



机器学习：监督学习

- 参考教材： 吴飞，《人工智能导论：模型与算法》，高等教育出版社
- 在线课程(MOOC)： <https://www.icourse163.org/course/ZJU-1003377027>
- 在线实训平台（智海-Mo）： https://mo.zju.edu.cn/classroom/class/zju_ai_2022
- 系列科普读物《走进人工智能》 <https://www.ximalaya.com/album/56494803>

提纲

一、机器学习基本概念

二、回归分析

三、决策树

四、线性判别分析

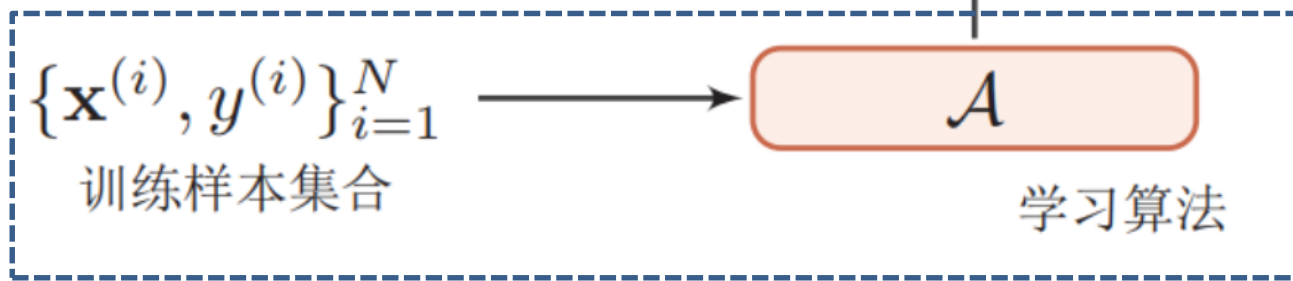
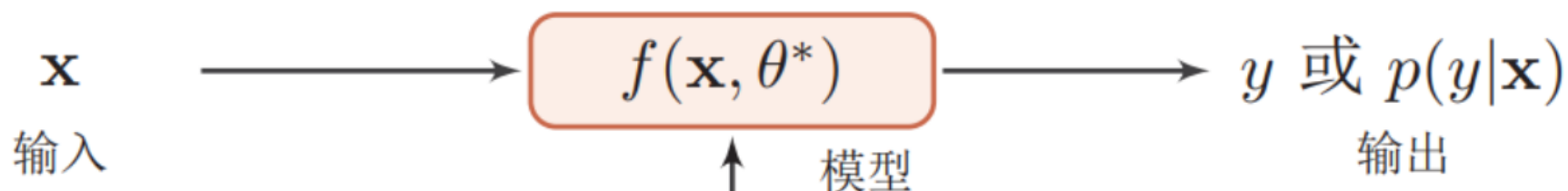
五、Ada Boosting

六、支持向量机

七、生成学习模型

机器学习三个基本要素

机器学习方法可以大致地分为三个基本要素：
模型、学习准则、优化算法。



机器学习：从数据中学习知识



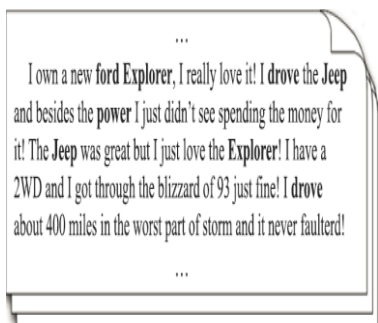
图像数据

$$f \left\{ \begin{array}{cccc} 81 & 116 & \dots & 133 \\ 104 & 130 & \dots & 159 \\ \vdots & \vdots & \ddots & \vdots \\ 155 & 189 & \dots & 218 \\ 197 & 221 & \dots & 216 \end{array} \right\}$$

- Person
- Dog
- ...

类别分类

- 从原始数据中提取特征
- 学习映射函数 f (又叫模型)



文本数据

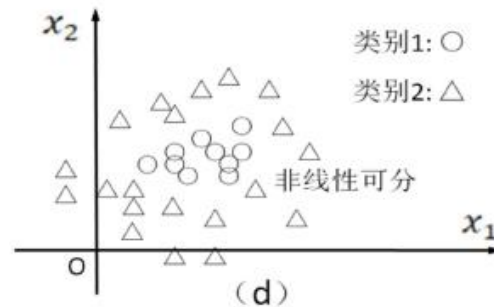
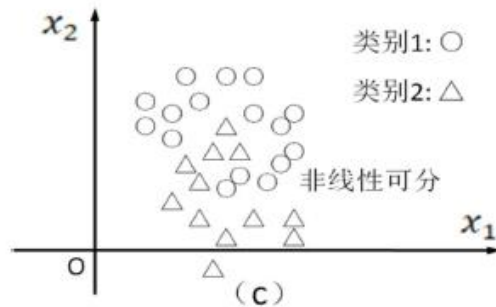
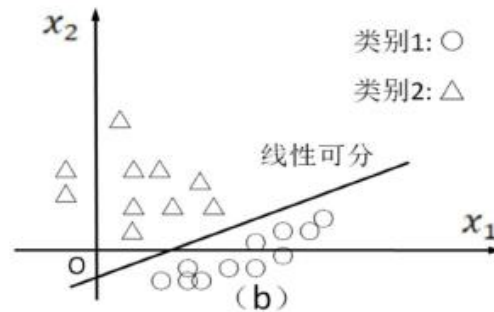
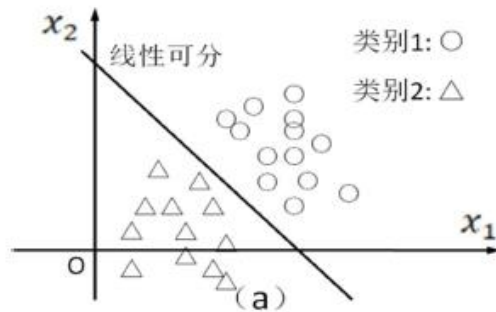
$$f \{ \text{car, money, drive, ...} \}$$

- 喜悦
- 愤怒
- ...

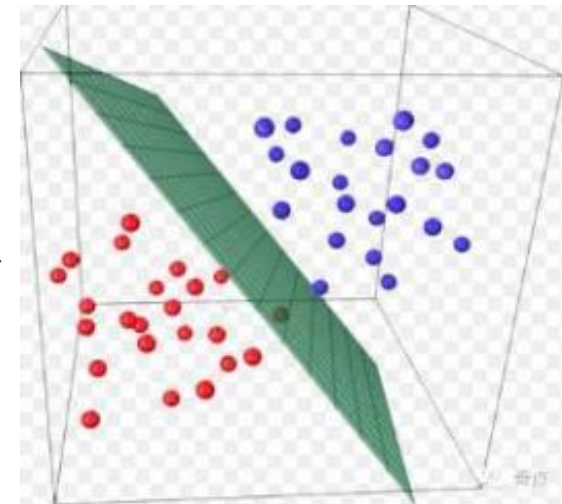
情感分类

- 通过映射函数 f 将原始数据映射到语义任务空间，即寻找数据和任务目标之间的关系

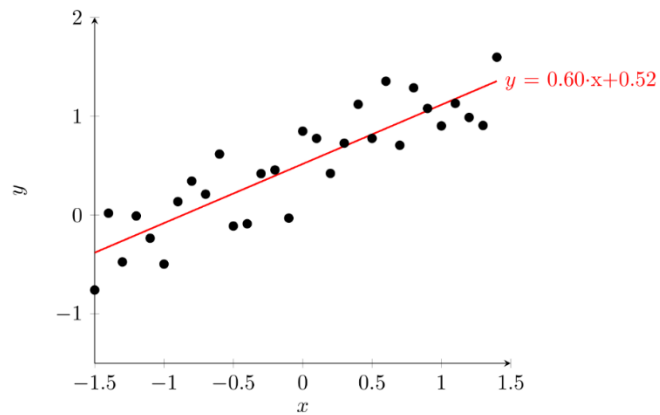
机器学习：从数据中学习知识



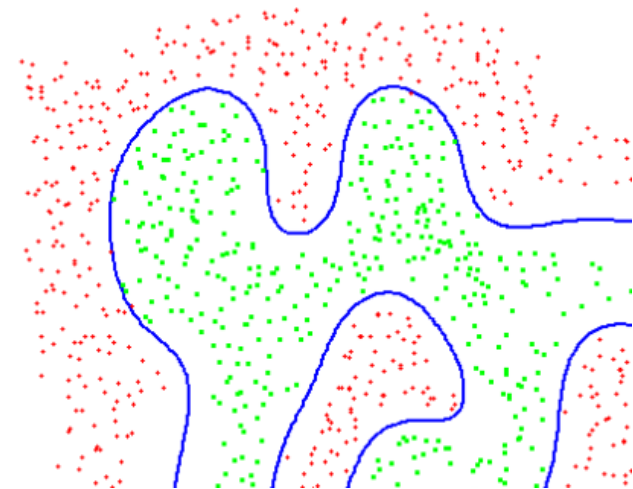
多维



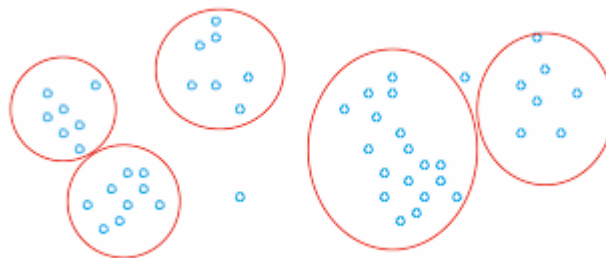
机器学习：常见的机器学习问题



回归（预测）



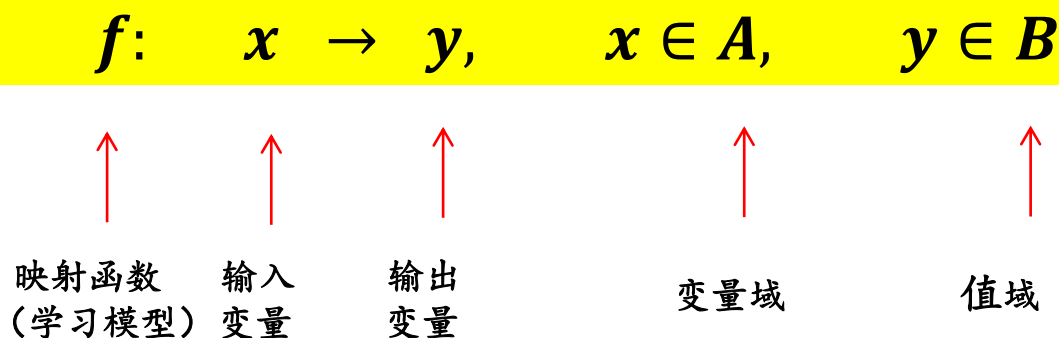
分类



聚类

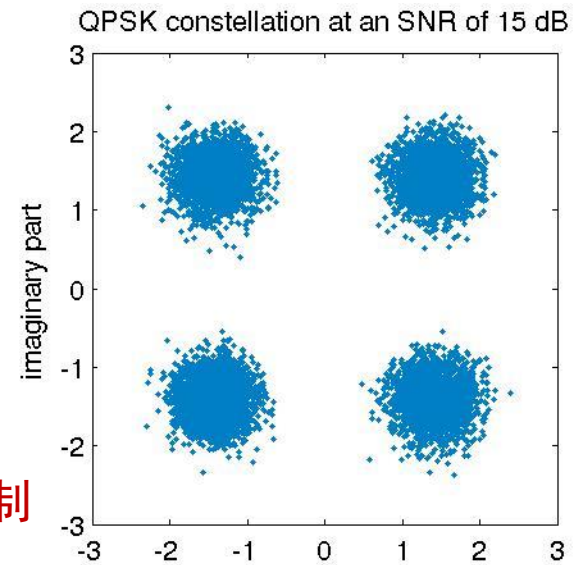
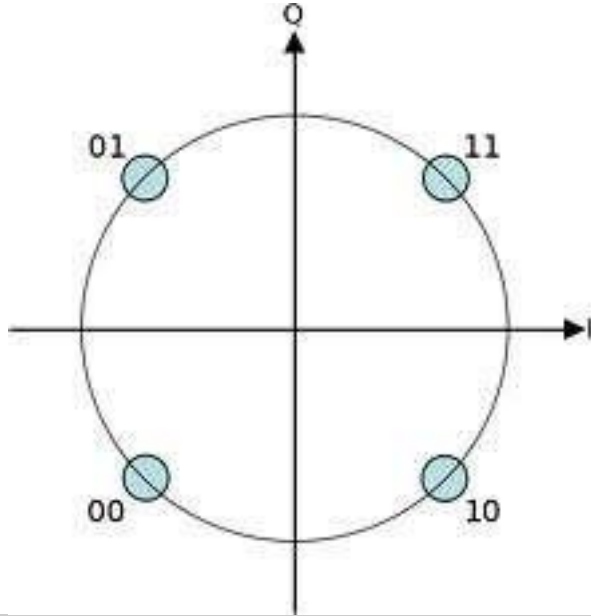
机器学习：回归与分类的区别

- 两者均是学习输入变量和输出变量之间潜在关系模型，基于学习所得模型将输入变量映射到输出变量。



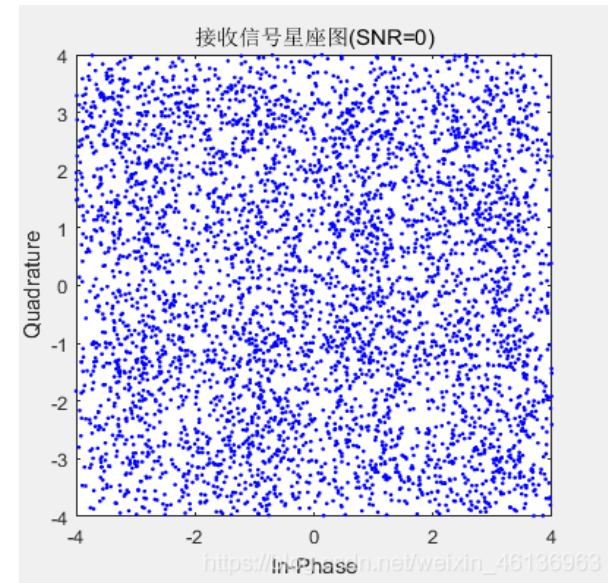
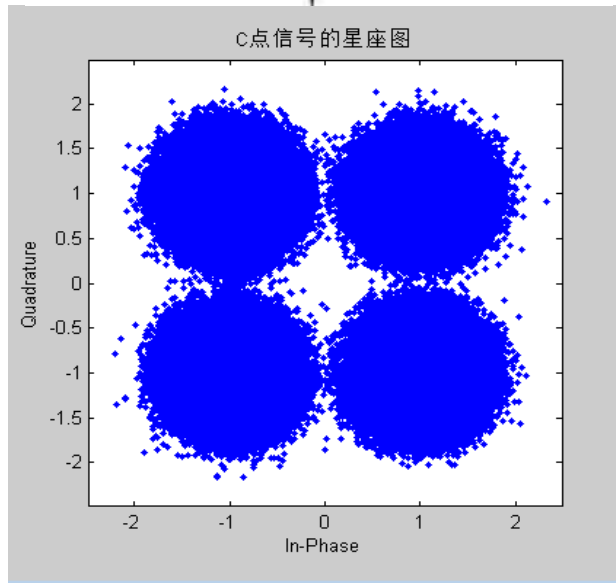
- 监督学习分为回归和分类两个类别。
- 在回归分析中，学习得到一个函数将输入变量映射到**连续**输出空间，如价格和温度等，即值域是**连续空间**。
- 在分类模型中，学习得到一个函数将输入变量映射到**离散**输出空间，如人脸和汽车等，即值域是**离散空间**。

