Part III-B: Medicine AI

Lecture by None
Note by THF

2025 年 4 月 23 日

目录

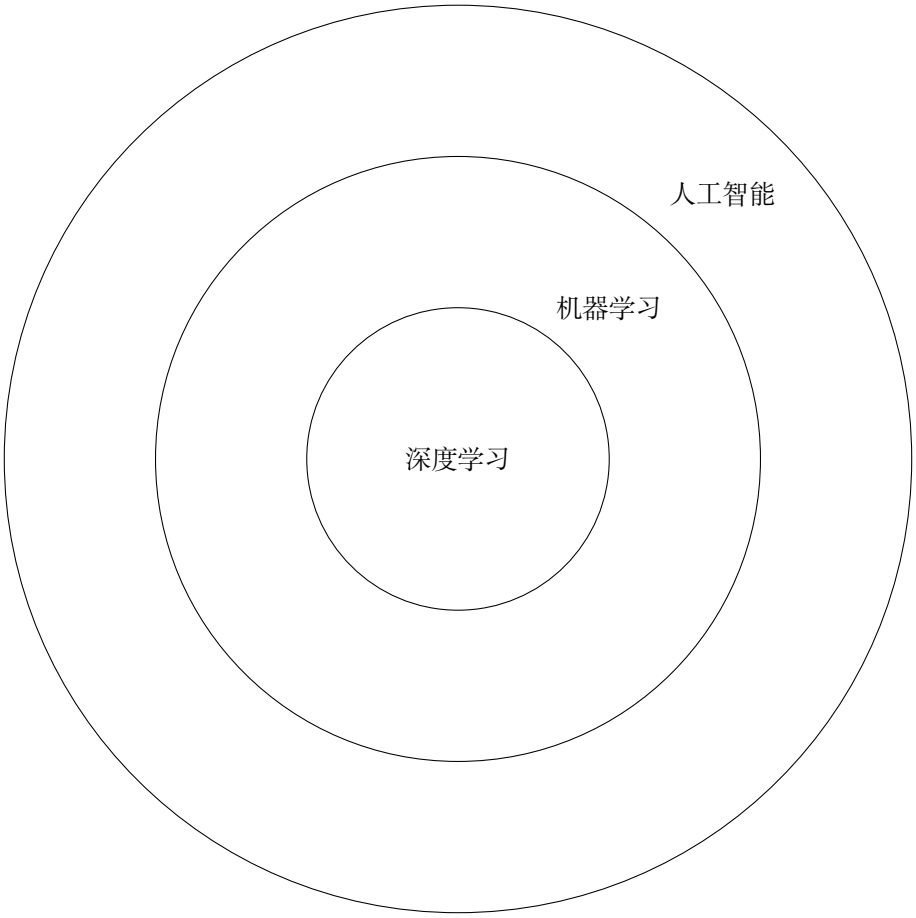
Learn 1 10.07

1 导论

Notation. 机器学习的流程：

1. 确立目标
2. 收集数据
3. 数据预处理
4. 数据分析
5. 模型训练
6. 模型评估优化
7. 预测

机器学习和人工智能的关系：



机器学习算法包含：无监督学习、监督学习、强化学习

1.1 监督学习

Notation. 机器学习选择数据要求：

1. 了解数据类型、属性、量纲
2. 分析分布特性
3. 选择高可信度数据
4. 进行数据表征（将原始数据转换为计算机可识别数据）

Example. 医药领域对小分子、蛋白质、核酸进行特征数字化方法

1.1.1 数据挖掘

1. 通过数据分析与统计学规律
2. 通过爬虫与自动化程序

1.1.2 数据选择

通过一部分数据来体现总体数据

1.1.3 数据表征

Example. 分子指纹：

首先提取分子结构特征（官能团等），使用分子结构特征生成比特向量，每个比特元素对应一种分子片段，通过对比比特向量的相似度来记录分子特征

分子指纹分类：基于子结构、拓扑或路径、药效集团的分子指纹和圆形分子指纹

Notation. SMILES/简化分子线性输入规范：

SMILES 是一种 ASCII 字符串，具体规则如下

Notation. InChI: 国际化合物标识，是规范的线性表示法、基于规范命名法则的唯一标识符

通过分层符号“/”将表示小分子的字符串分层，前三层简化连接表的信息，其他层处理额外问题

Learn InChI

10.07

InChI RULE

1. 主层

主层可包括三个子层：化学式、原子连接、氢原子

$$\text{主层}\begin{cases}\text{化学式}\\ \text{原子连接}\\ \text{氢原子}\end{cases}.$$

Learn 2

10.17

Notation. 氨基酸组成和二肽组成

基础知识：组成人体的二十种氨基酸

除此外还有用于终止密码子的硒半胱氨酸、吡咯赖氨酸（U）

Notation. 氨基酸组成的公式：

$$f(k)=\frac{N_k}{N}, k=1,2,\ldots,20.$$

其中 N_k 表示第 k 种氨基酸的数量，N 表示氨基酸序列长度

表 1: 20 amino acids

Alanine(A)	Arginine(R)	Asparagine(N)
Asparticacid(D)	Cysteine(C)	Glutamine(Q)
Glutamicaci(E)	Glycine(G)	Histidine(H)
Isoleucine(I)	Leucine(L)	Lysine(K)
Methionine(M)	Phenylalani(F)	Proline(P)
