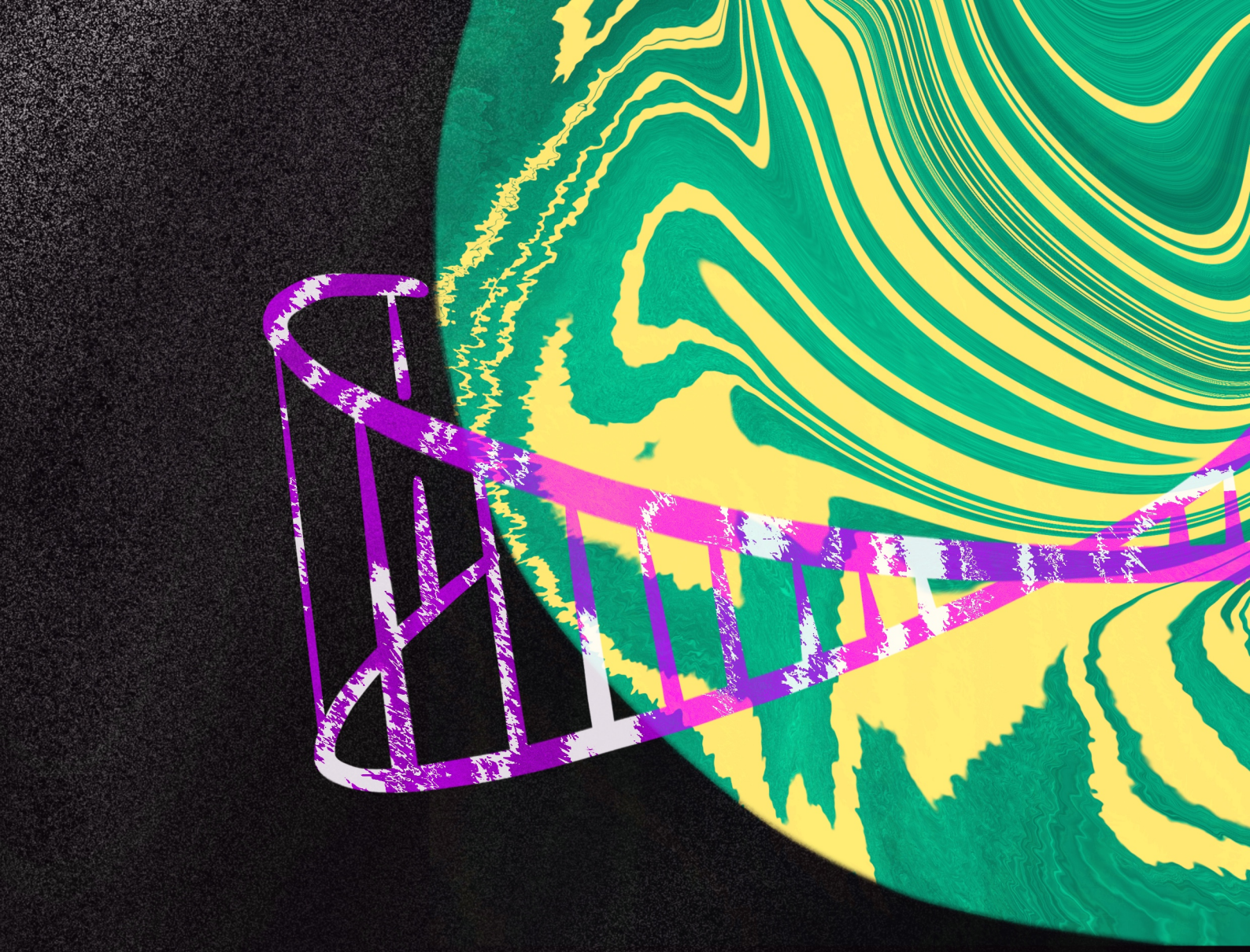

Machine Learning Lecture Note

Lin Ning

机器学习复习笔记 [1]

林宁[2]

November 26, 2022

[1]参考模版：https://github.com/amberj/latex-book-template

[2]邮箱：linning51400@ruc.edu.cn

谨以此书献给可爱的「高嶺の花」

目录

前言

比希望更炙热的，比绝望更深邃的，是 AI 啊。

– 自然对数

本书为中国人民大学高瓴人工智能学院在 2022 年春季开设的机器学习基础课程中相关内容的注解的整合，笔者为高瓴人工智能学院 2020 级本科生林宁，指导老师为本课程的开课教师许洪腾老师。机器学习是我们作为人工智能专业学生的基础课和必修课，其对于我们将来更加深入地学习人工智能理论的重要性不必言说。在本门课程中许老师对机器学习的基础内容进行了精心的编排整合进了课程对应的课件中，老师所花费的时间和精力、课件体现的广度和深度远远超出了笔者和这篇笔记。希望大家能够用心体验这门课程，体会许老师的良苦用心。本人也在老师的基础上也进行了一些个人主观想法的加工，希望我的观点能够让大家在阅读的过程中有所收获，对机器学习有更加深刻的认识。

使用说明

- 本书内容完全以许洪腾老师的提供的课件为基础进行相应的补充，内容零碎，请大家一定要以许老师的课件和参考教材为主。
- 本书部分内容与许老师的想法存在冲突，鉴于本人基础薄弱、水平有限，且本书赶工迹象严重，未经过严格的审核与考证，请大家对于冲突的观点谨慎采纳。
- 本书部分内容涉及对网络资料的大面积的借鉴和部分个人加工和整理，建议大家搭配本文提供的参考文献使用。参考资料中的大部分的证明思路清晰、启发性强，但是仍不可避免地存在部分证明稍显繁琐，且存在关键性证明被作者略去的情况。本文将沿着作者思路，向我们这一阶段的读者提供更加友好的证明。

- 本书将在必要时省略对于大部分前置课程涉及到的教材中均有提供的结论的证明，所以建议大家在阅读前温习一下前置课程高等数学、线性代数以及概率论中的相关内容，以获取更好的阅读体验。
- 本书采用的符号体系与课件一致，故本书不提供符号表。特别注意的是由于课件上矩阵的布局方式并未统一，这里有必要对本书的几点加以说明：
  - 不加粗字母代表标量，加粗字母代表向量或矩阵。
  - 矩阵布局均采用列向量布局的形式，即列向量在行的对应的轴上进行堆叠形成矩阵；小写字母加下标代表了下标对应的矩阵列向量，大写字母加下标代表了下标对应的矩阵行向量。使用这一规定的原因为线性代数的习惯。
  - 对角矩阵的幂运算代表的分别对对角线上的元素进行相应的运算，在这本书中可能会遇到幂为分数的情况。
  - 分号右侧代表了模型参数。符号 $p(\cdot \mid \boldsymbol{\theta})$，$p(\cdot\,;\boldsymbol{\theta})$ 和 $p_{\boldsymbol{\theta}}(\cdot)$ 代表的是同一个意思。使用这一规定的原因为 MLE 认为数据的分布中参数是一个常量，因而在非 Bayesian 统计的推导时参数是固定的，很多时候也不需要太关注，因而单独列出来。

鸣谢

感谢许洪腾老师提供的建设性意见！您在这本书编写的过程中真是给我提供了很多非常好的点子，完美地解决了我很多疑惑，没有你这本书的深度一定会减少很多。许老师谦虚地说自己原来是学电子通讯的，不是本专业出生的，路子很野，写的内容带有很强的个人理解，但是我觉得这恰恰是这门课启发性之所在。许老师给我们提供了很多崭新的思考问题的方向，尽管这样的思想可能也有一些不够完善之处，但足以给我们带来在专业的参考教材以外的新的启发。其实这本书也掺了我自己的很多私货，这也是我一开始不敢把笔记分享给同学们的原因，因为我真的担心这些私货会给大家带来很多的困扰，但是在老师的鼓励下我终于有勇气把笔记发给大家。我也希望这本书能够像许老师的课一样给大家带来新的启发！

感谢三明治亲制作的可爱的封面！你是这本书编写的 motivation，所以拜托你制作这本书的封面真是再合适不过了。我一直觉得整理机器学习这门

课的笔记是一个很大的挑战，原因在于知识点过于细碎而参考资料良莠不齐，缺乏能快速入手的途径，但是当你抱怨机器学习这门课要把你击垮的时候，我觉得自己也不是做不到。当时自己只整理了和 PCA 相关的内容，还有很多内容没有着手去了解，于是自己开始尝试基于课件以注释的方式进行细碎的知识点标注和证明补充，后来发现自己写的内容越来越像是一篇相对比较系统的 survey，自己写得很多内容也超出了课件所覆盖的范围。在这短短的一个多月内能够将笔记整理到这种程度我已经很满意了，但是等到我最近开始对细碎的笔记的内容进行进一步地筛选和整合以后，我发现自己写的很多内容还是太跳跃了。所以我对这份笔记最后是不是真的帮到了你不太有自信，如果真的帮助你度过了这道难关的话那真是太好了，那说明我起码没有偏离初心，哈哈。

感谢 Lucy 君在证明的严谨性上提供的建议！虽然有些问题提得比较尖锐，但是如果没有你那么认真的读者我这边审稿的压力会大很多。事实上我知道的，我们都对证明的严谨性有一种偏执，所以希望你以后能够多多给我写的内容 “挑刺”，多提一些尖锐的问题，我也会更加全副武装地应对它，当然了把矩阵的列向量改成行向量布局我是绝对不会接受的，这点是原则问题 (笑)。也希望我们以后继续保持密切交流，共同进步！还有很多方面我要向你学习的，也请你不吝赐教啦。

感谢戴姐姐自大二上以来坚持不懈地将我的笔记上传到 github 上去，这对我的工作的宣传帮助非常大！也感谢你一直以来对我的鼓励，你对学术和生活的热情对我是一种很大的激励！

感谢关开思君和张宇尧君对于证明一些瑕疵提出的及时的指正。你们是非常细心的读者，能够拥有你们这样的读者是我莫大的荣幸，希望我没有辜负你们的期望！

如果正在读这本书的大家感觉到这本书对自己有所帮助，我希望大家除了感谢我以外，再发自内心地感谢一下三明治亲、许洪腾老师、Lucy 君、戴姐姐、关开思君和张宇尧君以及所有向这本书表示过支持的人工智能学院的同学和老师、向这本书提供了参考的学术大牛和网络博主们。没有你们就没有这本书的出现，真心地感谢你们！还有一些疏漏的为本书提供了帮助但是在本书没有提及的人们，在这里我向你们深表歉意，也同时向你们表示真心的感谢！

第一部分

机器学习数学基础

第一章　导论

1.1　L^p 范数 [1]

向量范数的后两个性质保证了范数运算事实上是具有凸性的，因为

$$\|\lambda \boldsymbol{x} + (1-\lambda)\boldsymbol{y}\| \leqslant \|\lambda \boldsymbol{x}\| + \|(1-\lambda)\boldsymbol{y}\| = \lambda \|\boldsymbol{x}\| + (1-\lambda)\|\boldsymbol{y}\|$$

而对于如下定义的函数 $\|\cdot\|_p$

$$\|\boldsymbol{x}\|_p = \left(\sum_{i=1}^{n} |x_i|^p\right)^{1/p}$$

我们把这样定义的函数称为 L^p 范数（L^p norm），尽管有时候这样定义出的范数根本不符合范数的定义。当 $p < 1$ 时对于函数三角不等式是不成立的，因而函数不具有凸性，我们接下来推导的 $p = 0$ 的情况甚至连同质性（Homogeneity）$\|k\boldsymbol{x}\| = k\|\boldsymbol{x}\|$ 都不满足。在课件中所示的范数示意图如果标出 $0 \leqslant p < 1$ 的情况，我们将看到从 $p = 1$ 开始到 $p \to 0^+$ 变化过程中值为 1 对应的等值线最初菱形边界将向内凹陷直至于坐标轴贴合，这与 $p = 1$ 开始到 $p \to +\infty$ 值为 1 对应的等值线的变化情况实际上是相反的，后者最初菱形边界向外突出最终于外部的方形边界贴合[2]。这两个过程像一个固定在方盒里的气球放气和充气的过程。

我们发现 $p = 1$ 是以上运算是否具有凸性的分界点（在两个向量共线时，L^1 对应的三角不等式实际上是能够取等号的，这在 $p > 1$ 的情况不会出现），因而 Lp 范数需要规定 $p > 1$。尽管如此，我们有时为了方便讨论还会将 $0 \leqslant p < 1$ 称为 L^p 范数。我们现在推理一下 $p \to 0^+$ 和 $p \to +\infty$ 两种情况的极限性质：

[1] 课件 lecture1 p.35

[2] https://en.wikipedia.org/wiki/Lp_space

情况一　当 $p \to 0^+$ 时

$$\|\boldsymbol{x}\|_p = \left(\sum_{i=1}^{n} |x_i|^p \right)^{1/p}$$

是可能没有极限的，当存在两个以上非 0 元素时内部求和式极限大于 1，容易求得上式趋于 $+\infty$。当存在两个以下的非 0 元素时，非零元素才存在定义，此时值为 1 对应的等值线退化为边缘的点。

数学分析学的比较好的同学也许会转而考虑以下式子[3]

$$\left(\frac{1}{n} \sum_{i=1}^{n} |x_i|^p \right)^{1/p}$$

考虑指数和对数的一阶近似

$$\exp(x) = 1 + x + o(x)$$

$$\ln(1+x) = x + o(x)$$

得到

$$x^p = \exp(p \ln x) = 1 + p \ln x + o(p)$$

$$\frac{1}{n} \sum_{i=1}^{n} x_i^p = 1 + \frac{p}{n} \sum_{i=1}^{n} \ln x_i + o(p) = 1 + p \ln \sqrt[n]{\prod_{i=1}^{n} x_i} + o(p)$$

记

$$m = \ln \sqrt[n]{\prod_{i=1}^{n} x_i}$$

考虑

$$(1+pm)^{1/p} = \exp(\ln(1+pm)/p) = \exp(m + o(p))$$

得到

$$\lim_{p \to 0^+} \left(\frac{1}{n} \sum_{i=1}^{n} |x_i|^p \right)^{1/p} = \sqrt[n]{\prod_{i=1}^{n} |x_i|}$$

[3] https://math.stackexchange.com/questions/13125/what-is-the-0-norm

即极限为几何平均值。事实上上式也不是我们要得到的最终的定义式。这是由于当存在一个零元素时上式将得到 0，这不太符合我们的要求，因为 L^p 范数趋近于 0 的过程中有个重要的共性为值为 1 对应的等值线上总是存在边界点，即坐标轴上坐标的值为 1 的点，由于 0 的存在，上式几何均值趋于 0。

