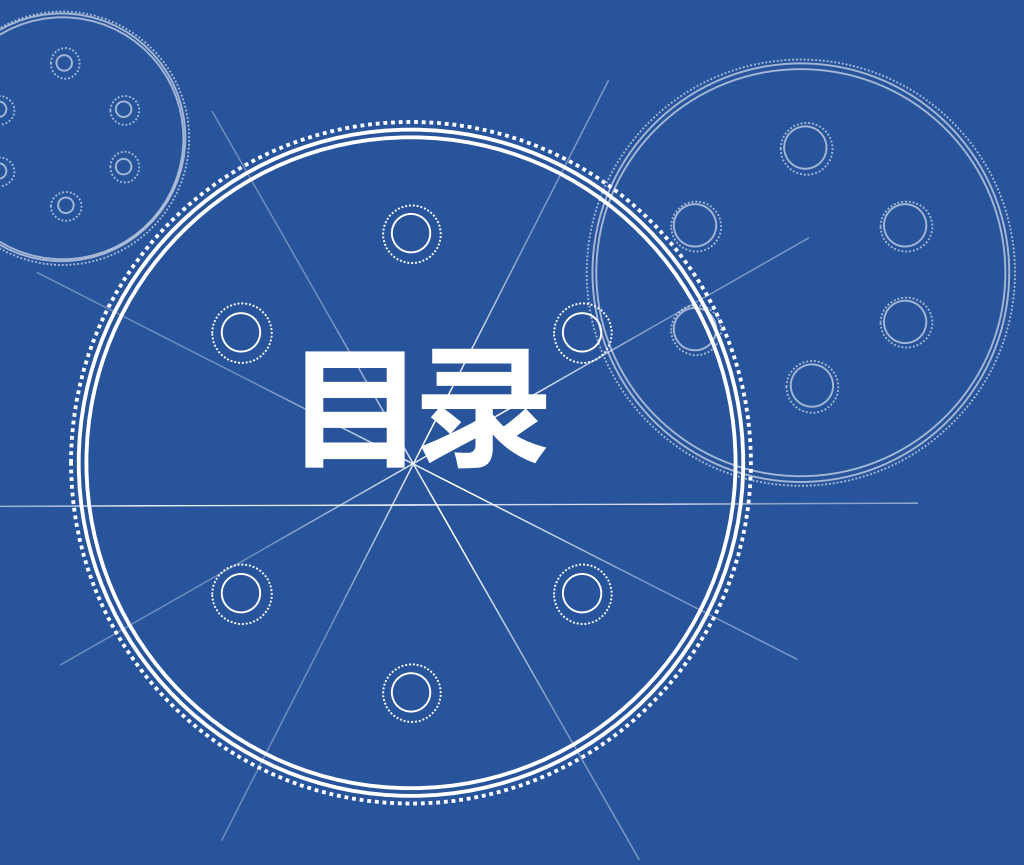




深度学习 — 基础

中国科学技术大学 | 大数据学院 | 连德富

dove.ustc@gmail.com

A decorative geometric pattern on the left side of the slide, featuring a large central circle with several smaller circles and lines radiating from the center, creating a star-like or web-like structure.

目录



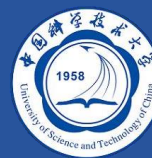
1. 线性代数

2. 概率与信息论

3. 数值计算与优化

4. 机器学习基础

标量与向量



- 标量对应一个数，用斜体小写字母表示

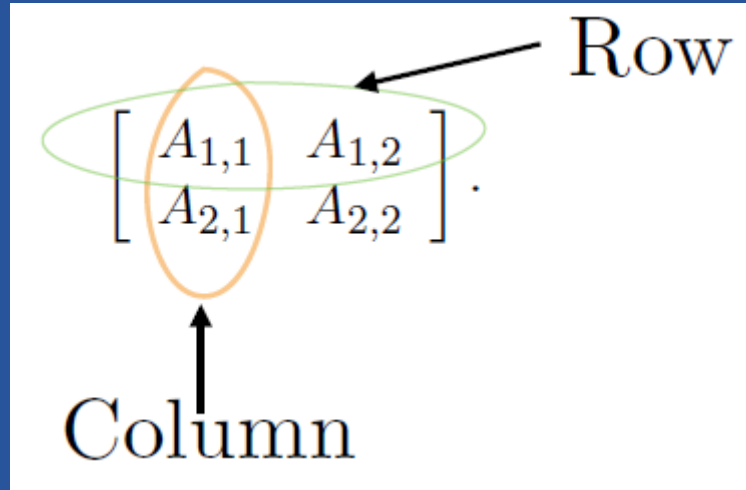
a, n, x

- 向量对应一维数组，用斜粗体小写字母表示

$$\boldsymbol{x} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} \in \mathbb{R}^n$$

矩阵与张量


- 矩阵对应二维数组，用大写的粗斜体表示 $A \in \mathbb{R}^{m \times n}$
- $A_{:,i}$ 表示第*i*列
- $A_{i,:}$ 表示第*i*行
- $A_{i,j}$ 表示第*i*行第*j*列元素



- 张量对应多维数组，用大写的粗体来表示
 - 0维张量，对应标量
 - 1维张量，对应向量
 - 2维张量，对应矩阵
 - 可以是3维或者更高维度

矩阵转置与矩阵乘积

- $(A^T)_{i,j} = A_{j,i}$

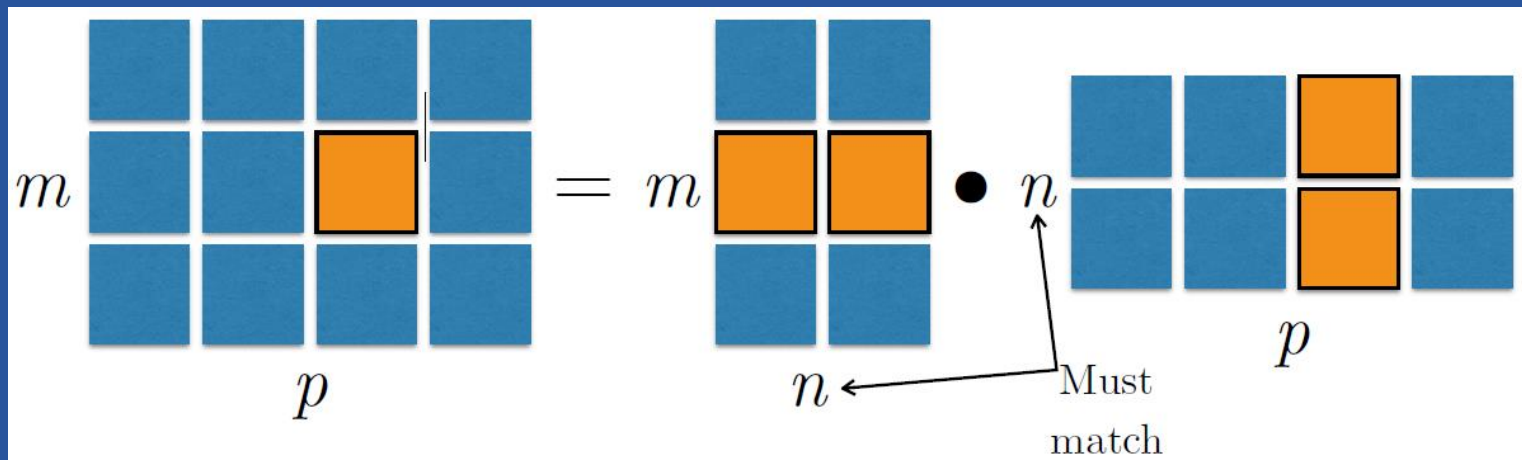

$$A = \begin{bmatrix} A_{1,1} & A_{1,2} \\ A_{2,1} & A_{2,2} \\ A_{3,1} & A_{3,2} \end{bmatrix} \Rightarrow A^T = \begin{bmatrix} A_{1,1} & A_{2,1} & A_{3,1} \\ A_{1,2} & A_{2,2} & A_{3,2} \end{bmatrix}$$

- $C = AB$ 等价于 $C_{ij} = \sum_k A_{i,k} B_{k,j}$

- 一般不满足交换律

- 但满足结合律和分配律

- $(AB)^T = B^T A^T$

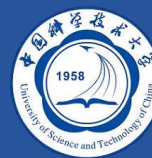


- Hadamard乘积 $C = A \odot B$ 满足 $C_{i,j} = A_{i,j} B_{i,j}$

单位矩阵、线性方程组、矩阵逆

- $\forall \mathbf{x} \in \mathbb{R}^n, I_n \mathbf{x} = \mathbf{x}$. 比如 $I_3 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$
- $A\mathbf{x} = \mathbf{b}$ 展开为
 - $A_{1,:}\mathbf{x} = b_1$
 - $A_{2,:}\mathbf{x} = b_2$
 - ...
 - $A_{m,:}\mathbf{x} = b_m$
- 矩阵逆 $A^{-1}A = I_n$
- 用矩阵逆解线性方程组
 - $A\mathbf{x} = \mathbf{b}$
 - $\mathbf{x} = I_n \mathbf{x} = A^{-1}A\mathbf{x} = A^{-1}\mathbf{b}$
- 不稳定，主要用于理论分析使用

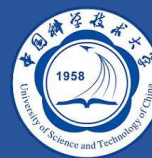
可逆性



7

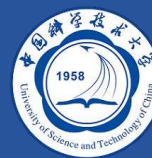
- 矩阵在如下情形下不可逆
 - 行数大于列数
 - 列数大于行数
 - 行或者列线性相关

线性相关与线性子空间



- 向量 $[v_1, v_2, \dots, v_n]$ 生成子空间为 $v = \sum_i x_i v_i$, 是这些向量线性组合之后能抵达点的集合
- $Ax = b$ 可以写成 $\sum_i A_{:,i} x_i = b$
 - Ax 表示 A 列向量的线性组合, 即为列向量的生成子空间, 称为矩阵 A 的值域
 - $Ax = b$ 是否有解, 就看 b 是否在 A 列向量的生成子空间中

解线性方程组



- 一个线性方程组 $Ax = b$ ($A \in \mathbb{R}^{m \times n}$) 可能
 - 没有解
 - b 不在 A 的列向量的生成子空间之中
 - 很多解：
 - b 在 A 的列向量的生成子空间之中
 - $m < n$
 - $m = n, \text{rank}(A) < n$
 - 有且仅有一个
 - b 在 A 的列向量的生成子空间之中
 - $m = n = \text{rank}(A)$

向量范数

- 函数 $f(x)$ 度量向量 x 的大小
- 类似于零点和向量对应空间中点的距离，满足
 - $f(x) \geq 0$ 而且 $f(x) = 0 \Leftrightarrow x = 0$
 - $f(x + y) \leq f(x) + f(y)$
 - $\forall \alpha \in \mathbb{R}, f(\alpha x) = |\alpha|f(x)$
- L^p norm
 - $\|x\|_p = (\sum_i |x_i|^p)^{\frac{1}{p}}$
- 最常见的2-范数， $p=2$
- L1范数， $p=1$: $\|x\|_1 = \sum_i |x_i|$
- 最大范数， p 等于无穷大： $\|x\|_\infty = \max_i |x_i|$

- 函数 $f(X)$ 衡量矩阵的大小，类似地满足三个条件
 - 正定、齐次、三角不等式
- p-诱导范数
 - $\|A\|_p = \sup_{x \neq 0} \frac{\|Ax\|_p}{\|x\|_p}$
 - $\|A\|_1 = \max_j \sum_i |A_{i,j}|$ 绝对值列和的最大值
 - $\|A\|_\infty = \max_i \sum_j |A_{i,j}|$ 绝对值行和的最大值
 - $\|A\|_2 = \sigma_{\max}(A)$
- 元范数
 - $\|A\|_F = \sqrt{\sum_i \sum_j |A_{i,j}|^2}$

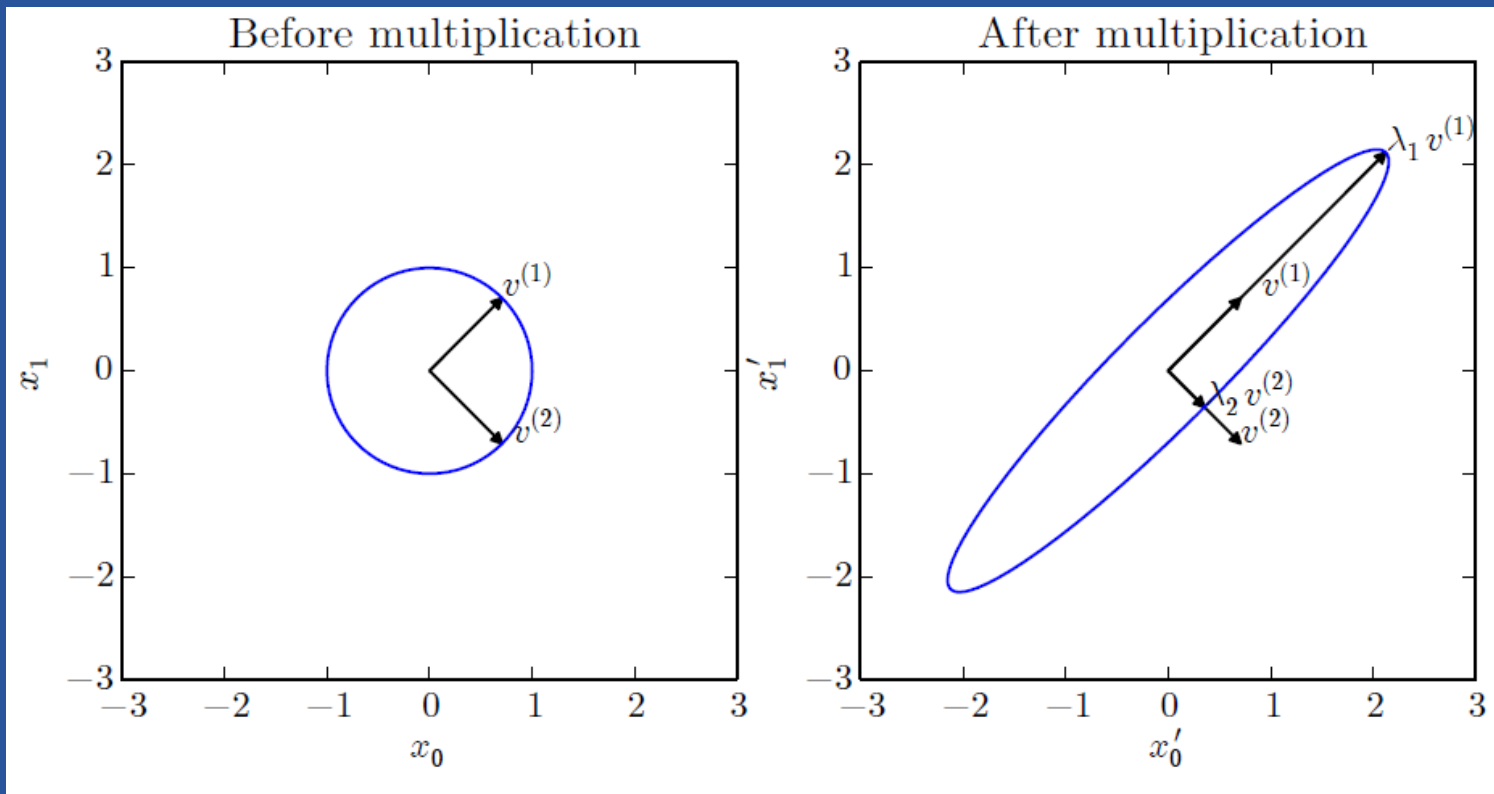
特殊矩阵和特殊向量

- 单位向量
 - $\|x\|_2 = 1$
- 对称矩阵
 - $A = A^T$
- 对角矩阵
 - $\forall i \neq j, D_{i,j} = 0$
- 正交矩阵
 - $A^T A = A A^T = I$
 - $A^{-1} = A$

