清華大學

第三章
线性学习机器

2022.3.1

多选题 10分

设置

复习：以下陈述正确的是?

D

- A 模式识别既指对样本分类的任务，也指实现分类的一些方法
- B 机器学习是一大类方法的总称，包括模式识别方法
- C 机器学习除了能够分类外，还可以用于其他任务
- D 机器学习是把任务分解成规则、由计算机自动执行规则进行决策
- E 在很多情况下，模式识别与机器学习基本是同义词

提交

线性分类器

$$g(\mathbf{x}) = \sum_{i=1}^{d} w_i x_i + w_0$$

特征

权值

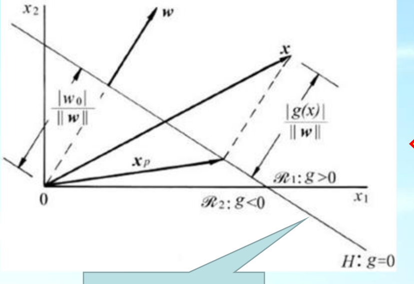
线性判别函数

$$y = \mathrm{sgn}\left(\sum_{i=1}^{d} w_i x_i + w_0\right)$$

决策规则

$$y = \begin{cases} +1 & \Rightarrow \text{class } A \ \ or \ \ \mathbf{x} \in \omega_1 \\ -1 & \Rightarrow \text{class } B \ \ or \ \ \mathbf{x} \in \omega_2 \end{cases}$$

分类标签



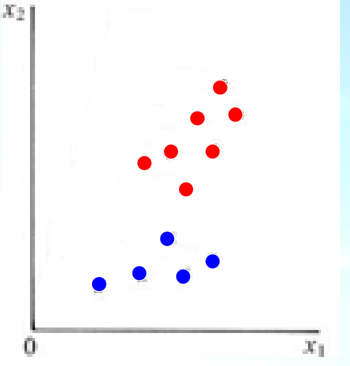
决策面（线）

3.1 Fisher线性判别(FLD）

Fisher’s Linear Discriminant

a.k.a. Fisher’s Linear Discriminant Analysis
(FLD or LDA)

- 假设数据分布如图所示，如何求解线性分类器？



Sir Ronald Aylmer Fisher (R.A. Fisher)

(17 February 1890 – 29 July 1962)

- British statistician and geneticist
  - “a genius who almost single-handedly created the foundations for modern statistical science”
  - “the single most important figure in 20th century statistics”
  - "the greatest of Darwin’s successors".
- Some of the stuff he invited or popularized
  - ANOVA (analysis of variance)
  - Maximum likelihood
  - Fisher’s z-distribution (F distribution)
  - Fisher’s method for data fusion (meta-analysis)
  - The 0.05 cutoff of p-value, the notion of hull hypothesis
  - Fisher’s exact test
  - Fisher’s Discriminant Analysis (in 1936)
  - ……
  - *The Genetical Theory of Natural Selection* (1930)
  - *The Design of Experiments* (1935)

From Wikipedia

Sir Ronald Aylmer Fisher (R.A. Fisher)

(17 February 1890 – 29 July 1962)

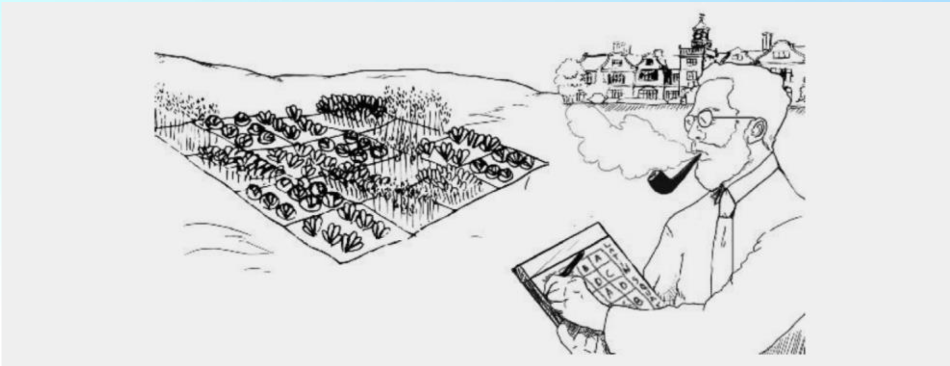


FIGURE 4.3. R. A. Fisher with one of his many innovations: a Latin square experimental design, intended to ensure that one plot of each plant type appears in each row (fertilizer type) and column (soil type). Such designs are still used in practice, but Fisher would later argue convincingly that a randomized design is even more effective. (*Source:* Drawing by Dakota Harr.)

Judea Pearl, *The Book of Why*, 2018

Suggestion: Find stories about Fisher's role in denying the harm of smoking on health, and learn about the topics of causal inference from the story.

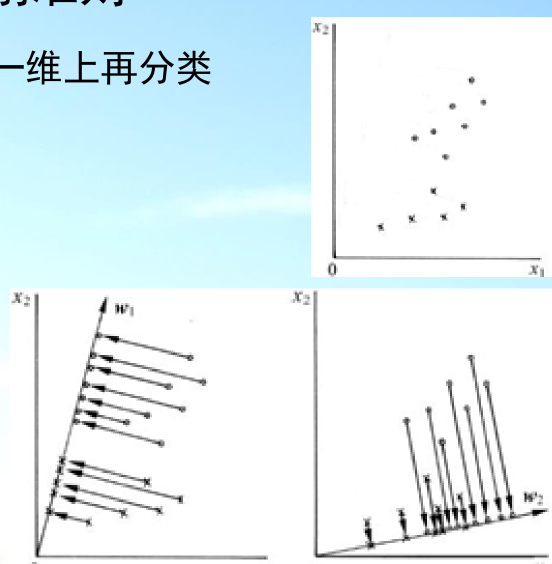
Fisher判别准则

- 求解最佳投影方向，把样本投影到一维上再分类
  - 类内：样本越紧密越好
  - 类间：两类离越远越好

- 样本集：$\mathcal{X} = \{\boldsymbol{x}_1, \cdots, \boldsymbol{x}_N\}$,

  其中，第一类 (ω_1): $\mathcal{X}_1 = \{\boldsymbol{x}_1^1, \cdots, \boldsymbol{x}_{N_1}^1\}$,

  第二类 (ω_2): $\mathcal{X}_2 = \{\boldsymbol{x}_1^2, \cdots, \boldsymbol{x}_{N_2}^2\}$
- 投影： $\mathcal{X} \to \mathcal{Y}$: $y_i = \boldsymbol{w}^T \boldsymbol{x}_i$, $i = 1, \cdots, N$



Fisher判别准则

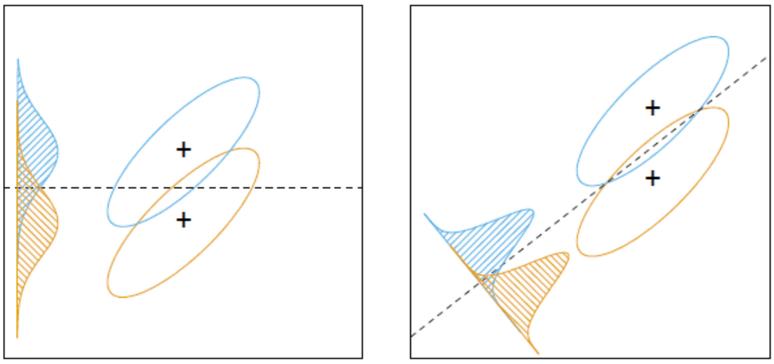


FIGURE 4.9. *Although the line joining the centroids defines the direction of greatest centroid spread, the projected data overlap because of the covariance (left panel). The discriminant direction minimizes this overlap for Gaussian data (right panel).*

T. Hastie, R. Tibshirani, J. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*, 2nd Edition, Springer

考查样本的类内和类间离散度

求解最佳投影方向，把样本投影到一维上再分类
- 类内：样本越紧密越好
- 类间：两类离越远越好

- 在$\mathcal{X}$空间：

类均值向量 $\boldsymbol{m}_i=\frac{1}{N_i}\sum_{x_j\in\mathcal{X}_i}\boldsymbol{x}_j$，$i=1,2$

类内离散度矩阵 within-class scatter matrix

$$\boldsymbol{S}_i=\sum_{x_j\in\mathcal{X}_i}(\boldsymbol{x}_j-\boldsymbol{m}_i)(\boldsymbol{x}_j-\boldsymbol{m}_i)^T,\qquad i=1,2$$

总类内离散度矩阵 $\boldsymbol{S}_w=\boldsymbol{S}_1+\boldsymbol{S}_2$

类间离散度矩阵 between-class scatter matrix

$$\boldsymbol{S}_b=(\boldsymbol{m}_1-\boldsymbol{m}_2)(\boldsymbol{m}_1-\boldsymbol{m}_2)^T$$

考查样本的类内和类间离散度

求解最佳投影方向，把样本投影到一维上再分类
- 类内：样本越紧密越好
- 类间：两类离越远越好

- 在$\mathcal{Y}$空间：

类均值 $\tilde{m}_i = \frac{1}{N_i}\sum_{y_j \in \mathcal{Y}_i} y_j, \quad i = 1,2$

类内离散度

$$\tilde{S}_i = \sum_{y_j \in \mathcal{Y}_i} (y_j - \tilde{m}_i)(y_j - \tilde{m}_i)^T, \qquad i = 1,2$$

总类内离散度 $\tilde{S}_w = \tilde{S}_1 + \tilde{S}_2$

类间离散度矩阵 $\tilde{S}_b = (\tilde{m}_1 - \tilde{m}_2)^2$

考查样本的类内和类间离散度

求解最佳投影方向，把样本投影到一维上再分类
- 类内：样本越紧密越好
- 类间：两类离越远越好

- 在$\mathcal{Y}$空间：

类均值 $\tilde{m}_i = \frac{1}{N_i}\sum_{y_j \in \mathcal{Y}_i} y_j, \quad i = 1,2$

类内离散度

$$\tilde{S}_i = \sum_{y_j \in \mathcal{Y}_i} (y_j - \tilde{m}_i)(y_j - \tilde{m}_i)^T, \qquad i = 1,2$$

总类内离散度 $\tilde{S}_w = \tilde{S}_1 + \tilde{S}_2$

类间离散度矩阵 $\tilde{S}_b = (\tilde{m}_1 - \tilde{m}_2)^2$

- Fisher准则

$$\max_{\boldsymbol{w}} \; J_F(\boldsymbol{w}) = \frac{(\tilde{m}_1 - \tilde{m}_2)^2}{\tilde{S}_1 + \tilde{S}_2}$$

$$y_i = \boldsymbol{w}^T \boldsymbol{x}_i$$

- Fisher准则：

$$\max_{\boldsymbol{w}} \ J_F(\boldsymbol{w}) = \frac{(\tilde{m}_1 - \tilde{m}_2)^2}{\tilde{S}_1 + \tilde{S}_2}$$

求解最佳投影方向，把样本投影到一维上再分类
- 类内：样本越紧密越好
- 类间：两类离越远越好

- 代入$y = \boldsymbol{w}^T\boldsymbol{x}$，求最优投影方向

$$\boldsymbol{w}^* = \underset{\boldsymbol{w}}{\operatorname{argmax}} \ J_F(\boldsymbol{w})$$

- Fisher准则：

$$J_F(\boldsymbol{w}) = \frac{\boldsymbol{w}^T\boldsymbol{S}_b\boldsymbol{w}}{\boldsymbol{w}^T\boldsymbol{S}_w\boldsymbol{w}}$$

思考：最大化$J_F(\boldsymbol{w})$会遇到什么问题？

$$\boldsymbol{w}^* = \underset{\boldsymbol{w}}{\operatorname{argmax}} \ J_F(\boldsymbol{w})$$

$$J_F(\boldsymbol{w}) = \frac{\boldsymbol{w}^T\boldsymbol{S}_b\boldsymbol{w}}{\boldsymbol{w}^T\boldsymbol{S}_w\boldsymbol{w}}$$

求解：

- 问题：改变$\boldsymbol{w}$的幅度, $J_F(\boldsymbol{w})$不会改变 → 无唯一解
- 不妨令分母$\boldsymbol{w}^T\boldsymbol{S}_w\boldsymbol{w} = c \neq 0$，最大化分子$\boldsymbol{w}^T\boldsymbol{S}_b\boldsymbol{w}$, 即：

$$\max \boldsymbol{w}^T\boldsymbol{S}_b\boldsymbol{w}$$

$$s.t. \ \boldsymbol{w}^T\boldsymbol{S}_w\boldsymbol{w} = c$$

--- 带有等式约束的优化问题

怎样求解？

$$\max \boldsymbol{w}^T \boldsymbol{S}_b \boldsymbol{w}$$
$$s.t.\ \boldsymbol{w}^T \boldsymbol{S}_w \boldsymbol{w} = c$$

- 拉格朗日乘子法求最优：
- 定义拉格朗日函数

$$L(\boldsymbol{w}, \lambda) = \boldsymbol{w}^T \boldsymbol{S}_b \boldsymbol{w} - \lambda(\boldsymbol{w}^T \boldsymbol{S}_w \boldsymbol{w} - c)$$