机器学习: QPSK解调 (分类)



四进制相位调制

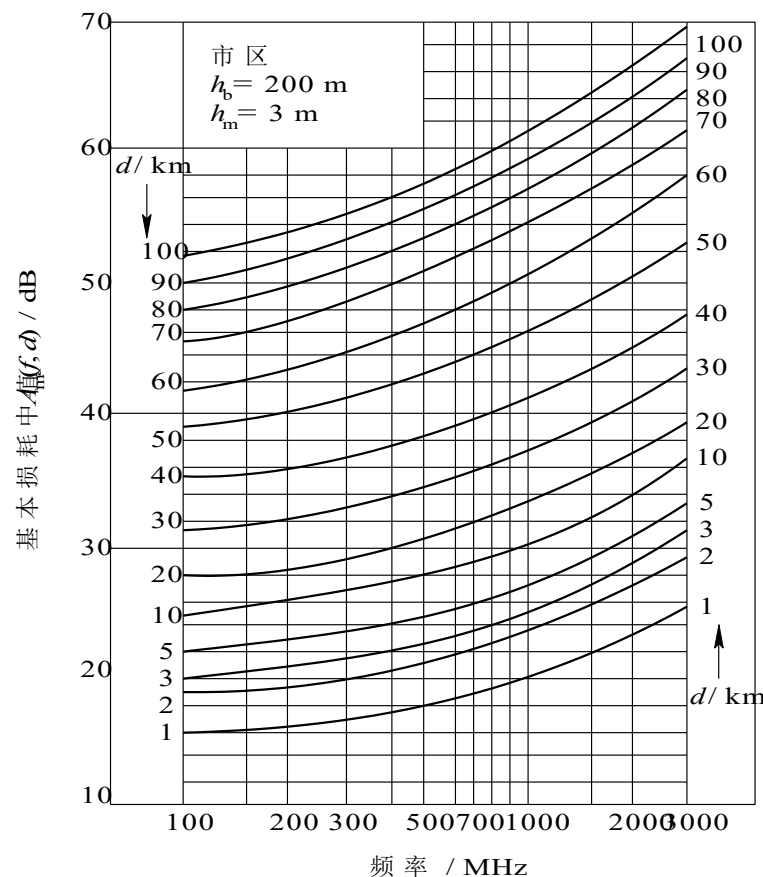
分为四种状态



机器学习：陆地移动信道的传输损耗（预测）

在大量实验、统计分析的基础上，可作出传播损耗基本中值的预测曲线。

20世纪60年代初，**Okumura**等人在日本东京地区进行了大量的场强测试。测试环境（地物特征）包括市区、郊区和开阔区等不同传播环境，测量频率分布在400MHz~2GHz范围内。发射天线高度范围30~1000m，接收天线高度范围2~7m。测量设备（场强计和记录仪）装在汽车上，在汽车行驶中实施测量。



中等起伏地上市区基本损耗中值

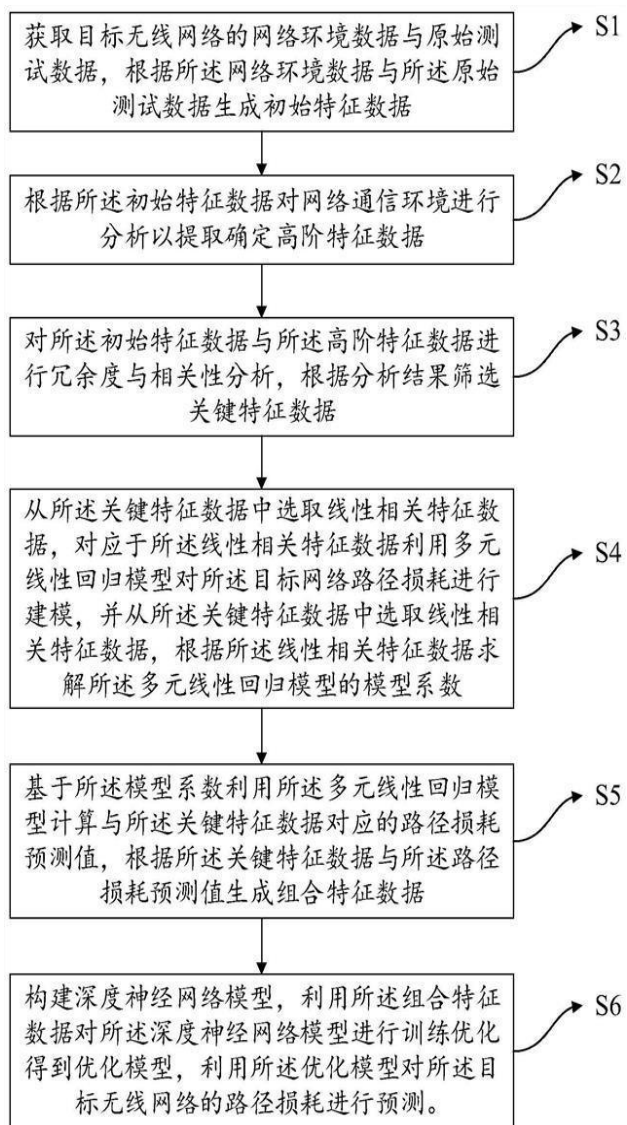
机器学习：传播损耗预测模型-Hata模型（预测）

Hata模型是针对由Okumura用图表给出的路径损耗数据的经验公式，该公式适用于150~1500 MHz频率范围。Hata将市区的传播损耗表示为一个标准的公式和一个应用于其他不同环境的附加校正公式。

在市区的中值路径损耗的标准公式为(CCIR采纳的建议)：

$$L_{\text{urban}}(\text{dB})=69.55+26.16\lg f_c-13.82\lg h_b-a(h_b)+(44.9-6.55\lg h_b)\lg d$$

机器学习：无线网络信号传播路径损耗预测方法及电子设备

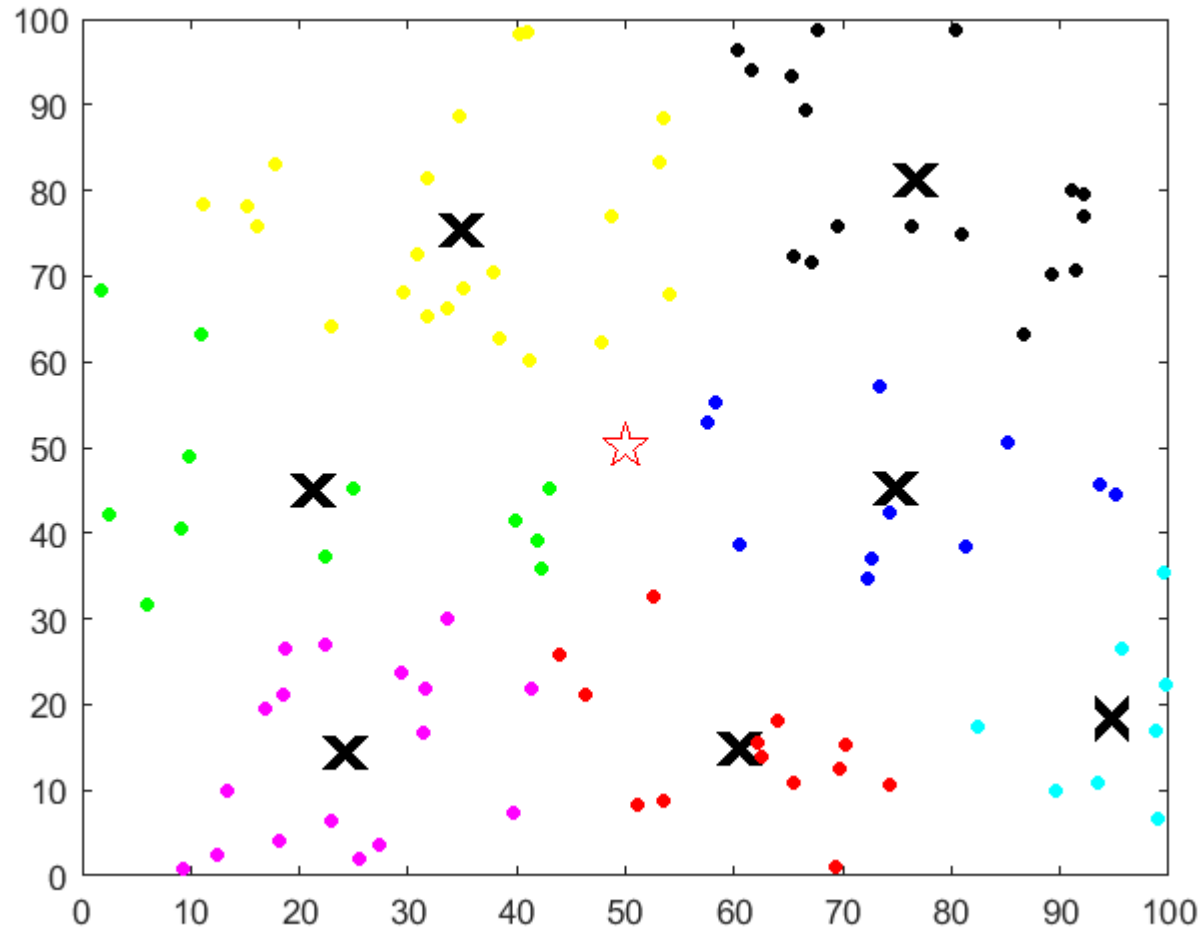


本公开提供一种无线网络信号传播路径损耗**预测**方法及电子设备。所述方法包括：

- 1、获取网络环境数据与原始测试数据，根据网络环境数据、原始测试数据生成初始特征数据并从中提取确定高阶特征数据；
- 2、对初始特征数据与高阶特征数据进行冗余度与相关性分析，筛选出关键特征数据；
- 3、构建**多元线性回归模型**，根据关键特征数据求解模型系数；
- 4、利用多元线性回归模型计算与所述关键特征数据对应的路径损耗预测值以生成组合特征数据；
- 5、构建深度**神经网络模型**，利用所述组合特征数据对所述深度神经网络模型进行训练优化得到优化模型，利用所述优化模型对所述目标无线网络的路径损耗进行预测。

所述电子设备用于实现所述无线网络信号传播路径损耗预测方法。

机器学习：基于K-means聚类算法的无线传感器网络分簇路由协议（聚类）



https://blog.csdn.net/weixin_43821559

https://blog.csdn.net/weixin_43821559/article/details/112687536

机器学习的分类

现有的机器学习种类繁多，我们一般可以进行如下的分类标准：

- ① 是否在人类监督下学习（**监督学习、非监督学习、半监督学习和强化学习**）
- ② 是否可以动态的增量学习（**在线学习和批量学习**）
- ③ 是简单的将新的数据点和已知的数据点进行匹配，还是像科学家那样对训练数据进行模型检测，然后建立一个预测模型（**基于实例的学习和基于模型的学习**）

机器学习的分类

是否在人类监督下学习

监督学习(supervised learning)

数据有标签、一般为回归或分类等任务



半监督学习 (semi-supervised learning)

无监督学习(un-supervised learning)

数据无标签、一般为聚类或若干降维任务

强化学习(reinforcement learning)

序列数据决策学习，一般为与从环境交互中学习

监督学习

- K近邻算法
- 线性回归
- logistic回归
- 支持向量机 (SVM)
- 决策树和随机森林
- 神经网络

无监督学习

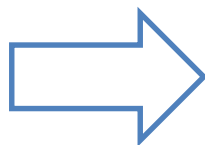
- 聚类算法
 - K均值算法 (K-means)
 - 基于密度的聚类方法 (DBSCAN)
 - 最大期望算法
- 可视化和降维
 - 主成分分析
 - 核主成分分析
- 关联规则学习
 - Apriori
 - Eclat

机器学习：分类问题

人员	数学好	身体好	会编程	嗓门大
程序员A	Yes	No	Yes	Yes
作家A	No	No	Yes	No
程序员B	Yes	Yes	No	No
...
医生A	Yes	Yes	Yes	Yes
程序员C	Yes	Yes	Yes	Yes
程序员D	Yes	Yes	Yes	No

标签数据 (训练集)

从数据
中学习



映射函数(模型)

f

(数学好 = Yes, 会编程 = Yes, 身体好 =?, 嗓门大 =?)

模式

→ 程序员

类别

监督学习的重要元素

标注数据

■ 标识了类别信息的数据
学什么

学习模型

■ 如何学习得到映射模型
如何学

损失函数

■ 如何对学习结果进行度量
学到否

没有免费午餐定理（NFL）: Wolpert 和 Macready 1997 年在最优化理论中提出，指出“任何机器学习模型在所有问题上的性能都是相同的，其总误差和模型本身是没有关系的。一种算法（算法A）在特定数据集上的表现优于另一种算法（算法B）的同时，一定伴随着算法A在另外某一个特定的数据集上有着不如算法B的表现”

Wolpert, D.H., Macready, W.G. (1997), No Free Lunch Theorems for Optimization, IEEE Transactions on Evolutionary Computation 1, 67

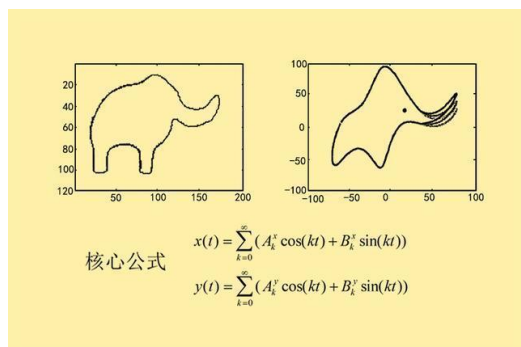
机器学习的初心：化繁为简、大道至简

机器学习是一种“数据驱动学习(data-driven learning)”的范式，其从数据出发来学习数据中所蕴含的模式，对数据进行抽象。统计学家Ronald Aylmer Fisher将这一过程概括为“化繁为简（the object of statistical methods is the reduction of data）”。

REQUIRE FOR LARGE-SCALE MODELS *

Douglas B. Lee, Jr.

collapsed rather than evolved



The task in this paper is to evaluate, in some detail, the fundamental flaws in attempts to construct and use large models and to examine the planning context in which the models, like dinosaurs, collapsed rather than evolved. The conclusions can be summarized in three points:

1. In general, none of the goals held out for large-scale models have been achieved, and there is little reason to expect anything different in the future.
2. For each objective offered as a reason for building a model, there is either a better way of achieving the objective (more information at less cost) or a better objective (a more socially useful question to ask).
3. Methods for long-range planning—whether they are called comprehensive planning, large-scale systems simulation, or something else—need to change drastically if planners expect to have any influence on the long run.

