

优化建模

文再文

北京大学北京国际数学研究中心

教材《最优化：建模、算法与理论》配套电子教案

<http://bicmr.pku.edu.cn/~wenzw/optbook.html>

致谢：本教案由**华奕轩**协助准备

优化建模

- 确定实际问题的目标，查找文献中是否存在类似的模型
- 收集数据. 大部分情况下，收集数据是十分耗时并且容易出错的，但有时可以很容易地找到数据集.
- 选择一个合适的模型，除了查找相关文献外，还需要结合问题的实际背景. 我们往往会局限于已知的建模工具，因此建立的模型并不一定是最合适的. 对于建立的模型，我们需要对其最简化的版本用实际数据来进行测试，并逐步提高复杂度来判断模型表达的正确性.
- 通过模型在已知情形的表现是否一致来检查模型的合理性.
- 为了让模型具有更高的可信度以及采用度，需要比较解决同一个实际问题已有的其他模型，在相同的数据集上至少给出精度差不多的解，并且某些数据集上的表现要比已有模型好. 如果该实际问题没有其他模型，那么建立的模型在已知情形的预测结果需要和我们的认知一致.

目标函数设计: 最小二乘法

- 设 $\phi_i(x): \mathbb{R}^n \rightarrow \mathbb{R}, i = 1, 2, \dots, m$ 为 n 元函数, 且有如下方程组:

$$b_i = \phi_i(x), \quad i = 1, 2, \dots, m,$$

其中 $b_i \in \mathbb{R}$ 是已知的实数.

- 方程组并不总是可解的. 首先, 方程组的个数 m 可以超过自变量个数 n , 因此方程组的解可能不存在; 其次, 由于测量误差等因素, 等式关系可能不是精确成立的.
- 最小二乘法的思想是极小化误差的 ℓ_2 范数平方, 即

$$\min_{x \in \mathbb{R}^n} \sum_{i=1}^m (b_i - \phi_i(x))^2.$$

- 当 ϕ_i 均为线性函数时称为线性最小二乘问题, 否则称其为非线性最小二乘问题

目标函数设计: 最小二乘法

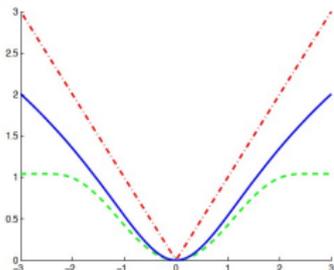
- 如果要保证偏差的绝对值之和最小, 相应的模型为:

$$\min_{x \in \mathbb{R}^n} \sum_{i=1}^m |b_i - \phi_i(x)|;$$

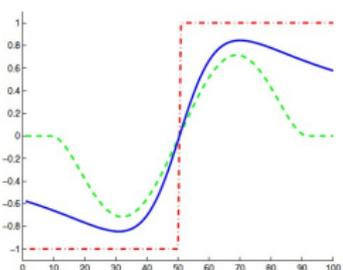
- 如果想要保证最大偏差最小化, 对应的优化模型为:

$$\min_{x \in \mathbb{R}^n} \max_i |b_i - \phi_i(x)|.$$

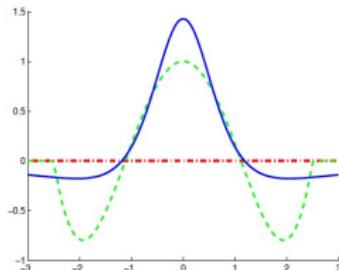
- 一般形式: $\min_{x \in \mathbb{R}^n} \sum_{i=1}^m \rho_i(b_i - \phi_i(x))$



(a) $\rho(r)$



(b) $\nabla \rho(r)$



(c) $\nabla^2 \rho(r)$

目标函数设计: 正则化

- 为了让解具有某种光滑性以及克服问题的病态性质

$$\min_{x \in \mathbb{R}^n} \sum_{i=1}^m (b_i - \phi_i(x))^2 + \mu \|x\|_2^2,$$

其中 $\mu > 0$ 为一个平衡参数.

- 如果想要得到的一个稀疏的解, 可以借助 ℓ_0 范数并构造如下模型:

$$\min_{x \in \mathbb{R}^n} \sum_{i=1}^m (b_i - \phi_i(x))^2 + \mu \|x\|_0,$$

其中 $\mu > 0$ 用来控制解的稀疏度.

- 由于 ℓ_0 范数在实际中难以处理, 我们往往使用 ℓ_1 范数来代替 ℓ_0 范数来保证稀疏性, 即构造模型

$$\min_{x \in \mathbb{R}^n} \sum_{i=1}^m (b_i - \phi_i(x))^2 + \mu \|x\|_1.$$

目标函数设计: 正则化

- 在图像处理中, x 本身可能不是稀疏的, 但是其在变换域中是稀疏的. 相应的模型为

$$\min_{x \in \mathbb{R}^n} \sum_{i=1}^m (b_i - \phi_i(x))^2 + \mu \|W(x)\|_0,$$

- 以及

$$\min_{x \in \mathbb{R}^n} \sum_{i=1}^m (b_i - \phi_i(x))^2 + \mu \|W(x)\|_1,$$

其中 $W: \mathbb{R}^n \rightarrow \mathbb{R}^p$ 表示某种变换, 常用的有全变差以及小波变换.

目标函数设计: 最大似然估计

- 在实际问题中有很多数据来自未知的概率分布, 从数据反推分布的具体形式是非常重要的课题. 最大似然估计就是统计中常用的一种估计概率分布的方法, 其通过最大化似然函数, 使得观测数据尽可能地服从假定的模型.
- 假设已经知道数据来自某种特定的分布, 但不知道该分布具体的参数. 为了方便起见, 令 $p(a; x)$ 是其分布律或概率密度函数, 其中 x 为未知参数. 为了估计 x , 我们选取一系列独立同分布的样本点 a_1, a_2, \dots, a_n . 似然函数定义为在参数 x 下的数据集 $\{a_i, i = 1, 2, \dots, n\}$ 发生的概率, 即

$$L(x) = \prod_{i=1}^n p(a_i; x).$$

我们看到 $L(x)$ 实际上就是这 n 个点的联合概率 (联合密度), 但此时的自变量变成了参数 x .

目标函数设计: 最大似然估计

- 参数的最大似然估计定义为

$$\hat{x} \in \arg \max_{x \in \mathcal{X}} L(x),$$

其中 \mathcal{X} 为参数空间. 假设最大似然估计存在, 则求解最大似然估计本质上是在一族分布中寻找最有可能产生该样本的参数.

- 似然函数的对数的最大值往往更容易求解, 即考虑最大化问题

$$\max_{x \in \mathcal{X}} \ell(x) := \ln L(x).$$

因为 $\ln(x)$ 是严格单调递增的, 因此上面问题的最优解也为 \hat{x} . 但由于取对数可将乘法变为加法, 实际更易操作, 所以在统计中更倾向于使用**对数似然函数** $\ell(x)$.

- 对于很多模型来说, 似然函数的表达式是已知的, 但是其最优解的显式表达式是未知的, 因此我们需要利用数值优化算法求其全局最优解.

目标函数设计

- **代价、损失、收益函数**: 很多问题就是极小化代价（损失）并极大化收益的过程。比如，玩俄罗斯方块每走一步都会有相应的分数奖励，期望最后的得分越高越好。在旅行的时候，我们一般会提前规划好想去的城市并确定行程。此时，我们希望在游览所有城市的情况下路程最短或者差旅费最少。这些实际问题，都可以写成优化问题的形式，其目标函数或者是极小化代价（损失），或者是极大化收益，或者是两者兼顾。
- **泛函、变分**: 物理、化学中的很多问题都可以表述为能量极小化的形式。比如，在电子结构计算中，我们通过极小化原子和电子之间的相互作用能量来计算稳定态。一般来说，能量泛函是定义在函数空间上的，即相应优化问题的自变量是无穷维空间中的函数。常用的另一种方式是利用合适的离散化，将能量泛函的极小化问题从无穷维空间中拉回到有限维空间中，从而得到相应问题的离散解。

目标函数设计: 松弛

- **松弛**: 在保留原问题部分性质的条件下, 使用简单的项替代目标函数中难以处理的项, 进而使得问题更易求解.
- 例如 l_0 范数是不可微且非凸的. 由于 l_1 范数是 l_0 范数在某种程度上的近似, 用 l_1 范数代替目标函数中的 l_0 范数.
- 对于低秩优化问题, 秩对应矩阵奇异值中非零元的个数, 其也是非凸且不可微的. 常用方式是将其用矩阵的核范数 (矩阵奇异值组成的向量的 l_1 范数) 代替
- 另一种松弛的策略是使用目标函数的一个下界 $f_R(x)$ 来替换 $f(x)$ 进行求解, 其中 $f_R(x)$ 应该满足:
 - 1 $f_R(x) \leq f(x), \forall x \in \mathcal{X}$;
 - 2 $f_R(x)$ 具有简单结构.