特征值分解

- 特征向量与特征值分解
 - $Av = \lambda v$, v 特征向量 , λ 为特征值
 - $A(\alpha v) = \lambda(\alpha v)$, 考虑单位特征向量
- 可对角化矩阵的特征值分解为
 - $A = V \text{diag}(\lambda) V^{-1}$, λ 对应特征值
- 每一个实对称矩阵都有特征值分解
 - $A = Q \Lambda Q^T$
 - 分解不一定唯一
 - 分解唯一的条件是所有特征值唯一
- 矩阵是奇异的当且仅当含有零特征值
- 特征值为正数的矩阵为正定阵 , 非负的为半正定

特征值的影响



- 左边是单位向量集合
- 右边是A乘之后的集合

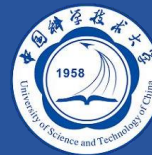
正交矩阵

- $A^T A = I = A A^T$
- $A^T A x = A^T \lambda x \Leftrightarrow A^T x = \frac{1}{\lambda} x$
- $x^T A^T x = \frac{1}{\lambda} x^T x = (Ax)^T x = \lambda x^T x$
- $\lambda^2 = 1 \Leftrightarrow |\lambda| = 1$
- 正交矩阵的特征值的绝对值为1，对应的是空间的旋转
- $A = \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix}$ 是二维空间逆时针旋转 θ 的矩阵

奇异值分解

- 奇异值分解类似于特征值分解，但是对任意矩阵都成立
- $A^T A v = \lambda v \Rightarrow v$ 对应 $A^T A$ 的特征值为 λ 特征向量
 - 令 $A v = \sigma u \Rightarrow A^T \sigma u = \lambda v$
 - $A A^T u = \frac{\lambda}{\sigma} A v = \lambda u \Rightarrow u$ 对应 $A A^T$ 的特征值为 λ 特征向量
 - $u^T A v = \sigma = v^T A^T u = \frac{\lambda}{\sigma} \Rightarrow \sigma^2 = \lambda$
- 将 $A v = \sigma u$ 写成矩阵形式为 $AV = U\Sigma$
- 假设 $k = \text{rank}(A)$, $\Sigma = \text{diag}([\sigma_1, \dots, \sigma_k, 0, \dots, 0])$ $\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_k \geq 0$
- 由于 V 是正交矩阵，所以 $A = AVV^T = U\Sigma V^T$
 - U 的列向量为左奇异向量， V 的列向量为右奇异向量

Moore-Penrose 伪逆



- $A^+ = \lim_{\alpha \rightarrow 0} (A^T A + \alpha I)^{-1} A^T$
- 可以通过如下方式计算，假设奇异值分解 $A = U \Sigma V^T$
 - $A^+ = V \Sigma^+ U^T$
 - 对角矩阵 Σ^+ 是 Σ 中非0元素取倒数再转置
 - 比如 $\Sigma = \begin{bmatrix} \sigma_1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$ ，那么 $\Sigma^+ = \begin{bmatrix} 1/\sigma_1 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}$

Moore-Penrose 伪逆



18

- 假设方程组 $Ax = b$
- 那么 $x = A^+b$ 可以概括无解、有一个解、有多个解的情形
 - 唯一解, $A^+ = A^{-1}$
 - 无解, $A^+b = \arg \min_x \|Ax - b\|^2$
 - 有多个解, $\|A^+b\| \leq \|x\|$, 满足 $Ax = b$

矩阵迹



- $\text{trace}(A) = \sum_i A_{i,i}$
- $\|A\|_F = \sqrt{\text{trace}(A^T A)}$
- $\text{trace}(A^T) = \text{trace}(A)$
- $\text{trace}(AB) = \text{trace}(BA)$

行列式

- $\det A = \prod_i \lambda_i$ 其中 λ_i 为 A 的特征值
- 行列式的绝对值衡量矩阵参与矩阵乘法后空间的缩放
 - 如果行列式是0，那么空间至少沿着某一维完全收缩了，使其失去了所有的体积。
 - 如果行列式是1，那么这个转换保持空间体积不变。

基于线性代数推导主成分分析

- 假设在 \mathbb{R}^n 空间中有个 m 个点 $\{x^{(1)}, \dots, x^{(m)}\}$ ，对这些点有损压缩
- 用低维来表示，即 $\forall x^{(i)} \in \mathbb{R}^n$ ，假设用编码向量 $c^{(i)} \in \mathbb{R}^l$ 表示， $l < n$
- 希望找到
 - 编码函数满足 $f(x) = c$;
 - 解码函数满足 $x \approx g(f(x))$
- 假设解码函数为 $g(c) = Dc$, $D \in \mathbb{R}^{n \times l}$ 是编码矩阵，应满足
 - $\min_{D, c^{(i)}} \sum_i \|x^{(i)} - Dc^{(i)}\|^2$ ，并简单起见假设 $D^T D = I$
 - 固定 D ，最优解 $c^{(i)} = D^T x^{(i)}$
 - $\min_D \sum_i \|x^{(i)} - DD^T x^{(i)}\|^2$ ， $D^T D = I$
 - 写成矩阵形式 $\min_D \|X - XDD^T\|_F^2$ ， $D^T D = I$

基于线性代数推导主成分分析

- $\min_D \|X - XDD^T\|_F^2, D^T D = I$
- 等价于 $\max_D \text{trace}(D^T X^T X D), D^T D = I$
- D 是 $X^T X$ 前 l 个特征值对应的特征向量
- $I - DD^T$ 是投影矩阵
 - $(I - DD^T)^2 = I - DD^T$
 - 考虑任意 $x \in \mathbb{R}^n, x = (I - DD^T)x + DD^T x$

A decorative geometric pattern on the left side of the slide, featuring several overlapping circles and lines, with the word '目录' (Table of Contents) in the center.

目录

1. 线性代数



2. 概率与信息论

3. 数值计算与优化

4. 机器学习基础

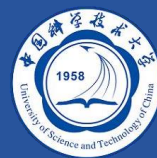
为什么要使用概率

- 机器学习通常必须处理不确定量，有时也可能需要处理随机量
- 不确定性来源
 - 被建模系统内在随机性
 - 纸牌游戏中假设纸牌被混洗成随机序列
 - 不完全观测
 - 如果明天下雨，我要带伞，否则不带。
 - 明天是否下雨是未观测的，所以我带不带伞就是不确定
 - 不完全建模
 - 网格划分空间建模
- 使用一些简单而不确定的规则要比复杂而确定的规则更为实用
 - 多数鸟儿都会飞
 - 除了那些还没学会飞翔的幼鸟，因为生病或是受伤而失去了飞翔能力的鸟以外，鸟儿会飞

两派相争

- 频率派：直接与事件发生的频率相联系
 - 事件往往是可以重复的
- 概率派：用概率来表示一种置信度
 - 医生诊断病人，用概率表示可能性
 - 1表示非常肯定患病
 - 0表示非常肯定不患病

随机变量与概率分布



- 随机变量是可以随机地去不同值的变量，用无格式表示 X ， x
 - 离散随机变量
 - 连续随机变量
- 概率分布用来描述随机变量或一簇随机变量在每一个可能取到的状态的可能性大小

概率质量函数

- $x \sim P(x)$, $P(x)$ 表示 $X = x$ 的概率 , 而且满足
 - P 的定义域必须是 x 所有可能状态的集合
 - $0 \leq P(x) \leq 1$
 - $\sum_x P(x) = 1$
- 均匀分布
 - $P(x = x_i) = \frac{1}{k}$

概率密度函数

- $x \sim p(x)$
 - p 的定义域必须是 x 所有可能状态的集合
 - $p(x) \geq 0$
 - $\int p(x)dx = 1$
- 均匀分布
 - $u(x; a, b) = \frac{1}{b-a}$

边缘概率与条件概率

- 边缘概率

- 离散随机变量

- $P(x = x) = \sum_y P(x = x, y = y)$

- 连续随机变量

- $p(x) = \int p(x, y) dy$

- 条件概率

- $P(y = y | x = x) = \frac{P(y=y, x=x)}{P(x=x)}$

- 只有在 $P(x = x) > 0$ 才有意义

- 链式法则

- $P(x^{(1)}, \dots, x^{(n)}) = P(x^{(1)}) \prod_{i=2}^n P(x^{(i)} | x^{(1)}, \dots, x^{(i-1)})$

独立性与条件独立性

- 独立性

- 对于任意的 x, y $P(x = x, y = y) = P(x = x)P(y = y)$

- 条件独立性

- $P(x = x, y = y | z = z) = P(x = x | z = z)P(y = y | z = z)$

- $\mathbb{E}_{X \sim P}[f(x)] = \sum_x P(x)f(x)$
- $\mathbb{E}_{X \sim p}[f(x)] = \int p(x)f(x)dx$
- 期望的线性性
 - $\mathbb{E}_X[\alpha f(x) + \beta g(x)] = \alpha \mathbb{E}_X[f(x)] + \beta \mathbb{E}_X[g(x)]$

方差与协方差

- $\text{Var}(f(x)) = \mathbb{E}[(f(x) - \mathbb{E}[f(x)])^2]$
- $\text{Cov}(f(x), g(y)) = \mathbb{E}[(f(x) - \mathbb{E}[f(x)])(g(y) - \mathbb{E}[g(y)])]$
- 协方差矩阵
 - $\text{Cov}(\mathbf{x})_{i,j} = \text{Cov}(x_i, x_j)$

- 伯努利分布：单个二值随机变量的分布
 - $P(x = 1) = \phi$
 - $P(x = 0) = 1 - \phi$
 - $P(x = x) = \phi^x(1 - \phi)^{1-x}$
 - $\mathbb{E}_x[x] = \phi$
 - $\text{Var}_x(x) = \phi(1 - \phi)$
- Multinoulli分布：k个不同状态的离散型随机变量的分布
 - 由向量 $p \in [0,1]^{k-1}$ 参数化， p_i 表示第i个状态的概率
 - 最后一个状态的概率通过 $1 - \mathbf{1}^T p$ 给出

常用概率分布

- 高斯分布、正态分布

- $\mathcal{N}(x; \mu, \sigma^2) = \sqrt{\frac{1}{2\pi\sigma^2}} \exp\left(-\frac{1}{2\sigma^2}(x - \mu)^2\right)$

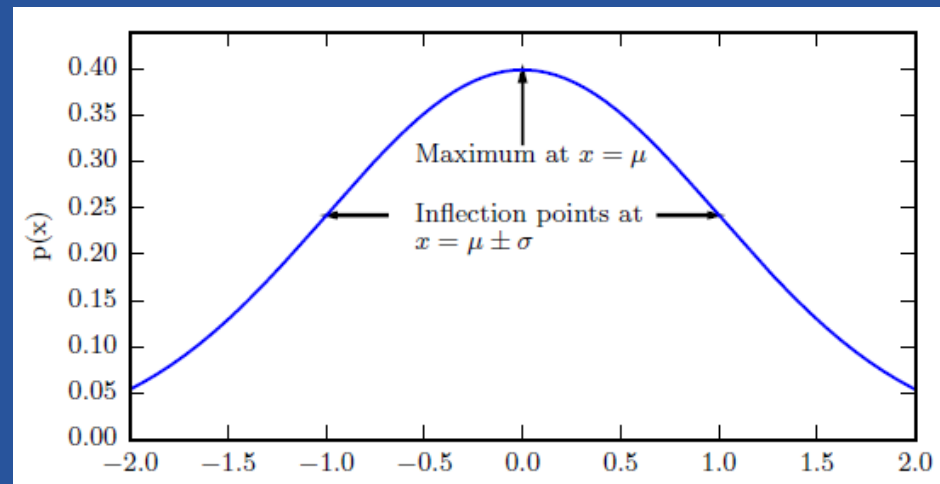
- 均值 μ 标准差 σ

- $\mathcal{N}(x; \mu, \beta^{-1}) = \sqrt{\frac{\beta}{2\pi}} \exp\left(-\frac{\beta}{2}(x - \mu)^2\right)$

- 多维高斯分布

- $\mathcal{N}(\mathbf{x}; \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \sqrt{\frac{1}{(2\pi)^n \det(\boldsymbol{\Sigma})}} \exp\left(-\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1}(\mathbf{x} - \boldsymbol{\mu})\right)$

- $\mathcal{N}(\mathbf{x}; \boldsymbol{\mu}, \boldsymbol{\beta}^{-1}) = \sqrt{\frac{\det(\boldsymbol{\beta})}{(2\pi)^n}} \exp\left(-\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu})^T \boldsymbol{\beta}(\mathbf{x} - \boldsymbol{\mu})\right)$



- 指数分布

- $p(x; \lambda) = \lambda 1_{x \geq 0} \exp(-\lambda x),$
- $1_{x \geq 0}$ 为示性函数

- 拉普拉斯分布

- $Laplace(x; \mu, \gamma) = \frac{1}{2\gamma} \exp\left(-\frac{|x-\mu|}{\gamma}\right)$

- 狄拉克函数

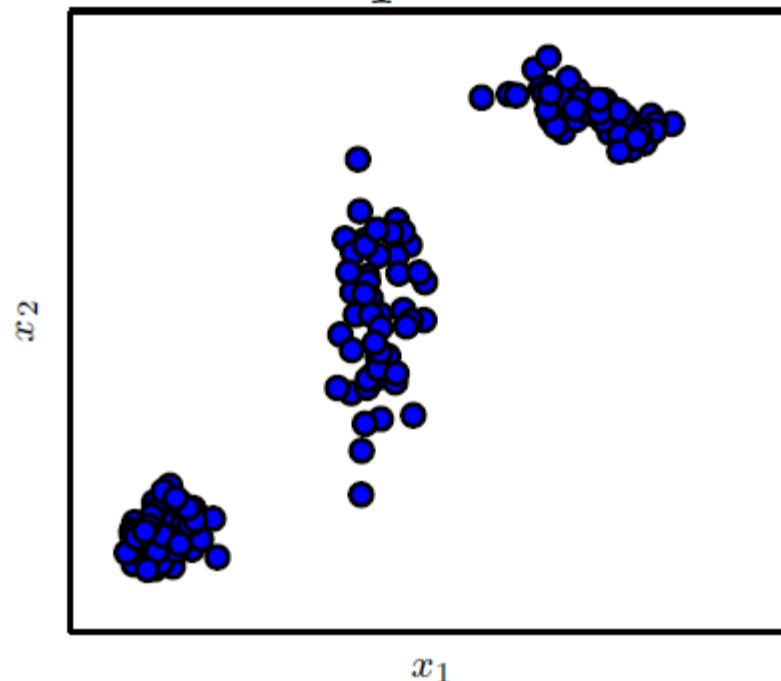
- $p(x) = \delta(x - \mu)$

- 经验分布

- $\hat{p}(x) = \frac{1}{m} \sum_i \delta(x - x^{(i)})$

- $P(x) = \sum_i P(c = i)P(x|c = i)$
 - 随机变量 c 称为隐变量，无法直接观测
 - $P(c = i)$ 先验概率，观测数据之前
 - $P(c = i|x)$ 后验概率，观测数据之后
- 高斯混合模型
 - $P(x|c = i)$ 高斯分布
 - 概率密度的万能近似器