令 $\frac{\partial L}{\partial w} = 0$，可得

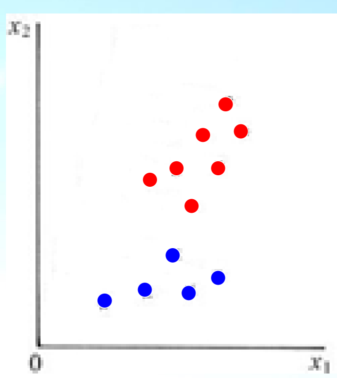
$$\boldsymbol{S}_w^{-1} \boldsymbol{S}_b \boldsymbol{w}^* = \lambda \boldsymbol{w}^*$$

即：w^*为$S_w^{-1}S_b$矩阵的本征向量(eigenvector)

只考虑投影的方向，得

$$\boldsymbol{w}^* \propto \boldsymbol{S}_w^{-1}(\boldsymbol{m}_1 - \boldsymbol{m}_2)$$

- 不忘初心：求分类器的目标实现了吗？



$$\boldsymbol{w}^* \propto \boldsymbol{S}_w^{-1}(\boldsymbol{m}_1 - \boldsymbol{m}_2)$$

- 有了投影方向，还需要确定决策的分界点

$$y = \operatorname{sgn}\Big(\sum_{i=1}^{n} w_i x_i + w_0\Big) = \operatorname{sgn}(\mathbf{w}^T\mathbf{x} + w_0), \qquad y = \begin{cases} +1 & \Rightarrow \mathbf{x} \in \omega_1 \\ -1 & \Rightarrow \mathbf{x} \in \omega_2 \end{cases}$$

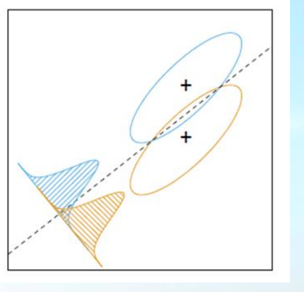
- 如何选择w_0?
  - 根据对数据的不同认识，可以有多种选择方法，比如

$$w_0 = -\frac{1}{2}(\tilde{m}_1 + \tilde{m}_2)$$

$$w_0 = -\tilde{m}$$

$$w_0 = -\frac{1}{2}(\tilde{m}_1 + \tilde{m}_2) + \frac{1}{N_1 + N_2 - 2}\ln\frac{P(\omega_1)}{P(\omega_2)}$$

  - 可以根据对错误率的要求来选择



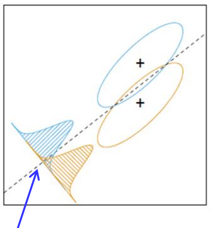
- Commonly used thresholds:

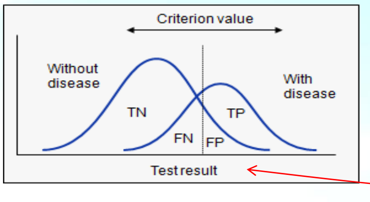
$$w_0 = -\frac{1}{2}(\tilde{m}_1 + \tilde{m}_2)$$

$$w_0 = -\tilde{m}$$

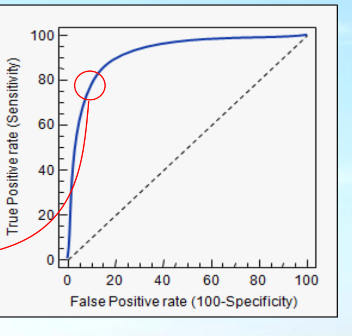
$$w_0 = -\frac{1}{2}(\tilde{m}_1 + \tilde{m}_2) + \frac{1}{N_1 + N_2 - 2}\ln\frac{P(\omega_1)}{P(\omega_2)}$$

- **Choose the threshold with ROC curve**





adjusting the threshold



- 优秀！
- 不过，这算机器学习吗？

D

人 “看”到东西 ➔ 认出东西、产生想法

观察 ➔ 判断

观测 ➔ 分类决策

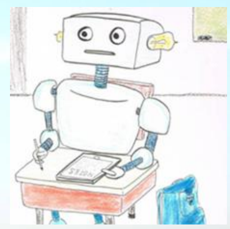
量化观测 ➔ 类别标签

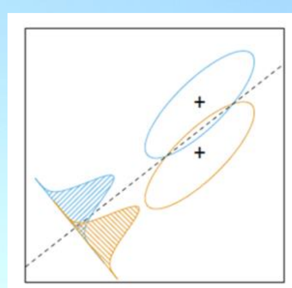
机器 $x \in R^d$ ➔ $y \in \{\omega_1, \omega_2, \cdots\}$

模式识别

如果机器通过实例学会识别，而不是在程序里写好怎样识别，那就是**机器学习**。

基于数据的机器学习





Feature extractor ➔ Classifier Predictor Generator ➔ OUTPUT (decision, action, …)

两种视角看FLD：

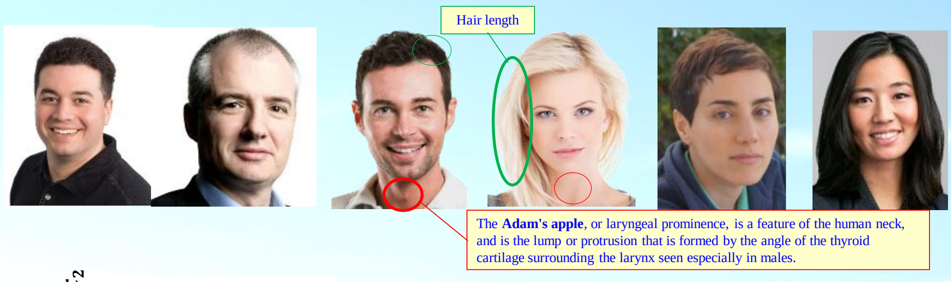
- 解析求解线性分类器
- 提取一维最优特征 + 在一维上用阈值分类器

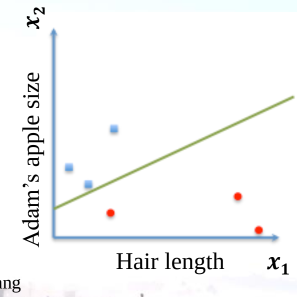
休息10秒钟

3.2 感知器 Perceptron

一个简单的例子

• 如何教小孩从照片识别性别？

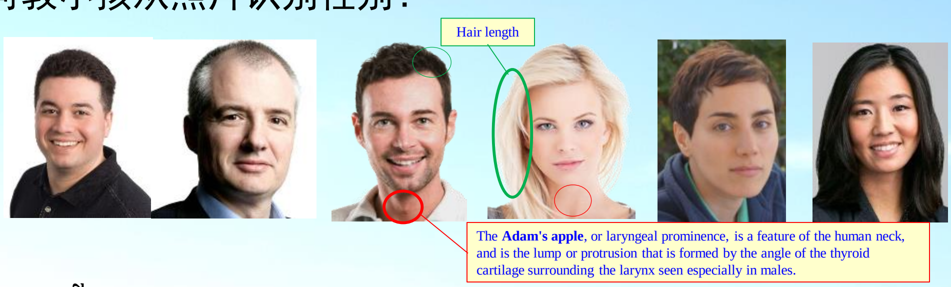


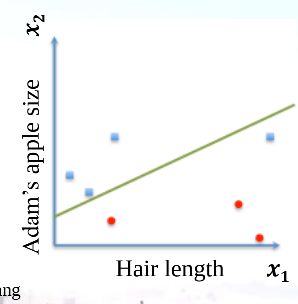


For the Green line:
$$w_1x_1 + w_2x_2 + w_0 = 0$$
For each blue dots:
$$w_1x_1 + w_2x_2 + w_0 > 0$$
For each red dots:
$$w_1x_1 + w_2x_2 + w_0 < 0$$

一个简单的例子

• 如何教小孩从照片识别性别？



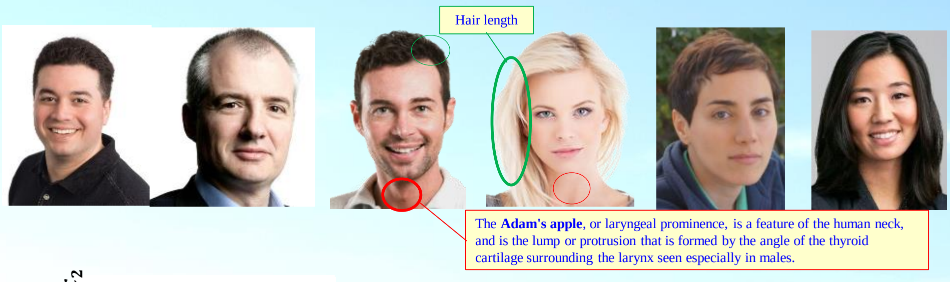


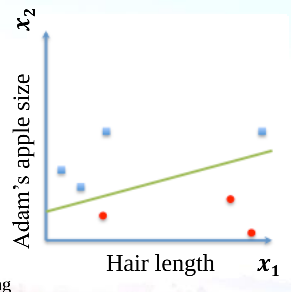
For the Green line:
$$w_1x_1 + w_2x_2 + w_0 = 0$$
For each blue dots:
$$w_1x_1 + w_2x_2 + w_0 > 0$$
For each red dots:
$$w_1x_1 + w_2x_2 + w_0 < 0$$

D

一个简单的例子

- 如何教小孩从照片识别性别？





For the Green line:

$$w_1x_1 + w_2x_2 + w_0 = 0$$

For each blue dots:

$$w_1x_1 + w_2x_2 + w_0 > 0$$

For each red dots:

$$w_1x_1 + w_2x_2 + w_0 < 0$$

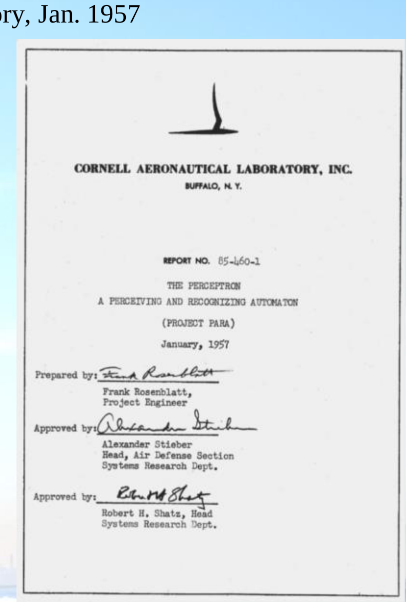
感知器（Perceptron）

Frank Rosenblatt, *The Perceptron – a perceiving and recognizing automaton*, Report 85-460-1, Cornell Aeronautical Laboratory, Jan. 1957

- 感知器：第一台学习机器

$$y = \mathrm{sgn}\left(\sum_{i=1}^{d} w_i x_i + w_0\right)$$

- 为什么把它叫做学习机器？ D


CORNELL AERONAUTICAL LABORATORY, INC.
BUFFALO, N. Y.

REPORT NO. 85-460-1

THE PERCEPTRON
A PERCEIVING AND RECOGNIZING AUTOMATON
(PROJECT PARA)

January, 1957

Prepared by: Frank Rosenblatt
Frank Rosenblatt,
Project Engineer

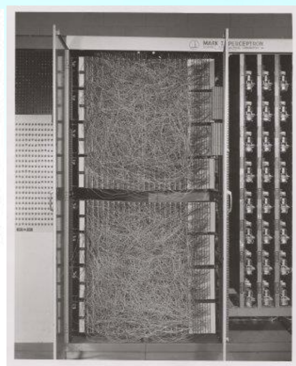
Approved by: Alexander Stieber
Alexander Stieber
Head, Air Defense Section
Systems Research Dept.

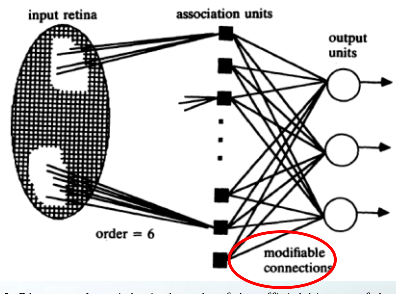
Approved by: Robert M. Shatz
Robert H. Shatz, Head
Systems Research Dept.