约束设计：问题本身的物理性质

根据实际问题的具体形式，优化问题的决策变量需要满足各种各样的约束。

- 在电子结构计算中，我们假设轨道函数之间是相互正交的。
- 在化学反应过程当中，各成分的浓度随着时间的变化对应于一个微分方程组。
- 在最优设计中，我们有时需要优化物体的形状。
- 在飞机机翼设计中，受机翼周围的气流影响，机翼形状的改变对应于一个微分方程。
- 在很多情况下，我们还需要非负约束，比如图像的像素值，物体的浓度，拥有的资源数量，等等。
- 当线性或者一般的等式观测带有噪声或者需要更多的鲁棒性时，我们也将等式约束放宽为不等式约束。

约束设计：等价转换

- **目标函数拆分**: 如果目标函数是复合函数的形式, 如

$$\min_x f(Ax + b),$$

引入变量 y 和等式约束 $y = Ax + b$ 而考虑其带约束的等价问题:

$$\min_y f(y), \quad \text{s.t.} \quad y = Ax + b.$$

- 对于优化问题 $\min_x f(x) = h(x) + r(x)$, 引入 y 以及约束 $x = y$:

$$\min_x h(x) + r(y), \quad \text{s.t.} \quad x = y.$$

- **约束拆分**: 对于不等式约束引入松弛变量将其转变为等式约束和简单的非负或非正约束. 如对约束 $c(x) \leq 0$, 引入 $y \geq 0$, 转化为

$$c(x) + y = 0, \quad y \geq 0.$$

- **利用上方图来得到问题的等价形式**. 对于问题

$\min_x f(x)$, 根据上方图的定义, 可知其等价于优化问题:

$$\min_{x,t} t, \quad \text{s.t.} \quad f(x) \leq t.$$

约束设计：松弛

- 当原始模型的约束过于复杂时，用松弛将难处理的约束替换为容易处理的约束
- 设原始问题的可行域为 \mathcal{X} ，松弛之后的可行域为 \mathcal{X}_R ，则 \mathcal{X}_R 需要满足 $\mathcal{X}_R \supset \mathcal{X}$ ，即松弛后问题的可行域会放大。
- 例如用盒约束 $x \in [0, 1]$ 来代替整数约束 $x \in \{0, 1\}$ ，或者用不等式约束 $c(x) \geq 0$ 来代替等式约束 $c(x) = 0$ 。
- 放大可行域要遵守两点基本原则：
 - 将原有约束进行简化，即松弛后的问题更易处理；
 - 不宜将可行域放得过大，过分放大可行域会丢失原问题的关键信息，进而使得求解松弛问题变得没有意义。
- 既然松弛问题的可行域变大，松弛问题的解和原问题的解有什么联系？在一般情况下，松弛问题和原问题不等价。但在一定条件下可以证明松弛问题的解就是原问题的解

约束设计：松弛

例：球约束 $\|x\|_2 = 1$ 的松弛。

- 令 $X = xx^T$ 并添加 $\text{Tr}(X) = 1$ 将其等价刻画。
- 注意到 $X = xx^T$ 仍然是非凸约束，我们将其替换为半正定约束 $X \succeq 0$ 。
- 实际上我们做了如下松弛：

$$\|x\|_2 = 1 \rightarrow X \in \mathcal{S}^n, \text{Tr}(X) = 1, X \succeq 0.$$

- 可以看到，当 X 的秩等于1时，松弛后的约束和球约束是等价的；当 X 的秩大于1时，我们无法找到与松弛问题变量 X 对应的原问题变量 x ，但可以通过一些技巧构造近似解。

机器学习中典型问题形式

很多机器学习中的问题可以写为：

$$\min_{x \in \mathcal{W}} \sum_{i=1}^N \frac{1}{2} \|a_i^\top x - b_i\|_2^2 + \mu \varphi(x) \quad \text{线性回归}$$

$$\min_{x \in \mathcal{W}} \frac{1}{N} \sum_{i=1}^N \log(1 + \exp(-b_i a_i^\top x)) + \mu \varphi(x) \quad \text{逻辑回归}$$

$$\min_{x \in \mathcal{W}} \frac{1}{N} \sum_{i=1}^N \ell(f(a_i, x), b_i) + \mu \varphi(x) \quad \text{一般形式}$$

- (a_i, b_i) 是给定的数据对， b_i 是数据 a_i 对应的标签
- $\ell_i(\cdot)$: 度量模型拟合数据点 i 的程度(避免拟合不足)
- $\varphi(x)$: 避免过拟合的正则项: $\|x\|_2^2$ 或者 $\|x\|_1$ 等等
- $f(a, x)$: 线性函数或者由深度神经网络构造的模型

回归分析

- 一般的回归模型可以写成如下形式：

$$b = f(a) + \varepsilon, \quad (1)$$

$a \in \mathbb{R}^d$ 为自变量， $b \in \mathbb{R}$ 为响应变量， $\varepsilon \in \mathbb{R}$ 是模型误差或噪声

- 模型(1) 的含义为响应变量 b 与自变量 a 通过函数 f 联系在一起。
- 一般只知道 a 和 b 的观测值，而误差 ε 是未知的。建立回归模型的最终任务是利用 m 个观测值 (a_i, b_i) 来求解出 f 的具体形式，然后可以利用新观测的自变量对响应变量做出预测。
- 函数 f 取值于函数空间中，一般将其进行参数化，即模型(1) 为

$$b = f(a; x) + \varepsilon,$$

其中 $f(a; x)$ 的含义是 f 以 $x \in \mathbb{R}^n$ 为参数，通过选取不同的 x 得到不同的 f 。参数化的重要意义在于其将 f 选取的范围缩小到了有限维空间 \mathbb{R}^n 中，求解函数 f 的过程实际上就是求解参数 x 的过程。

线性回归模型

- 考虑线性模型 $b = Ax + \varepsilon$
- 假设 ε_i 是高斯白噪声，即 $\varepsilon_i \sim \mathcal{N}(0, \sigma^2)$ 。那么我们有

$$p(b_i | a_i; x) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(b_i - a_i^T x)^2}{2\sigma^2}\right),$$

则对数似然函数为

$$\ell(x) = \ln \prod_{i=1}^m p(b_i | a_i; x) = -\frac{m}{2} \ln(2\pi) - m \ln \sigma - \sum_{i=1}^m \frac{(b_i - a_i^T x)^2}{2\sigma^2}.$$

- 最大似然估计是极大化对数似然函数得到最小二乘问题：

$$\min_{x \in \mathbb{R}^n} \frac{1}{2} \|Ax - b\|_2^2.$$

注意，在构建最大似然估计时不需要知道 ε_i 的方差 σ^2 。

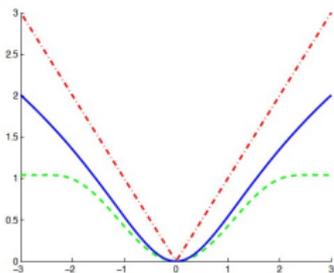
- 当假设误差是高斯白噪声时，最小二乘解就是线性回归模型的最大似然解。

线性回归模型

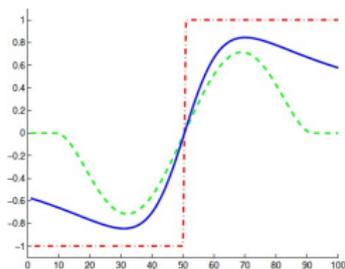
- 当 ε_i 不是高斯白噪声时，线性回归模型和最小二乘模型并不等价
- 在某些噪声下构造出的模型实际上为最小一乘问题：

$$\min_{x \in \mathbb{R}^n} \|Ax - b\|_1.$$

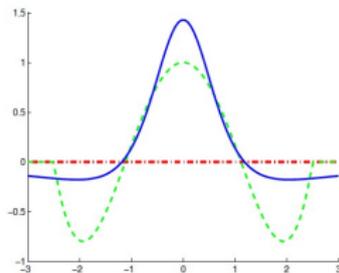
- 一般形式: $\min_{x \in \mathbb{R}^n} \rho(Ax - b)$



(a) $\rho(r)$



(b) $\nabla \rho(r)$



(c) $\nabla^2 \rho(r)$

Figure: Laplace (red dash-dot), Tukey (green dash), and Student's t (blue line).