我们理想中的 L^0 范数值为 1 对应的等值线应该是与坐标轴完全贴合的，而且不仅是在坐标轴上，其余的点上也应该有定义，我们也许可以考虑将 $1/p$ 次幂从定义中去掉，这样定义出来的 L^0 范数就比较符合我们要求了

$$\|\boldsymbol{x}\|_0 := \lim_{p\to 0^+} \sum_{i=1}^{n} |x_i|^p = |\{x_i \mid x_i \neq 0,\ i = 1, 2, \cdots, n\}|$$

情况二　当 $p \to \infty$ 时

$$\max_{i=1,2,\cdots,n} \{|x_i|\} \leqslant \left(\sum_{i=1}^{n} |x_i|^p\right)^{1/p} \leqslant n^{1/p} \max_{i=1,2,\cdots,n} \{|x_i|\}$$

两边取 $p \to \infty$ 后夹逼得到

$$\|\boldsymbol{x}\|_\infty = \max_{i=1,2,\cdots,n} \{|x_i|\}$$

这样定义的 L^0 范数实际上直接反映了向量的稀疏性。与 L^0 相关优化问题的求解通常是 NP 问题，L^0 不光是非凸的，且不是平滑的，解析性质很差。通常对于 $p < 1$ 时非凸的 L^p 范数我们需要利用凸函数对其进行近似（松弛）处理。当 $p < 1$ 时考虑点集 $\|\boldsymbol{x}\|_p \leqslant t$ 时，这些点集的边界在坐标轴上具有公共的不可导点，对于区域 $\|\boldsymbol{x}\|_p \leqslant t$ 不难得知其凸包就是 $\|\boldsymbol{x}\|_1 \leqslant t$（因为可以考虑以坐标轴上的不可导点为极点的凸包），即通过最小的代价将区域扩充为凸集。因而从这个角度 L^1 范数是 L^p 范数的最优近似[4]。

对于 L^0 范数而言情况是类似的，但是我们需要加一些限制，比如考虑介于 -1 到 1 之间的点，考虑 $\|\boldsymbol{x}\|_0 \leqslant 1$，则同样可以证明其凸包就是 $\|\boldsymbol{x}\|_1 \leqslant 1$。还有一种解读是在任意的作为 L^0 范数凸函数 f 中，当限制向量 $\boldsymbol{\beta}$ 的每一维元素介于 -1 到 1 之间时，我们有

$$f(\boldsymbol{\beta}) \leqslant \|\boldsymbol{\beta}\|_0$$

[4] https://en.wikipedia.org/wiki/Lasso_(statistics)

即 L^1 是 L^0 范数最紧的下界[5]。因而在后续讨论 L^0 范数的场合，我们在很多时候都会用 L^1 范数进行近似地代替。

1.2　矩阵范数的简单性质 [6]

矩阵范数（Matrix norm）可以通过向量范数按照如下方式拓展

$$\|\boldsymbol{A}\| := \max_{\boldsymbol{x}\neq 0} \frac{\|\boldsymbol{A}\boldsymbol{x}\|}{\|\boldsymbol{x}\|} = \max_{\boldsymbol{x}\neq 0} \left\|\boldsymbol{A}\frac{\boldsymbol{x}}{\|\boldsymbol{x}\|}\right\| = \max_{\|\boldsymbol{x}\|=1} \|\boldsymbol{A}\boldsymbol{x}\|$$

即 $\boldsymbol{x}$ 的缩放实际上不会影响目标函数取值。对于这样定义的矩阵范数，我们还有

$$\|k\boldsymbol{A}\| = \max_{\|\boldsymbol{x}\|=1} \|k\boldsymbol{A}\boldsymbol{x}\| = |k| \max_{\|\boldsymbol{x}\|=1} \|\boldsymbol{A}\boldsymbol{x}\| = |k|\,\|\boldsymbol{A}\|$$

有时候矩阵范数采用如上形式通过向量范数进行拓展，如谱范数，有时拓展仅仅是简单地对列向量范数求和，如 PCA 使用的 Frobenius 范数（Frobenius norm / F-norm）和 RPCA 中使用的矩阵的逐元素的绝对值求和的范数，注意区分。课件上罗列了几个矩阵范数的一般性质：

性质一　矩阵范数具有同质性，即

$$\|\boldsymbol{A}\|\,\|\boldsymbol{x}\| = \max_{\boldsymbol{x}\neq 0} \frac{\|\boldsymbol{A}\boldsymbol{x}\|}{\|\boldsymbol{x}\|}\|\boldsymbol{x}\| = \max_{\boldsymbol{x}\neq 0} \|\boldsymbol{A}\boldsymbol{x}\| \geqslant \|\boldsymbol{A}\boldsymbol{x}\|$$

性质二　矩阵范数和矩阵乘法之间存在不等关系

$$\begin{aligned}\|\boldsymbol{A}\boldsymbol{B}\| &= \|\boldsymbol{A}(\boldsymbol{b_1}, \boldsymbol{b_2}, \cdots, \boldsymbol{b_n})\| \\ &\leqslant \|(\|\boldsymbol{A}\|\,\boldsymbol{b_1}, \|\boldsymbol{A}\|\,\boldsymbol{b_2}, \cdots, \|\boldsymbol{A}\|\,\boldsymbol{b_n})\| \\ &= \|\boldsymbol{A}\|\,\|(\boldsymbol{b_1}, \boldsymbol{b_2}, \cdots, \boldsymbol{b_n})\| = \|\boldsymbol{A}\|\,\|\boldsymbol{B}\|\end{aligned}$$

性质三　对于对角矩阵，其矩阵范数为对角线上最大元素的绝对值

[5] https://www.zhihu.com/question/40644990/answer/87753458

[6] 课件 lecture1 pp.36-37

$$\begin{aligned}\|\boldsymbol{D}\|_p &= \max_{\|\boldsymbol{x}\|_p=1} \|\boldsymbol{D}\boldsymbol{x}\|_p = \max_{\|\boldsymbol{x}\|_p=1} \left(\sum_{i=1}^{D} |d_{ii}x_i|^p\right)^{1/p} \\ &\leqslant \max_i\{|d_{ii}|\} \max_{\|\boldsymbol{x}\|_p=1} \|\boldsymbol{x}\|_p = \max_i\{|d_{ii}|\}\end{aligned}$$

当 $\boldsymbol{x}$ 在最大的 d_{ii} 对应的行标上的元素取 1（其余取 0）时取到等号，即有

$$\|\boldsymbol{D}\|_p = \max_i\{|d_{ii}|\}$$

对于特殊的矩阵范数，课件上也罗列了几个重要的性质：

L^1 范数　对于任意矩阵的 L^1 范数

$$\begin{aligned}\|\boldsymbol{A}\|_1 &= \max_{\|\boldsymbol{x}\|_1=1} \|\boldsymbol{A}\boldsymbol{x}\|_1 = \max_{\|\boldsymbol{x}\|_1=1} \left\|\sum_{i=1}^{N} x_i\boldsymbol{a_i}\right\|_1 \\ &\leqslant \max_{\|\boldsymbol{x}\|_1=1} \sum_{i=1}^{N} |x_i| \|\boldsymbol{a_i}\|_1 \leqslant \max_i \|\boldsymbol{a_i}\|_1 \max_{\|\boldsymbol{x}\|_1=1} \|\boldsymbol{x}\|_1 = \max_i \|\boldsymbol{a_i}\|_1\end{aligned}$$

当 $\boldsymbol{x}$ 在最大的 $\|\boldsymbol{a_i}\|_1$ 对应的下标上的元素取 1（其余元素取 0）时取到等号，即有

$$\|\boldsymbol{A}\|_1 = \max_i \|\boldsymbol{a_i}\|_1$$

L^2 范数　对于任意矩阵的 L^2 范数有[7]

$$\|\boldsymbol{A}\|_2 = \max_{\boldsymbol{x}^T\boldsymbol{x}=1} \sqrt{\boldsymbol{x}^T\boldsymbol{A}^T\boldsymbol{A}\boldsymbol{x}} = \max_i\{\sigma_i\}$$

特别地，取 $\boldsymbol{A} = \boldsymbol{u}\boldsymbol{v}^T$，该矩阵的奇异值（Sigular value）[8]求解是容易的，只需将 $\boldsymbol{u}$ 和 $\boldsymbol{v}$ 单位化后利用 Gram-Schmidt 正交化（Gram-Schmidt orthogonalization）[9] 扩充为正交矩阵 $\boldsymbol{U}$ 和 $\boldsymbol{V}$ 即可

$$\boldsymbol{A} = \|\boldsymbol{u}\|_2 \|\boldsymbol{v}\|_2 \frac{\boldsymbol{u}}{\|\boldsymbol{u}\|_2} \left(\frac{\boldsymbol{v}}{\|\boldsymbol{v}\|_2}\right)^T = \boldsymbol{U} \begin{pmatrix} \|\boldsymbol{u}\|_2 \|\boldsymbol{v}\|_2 & \\ & \boldsymbol{O} \end{pmatrix} \boldsymbol{V}$$

[7]见 D.2.3 中间和谱范数（Spectral norm）相关的部分

[8]详见 D.2.2 中间部分

[9]详见 D.1.3

得到 $\boldsymbol{A}$ 最大奇异值为 $\|\boldsymbol{u}\|_2 \|\boldsymbol{v}\|_2$，该值为范数的值，即

$$\left\|\boldsymbol{u}\boldsymbol{v}^{\boldsymbol{T}}\right\|_2 = \|\boldsymbol{u}\|_2 \|\boldsymbol{v}\|_2$$

L^∞ 范数　对于任意矩阵的 L^∞ 范数

$$\begin{aligned}\|\boldsymbol{A}\|_\infty &= \max_{\|\boldsymbol{x}\|_\infty=1} \|\boldsymbol{A}\boldsymbol{x}\|_\infty = \max_{\|\boldsymbol{x}\|_\infty=1} \|(\boldsymbol{A_i}\boldsymbol{x})_{D\times 1}\|_\infty \\ &= \max_{\|\boldsymbol{x}\|_\infty=1} \max_i \{|\boldsymbol{A_i}\boldsymbol{x}|\} \leqslant \max_i \left\{\sum_{j=1}^{N} a_{ij}\mathrm{sgn}(a_{ij})\right\} = \max_i \|\boldsymbol{A_i}\|_1\end{aligned}$$

最后一个不等号取得的原因是当 $\boldsymbol{x}$ 在 $\boldsymbol{A_i}$ 对应的元素为正的维度取得 1 为负的维度取得 -1 时我们取到的函数的一个上界。当 $\boldsymbol{x}$ 在最大的 $\|\boldsymbol{A_i}\|_1$ 对应的下标上的元素取 1（其余元素取 0）时取到等号，即有

$$\|\boldsymbol{A}\|_\infty = \max_i \|\boldsymbol{A_i}\|_1$$

第二章　数理基础回顾

2.1　标量、向量和矩阵求导 [1]

矩阵标量、向量和矩阵之间的求导常见的定义如下：

标量对向量求导　对于向量映射到标量的函数

$$f:\mathbb{R}^N\mapsto\mathbb{R}$$

标量对向量的求导定义为

$$\nabla_{\boldsymbol{x}}f(\boldsymbol{x})=\frac{\partial f(\boldsymbol{x})}{\partial\boldsymbol{x}}=\frac{\partial y}{\partial\boldsymbol{x}}=\left(\frac{\partial y}{\partial x_1},\frac{\partial y}{\partial x_2},\cdots,\frac{\partial y}{\partial x_N}\right)^T=\left(\frac{\partial y}{\partial x_i}\right)_N$$

向量对向量求导　对于向量映射到向量的函数

$$f:\mathbb{R}^N\mapsto\mathbb{R}^M$$

向量对向量的求导定义为

$$\nabla_{\boldsymbol{x}}f(\boldsymbol{x})=\frac{\partial f(\boldsymbol{x})}{\partial\boldsymbol{x}}=\frac{\partial\boldsymbol{y}}{\partial\boldsymbol{x}}=\left(\frac{\partial y_1}{\partial\boldsymbol{x}},\frac{\partial y_2}{\partial\boldsymbol{x}},\cdots,\frac{\partial y_M}{\partial\boldsymbol{x}}\right)=\left(\frac{\partial y_j}{\partial x_i}\right)_{N\times M}$$

标量对矩阵求导　对于矩阵映射到标量的函数

$$f:\mathbb{R}^{N\times M}\mapsto\mathbb{R}$$

标量对矩阵求导定义为

[1] 课件 lecture2 p.29

$$\nabla_{\boldsymbol{X}} f(\boldsymbol{X}) = \frac{\partial f(\boldsymbol{X})}{\partial \boldsymbol{X}} = \frac{\partial y}{\partial \boldsymbol{X}} = \left(\frac{\partial y}{\partial \boldsymbol{x_1}}, \frac{\partial y}{\partial \boldsymbol{x_2}}, \cdots, \frac{\partial y}{\partial \boldsymbol{x_M}} \right) = \left(\frac{\partial y}{\partial x_{ij}} \right)_{N \times M}$$

定义为这三种形式的矩阵导数具有高度相关性。下面证明一些常用结论:

结论一

$$\frac{\partial \boldsymbol{x^T a}}{\partial \boldsymbol{x}} = \left(\frac{\partial}{\partial x_i} \sum_{k=1}^{N} a_k x_k \right)_N = (a_i)_N = \boldsymbol{a}$$

结论二

$$\frac{\partial \boldsymbol{a^T X b}}{\partial \boldsymbol{X}} = \left(\frac{\partial}{\partial x_{ij}} \sum_{k=1}^{N} \sum_{l=1}^{M} a_k b_l x_{kl} \right)_{N \times M} = (a_i b_j)_{N \times M} = \boldsymbol{a b^T}$$

特别地

$$\frac{\partial \boldsymbol{a^T X^T b}}{\partial \boldsymbol{X}} = \frac{\partial \boldsymbol{b^T X a}}{\partial \boldsymbol{X}} = \boldsymbol{b a^T}$$

和

$$\frac{\partial \boldsymbol{a^T X a}}{\partial \boldsymbol{X}} = \boldsymbol{a a^T}$$