感知器

- 为什么把它叫做学习机器？
  ① 因为它是一台机器
  ② 因为它会学习！
  - 它不是编好程序的冯诺依曼计算机，是一台根据训练数据自我调整的学习机器

$$y=\mathrm{sgn}\left(\sum_{i=1}^{n} w_i x_i + w_0\right)$$





M. Olazaran, A sociological study of the official history of the perceptrons controversy, *Social Studies of Science*, 1996

https://en.wikipedia.org/wiki/Perceptron

感知器

- 用数据 $\{(\boldsymbol{x}_1,y_1),\dots,(\boldsymbol{x}_N,y_N)\}$ 训练线性机器

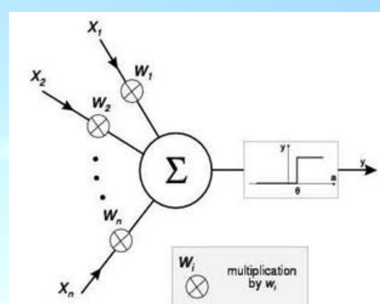
$$y=\mathrm{sgn}\left(\sum_{i=1}^{d} w_i x_i + w_0\right)$$

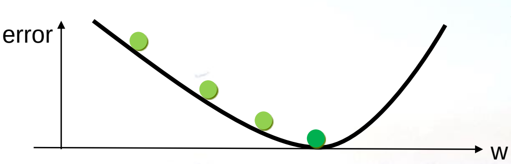
  - 目标：最优化目标函数 $J(w)$

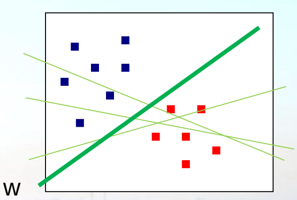
    $J(\boldsymbol{w})$ = 训练错误数

  - 学习算法
    - 梯度下降法

    do $\mathbf{w}(t+1) = \mathbf{w}(t) - \eta\nabla J$
    until $(\nabla J < threshold)$







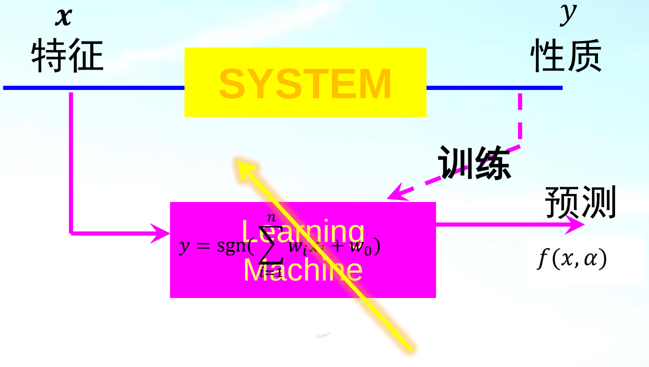
机器学习的基本要素

- 怎样造一个学习机器？
  - 它需要老师
    - → 我们设计它（特征和模型）
  - 它需要训练/学习材料
    - → 训练数据
  - 我们需要为它树立学习的目标
    - → 目标函数、学习准则
  - 我们需要告诉它怎样学
    - → 学习/训练算法

Xuegong Zhang

监督学习 Supervised Learning

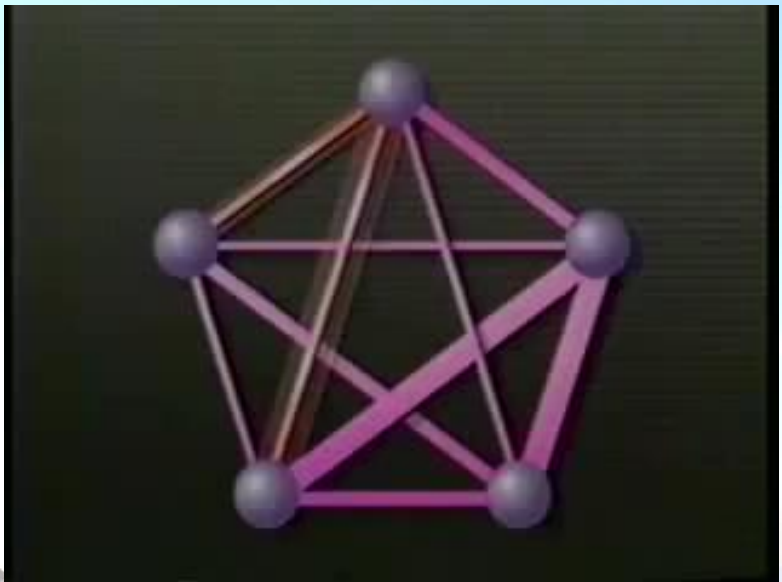
- 用已知答案的数据去训练 ➔ 监督学习



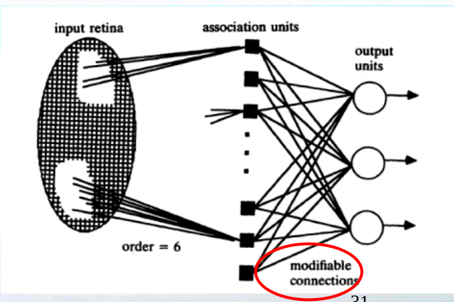
若y是离散类
就是模式识别
→监督模式识别

比如：
- x 身高体重→ y性别
- x 图像特征→ y性别
- x 图像像素→ y图像物体
- x 司机视觉→ y方向盘角度
- x 音频信号→ y语音内容
- x 基因表达→ y疾病类型
- x 肺CT影像→ y新冠肺炎
- x 近期疫情→ y未来走向
- …

感知器是怎么学习的？



$$y = \mathrm{sgn}\left(\sum_{i=1}^{n} w_i x_i + w_0\right)$$



M. Olazaran, A sociological study of the official history of the perceptrons controversy, *Social Studies of Science*, 1996

感知器的求解

- 线性判别函数的齐次简化

$$g(\boldsymbol{x}) = \boldsymbol{w}^{\mathrm{T}}\boldsymbol{x} + w_0 = \boldsymbol{\alpha}^{\mathrm{T}}\mathbf{y}$$

$\boldsymbol{y} = \begin{bmatrix} 1 \\ \mathbf{x} \end{bmatrix}$ 增广的特征向量

$\boldsymbol{\alpha} = \begin{bmatrix} w_0 \\ \boldsymbol{w} \end{bmatrix}$ 增广的权向量

为进一步简化推导，把样本向量再进行如下规范化：

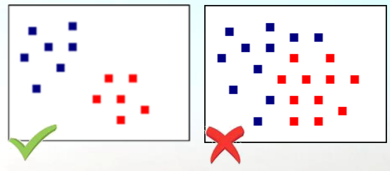
$$\boldsymbol{y}_i' = \begin{cases} \boldsymbol{y}_i, & \text{if } \boldsymbol{y}_i \in \omega_1, \\ -\boldsymbol{y}_i, & \text{if } \boldsymbol{y}_i \in \omega_2, \end{cases} \quad i = 1, \cdots, N$$

$\boldsymbol{y}_i'$：规范化增广样本向量，仍记作$\boldsymbol{y}_i$

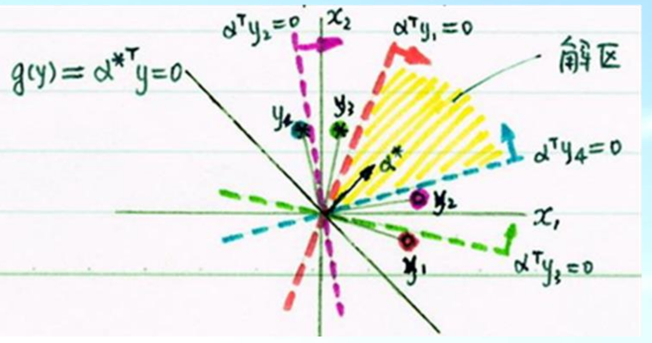
于是，对正确分类样本i：$\boldsymbol{\alpha}^{\mathrm{T}}\boldsymbol{y}_i > 0$

- 线性可分性：

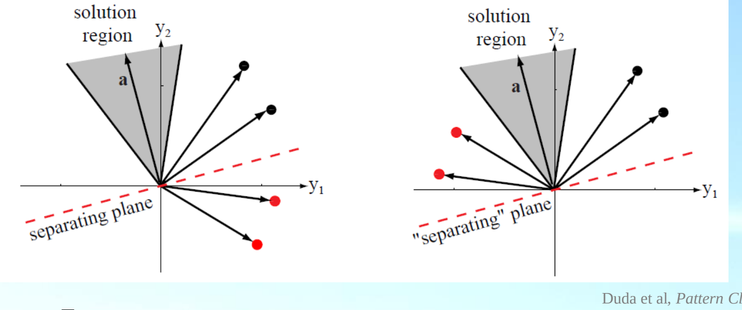
  $\exists \boldsymbol{\alpha}$， $\boldsymbol{\alpha}^T\boldsymbol{y}_i > 0, i = 1, \cdots, N$



- 解向量$\boldsymbol{\alpha}^*$：满足$\boldsymbol{\alpha}^T\boldsymbol{y}_i > 0, i = 1, \cdots, N$
- 解区：权值空间中所有解向量组成的区域

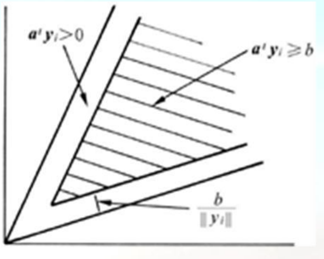


  - 对每个样本$\boldsymbol{y}_i$，$\boldsymbol{\alpha}^T\boldsymbol{y}_i = 0$为权空间中的一个超平面，解区只可能在超平面的正侧
  - 所有样本对应的超平面的正侧的交集就是解区



Duda et al, *Pattern Classification*

- 引入余量$\boldsymbol{\alpha}^T\boldsymbol{y}_i \geq b > 0$



感知器准则函数

$$J_P(\boldsymbol{\alpha}) = \sum_{\mathbf{y}_j \in \mathcal{Y}^k} (-\boldsymbol{\alpha}^T \mathbf{y}_j)$$

$\mathcal{Y}^k$: 在第k步被$\boldsymbol{\alpha}$ 错分的样本集合

- 感知器算法（Rosenblatt,1957）：

$$J_P(\boldsymbol{\alpha}^*) = \min J_P(\boldsymbol{\alpha}) = 0$$

Xuegong Zhang 35

感知器准则函数

$$J_P(\boldsymbol{\alpha}) = \sum_{\mathbf{y}_j \in \mathcal{Y}^k} (-\boldsymbol{\alpha}^T \mathbf{y}_j)$$

$\mathcal{Y}^k$: 在第k步被$\boldsymbol{\alpha}$ 错分的样本集合

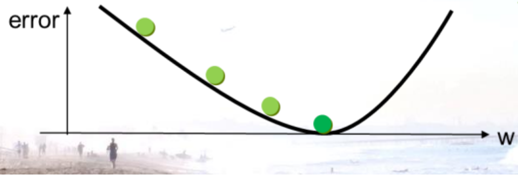
- 感知器算法（Rosenblatt,1957）：

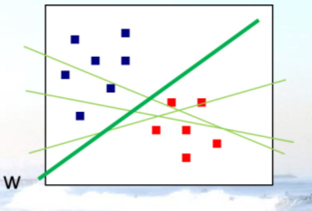
$$J_P(\boldsymbol{\alpha}^*) = \min J_P(\boldsymbol{\alpha}) = 0$$

- 用梯度下降法(Gradient descent)迭代求解

$$\boldsymbol{\alpha}(k+1) = \boldsymbol{\alpha}(k) - \rho_k \nabla J$$

$$\nabla J = \frac{\partial J_P(\boldsymbol{\alpha})}{\partial \boldsymbol{\alpha}} = \sum_{\mathbf{y}_j \in \mathrm{Y}^k} (-\mathbf{y}_j), \qquad \therefore \ \boldsymbol{\alpha}(k+1) = \boldsymbol{\alpha}(k) + \rho_k \sum_{\mathbf{y}_j \in \mathrm{Y}^k} \mathbf{y}_j$$





Xuegong Zhang 36

感知器学习算法

清华大学

单样本修正：

$$\boldsymbol{\alpha}(k+1) = \boldsymbol{\alpha}(k) + \rho_k \sum_{\mathbf{y}_j \in Y^k} \boldsymbol{y}_j$$

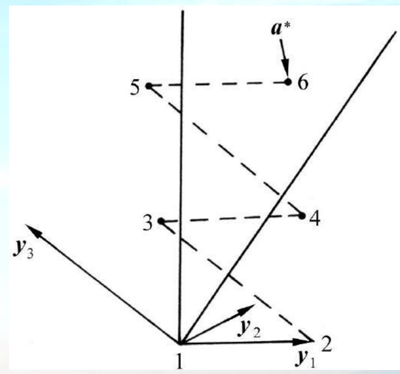
- 固定增量法：
  ① 初值任意
  ② 对样本$\boldsymbol{y}_j$，若$\boldsymbol{\alpha}(k)^T\boldsymbol{y}_j \le 0$（或$b$），则$\boldsymbol{\alpha}(k+1) = \boldsymbol{\alpha}(k) + \boldsymbol{y}_j$
  ③ 对所有样本重复(2)，直到$J_P = 0$

收敛性：

- 对线性可分样本集，
  经过有限次修正后一定可以找到一个解

- 变增量法，如绝对修正法

$$\rho_k = \frac{|\alpha(k)^T \boldsymbol{y}_j|}{\|\boldsymbol{y}_j\|^2}$$

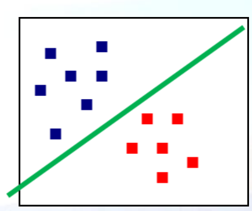


讨论

清华大学

- 感知器有什么问题？

$$\boldsymbol{\alpha}(k+1) = \boldsymbol{\alpha}(k) + \rho_k \sum_{\mathbf{y}_j \in Y^k} \boldsymbol{y}_j$$