Tikhonov 正则化

- 为了平衡模型的拟合性质和解的光滑性，Tikhonov 正则化或岭回归 (ridge regression) 添加 l_2 范数平方为正则项。
- 假设 ε_i 是高斯白噪声，则带 l_2 范数平方正则项的线性回归模型实际上是在求解如下问题：

$$\min_{x \in \mathbb{R}^n} \frac{1}{2} \|Ax - b\|_2^2 + \mu \|x\|_2^2.$$

由于正则项的存在，该问题的目标函数是强凸函数，解的性质得到改善。

- 另一种常见的变形是给定参数 $\sigma > 0$ ，求解：

$$\min_{x \in \mathbb{R}^n} \frac{1}{2} \|Ax - b\|_2^2, \quad \text{s.t.} \quad \|x\|_2 \leq \sigma. \quad (2)$$

- 上述两个问题的最优性条件类似，当参数 μ 和 σ 满足一定关系时，它们的解可以是相同的。

LASSO 问题及其变形

- 如果希望解 x 是稀疏的，可以考虑添加 l_1 范数为正则项得到LASSO问题：

$$\min_{x \in \mathbb{R}^n} \frac{1}{2} \|Ax - b\|_2^2 + \mu \|x\|_1,$$

其中 $\mu > 0$ 为已知的实数， x 是待估计的参数。LASSO问题通过惩罚参数的 l_1 范数来控制解的稀疏性

- 也可以考虑问题

$$\min_{x \in \mathbb{R}^n} \frac{1}{2} \|Ax - b\|_2^2, \quad \text{s.t.} \quad \|x\|_1 \leq \sigma.$$

- 考虑到噪声 ε 的存在，还可以给定 $\nu > 0$ 考虑模型：

$$\min_{x \in \mathbb{R}^n} \|x\|_1, \quad \text{s.t.} \quad \|Ax - b\|_2 \leq \nu.$$

- 后两个优化模型本质思想是相似的，即“在控制误差的条件下使得 x 的 l_1 范数尽量小”。但它们所属的优化问题种类实际上是不一样的

LASSO 问题及其变形

- 如果 ε 不是高斯白噪声，则需要根据具体类型选择损失函数，比如

$$\min_{x \in \mathbb{R}^n} \|Ax - b\|_2 + \mu \|x\|_1,$$
$$\min_{x \in \mathbb{R}^n} \|Ax - b\|_1 + \mu \|x\|_1.$$

上述两个模型和LASSO问题的差别在于对损失函数选择的范数不同，它们的性能可能很不一样。

- 当然损失函数项还有很多变化形式，如同时考虑 l_2 范数和 l_1 范数的组合，或选择Student- t 分布等

LASSO 问题及其变形

- 如果参数 x 具有分组稀疏性，即 x 的分量可分为 G 个组，每个组内的参数必须同时为零或同时非零，为此人们提出了分组LASSO模型：

$$\min_{x \in \mathbb{R}^n} \frac{1}{2} \|Ax - b\|_2^2 + \mu \sum_{\ell=1}^G \sqrt{n_\ell} \|x_{\mathcal{I}_\ell}\|_2,$$

其中 \mathcal{I}_ℓ 是属于第 ℓ 组变量的指标集且

$$|\mathcal{I}_\ell| = n_\ell, \quad \sum_{\ell=1}^G n_\ell = n.$$

- 当 $n_\ell = 1, \ell = 1, 2, \dots, G$ 时，问题就退化成传统的LASSO问题。
- 从分组LASSO问题正则项的设计可以看出，该正则项实际上是 $\|x_{\mathcal{I}_\ell}\|_2$ 的 ℓ_1 范数，因此只有少数 $\|x_{\mathcal{I}_\ell}\|_2$ 不为零。分组LASSO问题把稀疏性从单个特征提升到了组的级别上，但不要求组内的稀疏性

LASSO 问题及其变形

- 如果需要同时保证分组以及单个特征的稀疏性，可以考虑将两种正则项结合起来，即有稀疏分组LASSO模型：

$$\min_{x \in \mathbb{R}^n} \frac{1}{2} \|Ax - b\|_2^2 + \mu_1 \sum_{\ell=1}^G \sqrt{n_\ell} \|x_{\mathcal{I}_\ell}\|_2 + \mu_2 \|x\|_1,$$

其中 $\mu_1, \mu_2 > 0$ 为给定的常数。

- 稀疏分组LASSO模型的正则项由两部分组成： $\|x\|_1$ 是为了保证 x 本身的稀疏性，而 $\sum_{\ell=1}^G \sqrt{n_\ell} \|x_{\mathcal{I}_\ell}\|_2$ 是为了保证 x 的分组稀疏性

逻辑回归: logistic regression

- 对于二分类问题，预测变量只有两个取值，即 $-1, 1$ 。
- 给定特征 a ，逻辑回归假设这个样本属于类别1的概率

$$p(1|a; x) = P(t = 1 | a; x) = \theta(a^T x),$$

其中Sigmoid 函数

$$\theta(z) = \frac{1}{1 + \exp(-z)},$$

那么属于类别 -1 的概率

$$p(-1|a; x) = 1 - p(1 | a; x) = \theta(-a^T x).$$

因此对于 $b \in \{-1, 1\}$ ，上述概率可以简洁地写为

$$p(b | a; x) = \theta(b \cdot a^T x).$$

逻辑回归

- 假设数据对 $\{a_i, b_i\}, i = 1, 2, \dots, m$ 之间独立同分布，则在给定 a_1, a_2, \dots, a_m 情况下， b_1, b_2, \dots, b_m 的联合概率密度是

$$\begin{aligned} p(b_1, b_2, \dots, b_m \mid a_1, a_2, \dots, a_m; x) &= \prod_{i=1}^m p(b_i \mid a_i; x) \\ &= \frac{1}{\prod_{i=1}^m (1 + \exp(-b_i \cdot a_i^T x))}. \end{aligned}$$

- 最大似然估计是求解如下最优化问题：

$$\min_{x \in \mathbb{R}^n} \sum_{i=1}^m \ln(1 + \exp(-b_i \cdot a_i^T x)).$$

- 加上正则项，如 Tikhonov 和 ℓ_1 范数正则化模型：

$$\min_{x \in \mathbb{R}^n} \sum_{i=1}^m \ln(1 + \exp(-b_i \cdot a_i^T x)) + \lambda \|x\|_2^2$$

$$\min_{x \in \mathbb{R}^n} \sum_{i=1}^m \ln(1 + \exp(-b_i \cdot a_i^T x)) + \lambda \|x\|_1.$$

逻辑回归

- 假设数据对 $\{a_i, b_i\}$ 是由随机变量对 $\{\alpha, \beta\}$ 产生的，那么损失函数可以写成均值形式： $\mathbb{E}[\ln(1 + \exp(-\beta \cdot \alpha^T x))]$ 。而前面的问题可以写成下面随机优化问题的抽样形式：

$$\min_{x \in \mathbb{R}^n} \mathbb{E}[\ln(1 + \exp(-\beta \cdot \alpha^T x))] + \lambda r(x),$$

其中 $r(x)$ 为正则项， λ 为正则参数。

- 很多机器学习问题可以写成更一般的随机优化问题和它的离散版本：

$$\min_{x \in \mathbb{R}^n} f(x) + \lambda r(x),$$

其中

$$f(x) \stackrel{\text{def}}{=} \mathbb{E}[F(x, \xi)] = \int_{\Omega} F(x, \xi(\omega)) dP(\omega), \quad \text{或} \quad f(x) \stackrel{\text{def}}{=} \frac{1}{m} \sum_{i=1}^m f_i(x),$$

$\xi: \Omega \rightarrow W$ 是定义在给定概率空间 (Ω, \mathcal{F}, P) 上的随机变量， W 是可测空间， $F: \mathbb{R}^n \times W \rightarrow \mathbb{R}$ 以及 $f_i: \mathbb{R}^n \rightarrow \mathbb{R}, i = 1, 2, \dots, m$ 对应某个损失函数

支持向量机: support vector machine

- 给定训练数据集 D 中的样本点 (a_i, b_i) 且 $a_i \in \mathbb{R}^n$, $b_i \in \{-1, 1\}$, SVM的基本思想是找到一个超平面将 \mathbb{R}^n 中的样本点划分成两类.
- 我们先考虑一种简单情形: 假定训练数据集是线性可分的, 即正负两类刚好被划分到超平面 $x^T w + y = 0$ 两侧.