Gaussian mixture
with three
components



常用函数的有用性质

- Sigmoid函数

- $\sigma(x) = \frac{1}{1+\exp(-x)} \in (0,1)$

- 可以用来产生伯努利分布的 ϕ 参数

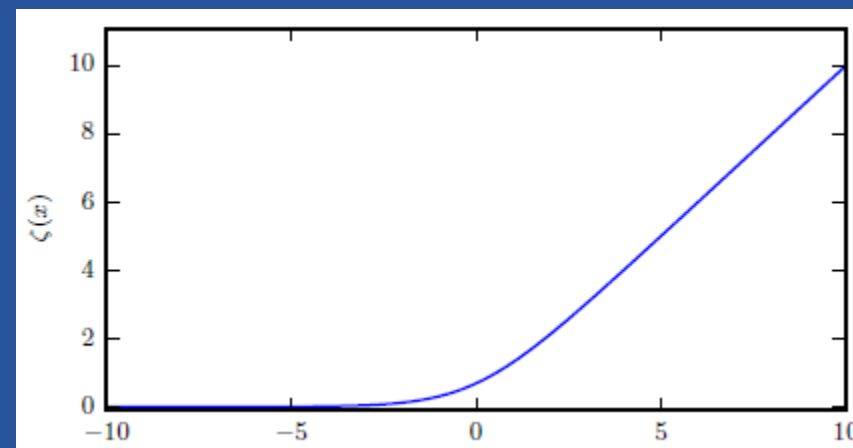
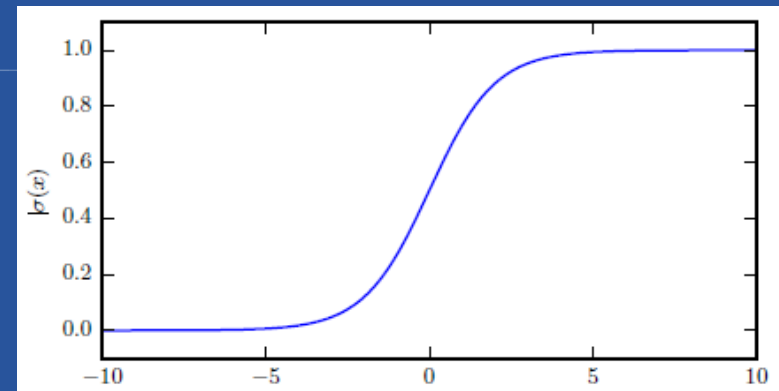
- 在 x 绝对值很大时候会出现饱和现象，对输入的微小改变变得不敏感

- Softplus函数

- $\zeta(x) = \log(1 + \exp(x)) \in (0, \infty)$

- 可以用来产生正态分布的 β 和 σ 参数

- $x^+ = \max(0, x)$ 的平滑版本



常用函数的有用性质

- $\sigma(x)$ 和 $\varsigma(x)$ 的有关性质
 - $\sigma'(x) = \sigma(x)(1 - \sigma(x))$
 - $1 - \sigma(x) = \sigma(-x)$
 - $\log \sigma(x) = -\varsigma(-x)$
 - $\varsigma'(x) = \sigma(x)$
 - $\forall x \in (0,1), \sigma^{-1}(x) = \log\left(\frac{x}{1-x}\right)$
 - $\forall x > 0, \varsigma^{-1}(x) = \log(\exp(x) - 1)$
 - $\varsigma(x) = \int_{-\infty}^x \sigma(y) dy$
 - $\varsigma(x) - \varsigma(-x) = x$

- 贝叶斯规则

- $P(\mathbf{x}|\mathbf{y}) = \frac{P(\mathbf{x})P(\mathbf{y}|\mathbf{x})}{P(\mathbf{y})}$

- 变量替换

- 已知 $p_y(y)$, 假设 $y = g(x)$

- $p_x(x) = p_y(g(x)) \left| \det \left(\frac{\partial g(x)}{\partial x} \right) \right|$

- 设 $\mathcal{N} \left(\begin{bmatrix} x \\ y \end{bmatrix}; \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \right) = \frac{1}{\sqrt{4\pi^2}} \exp \left(-\frac{1}{2} (x^2 + y^2) \right)$

- 利用坐标变换法, 设 $x = r \cos \theta, y = r \sin \theta$

- $p(r, \theta) = \frac{1}{\sqrt{4\pi^2}} \exp \left(-\frac{1}{2} r^2 \right) r$

- 自信息

- $I(x) = -\log_2 P(x)$

- 香农熵

- $H(x) = \mathbb{E}_{x \sim P}[I(x)] = -\mathbb{E}_{x \sim P} \log P(x)$

- 对依据概率分布 P 生成的符号进行编码所需的比特数在平均意义上的下界

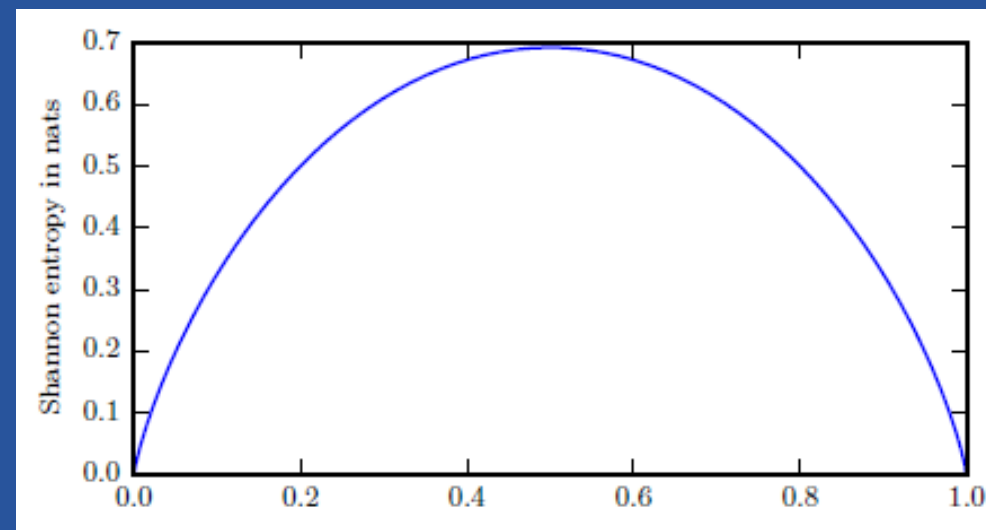
- 接近确定性的分布熵较低

- 接近均匀分布的概率分布熵较高

- 右图表示二值随机变量的香农熵

- x轴表示概率

- y轴表示熵



- KL散度：衡量两个分布的差异

- $D_{KL}(P||Q) = \mathbb{E}_{x \sim P} \left[\log \frac{P(x)}{Q(x)} \right] = \mathbb{E}_{x \sim P} [\log P(x) - \log Q(x)]$

- 当我们使用一种被设计成能够使得概率分布 Q 产生的消息的长度最小的编码，发送包含由概率分布 P 产生的符号的消息时，所需要的额外信息量

- 非负, $P=Q$ 时为零

- 不对称，但理论上最小值均当 $P=Q$

- 实际分布复杂，且未知

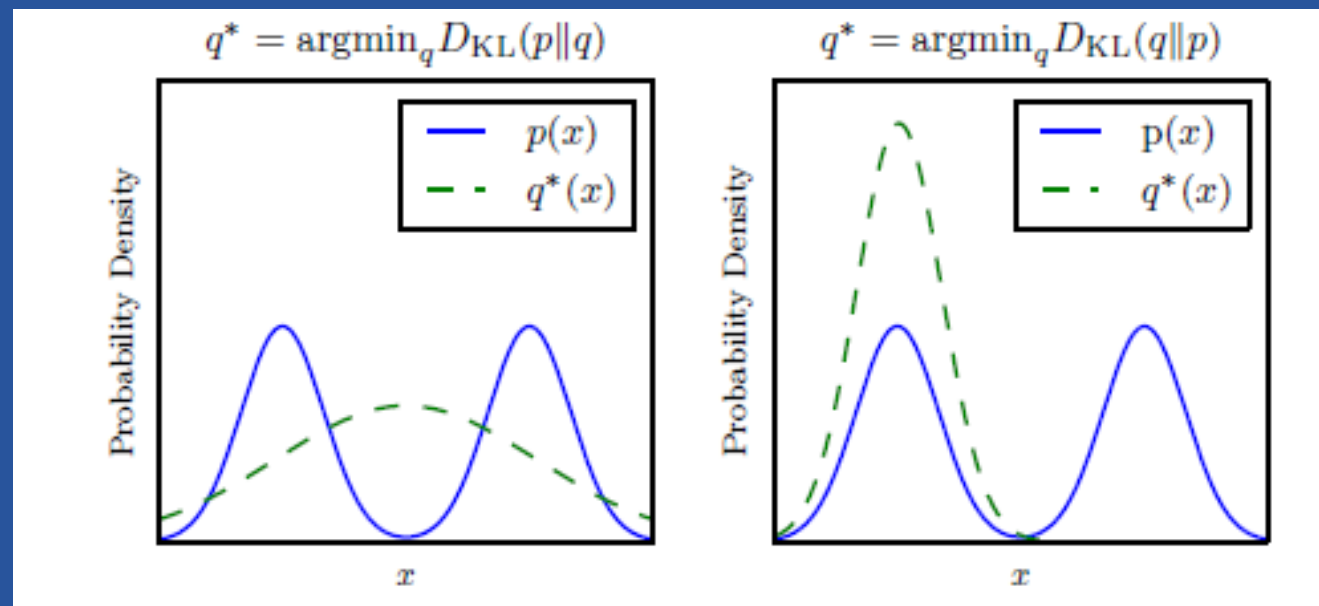
- 他们的最小值存在差异

- 第一个图

- Q 使得 P 高的地方高

- 第二个图

- Q 使得 P 低的地方低

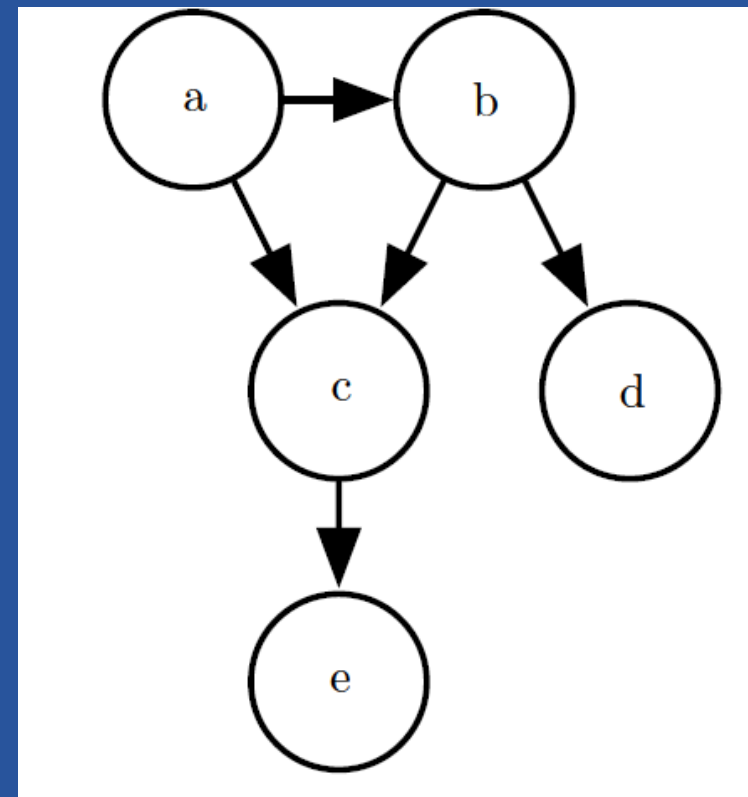


- 交叉熵 $H(P, Q) = -\mathbb{E}_{x \sim P} \log Q(x)$
- 针对Q优化交叉熵和KL散度是等价的

- 有向图图模型

- 模型使用带有有向边的图，它们用条件概率分布来表示分解
- $p(x) = \prod_i p(x_i | Pa(x_i))$
- 可以极大减少分布的参数

- $p(a, b, c, d, e) = p(a)p(b|a)p(c|a, b)p(d|b)p(e|c)$

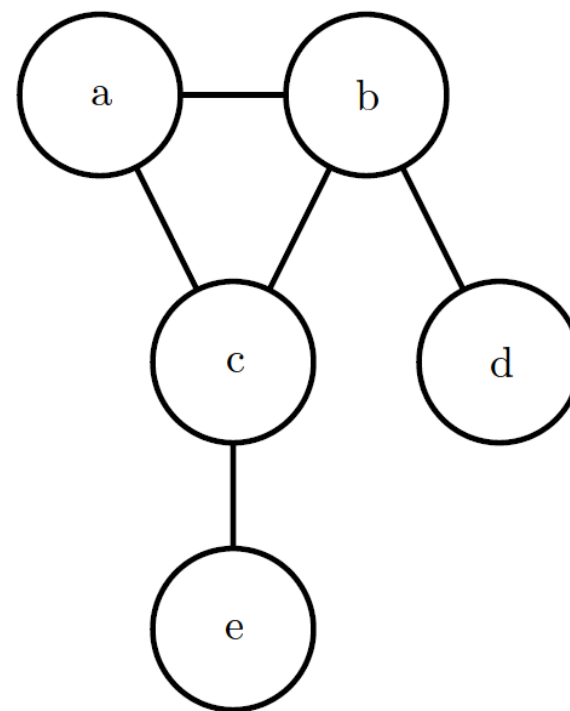


- 无向图图模型

- 使用带有无向边的图，它们将分解表示成一组函数
- 任何满足两两之间有边连接的顶点的集合被称为团
- 每个团都伴随一个因子（非负）
- 随机变量的联合概率与所有这些因子的乘积成比例
- $p(\mathbf{x}) = \frac{1}{Z} \prod_i \phi^{(i)}(C^{(i)})$, 其中 $C^{(i)}$ 为团

- 右图的概率分布

- $p(a, b, c, d, e) = \frac{1}{Z} \phi^{(1)}(a, b, c) \phi^{(2)}(b, d) \phi^{(3)}(c, e)$



A decorative geometric pattern on the left side of the slide, featuring a large central circle with several smaller circles and lines radiating from its center, creating a star-like or web-like structure.