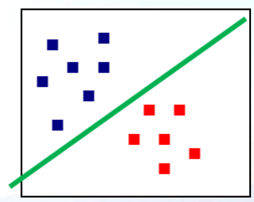


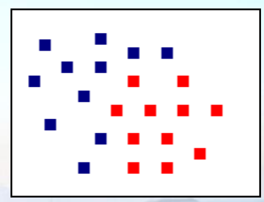
讨论

D

- 感知器有什么问题？
  - 样本线性不可分呢？
  - 线性可分时多解？
  - 多类呢？
- 有问题怎么办？
  - 容忍错误，使错误尽量小
    - 比如强制收敛、MSE
  - 寻求非线性方法
    - 比如神经网络、SVM
  - 寻求“最优分类器”
    - 比如支持向量机SVM
  - 多类分类方法
  - 用两类分类器完成多类分类

$$\boldsymbol{\alpha}(k+1)=\boldsymbol{\alpha}(k)+\rho_k\sum_{\mathbf{y}_j\in\mathrm{Y}^k}\boldsymbol{y}_j$$





单选题 1分 设置

休息4分钟，回到座位后请答题

A 已回座位

B 还没有

提交

3.3 线性回归
Linear Regression

线性分类器

$$\mathbf{y} = f(\mathbf{x})$$

预测 ↑ 分类器 ↑ 样本 ↑

$$\mathcal{X} = [\mathbf{x}_1, \cdots, \mathbf{x}_m] = \begin{bmatrix} x_{11} & \cdots & x_{m1} \\ \vdots & \ddots & \vdots \\ x_{1n} & \cdots & x_{mn} \end{bmatrix}$$

← 特征

样本 ↑

$$y = \mathrm{sgn}\left(\sum_{i=1}^{n} w_i x_i + w_0\right) = \mathrm{sgn}(\mathbf{w}^T\mathbf{x} + w_0), \qquad y = \begin{cases} +1 & \Rightarrow \mathbf{x} \in \omega_1 \\ -1 & \Rightarrow \mathbf{x} \in \omega_2 \end{cases}$$

线性回归

$$\mathbf{y} = f(\mathbf{x})$$

预测 回归 样本

$$\boldsymbol{X} = [\mathbf{x}_1, \cdots, \mathbf{x}_m] = \begin{bmatrix} x_{11} & \cdots & x_{m1} \\ \vdots & \ddots & \vdots \\ x_{1n} & \cdots & x_{mn} \end{bmatrix}$$

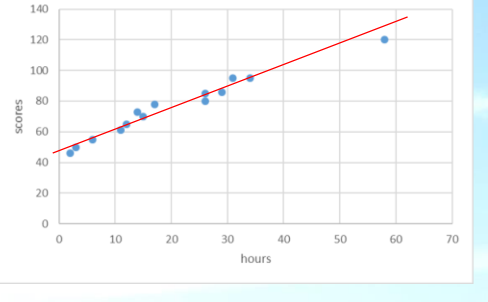
特征

样本

$$y = \sum_{i=1}^{n} w_i x_i + w_0 = \mathbf{w}^T \mathbf{x} + w_0$$

简单线性回归

Student id	Final score	Study Hours per Week
1	50	3
2	95	34
3	78	17
4	55	6
5	65	12
6	70	15
7	80	26
8	86	29
9	73	14
10	120	58
11	46	2
12	95	31
13	85	26
14	61	11



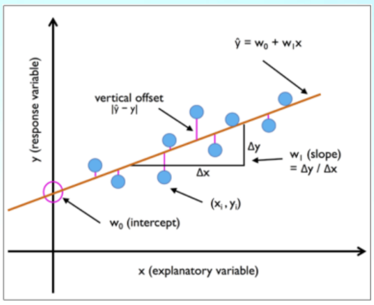
Simple Linear Regression

$$y = w_0 + w_1 x$$

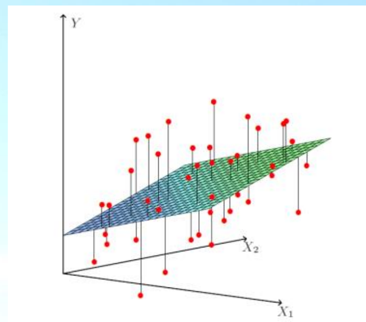
多元线性回归

Simple Linear Regression

$$y = w_0 + w_1 x$$



Raschka & Mirjalili, *Python Machine Learning*

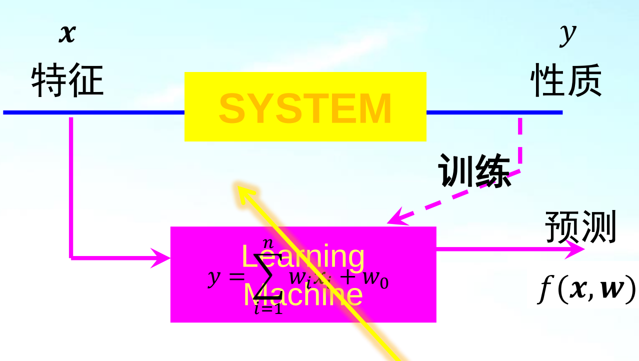


Hastie, Tibshirani, Friedman, *Elements of Statistical Learning*

Multiple Linear Regression

$$y = w_0 + w_1 x + \cdots + w_d x_d = \sum_{i=0}^{d} w_i x_i = \boldsymbol{w}^T \boldsymbol{x}$$

线性回归版的机器学习



$$y = \sum_{i=1}^{n} w_i x_i + w_0 = \boldsymbol{w}^{\mathrm{T}} \boldsymbol{x} + w_0$$

机器学习的基本要素

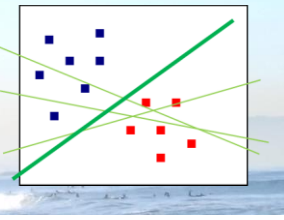
- 怎样造一个学习机器？
  - 它需要老师
    - → 我们设计它（特征和模型）
  - 它需要训练/学习材料
    - → 训练数据
  - 我们需要为它树立学习的目标
    - → 目标函数、学习准则
  - 我们需要告诉它怎样学
    - → 学习/训练算法

Xuegong Zhang

机器学习的基本要素：感知器版

- 怎样造一个学习机器？
  - 它需要老师
    - → 我们设计它（特征和模型）：____________________
  - 它需要训练/学习材料
    - → 训练数据：____________________
  - 我们需要为它树立学习的目标
    - → 目标函数、学习准则：____________________
  - 我们需要告诉它怎样学
    - → 学习/训练算法：____________________

D



Xuegong Zhang

机器学习的基本要素：感知器版

- 怎样造一个学习机器？
  - 它需要老师
    - → 我们设计它（特征和模型） $y = \mathrm{sgn}(\sum_{i=1}^{d} w_i x_i + w_0)$
  - 它需要训练/学习材料
    - → 训练数据 $\{(\boldsymbol{x}_1, y_1), \dots, (\boldsymbol{x}_N, y_N)\}$, $\boldsymbol{x}_j \in R^{d+1}, y_j \in \{-1,1\}$
  - 我们需要为它树立学习的目标
    - → 目标函数、学习准则 $\min J_P(\boldsymbol{\alpha}) = \sum_{\mathbf{y}_j \in \mathcal{Y}^k} (-\boldsymbol{\alpha}^{\mathrm{T}} \boldsymbol{y}_j)$
  - 我们需要告诉它怎样学
    - → 学习/训练算法 $\boldsymbol{\alpha}(k+1) = \boldsymbol{\alpha}(k) - \rho_k \nabla J = \boldsymbol{\alpha}(k) + \rho_k \sum_{\boldsymbol{y}_j \in \mathrm{Y}^k} \boldsymbol{y}_j$

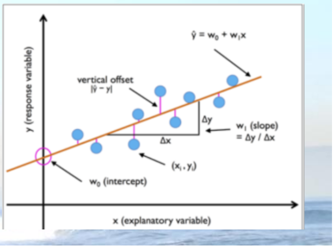
D

Xuegong Zhang 49

机器学习的基本要素：线性回归版

- 怎样造一个学习机器？
  - 它需要老师
    - → 我们设计它（特征和模型） $f(\boldsymbol{x}) = \sum_{i=0}^{d} w_i x_i = \boldsymbol{w}^T \boldsymbol{x}$
  - 它需要训练/学习材料
    - → 训练数据 $\{(\boldsymbol{x}_1, y_1), \dots, (\boldsymbol{x}_N, y_N)\}$, $\boldsymbol{x}_j \in R^{d+1}, y_j \in R$
  - 我们需要为它树立学习的目标
    - → 目标函数、学习准则 ____________________
  - 我们需要告诉它怎样学
    - → 学习/训练算法 ？

D



Xuegong Zhang

机器学习的基本要素：线性回归版

- 怎样造一个学习机器？
  - 它需要老师

    → 我们设计它（特征和模型） $f(\boldsymbol{x}) = \sum_{i=0}^{d} w_i x_i = \boldsymbol{w}^T\boldsymbol{x}$
  - 它需要训练/学习材料

    → 训练数据 $\{(\boldsymbol{x}_1, y_1), \dots, (\boldsymbol{x}_N, y_N)\},\ \boldsymbol{x}_j \in R^{d+1}, y_j \in R$
  - 我们需要为它树立学习的目标

    → 目标函数、学习准则 $\min E = \frac{1}{N}\sum_{j=1}^{N}\left(f(\boldsymbol{x}_j) - y_j\right)^2$
  - 我们需要告诉它怎样学

    → 学习/训练算法 ？

D

线性回归算法

$$\min_{\boldsymbol{w}} \ E(\boldsymbol{w}) = \frac{1}{N}\sum_{j=1}^{N}(\hat{y}_j - y_j)^2 = \frac{1}{N}\sum_{j=1}^{N}\left(f(\boldsymbol{x}_j) - y_j\right)^2 = \frac{1}{N}\|\boldsymbol{X}\boldsymbol{w} - \boldsymbol{y}\|^2 = \frac{1}{N}(\boldsymbol{X}\boldsymbol{w} - \boldsymbol{y})^{\mathrm{T}}(\boldsymbol{X}\boldsymbol{w} - \boldsymbol{y})$$

其中 $\boldsymbol{X} = \begin{bmatrix} \boldsymbol{x}_1^{\mathrm{T}} \\ \vdots \\ \boldsymbol{x}_N^{\mathrm{T}} \end{bmatrix}$， $\mathbf{y} = \begin{bmatrix} y_1 \\ \vdots \\ y_N \end{bmatrix}$.

解：

令 $\nabla E(\boldsymbol{w}) = \frac{\partial E(\boldsymbol{w})}{\partial \boldsymbol{w}} = \frac{2}{N}\boldsymbol{X}^{\mathrm{T}}(\boldsymbol{X}\boldsymbol{w} - \boldsymbol{y}) = 0$,

有 $\boldsymbol{X}^{\mathrm{T}}\boldsymbol{X}\boldsymbol{w} = \boldsymbol{X}^{\mathrm{T}}\boldsymbol{y}$.

若$(\boldsymbol{X}^{\mathrm{T}}\boldsymbol{X})$可逆，则 $\boldsymbol{w}^* = \left(\boldsymbol{X}^{\mathrm{T}}\boldsymbol{X}\right)^{-1}\boldsymbol{X}^{\mathrm{T}}\boldsymbol{y}$.

其中，$\boldsymbol{X}^{+} = \left(\boldsymbol{X}^{\mathrm{T}}\boldsymbol{X}\right)^{-1}\boldsymbol{X}^{\mathrm{T}}$也称作伪逆。

$$\left[\left[\quad\right]\left[\quad\right]\right]^{-1}\left[\quad\right]$$

dim: $(d+1) \times N$ $\quad N \times (d+1)$ $\quad (d+1) \times N$

Linear regression algorithm:

1: Construct the matrix X and the vector $\mathbf{y}$ from the data set $(\mathbf{x}_1, y_1), \cdots, (\mathbf{x}_N, y_N)$, where each $\mathbf{x}$ includes the $x_0 = 1$ bias coordinate, as follows

$$\underbrace{\mathrm{X} = \begin{bmatrix} -\mathbf{x}_1^{\mathrm{T}}- \\ -\mathbf{x}_2^{\mathrm{T}}- \\ \vdots \\ -\mathbf{x}_N^{\mathrm{T}}- \end{bmatrix}}_{\text{input data matrix}}, \quad \underbrace{\mathbf{y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{bmatrix}}_{\text{target vector}}.$$

2: Compute the pseudo-inverse $\mathrm{X}^\dagger$ of the matrix X. If $\mathrm{X}^{\mathrm{T}}\mathrm{X}$ is invertible,

$$\mathrm{X}^\dagger = (\mathrm{X}^{\mathrm{T}}\mathrm{X})^{-1}\mathrm{X}^{\mathrm{T}}.$$

3: Return $\mathbf{w}_{\mathrm{lin}} = \mathrm{X}^\dagger \mathbf{y}$.