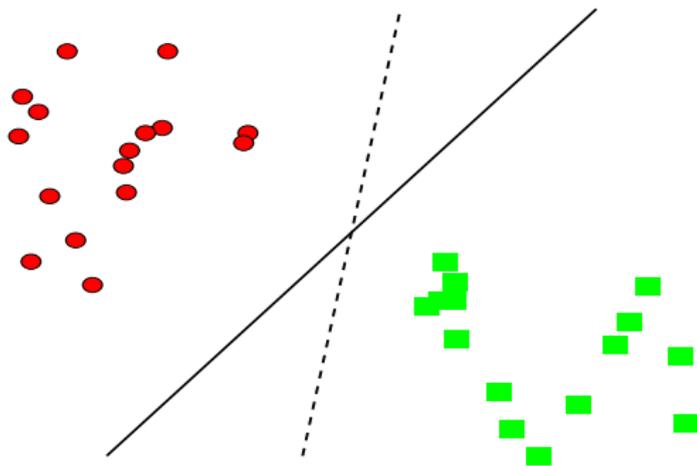


Figure: 线性可分的数据集

支持向量机

- 线性可分数据集比较理想的超平面：数据点距此平面的距离都比较远。使用这样的超平面的二分类模型有比较好的鲁棒性。
- 样本空间中一点 w 到超平面 $x^T w + y = 0$ 的距离

$$d = \frac{|x^T w + y|}{\|x\|_2}.$$

如果这个超平面对样本点 (a_i, b_i) 分类正确，则

$$b_i(a_i^T x + y) > 0, \quad i = 1, 2, \dots, m.$$

- 为了寻找理想的超平面来分离数据点，要求两类数据中的点到该超平面 $x^T w + y = 0$ 最小距离尽量大：

$$\begin{aligned} \max_{x, y, \gamma} \quad & \gamma, \\ \text{s.t.} \quad & \frac{b_i(a_i^T x + y)}{\|x\|_2} \geq \gamma, \quad i = 1, 2, \dots, m. \end{aligned}$$

- γ 表示所有样本点到超平面 $x^T w + y = 0$ 距离的最小值，而目标是将其最大化

支持向量机

- 注意到约束等价于

$$b_i(a_i^T x + y) \geq \gamma \|x\|_2, \quad i = 1, 2, \dots, m,$$

以相同正倍数对 x 和 y 进行放缩不会影响问题的约束和目标函数.

- 根据这个特点将可行域缩小, 强制取 $\|x\|_2 = \frac{1}{\gamma}$:

$$\begin{aligned} \max_{x,y,\gamma} \quad & \gamma, \\ \text{s.t.} \quad & b_i(a_i^T x + y) \geq \gamma \|x\|_2, \quad i = 1, 2, \dots, m, \\ & \|x\|_2 = \frac{1}{\gamma}. \end{aligned}$$

- 最终消去变量 γ 以及约束 $\|x\|_2 = \frac{1}{\gamma}$ 得到优化问题:

$$\begin{aligned} \min_{x,y} \quad & \frac{1}{2} \|x\|_2^2, \\ \text{s.t.} \quad & b_i(a_i^T x + y) \geq 1, \quad i = 1, 2, \dots, m. \end{aligned}$$

支持向量机

- 当线性可分的假设不成立时对每个数据点引入非负松弛变量 ξ_i ，允许有误差点。通过松弛变量的函数 $\sum_{i=1}^m \xi_i$ 来控制误分的程度：

$$\begin{aligned} \min_{x,y,\xi} \quad & \frac{1}{2} \|x\|_2^2 + \mu \sum_{i=1}^m \xi_i, \\ \text{s.t.} \quad & b_i(a_i^T x + y) \geq 1 - \xi_i, \\ & \xi_i \geq 0, \quad i = 1, 2, \dots, m, \end{aligned}$$

- 等价于无约束优化问题：

$$\min_{x,y} \quad \frac{1}{2} \|x\|_2^2 + \mu \sum_{i=1}^m \max\{1 - b_i(a_i^T x + y), 0\}.$$

- 当训练数据中含有冗余特征时，可以将 $\|x\|_2^2$ 项替换成 ℓ_1 范数求解：

$$\min_{x \in \mathbb{R}^n} \quad \|x\|_1 + \mu \sum_{i=1}^m \max\{1 - b_i(a_i^T x + y), 0\}.$$

稀疏协方差估计

- 给定 n 维高斯随机向量 $Y = (Y_1, Y_2, \dots, Y_n) \sim \mathcal{N}(\mu, \Sigma)$, $\mu \in \mathbb{R}^n$, $\Sigma \in \mathcal{S}_{++}^n$ 的一组实际取值 $\{y^1, y^2, \dots, y^m\}$, 其经验协方差矩阵为

$$S = \frac{1}{m} \sum_{i=1}^m (y^i - \bar{y})(y^i - \bar{y})^T,$$

其中 $\bar{y} = \frac{1}{m} \sum_{i=1}^m y^i$ 为样本均值.

- 关于精度矩阵 X 的对数似然函数为

$$\ell(X) = \ln \det(X) - \text{Tr}(XS), \quad X \succ 0,$$

其中 $X \succ 0$ 表示自变量 X 在正定矩阵空间取值.

- 通过最大化对数似然函数

$$\max_{X \succ 0} \ell(X),$$

我们可以得到精度矩阵 X 的估计.

稀疏协方差估计

- 这种方式得到的解往往不是稀疏的，也就意味着相应的概率图是全连接的（任意两个节点之间都存在直接相连的边），随机变量之间的相关性过密，解释性可能很差。
- 假设概率图中的边不是全连接的，我们建立如下的改进模型：

$$\max_{X \succ 0} \ell(X) - \lambda \|X\|_1,$$

其中 $\lambda > 0$ 是用来控制稀疏度的参数。上述模型得到的稀疏解可以用来估计高维随机变量之间的条件独立性

稀疏协方差估计

- 根据精度矩阵的定义，真实的精度矩阵 $\Theta = \Sigma^{-1}$ 。通过抽样的方式可以得到 Σ 的一个估计 S ，即经验协方差矩阵。
- 我们的目标是要估计 Σ^{-1} ，且使其具有稀疏结构，因此可设计如下优化问题：

$$\begin{aligned} \min_X \quad & \|SX - I\| + \lambda \|X\|_1, \\ \text{s.t.} \quad & X \succeq 0, \end{aligned}$$

其中 $\|\cdot\|$ 可以是任意一种范数（比较常用的为 F 范数或 ℓ_1 范数）， $X \succeq 0$ 表示 X 在半正定矩阵空间中取值。

- $\|SX - I\|$ 是要求 X 尽量为 S 的逆矩阵， $\|X\|_1$ 是要求 X 本身稀疏， $X \succeq 0$ 保证了求得的精度矩阵是半正定的

视频分离

- 将视频分成移动和静态的部分



矩阵分离问题: 稀疏和低秩矩阵

- 给定矩阵 M , 我们想找到一个低秩矩阵 W 和稀疏矩阵 E , 使得

$$W + E = M.$$

- 非凸模型:

$$\begin{aligned} \min_{W, E \in \mathbb{R}^{m \times n}} \quad & \text{rank}(W) + \mu \|E\|_0, \\ \text{s.t.} \quad & W + E = M. \end{aligned}$$