Serine(S)	Threonine(T)	Tryptophan(W)
Tyrosine(Y)	Valine(V)	

表 2: 20 种基本氨基酸

丙氨酸,A	精氨酸,R	天冬酰胺,N
天冬氨酸,D	半胱氨酸,C	谷氨酰胺,Q
谷氨酸,E	甘氨酸,G	组氨酸,H
异亮氨酸,I	亮氨酸,L	赖氨酸,K
甲硫氨酸,M	苯丙氨酸,F	脯氨酸,P
丝氨酸,S	苏氨酸,T	色氨酸,W
酪氨酸,Y	缬氨酸,V	

Notation. 二肽组成的公式：

$$f(k,s) = \frac{N_{ks}}{N-1}, k, s = 1, 2, \ldots, 20.$$

同理：N_{ks} 为第 k 种和第 s 种氨基酸形成的二肽数量

Notation. 蛋白质独热编码

使用 $20 \times L$ 的矩阵表示蛋白质的序列信息，L 为蛋白质的序列长度

Example. 含 556 个氨基酸的蛋白质序列可以用 20×556 的矩阵表示，纵向量为二十种氨基酸，横向量为蛋白质在某位置的氨基酸种类

Notation. CTD 描述符

组成、转换与分布（Composition, Transition and Distribution, CTD）根据蛋白质序列中残基的特性编码蛋白质

$$\text{CTD 编码分类方式}\begin{cases}\text{疏水性}\\ \text{范德华体积}\\ \text{极性}\\ \text{可极化性}\\ \text{带电性}\\ \text{表面张力}\\ \text{二级结构}\\ \text{溶剂可及性}\\ \cdots\end{cases}.$$

氨基酸残基分为三类：

表 3: CTD 分类

性质	A	B	C
疏水性	亲水	中性	疏水
范德华体积	(0,2.78)	(2.95,4)	(4.43,8.08)
极性	(0,0.456)	(0.6,0.696)	(0.792,1)
可极化性	(0,0.108)	(0.128,0.186)	(0.219,0.409)
带电性	正电	中性	负电
表面张力	(-0.2,0.16)	(-0.52,-0.3)	(-2.46,-0.98)
二级结构	螺旋	折叠	卷曲
溶剂可及性	包埋	中等	暴露

Notation. 蛋白质二级结构及蛋白质溶剂可及性

1. 蛋白质二级结构 (PSS)
2. 氨基酸溶剂可及性 (PSA)

Learn 3

10.18

编码规则：

$$\begin{cases}\text{二级结构}\begin{cases}\text{H：}\alpha\text{螺旋}\to(0,1,0)\\ \text{E：}\beta\text{折叠}\to(1,0,0)\\ \text{C：其他结构}\to(0,0,1)\end{cases}\\ \text{溶剂可及性}\begin{cases}\text{b: buried（包埋）}\to(1,0)\\ \text{e: exposed（暴露）}\to(0,1)\end{cases}\end{cases}.$$

