Chapter 1 Introduction

MingY

2017 年 12 月 10 日

==

- 一些概念
  - 有监督学习——训练数据的样本包含输入向量以及对应的目标向量。
    - 分类：输出是给每个输入向量分配到有限数量离散标签中的一个。
    - 回归：输出是由一个或多个连续变量组成。
  - 无监督学习——训练数据由一组输入向量 x 组成，没有任何对应的目标值。
    - 聚类：目标是发现数据中相似样本的分组。
    - 密度估计：目标是决定输入空间中数据的分布。
    - 数据可视化：把数据从高维空间投影到二维或三维空间。
  - 反馈学习——在给定条件下，找到合适的动作，使得奖励达到最大值。

==

1.1 例子：多项式曲线拟合

- 训练集
- 误差函数：每个数据点与函数 $y(x,\boldsymbol{w})$ 之间位移（绿色垂直线）的平方和（的一半）

$$y(x,\boldsymbol{w}) = w_0 + w_1 x + w_2 x^2 + \cdots + w_M x^M = \sum_{j=0}^{M} w_j x^j \tag{1.1}$$

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

- 通过选择使得 $E(\boldsymbol{w})$ 尽量小的 $\boldsymbol{w}$ 来解决曲线拟合问题，最终的多项式函数由 $y(x,\boldsymbol{w}^*)$ 给出。选择多项式的阶数 M 也是一个问题，图 1 给出了 4 个拟合多项式的结果，多项式的阶数分别为 M = 0,1,3,9。可以看出，常数（M = 0）和一阶（M = 1）多项式对于数据的拟合效果相当差，三阶（M = 3）多项式似乎给出了最好的拟合。而对于更高阶的多项式（M = 9），得到了一个对训练数据的完美的拟合，多项式函数精确地通过了每一个数据点，但是拟合的曲线剧烈震荡，称之为过拟合。
- 由于目标是通过对新数据的预测实现良好的泛化性，我们可以定量考察模型的泛化性与 M 的关系。方式为：考虑一个额外的测试集，这个测试集中 100 个数据点的生成方式与训练集完全相同，但是在目标值中包含的随机噪声的值不同。对于每个 M 的选择，我们之后可以用公式（1.2）计算训练集的 $E(\boldsymbol{w}^*)$，也可以计算测试集的 $E(\boldsymbol{w}^*)$。有时候使用根均方（RMS）误差更方便。这个误差由下式定义：

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

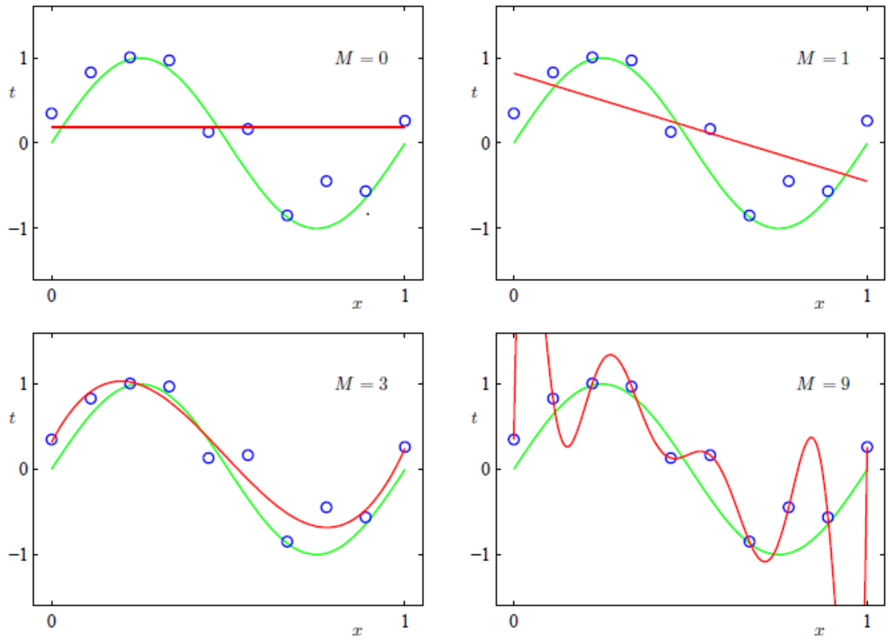

图 1: 不同阶数的多项式曲线，用红色曲线表示，拟合了给定的数据集

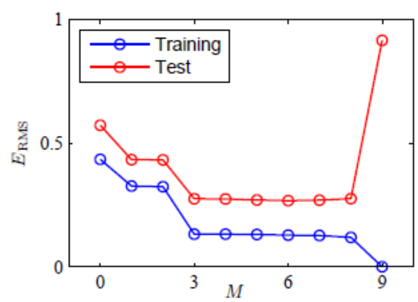

图 2: 对于不同的 M 值，训练数据和测试数据的 RMS 误差。

- 其中，除以 N 让我们能够以相同的基础对比不同大小的数据集，平方根确保了 E_{RMS} 与目标变量 t 使用相同的规模和单位进行度量。下图 2 展示了对于不同的 M 值，训练数据和测试数据的 RMS 误差。而且随着多项式阶数的增加，多项式系数 $\boldsymbol{w}^*$ 的值也是剧烈增大的。
- 观察给定模型的行为随数据集规模的变换情况。图 3 表明，当数据集的规模增加时，过拟合问题变得不那么严重。启发是，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（5 或 10）。但是，在 chapter3 将看到，参数的数量对于模型复杂度的大部分合理的度量来说都不是必要的。
- 于是，不得不根据可得到的训练集的规模限制参数的数量。经常用来控制过拟合现象的一种技术的正则化，这种方法给误差函数（1.2）增加了一个惩罚项，使得系数不会达到很大的值。修正后的误差函数如下：