- 凸松弛:

$$\min_{W, E} \|W\|_* + \mu \|E\|_1, \text{ s.t. } W + E = M$$

- 鲁棒主成分分析

2014 ACM SIGMOD Programming Contest

- Shortest Distance Over Frequent Communication Paths

定义社交网络的边: 相互直接至少有 x 条回复并且相互认识。给定网络里两个人 $p1$ 和 $p2$ 以及另外一个整数 x , 寻找图中 $p1$ 和 $p2$ 之间数量最少节点的路径

- Interests with Large Communities

- Socialization Suggestion

- Most Central People (All pairs shorted path)

定义网络: 论坛中有标签 t 的成员, 相互直接认识。给定整数 k 和标签 t ,寻找 k 个有highest closeness centrality values的人

DIMACS Implementation Challenges

<http://dimacs.rutgers.edu/Challenges/>

- 2014, 11th: Steiner Tree Problems
- 2012, 10th: Graph Partitioning and Graph Clustering
- 2005, 9th: The Shortest Path Problem
- 2001, 8th: The Traveling Salesman Problem
- 2000, 7th: Semidefinite and Related Optimization Problems
- 1998, 6th: Near Neighbor Searches
- 1995, 5th: Priority Queues, Dictionaries, and Multi-Dim. Point Sets
- 1994, 4th: Two Problems in Computational Biology: Fragment Assembly and Genome Rearrangements
- 1993, 3rd: Effective Parallel Algorithms for Combinatorial Problems
- 1992, 2nd: Maximum Clique, Graph Coloring, and Satisfiability
- 1991, 1st: Network Flows and Matching

相位恢复

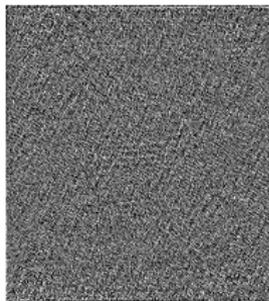
问题: 是否可以在不知道相位的情况下恢复信号



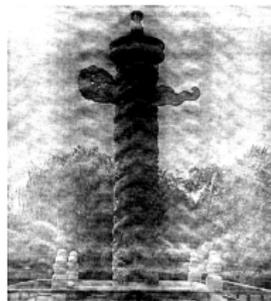
(a) Y



(b) $|\mathcal{F}(Y)|$



(c) $\text{phase}(\mathcal{F}(Y))$



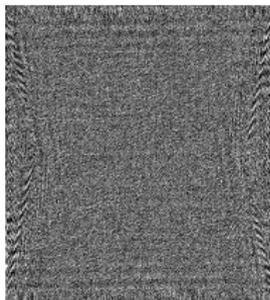
(d) S



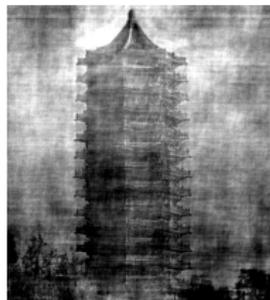
(e) S



(f) $|\mathcal{F}(S)|$



(g) $\text{phase}(\mathcal{F}(S))$



(h) \hat{Y}

传统相位恢复

可行性问题

寻找 $x \in S \cap \mathcal{M}$ 或 寻找 $x \in S_+ \cap \mathcal{M}$

- 给定傅里叶变换强度(Fourier magnitudes):

$$\mathcal{M} := \{x(r) \mid |\hat{x}(\omega)| = b(\omega)\}$$

其中 $\hat{x}(\omega) = \mathcal{F}(x(r))$, \mathcal{F} 为傅里叶变换

- 给定支撑集估计(support estimate):

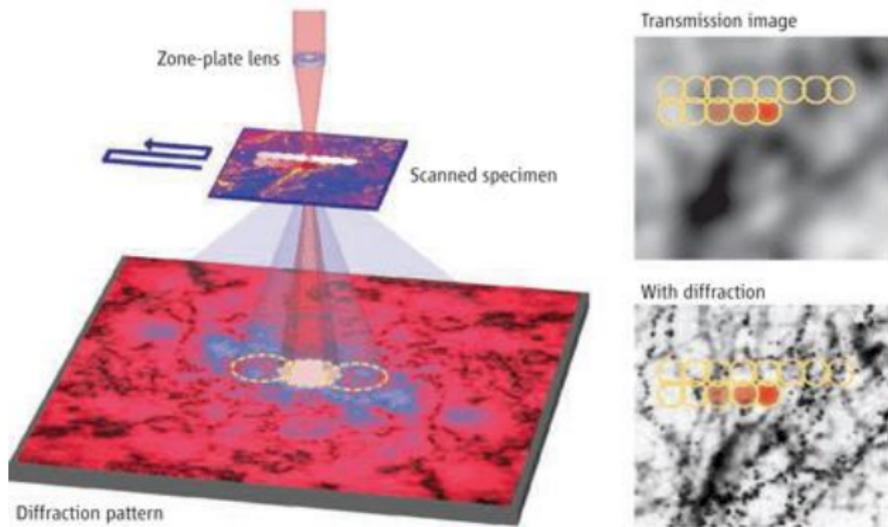
$$S := \{x(r) \mid x(r) = 0 \text{ for } r \notin D\}$$

或

$$S_+ := \{x(r) \mid x(r) \geq 0 \text{ and } x(r) = 0 \text{ if } r \notin D\}$$

叠层成像技术(Thanks: Chao Yang)

给定 $b_i = |\mathcal{F}(Q_i\psi)|$ ($i = 1, \dots, k$), 是否能恢复 ψ ?



叠层成像技术(Ptychography)以及检测器和计算机的进步,使得X射线显微镜无需使用镜头即可提高空间分辨率。

现代相位恢复模型

给定复信号 $x = (x_0, x_1, x_2, \dots, x_{n-1})^T \in \mathbb{C}^n$ 以及采样数 m ，我们可以逐分量定义如下线性变换：

$$(\mathcal{A}(x))_k = \langle a_k, x \rangle, \quad k = 1, 2, \dots, m,$$

其中 $a_k \in \mathbb{C}^n$ 为已知复向量。容易验证当该线性变换 \mathcal{A} 为离散傅里叶变换时， a_k 有如下形式：

$$a_k = \left(e^{2\pi i \frac{k-1}{n} t} \right)_{t=0}^{n-1}, \quad k = 1, 2, \dots, n.$$

针对一般形式的 a_k ，如果将其对应的振幅观测记为 b_k ，那么相位恢复问题本质上是求解如下的二次方程组：

$$b_k^2 = |\langle a_k, x \rangle|^2, \quad k = 1, 2, \dots, m. \quad (3)$$

虽然求解线性方程组很简单，但是求解二次方程组问题却是NP难的。

最小二乘模型

比较常见的模型是将问题(3)转化为非线性最小二乘问题：

$$\min_{x \in \mathbb{C}^n} \sum_{i=1}^m (|\langle a_i, x \rangle|^2 - b_i^2)^2. \quad (4)$$

这个模型的目标函数是可微（Wirtinger 导数）的四次函数，是非凸优化问题。相较于问题(3)，模型(4)能够更好地处理观测中自带的噪声。在实际中，我们也常常构造以下非光滑模型：

$$\min_{x \in \mathbb{C}^n} \sum_{i=1}^m (|\langle a_i, x \rangle| - b_i)^2. \quad (5)$$

假设 a_i 和 x 均为实的，我们可以得到相应的实数情况下的模型：

$$\min_{x \in \mathbb{R}^n} \sum_{i=1}^m (|\langle a_i, x \rangle|^2 - b_i^2)^2, \quad (6)$$

$$\min_{x \in \mathbb{R}^n} \sum_{i=1}^m (|\langle a_i, x \rangle| - b_i)^2. \quad (7)$$

相位提升

相位提升(phase lift)是求解相位恢复问题的另一种方法. 注意到

$$|\langle a_i, x \rangle|^2 = \bar{a}_i^T x \bar{x}^T a_i = \text{tr } x \bar{x}^T a_i \bar{a}_i^T,$$

令 $X = x \bar{x}^T$, 方程组(3) 可以转化为

$$\text{tr } X a_i \bar{a}_i^T = b_i^2, \quad i = 1, 2, \dots, m, \quad X \succeq 0, \quad \text{rank}(X) = 1. \quad (8)$$