Prot.		M	V	L	S	P	A	D	K	T	N
Sec.		C	C	C	C	E	H	E	E	H	H
PSSSA	PSS	0	0	0	0	1	0	1	1	0	0
		0	0	0	0	0	1	0	0	1	1
		1	1	1	1	0	0	0	0	0	0
	PSA	0	0	1	0	0	0	1	1	0	0
		1	1	0	1	1	1	0	0	1	1
S.A.		e	e	b	e	e	b	b	e	e	e

Example. 有一条 10 氨基酸长度的蛋白质序列：

PSSSA 使用 5×1000 的矩阵编码蛋白质，每一个氨基酸由一个 5 维向量表示

用 PSSSA 编码时，一般取序列羧基的一侧开始的 1000 个氨基酸编码，如不满 1000 个使用 0 向量补齐

1.2 核酸物质表征

Notation. 基本知识：碱基与核酸

1.2.1 碱基

表 4: 常见碱基

种类	DNA	RNA
嘌呤族（R）	腺嘌呤（A）	
	鸟嘌呤（G）	
嘧啶族（Y）	胞嘧啶（C）	
	胸腺嘧啶（T）	尿嘧啶（U）

Notation. 碱基配对方式：

$$\begin{cases} \text{DNA} \begin{cases} A = T \\ C \equiv G \end{cases} \\ \text{RNA} \begin{cases} A = U \\ C \equiv G \end{cases} \end{cases}.$$

Notation. K-mer

K: DNA 或 RNA 中一个长度为 K 的序列

以该序列为子序列，遍历核酸序列，计算该长度的所有子序列组合出现的频率

Example. 长度为 K 的 K-mer 种类共有 4^k 种可能

如长度为 3 的子序列，子序列每个位置有 A,G,C,U 四种选择，共 4^3 种组合

一段 15 个核酸的 RNA 序列如下：

表 5: 3-mer RNA

C	A	T	C	G	G	T	A	A	C	C	C	A	T

所有可能的长度为 3 的子序列及其频率：

表 6: 3-mers

	RNA seq.	freq.
1	CAT	0.111
2	ATC	0.056
3	TCG	0.056
4	CGG	0.056
...	...	...
12	CCA	0.056
13	ATG	0
...	...	...
64	...	0

Learn 4

10.20

1.3 数据预处理

1.3.1 标准化

Notation. 变量离差标准化：标准化后所有变量范围都在 [0,1] 内

$$y_i = \frac{x_i - x_{\min}}{x_{\max} - x_{\min}}.$$

Example. 一组变量如下：

$$X = (1.5, 1.7, 2.2, 1.2, 1.6, 1.4, 1.1)\,.$$

易得 $x_{\min}=1.1, x_{\max}=2.2$

$$\begin{aligned} y_i &= \frac{x_i - x_{\min}}{x_{\max} - x_{\min}} \\ &= \frac{x_i - 1.1}{2.2 - 1.1} \\ &= \frac{x_i - 1.1}{1.1} \\ &= \frac{x_i}{1.1} - 1. \end{aligned}$$

得 $Y = (0.364, 0.545, 1, 0.091, 0.455, 0.273, 0)$

Notation. *Z-score*（变量标准差）标准化

经过标准化后平均值为 0，标准差为 1

$$z_i = \frac{x_i - \bar{x}}{s} \quad s = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(x_i - \bar{x})^2}.$$

可以看出 s 为原数据的标准差，z_i 值其实等同于标准正态分布中的 u 值：

$$u = \frac{x - \mu}{\sigma} \quad y = \frac{1}{\sigma\sqrt{2\pi}}\mathrm{e}^{-u^2}/2.$$

1.3.2 插补缺失值

Notation. 均值插补

1. 数值性变量：采用平均值插补
2. 离散型：采用众数插补

Notation. 同类均值插补：使用层次聚类方法归类缺失值的样本，用该类别的特征均值插补

Notation. KNN(*K-nearest neighbor*) 缺失值插补：找到与含缺失值样本相似的 K 个样本，使用这 K 个样本在该缺失变量上的均值填充