目录

1. 线性代数

2. 概率与信息论



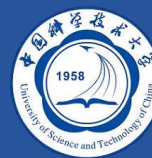
3. 数值计算与优化

4. 机器学习基础

上溢和下溢

- 数值及计算机中，一般用float32或者float64表示实数
- 实数 x 被舍入为 $x + \delta$, δ 为很小的数
- 下溢
 - 很小的数被舍入为 0
- 上溢
 - 很大的数被替换为 inf
- `>>> a = np.array([0., 1e-8]).astype('float32')`
- `>>> a.argmax()`
- `>>> (a+1).argmax()`

上溢和下溢



- $\exp(x)$ 在 x 很大的时候上溢
 - 在 $x = 89$ 上溢
 - 在 x 很大时, 该函数无法使用
- $\text{softmax}(\mathbf{x})_i = \frac{\exp(x_i)}{\sum_j \exp(x_j)}$ 用于预测与Multinoulli 分布相关联的概率
 - 通过 $\text{softmax}(\mathbf{z})$ 来解决, 其中 $\mathbf{z} = \mathbf{x} - \max_i x_i$
- 分子下溢时, 导致 $\log \text{softmax}$ 产生 $-\infty$
 - 利用类似技术实现单独函数, 通过数值稳定方式计算
 - $\log \sum_i \exp(x_i) = m + \log \sum_i \exp(x_i - m)$
- 底层库一般支持数值稳定的函数

上溢和下溢

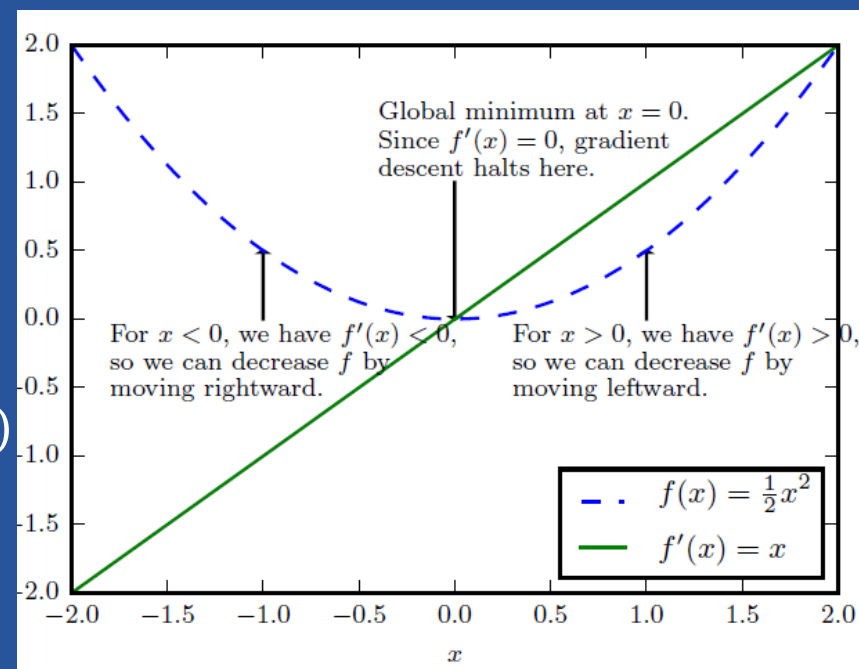
- $\log \sigma(x)$ 和 $x - \zeta(x)$ 在数学上和工程上的差别
- 数学上等价 $\log \sigma(x) = x - \zeta(x)$
- 工程上不等价
 - 考虑 $x = -710$, $\sigma(x) \rightarrow 0$, $\log \sigma(x)$ 是 \inf
 - $\zeta(x) \rightarrow 0$, $x - \zeta(x) \rightarrow -710$

- 如果发现了（NaN, Inf, 或者一些非常大的数），可以考虑如下函数
 - log
 - exp
 - sqrt
 - division

- 条件数表征函数对于输入的微小变化而变化的快慢程度
- 考虑 $f(x) = A^{-1}x$, 当 $A \in \mathbb{R}^{n \times n}$ 有特征值分解时, 其条件数
$$\max_{i,j} \left| \frac{\lambda_i}{\lambda_j} \right|$$
 - 为最大特征值和最小特征值的模之比
- 当该数很大时, 矩阵求逆对输入的误差特别敏感
- 这种敏感性是矩阵本有的特性, 但是实现时, 会和数值误差复合

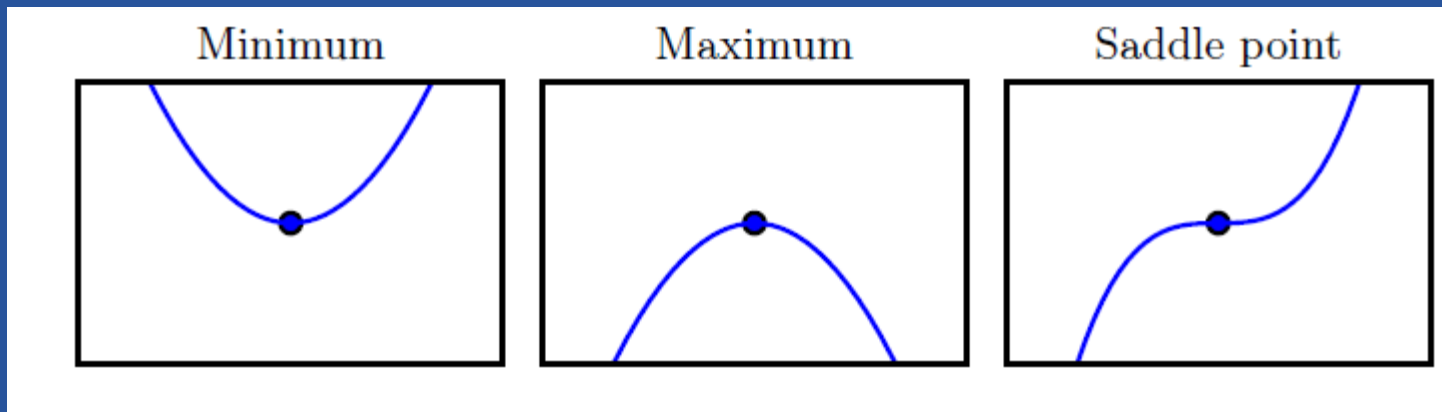
基于梯度的优化方法（1维）

- $\min f(x)$
 - $f(x)$ 称为 代价函数、损失函数、误差函数
 - $\max f(x)$ 等价于 $\min -f(x)$
 - $x^* = \arg \min f(x)$
- $f(x + \epsilon) \approx f(x) + \epsilon f'(x)$
 - 表明如何缩放输入的小变化才能在输出获得相应的变化
 - $f(x - \epsilon \text{sign}(f'(x))) \approx f(x) - \epsilon \text{sign}(f'(x))f'(x) < f(x)$
 - 将 x 往导数的反方向移动一小步来减小 $f(x)$
 - 梯度下降



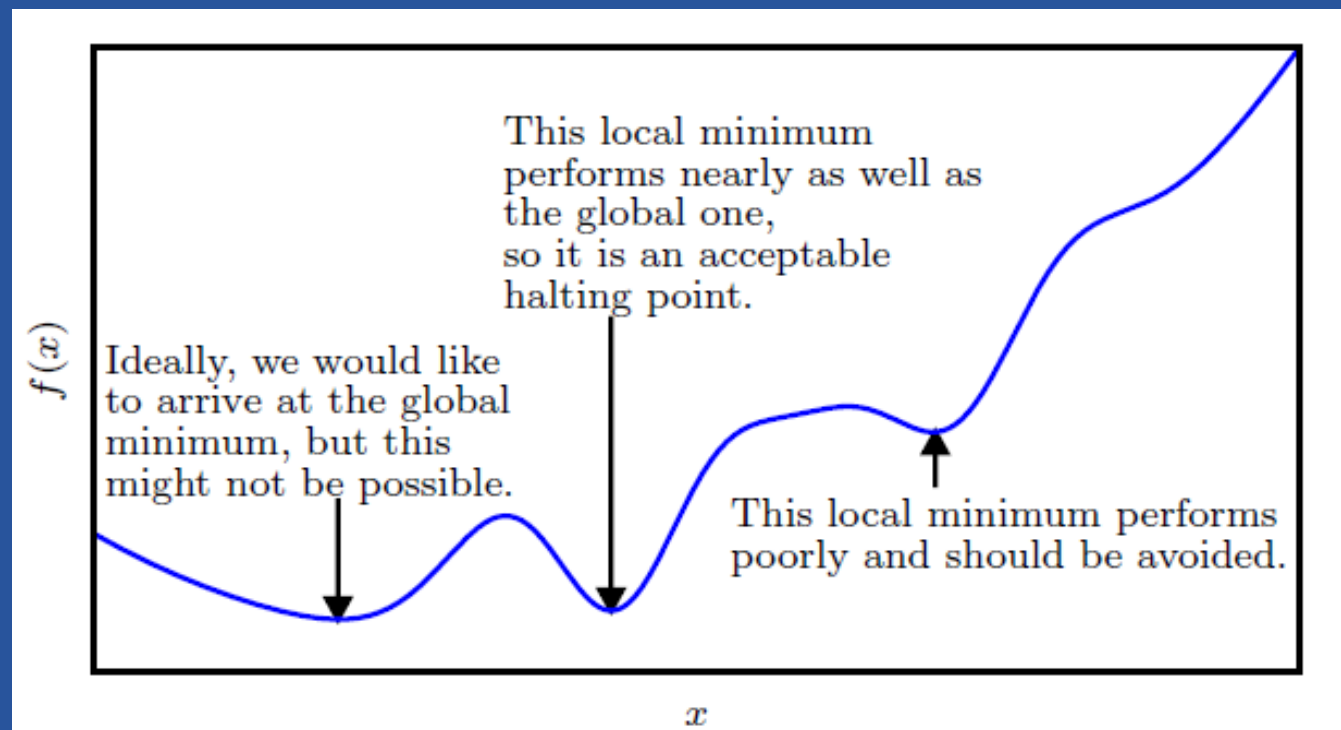
基于梯度的优化方法（1维）

- 临界点 (critical point)、驻点 (stationary point)
 - $f'(x) = 0$
 - 可能是局部最小点、局部最大点、鞍点
- 局部最小点 x
 - 对于 x 的 ϵ 邻域上任意的 c , $|x - c| < \epsilon$, $f(x) < f(c)$
- 局部最大点 x
 - 对于 x 的 ϵ 邻域上任意的 c , $|x - c| < \epsilon$, $f(x) > f(c)$



基于梯度的优化方法（1维）

- 全局最小点
 - $f(x)$ 取得最小值的点
- 深度学习中优化的函数
 - 可能有1个全局最小点
 - 存在多个全局最小点
 - 很多局部极小点
 - 很多鞍点



基于梯度的优化方法（多维）

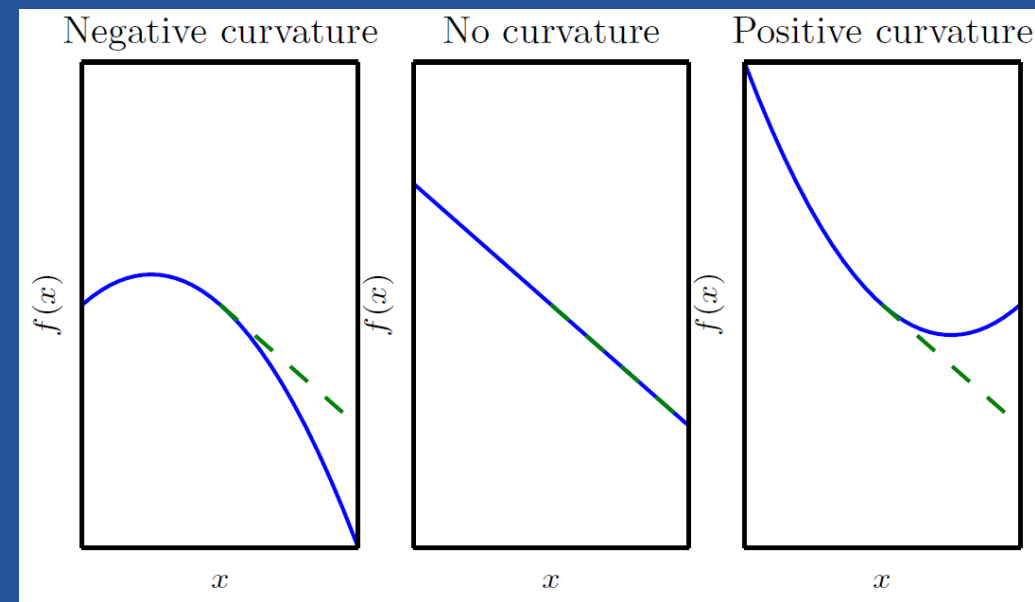
- 梯度： $(\nabla_x f(x))_i = \frac{\partial f(x)}{\partial x_i}$
- 临界点 $\nabla_x f(x) = \mathbf{0}$
- 在 u 方向上的方向导数
 - $\frac{\partial f(x+\alpha u)}{\partial \alpha} \big|_{\alpha=0} = u^T \nabla_x f(x)$
- $f(x + \alpha u) \approx f(x) + \alpha u^T \nabla_x f(x)$
 - 最速下降方向： $\min_{u, u^T u=1} u^T \nabla_x f(x)$
 - $u^* = -\frac{\nabla_x f(x)}{\|\nabla_x f(x)\|_2}$ ，沿着负梯度方向
- 梯度下降法： $x' = x - \epsilon \nabla_x f(x)$
 - $\epsilon > 0$ 为学习率

基于梯度的优化方法（多维）

- 学习率设置
 - 小常数
 - 线搜索 $\min_{\epsilon} f(x - \epsilon \nabla_x f(x))$
- 梯度下降推广到离散空间 — 爬山法
- 雅克比矩阵
 - $f: \mathbb{R}^m \rightarrow \mathbb{R}^n$
 - f 的雅克比矩阵为 $J \in \mathbb{R}^{n \times m}$ 定义为 $J_{i,j} = \frac{\partial}{\partial x_j} f(x)_i$

二阶导数

- 设 $f: \mathbb{R}^m \rightarrow R$
 - 二阶导数用 H 矩阵表示, 其中 $H_{i,j} = \frac{\partial^2}{\partial x_i \partial x_j} f(\mathbf{x})$
 - H 矩阵称为 Hessian 矩阵, 是对称矩阵
- 当 f 是一维函数时
 - 二阶导数是对曲率的衡量
 - 二阶导数指示梯度下降是否产生预期改善
 - 假设沿着梯度方向下降 ϵ (梯度为1)
 - 曲率=0, 损失下降 ϵ
 - 曲率>0, 损失下降小于 ϵ
 - 曲率<0, 损失下降大于 ϵ