如果方程组(3) 的解 x 存在, 那么 $X = x \bar{x}^T$ 就为方程组(8) 的解. 对于方程组(8), 我们考虑优化问题

$$\begin{aligned} \min_X \quad & \text{rank}(X), \\ \text{s.t.} \quad & \text{tr}(X a_i \bar{a}_i^T) = b_i^2, \quad i = 1, 2, \dots, m, \\ & X \succeq 0. \end{aligned} \quad (9)$$

把向量变量转化为矩阵变量的操作被称为“提升”, 使用“提升”的目的是将约束从关于向量 x 的二次函数转化为关于矩阵 X 的线性函数。

相位提升

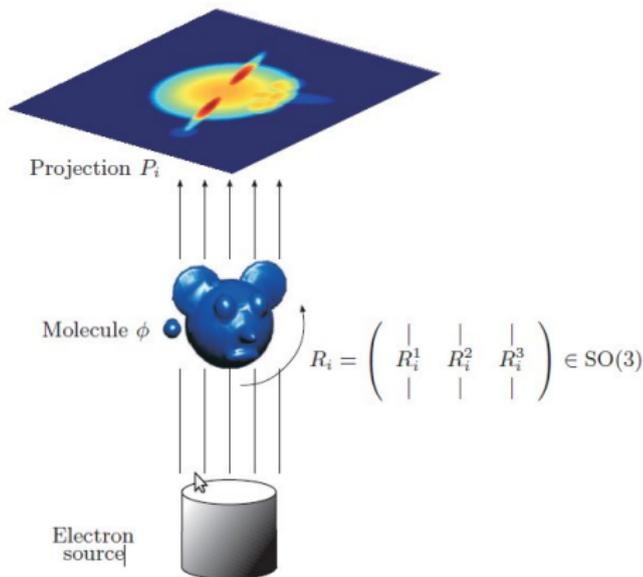
因为秩优化的计算复杂性，我们采用核范数对其进行松弛，得到如下优化问题：

$$\begin{aligned} \min_X \quad & \text{tr}(X), \\ \text{s.t.} \quad & \text{tr}(Xa_i\bar{a}_i^T) = b_i^2, \quad i = 1, 2, \dots, m, \\ & X \succeq 0, \end{aligned} \tag{10}$$

其中 $\text{tr} X = \|X\|_*$ 来自于矩阵 X 的半正定性。

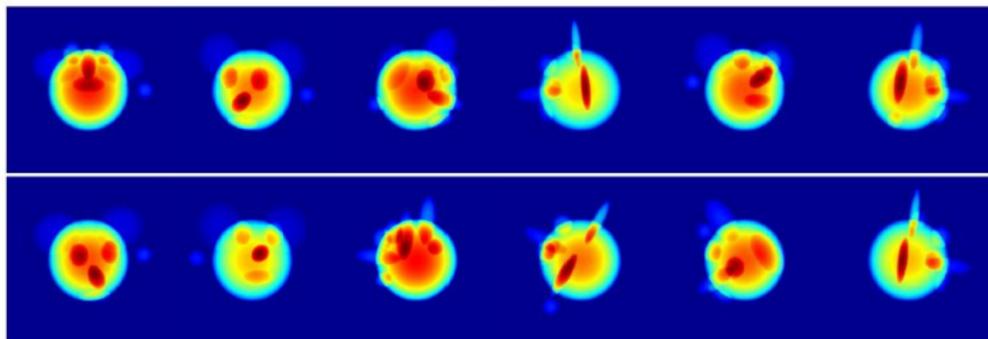
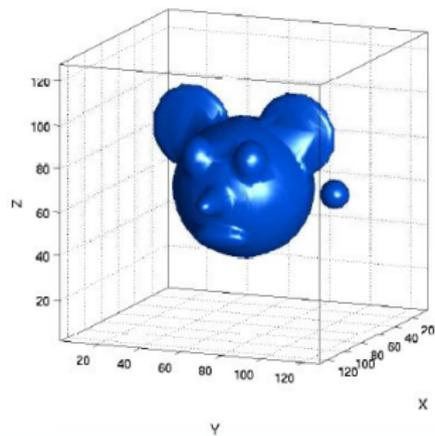
- 当问题(10)存在秩一解时，原始相位恢复问题的解可以通过秩一分解得到。
- 当 $m \geq c_0 n \ln n$ (c_0 为一个问题相关的常数) 时，问题(10)的解在高概率下是秩一的。

单粒子冷冻电镜



- 投影图像 $P_i(x, y) = \int_{-\infty}^{\infty} \phi(xR_i^1 + yR_i^2 + zR_i^3) dz + \text{噪声}$.
- $\phi: \mathbb{R}^3 \rightarrow \mathbb{R}$ 为分子的电势
- Cryo-EM 问题: 对于给定的 P_1, \dots, P_n , 寻找 ϕ 以及 R_1, \dots, R_n

例子



单粒子冷冻电镜

- 最小二乘模型

$$\min_{R_1, \dots, R_K \in \mathbf{SO}(3)} \sum_{i \neq j} w_{ij} \left\| R_i (\vec{c}_{ij}, 0)^T - R_j (\vec{c}_{ji}, 0)^T \right\|^2$$

- 因为 $\left\| R_i (\vec{c}_{ij}, 0)^T \right\| = \left\| R_j (\vec{c}_{ji}, 0)^T \right\| = 1$, 我们得到等价优化问题

$$\max_{R_1, \dots, R_K \in \mathbf{SO}(3)} \sum_{i \neq j} w_{ij} \langle R_i (\vec{c}_{ij}, 0)^T, R_j (\vec{c}_{ji}, 0)^T \rangle$$

- 半定松弛:

$$\begin{aligned} \max_{G \in \mathbb{R}^{2K \times 2K}} \quad & \text{tr}((W \circ S)G) \\ \text{s.t.} \quad & G_{ii} = I_2, \quad i = 1, 2, \dots, K, \\ & G \succeq 0 \end{aligned}$$

随机线性代数算法

(Thanks: Petros Drineas and Michael W. Mahoney)

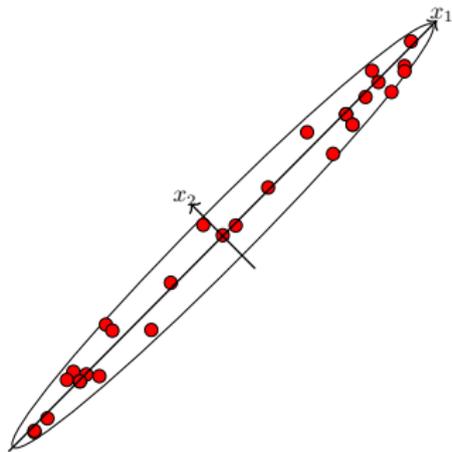
目标：针对大规模的矩阵和张量计算，发明快速的蒙特卡洛算法。

- 矩阵乘法
- 矩阵SVD分解
- 矩阵CUR分解
- 线性规划可行性问题
- 最小二乘逼近
- 张量计算：SVD的推广
- 张量计算：CUR的推广

这些计算需要的时间关于矩阵/张量的非零元个数通常是超线性的，例如对 $n \times n$ 的矩阵，这些算法通常需要 $O(n^3)$ 的时间复杂度。

主成分分析

主成分分析是数据处理和降维中的一个重要技巧，它提供了一种将高维空间中的点在低维子空间中表达的方法。



假设我们需要将 \mathbb{R}^p 中的数据点集 $\{a_i\}_{i=1}^n$ 投影到 \mathbb{R}^p 的一个 d 维子空间($d < p$)中。记 $X \in \mathbb{R}^{p \times d}$ 为该子空间的标准正交基形成的列正交矩阵。易知数据点 a_i 在 X 张成的子空间的投影为 $\mathcal{P}_X(a_i) \stackrel{\text{def}}{=} XX^T a_i$ 。

主成分分析

我们需要寻找最优的 X ，使得投影后的数据点集 $\{\mathcal{P}_X(a_i)\}$ 的方差最大。根据零均值假设，投影后数据点集的协方差矩阵为

$$\frac{1}{n} \sum_{i=1}^n XX^T a_i (XX^T a_i)^T = \frac{1}{n} XX^T AA^T XX^T,$$

而多元分布的方差大小可由协方差矩阵的迹来刻画，因此，得到优化问题

$$\max \operatorname{tr}(X^T AA^T X), \quad \text{s.t.} \quad X^T X = I, \quad (11)$$