Almost a decade ago, John Repp presented a paper to planners in which he attacked traditional modes of land-use control and offered alternatives: his paper was titled “Requiem for Zoning.” This attack, directed at physical planners from one of their own, came at a time when many thought that mathematical models and computer data banks would overturn the field. His effort deserves a symmetrical gesture. [1964]

This paper is about large-scale urban models. The characteristics exhibited by these models are

(1) they are large in the sense that the only practical way to operate them is on a computer; (2) commonly they are spatially disaggregated, and allocate activities to geographic zones; and (3) they pertain to a single specific metropolitan area, as opposed to being generalized abstract or hypothetical models. The epitome of the genre is the comprehensive land-use model of the type constructed in the middle of the last decade.

These models were begun in the early 1960's and largely abandoned by the end of the 1980's. Considerable effort was expended on them, and a good deal was learned. Contrary to what has often been claimed, what was learned had almost nothing to do with urban spatial structure; the knowledge that was increased was our understanding of model building and its relationship to policy analysis. For that alone it was a valuable experience, but not if the lessons are ignored. For many in planning and many in a number of related fields that have recently become interested in planning, the lessons are being ignored.

Some planners never accepted models as legitimate activity of the field, and they will claim this paper vindicates their position. This is incorrect: there was a need at that time for better analytic and quantitative procedures, and there was also a need for the development of better theory. Now, the need for both theory and method is even greater. It is not our intent to discourage those who would apply quantitative methods to urban problems; but, rather, to redirect their talents into more valuable pursuits than repeating the mistakes of the last decade.

A prototypical land-use model is broken down into subareas (called zones or districts) generally

An example of this viewpoint is Raymond (in Fisher, 1970). A JOURNAL reviewer offered the Raymond article as a possible supporting reference; in fact, my position is diametrically opposed to Raymond's. Especially conscientious readers might care to compare the two articles.



冯诺依曼说用四个参数我可以拟合出一头大象，而用五个参数我可以让它的鼻子晃。

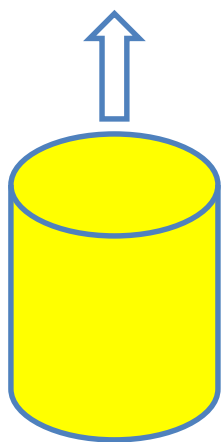
加州大学伯克利分校助理教授针对交通预测模型越来越复杂，阐述了在交通预测领域中使用大模型的七宗罪

公元 14 世纪，来自奥卡姆的威廉对当时无休无止的关于“共相”、“本质”之类的争吵感到厌倦，于是著书立说，宣传：“如无必要，勿增实体”。

监督学习：损失函数

训练映射函数 f

使得 $f(x_i)$ 预测结果尽量等于 y_i



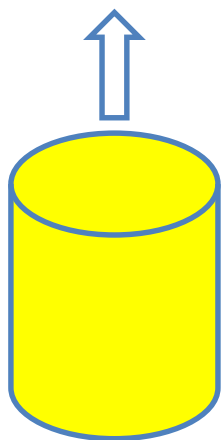
训练数据集
 $(x_i, y_i), i = 1, \dots, n$

- 训练集中一共有 n 个标注数据，第 i 个标注数据记为 (x_i, y_i) ，其中第 i 个**样本数据**为 x_i ， y_i 是 x_i 的**标注信息**。
- 从训练数据中学习得到的**映射函数**记为 f ， f 对 x_i 的**预测结果**记为 $f(x_i)$ 。**损失函数**就是用来计算 x_i 对应的标注值 y_i 与预测值 $f(x_i)$ 之间**差值（残差）**的函数。
- 很显然，在训练过程中希望映射函数在训练数据集上得到“损失”之和最小，即 $\min \sum_{i=1}^n \text{Loss}(f(x_i), y_i)$ 。

监督学习：损失函数

训练映射函数 f

使得 $f(x_i)$ 预测结果尽量等于 y_i



训练数据集

$(x_i, y_i), i = 1, \dots, n$

损失函数名称	损失函数定义
0-1损失函数	$Loss(y_i, f(x_i)) = \begin{cases} 1, f(x_i) \neq y_i \\ 0, f(x_i) = y_i \end{cases}$
平方损失函数	$Loss(y_i, f(x_i)) = (y_i - f(x_i))^2$
绝对损失函数	$Loss(y_i, f(x_i)) = y_i - f(x_i) $
对数损失函数/ 对数似然损失 函数/交叉熵损 失函数	$Loss(y_i, P(y_i x_i)) = -\log P((y_i x_i))$ 使用最大似然估计 (Maximum likelihood estimation) 来构造损失函数，找到使 $p(y_i x_i)$ 最大的函数 $f(x_i)$ 。

典型的损失函数

监督学习：交叉熵损失函数

交叉熵损失函数（Cross-Entropy Loss Function）一般用于分类问题。假设样本的标签 $y \in \{1, \dots, C\}$ 为离散的类别，模型 $f(\mathbf{x}; \theta) \in [0, 1]^C$ 的输出为类别标签的条件概率分布，即

$$p(y = c | \mathbf{x}; \theta) = f_c(\mathbf{x}; \theta)$$

用 $f_c(\mathbf{x}; \theta)$ 表示 $f(\mathbf{x}; \theta)$ 的输出向量的第 c 维（第 c 个元素），并满足：

$$f_c(\mathbf{x}; \theta) \in [0, 1], \quad \sum_{c=1}^C f_c(\mathbf{x}; \theta) = 1$$

我们可以用一个 C 维的 **one-hot 向量** \mathbf{y} 来表示样本的标签。假设样本的标签为 k ，那么标签向量 \mathbf{y} 只有第 k 维的值为 1，其余元素的值都为 0（只属于 1 个类别）。标签向量 \mathbf{y} 可以看作样本标签的 **真实条件概率分布** $p_r(\mathbf{y} | \mathbf{x})$ ，即第 c 维（记为 y_c ， $1 \leq c \leq C$ ）是类别为 c 的真实条件概率。假设样本的类别为 k ，那么它属于第 k 类别的概率为 1，属于其他类的概率为 0。

监督学习：交叉熵损失函数

对于两个概率分布，一般可以用交叉熵来衡量它们的差异。交叉熵主要刻画的是两个概率分布，比如**预测分布**（概率）与**真实分布**（概率）的距离，也就是交叉熵的值越小，两个概率分布就越接近。

标签的**真实分布** y (C 维的one-hot列向量)和模型**预测分布** $f(x; \theta)$ 之间的交叉熵为：

$$\mathcal{L}(y, f(x; \theta)) = -y^T \log f(x; \theta) = -\sum_{c=1}^C y_c \log f_c(x; \theta)$$

因为 y 为one-hot向量，也可以写为：

$$\mathcal{L}(y, f(x; \theta)) = -\log f_y(x; \theta)$$

其中 $f_y(x; \theta)$ 可以看作类别 y 的似然函数。因此，**交叉熵损失函数也就是负对数似然函数**（Negative Log-Likelihood）。

监督学习：交叉熵损失函数

例1:

对于3分类问题($C=3$), 某标签向量为 $\mathbf{y}=[0, 0, 1]^T$, 模型预测的标签分布为 $f(\mathbf{x}; \theta)=[0.3, 0.3, 0.4]^T$, 则它们的交叉熵为 $-(0 \times \log(0.3) + 0 \times \log(0.3) + 1 \times \log(0.4)) = -\log(0.4)$ 。

例2:

一个图像分类任务：我们希望根据图片动物的轮廓、颜色等特征（构成 \mathbf{x} ，可以是像素点灰度构成的向量），来预测动物的类别，有3种可预测类别：猫、狗、猪。假设我们当前有2个模型（参数不同），这2个模型都是通过sigmoid/softmax的方式得到对于每个预测结果的概率值：

<https://zhuanlan.zhihu.com/p/35709485>

监督学习：交叉熵损失函数

模型1：

预测 $f(x_i)$	真实 y_i	是否正确
0.3 0.3 0.4	0 0 1 (猪)	正确
0.3 0.4 0.3	0 1 0 (狗)	正确
0.1 0.2 0.7	1 0 0 (猫)	错误

模型1对于样本1和样本2以非常微弱的优势判断正确，对于样本3的判断则彻底错误。

模型2：

预测	真实	是否正确
0.1 0.2 0.7	0 0 1 (猪)	正确
0.1 0.7 0.2	0 1 0 (狗)	正确
0.3 0.4 0.3	1 0 0 (猫)	错误

模型2对于样本1和样本2判断非常准确，对于样本3判断错误，但是相对来说没有错得太离谱。

监督学习：交叉熵损失函数

模型1:

$$\text{sample 1 loss} = -(0 \times \log 0.3 + 0 \times \log 0.3 + 1 \times \log 0.4) = 0.91$$

$$\text{sample 2 loss} = -(0 \times \log 0.3 + 1 \times \log 0.4 + 0 \times \log 0.3) = 0.91$$

$$\text{sample 3 loss} = -(1 \times \log 0.1 + 0 \times \log 0.2 + 0 \times \log 0.7) = \underline{2.30}$$

对所有样本的loss求平均:

$$L = \frac{0.91 + 0.91 + 2.3}{3} = 1.37$$

模型2:

$$\text{sample 1 loss} = -(0 \times \log 0.1 + 0 \times \log 0.2 + 1 \times \log 0.7) = 0.35$$

$$\text{sample 2 loss} = -(0 \times \log 0.1 + 1 \times \log 0.7 + 0 \times \log 0.2) = 0.35$$

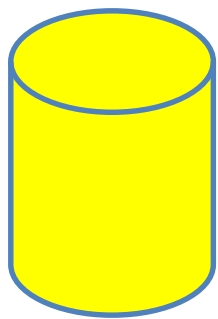
$$\text{sample 3 loss} = -(1 \times \log 0.3 + 0 \times \log 0.4 + 0 \times \log 0.4) = \underline{1.20}$$

对所有样本的loss求平均:

$$L = \frac{0.35 + 0.35 + 1.2}{3} = 0.63$$

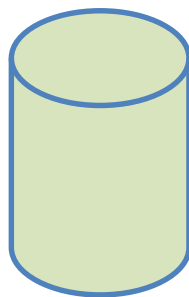
监督学习：训练数据与测试数据

从训练数据集学习
得到映射函数 f 和参数



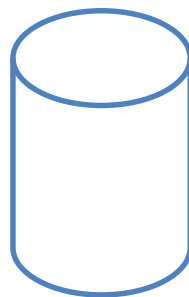
训练数据集
 $(x_i, y_i), i = 1, \dots, n$

通过验证数据
集调参，提高
学习能力



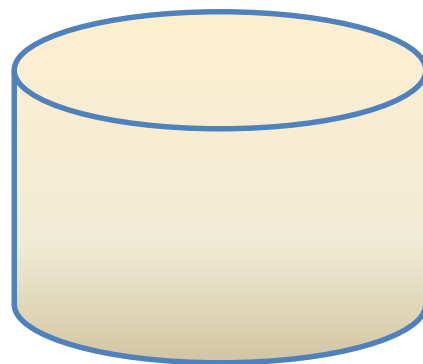
验证数据集

在测试数据集
测试映射函数 f 和参数



测试数据集
 $(x_i', y_i'), i = 1, \dots, m$

未知数据集
上使用映射函数 f



没有标签了

监督学习：经验风险与期望风险

损失函数：度量模型一次预测的好坏(0-1、平方、绝对值)

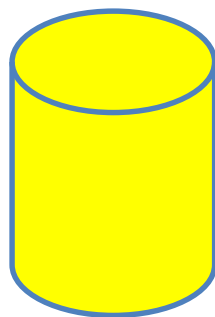
风险函数：度量平均意义下的**模型**预测好坏(期望风险、经验风险、结构风险)

从训练数据集学
习得到映射函数 f

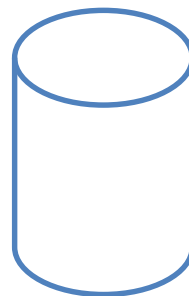
在测试数据集
测试映射函数 f

经验风险(empirical risk)

- 模型在训练集中数据产生的**损失**。经验风险越小说明学习模型对训练数据拟合程度越好。



训练数据集
 $(x_i, y_i), i = 1, \dots, n$



测试数据集
 $(x'_i, y'_i), i = 1, \dots, m$

期望风险(expected risk):

- 当模型在所有样本（训练集、测试集和其他样本）中产生的**损失**。期望风险越小，学习所得模型越好。

监督学习：经验风险与期望风险

映射函数训练目标：**经验风险**最小化
(empirical risk minimization, ERM)

算数平均：

$$\min_{f \in \Phi} \frac{1}{n} \sum_{i=1}^n \text{Loss}(y_i, f(x_i))$$

选取一个使得训练集所有数据
损失平均值最小的映射函数。
这样的考虑是否够？



训练数据集
 $(x_i, y_i), i = 1, \dots, n$

映射函数训练目标：**期望风险**最小化
(expected risk minimization)