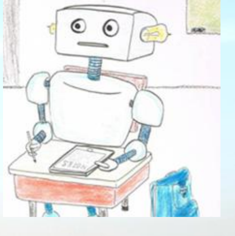
Abu-Mostafa, Magdon-Ismail, Lin, *Learning from Data*

---- Ordinary least squares (OLS) algorithm **最小二乘法**

- 优秀！
- 不过，这算机器学习吗？

D

- 这解析解也算是“机器学习”？
  - OLS由Legendre在1805和Gauss在1809发明，远早于机器学习概念的诞生
  - 如FLD一样，算法从数据中算出“预测规律”
    ➔ 也算是“机器学习”
  - 线性回归也可迭代求解，如感知器那样迭代“学习”



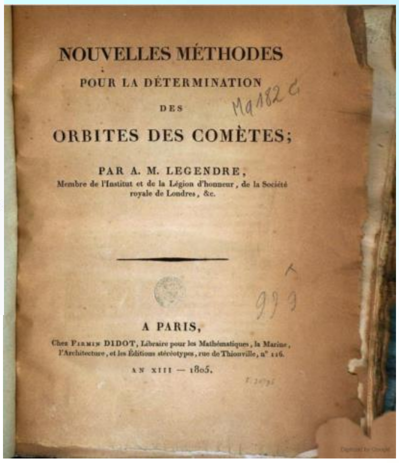
A.M. Legendre. Nouvelles m éhodes pour la d étermination des orbites des comètes, Firmin Didot, Paris, 1805. “Sur la Méthode des moindres quarrés” appears as an appendix.

C.F. Gauss. Theoria Motus Corporum Coelestium in Sectionibus Conicis Solem Ambientum. (1809)

Adrien-Marie Legendre

1820 watercolor caricature of Adrien-Marie Legendre by French artist Julien-Leopold Boilly (see portrait debacle), the only existing portrait known[1]

Born	18 September 1752 Paris, France
Died	10 January 1833 (aged 80) Paris, France

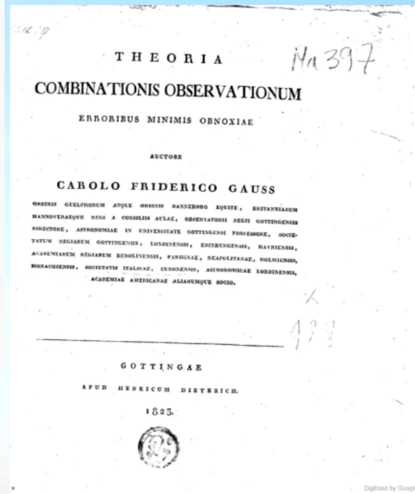


NOUVELLES MÉTHODES
POUR LA DÉTERMINATION
DES
ORBITES DES COMÈTES;
PAR A. M. LEGENDRE,
Membre de l'Institut et de la Légion d'honneur, de la Société royale de Londres, &c.
A PARIS,
Chez FIRMIN DIDOT, Libraire pour les Mathématiques, la Marine, l'Architecture, et les Éditions stéréotypes, rue de Thionville, n° 116.
AN XIII — 1805.

Carl Friedrich Gauss

Carl Friedrich Gauß (1777–1855), painted by Christian Albrecht Jensen

Born	Johann Carl Friedrich Gauss 30 April 1777 Brunswick, Principality of Brunswick-Wolfenbüttel
Died	23 February 1855 (aged 77) Göttingen, Kingdom of Hanover, German Confederation



THEORIA
COMBINATIONIS OBSERVATIONUM
ERRORIBUS MINIMIS OBNOXIAE
AUCTORE
CAROLO FRIDERICO GAUSS
GOTTINGAE
APUD HENRICUM DIETERICH.
1823.

\- Wikipedia

算法本身有没有问题？

Linear regression algorithm:

1: Construct the matrix X and the vector $\mathbf{y}$ from the data set $(\mathbf{x}_1, y_1), \cdots, (\mathbf{x}_N, y_N)$, where each $\mathbf{x}$ includes the $x_0 = 1$ bias coordinate, as follows

$$\mathrm{X} = \underbrace{\begin{bmatrix} -\mathbf{x}_1^{\mathrm{T}}- \\ -\mathbf{x}_2^{\mathrm{T}}- \\ \vdots \\ -\mathbf{x}_N^{\mathrm{T}}- \end{bmatrix}}_{\text{input data matrix}}, \qquad \mathbf{y} = \underbrace{\begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{bmatrix}}_{\text{target vector}}.$$

2: Compute the pseudo-inverse $\mathrm{X}^\dagger$ of the matrix X. If $\mathrm{X}^{\mathrm{T}}\mathrm{X}$ is invertible,

$$\mathrm{X}^\dagger = (\mathrm{X}^{\mathrm{T}}\mathrm{X})^{-1}\mathrm{X}^{\mathrm{T}}.$$

3: Return $\mathbf{w}_{\text{lin}} = \mathrm{X}^\dagger \mathbf{y}$.

D

算法本身有没有问题？

- 若$(\boldsymbol{X}^{\mathrm{T}}\boldsymbol{X})$可逆，则 $\boldsymbol{w}^* = (\boldsymbol{X}^{\mathrm{T}}\boldsymbol{X})^{-1}\boldsymbol{X}^{\mathrm{T}}\boldsymbol{y}$ ； 如果不可逆呢？
- 可逆（非奇异，非退化，满秩）

$$\boldsymbol{X} = \begin{bmatrix} \boldsymbol{x}_1^{\mathrm{T}} \\ \vdots \\ \boldsymbol{x}_N^{\mathrm{T}} \end{bmatrix} = \Big[\qquad \Big]_{N\times(d+1)}$$

- $(\boldsymbol{X}^{\mathrm{T}}\boldsymbol{X})$可逆：$\boldsymbol{X}$ 列满秩：特征间线性独立
  - 当$N \gg d + 1$时通常成立
- 当特征不是线性独立时，仍然可以计算伪逆，但解不唯一
- 解决方案：
  - 通过特征选择或变换去除冗余
  - 通过引入其他准则对解加以限制(如SVD或正则化）

如何评价回归效果？

Evaluation of regression models

$\boldsymbol{R^2}$: the ***goodness-of-fit***, the ***coefficient of determination***, …

R平方/拟合度/决定系数

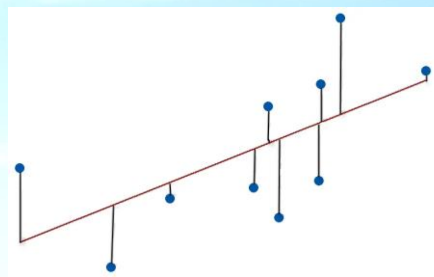
$$R^2 = 1 - \frac{\sum_{i=1}^{N}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{N}(y_i - \bar{y})^2}$$

Unexplained variation

Total variation

For OLS,

$$R^2 = \frac{\sum_{i=1}^{N}(\hat{y}_i - \bar{y})^2}{\sum_{i=1}^{N}(y_i - \bar{y})^2}$$



and $0 \le R^2 \le 1$.

$R^2 = 1$: Perfect regression.

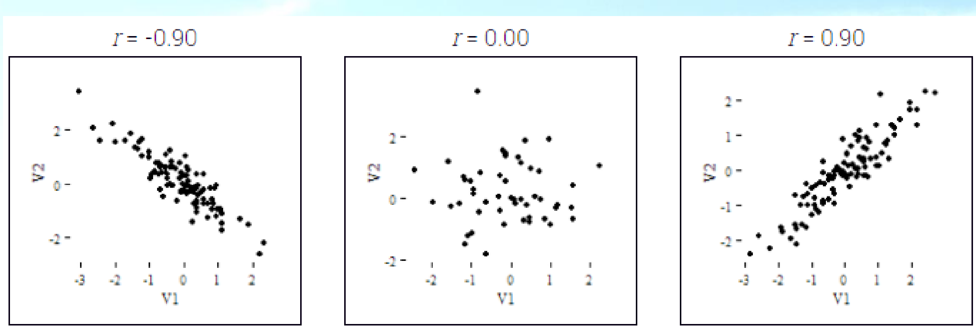
$R^2 = 0$: Baseline model. Predictions are the average.

$R^2 = 1 - \frac{\sum_{i=1}^{N}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{N}(y_i - \bar{y})^2} \in [-\infty, 1]$ for other types of regression as a general measure of goodness-of-fit, and should no longer be called R^2.

Pearson Correlation Coefficient

皮尔森相关系数

$$r_{XY} = \frac{cov(X,Y)}{\sqrt{var(X)} \cdot \sqrt{var(Y)}} = \frac{\sum(X_i - \bar{X})(Y_i - \bar{Y})}{\sqrt{\sum(X_i - \bar{X})^2} \cdot \sqrt{\sum(Y_i - \bar{Y})^2}}$$



What are the relation between R^2 and r ?

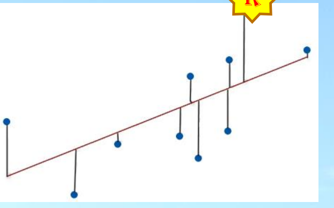
R^2 is not enough for evaluating regression

$$y_i = w_0 + \mathbf{w}^T \mathbf{x}_i + \epsilon_i, \qquad i = 1, \cdots, N$$

Dependent variable

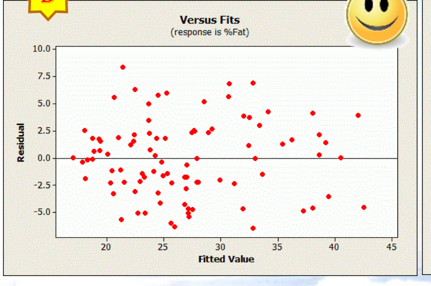
Fitted values (deterministic)

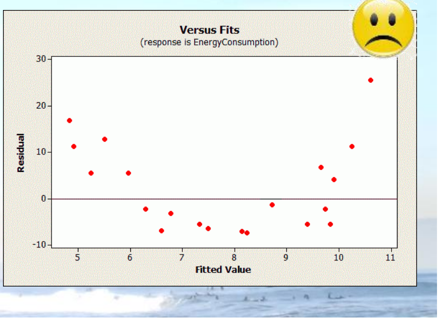
Error, residual, noise (stochastic)



- R^2 : percentage of dependent variable variations that the linear model explains.
- R^2 does not indicate if the regression model provides an adequate fit to the data.

- Use ***residual plots*** to check whether the model is adequate.
- Poor fitting if error is not random





Evaluating each coefficient

$$y_i = w_0 + \sum_{j=1}^{d} w_j x_{ij} + \epsilon_i, \qquad \epsilon_i \sim N(0, \sigma^2), i = 1, \cdots, N,$$

Does each w_j contribute?

Test for statistical significance of regression coefficients

$$\frac{\hat{w}_j - w_j}{s_{\hat{w}_j}} \sim t_{N-d-1}, \qquad j = 0, 1, \cdots, d.$$

休息10秒钟

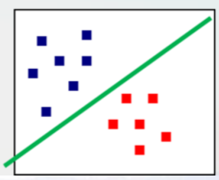
3.4 最小平方误差分类器
Minimum Squared Error (MSE)

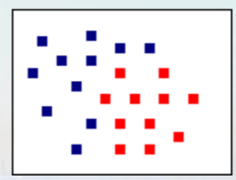
回顾

讨论

- 感知器有什么问题？
  - 样本线性不可分呢？
  - 线性可分时多解？
  - 多类呢？
- 有问题怎么办？
  - 容忍错误，使错误尽量小
    - 比如强制收敛、MSE
  - 寻求非线性方法
    - 比如神经网络、SVM
  - 寻求“最优分类器”
    - 比如支持向量机SVM
  - 多类分类方法
  - 用两类分类器完成多类分类

$$\boldsymbol{\alpha}(k+1) = \boldsymbol{\alpha}(k) + \rho_k \sum_{\mathbf{y}_j \in \mathrm{Y}^k} \mathbf{y}_j$$

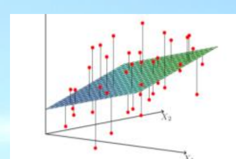




Xuegong Zhang 65

MSE方法的思想

- 对线性不可分情况，怎样最小化线性分类器的错误？



- 线性回归：求$\boldsymbol{w}$使$y_i = \boldsymbol{w}^T \boldsymbol{x}_i, i = 1, \cdots, N$
  - 不可能全部样本正好都满足，于是最小化平方误差，方法是：

$$\min E = \frac{1}{N} \sum_{j=1}^{N} \left(f(\boldsymbol{x}_j) - y_j\right)^2$$

- 线性分类器：求$\boldsymbol{\alpha}$使$\boldsymbol{\alpha}^T \boldsymbol{y}_i > 0, i = 1, \cdots, N$
  - 当样本集线性不可分时，使尽可能多的样本满足不等式

注意：这里切换回感知器
时的规范化增广向量形式

- 考虑：为每个样本引入b_i, 并令

$$\boldsymbol{\alpha}^T \boldsymbol{y}_i = b_i > 0, \quad j = 1, \cdots, N$$

不等式组 ➔ 等式组，可用最小二乘求解

Xuegong Zhang 66

MSE分类器准则

$$\boldsymbol{\alpha}^T\mathbf{y}_i > 0,\ i = 1,\cdots,N \quad \Longleftrightarrow \quad \boldsymbol{\alpha}^T\mathbf{y}_i = b_i > 0,\ i = 1,\cdots,N$$