其中利用了迹的性质

$$\operatorname{tr}(XX^T AA^T XX^T) = \operatorname{tr}(X^T AA^T XX^T X) = \operatorname{tr}(X^T AA^T X).$$

我们再从重构误差的角度来理解主成分分析模型，我们用投影 $\mathcal{P}_X(a_i)$ 来表示 a_i ，则该点的重构误差为 $\|XX^T a_i - a_i\|_2$ 。定义所有点的重构误差平方和为

$$\sum_{i=1}^n \|XX^T a_i - a_i\|_2^2 = \|XX^T A - A\|_F^2 = -\operatorname{tr}(X^T AA^T X) + \operatorname{tr}(A^T A),$$

主成分分析

我们需要寻找最优的 X ，使得重构误差平方和最小，即求解优化问题

$$\min -\operatorname{tr}(X^T A A^T X) + \operatorname{tr}(A^T A), \quad \text{s.t.} \quad X^T X = I,$$

此形式和(11)是等价的。因此我们得到结论：主成分分析寻找方差最大的子空间投影，实际上是极小化投影点的重构误差。

字典学习

字典学习的目的就是将已有的（超）大规模的数据集进行压缩，找到蕴藏在这些数据点背后的最基本的原理。考虑一个 m 维空间中的数据集 $\{a_i\}_{i=1}^n, a_i \in \mathbb{R}^m$ ，假定每个 a_i 都是由同一个字典生成的，且生成之后的数据带有噪声，因此字典学习的线性模型可以表示为

$$a = Dx + e,$$

这里 $D \in \mathbb{R}^{m \times k}$ 是某个未知的字典，它的每一列 d_i 是字典的一个基向量； x 是字典中基的系数，同样是未知的； e 是某种噪声。当 e 是高斯白噪声时，可以定义损失函数

$$f(D, X) = \frac{1}{2n} \|DX - A\|_F^2,$$

为了控制 X 的稀疏度，引入正则项 $\|X\|_1$ ，最终得到的优化问题为

$$\begin{aligned} \min_{D, X} \quad & \frac{1}{2n} \|DX - A\|_F^2 + \lambda \|X\|_1, \\ \text{s.t.} \quad & \|D\|_F \leq 1. \end{aligned} \tag{12}$$

K-均值聚类

聚类(clustering)不同于分类(classification), 在聚类问题中我们仅仅知道数据点本身, 而不知道每个数据点具体的标签。聚类分析的任务就是将一些无标签的数据点按照某种相似度来进行归类, 进而从数据点本身来学习其内蕴的类别特征。

给定 p 维空间中 n 个数据点 a_1, a_2, \dots, a_n , 聚类问题就是要寻找 k 个不相交的非空集合 S_1, S_2, \dots, S_k , 使得

$$\{a_1, a_2, \dots, a_n\} = S_1 \cup S_2 \cup \dots \cup S_k,$$

并且使得组内距离平方和最小, 即

$$\begin{aligned} \min_{S_1, S_2, \dots, S_k} \quad & \sum_{i=1}^k \sum_{a \in S_i} \|a - c_i\|^2, \\ \text{s.t.} \quad & S_1 \cup S_2 \cup \dots \cup S_k = \{a_1, a_2, \dots, a_n\}, \\ & S_i \cap S_j = \emptyset, \quad \forall i \neq j, \end{aligned} \tag{13}$$

K-均值聚类的等价表述一

为了表示聚类方式 S_1, S_2, \dots, S_k , 一个很自然的想法是使用一个向量 $\phi_i \in \mathbb{R}^k$ 来表示点 a_i 所处的类别. 具体地, 定义 ϕ_i 为

$$(\phi_i)_j = \begin{cases} 1, & a_i \in S_j, \\ 0, & a_i \notin S_j, \end{cases}$$

则聚类问题可以等价描述为

$$\begin{aligned} \min_{\Phi, H} \quad & \|A - \Phi H\|_F^2, \\ \text{s.t.} \quad & \Phi \in \mathbb{R}^{n \times k}, \text{ 每一行只有一个元素为1, 其余为0,} \\ & H \in \mathbb{R}^{k \times p}. \end{aligned} \quad (14)$$

在这里 Φ 的第 i 行的向量就是 ϕ_i^T .

K-均值聚类的等价表述二

首先定义 $\mathbf{1}_{S_t}$, $1 \leq t \leq k$ 为 n 维空间中每个分量取值0或1的向量, 且

$$\mathbf{1}_{S_t}(i) = \begin{cases} 1, & a_i \in S_t, \\ 0, & a_i \notin S_t. \end{cases}$$

可以证明, 第 t 类 S_t 中每个点到其中心点的距离平方和可以写成 $\frac{1}{2n_t} \text{tr}(D\mathbf{1}_{S_t}\mathbf{1}_{S_t}^T)$, 其中 $D \in \mathbb{R}^{n \times n}$ 的元素为 $D_{ij} = \|a_i - a_j\|^2$.

因此, 我们将问题(13)转化为

$$\begin{aligned} & \min_{S_1, S_2, \dots, S_k} \frac{1}{2} \text{tr}(DX), \\ & \text{s.t.} \quad X = \sum_{t=1}^k \frac{1}{n_t} \mathbf{1}_{S_t} \mathbf{1}_{S_t}^T, \\ & \quad S_1 \cup S_2 \cup \dots \cup S_k = \{a_1, a_2, \dots, a_n\}, \\ & \quad S_i \cap S_j = \emptyset, \quad \forall i \neq j. \end{aligned} \tag{15}$$

K-均值聚类的等价表述二

对半定矩阵 X 进行分解 $X = YY^T$, $Y \in \mathbb{R}^{n \times k}$, 我们可以进一步得到如下矩阵优化问题 (这里 $\mathbf{1}$ 是 n 维向量且分量全为1) :

$$\begin{aligned} \min_{Y \in \mathbb{R}^{n \times k}} \quad & \text{tr}(Y^T D Y), \\ \text{s.t.} \quad & Y Y^T \mathbf{1} = \mathbf{1}, \\ & Y^T Y = I_k, Y \geq 0. \end{aligned} \tag{16}$$

图像处理中的全变差模型

令 $b(x, y)$ 是观测到的带噪声的图像, \mathcal{A} 是线性算子. 在经典的Rudin-Osher-Fatemi(ROF)模型下, 图像去噪和去模糊问题可以写成

$$\min_u \|\mathcal{A}u - b\|_{L_2}^2 + \lambda \|u\|_{TV}, \quad (17)$$

这里, 定义域为 Ω 的函数 f 的 L_2 范数定义为

$$\|f\|_{L_2} = \left(\int_{\Omega} f^2 dx \right)^{1/2}.$$

全变差定义为

$$\|u\|_{TV} = \int_{\Omega} \|Du\| dx, \quad (18)$$

目标函数中的第一项是数据保真项, 即重构出的图片要与已有的采集信息相容. 第二项是正则项, 用来保证重构出的图像的阶跃是稀疏的, 或者说使得重构出的图像类似于一个分片常数函数.

图像处理中的全变差模型



Figure: 图像去噪

运用合适的离散格式处理后，我们可得到离散的线性算子 \mathcal{A} 和图像 B （这里沿用了连续情形的记号，但 \mathcal{A} 的含义完全不同）。因此由连续问题(17)得到离散问题：

$$\min_{U \in \mathbb{R}^{n \times n}} \|\mathcal{A}U - B\|_F^2 + \lambda \|U\|_{TV}. \quad (19)$$

图像处理中的全变差模型

在实际中，除了考虑ROF模型外，我们还考虑其一个变形，TV- L^1 模型。离散格式为

$$\min_{U \in \mathbb{R}^{n \times n}} \|AU - B\|_1 + \lambda \|U\|_{TV}. \quad (20)$$

上述模型的一个好处是可以更好地处理非高斯噪声的情形，比如椒盐噪声等。

图像处理中还有大量其他问题，如图像识别，图像分割，图像匹配，生物医学图像（CT，MRI，fMRI）处理，计算机图像和视觉，遥感图像，等等。它们中间产生了丰富的优化问题。