概率平均：

$$\min_{f \in \Phi} \int_{x \times y} \text{Loss}(y, f(x)) P(x, y) dx dy$$

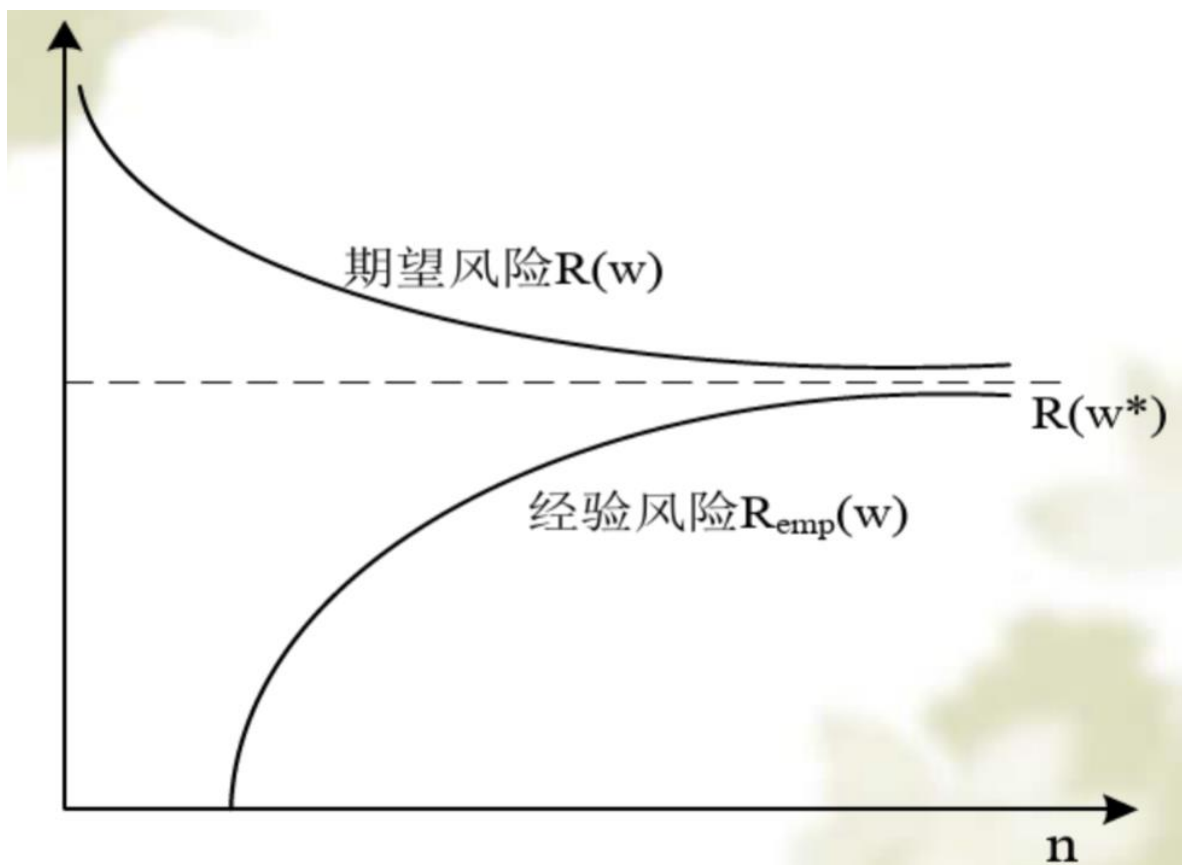


测试数据集数据无穷多
 $(x_i', y_i'), i = 1, \dots, \infty$

- 经验风险是模型关于训练样本集平均损失，期望风险是模型关于联合分布期望损失。
- 根据大数定律，当样本容量趋于无穷时，经验风险趋于期望风险。所以在实践中很自然用经验风险来估计期望风险。
- 由于现实中训练样本数目有限，用经验风险估计期望风险并不理想，要对经验风险进行一定的约束。

监督学习：经验风险与期望风险

- **学习的一致性**：当训练样本数趋于无穷大时，经验风险的最优值收敛到期望风险的最优值。（为什么期望风险大于经验风险？为什么随着 n 增加，期望风险降低，经验风险增加？）



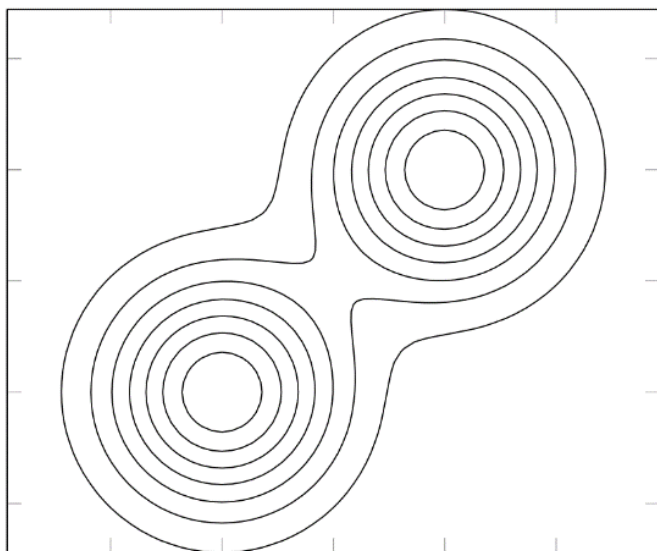
大数定理

样本数

监督学习：经验风险与期望风险

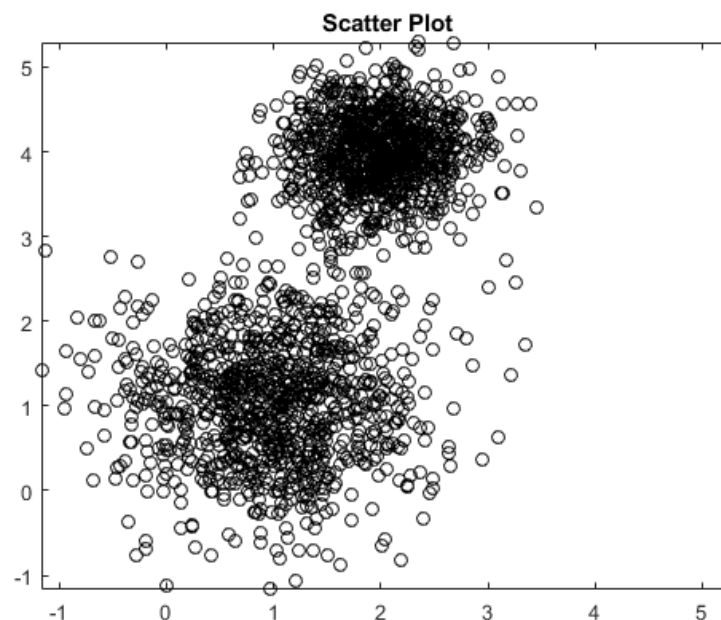
期望风险

真实分布 p_r



经验风险

\neq



泛化误差 = 期望风险 - 经验风险

泛化误差：模型在新样本上的误差。

监督学习：“过学习(over-fitting)”与“欠学习(under-fitting)”

经验风险最小化

$$\min_{f \in \Phi} \frac{1}{n} \sum_{i=1}^n \text{Loss}(y_i, f(x_i))$$

期望风险最小化

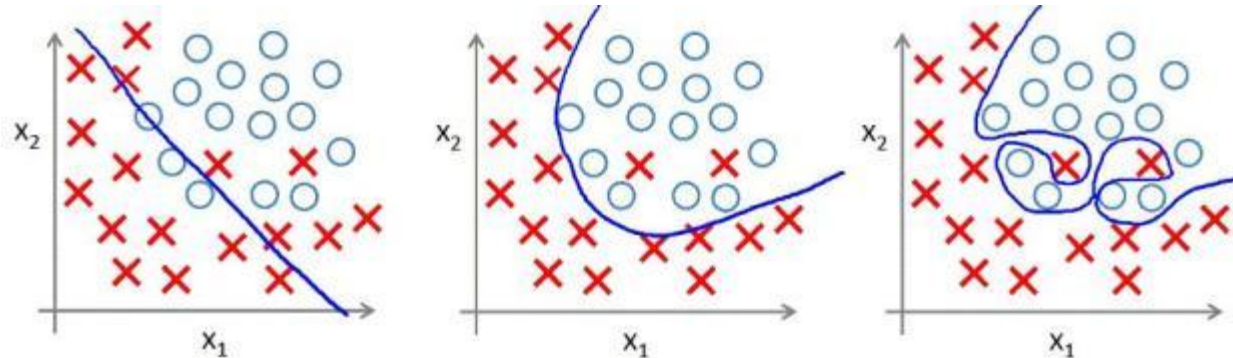
$$\min_{f \in \Phi} \int_{x \times y} \text{Loss}(y, f(x)) P(x, y) dx dy$$

经验风险小（训练集上表现好）	期望风险小（测试集上表现好）	泛化能力强
经验风险小（训练集上表现好）	期望风险大（测试集上表现不好）	过学习/过拟合 （模型过于复杂）
经验风险大（训练集上表现不好）	期望风险大（测试集上表现不好）	欠学习/欠拟合
经验风险大（训练集上表现不好）	期望风险小（测试集上表现好）	“神仙算法” 或“黄粱美梦”

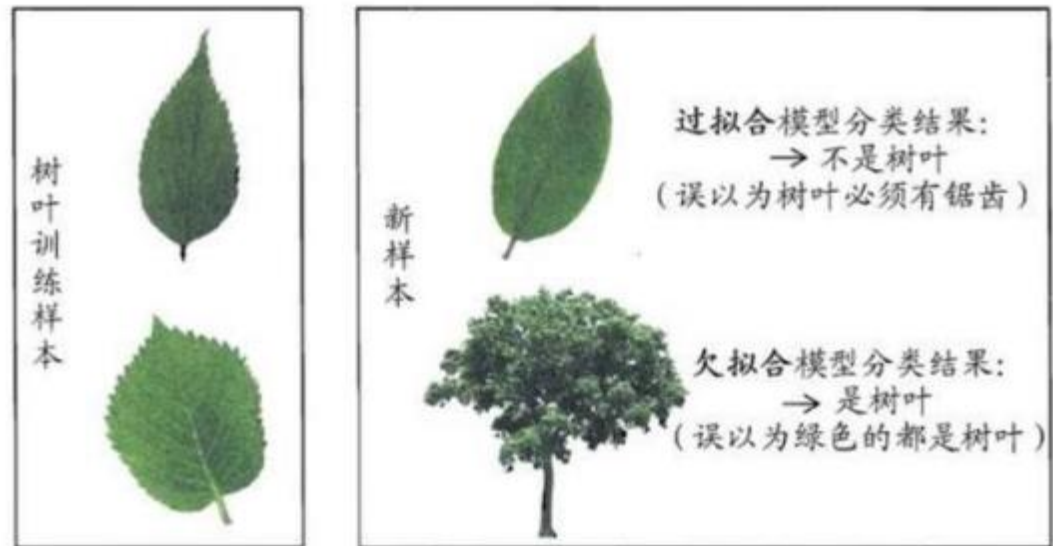
表4.3 模型泛化能力与经验风险、期望风险的关系

监督学习：过学习与欠学习

通俗对比：欠学习学得太少，分得太粗糙；过学习学得太多太细，拿着放大镜看世界，看到的都是差异看不到相同点。

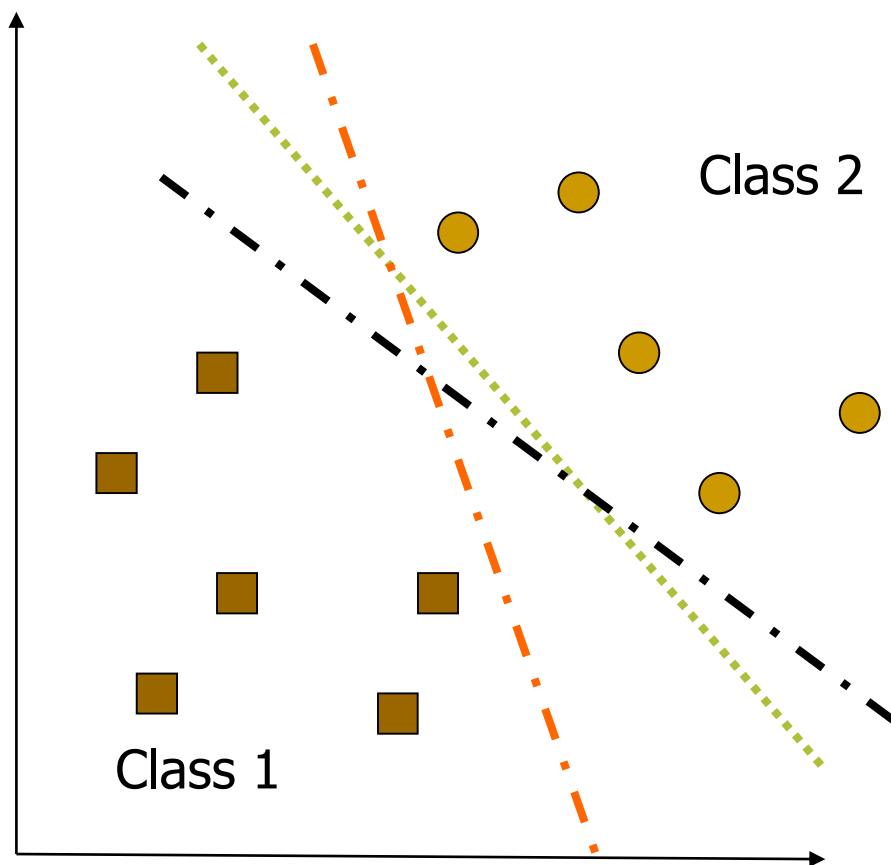


过学习：**经验风险最小化原则**很容易导致模型在训练集上错误率很低，但是在未知数据上错误率很高。过学习问题往往是由于训练数据少（往往需要大数据）和噪声等原因造成的。



过拟合、欠拟合的直观类比

监督学习:最优分类面



目标: 最优分类面

满足条件: 1、经验风险最小 (错分最少) ; 2、泛化能力最大 (空白最大)

监督学习: 结构风险最小

- 当样本容量足够大时, 经验风险最小化能够保证很好的学习效果, 在现实中被广泛采用, 如极大似然估计。
- 然而通常情况下, 我们无法获取无限的训练样本, 并且训练样本往往是真实数据的一个很小的子集或者包含一定的噪声数据, 不能很好地反映全部数据的真实分布。经验风险最小化原则很容易导致模型在训练集上错误率很低, 但是在未知数据上错误率很高。这就是所谓的过拟合 (over-fitting) 现象。
- 结构风险最小化 (Structural Risk Minimization, SRM) 准则是为了防止过拟合而提出来的策略。过拟合问题往往是由于训练数据少和噪声以及模型能力强等原因造成的。为了解决过拟合问题, 一般在经验风险最小化的基础上再引入参数的正则化 (regularization), 来限制模型能力, 使其不要过度地最小化经验风险。

结构风险=经验风险+正则化项

结构风险最小化 \approx 模型参数正则化

监督学习: 结构风险最小

经验风险最小化: 仅反映了局部数据

$$\min_{f \in \Phi} \frac{1}{n} \sum_{i=1}^n \text{Loss}(y_i, f(x_i))$$

期望风险最小化: 无法得到全量数据

$$\min_{f \in \Phi} \int_{x \times y} \text{Loss}(y, f(x)) P(x, y) dx dy$$

结构风险最小化(structural risk minimization):