Learn KNN

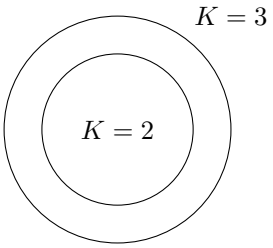
10.20

K-nearest neighbor

基本思路

找到与新输入的待预测样本最临近的 K 个样本，判断这 K 个样本中绝大多数的所属类别作为分类结果输出

条件：已经具有较大的样本量



Notation. KNN 算法的基本要素：距离度量、K 值、分类决策规则

距离度量

Notation. KNN 算法能够分类：特征空间内的样本点之间的距离能够反映样本特征的相似程度

设有两个样本点 $\boldsymbol{x}_i, \boldsymbol{x}_j$ ，以 n 维向量空间作为特征空间，将这两个点表示为：

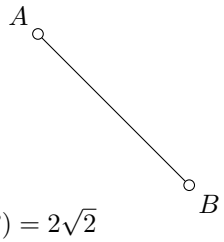
$$\boldsymbol{x}_i, \boldsymbol{x}_j \in \boldsymbol{X}.$$
$$\boldsymbol{x}_i = \left(x_i^1, x_i^2, \ldots, x_i^n\right)^T.$$
$$\boldsymbol{x}_j = \left(x_j^1.x_j^2, \ldots, x_j^n\right)^T.$$

特征点之间的距离定义为：

$$L_p\left(\boldsymbol{x}_i, \boldsymbol{x}_j\right) = \left(\sum_{l=1}^{n} \left|x_i^l - x_j^l\right|^p\right)^{\frac{1}{p}}.$$

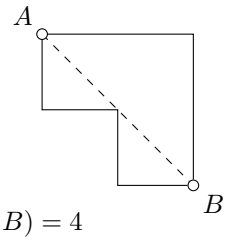
Example. 代入 $p=2$ ，易得 $L_2\left(\boldsymbol{x}_i, \boldsymbol{x}_j\right)$ 为平面上两点间的距离公式，该距离又称为欧氏距离：

$$L_2\left(\boldsymbol{x}_i, \boldsymbol{x}_j\right) = \sqrt{\left(x_{i_1} - x_{j_1}\right)^2 + \left(x_{i_2} - x_{j_2}\right)^2}.$$



$$L_2\left(A, B\right) = 2\sqrt{2}$$

代入 $p=1$ ： $L_1\left(x_i, x_j\right)$ 称为曼哈顿距离：



$$L_1\left(A, B\right) = 4$$

K 值的选择

使用交叉验证方法确定最合适的 K 值

Learn 5

10.23

1.4 模型评估和性能度量

Notation. 留出法 (*hold-out*)：

将原始数据集 D 分为两个互斥的子集 S,T，S 作为训练数据集，T 作为测试数据集：$D=S\cup T, S\cap T=\varnothing$

在划分任务时要尽量保证 S 和 T 中的样本类别比例相似

Example.

$$D\left(a,b\right)\to S\left(\lambda a,\lambda b\right)\cup T\left(\left(1-\lambda\right)a,\left(1-\lambda\right)b\right).$$

该过程称为分层采样法，其中 $\lambda\in[\frac{2}{3},\frac{4}{5}]$

使用 S 训练模型，T 进行模型测试，多次随机划分 a,b 在 S 和 T 内的内容，多次实验取测试结果平均值

Notation. 交叉验证法/k 折交叉验证 (*cross validation*/*k-fold cross validation*)：

$$D=D_1\cup D_2\cup\ldots\cup D_k \text{ 且} D_i\cap D_j=\varnothing\left(i\neq j\right).$$