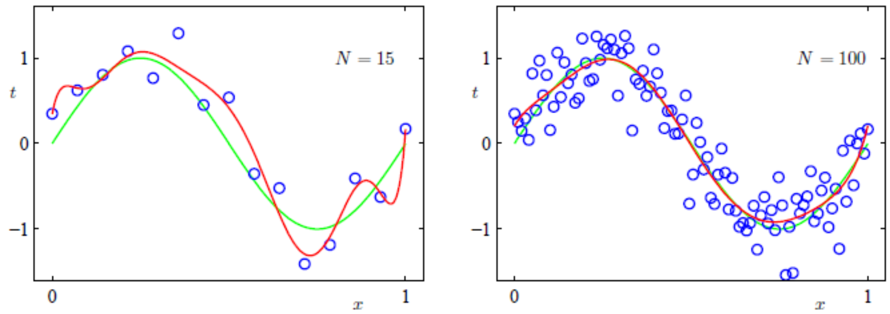

图 3: 使用 M=9 的多项式对 M=15 个数据点（左图）和 N=100 个数据点（右图）通过最小化平方和误差函数的方法得到的解。我们看到增大数据集的规模会减小过拟合问题。

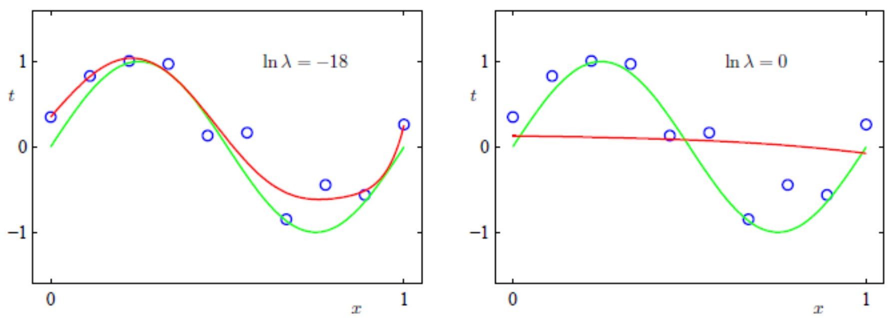

图 4: 使用正则化的误差函数（1.4），用 M=9 的多项式拟合图中的数据集。其中正则化参数 lambda 选择了两个值，分别对应于 ln(lambda)=-18 和 ln(lambda)=0。没有正则化项的情形，即 lambda=0，对应于 ln(lambda)= 负无穷，在图 1 的右下角给出。

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

其中，$\| \ \boldsymbol{w} \ \|^2 \equiv \boldsymbol{w}^T\boldsymbol{w} = w_1^2 + w_1^2 + \cdots + w_M^2$，系数 λ 控制了正则化项相对于平方和误差项的重要性。注意，通常系数 w_0^2 被省略。

- 图 4 展示了在 M = 9 的情况下使用与之前相同的数据拟合多项式的结果。这次使用的是公式（1.4）的正则化误差函数。
- 图 5 给出了正则化对于泛化错误的影响。可以看到，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。
- 模型复杂度是一个重要的话题，将在 1.3 节详细讨论。简单地说，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，那么我们不得不寻找一种方式来确定模型复杂度的合适值。上面的结果给出了一种完成这一目标的简单方式，即通过把给定的数据中的一部分从测试集中分离出，来确定系数 w。这个分离出来的验证集用来最优化模型的复杂度（M 或者 λ）。但是在许多情况下，太浪费有价值的训练数据，所以不得不寻找更高级的方法。

==

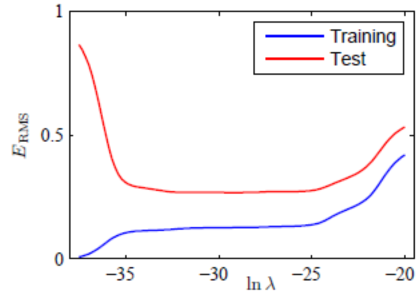

图 5: 对于 M=9 的多项式，均方根误差（1.3）与 ln(lambda) 的关系。

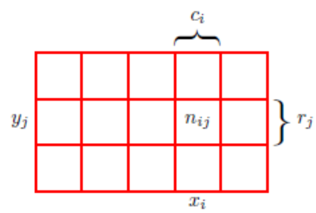

图 6: 我们可以这样推导概率的加和规则和乘积规则：考虑两个随机变量，X，取值为 {x_i}，其中 i = 1,...,M, 和 Y，取值为 {y_j}，其中 j =1,...,L。在这个例子中，我们取 M = 5 和 L = 3。如果我们考虑这些变量的总计 N 个实例，那么我们将 X = x_i 且 Y =y_j 的实例的数量记作 n_{ij}，它是对应的单元格中点的数量。列 i 中的点的数量，对应于 X = x_i，被记作 c_i，行 j 中的点的数量，对应于 Y = y_j，被记作 r_j。

1.2 概率论

- 联合概率：

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

- 一些推导：

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

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

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

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

- 两个基本规则：
  - 加和规则

$$p(X) = sum_Y p(X, Y)$$

  - 乘积规则

$$P(X, Y) = p(Y|X)p(X)$$

  - 这里 $p(X, Y)$ 是联合概率，可以表述为 “X 且 Y 的概率”。$p(X, Y)$ 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”。

- 概率密度
- 期望和协方差
- 贝叶斯概率
- 高斯分布
- 重新考察曲线拟合问题
- 贝叶斯曲线拟合

1.3 模型选择

1.4 维度灾难

1.5 决策论

- 最小化错误分类率
- 最小化期望损失
- 拒绝选项
- 推断和决策
- 回归问题的损失函数

1.6 信息论

- 相对熵和互信息