为了防止过拟合, 在经验风险上加上表示模型复杂度的正则化项 (regulatizer) 或惩罚项 (penalty term), 即通过减小模型复杂度来防止过拟合。

$$\min_{f \in \Phi} \frac{1}{n} \sum_{i=1}^n \text{Loss}(y_i, f(x_i)) + \lambda J(f)$$

经验风险 模型复杂度

在最小化经验风险与降低模型复杂度之间寻找平衡

监督学习: 结构风险最小

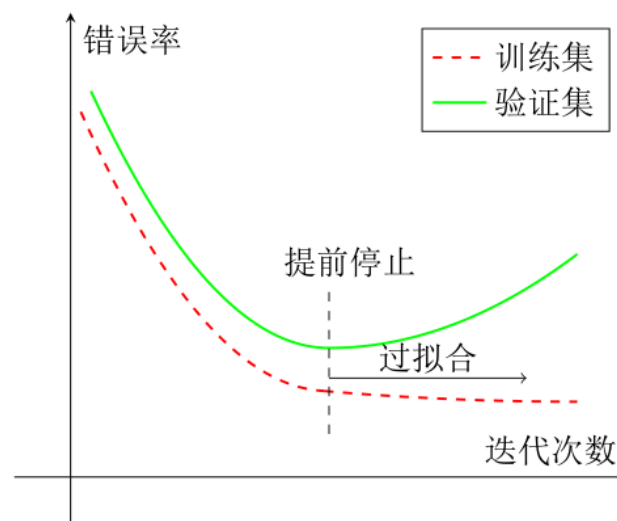
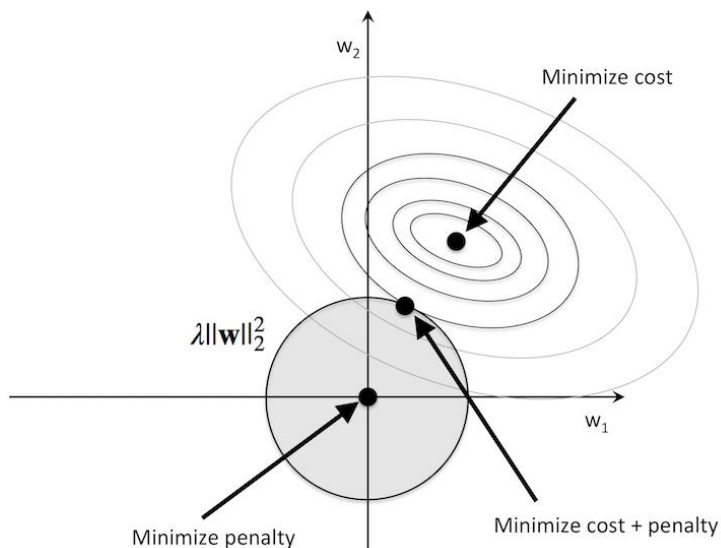
所有损害“优化”的方法都是正则化。

增加优化约束

L1/L2约束、数据增强

干扰优化过程

权重衰减、随机梯度下降、提前停止



监督学习两种方法：判别模型与生成模型

监督学习方法又可以分为生成方法 (generative approach) 和判别方法 (discriminative approach)。所学到的模型分别称为生成模型 (generative model) 和判别模型 (discriminative model)。

- 判别方法直接学习判别函数 $f(X)$ 或者条件概率分布 $P(Y|X)$ 作为预测的模型，即判别模型。
- 判别模型关心在给定输入数据下，预测该数据的输出是什么。
- 典型判别模型包括回归模型、神经网络、支持向量机和 Ada boosting 等。

$$f(\text{人脸}) \longrightarrow \text{人脸}$$

$$P(\text{人脸} | \text{人脸}) = 0.99$$

监督学习两种方法：判别模型与生成模型

- 生成模型从数据中学习联合概率分布 $P(X, Y)$ （通过似然概率 $P(X|Y)$ 和类概率 $P(Y)$ 的乘积来求取）

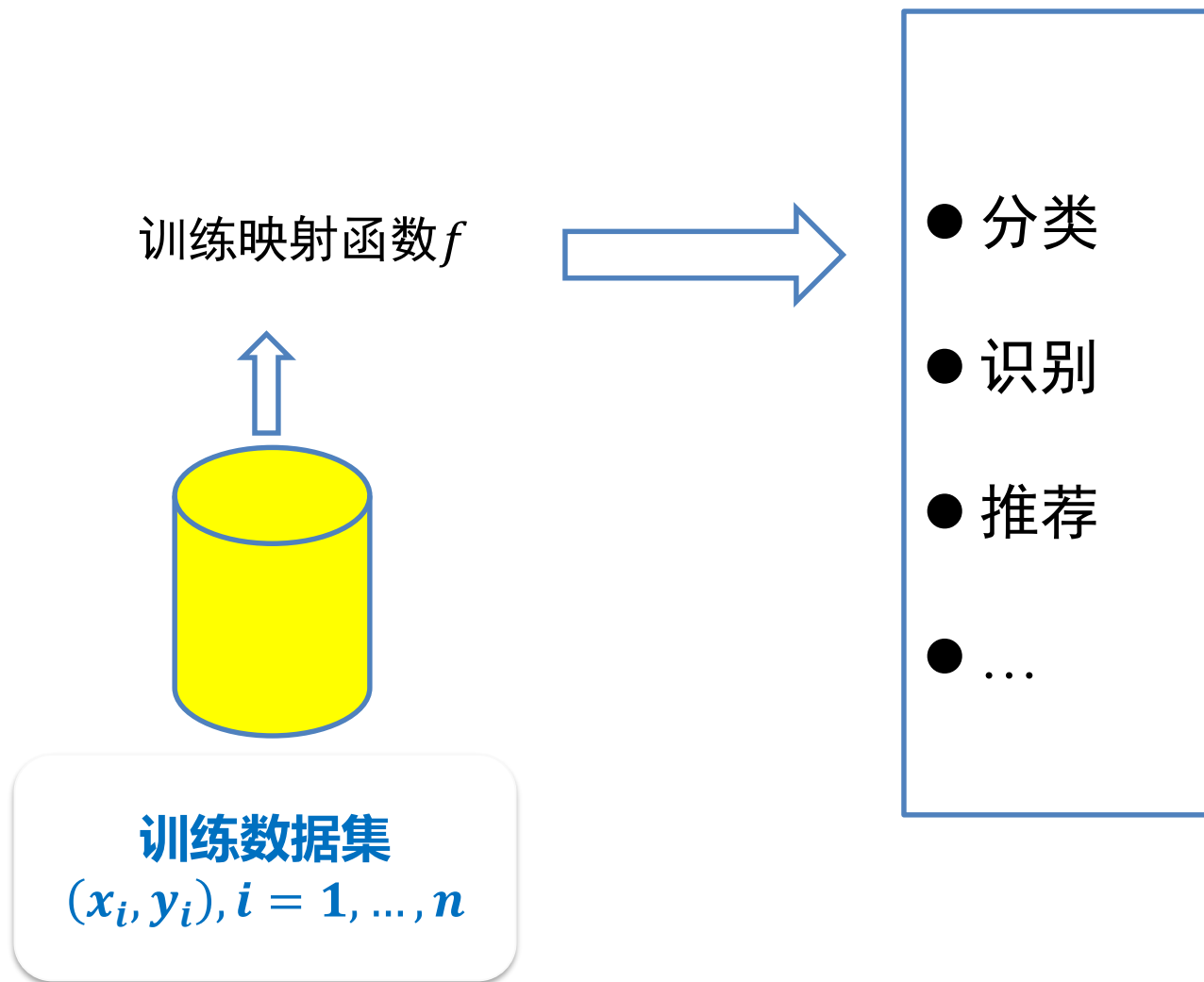
$$P(Y|X) = \frac{P(X, Y)}{P(X)} \text{ 或者 } P(Y|X) = \frac{P(X|Y) \times P(Y)}{P(X)}$$

- 典型方法为贝叶斯方法、隐马尔可夫链
- 授之于鱼、不如授之于“渔”
- 联合分布概率 $P(X, Y)$ 或似然概率 $P(X|Y)$ 求取很困难

似然概率：计算
导致样本 X 出现的
模型参数值

$$P(Y|X) = \frac{P(X|Y) \times P(Y)}{P(X)}$$

监督学习



提纲

一、机器学习基本概念

二、回归分析

三、决策树

四、线性判别分析

五、Ada Boosting

六、支持向量机

七、生成学习模型

线性回归 (linear regression)

- 在现实生活中，往往需要分析若干变量之间的关系，如碳排放量与气候变暖之间的关系、某一商品广告投入量与该商品销售量之间的关系等，这种分析不同变量之间存在关系的研究叫回归分析，刻画不同变量之间关系的模型被称为回归模型。**如果这个模型是线性的，则称为线性回归模型。**
- 一旦确定了回归模型，就可以进行预测等分析工作，如从碳排放量预测气候变化程度、从广告投入量预测商品销售量等。

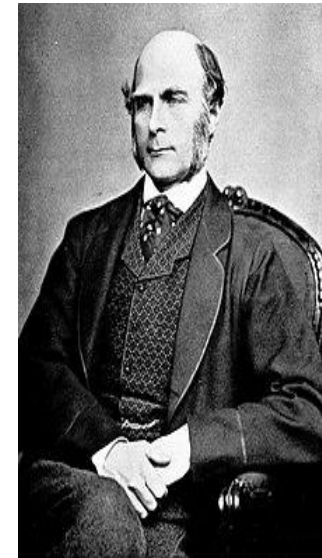
线性回归 (linear regression)

$$y = 33.73(\text{英寸}) + 0.516x$$

y : 子女平均身高

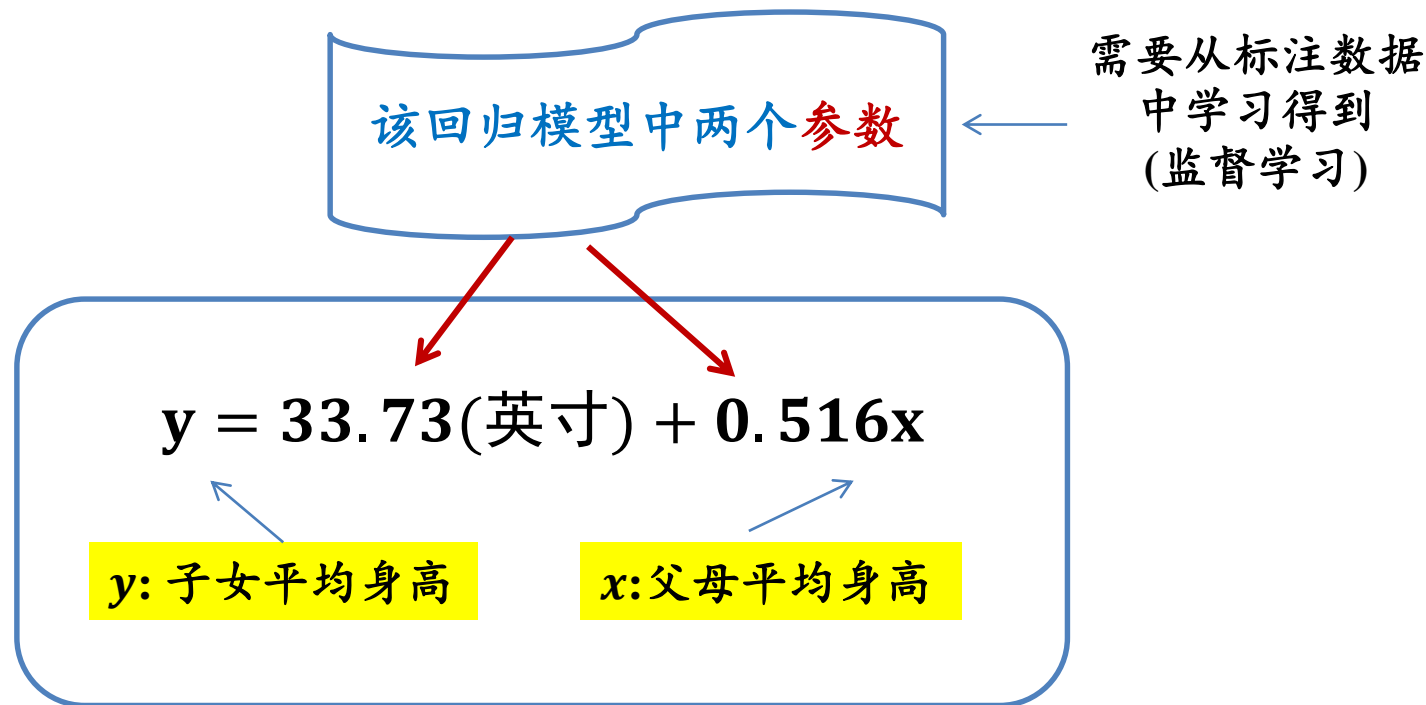
x : 父母平均身高

- 父母平均身高每增加一个单位，其成年子女平均身高只增加0.516个单位，它反映了这种“衰退 (regression)”效应（“回归”到正常人平均身高）。
- 虽然 x 和 y 之间并不总是具有“衰退”（回归）关系，但是“线性回归”这一名称就保留下来了。



英国著名生物学家兼
统计学家高尔顿
Sir Francis Galton
(1822-1911)

线性回归 (linear regression)



- 给出任意一对父母平均身高，则可根据上述方程，计算得到其子女平均身高
- 从父母平均身高来预测其子女平均身高
- 如何求取上述线性方程（预测方程）的参数？

线性回归：一元线性回归

一元线性回归模型例子

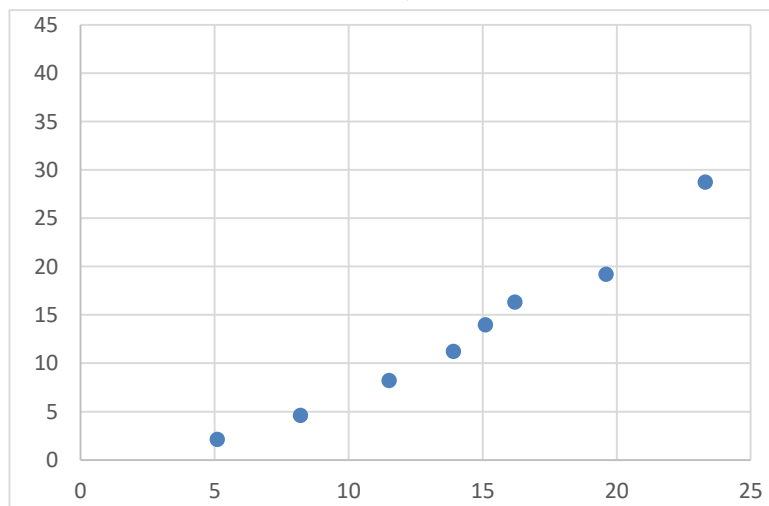
下表给出了芒提兹尼欧（Montesinho）地区发生森林火灾的部分历史数据。

火灾次数

气温温度 x	5.1	8.2	11.5	13.9	15.1	16.2	19.6	23.3
火灾影响面积 y	2.14	4.62	8.24	11.24	13.99	16.33	19.23	28.74