结论三

$$\begin{aligned} \frac{\partial \boldsymbol{x^T B x}}{\partial \boldsymbol{x}} &= \left(\frac{\partial}{\partial x_i} \sum_{k=1}^{N} \sum_{l=1}^{N} b_{kl} x_k x_l \right)_N = \left(\frac{\partial}{\partial x_i} \left(\sum_{k \neq i} (b_{ki} + b_{ik}) x_k x_i + b_{ii} x_i^2 \right) \right)_N \\ &= \left(\sum_{k=1}^{N} (b_{ki} + b_{ik}) x_k \right)_N = (\boldsymbol{B} + \boldsymbol{B^T}) \boldsymbol{x} \end{aligned}$$

这里的形式与二次型中实对称方阵的构造类似，因而在实的二次型方阵中我们通常只考虑实对称方阵

$$\boldsymbol{x^T B x} = \boldsymbol{x^T B^T x} \Rightarrow \boldsymbol{x^T B x} = \boldsymbol{x^T} \left(\frac{\boldsymbol{B} + \boldsymbol{B^T}}{2} \right) \boldsymbol{x}$$

特别地，对于实对称方阵 $\boldsymbol{A}$ 而言

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

更特别地，取 $\boldsymbol{A}$ 为单位矩阵，得到

$$\frac{\partial \boldsymbol{x}^{\boldsymbol{T}}\boldsymbol{x}}{\partial \boldsymbol{x}} = 2\boldsymbol{x}$$

2.2　常见统计量的无偏估计 [2]

假设对于观测到的 N 个独立同分布的样本 $\boldsymbol{x} = (x_1, x_2, \cdots, x_N)^T$，总体期望为 μ，方差为 σ^2，我们考察期望和方差的无偏估计量（Unbiased estimator）：

期望的无偏估计　根据期望表达式设估计量

$$\mu' := \mathrm{E}[\boldsymbol{x}] = \frac{1}{N}\sum_{n=1}^{N} x_n$$

考虑等式

$$\mathrm{E}[x_n] = \mu$$

得到

$$\mathrm{Bias}(\mu', \mu) = \mathrm{E}[\mu'] - \mathrm{E}[\mu] = \frac{1}{N}\sum_{n=1}^{N}\mathrm{E}[x_n] - \mu = 0$$

因而期望的无偏估计量为

$$\hat{\mu} := \mu' = \frac{1}{N}\sum_{n=1}^{N} x_n = \mathrm{E}[\boldsymbol{x}] \Rightarrow \mathrm{E}[\hat{\mu}] = \mu$$

方差的无偏估计　根据方差表达式设估计量

[2] 课件 lecture2 p.35

$$\sigma'^2 := \mathrm{Var}[\boldsymbol{x}] = \frac{1}{N}\sum_{n=1}^{N}(x_n - \hat{\mu})^2 = \mathrm{E}[\boldsymbol{x}^2] - \mathrm{E}[\boldsymbol{x}]^2$$

考虑等式

$$\mathrm{Var}[x_n] = \sigma^2 = \mathrm{E}[x_n^2] - \mathrm{E}[x_n]^2 \Rightarrow \mathrm{E}[x_n^2] = \sigma^2 + \mu^2$$

$$\begin{aligned}\mathrm{Var}[\hat{\mu}] &= \frac{1}{N^2}\sum_{n=1}^{N}\mathrm{Var}[x_n] = \frac{1}{N}\sigma^2 = \mathrm{E}[\hat{\mu}^2] - \mathrm{E}[\hat{\mu}]^2 \\ &= \mathrm{E}[\hat{\mu}^2] - \mu^2 \Rightarrow \mathrm{E}[\hat{\mu}^2] = \frac{1}{N}\sigma^2 + \mu^2\end{aligned}$$

得到

$$\begin{aligned}\mathrm{Bias}(\sigma'^2, \sigma^2) &= \mathrm{E}[\sigma'^2] - \mathrm{E}[\sigma^2] = \mathrm{E}[\mathrm{E}[\boldsymbol{x}^2] - \mathrm{E}[\boldsymbol{x}]^2] - \sigma^2 \\ &= \mathrm{E}[\mathrm{E}[\boldsymbol{x}^{\boldsymbol{2}}]] - \mathrm{E}[\hat{\mu}^2] - \sigma^2 = \frac{1}{N}\sum_{n=1}^{N}\mathrm{E}[x_n^2] - \mathrm{E}[\hat{\mu}^2] - \sigma^2 \\ &= -\frac{1}{N}\sigma^2 = -\frac{1}{N}\mathrm{E}[\sigma^2] \Rightarrow \mathrm{E}[\sigma'^2] = \frac{N-1}{N}\mathrm{E}[\sigma^2]\end{aligned}$$

通过 σ' 构造无偏估计量 $\hat{\sigma} := k\sigma'$，使得

$$\mathrm{E}[k\sigma'] = k\mathrm{E}[\sigma'] = k\frac{N-1}{N}\mathrm{E}[\sigma^2] = \mathrm{E}[\sigma^2]$$

得到

$$k = \frac{N}{N-1}$$

和

$$\hat{\sigma} = k\sigma' = \frac{1}{N-1}\sum_{n=1}^{N}(x_n - \hat{\mu})^2$$

考虑二元的统计量，对于来自期望为 μ_x，方差为 σ_x 的总体的 N 个独立同分布的样本 $\boldsymbol{x} = (x_1, x_2, \cdots, x_N)^T$，来自另一个期望为 μ_y，方差为 σ_y 的总体的 N 个独立同分布的样本 $\boldsymbol{y} = (y_1, y_2, \cdots, y_N)^T$，在假设两个总体在同一时刻观测（即下标相同的）的样本是相关的而不同时刻（即下标不同的）观测到的样本是不相关的，设 $\boldsymbol{x}$ 和 $\boldsymbol{y}$ 总体的协方差为 σ_{xy}，我们考察协方差的无偏估计量：

协方差的无偏估计　根据协方差表达式设估计量

$$\sigma'_{xy} := \mathrm{Cov}[\boldsymbol{x}, \boldsymbol{y}] = \frac{1}{N}\sum_{n=1}^{N}(x_n - \hat{\mu}_x)(y_n - \hat{\mu}_y) = \mathrm{E}[\boldsymbol{xy}] - \mathrm{E}[\boldsymbol{x}]\mathrm{E}[\boldsymbol{y}]$$

类似地考虑等式

$$\mathrm{Cov}[x_n, y_n] = \sigma_{xy} = \mathrm{E}[x_n y_n] - \mathrm{E}[x_n]\mathrm{E}[y_n] \Rightarrow \mathrm{E}[x_n y_n] = \sigma_{xy} + \mu_x \mu_y$$

$$\begin{aligned}\mathrm{Cov}[\hat{\mu}_x, \hat{\mu}_y] &= \frac{1}{N^2}\sum_{n=1}^{N}\mathrm{Cov}[x_n, y_n] = \frac{1}{N}\sigma_{xy} = \mathrm{E}[\hat{\mu}_x\hat{\mu}_y] - \mathrm{E}[\hat{\mu}_x]\mathrm{E}[\hat{\mu}_y] \\ &= \mathrm{E}[\hat{\mu}_x\hat{\mu}_y] - \mu_x\mu_y \Rightarrow \mathrm{E}[\hat{\mu}_x\hat{\mu}_y] = \frac{1}{N}\sigma_{xy} + \mu_x\mu_y\end{aligned}$$

类似地得到

$$\begin{aligned}\mathrm{Bias}(\sigma'_{xy}, \sigma_{xy}) &= \mathrm{E}[\sigma'_{xy}] - \mathrm{E}[\sigma_{xy}] = \mathrm{E}[\mathrm{E}[\boldsymbol{xy}] - \mathrm{E}[\boldsymbol{x}]\mathrm{E}[\boldsymbol{y}]] - \sigma_{xy} \\ &= \mathrm{E}[\mathrm{E}[\boldsymbol{xy}]] - \mathrm{E}[\hat{\mu}_x\hat{\mu}_y] - \sigma_{xy} = \frac{1}{N}\sum_{n=1}^{N}\mathrm{E}[x_n y_n] - \mathrm{E}[\hat{\mu}_x\hat{\mu}_y] - \sigma_{xy} \\ &= -\frac{1}{N}\sigma_{xy} = -\frac{1}{N}\mathrm{E}[\sigma_{xy}] \Rightarrow \mathrm{E}[\sigma'_{xy}] = \frac{N-1}{N}\mathrm{E}[\sigma_{xy}]\end{aligned}$$

得到无偏估计量为

$$\hat{\sigma}_{xy} = \frac{1}{N-1}\sum_{n=1}^{N}(x_n - \hat{\mu}_x)(y_n - \hat{\mu}_y)$$

2.3　频率学派与 Bayesian 学派 [3]

频率学派认为参数是个常数，数据是从参数确定的分布中采集而来的，基于这种观点的极大似然估计（Maximum likelihood estimation / MLE）希望能够最大化在给定模型参数的基础上观测到数据的概率。Bayesian 学派认为参数是未知的，其可被视为一个随机变量，基于这种观点的最大后验概率（Maximum a posteriori probability / MAP）对参数的估计事实上是在已知信息的基础上最大化时其对应的后验概率。频率学派倾向于通过对参数的

[3] 课件 lecture2 pp.41-42

点估计得到参数，而 Bayesian 学派倾向于将参数从参数分布中进行采样得到参数。由于 Bayesian 学派估计了参数的分布，因而可以对模型训练的稳定性进行衡量。

注意 MLE 和 MAP 的本质区别在于是否对模型参数 $\boldsymbol{\theta}$ 有一个先验的知识。即 MAP 在计算目标函数时，会乘一项 $p(\boldsymbol{\theta})$。在对目标函数负对数后，MAP 相较于 MLE 多减了一项 $\log p(\boldsymbol{\theta})$，这是我们接下来设计正则项（Regularization）的依据。设计正则项的目的是为了缩小最后求得的目标参数距离先验知识之间的差距，这使得先验知识对参数的求解起指导和约束作用，在输入的数据量较少、迭代轮数不足的情况下这种指导对最后结果的影响可能是关键性的。

第二部分

回归分析

第三章　多项式回归

3.1　多项式回归有解判定 [1]

首先考虑 Vandermonde 矩阵的性质，对于不含零的 $\boldsymbol{x}$，Vandermonde 矩阵（Vandermonde matrix）定义如下

$$\boldsymbol{V_D}(\boldsymbol{x}) = \boldsymbol{V_D}(x_1, x_2, \cdots, x_N) = \left(x_j^{i-1}\right)_{D\times N},\ D \leqslant N$$

Vandermonde 矩阵是行满秩的当且仅当 $\boldsymbol{x}$ 的每个元素均不相同。考虑以矩阵的最大非零子式定义的矩阵的秩和 Vandermonde 方阵的行列式即可

$$\det(\boldsymbol{V}(\boldsymbol{x})) = \prod_{i\neq j}(x_i - x_j)$$

该式确保了在 $\boldsymbol{x}$ 的每个元素均不相同且不为零的情况下多项式回归是存在闭式解的。考虑对多项式回归的损失函数求导得到

$$\begin{aligned}\frac{\partial L}{\partial \boldsymbol{w}} &= \left(\frac{\partial}{\partial w_i}\sum_{n=1}^{N}(y_n - \boldsymbol{x_n^T w})^2\right)_{D\times 1} = 2\left(\sum_{n=1}^{N}(y_n - \boldsymbol{x_n^T w})\frac{\partial}{\partial w_i}\left(y_n - \sum_{d=1}^{D} w_d x_{nd}\right)\right)_{D\times 1}\\ &= 2\left(\sum_{n=1}^{N}(\boldsymbol{x_n^T w} - y_n)x_{id}\right)_{D\times 1} = 2\left(\boldsymbol{X_i}(\boldsymbol{X^T w} - \boldsymbol{y})\right)_{D\times 1} = 2\boldsymbol{X}(\boldsymbol{X^T w} - \boldsymbol{y})\end{aligned}$$

令导数为 0，在 $\boldsymbol{XX^T}$ 的逆存在时得到闭式解

$$\boldsymbol{w^*} = (\boldsymbol{XX^T})^{-1}\boldsymbol{Xy}$$

考虑矩阵伪逆后闭式解为

$$\boldsymbol{w^*} = (\boldsymbol{X^T})^{+}\boldsymbol{y}$$

[1] 课件 lecture3 pp.3,7,9

从而 $\boldsymbol{X}\boldsymbol{X}^{\boldsymbol{T}}$ 的逆存在当且仅当 $\boldsymbol{X}^{\boldsymbol{T}}$ 是列满秩的[2]，即 $\boldsymbol{X}$ 是行满秩的。我们得到了只要 $\boldsymbol{x}$ 的每个元素均不相同且不为零时多项式回归存在唯一的闭式解。

3.2　SGD 优化算法 [3]

机器学习中的回归问题可以表达为如下等式

$$\boldsymbol{w}^* := \arg\min_{\boldsymbol{w}} \mathrm{E}_{\boldsymbol{x},y\sim\mathcal{P}}[L(y, f_{\boldsymbol{w}}(\boldsymbol{x}))]$$

考虑正则项 [4]后式子化为

$$\boldsymbol{w}^* := \arg\min_{\boldsymbol{w}} \mathrm{E}_{\boldsymbol{x},y\sim\mathcal{P}}[L(y, f_{\boldsymbol{w}}(\boldsymbol{x}))] + \mathcal{R}(\boldsymbol{w})$$

此处我们暂时不考虑正则项。在一轮的学习中随机梯度下降法（Stochastic gradient descent / SGD）对参数的优化考虑了如下近似式

$$\begin{aligned}\boldsymbol{w}^{(t+1)} &:= \boldsymbol{w}^{(t)} - \tau\nabla_{\boldsymbol{w}}\frac{1}{|B|}\sum_{(\boldsymbol{x},y)\in B} L(y, f_{\boldsymbol{w}}(\boldsymbol{x})) \\ &\approx \boldsymbol{w}^{(t)} - \tau\nabla_{\boldsymbol{w}}\mathrm{E}_{\boldsymbol{x},y\sim\mathcal{P}}[L(y, f_{\boldsymbol{w}}(\boldsymbol{x}))]\end{aligned}$$

即我们近似地认为一个随机的 batch 的样本其分布和总体是相近的，其梯度的期望也应与总体相近。因而在凸优化（Convex optimization）的情况下我们使用 SGD 进行梯度下降时总能够使得真实的损失下降从而达到我们的最小化损失的目的。