此处 $\forall D_i$ 由 D 分层采样得到

每次实验使用 $k-1$ 个子集的并集训练，剩下的一个子集作为测试集：

$$S=\sum_{i=1}^{m-1}D_i+\sum_{i=m+1}^{k}D_i\quad T=D_m.$$

取不同的 m 值共可以得到 k 组“训练集-测试集”，得到 k 个结果，取 k 个结果的平均值

Example. 5 折交叉验证的数据划分：

$$\begin{array}{|ccccc|}\hline D_1&D_2&D_3&D_4&\boldsymbol{D}_5\\ D_1&D_2&D_3&\boldsymbol{D}_4&D_5\\ D_1&D_2&\boldsymbol{D}_3&D_4&D_5\\ D_1&\boldsymbol{D}_2&D_3&D_4&D_5\\ \boldsymbol{D}_1&D_2&D_3&D_4&D_5\\ \hline\end{array}\Rightarrow\begin{cases}\mathrm{Res}_1\\ \mathrm{Res}_2\\ \mathrm{Res}_3\\ \mathrm{Res}_4\\ \mathrm{Res}_5\end{cases}\xrightarrow{\mathrm{Avg}\cdot}\mathrm{Result}$$

Notation. 若样本量 m 等于子集数 k ，交叉验证法等同于留一法 (*leave one out*, LOO)

留一法的优点：训练结果更准确

缺点：样本量太大的时候消耗过多资源

1.5 模型性能度量

Notation. 错误率：

$$E = \frac{1}{m} N\left(f\left(x_i\right) \neq y_i\right).$$

准确率：

$$\mathrm{Acc} = \frac{1}{m} N\left(f\left(x_i\right) = y_i\right).$$

m 为样本总数，$N\left(f\left(x_i\right) = y\right)$ 表示符合特征 $f : x \to y$ 的样本数量

Learn 6

10.31

Notation. 二分类问题：

将一个样本分至两个类别的问题，如：鉴定邮件是否为垃圾邮件，预测某人是否会患上某种疾病等问题

对于二分类问题，真实结果有两种，使用模型预测也会产生两种结果，组合得到混淆矩阵：

$$\begin{bmatrix} \text{真阳性 (TP)} & \text{假阴性 (FN)} \\ \text{假阳性 (FP)} & \text{真阴性 (TN)} \end{bmatrix}.$$

其中：阳性/阴性为模型预测结果，真/假为真实结果

准确率 (Acc) 根据混淆矩阵的计算：

$$\mathrm{Acc} = \frac{\mathrm{TP} + \mathrm{TN}}{\mathrm{TP} + \mathrm{TN} + \mathrm{FP} + \mathrm{FN}}.$$

Notation. 马修斯相关系数 (Matthews Correlation Coefficient, MCC)：

MCC 比 Acc 更加全面（正负数据不平衡）

$$\mathrm{MCC} = \frac{\mathrm{TP} \times \mathrm{TN} - \mathrm{FP} \times \mathrm{FN}}{\sqrt{\left(\mathrm{TP} + \mathrm{FP}\right)\left(\mathrm{TP} + \mathrm{FN}\right)\left(\mathrm{TN} + \mathrm{FP}\right)\left(\mathrm{TN} + \mathrm{FN}\right)}} \in [-1, 1].$$

Notation. MCC 结果解读：

- $\mathrm{FP} = \mathrm{FN} = 0$ ： 无误判结果，代入得：$\mathrm{MCC} = 1$ ，表示模型完美
- $\mathrm{TP} = \mathrm{TN} = 0$ ： 全部误判，代入得：$\mathrm{MCC} = -1$ ，表示最差
- $\mathrm{TP} \times \mathrm{TN} = \mathrm{FP} \times \mathrm{FN}$ ，即 $\mathrm{MCC} = 0$ ，表示模型完全随机判断

当样本中阴性样本远少于阳性样本时，Acc 计算不能涉及到假阴性与假阳性而 MCC 可以

若第一个模型对阳性和阴性样本判断接近，而第二个模型对阳性样本表现极佳但对阴性样本表现极差，则 $\mathrm{MCC}_1 > \mathrm{MCC}_2$，而 Acc 可能接近

Notation. 查准率 $\boldsymbol{P}$ ，查全率 $\boldsymbol{R}$，$\boldsymbol{F}_1$ 度量：

- 查准率 (precision, $\boldsymbol{P}$) ： 又叫精确率

$$\begin{aligned}\boldsymbol{P} &= \frac{N_{\mathrm{TP}}}{N_{\mathrm{P}_p}} \\ &= \frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FP}}.\end{aligned}$$