可否对 x 和 y 进行建模呢？

初步观察之后，可以使用简单的线性模型构建两者之间关系，即 x 与 y 之间存在 $y = ax + b$ 形式的关系。



线性回归：一元线性回归

一元线性回归模型例子

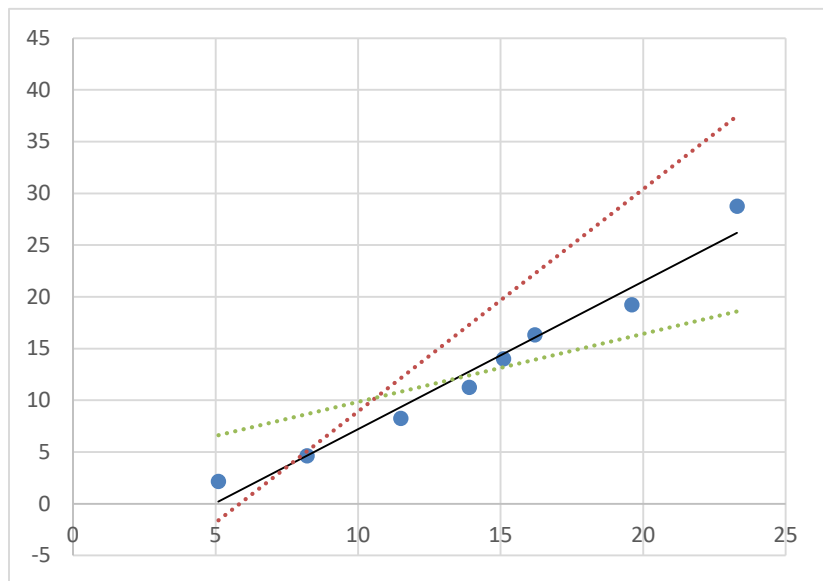


图4.2 气温温度取值和受到火灾影响森林面积之间的一元线性回归模型（实线为最佳回归模型）

回归模型： $y = ax + b$

求取：最佳回归模型是最小化**残差**平方和的均值（类似**均方误差**），即要求8组 (x, y) 数据得到的残差平方的样本平均值 $\frac{1}{N} \sum (y - \tilde{y})^2$ 最小， $(\tilde{y} = f(x))$ 。
残差平均值最小只与参数 a 和 b 有关，最优解即是使得残差最小所对应的 a 和 b 的值。

残差：预测值（拟合值、估计值）与标注值之间的差；

误差：观测值与真实值之间的差。

线性回归：一元线性回归

回归模型参数求取： $y_i = ax_i + b \ (1 \leq i \leq n)$

- 记在当前参数下第 i 个训练样本 x_i 的预测值为 \hat{y}_i
- x_i 的标注值（实际值） y_i 与预测值 \hat{y}_i 之差记为 $(y_i - \hat{y}_i)^2$
- 训练集中 n 个样本所产生残差总和为： $L(a, b) = \sum_{i=1}^n (y_i - a \times x_i - b)^2$

目标：寻找一组 a 和 b ，使得残差总和 $L(a, b)$ 值最小。在线性回归中，解决如此目标的方法叫最小二乘法。

一般而言，要使函数具有最小值，可对 $L(a, b)$ 参数 a 和 b 分别求导，令其导数值为零，再求取参数 a 和 b 的取值。

线性回归：一元线性回归

回归模型参数求取： $y_i = ax_i + b$ ($1 \leq i \leq n$) $\min_{a,b} L(a,b) = \sum_{i=1}^n (y_i - a \times x_i - b)^2$

1、对**b**求偏导

$$\frac{\partial L(a,b)}{\partial b} = \sum_{i=1}^n 2(y_i - ax_i - b)(-1) = 0$$

$$\rightarrow \sum_{i=1}^n (y_i - ax_i - b) = 0$$

$$\rightarrow \sum_{i=1}^n (y_i) - a \sum_{i=1}^n x_i - \sum_{i=1}^n b = 0$$

$$\rightarrow n\bar{y} - an\bar{x} - nb = 0$$



$$b = \bar{y} - a\bar{x}$$

线性回归：一元线性回归

回归模型参数求取： $y_i = ax_i + b$ ($1 \leq i \leq n$) $\min_{a,b} L(a,b) = \sum_{i=1}^n (y_i - a \times x_i - b)^2$

2、对a求偏导

$$\frac{\partial L(a,b)}{\partial a} = \sum_{i=1}^n 2(y_i - ax_i - b)(-x_i) = 0$$

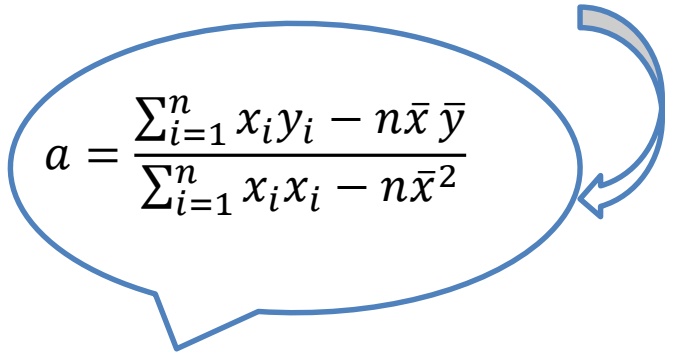
将 $b = \bar{y} - a\bar{x}$ ($\bar{y} = \frac{\sum_{i=1}^n y_i}{n}$, $\bar{x} = \frac{\sum_{i=1}^n x_i}{n}$) 代入上式

$$\rightarrow \sum_{i=1}^n (y_i - ax_i - \bar{y} + a\bar{x})(x_i) = 0$$

$$\rightarrow \sum_{i=1}^n (y_i x_i - ax_i x_i - \bar{y} x_i + a\bar{x} x_i) = 0$$

$$\rightarrow \sum_{i=1}^n (y_i x_i - \bar{y} x_i) - a \sum_{i=1}^n (x_i x_i - \bar{x} x_i) = 0$$

$$\rightarrow \left(\sum_{i=1}^n x_i y_i - n\bar{x}\bar{y} \right) - a \left(\sum_{i=1}^n x_i x_i - n\bar{x}^2 \right) = 0$$


$$a = \frac{\sum_{i=1}^n x_i y_i - n\bar{x}\bar{y}}{\sum_{i=1}^n x_i x_i - n\bar{x}^2}$$

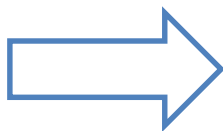
线性回归：一元线性回归

回归模型参数求取： $y_i = ax_i + b$ ($1 \leq i \leq n$) $\min_{a,b} L(a,b) = \sum_{i=1}^n (y_i - a \times x_i - b)^2$

总结

$$b = \bar{y} - a\bar{x}$$

$$a = \frac{\sum_{i=1}^n x_i y_i - n\bar{x}\bar{y}}{\sum_{i=1}^n x_i^2 - n\bar{x}^2}$$



可以看出：只要给出了训练样本 (x_i, y_i) ($i = 1, \dots, n$)，我们就可以从训练样本出发，建立一个线性回归方程，使得对训练样本数据而言，该线性回归方程预测的结果与样本标注结果之间的差值（残差）和最小。

线性回归：一元线性回归

回归模型参数求取： $y_i = ax_i + b$ ($1 \leq i \leq n$) $\min_{a,b} L(a,b) = \sum_{i=1}^n (y_i - a \times x_i - b)^2$

$$b = \bar{y} - a\bar{x}$$

$$a = \frac{\sum_{i=1}^n x_i y_i - n\bar{x}\bar{y}}{\sum_{i=1}^n x_i^2 - n\bar{x}^2}$$

$$a = \frac{x_1 y_1 + x_2 y_2 + \cdots + x_8 y_8 - 8\bar{x}\bar{y}}{x_1^2 + x_2^2 + \cdots + x_8^2 - 8\bar{x}^2} = 1.428$$
$$b = \bar{y} - a\bar{x} = -7.09$$

即**预测**芒提兹尼欧地区火灾所影响森林面积与气温温度之间的一元线性回归模型为“火灾所影响的森林面积 = $1.428 \times$ 气温温度 $- 7.09$ ”，即 $y = 1.428x - 7.09$

线性回归：多元线性回归

多元线性回归模型例子

接下来扩展到数据特征的维度是多维的情况，在上述数据中增加一个影响火灾影响面积的潜在因素—风力。

火灾次数

气温 x	5.1	8.2	11.5	13.9	15.1	16.2	19.6	23.3
风力 z	4.5	5.8	4	6.3	4	7.2	6.3	8.5
火灾影响面积 y	2.14	4.62	8.24	11.24	13.99	16.33	19.23	28.74

多维数据特征中线性回归的问题定义如下：假设总共有 m 个训练数据 $\{(x_i, y_i)\}_{i=1}^m$ ，其中 $x_i = [x_{i,1}, x_{i,2}, \dots, x_{i,D}] \in \mathbb{R}^D$ ， D 为数据特征的维度，线性回归就是要找到一组参数 $a = [a_0, a_1, \dots, a_D]$ ，使得线性函数：

$$f(x_i) = a_0 + \sum_{j=1}^D a_j x_{i,j} = a_0 + \mathbf{a}^T \mathbf{x}_i$$

线性回归：多元线性回归

最小化损失函数：

$$J_m = \frac{1}{m} \sum_{i=1}^m (y_i - f(\mathbf{x}_i))^2$$

为了方便，使用矩阵来表示所有的训练数据和数据标签。

$$X = [\mathbf{x}_1, \dots, \mathbf{x}_m], \quad \mathbf{y} = [y_1, \dots, y_m]$$

其中每一个数据 \mathbf{x}_i 会扩展一个维度，其值为1，对应参数 a_0 。损失函数可以表示为：

$$J_m(\mathbf{a}) = (\mathbf{y} - X^T \mathbf{a})^T (\mathbf{y} - X^T \mathbf{a})$$

损失函数 $J_n(\mathbf{a})$ 对所有参数 \mathbf{a} 求导可得：

$$\nabla J(\mathbf{a}) = -2X(\mathbf{y} - X^T \mathbf{a})$$

因为损失函数 $J_n(\mathbf{a})$ 是一个二次的凸函数，所以函数只存在一个极小值点，也就是最小值点，所以令 $\nabla J(\mathbf{a}) = 0$ 可得

$$\begin{aligned} XX^T \mathbf{a} &= X\mathbf{y} \\ \mathbf{a} &= (XX^T)^{-1} X\mathbf{y} \end{aligned}$$

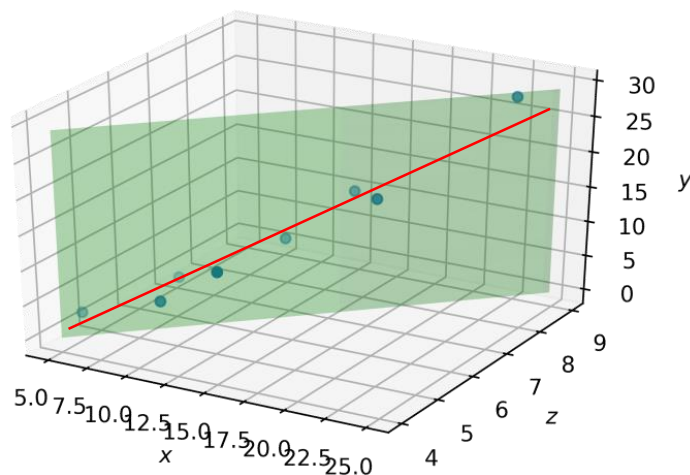
线性回归：多元线性回归

对于上面的例子，转化为矩阵的表示形式为：

$$X = \begin{bmatrix} 5.1 & 8.2 & 11.5 & 13.9 & 15.1 & 16.2 & 19.6 & 23.3 \\ 4.5 & 5.8 & 4. & 6.3 & 4. & 7.2 & 6.3 & 8.5 \\ 1. & 1. & 1. & 1. & 1. & 1. & 1. & 1. \end{bmatrix}$$
$$\mathbf{y} = [2.14 \quad 4.62 \quad 8.24 \quad 11.24 \quad 13.99 \quad 16.33 \quad 19.23 \quad 28.74]^T$$

其中矩阵 X 多出一行全1，是因为常数项 a_0 ，可以看作是数值为全1的特征的对应系数。计算可得

$$\mathbf{a} = [1.312 \quad 0.626 \quad -9.103]$$
$$y = -9.103 + 1.312x + 0.626z$$



线性回归：逻辑斯蒂回归/对数几率回归

逻辑斯蒂回归/对数几率回归模型例子

线性回归一个明显的问题是对离群点（和大多数数据点距离较远的数据点，outlier）非常敏感，导致模型建模不稳定，使结果有偏，为了缓解这个问题（特别是在二分类场景中）带来的影响，可考虑逻辑斯蒂回归(logistic regression)[Cox 1958]，逻辑斯蒂回归又叫逻辑回归、对数几率回归。

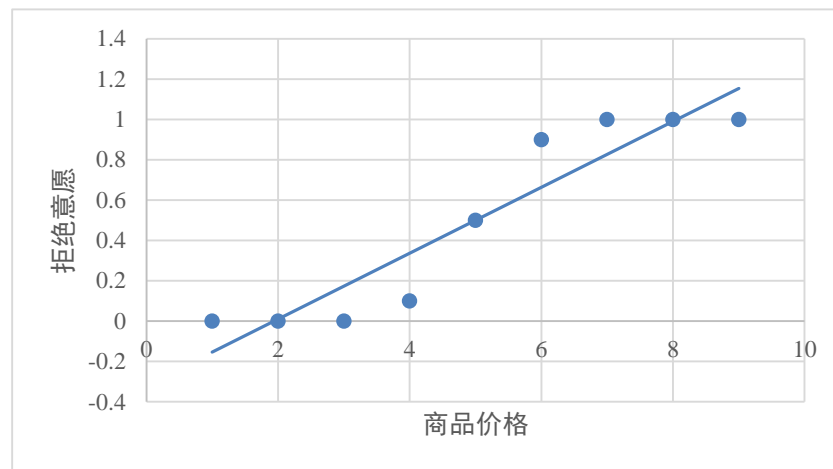


图4.4 用户对某件商品拒绝购买意愿（选择不购买商品的人数/受调查的总人数）与商品价格之间的关系。把0.5作为阈值，拒绝意愿大于0.5即用户拒绝购买。因此本质来讲这是个二分类问题。