- 不等式转化成等式 $\quad \mathbf{Y}\boldsymbol{\alpha} = \boldsymbol{b},\ \boldsymbol{b} = [b_1, b_2, \cdots, b_N]^T$
- MSE准则 $\quad \boldsymbol{\alpha}^*:\ \min\limits_{\boldsymbol{\alpha}} J_S(\boldsymbol{\alpha})$

$$J_S(\boldsymbol{\alpha}) = \|\mathbf{Y}\boldsymbol{\alpha} - \mathbf{b}\|^2 = \sum_{i=1}^{N}(\boldsymbol{\alpha}^T\mathbf{y}_i - b_i)^2$$

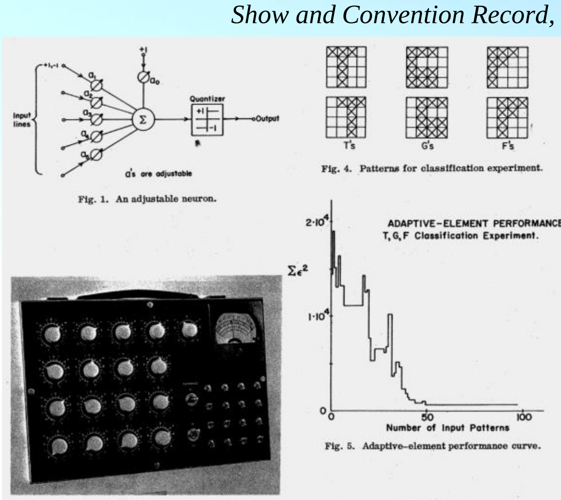
- 解法：
  - 最小二乘伪逆解 $\quad \boldsymbol{\alpha}^* = (\mathbf{Y}^T\mathbf{Y})^{-1}\mathbf{Y}^T\mathbf{b} = \mathbf{Y}^+\mathbf{b}, \quad \mathbf{Y}^+ = (\mathbf{Y}^T\mathbf{Y})^{-1}\mathbf{Y}^T$
  - 梯度下降学习

$$\nabla J_s(\boldsymbol{\alpha}) = 2\mathbf{Y}^T(\mathbf{Y}\boldsymbol{\alpha} - \mathbf{b})$$
$$\boldsymbol{\alpha}(k+1) = \boldsymbol{\alpha}(k) + \rho_k(b_k - \boldsymbol{\alpha}(k)^T\mathbf{y}^k)\mathbf{y}^k$$

--- Widrow-Hoff算法，ADALINE

ADALINE

Widrow & Hoff, Adaptive switching circuits, *1960 IRE Western Electric Show and Convention Record, Part 4*, pp.96-104, Aug, 1960



Widrow & Hoff, Adaptive switching circuits, *1960 IRE Western Electric Show and Convention Record, Part 4*, pp.96-104, Aug, 1960

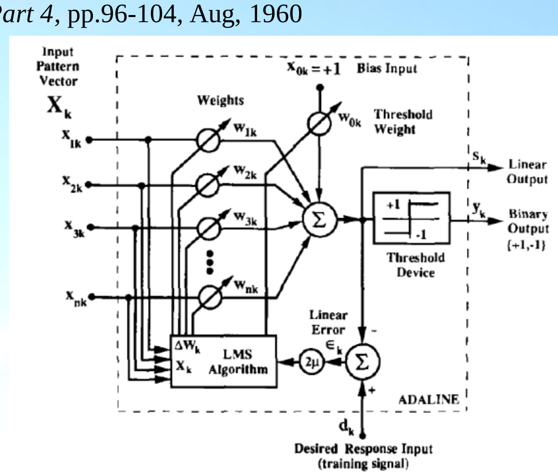


Fig. 2. Adaptive linear element (Adaline).

Widrow & Lehr, 30 years of adaptive neural networks: Perceptron, Madaline, and Backpropagation, *Proceedings of the IEEE*, 78(9): 1415-1442, 1990

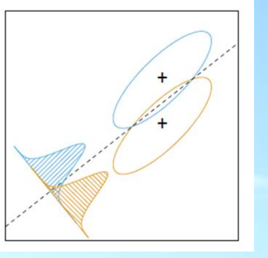
有没有问题？

如何给定 $\boldsymbol{b}=[b_1,b_2,\cdots,b_N]^T$ ？

- 可以证明，如果$\boldsymbol{b}$选为

$$b_i=\begin{cases}N/N_1\,, & if\ y_i\epsilon\omega_1\\ N/N_2\,, & if\ y_i\epsilon\omega_2\end{cases},$$

则MSE解等价于$w_0=-\widehat{m}$的FLD解。



- 如果$\boldsymbol{b}$选为：

$$b_i=1,\ i=1,\cdots,N\ ,$$

则当$N\to\infty$时，MSE解以最小均方误差最优逼近贝叶斯判别函数

$$g_0(\boldsymbol{x})=P(\omega_1|\boldsymbol{x})-P(\omega_2|\boldsymbol{x})$$

即$\boldsymbol{\alpha}_{\text{MSE}}$使$e^2=\int[\boldsymbol{\alpha}^T\boldsymbol{y}-g_0(\boldsymbol{x})]^2p(\boldsymbol{x})d(\boldsymbol{x})$最小。

单选题 1分 设置

休息4分钟，回到座位后请答题

A 已回座位

B 还没有

提交

3.5 罗杰斯特回归
Logistic Regression

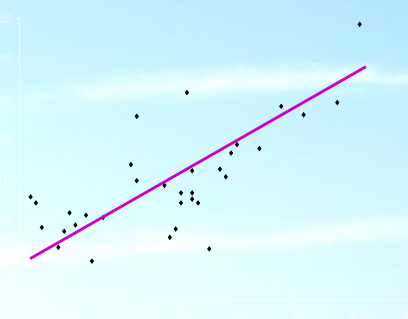
注：单词logistic与logic没有关系，不应译作“逻辑回归”，
应音译作“罗杰斯特回归”或“罗杰斯蒂回归”

简单线性回归

Age	SBP	Age	SBP	Age	SBP
22	131	41	139	52	128
23	128	41	171	54	105
24	116	46	137	56	145
27	106	47	111	57	141
28	114	48	115	58	153
29	123	49	133	59	157
30	117	49	128	63	155
32	122	50	183	67	176
33	99	51	130	71	172
35	121	51	133	77	178
40	147	51	144	81	217

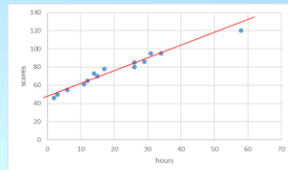
Table 1 Age and systolic blood pressure (SBP) among 33 adult women

$$SBP = 81.54 + 1.222 \cdot Age$$



回归用来预测二值结果？

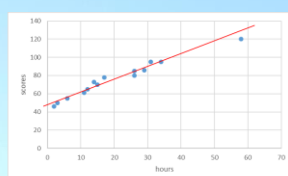
Student id	Final score	Study Hours per Week
1	0	3
2	1	34
3	0	17
4	0	6
5	0	12
6	1	15
7	1	26
8	1	29
9	0	14
10	1	58
11	0	2
12	1	31
13	1	26
14	0	11



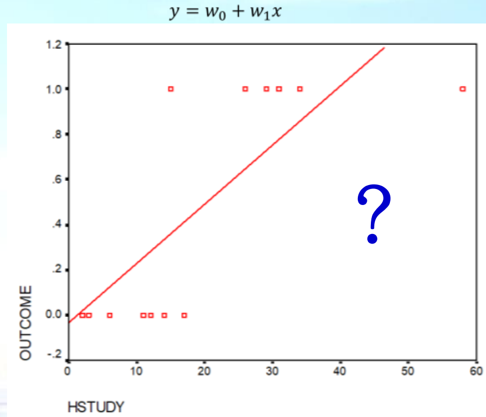
Simple Learn Regression
$y = w_0 + w_1 x$

回归用来预测二值结果？

Student id	Final score	Study Hours per Week
1	0	3
2	1	34
3	0	17
4	0	6
5	0	12
6	1	15
7	1	26
8	1	29
9	0	14
10	1	58
11	0	2
12	1	31
13	1	26
14	0	11



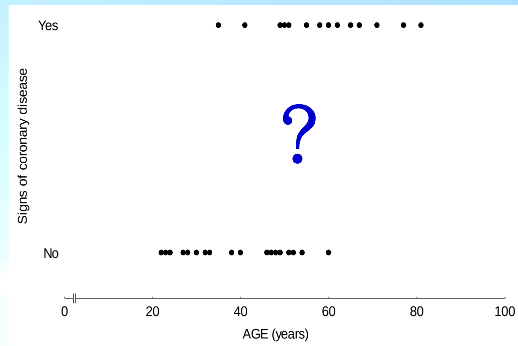
Simple Learn Regression
$y = w_0 + w_1 x$



用年龄回归冠心病

Age and signs of coronary heart disease (CD)

Age	CD	Age	CD	Age	CD
22	0	40	0	54	0
23	0	41	1	55	1
24	0	46	0	58	1
27	0	47	0	60	1
28	0	48	0	60	0
30	0	49	1	62	1
30	0	49	0	65	1
32	0	50	1	67	1
33	0	51	0	71	1
35	1	51	1	77	1
38	0	52	0	81	1

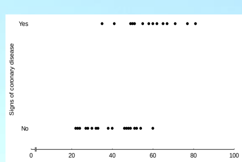


Xuegong Zhang 75

让我们来计数

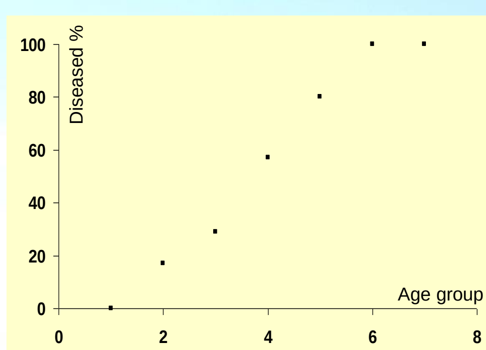
Age and signs of coronary heart disease (CD)

Age	CD	Age	CD	Age	CD
22	0	40	0	54	0
23	0	41	1	55	1
24	0	46	0	58	1
27	0	47	0	60	1
28	0	48	0	60	0
30	0	49	1	62	1
30	0	49	0	65	1
32	0	50	1	67	1
33	0	51	0	71	1
35	1	51	1	77	1
38	0	52	0	81	1



Prevalence (%) of signs of CD according to age group

		Diseased	
Age group	# in group	#	%
20 - 29	5	0	0
30 - 39	6	1	17
40 - 49	7	2	29
50 - 59	7	4	57
60 - 69	5	4	80
70 - 79	2	2	100
80 - 89	1	1	100

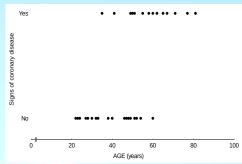


Xuegong Zhang 76

让我们来计数

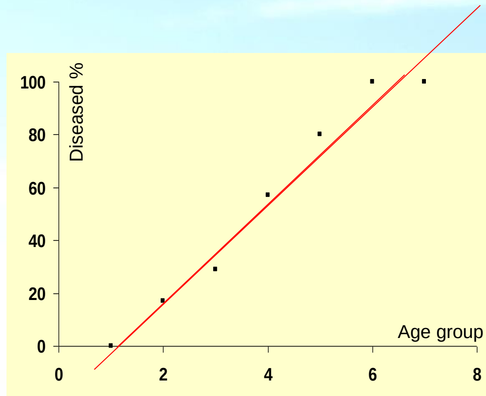
Age and signs of coronary heart disease (CD)

Age	CD	Age	CD	Age	CD
22	0	40	0	54	0
23	0	41	1	55	1
24	0	46	0	58	1
27	0	47	0	60	1
28	0	48	0	60	0
30	0	49	1	62	1
30	0	49	0	65	1
32	0	50	1	67	1
33	0	51	0	71	1
35	1	51	1	77	1
38	0	52	0	81	1



Prevalence (%) of signs of CD according to age group

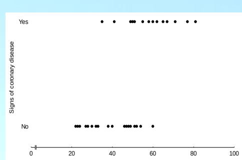
Age group	# in group	Diseased #	Diseased %
20 - 29	5	0	0
30 - 39	6	1	17
40 - 49	7	2	29
50 - 59	7	4	57
60 - 69	5	4	80
70 - 79	2	2	100
80 - 89	1	1	100



让我们来计数

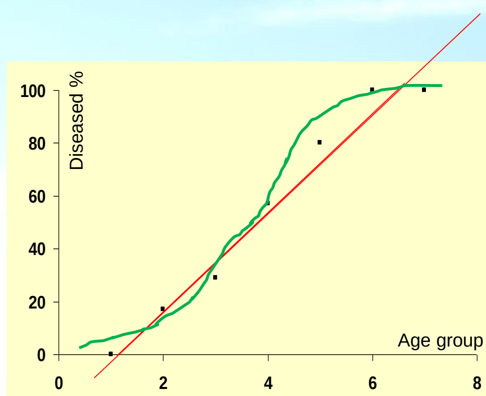
Age and signs of coronary heart disease (CD)