二阶导数

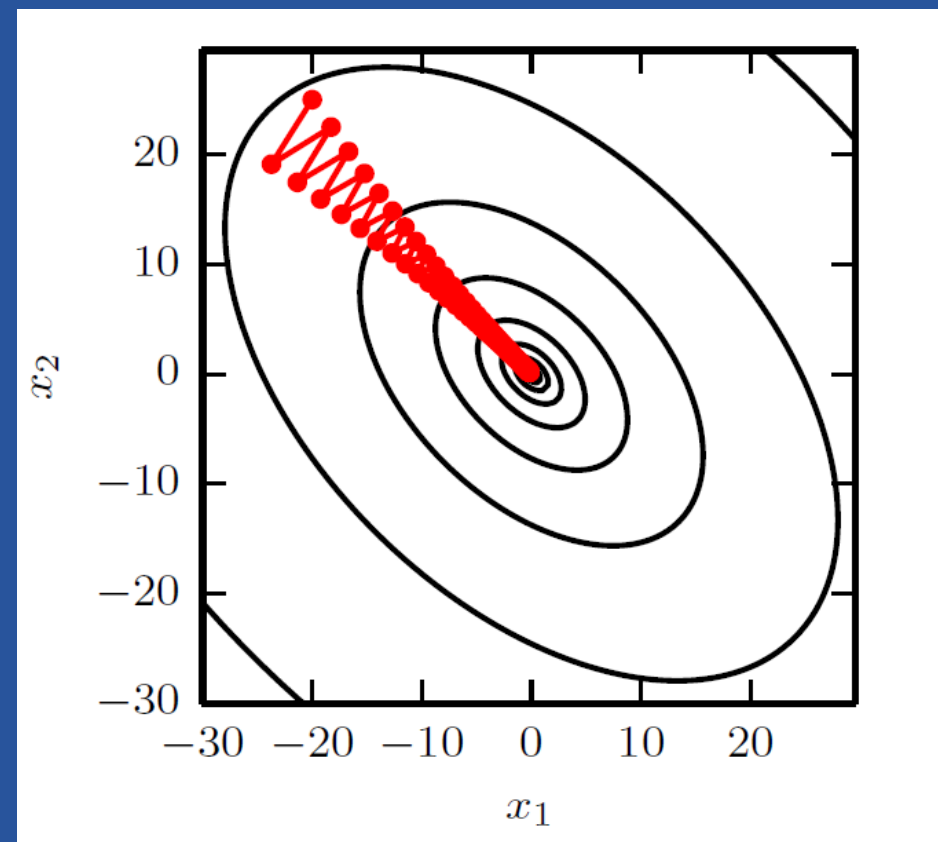
- Hessian 矩阵 H 能被特征值分解 , $H = V\Lambda V^T = \sum_i \lambda_i v_i v_i^T$
- $f(x + \alpha\mu)$ 在 μ 上的二阶导数为 $\mu^T H \mu$
 - $\mu^T H \mu = \sum_i \lambda_i (\mu^T v_i)^2$
 - 如果 μ 是 H 的特征向量 , 那么二阶导数为 H 的特征值
 - 如果 μ 不是 H 特征向量 , 那么二阶导数为 H 特征值的加权平均
- $f(x - \epsilon g) \approx f(x) - \epsilon g^T g + \frac{1}{2} \epsilon^2 g^T H g$
 - $g^T H g \leq 0$, ϵ 可以较大
 - $g^T H g > 0$, $\epsilon^* = \frac{g^T g}{g^T H g}$
 - 如果梯度与 H 的最大特征值对应特征向量对齐 , $\epsilon^* = \frac{1}{\lambda_{\max}}$

二阶导数

- 临界点判别
 - Hessian 矩阵正定，临界点是极小值
 - Hessian 矩阵负定，临界点是极大值

梯度下降的问题

- Hessian矩阵 H 的条件数 衡量 方向二阶导数 $\mu^T H \mu$ 的变化范围
 - 很差的条件数，梯度下降法也很差
 - 特征值大的方向导数增长快
 - 特征值小的方向导数增长慢
 - 很差的条件数，很难选择合适的学习率
-
- 右图：条件数为5



- 假设函数 $f(x)$ 的二阶泰勒展开
 - $f(x) \approx f(x_0) + (x - x_0)^T g + \frac{1}{2}(x - x_0)^T H(x - x_0)$
 - 其临界点为 $x^* = x_0 - H^{-1}g$
 - 若 $f(x)$ 为正定二次函数，牛顿法一次到最优值
 - 若 $f(x)$ 局部近似为正定二次，需要多次迭代
- 只有在局部极小值附近才适用
- 一般会限定优化目标函数来满足这些条件
 - 函数满足Lipschitz 连续 $\forall x, y, |f(x) - f(y)| \leq L\|x - y\|_2$
 - 函数导数满足Lipschitz 连续
 - 凸函数

- $\min f(x), s. t. x \in \mathbb{S}$
 - $\mathbb{S} = \{x | \|x\|_2 \leq 1\}$
- 简单优化策略
 - 梯度下降+投影
- KKT方法：一种通用解决方案
- 通过等式和不等式定义 \mathbb{S}
 - $\mathbb{S} = \{x | \forall i, g^{(i)}(x) = 0 \text{ 且 } \forall j, h^{(j)} \leq 0\}$

- 广义拉格朗日函数

- $L(x, \lambda, \alpha) = f(x) + \sum_i \lambda_i g^{(i)}(x) + \sum_j \alpha_j h^{(j)}(x)$
- λ_i 和 α_j 称为拉格朗日乘子

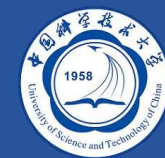
- $\min_x \max_{\lambda, \alpha, \alpha > 0} L(x, \lambda, \alpha) \Leftrightarrow \min_{x \in \mathbb{S}} f(x)$

- 如果 $x \in \mathbb{S}$, $\max_{\lambda, \alpha, \alpha > 0} L(x, \lambda, \alpha) = f(x)$
- 否则 $\max_{\lambda, \alpha, \alpha > 0} L(x, \lambda, \alpha) = \infty$

- 有些情况下 min和max的顺序是可以等价交换

- SVM

KKT方法的最优解



- 广义拉格朗日函数梯度为0
- 所有x和乘子满足的约束满足
- 不等式约束的互补松弛性： $\forall i$, 满足 $h^{(i)}(x) \times \alpha_i = 0$

线性最小二乘

- $f(x) = \frac{1}{2} \|Ax - b\|_2^2$
 - $\nabla_x f(x) = A^T (Ax - b) = A^T Ax - A^T b$

将步长 (ϵ) 和容差 (δ) 设为小的正数。

while $\|A^T Ax - A^T b\|_2 > \delta$ **do**

$x \leftarrow x - \epsilon \left(A^T Ax - A^T b \right)$

end while

线性最小二乘（约束）

- $f(x) = \frac{1}{2} \|Ax - b\|_2^2, \text{ s.t. } x \in \mathbb{S} = \{x | x^T x \leq 1\}$
- $L(x, \lambda) = \frac{1}{2} \|Ax - b\|_2^2 + \lambda(x^T x - 1)$
- 原问题转化为 $\min_x \max_{\lambda} L(x, \lambda)$
- 如果 $x = A^+ b \in \mathbb{S}$, x 为该问题的解
- 否则 $x = (A^T A + 2\lambda)^{-1} A^T b$
 - λ 需要使得满足 $x \in \mathbb{S}$, 可以通过关于 λ 的梯度上升来找到该值
 - $\frac{\partial L(x, \lambda)}{\partial \lambda} = x^T x - 1 > 0$
 - 为了增大 $L(x, \lambda)$, 需增加 λ
 - 依据KKT条件, 最优的 λ 使得 $x^T x = 1$

A decorative geometric pattern on the left side of the slide, featuring a large central circle with several smaller circles and lines radiating from its center, creating a star-like or web-like structure.

目录

1. 线性代数

2. 概率与信息论

3. 数值计算与优化



4. 机器学习基础

- 对于任务 T 和性能度量 P ，计算机程序被认为可以从经验 E 中学习
 - 通过经验 E 改进后，它在任务 T 上由性能度量 P 衡量的性能有所提升
- 任务 T
 - 分类： $f: \mathbb{R}^n \rightarrow \{1, \dots, k\}$
 - 回归： $f: \mathbb{R}^n \rightarrow \mathbb{R}$
 - 转录：光学字符识别、语音识别
 - 机器翻译：序列到序列的映射
 - 结构化输出：输出值之间有关系（图像描述）
 - 异常检测：标记不正常或非典型的样本（信用卡欺诈）
 - 合成和采样：生成一些和训练数据相似的新样本
 - 缺失值填充/去噪/密度估计

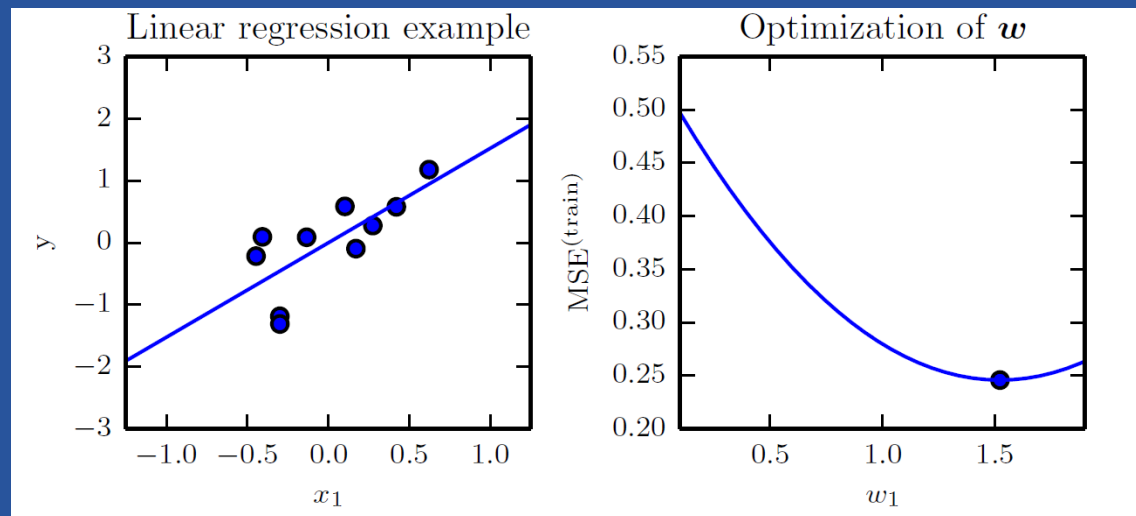
- 性能度量 P
 - 测试集上的准确率
 - 测试集上的错误率
 - 测试集上的均方误差
 - 测试集上的绝对误差

- 经验E

- 无监督学习：训练含很多特征的数据集，学习出数据集上有用的结构性质
 - 学习 $p(x)$
- 有监督学习：训练含很多特征的且带标签的数据集
 - 学习 $p(y|x)$
- 半监督学习：一些样本有监督目标，但其他样本没有
- 强化学习：算法会和环境进行交互，学习系统和训练过程会有反馈回路

示例：线性回归

- 任务T: 学习函数 $f: \mathbb{R}^n \rightarrow \mathbb{R}$, 而且 $f(x) = w^T x$
- 性能度量P : 测试集的设计矩阵 $X_{test} \in \mathbb{R}^{m \times n}$, 回归目标 $y_{test} \in \mathbb{R}^m$
 - $MSE_{test} = \frac{1}{m} \|\hat{y}_{test} - y_{test}\|^2$
 - $\hat{y}_{test} = X_{test} w$ 表示预测值
- 从训练集 (X_{train}, y_{train}) 获得经验 , 最小化 MSE_{train}
 - 最优参数 $w^* = (X_{train}^T X_{train})^{-1} X_{train}^T y_{train}$



容量、过拟合和欠拟合

- 泛化能力：在未观测输入上表现良好的能力
- 泛化误差：未观测输入的误差期望
 - 通过度量模型在训练集中分出来的测试集（test set）样本上的性能

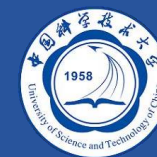
训练误差

$$\frac{1}{m^{(\text{train})}} \left\| \mathbf{X}^{(\text{train})} \mathbf{w} - \mathbf{y}^{(\text{train})} \right\|_2^2$$

测试误差

$$\frac{1}{m^{(\text{test})}} \left\| \mathbf{X}^{(\text{test})} \mathbf{w} - \mathbf{y}^{(\text{test})} \right\|_2^2$$

容量、过拟合和欠拟合

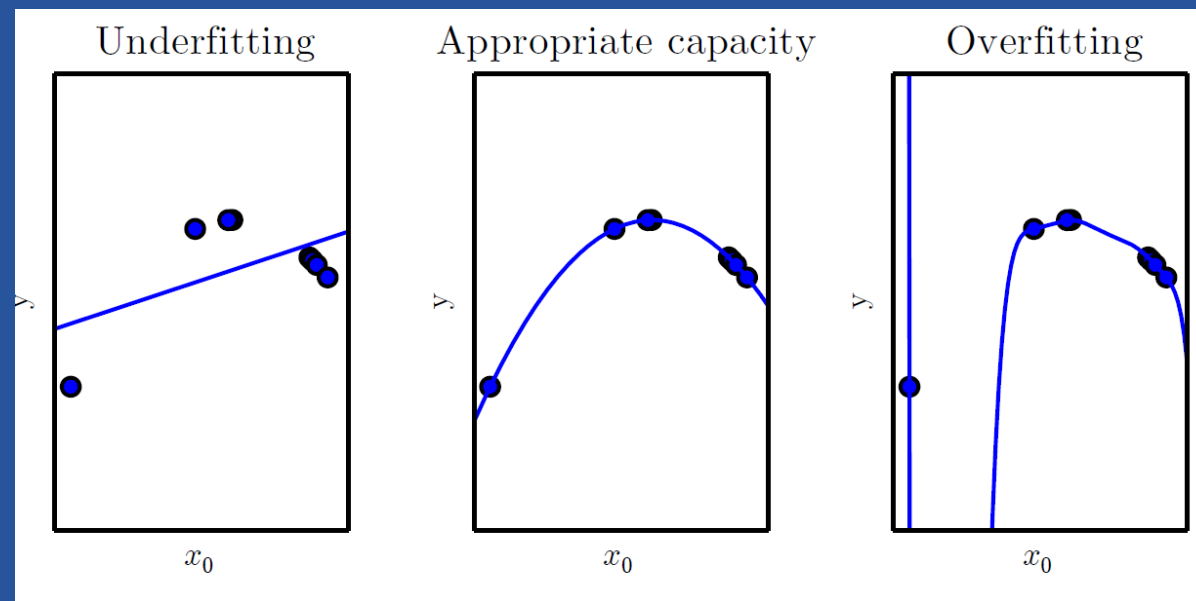


72

- 训练集与测试集 独立同分布，通过 $p_{data}(x)$ 生成
 - 任意随机选择模型，期望训练误差和期望测试误差相等
- 机器学习过程
 - 采样训练集，估计模型参数，计算训练误差
 - 采样测试集，计算测试误差
 - 测试误差 \geq 训练误差
- 机器学习算法效果好的因素
 - 降低训练误差（欠拟合）
 - 缩小训练误差和测试误差的差距（过拟合）