这样的一个过程事实上在寻找给定步长的情况下近似的局部最值点，因为在局部梯度方向事实上是函数值变化最快的方向。因为这种参数空间上的局部性质和全局的性质的差异，SGD 不一定是收敛最快的算法。在 SGD 优化算法的基础上后续还有很多的改进，感兴趣的同学可以查阅相关资料进行了解。

---

[2]详见D.1.4 中间部分

[3]课件 lecture3 p.9

[4]详见 4.2 结尾部分

3.3　正态分布与 Laplace 分布简介 [5]

对于服从正态分布（Normal distribution）$\mathcal{N}(\mu, \sigma^2)$ 的 1 维随机变量 X，其密度函数为

$$f_X(x) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{1}{2\sigma^2}(x-\mu)^2\right)$$

其期望满足

$$\mathrm{E}[X] = \mu$$

方差满足

$$\mathrm{Var}[X] = \sigma^2$$

对于服从 Laplace 分布（Laplace distribution）Laplace(μ, b) 的 1 维随机变量 X，其概率密度函数为

$$f_X(x) = \frac{1}{2b} \exp(-\frac{|x-\mu|}{b})$$

其相当于两个指数函数沿着 $x = \mu$ 拼接在一起，在 $x = \mu$ 时函数是不可导的。期望满足

$$\mathrm{E}[X] = \mu$$

方差满足

$$\mathrm{Var}[X] = 2b^2$$

Laplace 分布和正态分布形状上比较相似，但是对于标准 Laplace 分布（参数为 $\mu = 0$ 和 $b = 1$）和标准正态分布，正态分布的尾部（Tail）较细长，即出现极端值的概率更低。

Laplace 分布和指数分布以及均匀分布等概率论常见的分布均有联系 [6]。在机器学习中，选择误差为平均平方误差（Mean squared error / MSE）时，模型对应的噪声服从正态分布，选择误差为平均绝对误差（Mean absolute error / MAE）时，模型对应的噪声服从 Laplace 分布。

[5] 课件 lecture3 p.15

[6] https://en.wikipedia.org/wiki/Laplace_distribution

第四章　线性回归与正则

4.1　GLM [1]

对于一般的线性回归，标签和数据之间存在关系

$$y = \boldsymbol{x}^{\boldsymbol{T}}\boldsymbol{w} + \varepsilon$$

其中

$$\mathrm{E}[\varepsilon] = 0,\ \mathrm{Var}[\varepsilon] = \sigma^2$$

我们可以进一步地取

$$\varepsilon \sim \mathcal{N}(0, \sigma^2)$$

也可以考虑所有数据，写为

$$\boldsymbol{y} = \boldsymbol{X}^{\boldsymbol{T}}\boldsymbol{w} + \boldsymbol{\varepsilon}$$

其中

$$\boldsymbol{\varepsilon} \sim \mathcal{N}(\boldsymbol{0_N}, \sigma^2\boldsymbol{I})$$

我们也可以将模型理解为

$$\boldsymbol{y} \sim \mathcal{N}(\boldsymbol{\mu}, \sigma^2\boldsymbol{I}),\ \mathrm{E}_{\boldsymbol{y}\sim\mathcal{N}(\boldsymbol{\mu},\sigma^2\boldsymbol{I})}[\boldsymbol{y}] = \boldsymbol{\mu},\ \boldsymbol{\mu} = \boldsymbol{\eta} = \boldsymbol{X}^{\boldsymbol{T}}\boldsymbol{w}$$

假如 $\boldsymbol{y}$ 不再服从正态分布而是其他分布时，我们该如何建立 $\boldsymbol{y}$ 与线性部分 $\boldsymbol{\eta}$ 之间的关联呢？模仿线性回归，我们可以考虑通过 $\boldsymbol{y}$ 的分布的期望 $\boldsymbol{\mu}$ 建

[1] 课件 lecture4 pp.6-13

立和 $\boldsymbol{\eta}$ 的联系，从而得到广义线性模型（Generalized Linear Model / GLM）[2]

$$\boldsymbol{y} \sim f(\boldsymbol{\theta}, \tau),\ \mathrm{E}_{\boldsymbol{y}\sim f(\boldsymbol{\theta},\tau)}[\boldsymbol{y}] = \boldsymbol{\mu},\ g(\boldsymbol{\mu}) = \boldsymbol{\eta} = \boldsymbol{X}^{\boldsymbol{T}}\boldsymbol{w}$$

以上三个部分为 GLM 的三个组成部分，我们逐一进行介绍：

随机部分　特定的指数分布族（Exponential families）中的分布函数，用于对 $\boldsymbol{y}$ 的概率分布进行建模

$$p(\boldsymbol{y};\boldsymbol{\theta}) = h(\boldsymbol{y}) \exp\left(\boldsymbol{\theta}^{\boldsymbol{T}}\boldsymbol{T}(\boldsymbol{y}) - A(\boldsymbol{\theta})\right)$$

其中 $\boldsymbol{T}(\boldsymbol{y})$ 为 $\boldsymbol{\theta}$ 的充分统计量[3]（Sufficient statistic），其定义为在给定 $\boldsymbol{T}(\boldsymbol{y})$ 为一个特定的值时 $\boldsymbol{y}$ 的分布与 $\boldsymbol{\theta}$，从信息论的角度看，$\boldsymbol{T}(\boldsymbol{y})$ 和 $\boldsymbol{y}$ 关于模型参数 $\boldsymbol{\theta}$ 的互信息 [4] 满足

$$I(\boldsymbol{\theta};\boldsymbol{T}(\boldsymbol{y})) = I(\boldsymbol{\theta};\boldsymbol{y})$$

模型参数 $\boldsymbol{\theta}$ 被称为自然参数（Natural parameter）。更一般地考虑散度参数（Dispersion parameter），散度参数是方差的推广，且与 $\boldsymbol{X}$ 无关

$$p(\boldsymbol{y};\boldsymbol{\theta},\tau) = h(\boldsymbol{y},\tau) \exp\left(\frac{\boldsymbol{\theta}^{\boldsymbol{T}}\boldsymbol{T}(\boldsymbol{y}) - A(\boldsymbol{\theta})}{d(\tau)}\right)$$

这部分建立起观测值 $\boldsymbol{y}$ 和参数 $\boldsymbol{\theta}$ 以及 $\boldsymbol{\mu}$ 的联系，称为模型的随机部分（Random component）。

系统部分　一个线性预测器（Linear predictor）$\boldsymbol{\eta} = \boldsymbol{X}^{\boldsymbol{T}}\boldsymbol{\beta}$，为 GLM 的系统部分（Systematic component），是 GLM 中线性的来源。

连接函数　一个连接函数（Link function）$\boldsymbol{g}$，提供分布的期望 $\boldsymbol{\mu}$ 和 $\boldsymbol{\eta}$ 之间的联系 $\boldsymbol{\mu} = \boldsymbol{g}^{-\mathbf{1}}(\boldsymbol{\eta})$ 或 $\boldsymbol{\eta} = \boldsymbol{g}(\boldsymbol{\mu})$。特别地，当选取 $\boldsymbol{g}(\boldsymbol{\mu}) = \boldsymbol{\mu}$ 时得到的连接函数被称为自连接函数（Identity link function），当选取 $\boldsymbol{g}(\boldsymbol{\mu}) = \boldsymbol{\theta} = \boldsymbol{\eta}$ 时得到的连接函数为自然连接函数（Canonical link function）[2]。由于 $\boldsymbol{\mu}$ 是和 $\boldsymbol{\theta}$ 有关的函数，因而一种比较便于理解的计法是 $\boldsymbol{g}(\boldsymbol{\mu}(\boldsymbol{\theta})) = \boldsymbol{\theta} = \boldsymbol{\eta}$。

[2] https://zhuanlan.zhihu.com/p/420499972

[3] https://en.wikipedia.org/wiki/Sufficient_statistic

[4] 详见 F.1.3 中间部分

函数 $A(\boldsymbol{\theta})$ 是一个非常重要的函数，该函数事实上是一个归一化因子（Normalization factor），由概率归一化得到

$$\int_{\mathcal{Y}} p(\boldsymbol{y};\boldsymbol{\theta},\tau)\mathrm{d}\boldsymbol{y} = \int_{\mathcal{Y}} h(\boldsymbol{y},\tau)\exp\left(\frac{\boldsymbol{\theta}^T\boldsymbol{T}(\boldsymbol{y}) - A(\boldsymbol{\theta})}{d(\tau)}\right) = 1$$

从而得到 $A(\boldsymbol{\theta})$ 和分布的关系

$$A(\boldsymbol{\theta}) = d(\tau)\log\int_{\mathcal{Y}} h(\boldsymbol{y},\tau)\exp\left(\frac{\boldsymbol{\theta}^{\boldsymbol{T}}\boldsymbol{T}(\boldsymbol{y})}{d(\tau)}\right)\mathrm{d}\boldsymbol{y}$$

函数 $A(\boldsymbol{\theta})$ 的一阶导和二阶导和分布的矩（Moment）具有很强的关联性[5]，考虑对其求一阶导得到

$$\begin{aligned}\frac{\mathrm{d}A}{\mathrm{d}\boldsymbol{\theta}} &= \frac{d(\tau)}{\exp(A(\boldsymbol{\theta}))}\int_{\mathcal{Y}} h(\boldsymbol{y},\tau)\frac{\mathrm{d}}{\mathrm{d}\boldsymbol{\theta}}\exp\left(\frac{\boldsymbol{\theta}^{\boldsymbol{T}}\boldsymbol{T}(\boldsymbol{y})}{d(\tau)}\right)\mathrm{d}\boldsymbol{y}\\ &= \int_{\mathcal{Y}} \boldsymbol{T}(\boldsymbol{y})h(\boldsymbol{y},\tau)\exp\left(\frac{\boldsymbol{\theta}^{\boldsymbol{T}}\boldsymbol{T}(\boldsymbol{y}) - A(\boldsymbol{\theta})}{d(\tau)}\right)\mathrm{d}\boldsymbol{y}\\ &= \int_{\mathcal{Y}} \boldsymbol{T}(\boldsymbol{y})p(\boldsymbol{y};\boldsymbol{\theta},\tau)\mathrm{d}\boldsymbol{y} = \mathrm{E}_{\boldsymbol{y};\boldsymbol{\theta},\tau}[\boldsymbol{T}(\boldsymbol{y})]\end{aligned}$$

考虑对其求二阶导得到，从中我们可以发掘散度参数和协方差的联系

$$\begin{aligned}\frac{\mathrm{d}^2A}{\mathrm{d}\boldsymbol{\theta}^2} &= \frac{\mathrm{d}}{\mathrm{d}\boldsymbol{\theta}}\int_{\mathcal{Y}} \boldsymbol{T}(\boldsymbol{y})h(\boldsymbol{y},\tau)\exp\left(\frac{\boldsymbol{\theta}^{\boldsymbol{T}}\boldsymbol{T}(\boldsymbol{y}) - \boldsymbol{A}(\boldsymbol{\theta})}{d(\tau)}\right)\mathrm{d}\boldsymbol{y}\\ &= \frac{\mathrm{d}}{\mathrm{d}\boldsymbol{\theta}}\int_{\mathcal{Y}} \frac{\boldsymbol{T}(\boldsymbol{y})\boldsymbol{T}(\boldsymbol{y})^T}{d(\tau)}h(\boldsymbol{y},\tau)\exp\left(\boldsymbol{\theta}^{\boldsymbol{T}}\boldsymbol{T}(\boldsymbol{y}) - \boldsymbol{A}(\boldsymbol{\theta})\right)\mathrm{d}\boldsymbol{y} - \frac{1}{d(\tau)}\frac{\mathrm{d}A}{\mathrm{d}\boldsymbol{\theta}}\mathrm{E}_{\boldsymbol{y};\boldsymbol{\theta},\tau}[\boldsymbol{T}(\boldsymbol{y})]\\ &= \frac{\mathrm{E}_{\boldsymbol{y};\boldsymbol{\theta},\tau}[\boldsymbol{T}(\boldsymbol{y})\boldsymbol{T}(\boldsymbol{y})^T] - \mathrm{E}_{\boldsymbol{y};\boldsymbol{\theta},\tau}[\boldsymbol{T}(\boldsymbol{y})]\mathrm{E}_{\boldsymbol{y};\boldsymbol{\theta},\tau}[\boldsymbol{T}(\boldsymbol{y})]^T}{d(\tau)} = \frac{\mathrm{Cov}_{\boldsymbol{y};\boldsymbol{\theta},\tau}[\boldsymbol{T}(\boldsymbol{y})]}{d(\tau)}\end{aligned}$$

我们考虑充分统计量 $\boldsymbol{T}(\boldsymbol{y}) = \boldsymbol{y}$ 的情况，这用来应对机器学习这门课程已经足够了

$$p(\boldsymbol{y};\boldsymbol{\theta},\tau) = h(\boldsymbol{y},\tau)\exp\left(\frac{\boldsymbol{\theta}^{\boldsymbol{T}}\boldsymbol{y} - A(\boldsymbol{\theta})}{d(\tau)}\right)$$

从而得到

$$\nabla_{\boldsymbol{\theta}} A(\boldsymbol{\theta}) = \mathrm{E}_{\boldsymbol{y};\boldsymbol{\theta},\tau}[\boldsymbol{y}] = \boldsymbol{\mu}$$

[5] https://zhuanlan.zhihu.com/p/282794218

选取连接函数为自然连接函数

$$\boldsymbol{\mu} = \boldsymbol{g^{-1}}(\boldsymbol{\theta}) = \nabla_{\boldsymbol{\theta}} A(\boldsymbol{\theta}) \Rightarrow \boldsymbol{g} = (\nabla_{\boldsymbol{\theta}} A)^{-1}$$

这表明选取连接函数为自然连接函数时连接函数可以由分布直接确定。在模型确定以后，可以利用连接函数通过 MLE 或 MAP 进行参数 β 的迭代更新。

重新回顾线性回归

$$y \sim \mathcal{N}(\mu', \sigma^2)$$

得到

$$p(y;\theta,\tau) = \frac{1}{\sqrt{2\pi}\sigma^2}\exp\left(-\frac{(y-\mu')^2}{2\sigma^2}\right) = \frac{1}{\sqrt{2\pi}\sigma^2}\exp(-\frac{y^2}{2\sigma^2})\exp\left(\frac{\mu' y - \frac{1}{2}\mu'^2}{\sigma^2}\right)$$

