。

交叉验证的一个主要的缺点是需要进行的训练的次数随着因子S增加，这对于训练本身很耗时的模型来说是个大问题。像这样使用分开的数据集来评估表现的交叉验证还有一个问题，对于那些有多个复杂度参数的模型（举个例子：有多个正则化参数的模型）在最坏的情况下，确定这些参数的组合所需的训练次数可能是参数个数的指数函数。显然我们需要一种更好的方法。理想情况下，这应该只依赖于训练数据，并且超参数的确定与模型类型的选择可以通过一次训练得出。因此，需要找到一种只依赖于训练数据的表现度量，并且不会受过拟所产生的偏置的影响。

历史上，各种各样的通过增加惩罚项来补偿复杂模型的过拟问题来尝试修正最大似然的偏置的“信息准则”被提出来。例如：，赤池信息准则（Akaike information criterion），或者简称为AIC（Akaike, 1974），选择使下面表达式最大的模型：

$$\ln p(D|w_{ML}) - M \tag{1.73}$$

这里的$p(D|w_{ML})$是最合适的对数似然函数，M是模型中的可调节参数。这个量的一个被叫做贝叶斯信息准则(Bayesian information criterion)或简称BIC的变体将在4.4.1节中介绍。这样的准则没有考虑到模型参数的不确定性，所以在实际应用中它们倾向于选择过于简单的模型。因此，在3.4节的完全的贝叶斯方法的讨论中，我们会看到，这种方法如果自然的，有原则的确定复杂度的惩罚项。

在多项式拟合的例子中，只有一个输入变量x。在模式识别的实际运用中，我们不得不处理包含许多输入变量组成的高维空间。正如我们现在讨论的那样,这个问题是个很大的挑战,也是影响模式识别技术设计的重要因素。