Age	CD	Age	CD	Age	CD
22	0	40	0	54	0
23	0	41	1	55	1
24	0	46	0	58	1
27	0	47	0	60	1
28	0	48	0	60	0
30	0	49	1	62	1
30	0	49	0	65	1
32	0	50	1	67	1
33	0	51	0	71	1
35	1	51	1	77	1
38	0	52	0	81	1



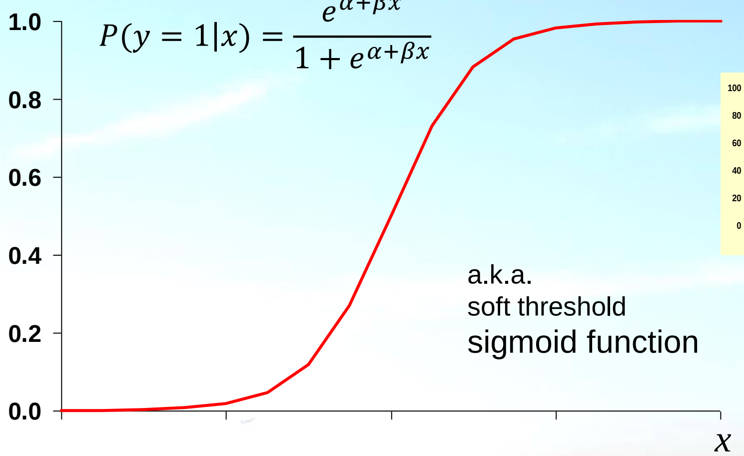
Prevalence (%) of signs of CD according to age group

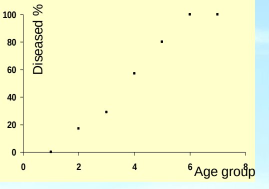
Age group	# in group	Diseased #	Diseased %
20 - 29	5	0	0
30 - 39	6	1	17
40 - 49	7	2	29
50 - 59	7	4	57
60 - 69	5	4	80
70 - 79	2	2	100
80 - 89	1	1	100



罗杰斯特函数Logistic function

$$P(y=1|x)=\frac{e^{\alpha+\beta x}}{1+e^{\alpha+\beta x}}$$

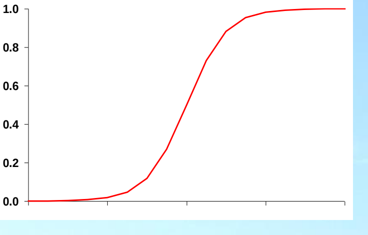




a.k.a.
soft threshold
sigmoid function

罗杰斯特回归与几率

$$P(y|x)=\frac{e^{\alpha+\beta x}}{1+e^{\alpha+\beta x}}$$



Odds 几率 $$\frac{P(y|x)}{1-P(y|x)}=e^{\alpha+\beta x}$$

Log odds 对数几率 $$\ln\left[\frac{P(y|x)}{1-P(y|x)}\right]=\alpha+\beta x$$

$P(y|x)$的罗杰特（logit）函数

注意中文中的混淆词：几率、机率、概率、可能性、…

多元罗杰斯特回归

$$P(y|\boldsymbol{x})=\frac{e^{\alpha+\beta_1x_1+\beta_2x_2+\cdots+\beta_px_p}}{1+e^{\alpha+\beta_1x_1+\beta_2x_2+\cdots+\beta_px_p}}$$

$$\ln\left(\frac{P(y|\boldsymbol{x})}{1-P(y|\boldsymbol{x})}\right)=\alpha+\beta_1x_1+\beta_2x_2+\cdots+\beta_px_p$$

$$odds=\frac{P(y|\boldsymbol{x})}{1-P(y|\boldsymbol{x})}=e^{\alpha+\beta_1x_1+\beta_2x_2+\cdots+\beta_px_p}$$

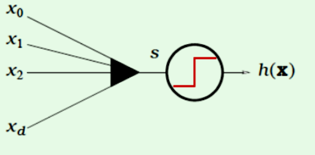
- β_i的解释
  - 在其他因素不变的情况下，因素x_i增加一个单位带来的对数几率的增加
  - 可以用来从流行病学数据中研究各种因素与患病的关系

三种线性机器

$$s=\sum_{i=0}^{d}w_ix_i$$

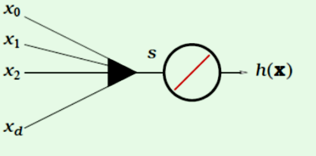
linear classification

$$h(\mathbf{x})=\mathrm{sign}(s)$$



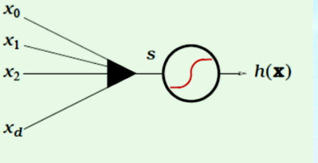
linear regression

$$h(\mathbf{x})=s$$



logistic regression

$$h(\mathbf{x})=\theta(s)$$

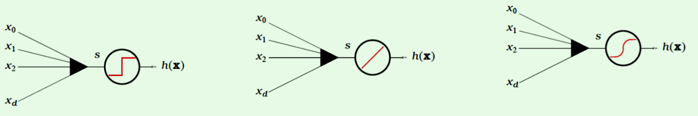


Abu-Mostafa, Magdon-Ismail, Lin, *Learning from Data, Lecture 9*

罗杰斯特回归的概率意义

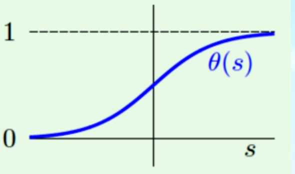
$$s = \sum_{i=0}^{d} w_i x_i$$

linear classification	linear regression	logistic regression
$h(\mathbf{x}) = \text{sign}(s)$	$h(\mathbf{x}) = s$	$h(\mathbf{x}) = \theta(s)$



Abu-Mostafa, Magdon-Ismail, Lin, *Learning from Data, Lecture 9*

$$\theta(s) = \frac{e^s}{1+e^s}$$



- $s = \boldsymbol{w}^T\boldsymbol{x}$是事件的总信息，是各种因素（特征）的加权求和
- $h(\boldsymbol{x}) = \theta(s)$是对事件$y = 1$ 概率的估计

机器学习的基本要素

- 怎样造一个学习机器？
  - 它需要老师
    → 我们设计它（特征和模型）
  - 它需要训练/学习材料
    → 训练数据
  - 我们需要为它树立学习的目标
    → 目标函数、学习准则
  - 我们需要告诉它怎样学
    → 学习/训练算法

机器学习的基本要素：感知器版

- 怎样造一个学习机器？
  - 它需要老师
    - → 我们设计它（特征和模型） $y=\mathrm{sgn}(\sum_{i=1}^{d} w_i x_i + w_0)$
  - 它需要训练/学习材料
    - → 训练数据 $\{(\boldsymbol{x}_1, y_1), \dots, (\boldsymbol{x}_N, y_N)\}$, $\boldsymbol{x}_j \in R^{d+1}, y_j \in \{-1,1\}$
  - 我们需要为它树立学习的目标
    - → 目标函数、学习准则 $\min J_P(\boldsymbol{\alpha}) = \sum_{\mathbf{y}_j \in \mathcal{Y}^k} (-\boldsymbol{\alpha}^T \mathbf{y}_j)$
  - 我们需要告诉它怎样学
    - → 学习/训练算法 $\boldsymbol{\alpha}(k+1) = \boldsymbol{\alpha}(k) - \rho_k \nabla J = \boldsymbol{\alpha}(k) + \rho_k \sum_{\mathbf{y}_j \in \mathrm{Y}^k} \mathbf{y}_j$

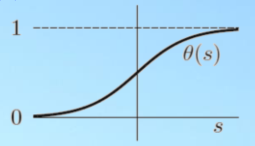
机器学习的基本要素：线性回归版

- 怎样造一个学习机器？
  - 它需要老师
    - → 我们设计它（特征和模型） $f(\boldsymbol{x}) = \sum_{i=0}^{d} w_i x_i = \boldsymbol{w}^T \boldsymbol{x}$
  - 它需要训练/学习材料
    - → 训练数据 $\{(\boldsymbol{x}_1, y_1), \dots, (\boldsymbol{x}_N, y_N)\}$, $\boldsymbol{x}_j \in R^{d+1}, y_j \in R$
  - 我们需要为它树立学习的目标
    - → 目标函数、学习准则 $\min E = \frac{1}{N} \sum_{j=1}^{N} \left(f(\boldsymbol{x}_j) - y_j \right)^2$
  - 我们需要告诉它怎样学
    - → 学习/训练算法 $\boldsymbol{w}(k+1) = \boldsymbol{w}(k) - \rho_k \nabla E$

机器学习的基本要素：罗杰斯特回归版？

- 怎样造一个学习机器？

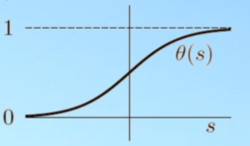
$$\theta(s)=\frac{e^s}{1+e^s}$$



  – 它需要老师
    → 我们设计它（特征和模型） $h(\boldsymbol{x})=\theta(\boldsymbol{w}^T\boldsymbol{x})$
  – 它需要训练/学习材料
    → 训练数据 $\{(\boldsymbol{x}_1,y_1),\ldots,(\boldsymbol{x}_N,y_N)\}$, $\boldsymbol{x}_j\in R^{d+1}, y_j\in\{-1,1\}$
  – 我们需要为它树立学习的目标
    → 目标函数、学习准则 ？
  – 我们需要告诉它怎样学
    → 学习/训练算法？

休息10秒钟

机器学习的基本要素：罗杰斯特回归版？



$$\theta(s)=\frac{e^s}{1+e^s}$$

- 怎样造一个学习机器？
  - 它需要老师
    - → 我们设计它（特征和模型） $h(\boldsymbol{x})=\theta(\boldsymbol{w}^T\boldsymbol{x})$
  - 它需要训练/学习材料
    - → 训练数据 $\{(\boldsymbol{x}_1,y_1),\ldots,(\boldsymbol{x}_N,y_N)\}$, $\boldsymbol{x}_j\in R^{d+1}, y_j\in\{-1,1\}$
  - 我们需要为它树立学习的目标
    - → 目标函数、学习准则 ？
  - 我们需要告诉它怎样学
    - → 学习/训练算法？

考虑样本的产生模型

- 设独立同分布(i.i.d.)样本集 $\{(\boldsymbol{x}_1,y_1),\ldots,(\boldsymbol{x}_N,y_N)\}$, $\boldsymbol{x}_j\in R^{d+1}, y_j\in\{-1,1\}$ 依以下概率产生：

$$P(y|\boldsymbol{x})=\begin{cases} f(\boldsymbol{x}) & for\ y=+1 \\ 1-f(\boldsymbol{x}) & for\ y=-1 \end{cases}$$

生成模型
Generative model

- 罗杰斯特回归用 $h(\boldsymbol{x})=\theta(\boldsymbol{w}^T\boldsymbol{x})$ 估计 $f(\boldsymbol{x})$

考虑样本的产生模型

- 设独立同分布(i.i.d.)样本集 $\{(\boldsymbol{x}_1, y_1), \ldots, (\boldsymbol{x}_N, y_N)\}$, $\boldsymbol{x}_j \in R^{d+1}, y_j \in \{-1,1\}$ 依以下概率产生：

$$P(y|\boldsymbol{x}) = \begin{cases} f(\boldsymbol{x}) & for\ y = +1 \\ 1 - f(\boldsymbol{x}) & for\ y = -1 \end{cases}$$

生成模型
Generative model

- 罗杰斯特回归用$h(\boldsymbol{x}) = \theta(\boldsymbol{w}^T\boldsymbol{x})$估计$f(\boldsymbol{x})$
- 似然函数（Likelihood）：
  - 对数据中的一个实例$(\boldsymbol{x}_j, y_j)$，如果$h = f$，我们有多大可能对$\boldsymbol{x}_j$得到y_j？

$$P(y_j|\boldsymbol{x}_j) = \begin{cases} h(\boldsymbol{x}_j) & for\ y_j = +1 \\ 1 - h(\boldsymbol{x}_j) & for\ y_j = -1 \end{cases}$$

  - 换言之，已经有这个数据实例，h有多大可能是产生数据的模型？

D

似然函数

- 设独立同分布(i.i.d.)样本集 $\{(\boldsymbol{x}_1, y_1), \ldots, (\boldsymbol{x}_N, y_N)\}$, $\boldsymbol{x}_j \in R^{d+1}, y_j \in \{-1,1\}$ 依以下概率产生：

$$P(y|\boldsymbol{x}) = \begin{cases} f(\boldsymbol{x}) & for\ y = +1 \\ 1 - f(\boldsymbol{x}) & for\ y = -1 \end{cases}$$

生成模型
Generative model

- 罗杰斯特回归用$h(\boldsymbol{x}) = \theta(\boldsymbol{w}^T\boldsymbol{x})$估计$f(\boldsymbol{x})$
- 似然函数（Likelihood）：
  - 对数据中的一个实例$(\boldsymbol{x}_j, y_j)$，如果$h = f$，我们有多大可能对$\boldsymbol{x}_j$得到y_j？

$$P(y_j|\boldsymbol{x}_j) = \begin{cases} h(\boldsymbol{x}_j) & for\ y_j = +1 \\ 1 - h(\boldsymbol{x}_j) & for\ y_j = -1 \end{cases}$$