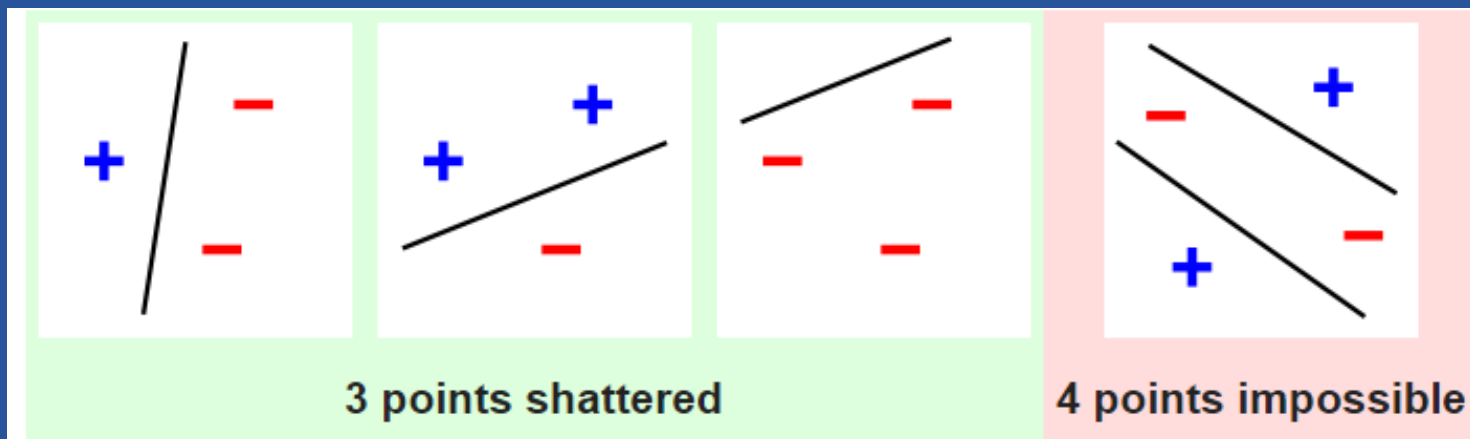
容量、过拟合和欠拟合

- 容量：拟合各种函数的能力
 - 容量低，欠拟合
 - 容量高，过拟合
- 选择假设空间是控制容量的重要方法
 - 学习算法可以选择的函数集
 - 线性回归：所有线性函数
 - 广义线性回归：所有多项式函数
- 例子
 - $f(x) = wx + b$
 - $f(x) = b + w_1x + w_2x^2$
 - $f(x) = b + \sum_{i=1}^9 w_ix^i$



容量、过拟合和欠拟合

- 奥卡姆剃刀原则
 - 在同样能够解释已知观测现象的假设中，应该挑选“最简单”的那一个
- 量化模型容量 — VC维度
 - 度量二元分类器的容量
 - 分类器能够分类的训练样本的最大数目
 - 假设存在 m 个不同 x 点的训练集，分类器可以任意地标记该 m 个不同的 x 点，VC维被定义为 m 的最大可能值
 - 二维平面上的线性分类器的VC维=3



容量、过拟合和欠拟合



75

- 训练误差和泛化误差的差异

Theorem. Let \mathcal{H} be given, and let $d = \text{VC}(\mathcal{H})$. Then with probability at least $1 - \delta$, we have that for all $h \in \mathcal{H}$,

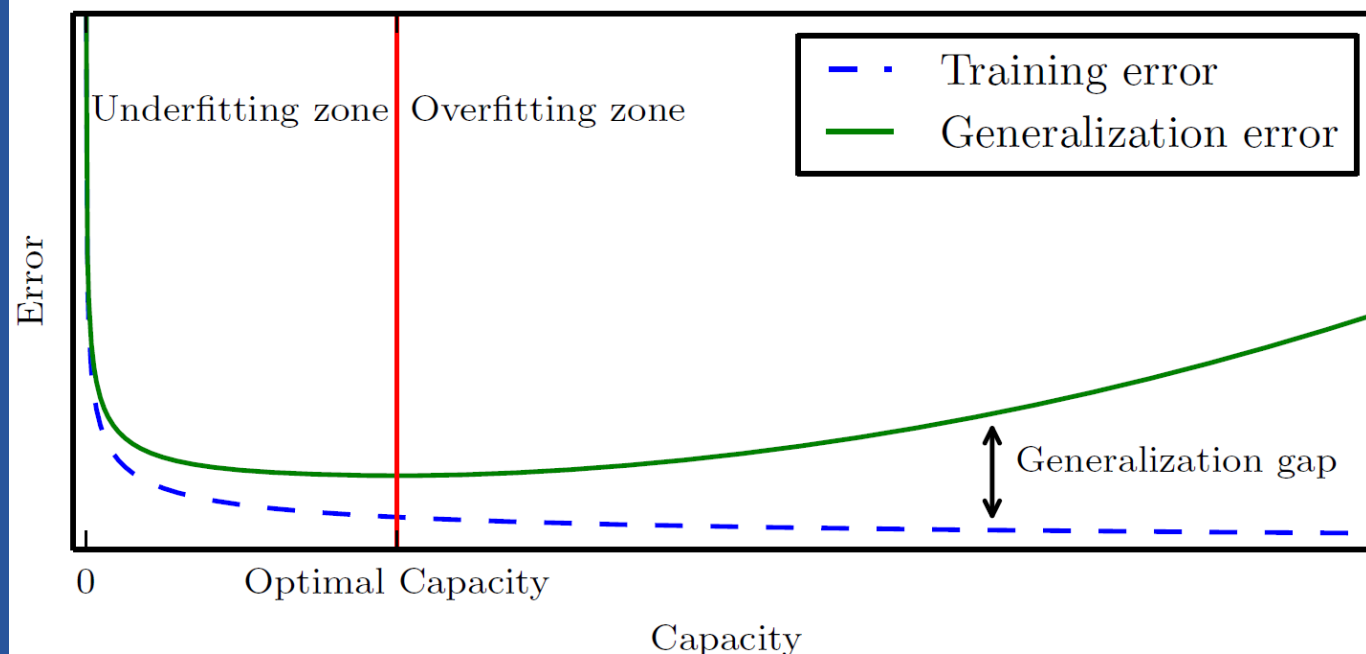
$$|\varepsilon(h) - \hat{\varepsilon}(h)| \leq O \left(\sqrt{\frac{d}{m} \log \frac{m}{d} + \frac{1}{m} \log \frac{1}{\delta}} \right).$$

- VC维越大，即容量越大，差异越大
- 样本越多，差异越小

容量、过拟合和欠拟合

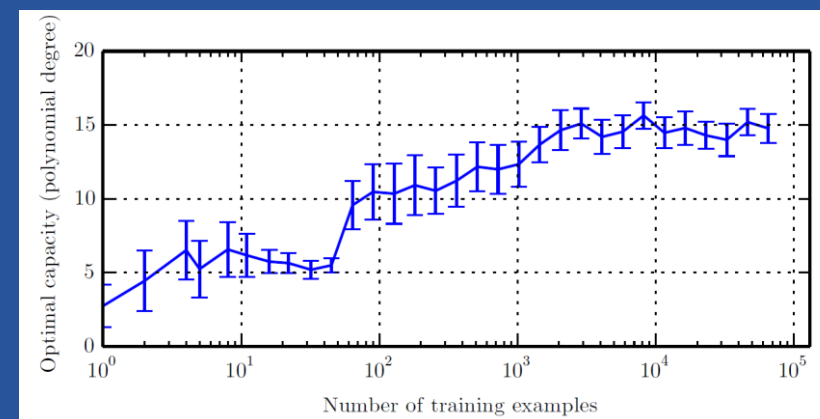
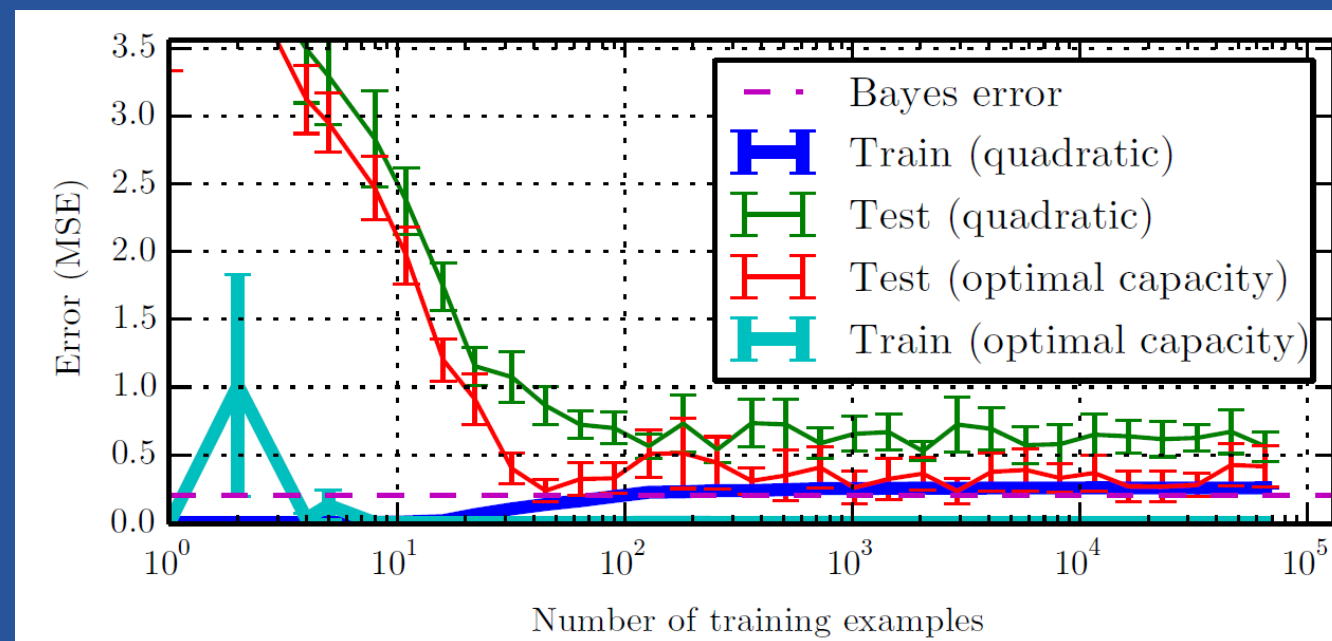
- 足够复杂的模型，足够大的容量，获得足够低的训练误差
- 模型不能过于复杂，使得泛化间隔不会太大。
- 模型容量的极限是最近邻模型，获得最小的训练误差

Generalization and Capacity

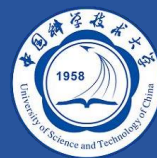


容量、过拟合和欠拟合

- 构造回归问题：给一个5阶多项式添加适当大小的噪声
 - 生成单个测试集
 - 生成一些不同尺寸的训练集
- 二阶模型
 - 数据集越大，训练误差越大，但测试误差越小
 - 测试误差渐进于较高值
- 最优阶模型
 - 测试误差渐进于贝叶斯误差
 - 训练误差低于贝叶斯误差



没有免费午餐定理



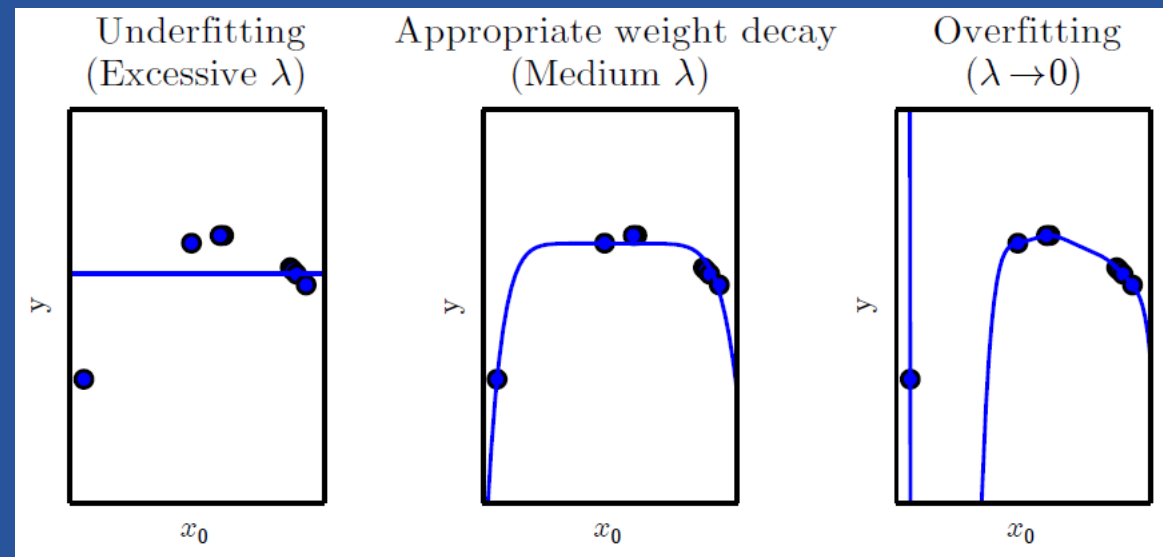
- 在所有可能的数据生成分布上平均之后，每一个分类算法在未事先观测的点上都有相同的错误率
- 没有一个机器学习算法总是比其他的要好
- 仅考虑所有可能的数据生成分布时成立
- 需要设计在特定任务上性能良好的机器学习算法

正则化

- 权重衰减是通过偏好设定来控制算法的性能的一种方法

$$J(w) = \text{MSE}_{\text{train}} + \lambda w^T w,$$

- $\lambda = 0$ 没有任何偏好
- 越大的 λ 偏好越小的权重
- 权重衰减控制模型欠拟合或过拟合



- 正则化一个学习函数 $f(x; w)$ 的模型，我们可以给代价函数添加被称为正则化项的惩罚。
 - $\Omega(w) = w^T w$
 - $\Omega(w) = |w|_1$
- 正则化是指我们修改学习算法，使其降低泛化误差而非训练误差
- 没有免费午餐定理也说明没有最优的正则化形式

超参数和验证集

- 超参数可以设置来控制算法行为
- 超参数的值不是通过学习算法本身学习出来的
 - 难优化
 - 不适合在训练集上学习
- 验证集：用于选择/学习超参数
 - 训练集观测不到
 - 不含测试样本
 - 将数据集一分为二，比例通常为4:1

- 小规模测试集带来较大的不确定性，难以判断算法性能的差异性
- 通过k-fold交叉验证，计算整个数据集的测试误差
 - 将数据集分成 k 份不重合子集
 - 第 i 次测试时，选择第 i 个子集来作为测试集，其他作为训练集
 - 用 k 次测试计算后的平均测试误差作为整个数据集的测试误差

点估计、偏差与方差

- 令 $\{x_1, \dots, x_m\}$ 独立同分布，点估计是这些数据的任意函数
 - $\hat{\theta}_m = g(x_1, \dots, x_m)$
- $\text{bias}(\hat{\theta}_m) = \mathbb{E}(\hat{\theta}_m) - \theta$
 - $\text{bias}(\hat{\theta}_m) = 0$ 无偏
 - $\lim_{m \rightarrow \infty} \text{bias}(\hat{\theta}_m) = 0$ 渐进无偏
- 贝努利分布 $p(x; \theta) = \theta^x (1 - \theta)^{1-x}$ 采样
 - $\hat{\theta}_m = \frac{1}{m} \sum_i x_i$ 无偏
- 高斯分布 $p(x; \mu, \sigma^2)$ 采样
 - $\hat{\mu}_m = \frac{1}{m} \sum_i x_i$ 无偏
 - $\hat{\sigma}_m^2 = \frac{1}{m} \sum_i (x_i - \hat{\mu}_m)^2$ 有偏