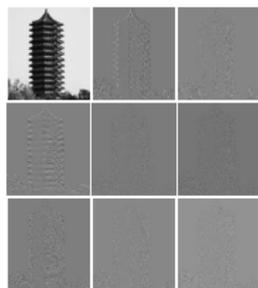
小波模型

小波框架 $W \in \mathbb{R}^{m \times n}$

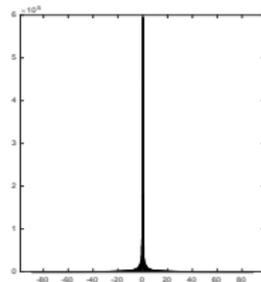
- 目的：在小波框架下，可以对图像 x 做分解得到小波系数 $\alpha \in \mathbb{R}^m$ ，即 $\alpha = Wx$ 。反之，给定小波系数 α ，可以重构出图像 $x = W^T \alpha$ 。
- 为了保证重构的完整性，我们要求 $W^T W = I$ 。
- 冗余性： m 可以比 n 大，但是有 $\text{rank}(W) = n$ ，并且 $W W^T \neq I$ 。
- 如下图所示，小波系数通常具有稀疏性



(a) 原图



(b) 小波系数



(c) 稀疏的系数

Figure: 小波分解

小波模型

- 分解模型：直接求解重构图像，其通过惩罚图像的小波系数的 l_1 范数来去除图像中不必要的噪声信息。问题形式为

$$\min_{x \in \mathbb{R}^n} \|\lambda \odot (Wx)\|_1 + \frac{1}{2} \|Ax - b\|_2^2, \quad (21)$$

其中 $\lambda \in \mathbb{R}^m$ 是给定的非负向量， \odot 表示逐个分量相乘。

- 合成模型：求解图像对应的小波系数来重构图像，其通过小波系数的 l_1 范数来去除图像中不必要的噪声信息。问题形式为

$$\min_{\alpha \in \mathbb{R}^m} \|\lambda \odot \alpha\|_1 + \frac{1}{2} \|AW^T \alpha - b\|_2^2. \quad (22)$$

- 平衡模型：在合成模型中， α 不一定对应于真实图像的小波系数。因此，平衡模型添加 $(I - WW^T)\alpha$ 的二次罚项来保证 α 更接近真实图像的小波系数。问题形式为

$$\min_{\alpha \in \mathbb{R}^m} \|\lambda \odot \alpha\|_1 + \frac{1}{2} \|AW^T \alpha - b\|_2^2 + \frac{\kappa}{2} \|(I - WW^T)\alpha\|_2^2, \quad (23)$$

其中 κ 为给定常数。

强化学习

强化学习们一般可以抽象出智能体(agent)和环境(environment)两个概念. 智能体持续地与环境互动并从环境中学到经验和规则, 如下图所示: 智能体在状态 s_t 下执行动作 a_t 后, 环境根据其内在的规则回应, 到达新的状态 s_{t+1} , 奖励 r_{t+1} . 这个系统持续不断地重复这个过程, 直到系统中止.

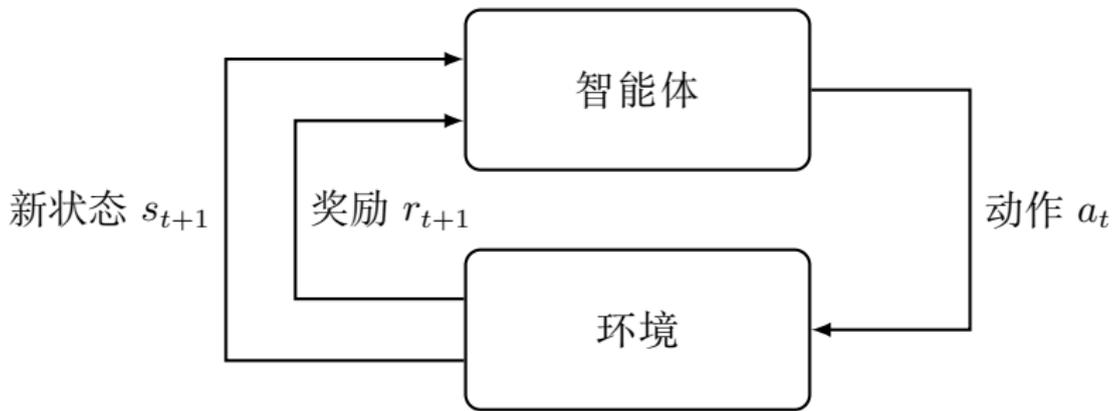


Figure: 强化学习智能体-环境交互图

强化学习

强化学习跟其他机器学习相比有如下不同点：

- 这个过程是无监督的。没有标签告诉智能体做什么动作是最好的，只有之前动作所获得的奖励会让智能体更偏向于执行某一类动作。
- 环境给智能体动作的反馈是有延迟的。当前动作的效果也许不会立刻体现，但是它可能影响许多步后的奖励。
- 时间顺序在强化学习中是非常重要的。所做决策的顺序将会决定最终的结果。
- 智能体所做的动作会影响观察到的环境状态。在这个学习过程中观察到的环境状态或接收到的反馈不是独立的，它们是智能体动作的函数。这一点与监督学习中的样本独立性假设有很大差别。

强化学习

强化学习经常可以用马尔可夫决策过程(Markov decision process, MDP)来描述. 在马尔可夫决策过程中, 环境状态转移的概率只取决于当前的状态和动作, 而与所有历史信息无关, 即

$$P(s_{t+1} = s' | s_t, a_t, s_{t-1}, a_{t-1}, \dots, s_0, a_0) = P(s_{t+1} = s' | s_t, a_t).$$

其次, 环境所反馈的奖励的期望也只依赖于当前的状态和动作, 所以该期望可以表示成

$$\mathbb{E}[r_{t+1} | s_t, a_t, s_{t-1}, a_{t-1}, \dots, s_0, a_0] = \mathbb{E}[r_{t+1} | s_t = s, a_t = a] = r(s, a).$$

智能体要做的是通过在环境中不断尝试学习得到一个策略(policy) π . 假设 S 和 A 为正整数, 令 $S = \{1, 2, \dots, S\}$ 和 $A = \{1, 2, \dots, A\}$ 分别为状态空间和决策空间. 我们要学习的策略事实上是从状态空间 S 到动作空间的概率分布集合 Δ_A 的映射, $\pi(a|s)$ 表示使用策略 π 时, 在当前状态 s 下选择决策 a 的概率.

强化学习

设轨道 τ 是一系列的状态和动作的集合 $\tau = \{s_0, a_0, s_1, a_1, \dots\}$.
给定折扣因子 $\gamma \in [0, 1)$, 有折扣累积奖励可以表示为

$$R(\tau) = \sum_{t=0}^{\infty} \gamma^t r_t,$$

其中 $r_t = r(s_t, a_t)$. 那么最优的策略是能最大化MDP收益的策略:

$$\max_{\pi} \mathbb{E}_{\tau \sim \pi} [R(\tau)], \quad (24)$$

其中 $\tau \sim \pi$ 表示轨道 τ 是按照策略 π 生成的. 令 $V(i)$ 为最优策略在任意时刻从状态 i 出发得到的期望奖励, 那么问题(24)也等价于求解

$$V(i) = \max_a \left\{ \sum_j P_a(i, j) (r(i, a) + \gamma V(j)) \right\}. \quad (25)$$

其中 $P: \mathcal{S} \times \mathcal{A} \rightarrow \Delta_{\mathcal{S}}$ 为状态转移算子, 用 $P_a(i, j)$ 表示采用决策 a 从状态 i 跳到状态 j 的概率.

强化学习

上述方程称为Bellman方程. 在任意时刻, 状态 i 处的决策 $a(i)$ 应满足:

$$a(i) = \operatorname{argmax}_a \left\{ \sum_j P_a(i,j) (r(i,a) + \gamma V(j)) \right\}.$$

在这里指出, $V(i)$ 与 $a(i)$ 都不依赖于时间 t , 其根本原因是下一步的状态仅依赖于当前状态和动作(马尔可夫性), 且转移概率和奖励函数也与时间无关(时齐性).

实际问题中, 状态集合或者动作集合中元素的数量可能非常多. 常用的一个方式是将策略用神经网络或者深度神经网络来表达, 它依赖于一系列参数, 因此问题(24)也可以表达为:

$$\max_{\theta} \mathbb{E}_{\tau \sim \pi_{\theta}} [R(\tau)]. \quad (26)$$