线性回归：逻辑斯蒂回归/对数几率回归

逻辑斯蒂回归/对数几率回归模型例子

线性回归一个明显的问题是对离群点（和大多数数据点距离较远的数据点，outlier）非常敏感，导致模型建模不稳定，使结果有偏，为了缓解这个问题（特别是在二分类场景中）带来的影响，可考虑逻辑斯蒂回归(logistic regression)[Cox 1958]。

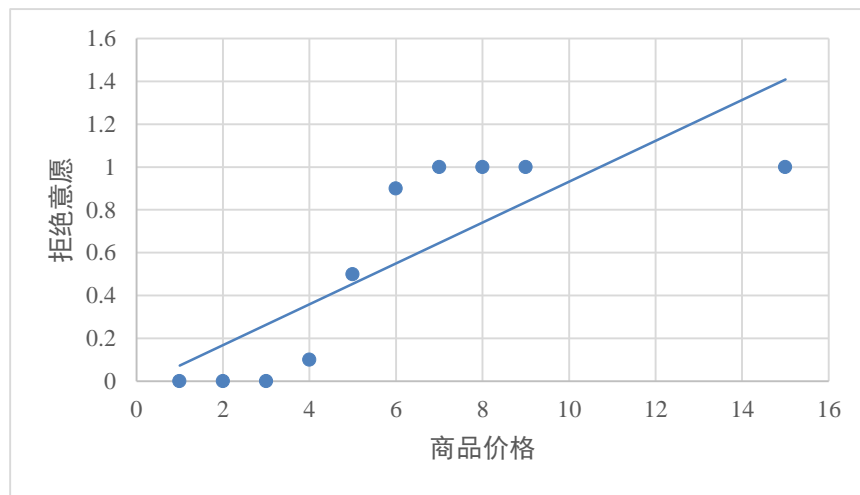


图4.5 加入一个离群点，该点表示当商品价格为15时，用户拒绝意愿为1（即用户不愿意购买该商品）

线性回归：逻辑斯蒂回归/对数几率回归

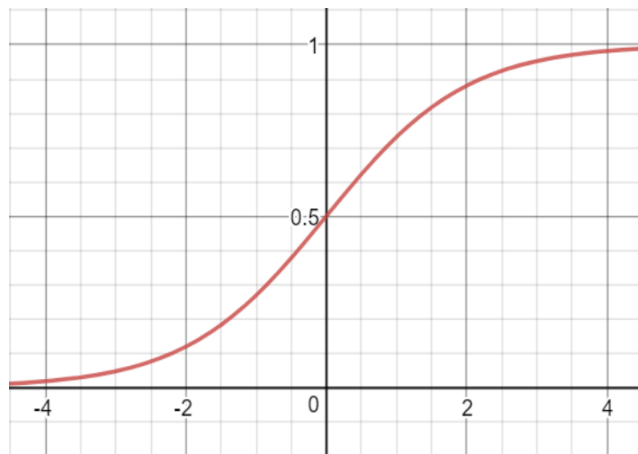


图4.6 Sigmoid函数

逻辑斯蒂回归(logistic regression)就是在回归模型中引入 sigmoid函数的一种非线性回归模型。Logistic回归模型可如下表示：

$$y = \frac{1}{1+e^{-z}} = \frac{1}{1+e^{-(\mathbf{w}^T \mathbf{x} + b)}} \quad , \quad \text{其中 } y \in (0,1), z = \mathbf{w}^T \mathbf{x} + b, \quad \mathbf{w}^T \mathbf{x} \text{ 是 } \mathbf{w} \text{ 和 } \mathbf{x} \text{ 的内积。}$$

这里 $\frac{1}{1+e^{-z}}$ 是 sigmoid 函数、 $\mathbf{x} \in \mathbb{R}^d$ 是输入数据、 $\mathbf{w} \in \mathbb{R}^d$ 和 $b \in \mathbb{R}$ 是回归函数的参数， \mathbf{x} 和 \mathbf{w} 都是 d 维向量。

注意：这里的 y 不等同于后面的标注 y 。这里的 $y = p(y = \mathbf{1} | \mathbf{x}) = h_{\theta}(\mathbf{x})$ 。

线性回归：逻辑斯蒂回归/对数几率回归

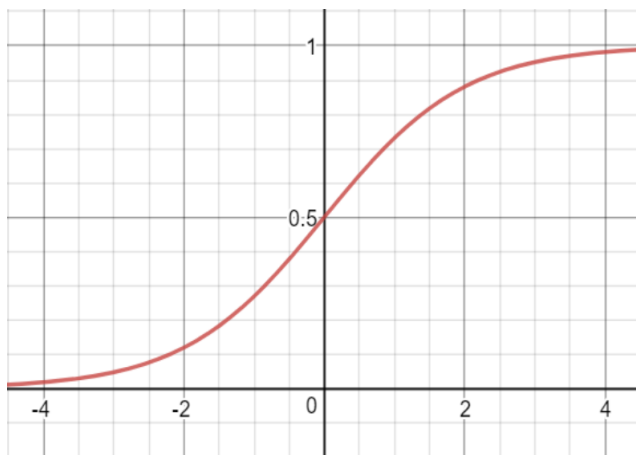


图4.6 Sigmoid函数（像概率分布函数）

Sigmoid函数的特点

- sigmoid函数是单调递增的，其值域为 $(0, 1)$ ，因此使sigmoid函数输出可作为概率值。在前面介绍的线性回归中，回归函数的值域一般为 $(-\infty, +\infty)$
- 对输入 z 取值范围没有限制，但当 z 大于一定数值后，函数输出无限趋近于1，而小于一定数值后，函数输出无限趋近于0。特别地，当 $z = 0$ 时，函数输出为0.5。这里 z 是输入数据 x 和回归函数的参数 w 内积结果（可视为 x 各维度进行加权叠加）
- x 各维度加权叠加之和结果取值在0附近时（ z 在0附近），函数输出值的变化幅度比较大（函数值变化陡峭），且是非线性变化。但是，各维度加权叠加之和结果取值很大或很小时，函数输出值几乎不变化，这是基于概率的一种认识与需要。

线性回归：逻辑斯蒂回归/对数几率回归

逻辑斯蒂回归虽可用于对输入数据和输出结果之间复杂关系进行建模，但由于逻辑斯蒂回归函数的**输出具有概率意义**，使得**逻辑斯蒂回归函数更多用于二分类问题**：

➤ $y = \mathbf{1}$ （用one-hot向量编码： $y = [1, 0]$ ）表示输入数据 \mathbf{x} 属于**正例**；

➤ $y = \mathbf{0}$ （用one-hot向量编码： $y = [0, 1]$ ）表示输入数据 \mathbf{x} 属于**负例**。

$p(y = \mathbf{1}|\mathbf{x}) = \frac{1}{1+e^{-(\mathbf{w}^T \mathbf{x} + b)}}$ 理解为输入数据 \mathbf{x} 为**正例的概率**、 $p(y = \mathbf{0}|\mathbf{x})$ 理解为输入数据 \mathbf{x} 为**负例的概率**。令 $p(y = \mathbf{1}|\mathbf{x})=p$ ，对比值 $\frac{p}{1-p}$ 取对数(即 $\log\left(\frac{p}{1-p}\right)$)来表示输入数据 \mathbf{x} 属于**正例的相对可能性**。 $\frac{p}{1-p}$ 被称为**几率(odds)**，反映了输入数据 \mathbf{x} 作为正例的**相对可能性**。 $\frac{p}{1-p}$ 的**对数几率(log odds)**或**logit函数**可表示为 $\log\left(\frac{p}{1-p}\right)$ 。

显然，令 $p(y = \mathbf{1}|\mathbf{x}) = h_{\theta}(\mathbf{x}) = \frac{1}{1+e^{-(\mathbf{w}^T \mathbf{x} + b)}}$ （ h_{θ} 即 **sigmoid**）和 $p(y = \mathbf{0}|\mathbf{x}) = 1 - h_{\theta}(\mathbf{x}) = \frac{e^{-(\mathbf{w}^T \mathbf{x} + b)}}{1+e^{-(\mathbf{w}^T \mathbf{x} + b)}}$ 。 θ 表示**模型参数**（ $\theta = \{\mathbf{w}, b\}$ ）。于是有：

$$\text{logit}(p) = \text{logit}(p(y = \mathbf{1}|\mathbf{x})) = \log\left(\frac{p(y = \mathbf{1}|\mathbf{x})}{p(y = \mathbf{0}|\mathbf{x})}\right) = \log\left(\frac{p}{1-p}\right) = \mathbf{w}^T \mathbf{x} + b$$

线性回归：逻辑斯蒂回归/对数几率回归

- **分类准则（样本是正例还是负例的判定）**：如果输入数据 \mathbf{x} 属于正例的概率大于其属于负例的概率，即 $p(y = \mathbf{1}|\mathbf{x}) > 0.5$ （**0.5判决阈值**），则输入数据 \mathbf{x} 可被判

断属于正例。这一结果等价于 $\frac{p(y = \mathbf{1}|\mathbf{x})}{p(y = \mathbf{0}|\mathbf{x})} > 1$ ，即 $\log\left(\frac{p(y = \mathbf{1}|\mathbf{x})}{p(y = \mathbf{0}|\mathbf{x})}\right) > \log 1 = 0$ ，

也就是 $\mathbf{w}^T \mathbf{x} + b > 0$ 成立。

- 从这里可以看出，**logistic回归是一个线性模型（把非线性的odds取对数变成logistic）**。在预测时，可以计算线性函数 $\mathbf{w}^T \mathbf{x} + b$ 取值是否大于0来判断输入数据 \mathbf{x} 的类别归属。

有了分类准则，接下来是求参数。

线性回归：似然函数

我们常常用**概率(Probability)**来描述一个事件发生的可能性。而**似然性(Likelihood)**正好反过来，意思是**一个事件实际已经发生了（样本已经获得），反推在什么参数条件下，这个事件发生（出现这些样本）的概率最大**。用数学公式来表达上述意思，就是：

1. 已知参数 β 前提下，预测某事件 x 发生的条件概率为 $p(x|\beta)$ ；
2. 已知某个已发生的事件 x ，未知参数 β 的似然函数为 $\mathcal{L}(\beta|x)$ ；
3. 上面两个值相等，即： $\mathcal{L}(\beta|x)=p(x|\beta)$ 。

一个参数 β 对应一个似然函数的值，当 β 发生变化， $\mathcal{L}(\beta|x)$ 、 $p(x|\beta)$ 也会随之变化。当我们在取得某个参数的时候，似然函数或者条件概率的值到达了最大值，说明在这个参数下最有可能发生 x 事件，即这个参数最合理。

因此，最优 β ，就是使当前观察到的数据样本出现的可能性最大的 β 。

线性回归：基于似然函数的参数优化

模型参数的似然函数被定义为 $\mathcal{L}(\theta|\mathcal{D}) = p(\mathcal{D}|\theta)$ ，其中 $\mathcal{D} = \{(x_i, y_i) | 1 \leq i \leq n\}$ 表示所有观测数据（或训练数据）， θ 表示**模型参数**（ $\theta = \{w, b\}$ ）。在最大化对数似然函数过程中，一般假设观测所得每一个样本数据是独立同分布 (Independent and Identically Distributed, i.i.d)，于是可得：

$$\mathcal{L}(\theta|\mathcal{D}) = p(\mathcal{D}|\theta) = \prod_{i=1}^n p(y_i|x_i, \theta) = \prod_{i=1}^n (h_{\theta}(x_i))^{\alpha_i} (1 - h_{\theta}(x_i))^{1-\alpha_i}$$

对上述公式取对数：

$$\alpha_i = \begin{cases} 1 & \text{if } y_i = \mathbf{1} \\ 0 & \text{if } y_i = \mathbf{0} \end{cases}$$

$$l(\theta) = \log(\mathcal{L}(\theta|\mathcal{D})) = \sum_{i=1}^n \alpha_i \log(h_{\theta}(x_i)) + (1 - \alpha_i) \log(1 - h_{\theta}(x_i))$$

最大似然估计目的是**计算似然函数的最大值**，而分类过程是**需要损失函数最小化**。因此，在上式前加一个负号得到**交叉熵损失函数**：

$$J(\theta) = -l(\theta) = -\log(\mathcal{L}(\theta|\mathcal{D}))$$

$$= -\left(\sum_{i=1}^n \alpha_i \log(h_{\theta}(x_i)) + (1 - \alpha_i) \log(1 - h_{\theta}(x_i)) \right)$$

两个
真实
分布

← 1	0	(1)
0	1	(0)

两者构成预测分布

$$J(\theta) \text{ 等价于: } J(\theta) = \sum_{i=1}^n \begin{cases} -\log(h_{\theta}(x_i)) & \text{if } y_i = \mathbf{1} \\ -\log(1 - h_{\theta}(x_i)) & \text{if } y_i = \mathbf{0} \end{cases}$$

线性回归：逻辑斯蒂回归/对数几率回归

需要用**梯度下降法最小化损失函数**来求解参数。损失函数对参数 θ 的偏导如下（其中， $h'_\theta(x) = h_\theta(x)(1 - h_\theta(x))$, $\log' x = \frac{1}{x}$ ）

$$\begin{aligned}\frac{\partial \mathcal{J}(\theta)}{\partial \theta_j} &= - \sum_{i=1}^n \left(\alpha_i \frac{1}{h_\theta(x_i)} \frac{\partial h_\theta(x_i)}{\partial \theta_j} + (1 - \alpha_i) \frac{1}{1 - h_\theta(x_i)} \frac{\partial (1 - h_\theta(x_i))}{\partial \theta_j} \right) \\&= - \sum_{i=1}^n \frac{\partial h_\theta(x_i)}{\partial \theta_j} \left(\frac{\alpha_i}{h_\theta(x_i)} - \frac{1 - \alpha_i}{1 - h_\theta(x_i)} \right) \\&= - \sum_{i=1}^n x_i h_\theta(x_i) (1 - h_\theta(x_i)) \left(\frac{\alpha_i}{h_\theta(x_i)} - \frac{1 - \alpha_i}{1 - h_\theta(x_i)} \right) \\&= - \sum_{i=1}^n x_i (\alpha_i (1 - h_\theta(x_i)) - (1 - \alpha_i) h_\theta(x_i)) \\&= \sum_{i=1}^n (\alpha_i - h_\theta(x_i)) x_i\end{aligned}$$

最大似然估计

(Maximum Likelihood Estimation, MLE)