点估计、偏差与方差

- 高斯分布 $p(x; \mu, \sigma^2)$ 方差的无偏估计

- $\hat{\sigma}_m^2 = \frac{1}{m-1} \sum_i (x_i - \hat{\mu}_m)^2$

$$\begin{aligned} \text{bias}(\hat{\sigma}_m^2) &= \mathbb{E}(\hat{\sigma}_m^2) - \sigma^2 \\ &= \frac{1}{m-1} \mathbb{E} \left(\sum_i (x_i - \hat{\mu}_m)^2 \right) - \sigma^2 \\ &= \frac{1}{m-1} \mathbb{E} \left(\sum_i (x_i - \mu + \mu - \hat{\mu}_m)^2 \right) - \sigma^2 \\ &= \frac{1}{m-1} \sum_i (\sigma^2 - \mathbb{E}(\mu - \hat{\mu}_m)^2) - \sigma^2 \\ &= \frac{m}{m-1} \left(1 - \frac{1}{m} \right) \sigma^2 - \sigma^2 \\ &= 0 \end{aligned}$$

- $\mathbb{E}(\mu - \hat{\mu}_m)^2 = \frac{1}{m^2} \sum_i \mathbb{E}(x_i - \mu)^2 = \frac{\sigma^2}{m}$

- $Var(\hat{\theta}_m) = SE(\hat{\theta}_m)^2$
- 使用有限的样本计算任何统计量，真实参数的估计是不确定的
- $Var(\hat{\mu}_m) = \mathbb{E}(\mu - \hat{\mu}_m)^2 = \frac{\sigma^2}{m}$
- 根据中心极限定理， $\hat{\mu}_m$ 满足高斯分布，用标准差计算 μ 落在选定区间的概率。
- 以均值 $\hat{\mu}_m$ 为中心的95% 置信区间是

$$(\hat{\mu}_m - 1.96SE(\hat{\mu}_m), \hat{\mu}_m + 1.96SE(\hat{\mu}_m)).$$

- 在机器学习实验中，算法A 比算法B 差，指算法A 的误差95% 置信区间的上界小于算法B 误差的95% 置信区间的下界。

偏差和方差权衡

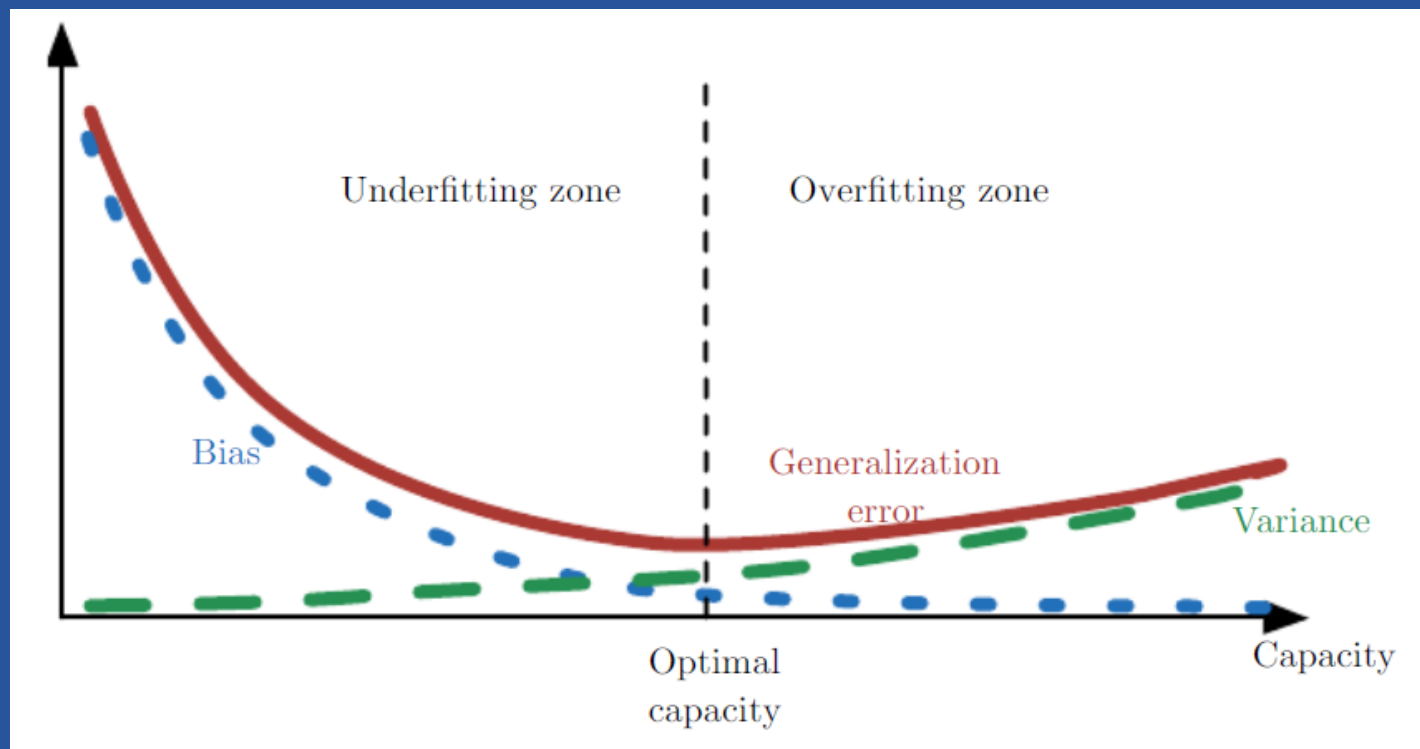
- 偏差度量 偏离真实函数或参数的误差期望。
- 方差度量 数据上任意特定采样可能导致的估计期望的偏差
- 考虑点估计和真实值的均方误差₂

$$\begin{aligned}MSE &= \mathbb{E}(\hat{\theta} - \theta)^2 \\&= \mathbb{E}(\hat{\theta} - \mathbb{E}\hat{\theta} + \mathbb{E}\hat{\theta} - \theta)^2 \\&= \mathbb{E}(\hat{\theta} - \mathbb{E}\hat{\theta})^2 + (\mathbb{E}\hat{\theta} - \theta)^2 \\&= \text{Var}(\hat{\theta}) + \text{bias}(\hat{\theta})^2\end{aligned}$$

- 最小化均方误差要选择偏差小且方差小的估计

偏差和方差权衡

- 用MSE度量泛化误差
 - 增加容量会增加方差，降低偏差



一致性

- 依概率收敛 $\text{plim}_{m \rightarrow \infty} \hat{\theta}_m = \theta$
- 对于任意的 $\epsilon > 0$, 当 $m \rightarrow \infty$ 时, 有 $P(|\hat{\theta}_m - \theta| > \epsilon) \rightarrow 0$
- 一致性保证了估计量的偏差会随数据样本数目的增多而减少

估计准则——极大似然估计

- $p_{data}(x)$ 生成 m 个样本的数据集 $\mathbb{X} = \{x_1, \dots, x_m\}$.
- $p_{model}(x; \theta)$ 一族由 θ 确定在相同空间上的概率分布
- θ 的极大似然估计为
 - $\theta_{ML} = \arg \max p_{model}(\mathbb{X}; \theta)$
 - $= \arg \max \prod_i p_{model}(x_i; \theta)$
 - $= \arg \max \sum_i \log p_{model}(x_i; \theta)$
 - $= \arg \max \mathbb{E}_{x \sim \hat{p}_{data}}(\log p_{model}(x; \theta))$
- $\mathbb{E}_{x \sim \hat{p}_{data}}(\log p_{model}(x; \theta))$ 等价于 $KL(\hat{p}_{data} | p_{model})$
 - $KL(\hat{p}_{data} | p_{model}) = \mathbb{E}_{x \sim \hat{p}_{data}}(\log \hat{p}_{data}(x) - \log p_{model}(x))$

估计准则——极大似然估计

- 条件对数似然

- $\theta_{ML} = \arg \max_{\theta} \sum_i \log P(y^i | x^i; \theta)$

- 线性回归作为最大似然

- $p(y|x) = \mathcal{N}(y; \hat{y}(x; w), \sigma^2)$ $\hat{y}(x; w)$ 预测均值, 方差是常量

$$\begin{aligned} & \sum_{i=1}^m \log p(y^{(i)} | \mathbf{x}^{(i)}; \boldsymbol{\theta}) \\ &= -m \log \sigma - \frac{m}{2} \log(2\pi) - \sum_{i=1}^m \frac{\|\hat{y}^{(i)} - y^{(i)}\|^2}{2\sigma^2}, \end{aligned}$$

- 均方误差和对数似然等价

最大似然的性质

- 在合适的条件下，最大似然估计具有一致性
 - 真实分布 p_{data} 必须在模型族 $p_{model}(\cdot; \theta)$
 - 真实分布 p_{data} 必须刚好对应一个 θ
- 参数的均方误差估计随着 m 的增加而减少，当 m 较大时，不存在均方误差低于最大似然估计的一致估计
- 最大似然通常是机器学习中的首选估计

- 频率派统计：基于点估计作所有的预测，参数是固定值
- 贝叶斯统计：在做预测时会考虑所有可能的估计，参数是随机变量
 - 先验概率分布： $p(\theta)$ 表示 θ 的已知知识，一般会选择高熵的先验
 - 后验概率分布：
$$p(\theta|x_1, \dots, x_m) = \frac{p(x_1, \dots, x_m|\theta)p(\theta)}{p(x_1, \dots, x_m)}$$
 - $p(x_{m+1}|x_1, \dots, x_m) = \int p(x_{m+1}|\theta)p(\theta|x_1, \dots, x_m)d\theta$
 - 先验分布表明参数空间中偏好，偏好更简单或更光滑的模型

- 似然

$$\begin{aligned} p(\mathbf{y}^{(\text{train})} \mid \mathbf{X}^{(\text{train})}, \mathbf{w}) &= \mathcal{N}(\mathbf{y}^{(\text{train})}; \mathbf{X}^{(\text{train})} \mathbf{w}, \mathbf{I}) \\ &\propto \exp \left(-\frac{1}{2} (\mathbf{y}^{(\text{train})} - \mathbf{X}^{(\text{train})} \mathbf{w})^\top (\mathbf{y}^{(\text{train})} - \mathbf{X}^{(\text{train})} \mathbf{w}) \right) \end{aligned} \quad (5)$$

- 先验

$$p(\mathbf{w}) = \mathcal{N}(\mathbf{w}; \boldsymbol{\mu}_0, \boldsymbol{\Lambda}_0) \propto \exp \left(-\frac{1}{2} (\mathbf{w} - \boldsymbol{\mu}_0)^\top \boldsymbol{\Lambda}_0^{-1} (\mathbf{w} - \boldsymbol{\mu}_0) \right)$$

- 后验

$$p(\mathbf{w} \mid \mathbf{X}, \mathbf{y}) \propto p(\mathbf{y} \mid \mathbf{X}, \mathbf{w})p(\mathbf{w}) \quad (5.74)$$

$$\propto \exp \left(-\frac{1}{2}(\mathbf{y} - \mathbf{X}\mathbf{w})^\top (\mathbf{y} - \mathbf{X}\mathbf{w}) \right) \exp \left(-\frac{1}{2}(\mathbf{w} - \boldsymbol{\mu}_0)^\top \boldsymbol{\Lambda}_0^{-1}(\mathbf{w} - \boldsymbol{\mu}_0) \right) \quad (5.75)$$

$$\propto \exp \left(-\frac{1}{2} \left(-2\mathbf{y}^\top \mathbf{X}\mathbf{w} + \mathbf{w}^\top \mathbf{X}^\top \mathbf{X}\mathbf{w} + \mathbf{w}^\top \boldsymbol{\Lambda}_0^{-1}\mathbf{w} - 2\boldsymbol{\mu}_0^\top \boldsymbol{\Lambda}_0^{-1}\mathbf{w} \right) \right).$$

$$p(\mathbf{w} \mid \mathbf{X}, \mathbf{y}) \propto \exp \left(-\frac{1}{2}(\mathbf{w} - \boldsymbol{\mu}_m)^\top \boldsymbol{\Lambda}_m^{-1}(\mathbf{w} - \boldsymbol{\mu}_m) + \frac{1}{2}\boldsymbol{\mu}_m^\top \boldsymbol{\Lambda}_m^{-1}\boldsymbol{\mu}_m \right) \quad (5.77)$$

$$\propto \exp \left(-\frac{1}{2}(\mathbf{w} - \boldsymbol{\mu}_m)^\top \boldsymbol{\Lambda}_m^{-1}(\mathbf{w} - \boldsymbol{\mu}_m) \right). \quad (5.78)$$

$$\boldsymbol{\Lambda}_m = (\mathbf{X}^\top \mathbf{X} + \boldsymbol{\Lambda}_0^{-1})^{-1}$$

$$\boldsymbol{\mu}_m = \boldsymbol{\Lambda}_m(\mathbf{X}^\top \mathbf{y} + \boldsymbol{\Lambda}_0^{-1}\boldsymbol{\mu}_0)$$

- 贝叶斯后验的计算常常是非常棘手的，点估计提供可行的近似解
- 最大后验估计

$$\theta_{\text{MAP}} = \arg \max_{\theta} p(\theta \mid \mathbf{x}) = \arg \max_{\theta} \log p(\mathbf{x} \mid \theta) + \log p(\theta).$$

- 如果先验是 $\mathcal{N}(w; 0, \frac{1}{\lambda} I)$, 那么 $\log p(\theta)$ 正比于权重衰减 $\lambda w^T w$
- 高斯先验的MAP 贝叶斯推断对应着权重衰减

- 概率监督学习

- 回归模型： $p(y|x) = \mathcal{N}(y; \theta^T x, I)$ 解析解

- 逻辑斯特回归： $p(y|x; \theta) = \sigma(\theta^T x)^y (1 - \sigma(\theta^T x))^{1-y}$ 迭代优化

- 支持向量机

- 决策函数 $\hat{y} = \begin{cases} 1, w^T x + b > 0 \\ -1, w^T x + b < 0 \end{cases}$

- $w^* = \sum_i \alpha_i x_i$

- $f(x) = w^T x + b = \sum_i \alpha_i x^T x_i + b$

- 核技巧：用核函数 $k(x, x_i)$ 替换点积： $f(x) = b + \sum_i \alpha_i k(x, x_i)$

- 高斯核 $k(u, v) = \mathcal{N}(u - v; \mathbf{0}, \sigma^2 I)$, 执行模板匹配

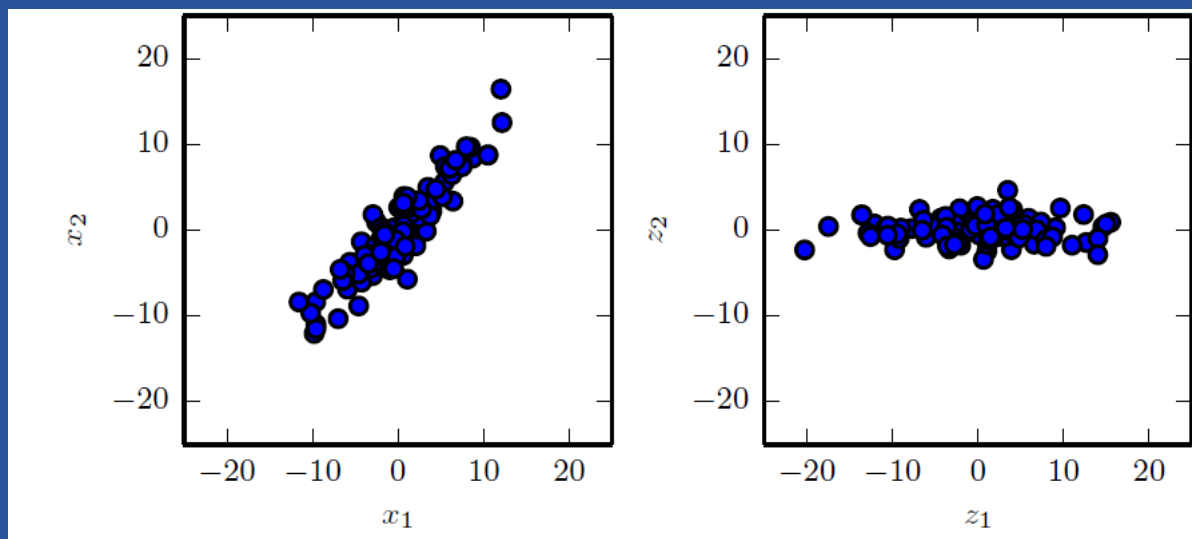
- $f(x)$ 的计算成本关于训练样本是线性的

- 但支持向量机模型学出的 α_i 是稀疏的，非零 α_i 对应的样本称为支持向量

- 经典的无监督学习任务是找到数据的“最佳”表示
 - 最佳表示在比 x 本身表示的信息更简单或更易访问而受到一些惩罚或限制的情况下，尽可能地保存关于 x 更多的信息
- 常见的三种表示
 - 低维表示：尝试将 x 中的信息尽可能压缩在一个较小的表示中
 - 稀疏表示：将数据集嵌入到输入项大多数为零的表示
 - 独立表示：试图分开数据分布中变化的来源，使得表示的维度是统计独立的