从而我们确定了 GLM 的组成部分

$$\theta = \mu',\ d(\tau) = \sigma^2$$
$$A(\theta) = \frac{1}{2}\mu'^2$$
$$d(\tau) = \sigma^2$$
$$h(y,\tau) = \frac{1}{\sqrt{2\pi}\sigma^2}\exp(-\frac{y^2}{2\sigma^2})$$

考虑标准连接函数，以下实际上给出的是一个自连接函数

$$\theta = \eta = \boldsymbol{x^T w} = \mu'$$

从而得到

$$p(y \mid \boldsymbol{x};\boldsymbol{w}) = \frac{1}{\sqrt{2\pi}\sigma^2}\exp\left(-\frac{(y-\boldsymbol{x^T w})^2}{2\sigma^2}\right)$$

这就回到了一般的线性回归。

4.2　偏差-方差分析与正则 [6]

对于给定样本 $\boldsymbol{x}$，我们以如下方式生成标签 y，假定噪声 ε 和 $\boldsymbol{x}^{\boldsymbol{T}}\boldsymbol{w}$ 是相互独立的

$$y = \boldsymbol{x}^{\boldsymbol{T}}\boldsymbol{w} + \varepsilon,\ \varepsilon \in \mathcal{N}(0, \sigma^2)$$

设估计参数为 $\hat{\boldsymbol{w}}$，估计参数确定的线性模型为

$$\hat{y} = \boldsymbol{x}^{\boldsymbol{T}}\hat{\boldsymbol{w}}$$

取

$$f := \boldsymbol{x}^{\boldsymbol{T}}\boldsymbol{w},\ \hat{f} := \boldsymbol{x}^{\boldsymbol{T}}\hat{\boldsymbol{w}}$$

对于估计参数为 $\hat{\boldsymbol{w}}$ 在已知生成数据使用的参数 $\boldsymbol{x}$ 和 $\boldsymbol{w}$ 时估计其 MSE [7]

$$\mathrm{MSE}(y, \hat{y}) = \mathrm{E}[(y-\hat{y})^2] = \mathrm{E}[(f+\varepsilon-\hat{f})^2] = \mathrm{E}[\varepsilon^2] + 2\mathrm{E}[\varepsilon(f-\hat{f})] + \mathrm{E}[(f-\hat{f})^2]$$

考虑

$$\mathrm{E}[\varepsilon^2] = \mathrm{Var}[\varepsilon] - \mathrm{E}[\varepsilon]^2 = \sigma^2$$

和噪声的独立性

$$\mathrm{E}[\varepsilon(f-\hat{f})] = \mathrm{E}[\varepsilon]\mathrm{E}[f-\hat{f}] = 0$$

得到

$$\mathrm{MSE}(y, \hat{y}) = \sigma^2 + \mathrm{E}[(f-\hat{f})^2]$$

考虑配凑方差，这是在数学分析中极限性质证明里常用的技巧

$$\begin{aligned}\mathrm{MSE}(y, \hat{y}) &= \sigma^2 + \mathrm{E}[(f - \mathrm{E}[\hat{f}] + \mathrm{E}[\hat{f}] - \hat{f})^2] \\ &= \sigma^2 + \mathrm{E}[(f - \mathrm{E}[\hat{f}])^2] + 2\mathrm{E}[(f - \mathrm{E}[\hat{f}])(\mathrm{E}[\hat{f}] - \hat{f})] + \mathrm{E}[(\mathrm{E}[\hat{f}] - \hat{f})^2]\end{aligned}$$

[6] 课件 lecture4 pp.18-21,23

[7] 详见 3.3

生成数据的 f 是已知的，而 $\mathrm{E}[\hat{f}]$ 同样是一个常数，因而在期望求解可以提出对应的常数

$$\begin{aligned}\mathrm{MSE}(y,\hat{y}) &= \sigma^2 + (f-\mathrm{E}[\hat{f}])^2 + 2(f-\mathrm{E}[\hat{f}])\mathrm{E}[\mathrm{E}[\hat{f}]-\hat{f}] + \mathrm{E}[(\mathrm{E}[\hat{f}]-\hat{f})^2] \\ &= \sigma^2 + (\mathrm{E}[f]-\mathrm{E}[\hat{f}])^2 + 2(f-\mathrm{E}[\hat{f}])(\mathrm{E}[\hat{f}]-\mathrm{E}[\hat{f}]) + \mathrm{E}[(\mathrm{E}[\hat{f}]-\hat{f})^2] \\ &= \sigma^2 + \mathrm{Bias}(f,\hat{f}) + \mathrm{Var}[\hat{f}]\end{aligned}$$

该 MSE 的分解（Bias-variance decomposition）刻画了线性回归模型影响均方误差的两个因素，偏差和方差。通常来说偏差和方差的优化不可兼得[8]。由大数定律，当训练数据增加时，新的样本的引入将减少模型方差，但与之相对的将带来新的预测偏差，造成模型的欠拟合（Underfitting）；而当模型参数量上升时，模型的预测能力上升，偏差将由所下降，但与之相对的模型的不确定性提高，方差增大，造成模型的过拟合（Overfitting）。以引入额外的偏差为代价降低方差，提高模型稳定性，这样的思想引出了我们接下来要谈的模型正则项（Regularization）的设计。

在 Bayesian 统计中正则项对应着某种先验，先验信息往往以类似新的样本的形式添加至训练过程，样本的权重由正则项的权重控制，约束着训练过程，因而先验在样本量较少时起着较为关键的作用。在前面我们提到了样本量的增加以新的预测偏差为代价换取训练的稳定性，我们也可以这样理解，人为引入的先验知识不是从数据本身出发得到的，这种依靠猜测得到的结论往往会引入新的偏差，然而在我们先验知识的约束下，模型优化的方向更加确定，其预测结果的确定性也提高了，因而方差下降了。

在已知数据集 $\boldsymbol{X}$ 的情况下，考虑 MAP 得到

$$\max_{\boldsymbol{w}} p(\boldsymbol{w} \mid \boldsymbol{y}, \boldsymbol{X}) \Rightarrow \max_{\boldsymbol{w}} p(\boldsymbol{w}, \boldsymbol{y}, \boldsymbol{X}) \Rightarrow \max_{\boldsymbol{w}} p(\boldsymbol{y} \mid \boldsymbol{w}, \boldsymbol{X}) p(\boldsymbol{w} \mid \boldsymbol{X})$$

由于先验知识与数据无关，得到

$$p(\boldsymbol{w} \mid \boldsymbol{X}) = p(\boldsymbol{w})$$

从而

$$\max_{\boldsymbol{w}} p(\boldsymbol{y} \mid \boldsymbol{w}, \boldsymbol{X}) p(\boldsymbol{w}) \Rightarrow \min_{\boldsymbol{w}} -\log p(\boldsymbol{y} \mid \boldsymbol{X}; \boldsymbol{w}) - \log p(\boldsymbol{w})$$

[8] https://en.wikipedia.org/wiki/Bias%E2%80%93variance_tradeoff

4.3　岭回归及拓展 [9]

4.3.1　L^2 正则与岭回归 [10]

岭回归模型（Ridge regression）沿用了传统线性模型对标签分布的假设

$$y = \boldsymbol{x}^{\boldsymbol{T}}\boldsymbol{w} + \varepsilon,\ \varepsilon \in \mathcal{N}(0, \sigma^2)$$

或者表述为

$$\boldsymbol{y} \sim \mathcal{N}(\boldsymbol{X}^{\boldsymbol{T}}\boldsymbol{w}, \sigma^2\boldsymbol{I_N})$$
$$p(\boldsymbol{y} \mid \boldsymbol{X}; \boldsymbol{w}) = \det(2\pi\sigma^2\boldsymbol{I_N})^{-1/2} \exp\left(-\frac{1}{2\sigma^2}\left\|\boldsymbol{y} - \boldsymbol{X}^{\boldsymbol{T}}\boldsymbol{w}\right\|_2^2\right)$$

岭回归的参数 $\boldsymbol{w}$ 的先验满足

$$\boldsymbol{w} \sim \mathcal{N}(\boldsymbol{0_D}, \gamma^2\boldsymbol{I_D})$$
$$p(\boldsymbol{w}) = (2\pi\gamma^2)^{-D/2} \exp\left(-\frac{1}{2\gamma^2}\left\|\boldsymbol{w}\right\|_2^2\right)$$

选取对数为自然对数，代入 MAP 中得到岭回归的目标函数表达式

$$\min_{\boldsymbol{w}} \left\|\boldsymbol{y} - \boldsymbol{X}^{\boldsymbol{T}}\boldsymbol{w}\right\|_2^2 + \frac{\sigma^2}{\gamma^2}\left\|\boldsymbol{w}\right\|_2^2$$

对目标函数

$$\min_{\boldsymbol{w}} \left\|\boldsymbol{y} - \boldsymbol{X}^{\boldsymbol{T}}\boldsymbol{w}\right\|_2^2 + \lambda\left\|\boldsymbol{w}\right\|_2^2$$

求导得到

$$\frac{\partial L}{\partial \boldsymbol{w}} = 2\boldsymbol{X}(\boldsymbol{X}^{\boldsymbol{T}}\boldsymbol{w} - \boldsymbol{y}) + 2\lambda\boldsymbol{w}$$

令导数为 0 得到

$$\boldsymbol{w}^* = (\boldsymbol{X}\boldsymbol{X}^{\boldsymbol{T}} + \lambda\boldsymbol{I})^{-1}\boldsymbol{X}\boldsymbol{y}$$

岭回归大大增强了求逆运算的稳定性，这是由于取 $\lambda > 0$ 对半正定的实对称矩阵 $\boldsymbol{X}\boldsymbol{X}^{\boldsymbol{T}}$ 进行特征值分解得到

[9] 课件 lecture4 pp.22-24

[10] 课件 lecture4 pp.22-23

$$\boldsymbol{X}\boldsymbol{X}^{\boldsymbol{T}} = \boldsymbol{V}\boldsymbol{\Lambda}\boldsymbol{V}^{\boldsymbol{T}},\ \boldsymbol{V}\boldsymbol{V}^{\boldsymbol{T}} = \boldsymbol{I}$$

和

$$\boldsymbol{V}\boldsymbol{\Lambda}\boldsymbol{V}^{\boldsymbol{T}} + \lambda\boldsymbol{I} = \boldsymbol{V}\boldsymbol{\Lambda}\boldsymbol{V}^{\boldsymbol{T}} + \lambda\boldsymbol{V}\boldsymbol{V}^{\boldsymbol{T}} = \boldsymbol{V}(\boldsymbol{\Lambda} + \lambda\boldsymbol{I})\boldsymbol{V}^{\boldsymbol{T}}$$

得到

$$\operatorname{rank}(\boldsymbol{V}\boldsymbol{\Lambda}\boldsymbol{V}^{\boldsymbol{T}} + \lambda\boldsymbol{I}) = \operatorname{rank}(\boldsymbol{\Lambda} + \lambda\boldsymbol{I})$$

对于半正定矩阵而言，其特征值均非负，因而正则项的添加将使得所有特征值为正，从而保证了矩阵的逆存在。

也可以使用用矩阵的正定性间接证明矩阵的可逆性

$$\forall \boldsymbol{z} \neq \boldsymbol{0_D},\ \boldsymbol{z}^{\boldsymbol{T}}(\boldsymbol{X}\boldsymbol{X}^{\boldsymbol{T}} + \lambda\boldsymbol{I})\boldsymbol{z} = \left\|\boldsymbol{X}^{\boldsymbol{T}}\boldsymbol{z}\right\|_2^2 + \lambda\left\|\boldsymbol{z}\right\|_2^2 > 0$$

4.3.2　Tikhonov 正则 [11]

将先验推广至更一般的正态分布

$$\boldsymbol{w} \sim \mathcal{N}(\boldsymbol{0_D}, \boldsymbol{\Sigma_w})$$
$$p(\boldsymbol{w}) = \det(2\pi\boldsymbol{\Sigma_w})^{-1/2}\exp\left(-\frac{1}{2}\left\|\boldsymbol{w}\right\|_{\boldsymbol{\Sigma_w^{-1}}}^2\right)$$

考虑实对称矩阵 $\boldsymbol{\Sigma_w^{-1}}$ 的特征值分解

$$\boldsymbol{\Sigma_w^{-1}} = \boldsymbol{V}\boldsymbol{\Lambda}\boldsymbol{V}^{\boldsymbol{T}},\ \boldsymbol{V}\boldsymbol{V}^{\boldsymbol{T}} = \boldsymbol{I}$$

记

$$\boldsymbol{\Gamma} = \boldsymbol{\Lambda}^{\boldsymbol{1/2}}\boldsymbol{V}^{\boldsymbol{T}}$$

得到

$$p(\boldsymbol{w}) = \det(2\pi\boldsymbol{\Sigma_w})^{-1/2}\exp\left(-\frac{1}{2}\left\|\boldsymbol{\Gamma w}\right\|_2^2\right)$$

[11] 课件 lecture4 pp.24

得到 Tikhonov 正则（Tikhonov regularization）的目标函数表达式，其中 $\boldsymbol{\Gamma}$ 被称为 Tikhonov 矩阵

$$\min_{\boldsymbol{w}} \left\|\boldsymbol{y}-\boldsymbol{X}^{\boldsymbol{T}}\boldsymbol{w}\right\|_2^2+\sigma^2\left\|\boldsymbol{\Gamma}\boldsymbol{w}\right\|_2^2=\min_{\boldsymbol{w}} \left\|\boldsymbol{y}-\boldsymbol{X}^{\boldsymbol{T}}\boldsymbol{w}\right\|_2^2+\left\|\sigma\boldsymbol{\Gamma}\boldsymbol{w}\right\|_2^2$$

对于 Tikhonov 正则，这里要求 $\boldsymbol{\Gamma}$ 是列满秩的，因为要满足可逆的条件

$$\min_{\boldsymbol{w}} \left\|\boldsymbol{y}-\boldsymbol{X}^{\boldsymbol{T}}\boldsymbol{w}\right\|_2^2+\left\|\boldsymbol{\Gamma}\boldsymbol{w}\right\|_2^2$$

求导得到

$$\frac{\partial L}{\partial \boldsymbol{w}}=2\boldsymbol{X}(\boldsymbol{X}^{\boldsymbol{T}}\boldsymbol{w}-\boldsymbol{y})+2\boldsymbol{\Gamma}^{\boldsymbol{T}}\boldsymbol{\Gamma}\boldsymbol{w}$$