Notation. ∘ 查全率 (recall, $\boldsymbol{R}$)：又叫召回率

$$\begin{aligned}\boldsymbol{R} &= \frac{N_{\mathrm{TP}}}{N_{\mathrm{P}_a}} \\ &= \frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FN}}.\end{aligned}$$

一般情况下：查全率和查准率相矛盾

Learn 7

11.01

Notation. 在模型下，对样本阳/阴性的预测结果为一个概率 $p \in [0,1]$ ，通过设定一个阈值 m 来区分由模型预测的结果；在该阈值下，计算查全率和查准率，绘制一个点；设定不同的阈值，将所有点连接，得到 $\boldsymbol{P}-\boldsymbol{R}$ 曲线

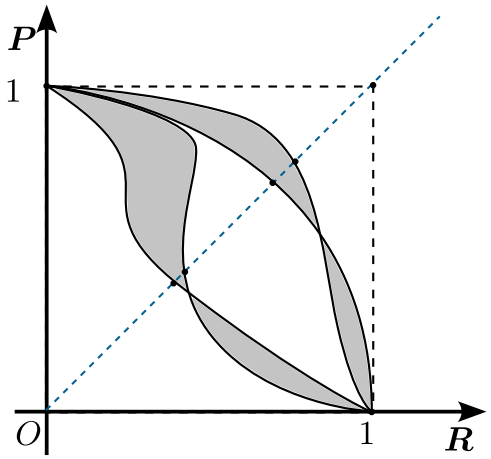


图 1: P-R 曲线

Learn 8

11.04

Notation. 当 $L^{(1)}_{\boldsymbol{P-R}}$ 完全包裹 $L^{(2)}_{\boldsymbol{P-R}}$ 时，代表模型 1 在各个阈值下查全率和查准率都较模型 2 更好，但当 $L^{(m,n)}_{\boldsymbol{P-R}}$ 相交时，无法通过曲线直接判断

缺点：未知曲线的面积不好求，无法判断相交曲线之间的性能关系，因此采用其他方法评估 $\boldsymbol{P-R}$ 值的关联

Notation. 平衡点 BEP：

作平衡线（一般为 $y=ax, a\in[0,+\infty]$），交曲线 $L^{(m,n)}_{\boldsymbol{P-R}}$ 于两个点，判断点的高低

缺点：太过简单

Notation. $\boldsymbol{F}_1$ 度量：

Learn 9

03.06

1.6 回归

简单来说，回归通过一系列**离散**的点，估计出一个**连续**的关系。回归的种类：

- 线性回归：使用线性函数 $f(x_i)=wx_i+b$
- 对数几率回归：本质是一种分类算法，通过引入 sigmoid 函数来确定哪些数据被激活
- 多项式回归：通过 $y=\beta_0+\beta_1x+\beta_2x^2+\ldots+\beta x^n+\varepsilon$ 来拟合数据
- 岭回归
- 套索回归
- …

sigmoid 函数：

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

图像如下：

这个函数对应的回归模型为：

$$P(y=1\mid\boldsymbol{x})=\sigma\left(\boldsymbol{\theta}^\top\boldsymbol{x}\right)=\frac{1}{1+e^{-\boldsymbol{\theta}^\top\boldsymbol{x}}}$$
$$P(y=0\mid\boldsymbol{x})=1-P(y=1\mid\boldsymbol{x}).$$

首先来看线性回归，有一个数据集合如下：

$$D=\{(\boldsymbol{x_1},y_1),(\boldsymbol{x_2},y_2),\ldots(\boldsymbol{x_n},y_n)\}.$$

其中集合中元素的第一个参数为一个 m 维向量，即：

$$\boldsymbol{x}_i=\begin{bmatrix}x_i^{(1)} & x_i^{(2)} & x_i^{(3)}\ldots x_i^{(m)}\end{bmatrix}^\top.$$