主成分分析 (PCA)

- 主成分分析学习一种比原始输入维数更低的表示。它也学习了一种元素之间彼此没有线性相关的表示
- 下图学习数据的正交线性变换
 - 左边原始数据 , x_1, x_2 相关
 - 右边旋转之后的数据 , z_1, z_2 独立



主成分分析 (PCA)

- 考虑设计矩阵 $X \in \mathbb{R}^{m \times n}$ 含 m 个数据点，每个点维度 n
 - 假设 $\mathbb{E}(x) = 0$
 - $Var(x) = \frac{1}{m-1} \sum_i x_i x_i^T = \frac{1}{m-1} X^T X$
- PCA 通过正交变换 $z = W^T x$ 使得 $Var(z)$ 是对角的
- 令 $X = U \Sigma W^T$ 是 X 的奇异值分解，则 W 可以满足
 - $Var(x) = \frac{1}{m-1} W \Sigma^2 W^T$
 - $Var(z) = \frac{1}{m-1} Z^T Z = \frac{1}{m-1} W^T X^T X W = \frac{1}{m-1} W^T W \Sigma^2 W^T W = \frac{1}{m-1} \Sigma^2$
 - z 中元素彼此无关
- 但难处理非线性相关关系

K均值聚类

- k -均值聚类算法将训练集分成 k 个靠近彼此的不同样本聚类
- 提供了 k -维的one-hot 编码向量 h 以表示输入 x

经典 K-means 算法
Step 1: 从数据集中随机选取 K 个样本作为初始聚类中心 $C = \{c_1, c_2, \dots, c_k\}$;
Step 2: 针对数据集中每个样本 x_i , 计算它到 K 个聚类中心的距离并将其分到距离最小的聚类中心所对应的类中;
Step 3: 针对每个类别 c_i , 重新计算它的聚类中心 $c_i = \frac{1}{ c_i } \sum_{x \in c_i} x$ (即属于该类的所有样本的质心);
Step 4: 重复第 2 步和第 3 步直到聚类中心的位置不再变化;

- 问题：难以衡量聚类的好坏

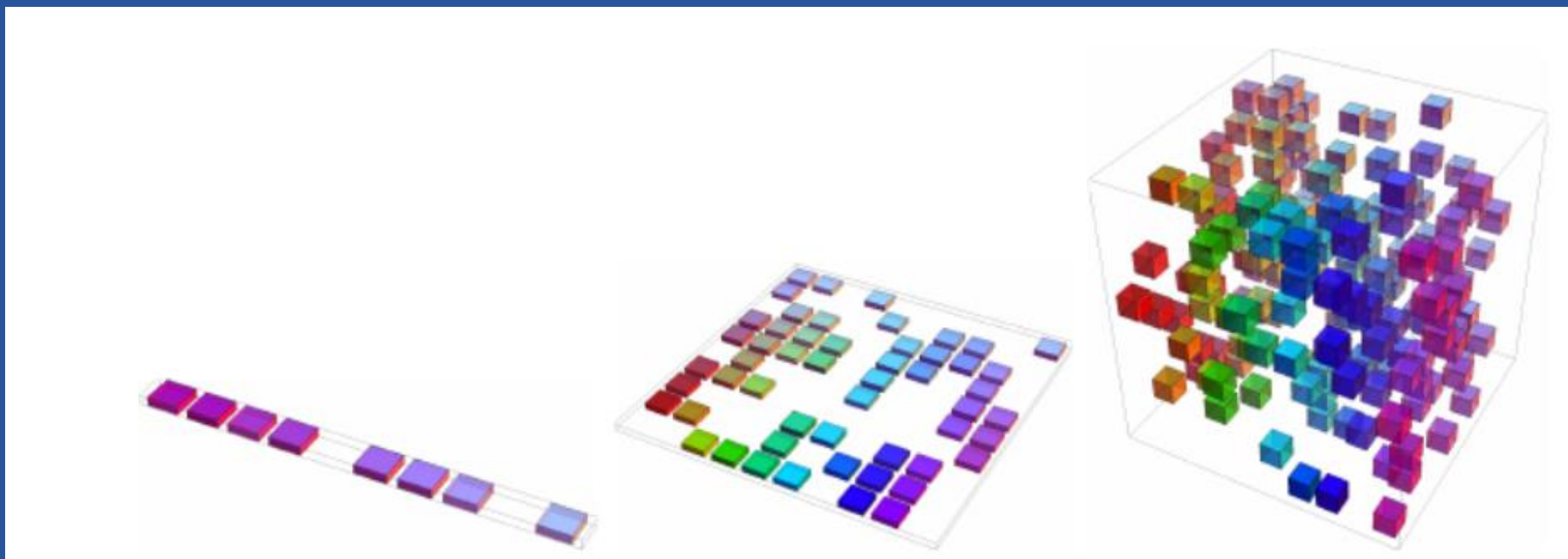
随机梯度下降

- 机器学习中反复出现的问题是好的泛化需要大的训练集，但大的训练集的计算代价也更大
- 假设 $J(\theta) = \mathbb{E}_{x,y \sim \hat{p}_{data}} L(x, y, \theta) = \frac{1}{m} \sum_i L(x_i, y_i, \theta)$, $L(x_i, y_i, \theta) = -\log p(y|x; \theta)$
- 梯度为 $\nabla_{\theta} J(\theta) = \frac{1}{m} \sum_i \nabla_{\theta} L(x_i, y_i, \theta)$ 计算代价为 $O(m)$
 - 随着训练集规模增长为数十亿的样本，计算一步梯度会消耗相当长的时间
- 随机梯度下降的核心：**梯度是期望**，可使用小规模样本近似估计
 - 在算法的每一步，我们从训练集中均匀抽出一小批量样本（ $m' = \text{几十到几百}$ ）
 - $g = \frac{1}{m'} \sum_{i=1}^{m'} \nabla_{\theta} L(x_i, y_i, \theta)$
 - **$\theta \leftarrow \theta - \epsilon g$**

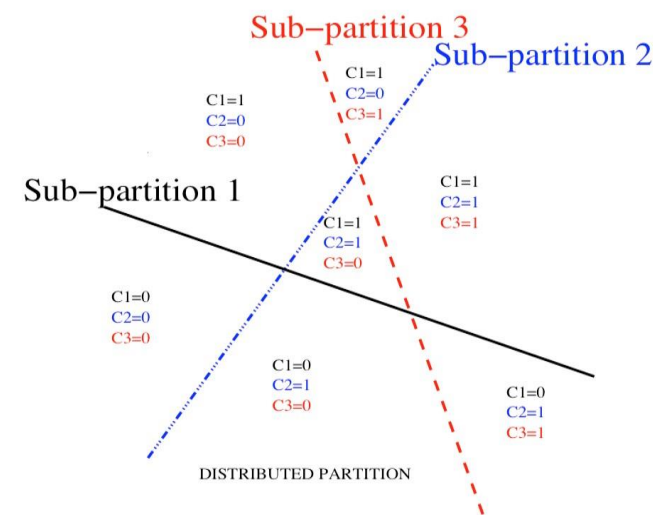
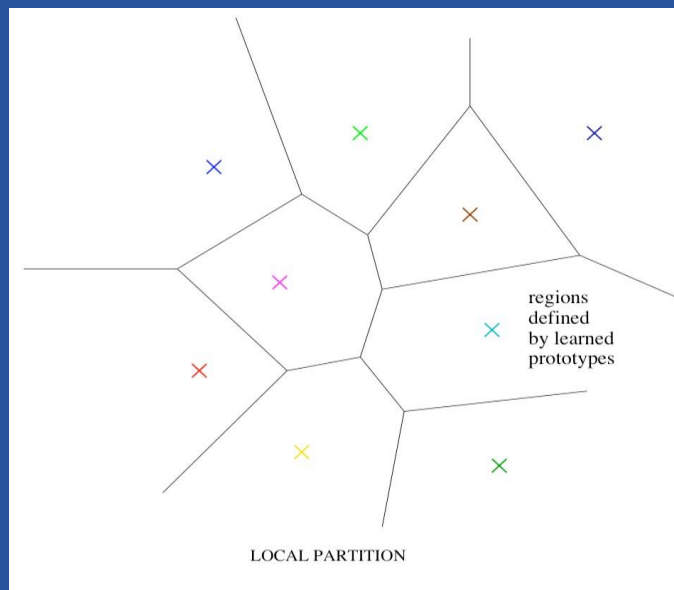
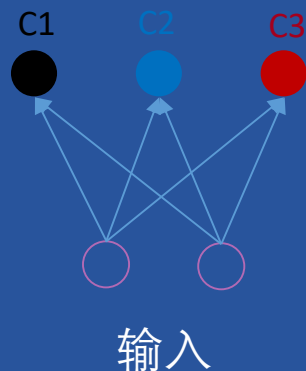
- 配方
 - 特定数据集
 - 模型
 - 代价函数
 - 优化过程
- 线性回归 (数据集 X, y)
 - 模型 $p_{model}(y|x) = \mathcal{N}(y; x^T w + b, 1)$
 - 代价函数 $J(w, b) = -\mathbb{E}_{x, y \sim \hat{p}_{data}} \log p_{model}(y|x)$, 还有正则化项
 - 优化算法：求解正规方程 (代价函数梯度为0)
- PCA
 - 模型 $r(x; w) = w^T x w$
 - 代价函数 $J(w) = \|x - r(x; w)\|_2$
 - 优化算法：SVD

机器学习 与 深度学习

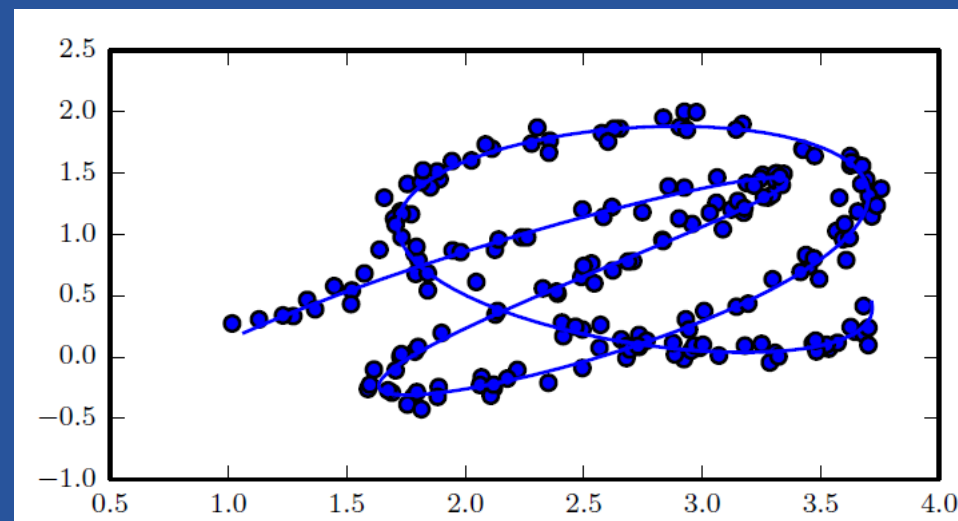
- 维数灾难
 - 机器学习在特征维度很高的时候变得异常困难



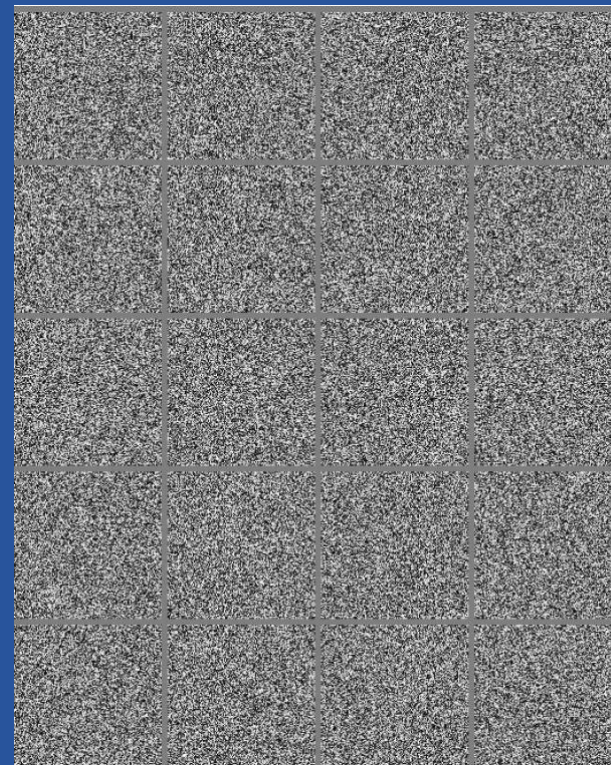
- 局部不变性和平滑正则化
 - 平滑/局部不变先验: $f^*(x) \approx f^*(x + \epsilon)$
 - 具平滑/局部不变先验的简单机器学习方法：样本数决定不同区域数
 - 最近邻算法：不同区域数不会多于训练样本数
 - 无法很好地泛化到未观测样本
- 深度学习假设数据由因素或特征组合产生
 - 区域之间的依赖性
 - $O(k)$ 样本, $O(2^k)$ 区域



- 流形
 - 连接在一起的区域
 - 数学上，它是指一组点，且每个点都有其邻域
 - 给定一个任意的点，其流形局部看起来像是欧几里得空间
- 流形学习
 - 输入只分布在包含少量数据点的子集构成的一组流形中
 - 输出中有意义的变化都沿着流形的方向或仅发生在我们切换到另一流形时



- 流形假设证据
 - 现实生活中的图像、文本、声音的概率分布都是高度集中的
 - 现实中的领域和变换：逐渐变暗或变亮光泽、逐步移动或旋转图中对象、逐渐改变对象表面的颜色
- 用流形中的坐标表示样本更加合适





谢谢