令导数为 0 得到

$$\boldsymbol{w}^*=(\boldsymbol{X}\boldsymbol{X}^{\boldsymbol{T}}+\boldsymbol{\Gamma}^{\boldsymbol{T}}\boldsymbol{\Gamma})^{-1}\boldsymbol{X}\boldsymbol{y}=(\boldsymbol{X}\boldsymbol{X}^{\boldsymbol{T}}+\boldsymbol{\Sigma}_{\boldsymbol{w}}^{-\mathbf{1}})^{-1}\boldsymbol{X}\boldsymbol{y}$$

可以使用用矩阵的正定性间接证明矩阵的可逆性，从而说明 Tikhonov 正则同样可以增强数值运算的稳定性

$$\forall \boldsymbol{z}\neq\mathbf{0},\ \boldsymbol{z}^{\boldsymbol{T}}(\boldsymbol{X}\boldsymbol{X}^{\boldsymbol{T}}+\boldsymbol{\Gamma}^{\boldsymbol{T}}\boldsymbol{\Gamma})\boldsymbol{z}=\left\|\boldsymbol{X}^{\boldsymbol{T}}\boldsymbol{z}\right\|_2^2+\left\|\boldsymbol{\Gamma}\boldsymbol{z}\right\|_2^2>0$$

4.3.3　广义 Tikhonov 正则 [12]

考虑 GLM [13]，更换指数分布族中的分布函数为一般的正态分布得到

$$\boldsymbol{y}\sim\mathcal{N}(\boldsymbol{X}^{\boldsymbol{T}}\boldsymbol{w},\boldsymbol{\Sigma}_{\boldsymbol{y}})$$

$$p(\boldsymbol{y}\mid\boldsymbol{X};\boldsymbol{w})=\det(2\pi\boldsymbol{\Sigma}_{\boldsymbol{y}})^{-1/2}\exp\left(-\frac{1}{2}\left\|\boldsymbol{y}-\boldsymbol{X}^{\boldsymbol{T}}\boldsymbol{w}\right\|_{\boldsymbol{\Sigma}_{\boldsymbol{y}}^{-\mathbf{1}}}^2\right)$$

将先验推广至一般的正态分布

$$\boldsymbol{w}\sim\mathcal{N}(\boldsymbol{\mu}_{\boldsymbol{w}},\boldsymbol{\Sigma}_{\boldsymbol{w}})$$

$$p(\boldsymbol{w})=\det(2\pi\boldsymbol{\Sigma}_{\boldsymbol{w}})^{-1/2}\exp\left(-\frac{1}{2}\left\|\boldsymbol{w}-\boldsymbol{\mu}_{\boldsymbol{w}}\right\|_{\boldsymbol{\Sigma}_{\boldsymbol{w}}^{-\mathbf{1}}}^2\right)$$

[12] 课件 lecture4 pp.24

[13] 详见 4.1

从而得到广义 Tikhonov 正则（Generalized Tikhonov regularization）的目标函数表达式

$$\min_{\boldsymbol{w}} \left\| \boldsymbol{y} - \boldsymbol{X}^{\boldsymbol{T}} \boldsymbol{w} \right\|_{\boldsymbol{\Sigma}_{\boldsymbol{y}}^{-1}}^{2} + \left\| \boldsymbol{w} - \boldsymbol{\mu}_{\boldsymbol{w}} \right\|_{\boldsymbol{\Sigma}_{\boldsymbol{w}}^{-1}}^{2}$$

对于广义 Tikhonov 正则，这里要求 $\boldsymbol{P}$ 和 $\boldsymbol{Q}$ 都是对称且正定的，因为需要保证协方差矩阵的性质

$$\min_{\boldsymbol{w}} \left\| \boldsymbol{y} - \boldsymbol{X}^{\boldsymbol{T}} \boldsymbol{w} \right\|_{\boldsymbol{P}}^{2} + \left\| \boldsymbol{w} - \boldsymbol{w}_{\boldsymbol{0}} \right\|_{\boldsymbol{Q}}^{2}$$

求导得到

$$\frac{\partial L}{\partial \boldsymbol{w}} = 2\boldsymbol{X}\boldsymbol{P}(\boldsymbol{X}^{\boldsymbol{T}}\boldsymbol{w} - \boldsymbol{y}) + 2\boldsymbol{Q}(\boldsymbol{w} - \boldsymbol{w}_{\boldsymbol{0}})$$

令导数为 0 得到

$$\boldsymbol{w}^{*} = (\boldsymbol{X}\boldsymbol{P}\boldsymbol{X}^{\boldsymbol{T}} + \boldsymbol{Q})^{-1}(\boldsymbol{X}\boldsymbol{P}\boldsymbol{y} + \boldsymbol{Q}\boldsymbol{w}_{\boldsymbol{0}})$$

可以非常类似地使用用矩阵的正定性间接证明矩阵的可逆性。

4.4　LASSO 回归 [14]

4.4.1　L^1 正则与 LASSO 回归模型[15]

仿照上文 MAP 的分析，当选择先验条件

$$\boldsymbol{w} \sim \text{Laplace}(\boldsymbol{0}_{\boldsymbol{D}}, b\boldsymbol{I}_{\boldsymbol{D}})$$
$$p(\boldsymbol{w}) = (2b)^{-D} \exp\left(-\frac{1}{b} \left\| \boldsymbol{w} \right\|_1 \right)$$

线性回归的优化问题转变为

$$\min_{\boldsymbol{w}} \frac{1}{2} \left\| \boldsymbol{y} - \boldsymbol{X}^{\boldsymbol{T}} \boldsymbol{w} \right\|_2^2 + \frac{\sigma^2}{b} \left\| \boldsymbol{w} \right\|_1$$

这样的回归方法被称为 Lasso 回归（Lasso regression）。

[14] 课件 lecture4 pp.25-29

[15] 课件 lecture4 pp.25-27

示意图[16]表明，当优化函数取得最优值（图中蓝色圆圈代表平方误差 $\left\|\boldsymbol{y}-\boldsymbol{X}^{\boldsymbol{T}}\boldsymbol{w}\right\|_2^2$ 的等值线张成的 L^2 球，红色部分代表正则项对参数 $\boldsymbol{w}$ 的约束形成的 L^2 和 L^1 球，在约束下目标函数的最优解已在图中标为 $\boldsymbol{w}^*$）时，对 Lasso 回归，$\boldsymbol{w}$ 的分量 w_1 被压缩为 0，从而在某种程度上说明 Lasso 回归具有减少模型参数量，达到防止模型过拟合的目的。之前在和徐君老师的谈话中徐老师用了一个很有意思的比喻就是“磕桌角”，Lasso 问题的正则项约束的等值线在每个边界点（存在某些参数为 0）都设置了一个“桌角”，原来的凸的损失函数就像吹起来的气球，最终两者在最优值时，气球总会磕到桌角上去。

事实上取 L^p 范数 $p<1$ 也能达到类似的效果，因为约束形成的 L^p 球也存在着这样的桌角，而使得最终求解的参数稀疏化的最直接的方法是使用 L^0 正则，但是此时 L^p 范数严格意义来说不能称为范数，为了取得更好的近似很多情况下我们需要对 L^p 范数进行凸的近似处理，而某种角度上最佳的凸近似就是 L^1 范数 [17]，因而为了达到相同的减少模型参数的目的更倾向于采用 L^1 正则范数设计正则项。

4.4.2　软阈值迭代法[18]

对于 Lasso 回归问题

$$\min_{\boldsymbol{w}}\frac{1}{2}\left\|\boldsymbol{y}-\boldsymbol{X}^{\boldsymbol{T}}\boldsymbol{w}\right\|_2^2+\lambda\|\boldsymbol{w}\|_1$$

这里引入凸集的次微分的概念，对于一阶可导的凸函数

$$\forall\boldsymbol{x_0},\boldsymbol{x}\in I, f(\boldsymbol{x})-f(\boldsymbol{x_0})\geqslant\nabla f(\boldsymbol{x_0})^T(\boldsymbol{x}-\boldsymbol{x_0})$$

事实上不是所有的凸函数在定义域上都具有一阶可导的性质，故我们考虑拓展梯度的定义。若对于凸函数 $f(\boldsymbol{x})$，存在 $\boldsymbol{c}$ 满足

$$\forall\boldsymbol{x},\boldsymbol{x_0}\in I, f(\boldsymbol{x})-f(\boldsymbol{x_0})\geqslant\boldsymbol{c}^{\boldsymbol{T}}(\boldsymbol{x}-\boldsymbol{x_0})$$

则 $\boldsymbol{c}$ 被称为次梯度（Subgradient）。我们将所有满足条件的 $\boldsymbol{c}$ 的集合称为凸函数关于 $\boldsymbol{x}$ 的次微分（Subdifferential），记为 $\partial f(\boldsymbol{x})$。当函数在该点可导时，

[16]课件 lecture4 p.27

[17]详见 1.1 结尾部分

[18]课件 lecture4 pp.28-29

次微分是唯一的，即为该点的梯度。有关次微分我们有以下重要结论 [19]

$$\boldsymbol{x}^* = \arg\min_{\boldsymbol{x}} f(\boldsymbol{x}) \Leftrightarrow \mathbf{0} \in \partial f(\boldsymbol{x})$$

有关凸函数和次微分更多相关内容请见附录 A。

回到 Lasso 回归问题，我们需要求解

$$\mathbf{0} \in \partial \left(\frac{1}{2} \left\| \boldsymbol{y} - \boldsymbol{X}^{\boldsymbol{T}} \boldsymbol{w} \right\|_2^2 + \lambda \left\| \boldsymbol{w} \right\|_1 \right) = \boldsymbol{X}(\boldsymbol{X}^{\boldsymbol{T}} \boldsymbol{w} - \boldsymbol{y}) + \lambda \partial \left\| \boldsymbol{w} \right\|_1$$

为了简化问题取 $\boldsymbol{X}\boldsymbol{X}^{\boldsymbol{T}} = \boldsymbol{I}$ 即保证 $\boldsymbol{X}$ 的列向量是单位正交的，问题变为

$$\mathbf{0} \in \boldsymbol{w} - \boldsymbol{X}\boldsymbol{y} + \lambda \partial \left\| \boldsymbol{w} \right\|_1$$

对于向量的 L^1 范数，容易利用单侧导数求得次微分的一个子集[19]

$$\left(\partial \left\| \boldsymbol{x} \right\|_1\right)_i = \begin{cases} 1 & x_i > 0 \\ -1 & x_i < 0 \\ [1,-1] & x_i = 0 \end{cases} \quad , \ , \ i = 1, 2, \cdots, D$$

考虑向量的每一维元素

$$0 \in w_i - \boldsymbol{x}_{\boldsymbol{i}}^{\boldsymbol{T}} \boldsymbol{y} + \lambda \partial \left| w_i \right|$$

分类讨论得到：

情况一　$w_i > 0$

$$\Rightarrow \partial \left| w_i \right| = 1 \Rightarrow w_i = \boldsymbol{x}_{\boldsymbol{i}}^{\boldsymbol{T}} \boldsymbol{y} - \lambda > 0 \Rightarrow \boldsymbol{x}_{\boldsymbol{i}}^{\boldsymbol{T}} \boldsymbol{y} > \lambda$$

情况二　$w_i < 0$

$$\partial \left| w_i \right| = -1 \Rightarrow w_i = \boldsymbol{x}_{\boldsymbol{i}}^{\boldsymbol{T}} \boldsymbol{y} + \lambda \Rightarrow \boldsymbol{x}_{\boldsymbol{i}}^{\boldsymbol{T}} \boldsymbol{y} < -\lambda$$

情况三　$w_i = 0$

$$-1 \leqslant \partial \left| w_i \right| \leqslant 1 \Rightarrow \boldsymbol{x}_{\boldsymbol{i}}^{\boldsymbol{T}} \boldsymbol{y} - \lambda \leqslant w_i \leqslant \boldsymbol{x}_{\boldsymbol{i}}^{\boldsymbol{T}} \boldsymbol{y} + \lambda \Rightarrow \left| \boldsymbol{x}_{\boldsymbol{i}}^{\boldsymbol{T}} \boldsymbol{y} \right| \leqslant \lambda$$

[19] 详见 A.3 结尾部分

因而考虑所有情况下最后的结果满足

$$w_i^* = \begin{cases} \boldsymbol{x_i^T y} - \lambda & \boldsymbol{x_i^T y} > \lambda \\ 0 & \left|\boldsymbol{x_i^T y}\right| \leqslant \lambda \\ \boldsymbol{x_i^T y} + \lambda & \boldsymbol{x_i^T y} < -\lambda \end{cases}$$

我们定义软阈值算子（Soft thresholding operator）

$$(S_\lambda(\boldsymbol{x}))_i = \begin{cases} x_i - \lambda & x_i > \lambda \\ 0 & |x_i| \leqslant \lambda \\ x_i + \lambda & x_i < -\lambda \end{cases}, \ i = 1, 2, \cdots, D$$

于是最终结果可以写为

$$\boldsymbol{w^*} = S_\lambda(\boldsymbol{Xy})$$

式子 $\boldsymbol{Xy}$ 也是在满足 $\boldsymbol{XX^T} = \boldsymbol{I}$ 条件下线性回归的最小二乘解。显然对于一般的 Lasso 回归问题 $\boldsymbol{X}$ 的列向量是单位正交通常不成立，此时我们需要考虑对表达式进行变形，配凑出单位正交的向量。考虑固定 $\boldsymbol{w}$ 其余维度的元素，单独对 $\boldsymbol{w}$ 的某一维元素进行优化

$$\begin{aligned} w_d^{(t+1)} &= \arg\min_w \frac{1}{2} \left\| \boldsymbol{y} - \sum_{i \neq d} \boldsymbol{x_i} w_i^{(t)} - \boldsymbol{x_d} w \right\|_2^2 + \lambda |w| \\ &= \arg\min_w \frac{1}{2} \left\| \frac{1}{\|\boldsymbol{x_d}\|_2} \left(\boldsymbol{y} - \sum_{i \neq d} \boldsymbol{x_i} w_i^{(t)} \right) - \frac{\boldsymbol{x_d}}{\|\boldsymbol{x_d}\|_2} w \right\|_2^2 + \frac{\lambda}{\|\boldsymbol{x_d}\|_2^2} |w| \end{aligned}$$