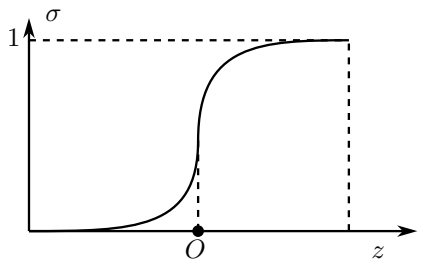


图 2: sigmoid 函数

首先考虑一维向量，即常数，这个计算就是常见的线性回归计算，使用最小二乘法，求出 w, b 并拟合为以下表达式：

$$f(x_i) = wx_i + b.$$

如果输入为多维向量，如三维向量，那么由于向量需要通过点乘得到常数，因此 w 成为一个向量，即：

$$f(\boldsymbol{x}_i) = \boldsymbol{w}^\top \boldsymbol{x}_i + b + \varepsilon_i.$$

这里有一个误差项，表示回归所不能解释的部分（常常看作一个 $\mathbb{E}_{\varepsilon_i \sim p} = 0$ 的高斯噪声），一般不考虑，因此也写为：

$$f(\boldsymbol{x}_i) \approx \boldsymbol{w}^\top \boldsymbol{x}_i + b.$$

归类到多维后，称 $\boldsymbol{w} \in \mathbb{R}^m$ 为模型的**权重向量**。回归中常用 MSE 来评价性能，求解 $\boldsymbol{w}$ 和 b 一般使用**最小二乘法**，过程如下：

证明. 回顾残差平方和：

$$\mathrm{RSS} = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2.$$

把 $\hat{y}_i$ 换为 $f(\boldsymbol{x}_i)$ 写出这个回归的均方误差表达：

$$E_{(\boldsymbol{w},b)} = \arg\min_{(\boldsymbol{w},b)} \sum_{i=1}^{m} (f(\boldsymbol{x}_i) - y_i)^2 = \arg\min_{(\boldsymbol{w},b)} \sum_{i=1}^{m} \left(\boldsymbol{w}^\top \boldsymbol{x}_i + b - y_i\right)^2 \tag{1}$$

这里 $\arg\min$ 的意思是通过取一组 $\boldsymbol{w}, b$ 的值使得表达式最小，严格定义上，这个式子所代表的过程就称为线性回归。求一个二元函数 $E_{(\boldsymbol{w},b)} = \sum_{i=1}^{m} \left(\boldsymbol{w}^\top \boldsymbol{x}_i + b - y_i\right)^2$ 的最小值，由于向量内积的性质 $\boldsymbol{w}^\top \boldsymbol{x}_i = \boldsymbol{x}_i^\top \boldsymbol{w}$ 也可以写为：

$$E = \sum_{i=1}^{m} \left(\boldsymbol{x}_i^\top \boldsymbol{w} + b - y_i\right)^2.$$

这个式子可以直接求偏导来计算最小值，但是为了机器运算方便，尽量转化为矩阵运算。平方和很容易让人想到 L^2 范数，或者 Frobenius 范数。观察可得，这里的平方和可以拆成 y 和 $\boldsymbol{x}_i^\top \boldsymbol{w} + b$

两部分。记住 $\boldsymbol{x}_i$ 和 $\boldsymbol{w}$ 都是列向量，这里为了统一输入，即输入的矩阵不含变量，给每一个 $\boldsymbol{x}_i$ 的末端加上一个 1，给 $\boldsymbol{w}$ 的末端加上一个 b ，这样做之后两个向量变成下列形式：

$$\widetilde{\boldsymbol{x}}_i = \begin{bmatrix} \boldsymbol{x}_i^\top & 1 \end{bmatrix}^\top = \begin{bmatrix} x_i^{(1)} \\ x_i^{(2)} \\ \vdots \\ x_i^{(n)} \\ 1 \end{bmatrix}$$

$$\widetilde{\boldsymbol{w}} = \begin{bmatrix} \boldsymbol{w}^\top & b \end{bmatrix}^\top = \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_n \\ b \end{bmatrix}.$$