  - 换言之，已经有这个数据实例，h有多大可能是产生数据的模型？
- 注意到$\theta(-s) = 1 - \theta(s)$，$\boldsymbol{x}_j$上的似然函数可写为：

H

$$P(y_j|\boldsymbol{x}_j) = \theta(y_j \boldsymbol{w}^T \boldsymbol{x}_j)$$

似然函数

- 设独立同分布(i.i.d.)样本集 $\{(\boldsymbol{x}_1, y_1), \ldots, (\boldsymbol{x}_N, y_N)\}$, $\boldsymbol{x}_j \in R^{d+1}, y_j \in \{-1,1\}$ 依以下概率产生：

$$P(y|\boldsymbol{x}) = \begin{cases} f(\boldsymbol{x}) & for\ y = +1 \\ 1 - f(\boldsymbol{x}) & for\ y = -1 \end{cases}$$

生成模型
Generative model

- 罗杰斯特回归用 $h(\boldsymbol{x}) = \theta(\boldsymbol{w}^T\boldsymbol{x})$ 估计 $f(\boldsymbol{x})$
- 似然函数（Likelihood）：
  - 对数据中的一个实例 $(\boldsymbol{x}_j, y_j)$，如果 $h = f$，我们有多大可能对 $\boldsymbol{x}_j$ 得到 y_j？

$$P(y_j|\boldsymbol{x}_j) = \begin{cases} h(\boldsymbol{x}_j) & for\ y_j = +1 \\ 1 - h(\boldsymbol{x}_j) & for\ y_j = -1 \end{cases}$$

  - 换言之，已经有这个数据实例，h 有多大可能是产生数据的模型？

问题：似然函数是谁的函数？

D

- 注意到 $\theta(-s) = 1 - \theta(s)$，$\boldsymbol{x}_j$ 上的似然函数可写为：

$$P(y_j|\boldsymbol{x}_j) = \theta(y_j\boldsymbol{w}^T\boldsymbol{x}_j)$$

单选题 10分 设置

似然函数是谁的函数？

$$P(y_j|\boldsymbol{x}_j) = \theta(y_j\boldsymbol{w}^T\boldsymbol{x}_j)$$

A 是 y_i 的函数

B 是 $\boldsymbol{w}$ 的函数

C 是 $\boldsymbol{x}_i$ 的函数

D 是 $(\boldsymbol{x}_j, y_j)$ 的函数

提交

似然函数

- 设独立同分布(i.i.d.)样本集 $\{(\boldsymbol{x}_1, y_1), \dots, (\boldsymbol{x}_N, y_N)\}$, $\boldsymbol{x}_j \in R^{d+1}, y_j \in \{-1,1\}$ 依以下概率产生：

$$P(y|\boldsymbol{x}) = \begin{cases} f(\boldsymbol{x}) & for\ y = +1 \\ 1 - f(\boldsymbol{x}) & for\ y = -1 \end{cases}$$

生成模型
Generative model

- 罗杰斯特回归用$h(\boldsymbol{x}) = \theta(\boldsymbol{w}^T\boldsymbol{x})$估计$f(\boldsymbol{x})$
- 似然函数（Likelihood）：
  - 对数据中的一个实例$(\boldsymbol{x}_j, y_j)$，如果$h = f$，我们有多大可能对$\boldsymbol{x}_j$得到y_j？

$$P(y_j|\boldsymbol{x}_j) = \begin{cases} h(\boldsymbol{x}_j) & for\ y_j = +1 \\ 1 - h(\boldsymbol{x}_j) & for\ y_j = -1 \end{cases}$$

  - 换言之，已经有这个数据实例，h有多大可能是产生数据的模型？
- 注意到$\theta(-s) = 1 - \theta(s)$，**模型在$\boldsymbol{x}_j$上的似然函数**可写为：

$$l\left(h|(\boldsymbol{x}_j, y_j)\right) = P\left(y_j|\boldsymbol{x}_j, h\right) = \theta(y_j \boldsymbol{w}^T \boldsymbol{x}_j)$$

罗杰斯特回归的目标：最大化似然函数

- 参数为$\boldsymbol{w}$的罗杰斯特模型在i.i.d.数据$\{(\boldsymbol{x}_1, y_1), \dots, (\boldsymbol{x}_N, y_N)\}$, $\boldsymbol{x}_j \in R^{d+1}, y_j \in \{-1,1\}$上的似然函数是

$$L(\boldsymbol{w}) = \prod_{j=1}^{N} P\left(y_j|\boldsymbol{x}_j\right) = \prod_{j=1}^{N} \theta(y_j \boldsymbol{w}^T \boldsymbol{x}_j)$$

罗杰斯特回归的目标：最大化似然函数

- 参数为$\boldsymbol{w}$的罗杰斯特模型在i.i.d.数据$\{(\boldsymbol{x}_1, y_1), \ldots, (\boldsymbol{x}_N, y_N)\}$，$\boldsymbol{x}_j \in R^{d+1}, y_j \in \{-1,1\}$上的似然函数是

$$L(\boldsymbol{w}) = \prod_{j=1}^{N} P(y_j|\boldsymbol{x}_j) = \prod_{j=1}^{N} \theta(y_j \boldsymbol{w}^T \boldsymbol{x}_j)$$

- 最大化似然函数，等价于最小化以下目标函数：

$$\min \quad E(\boldsymbol{w}) = -\frac{1}{N}\ln(L(\boldsymbol{w})) = -\frac{1}{N}\ln\left(\prod_{j=1}^{N} \theta(y_j \boldsymbol{w}^T \boldsymbol{x}_j)\right)$$

$$= \frac{1}{N}\sum_{j=1}^{N} \ln\left(\frac{1}{\theta(y_j \boldsymbol{w}^T \boldsymbol{x}_j)}\right)$$

$$= \frac{1}{N}\sum_{j=1}^{N} \ln\left(1 + e^{-y_j \boldsymbol{w}^T \boldsymbol{x}_j}\right)$$

$$\left[\theta(s) = \frac{1}{1 + e^{-s}}\right]$$

H

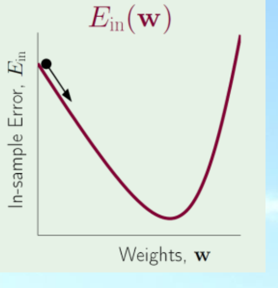
求解：梯度下降法

梯度下降的一般原理：

$$\boldsymbol{w}(k+1) = \boldsymbol{w}(k) + \eta \hat{\boldsymbol{v}}$$

其中，η：步长（学习率）

$$\hat{\boldsymbol{v}} = -\nabla E(\boldsymbol{w}(k))$$



Abu-Mostafa, Magdon-Ismail, Lin, *Learning from Data, Lecture 9*

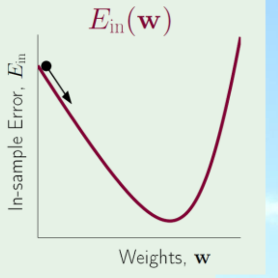
求解：梯度下降法

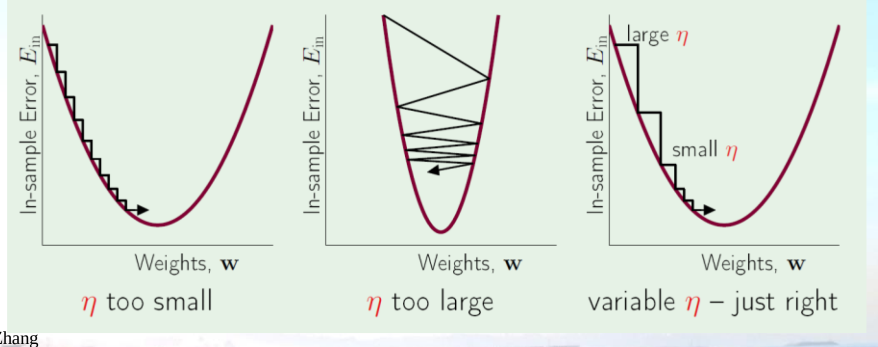
梯度下降的一般原理：

$$\boldsymbol{w}(k+1) = \boldsymbol{w}(k) + \eta\hat{\boldsymbol{v}}$$

其中，η：步长（学习率）

$$\hat{\boldsymbol{v}} = -\nabla E(\boldsymbol{w}(k))$$





Abu-Mostafa, Magdon-Ismail, Lin, *Learning from Data, Lecture 9*

罗杰斯特回归算法

1. Set $k=0$, initialize $\boldsymbol{w}(0)$
2. Do
   – Compute the gradient $\nabla E = -\frac{1}{N}\sum_{j=1}^{N}\frac{y_j \boldsymbol{x}_j}{1+e^{y_j \boldsymbol{w}(k)^T \boldsymbol{x}_j}}$

   – Update the weights $\boldsymbol{w}(k+1) = \boldsymbol{w}(k) - \eta\nabla E$, set $k=k+1$

   Until the stopping criterion met
3. Return the final weights $\boldsymbol{w}$

Abu-Mostafa, Magdon-Ismail, Lin, *Learning from Data, Lecture 9*

罗杰斯特回归算法

1. Set $k = 0$, initialize $\boldsymbol{w}(0)$
2. Do
   – Compute the gradient $\nabla E = -\frac{1}{N}\sum_{j=1}^{N}\frac{y_j \boldsymbol{x}_j}{1+e^{y_j \boldsymbol{w}(k)^T \boldsymbol{x}_j}}$
   – Update the weights $\boldsymbol{w}(k+1) = \boldsymbol{w}(k) - \eta\nabla E$, set $k = k+1$
   Until the stopping criterion met
3. Return the final weights $\boldsymbol{w}$

Abu-Mostafa, Magdon-Ismail, Lin, *Learning from Data, Lecture 9*

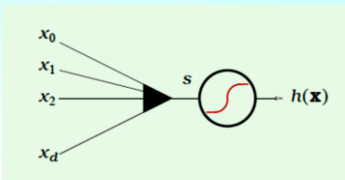
- 初始化：可以全零，更好是小随机数，比如0均值、小方差的正态分布
- 终止条件：梯度低于一定阈值，或迭代次数达到预设上限

机器学习的基本要素：罗杰斯特回归版

- 怎样造一个学习机器？
  – 它需要老师
    → 我们设计它（特征和模型） $h(\boldsymbol{x}) = \theta(\boldsymbol{w}^T\boldsymbol{x})$
  – 它需要训练/学习材料
    → 训练数据 $\{(\boldsymbol{x}_1, y_1), \dots, (\boldsymbol{x}_N, y_N)\}$, $\boldsymbol{x}_j \in R^{d+1}, y_j \in \{-1,1\}$
  – 我们需要为它树立学习的目标
    → 目标函数、学习准则 $\min E(\boldsymbol{w}) = \frac{1}{N}\sum_{j=1}^{N} ln\left(1 + e^{-y_j \boldsymbol{w}^T \boldsymbol{x}_j}\right)$
  – 我们需要告诉它怎样学
    → 学习/训练算法 $\boldsymbol{w}(k+1) = \boldsymbol{w}(k) - \eta\nabla E$

有没有问题？

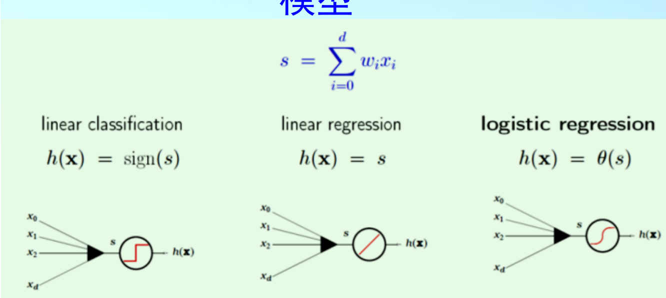
$h(\boldsymbol{x}) = \theta(\boldsymbol{w}^T\boldsymbol{x})$



- 如何分类决策？分类器是什么？
  - $h(\boldsymbol{x}) \gtrless 0.5$
    - 最小错误率决策
  - 根据ROC曲线
  - 根据两类错误率的相对损失（风险）
    - 最小风险决策

线性学习机器小结

模型



学习算法

$$\boldsymbol{w}(k+1) = \boldsymbol{w}(k) - \rho_k \nabla E$$

目标

- For perceptron

$$\min \quad J_P(\boldsymbol{\alpha}) = \sum_{\boldsymbol{y}_j \in Y^k} (-\boldsymbol{\alpha}^T \boldsymbol{y}_j)$$

- For linear regression

$$\min \quad E(\boldsymbol{w}) = \frac{1}{N}\sum_{j=1}^{N}(\boldsymbol{w}^T\boldsymbol{x}_j - y_j)^2$$

- For logistic regression

$$\min \quad E(\boldsymbol{w}) = \frac{1}{N}\sum_{j=1}^{N} ln\left(1 + e^{-y_j \boldsymbol{w}^T \boldsymbol{x}_j}\right)$$

本章知识点

- 机器学习的基本概念
- 线性学习机器的基本思想
  - 模型、目标函数、梯度下降优化
- Fisher线性判别、感知器、线性回归、MSE(ADALINE)、罗杰斯特回归
- 似然函数的概念

Xuegong Zhang 105