此时我们构造出单位正交的向量 $\boldsymbol{x_d} / \|\boldsymbol{x_d}\|_2^2$ 是正交的。记取除第 d 维向量的 $\boldsymbol{X}$ 和 $\boldsymbol{w^{(t)}}$ 为 $\boldsymbol{X_{-d}}$ 和 $\boldsymbol{w_{-d}^{(t)}}$，即

$$\begin{aligned} \boldsymbol{X_{-d}} &= (\boldsymbol{x_1}, \boldsymbol{x_2}, \cdots, \boldsymbol{x_{d-1}}, \boldsymbol{x_{d+1}}, \cdots, \boldsymbol{x_D}) \\ \left(\boldsymbol{w_{-d}^{(t)}}\right)^T &= (w_1^{(t)}, w_2^{(t)}, \cdots, w_{d-1}^{(t)}, w_{d+1}^{(t)}, \cdots, w_D^{(t)})^T \end{aligned}$$

我们构造得到了单位向量（由于向量组中只有一个向量，它当然是正交的），代入参数得到

$$
\begin{aligned}
w_d^{(t+1)} &= S_{\lambda/\|\boldsymbol{x_d}\|_2^2}\left(\frac{\boldsymbol{x_d^T}}{\|\boldsymbol{x_d}\|_2^2}\left(\boldsymbol{y}-\sum_{i\neq d}\boldsymbol{x_i}w_i^{(t)}\right)\right) \\
&= \frac{1}{\|\boldsymbol{x_d}\|_2^2}S_\lambda\left(\boldsymbol{x_d^T}\left(\boldsymbol{y}-\boldsymbol{X_{-d}^T}\boldsymbol{w_{-d}^{(t)}}\right)\right)
\end{aligned}
$$

于是我们得到了求解 Lasso 回归问题的软阈值迭代法。

4.5　弹性网络正则 [20]

弹性网络正则（Elastic net Regularization）实际上就是岭回归和 Lasso 回归的折衷，兼具岭回归数值运算的稳定性和 Lasso 回归能够减少模型参数量的特点。弹性网络正则很容易转化为 Lasso 问题进行求解，只需使用深度学习中常用的参数合并的技巧

$$
\begin{aligned}
&\min_{\boldsymbol{w}}\frac{1}{2}\left\|\boldsymbol{y}-\boldsymbol{X^T}\boldsymbol{w}\right\|_2^2+\lambda_1\|\boldsymbol{w}\|_1+\lambda_2\|\boldsymbol{w}\|_2^2 \\
&\quad=\min_{\boldsymbol{w}}\frac{1}{2}\left\|\begin{pmatrix}\boldsymbol{y}\\ \boldsymbol{0_D}\end{pmatrix}-\begin{pmatrix}\boldsymbol{X^T}\\ \sqrt{\lambda_2}\boldsymbol{I_D}\end{pmatrix}\boldsymbol{w}\right\|_2^2+\lambda_1\|\boldsymbol{w}\|_1
\end{aligned}
$$

如果要使用 MAP 的分析弹性网络正则，我们可以考虑选择先验条件为基于正态分布和 Laplace 分布的混合模型，其中 c 介于 0 和 1 之间，控制了两个成分的比例

$$
p(\boldsymbol{w})=c(2\pi\gamma^2)^{-D/2}\exp\left(-\frac{1}{2\gamma^2}\|\boldsymbol{w}\|_2^2\right)+(1-c)(2b)^{-D}\exp\left(-\frac{1}{b}\|\boldsymbol{w}\|_1\right)
$$

利用 Jensen 不等式（Jensen's inequality）[21] 选取对数为自然对数考虑上式的负对数的上界得到

$$
-\ln p(\boldsymbol{w})\leqslant\frac{c}{2\gamma^2}\|\boldsymbol{w}\|_2^2+\frac{1-c}{b}\|\boldsymbol{w}\|_1+C
$$

其中 C 是一个和 $\boldsymbol{w}$ 无关的常数。因而如果考虑优化后验概率的上界我们就得到了弹性网络正则。

[20] 课件 lecture4 p.30

[21] 详见 A.1 中间部分

4.6　最小化 MAE 的求解 [22]

对于最小化 MAE [23] 优化问题的求解，我们可以考虑将目标函数和约束条件转化为线性约束。以下为了和实数的比较运算区分，我们定义向量的按维度的比较运算，按维度 $<$ 记为 $\prec$，按维度 $\leqslant$ 记为 $\preccurlyeq$，按维度 $>$ 记为 $\succ$，按维度 $\geqslant$ 记为 $\succcurlyeq$。MAE 问题可以使用单纯形法（Simplex method）进行求解，考虑

$$\min_{\boldsymbol{a}} \sum_{i=1}^{D} a_i = \mathbf{1}_{\boldsymbol{D}}^{\boldsymbol{T}} \boldsymbol{a}$$

$$s.t. \begin{cases} \left| \boldsymbol{y} - \boldsymbol{X}^{\boldsymbol{T}} \boldsymbol{w} \right| = \boldsymbol{a} \\ \boldsymbol{a} \succcurlyeq 0 \end{cases}$$

可以将 a_i 修改为 $|y_i - \boldsymbol{X_i w}|$ 的上界从而在优化的过程中迫使最优解在 a_i 触碰到边界时取到，从而将优化条件修改为

$$s.t. \begin{cases} \left| \boldsymbol{y} - \boldsymbol{X}^{\boldsymbol{T}} \boldsymbol{w} \right| \preccurlyeq \boldsymbol{a} \\ \boldsymbol{a} \succcurlyeq 0 \end{cases} \quad \Rightarrow s.t. \begin{cases} \boldsymbol{y} - \boldsymbol{X}^{\boldsymbol{T}} \boldsymbol{w} \preccurlyeq \boldsymbol{a} \\ -\boldsymbol{y} + \boldsymbol{X}^{\boldsymbol{T}} \boldsymbol{w} \preccurlyeq -\boldsymbol{a} \\ \boldsymbol{a} \succcurlyeq 0 \end{cases}$$

但是单纯形法复杂度太高，对于大规模矩阵运算而言我们通常不会采用这个思路，有时候我们甚至宁愿放弃一部分精度去选择更高效的解法。IRLS（Iteratively reweighted least squares）实际上将由 L^p 范数定义的误差项（$p \neq 0$）通过每一项的拆分转化为了求解最小二乘解的过程，这里采用了交替优化的思想，对引入的中间矩阵 $\boldsymbol{A}^{(t)}$ 和 $\boldsymbol{w}^{(t)}$ 进行交替优化，这是因为直接求取 $\boldsymbol{w}$ 的闭式解是困难的。

[22] 课件 lecture4 p.31

[23] 详见 3.3

第五章　非线性回归与核技巧

5.1　Jacobian 矩阵简介 [1]

对 $\mathbb{R}^M$ 上可导的向量函数 $\boldsymbol{f}(\boldsymbol{x})$

$$\boldsymbol{f}:\mathbb{R}^M\mapsto\mathbb{R}^N$$

对向量 $\boldsymbol{x}$ 求导得到的 Jacobian 矩阵（Jacobian matrix）定义如下

$$\boldsymbol{J_f}=\left(\frac{\partial \boldsymbol{f}}{\partial x_1},\frac{\partial \boldsymbol{f}}{\partial x_2},\cdots,\frac{\partial \boldsymbol{f}}{\partial x_M}\right)=(\nabla_{\boldsymbol{x}}\boldsymbol{f_1},\nabla_{\boldsymbol{x}}\boldsymbol{f_2},\cdots,\nabla_{\boldsymbol{x}}\boldsymbol{f_N})^T=\nabla_{\boldsymbol{x}}^T\boldsymbol{f}$$

对于复合函数 $\boldsymbol{h}(\boldsymbol{x})=\boldsymbol{f}(\boldsymbol{g}(\boldsymbol{x}))$ 而言，存在链式法则（Chain rule）

$$\boldsymbol{J_h}(\boldsymbol{x})=\boldsymbol{J_f}(\boldsymbol{g}(\boldsymbol{x}))\boldsymbol{J_g}(\boldsymbol{x})$$

因而对于矩阵求导运算，也存在链式法则，只不过链需要进行反转

$$\frac{\partial \boldsymbol{y}}{\partial \boldsymbol{x}}=\boldsymbol{J_y}(\boldsymbol{x})^T=\boldsymbol{J_u}(\boldsymbol{x})^T\boldsymbol{J_y}(\boldsymbol{u}(\boldsymbol{x}))^T=\frac{\partial \boldsymbol{u}}{\partial \boldsymbol{x}}\frac{\partial \boldsymbol{y}}{\partial \boldsymbol{u}}$$

链式法则对简化复杂的向量函数求导的计算意义重大，它给我们计算提供的更多的是一种“并行”和“整体”的思维。当遇到对矩阵求导时，上面提供的简单链式法则往往不能直接套用，因而可以考虑将对矩阵的求导运算拆分为对向量的求导运算。来看看以下几个例子：

例一

$$\frac{\partial\left\|\boldsymbol{y}-\boldsymbol{X^T}\boldsymbol{w}\right\|_2^2}{\partial \boldsymbol{w}}=\frac{\partial\left(\boldsymbol{y}-\boldsymbol{X^T}\boldsymbol{w}\right)}{\partial \boldsymbol{w}}\frac{\partial\left\|\boldsymbol{y}-\boldsymbol{X^T}\boldsymbol{w}\right\|_2^2}{\partial\left(\boldsymbol{y}-\boldsymbol{X^T}\boldsymbol{w}\right)}=2\boldsymbol{X}(\boldsymbol{X^T}\boldsymbol{w}-\boldsymbol{y})$$

[1] 课件 lecture5 pp.10-11

例二

$$\begin{aligned}\frac{\partial\left\|\boldsymbol{y}-\boldsymbol{X}^{\boldsymbol{T}}\boldsymbol{w}\right\|_{\boldsymbol{P}}^{2}}{\partial\boldsymbol{w}}&=\frac{\partial\left(\boldsymbol{y}-\boldsymbol{X}^{\boldsymbol{T}}\boldsymbol{w}\right)}{\partial\boldsymbol{w}}\frac{\partial\left\|\boldsymbol{y}-\boldsymbol{X}^{\boldsymbol{T}}\boldsymbol{w}\right\|_{\boldsymbol{P}}^{2}}{\partial\left(\boldsymbol{y}-\boldsymbol{X}^{\boldsymbol{T}}\boldsymbol{w}\right)}\\&=\boldsymbol{X}(\boldsymbol{P}+\boldsymbol{P}^{\boldsymbol{T}})(\boldsymbol{X}^{\boldsymbol{T}}\boldsymbol{w}-\boldsymbol{y})\end{aligned}$$

例三

$$\begin{aligned}\frac{\partial\left\|\boldsymbol{y}-\boldsymbol{X}^{\boldsymbol{T}}\boldsymbol{w}\right\|_{2}^{2}}{\partial\boldsymbol{X}}&=\left(\frac{\partial\left\|\boldsymbol{y}-\boldsymbol{X}^{\boldsymbol{T}}\boldsymbol{w}\right\|_{2}^{2}}{\partial\boldsymbol{x}_{\boldsymbol{j}}}\right)_{D\times N}=\left(\frac{\partial\left(\boldsymbol{y}-\boldsymbol{X}^{\boldsymbol{T}}\boldsymbol{w}\right)}{\partial\boldsymbol{x}_{\boldsymbol{j}}}\frac{\partial\left\|\boldsymbol{y}-\boldsymbol{X}^{\boldsymbol{T}}\boldsymbol{w}\right\|_{2}^{2}}{\partial\left(\boldsymbol{y}-\boldsymbol{X}^{\boldsymbol{T}}\boldsymbol{w}\right)}\right)_{D\times N}\\&=2\left(\frac{\partial}{\partial\boldsymbol{x}_{\boldsymbol{j}}}\left(\boldsymbol{y}-\sum_{d=1}^{D}\boldsymbol{x}_{\boldsymbol{d}}w_{d}\right)\left(\boldsymbol{y}-\boldsymbol{X}^{\boldsymbol{T}}\boldsymbol{w}\right)\right)_{D\times N}\\&=2\left(w_{j}\boldsymbol{I}\left(\boldsymbol{X}^{\boldsymbol{T}}\boldsymbol{w}-\boldsymbol{y}\right)\right)_{D\times N}=2(\boldsymbol{X}^{\boldsymbol{T}}\boldsymbol{w}-\boldsymbol{y})\boldsymbol{w}^{\boldsymbol{T}}\end{aligned}$$

Jacobian 矩阵的更多相关内容请见附录 B。

5.2　RKHS 简介 [2]

理解核（Kernel）最重要的不是泛函分析（Functional analysis）的理论推导，而是核的思想。在传统的线性回归和线性分类任务中，我们通常需要分别假设数据大致是线性相关和线性可分的；在传统的线性降维任务中，降维后的数据有时将被投入到回归和分类任务中，因而保证原数据的线性有时也是必要的。事实上原数据的线性相关和线性可分往往很难做到，低维空间对数据的表征能力已经达到了极限，于是这给我们带来了一种想法，能否通过将数据升维至表征能力更强的高维空间，使得数据在高维空间中变得线性相关或是线性可分呢？将数据从非线性变为线性不一定要直接对数据进行移动，而是通过对空间的变换间接地实现这一点（这点有点像线性空间的基变换），这种变换使得低维空间上的数据点被抬升至了高维，用一个经典的解释是将数据点从平面上扬起到空间中，使得非线性的数据在空间中具有线性。显然线性变换不能胜任这个任务，因为通常而言线性变换不会改变数据的线性，我们需要向数据中增加非线性的因素。举一个简单的例子，对于两

[2]课件 lecture5 pp.17-23

类二维的样本点，其中 A 类在椭圆边界之内，B 类在椭圆边界之外。为了简化问题，假定椭圆边界为

$$f(x,y)=2(x-1)^2+(y-1)^2=1$$

即

$$\forall(x_0,y_0)\in A, f(x_0,y_0)<1$$

$$\forall(x_0,y_0)\in B, f(x_0,y_0)>1$$

这样的两类样本点显然不是线性可分的,但是如果将数据升维至 (x,y,x^2,y^2),我们显然能够找到一个分界面

$$2x^2-4x+y^2-2y+2=0$$

这个分界面对于升维后的空间而言是线性的，因而数据又变得线性可分了。

由我们以上分析，我们需要找到一个合适的从低维空间映向高维的非线性映射，使得数据具有某种线性。回到我们最初想解决的问题，即将数据从表征能力弱的低维空间映向表征能力强的高维空间，为了定义这个高维空间，我们先定义 RKHS 。定义 Hilbert 空间 $\mathcal{H}$ 上的连续的 evaluation functional $L_{\boldsymbol{x}}$

$$L_{\boldsymbol{x}}:\mathcal{H}\mapsto\mathbb{R}, L_{\boldsymbol{x}}(f)=f(\boldsymbol{x})$$