将所有的 $\widetilde{\boldsymbol{x}}_i$ 横向合并，记为 $\boldsymbol{X}$ ，形如：

$$\boldsymbol{X} = \begin{bmatrix} \widetilde{\boldsymbol{x_1}}^\top \\ \widetilde{\boldsymbol{x_2}}^\top \\ \vdots \\ \widetilde{\boldsymbol{x_m}}^\top \end{bmatrix} = \begin{bmatrix} \boldsymbol{x_1}^\top & 1 \\ \boldsymbol{x_2}^\top & 1 \\ \vdots & \vdots \\ \boldsymbol{x_m}^\top & 1 \end{bmatrix} \quad \boldsymbol{X} \in \mathbb{R}^{m\times(n+1)}.$$

带回 $E_{(\boldsymbol{w},b)}$ 后可以把两个参数整合为一个，同时由于所有的 $\widetilde{\boldsymbol{x}}_i$ 在 $\boldsymbol{X}$ 中，计算发现：

$$E_{\widetilde{\boldsymbol{w}}} = (\boldsymbol{X}\widetilde{\boldsymbol{w}} - \boldsymbol{y})^2 = \|\boldsymbol{y} - \boldsymbol{X}\widetilde{\boldsymbol{w}}\|_2^2.$$

想要求一个函数的最小值还可以用梯度法，即 $\nabla f = 0$ 时 f 可能取得一个极值（在这里就是求导），对于 $E_{\widetilde{\boldsymbol{w}}}$，对 $\widetilde{\boldsymbol{w}}$ 求梯度：

$$\nabla_{\widetilde{\boldsymbol{w}}} E_{(\widetilde{\boldsymbol{w}})} = \frac{\partial E_{\widetilde{\boldsymbol{w}}}}{\partial \widetilde{\boldsymbol{w}}}.$$

这里有一些小细节，首先将 E 展开为内积形式：

$$\begin{aligned} E &= (\boldsymbol{y} - \boldsymbol{X}\widetilde{\boldsymbol{w}})^\top (\boldsymbol{y} - \boldsymbol{X}\widetilde{\boldsymbol{w}}) = \left(\boldsymbol{y}^\top - \widetilde{\boldsymbol{w}}^\top \boldsymbol{X}^\top\right)(\boldsymbol{y} - \boldsymbol{X}\widetilde{\boldsymbol{w}}) \\ &= \boldsymbol{y}^\top\boldsymbol{y} - 2\widetilde{\boldsymbol{w}}^\top\boldsymbol{X}^\top\boldsymbol{y} + \widetilde{\boldsymbol{w}}^\top\boldsymbol{X}^\top\boldsymbol{X}\widetilde{\boldsymbol{w}}. \end{aligned}$$

此处注意：$2\widetilde{\boldsymbol{w}}^\top\boldsymbol{X}^\top\boldsymbol{y}$ 是一个**标量**，因此对其求转置仍为其本身，即：

$$\begin{aligned} E &= \boldsymbol{y}^\top\boldsymbol{y} - 2\widetilde{\boldsymbol{w}}^\top\boldsymbol{X}^\top\boldsymbol{y} + \widetilde{\boldsymbol{w}}^\top\boldsymbol{X}^\top\boldsymbol{X}\widetilde{\boldsymbol{w}} \\ &= \boldsymbol{y}^\top\boldsymbol{y} - 2\boldsymbol{y}^\top\widetilde{\boldsymbol{w}}\boldsymbol{X} + \widetilde{\boldsymbol{w}}^\top\boldsymbol{X}^\top\boldsymbol{X}\widetilde{\boldsymbol{w}}. \end{aligned}$$

引入两个对矩阵求导的公式：

$$\frac{\partial}{\partial \boldsymbol{x}}\boldsymbol{x}^\top\boldsymbol{A} = \boldsymbol{A}^\top \tag{2}$$

$$\frac{\partial}{\partial \boldsymbol{x}} \boldsymbol{x}^{\top} \boldsymbol{A} \boldsymbol{x}=2 \boldsymbol{A} \boldsymbol{x}(\text{对称矩阵}) \tag{3}$$

对于第一个公式，令 $\boldsymbol{A}=\boldsymbol{X}^{\top} \boldsymbol{y}$ ，可以解得

$$\frac{\partial}{\partial \widetilde{\boldsymbol{w}}} 2 \widetilde{\boldsymbol{w}}^{\top} \boldsymbol{X}^{\top} \boldsymbol{y}=.$$

对于 □