将求导结果代入**梯度下降迭代公式**得：

$$\theta_j = \theta_j - \eta \sum_{i=1}^n (\alpha_i - h_\theta(x_i)) x_i$$

梯度

$$\alpha_i = \begin{cases} 1 & \text{if } y_i = \mathbf{1} \\ 0 & \text{if } y_i = \mathbf{0} \end{cases}$$

线性回归：逻辑斯蒂回归/对数几率回归

预测分布：

真实分布

$$h_{\theta}(x) \qquad 1 - h_{\theta}(x) \qquad 1 \qquad 0 \qquad (\mathbf{1})$$

$$h_{\theta}(x) \qquad 1 - h_{\theta}(x) \qquad 0 \qquad 1 \qquad (\mathbf{0})$$

真实分布：用one-hot编码生成概率分布；

预测分布：用sigmoid函数生成概率分布。

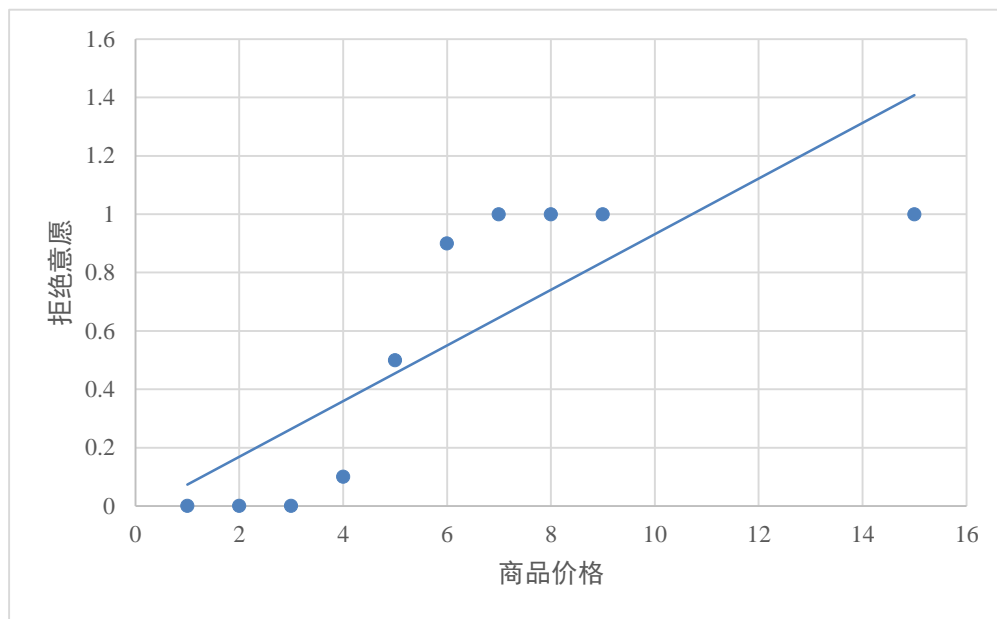
从交叉熵的角度：求取每个样本预测分布和真实分布的内积，以最小化交叉熵损失函数为目标，求取参数 θ ；

从似然函数的角度：用 $h_{\theta}(x)$ 来描述标注 $y_i \in \{\mathbf{0}, \mathbf{1}\}$ 的概率分布，通过构建似然函数，以最大化似然函数为目标，求取参数 θ 。

$$p(y_i = \mathbf{1} | x) = h_{\theta}(x)$$

$$p(y_i = \mathbf{0} | x) = 1 - h_{\theta}(x)$$

线性回归：逻辑斯蒂回归/对数几率回归



在这个例子中：

1. 逻辑回归的本质是将样本 x_i 带入参数优化后的逻辑斯特函数，并构造概率分布。此分布与各分类 y_i 的编码进行交叉熵运算，交叉熵的大小决定了此样本属于哪一类。
2. 如图所示，低价格样本构造的分布函数与不购买这一类的交熵较小，因此这种样本可以被归入不够卖。反之则反。
3. 显然，对于所谓被离群点带偏的点也可如此处理，并实现有效分类。

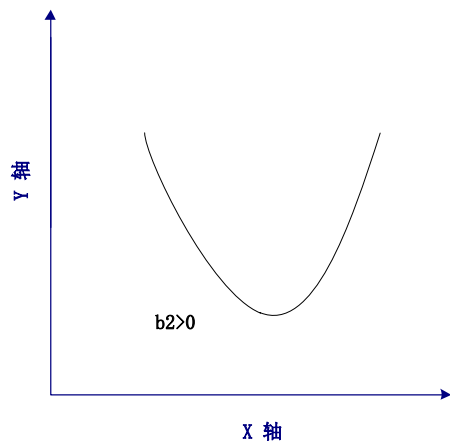
多项式回归：多项式回归模型

研究一个因变量与多个自变量之间的多项式关系称为多项式回归（Polynomial Regression），若自变量的个数为1，则称为一元多项式回归；若自变量的个数大于1，则成为多元多项式回归。

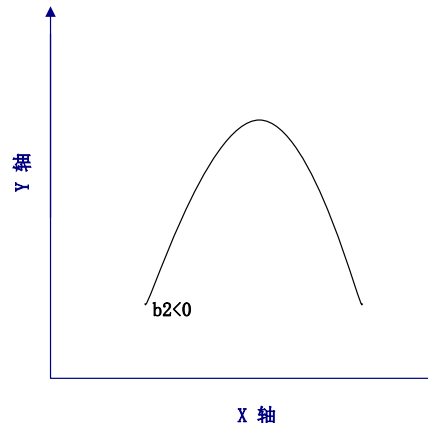
一元 k 次多项式回归方程为：
$$\hat{y} = \hat{a} + \hat{b}_1x + \hat{b}_2x^2 + \dots + \hat{b}_kx^k$$

其中，只有一个自变量 x ， b_1 、 $b_2 \dots b_k$ 为多项式的系数， a 为多项式的截距。

最简单的多项式是二次多项式。其中一元二次多项式方程为：
$$\hat{y} = \hat{a} + \hat{b}_1x + \hat{b}_2x^2$$



$b_2 > 0$ 时多项式形状



$b_2 < 0$ 时多项式形状

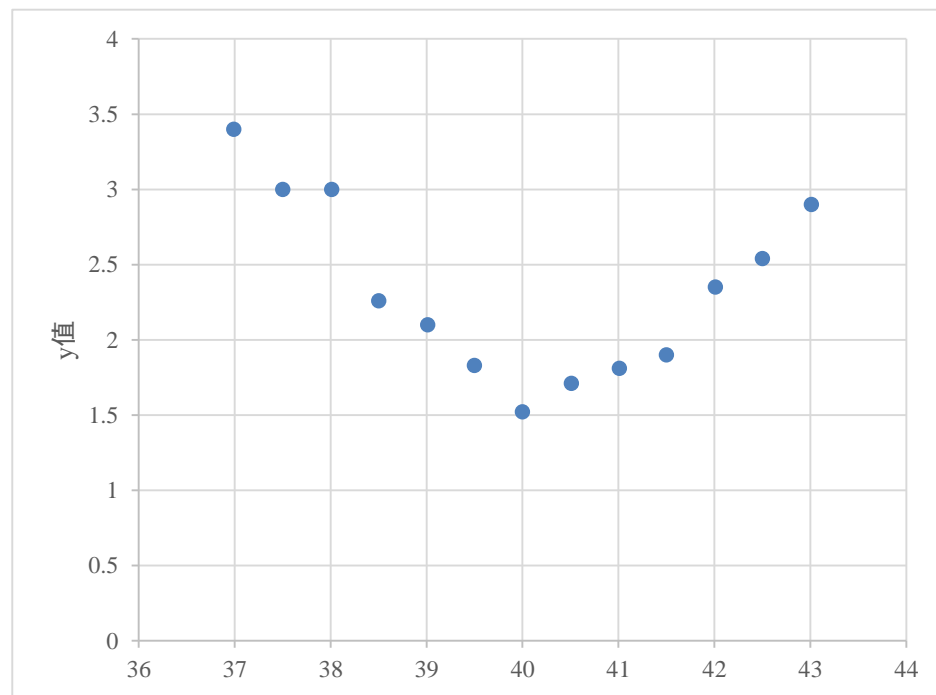
多项式回归：多项式回归实例

例：多项式回归例子

下表是某曲线自变量 x 与因变量 y 的数据集，上图是该曲线中 x 与 y 的散点图，试求出 x 与 y 之间的回归关系。

x	y
42.50	2.54
41.01	1.81
36.99	3.40
37.50	3.00
38.50	2.26
38.01	3.00
39.01	2.10
42.01	2.35
41.50	1.90
39.50	1.83
40.00	1.52
43.01	2.90
40.51	1.71

x 和 y 的数据集



x 和 y 的散点图

多项式回归：多项式回归实例

(1) 多项式回归方程求解

从上图可知， x 和 y 之间的关系可近似用一个一元二次多项式来表示，故假设 x 和 y 之间的关系表达式为：

$$y = a + b_1x + b_2x^2$$

采用**最小二乘法（最小平方方法）**求解多项式回归方程。

残差平方之和分别对 \hat{a} 、 \hat{b}_1 和 \hat{b}_2 求偏导且令偏导值为0。得：

$$\begin{cases} 2\sum_{i=1}^n (y_i - \hat{a} - \hat{b}_1x_i - \hat{b}_2x_i^2) \times (-1) = 0 \\ 2\sum_{i=1}^n (y_i - \hat{a} - \hat{b}_1x_i - \hat{b}_2x_i^2) \times (-x_i) = 0 \\ 2\sum_{i=1}^n (y_i - \hat{a} - \hat{b}_1x_i - \hat{b}_2x_i^2) \times (-x_i^2) = 0 \end{cases}$$

多项式回归：多项式回归实例

多项式回归方程求解（续）

化简后得：

$$\begin{cases} \sum_{i=1}^n y_i - n\hat{a} - \hat{b}_1 \sum_{i=1}^n x_i - \hat{b}_2 \sum_{i=1}^n x_i^2 = 0 \\ \sum_{i=1}^n x_i y_i - \hat{a} \sum_{i=1}^n x_i - \hat{b}_1 \sum_{i=1}^n x_i^2 - \hat{b}_2 \sum_{i=1}^n x_i^3 = 0 \\ \sum_{i=1}^n x_i^2 y_i - \hat{a} \sum_{i=1}^n x_i^2 - \hat{b}_1 \sum_{i=1}^n x_i^3 - \hat{b}_2 \sum_{i=1}^n x_i^4 = 0 \end{cases}$$

带入上表中数据，求解得 $\hat{a} = 269.7711, \hat{b}_1 = -13.294, \hat{b}_2 = 0.16484$

所以求得的多项式回归方程为： $\hat{y} = 269.7711 - 13.294x + 0.16484x^2$

多项式回归：多项式回归实例

(2) 多项式回归方程拟合优度检验

根据TSS和ESS定义求解得：

$$TSS = \sum_{i=1}^n (y_i - \bar{y})^2 = 4.2156$$

$$ESS = \sum_{i=1}^n (\hat{y}_i - \bar{y})^2 = 3.9502$$

决定系数：
$$R^2 = \frac{ESS}{TSS} = \frac{3.9502}{4.2156} = 0.937044$$

R^2 取值范围是[0 1]， R^2 接近于1，说明该回归方程拟合优度较好。拟合优度越大，自变量对因变量的解释程度越高，自变量引起的变动占总变动的百分比高。观察点在回归直线附近越密集。

预测值 \hat{y}

x	y	预测值 \hat{y}
42.5	2.54	2.517353
41.01	1.81	1.814397
36.99	3.4	3.569082
37.5	3	3.051392
38.5	2.26	2.285223
38.01	3	2.619452
39.01	2.1	2.021420
42.01	2.35	2.205408
41.5	1.9	1.964799
39.5	1.83	1.848734
40	1.52	1.754120
43.01	2.9	2.926099
40.51	1.71	1.742523

TSS：总体平方和；ESS：回归平方和；RSS：残差平方和。
$$RSS = \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

TSS=RSS+ESS。

多项式回归：多项式回归实例

(3) 回归方程F检验（联合假设检验，Joint Hypotheses Test）

本例中， $k=2$ ， $n=13$ ，假设 $\alpha = 0.01$ ，经查F值表知： $F_{0.01}(k, n-k-1)=F_{0.01}(2,10)=7.56$

然后求解RSS， $RSS = TSS - ESS = 4.2156 - 3.9502 = 0.2654$

最后求解F值，
$$F = \frac{ESS / k}{RSS / (n - k - 1)} = \frac{3.9502 / 2}{0.2654 / (13 - 2 - 1)} = 74.42$$

求得的F值为 $74.42 > F_{0.01}(2,10) = 7.56$ ，所以在显著性概率为0.01的条件下，回归方程显著成立。

多项式回归：多项式回归实例

(3) 回归方程F检验

本例中， $k=2$ ， $n=13$ ，假设 $\alpha = 0.01$ ，经查F值表知：

$$F_{0.01}(k, n - k - 1) = F_{0.01}(2, 10) = 7.56$$

F 分布临界值表 ($\alpha=0.01$)

		续 表							
$V_1 \backslash V_2$	1	2	3	4	5	6	8	10	15
1	4052	4999	5403	5625	5764	5859	5981	6065	6157
2	98.50	99.00	99.17	99.25	99.30	99.33	99.37	99.40	99.43
3	34.12	30.82	29.46	28.71	28.24	27.91	27.49	27.23	26.87
4	21.20	18.00	16.69	15.98	15.52	15.21	14.80	14.55	14.20
5	16.26	13.27	12.06	11.39	10.97	10.67	10.29	10.05	9.72
6	13.75	10.92	9.78	9.15	8.75	8.47	8.10	7.87	7.56
7	12.25	9.55	8.45	7.85	7.46	7.19	6.84	6.62	6.31
8	11.26	8.65	7.59	7.01	6.63	6.37	6.03	5.81	5.52
9	10.56	8.02	6.99	6.42	6.06	5.80	5.47	5.26	4.96
10	10.04	7.56	6.55	5.99	5.64	5.39	5.06	4.85	4.56
11	9.65	7.21	6.22	5.67	5.32	5.07	4.74	4.54	4.25
12	9.33	6.93	5.95	5.41	5.06	4.82	4.50	4.30	4.01
13	9.07	6.70	5.74	5.21	4.86	4.62	4.30	4.10	3.82

多项式回归：多项式回归实例

(4) 多项式回归方程t检验

此例中， $n = 13$ ，在置信度水平为0.01的情况下，经查 t 分布表，知 t 值为2.681。然后根据式(5-14)和式(5-15)求解得：

$$t_1 = -11.46478$$

$$t_2 = 11.37554$$

t_1 和 t_2 分别是回归方程回归系数 b_1 和 b_2 的 t 检验

$|t_1|$ 和 $|t_2|$ 值均大于 t 分布值2.681，所以，两个自变量均对因变量有显著性影响。