$L_{\boldsymbol{x}}$ 是 $\mathcal{H}$ 上的有界算子，考虑 $\mathcal{H}$ 上的内积运算，其满足

$$\forall\boldsymbol{x}\in\mathcal{X},\ \exists M_{\boldsymbol{x}}>0,\ |L_{\boldsymbol{x}}(f)|=|f(\boldsymbol{x})|\leqslant M_{\boldsymbol{x}}\langle f,f\rangle_{\mathcal{H}}$$

对于任意 $\boldsymbol{x}\in\mathcal{X}$，存在一个函数 $K_{\boldsymbol{x}}\in\mathcal{H}$，其满足再生性质（Reproducing property）

$$f(\boldsymbol{x})=L_{\boldsymbol{x}}(f)=\langle f,K_{\boldsymbol{x}}\rangle_{\mathcal{H}}=\int_{\boldsymbol{y}\in\mathcal{X}}f(\boldsymbol{y})K_{\boldsymbol{x}}(\boldsymbol{y})\mathrm{d}\boldsymbol{y}$$

由此可以定义 $\mathcal{H}$ 上的一个再生核（Reproducing kernel）K

$$K:\mathcal{X}^2\mapsto\mathbb{R},\ K(\boldsymbol{x},\boldsymbol{y})=\langle K_{\boldsymbol{y}},K_{\boldsymbol{x}}\rangle_{\mathcal{H}}=K_{\boldsymbol{x}}(\boldsymbol{y})=K_{\boldsymbol{y}}(\boldsymbol{x})$$

即再生核函数实际上满足对称性，当取 $\boldsymbol{x} = \boldsymbol{y}$ 时由内积的正定性保证了函数值的是正的。由此我们把这个关于再生核 K 的 Hilbert 空间称为再生核 Hilbert 空间（Reproducing kernel Hilbert space / RKHS），记作 $\mathcal{H}_K$。

定义映射

$$\phi(\boldsymbol{x}) = K_{\boldsymbol{x}}$$

函数 ϕ 实际上定义了一种从低维空间的样本点到高维空间 $\mathcal{F}$ 的映射，我们事实上想要求解的非线性映射就是这个。我们也把这个高维空间 $\mathcal{F}$ 称为特征空间（Feature space）

$$\phi : \mathcal{X} \mapsto \mathcal{F},\ \mathcal{F} = \operatorname{span}(\{K_{\boldsymbol{x}}\}_{\boldsymbol{x}\in\mathcal{X}}) \subset \mathcal{H}_K$$

事实上无需过分关心 $\mathcal{H}_K$ 所具有的形式，一个 trivial 的 $\mathcal{H}_K$ 事实上就可以直接取

$$\mathcal{H}_K := \mathcal{F} = \operatorname{span}(\{K_{\boldsymbol{x}}\}_{\boldsymbol{x}\in\mathcal{X}}) = \operatorname{span}(\{\phi(\boldsymbol{x})\}_{\boldsymbol{x}\in\mathcal{X}})$$

即满足定义的 $\mathcal{H}$ 事实上可以直接以给定的 ϕ 映射样本点得到基生成。然而事实上我们一般不会通过直接构造映射 ϕ 来确定由低维空间到 $\mathcal{F}$ 的映射，而是借助再生核 K 和 $\mathcal{F}$ 上内积运算的联系得到内积运算的结果。如此借助核函数来实现从低维的样本空间到高维的特征空间映射的技巧被称为核技巧（Kernel trick）。其中核函数的选取保证对称性和正定性即可[3]。上面引入 RKHS 的过程也许看起来比较抽象，我们在附录 C 提供了一个 RKHS 更直观且更贴近核技巧的实现的构造过程。

在课件中[4]出现了一元函数 κ 是非参统计（Nonparametric statistics）中的核函数（Kernel function）[5]。这样的一元的核函数和我们接下要讨论的 RKHS 中二元的核函数虽然形式可能比较相近但是在本质上有所不同。核这个词在机器学习中有很多不同的含义，注意加以区分。以下列出了我们常用的核函数，其中前三个均含有超参数 h [6]，从如下定义的核函数可以非常自然地过渡到非参统计中的核函数：

RBF

[3] 见 C.3 中间部分

[4] 课件 lecture5 pp.15-16,20,24

[5] 详见 G.2

[6] 这个超参数 h 可以解读为 G.2 非参统计定义的核函数的带宽（bandwidth）

$$K_h(\boldsymbol{x},\boldsymbol{y}) = \exp\left(-\frac{\|\boldsymbol{x}-\boldsymbol{y}\|_2^2}{h}\right)$$

Gate

$$K_h(\boldsymbol{x},\boldsymbol{y}) = \begin{cases} 1/h & \|\boldsymbol{x}-\boldsymbol{y}\|_1 \leqslant h \\ 0 & \text{otherwise} \end{cases}$$

Triangle

$$K_h(\boldsymbol{x},\boldsymbol{y}) = \begin{cases} \frac{2}{h}(1-\frac{\|\boldsymbol{x}-\boldsymbol{y}\|_1}{h}) & \|\boldsymbol{x}-\boldsymbol{y}\|_1 \leqslant /h \\ 0 & \text{otherwise} \end{cases}$$

Linear

$$K(\boldsymbol{x},\boldsymbol{y}) = \boldsymbol{x}^{\boldsymbol{T}}\boldsymbol{y}$$

5.3　核岭回归 [7]

在非线性回归问题中，我们可以寻找 $\mathcal{H}_K$ 中的一个最优的函数 f^*，使得对于任意给定数据 $\boldsymbol{x}$，其与真实标签 y 之间的平均损失最小化。则基于 MAP 的带正则项[8]的优化问题可以表述为

$$f^* := \arg\min_{f\in\mathcal{H}_K} \mathrm{E}_{\boldsymbol{x},y\sim\mathcal{P}}[L(y,f(\boldsymbol{x}))] + \mathcal{R}(f)$$

Representer 定理（Representer Theorem）指出，$\mathcal{H}_K$ 中的最优解必然能够表示为有限个核函数的边缘函数的线性组合，且个数小于数据量的大小 N。该定理论证了回归问题中前文所述的这种积分近似的合理性，且表明这种线性组合实际上是可以进一步约减的，用数学符号表述为

$$\exists\boldsymbol{\alpha}\in\mathbb{R}^M,\ f^*(\boldsymbol{x}) = \sum_{n=1}^{M}\alpha_n K_{\boldsymbol{x}}(\boldsymbol{x}'_{\boldsymbol{n}}),\ M<N \Rightarrow f^* = \sum_{n=1}^{M}\alpha_n K_{\boldsymbol{x}'_{\boldsymbol{n}}}$$

[7] 课件 lecture5 pp.21,25

[8] 详见 4.2 结尾部分

这为我们积分的近似

$$f(\boldsymbol{x}) = \langle f, K_{\boldsymbol{x}} \rangle_{\mathcal{H}_K} = \int_{\mathcal{X}} f(\boldsymbol{y}) K_{\boldsymbol{x}}(\boldsymbol{y}) \mathrm{d}\boldsymbol{y} \approx \sum_{i=1}^{N} f(\boldsymbol{x_i}) K_{\boldsymbol{x}}(\boldsymbol{x_i}) = \sum_{i=1}^{N} \langle f(\boldsymbol{x_i}) \phi(\boldsymbol{x_i}), \phi(\boldsymbol{x}) \rangle$$

提供了坚实的理论依据。为了方便表达式求解，我们通常仍考虑所有的样本点，即使这有时会为我们带来高昂的计算量。

选取正则项为 $\mathcal{H}_K$ 中拟合函数 f 的内积，核岭回归（Kernel ridge regression / KRR）问题可以表述为如下优化问题

$$f^* := \arg\min_{f \in \mathcal{H}_K} \|\boldsymbol{y} - f(\boldsymbol{X})\|_2^2 + \lambda \langle f, f \rangle_{\mathcal{H}_K}$$

根据 Representer 定理最优解必然可以写为如下形式

$$f^* = \sum_{n=1}^{N} \alpha_n K_{\boldsymbol{x_n}}$$

在这种形式下，我们使用了和数据集大小相等个数的向量线性地表出了 f^*，在上述约束下 $\{K_{\boldsymbol{x_i}}\}$ 即 $\{\phi(\boldsymbol{x_i})\}$ 可以理解为空间的一组表出最优解的“基”（之所以打引号是因为它们可能是线性相关的，这和向量空间的基的定义不太一样），从而解决了原来空间的一组基可能为无穷多个的情况。对于 f 而言，其在“基”下的坐标为 $\boldsymbol{\alpha}$，对 $K_{\boldsymbol{x_j}}$ 而言，其在“基”下的坐标显然为 $\boldsymbol{e_j}$。记“基”对应的内积矩阵为

$$\boldsymbol{K} := (K(\boldsymbol{x_i}, \boldsymbol{x_j}))_{N \times N}$$

以及映射得到的特征矩阵

$$\phi(\boldsymbol{X}) := (\phi(\boldsymbol{x_i}))_{\dim(\mathcal{F}) \times N}$$

这里取向量 $\boldsymbol{X}$ 为 $D \times N$ 向量，则

$$f(\boldsymbol{X}) := (f(\boldsymbol{x_j}))_{N \times 1} = \left(\langle f, K_{\boldsymbol{x_j}} \rangle \right)_{N \times 1} = (\boldsymbol{e_j} \boldsymbol{K} \boldsymbol{\alpha^T})_{N \times 1} = \boldsymbol{K\alpha} = \phi(\boldsymbol{X})^T \phi(\boldsymbol{X}) \boldsymbol{\alpha}$$

f 和自身的内积也容易得到为

$$\langle f, f \rangle_{\mathcal{H}_K} = \boldsymbol{\alpha^T K \alpha} = \boldsymbol{\alpha^T} \phi(\boldsymbol{X})^T \phi(\boldsymbol{X}) \boldsymbol{\alpha} = \|\phi(\boldsymbol{X}) \boldsymbol{\alpha}\|_2^2$$

因而当考虑最优解满足的形式时，优化问题变为

$$\min_{\boldsymbol{\alpha}} \|\boldsymbol{y} - \boldsymbol{K}\boldsymbol{\alpha}\|_2^2 + \lambda \boldsymbol{\alpha}^{\boldsymbol{T}} \boldsymbol{K} \boldsymbol{\alpha}$$

取 $\boldsymbol{w} := \phi(\boldsymbol{X})\boldsymbol{\alpha}$，其代表了特征空间 $\mathcal{F}$ 上的通过数据特征为线性组合得到的一个向量。得到优化问题为

$$\min_{\boldsymbol{w}} \left\|\boldsymbol{y} - \phi(\boldsymbol{X})^T \boldsymbol{w}\right\|_2^2 + \lambda \|\boldsymbol{w}\|_2^2$$

因而我们从 $\mathcal{H}_K$ 中寻找一个用于拟合数据的非线性函数这样的非线性回归问题就是对特征空间 $\mathcal{F}$ 中数据点高维特征的线性回归问题。当考虑以函数与自身的内积为正则项时，问题转变为特征空间中的岭回归问题。特别地，当使用线性核 $K(\boldsymbol{x}, \boldsymbol{y}) = \boldsymbol{x}^{\boldsymbol{T}} \boldsymbol{y}$ 时上式退化为原空间的岭回归。

这是在核技巧中常说的一句话，尽管我们有时甚至完全不知道用于升维的映射 ϕ 的具体表达式，但是我们还是通过自己定义的内积矩阵 $\boldsymbol{K}$ 实现了特征空间的运算。通过核技巧我们利用对称函数 K 避免了对升维的非线性映射 ϕ 的直接建模和计算，尽管构造出的特征空间的维度可能非常高甚至有可能是无限维的，但是我们往往能够找到一种巧妙的方式将优化问题解出来，我们接下来的章节会看到更多的案例。但是需要注意的是，核技巧在很多时候具有参数量大、数据量需求大和对 SGD 不友好等不足，因而虽然理论推导十分具有美感，在实际应用中却存在着很多限制。只要你有一双发现内积的眼睛，很多机器学习中的问题都可以套用核技巧，在后续我们会看到几个具体的案例。

第三部分

数据降维

第六章　线性降维

6.1　维度诅咒 [1]

我们在上一讲总是在强调高维数据具有的优势，数据表征能力强，将低维数据升至高维能够提升模型更多的可能性，观察到数据在低维更难以观察到的特征等等..... 但是高维既是祝福（Blessing）也是诅咒（Curse），高维数据带来的危害常常是不能忽视的。

6.1.1　数据量需求的激增[2]

在 5.3 中我们提到了数据通过核技巧提升至高维这种技巧在实际应用中存在对参数量需求大和计算量大等不足的问题。假设另一种情景，考虑数据在 D 维空间分布，假定每一维数据介于 0 和 1 之间，若想覆盖一个固定的比例 α 的数据，在每个维度上考虑区间长度为 p 的数据，则 p 需要满足

$$p^D = \alpha \Rightarrow \alpha = p^{1/D}$$

即随着维度升高，仅仅要覆盖一个很小比例的数据，在每个维度上几乎都要考虑到所有的数据在该维度的分量，因而在高维空间中容易出现数据不足的问题。

另外，考虑 D 维的 $\{0,1\}$ 向量，它也许代表了某个 D 个阶段决策的序列，其组合总共有 2^D 种；对于 D 维的单位超立方体（Hypercube），以 10^{-1} 为步长对立方体进行采样，所采样的点数为 10^D，以上几个案例均表明，随着维度的升高，在机器学习中所需要考虑的对象的数量不仅可能趋于正无穷，而且其增长可能是指数级的，在高维空间的一个简单的操作，其时

[1] 课件 lecture6 pp.3-5

[2] 课件 lecture6 pp.5

间复杂度可能高到难以接受[3]。

6.1.2　Euclidean 距离的失效[4]

高维 Euclidean 空间上的以 $\boldsymbol{c}$ 为球心、半径为 R 高维超球(Hypersphere)被定义为到 $\boldsymbol{c}$ 的 Euclidean 距离小于常数 R 的点的集合。考虑半径为 R 的 D 维超球的体积计算公式，这里用 $V_D(R)$ 表示；并考虑 $D+1$ 维超球的 D 维表面的表面积计算公式，这里用 $S_D(R)$ 表示 [5]

$$V_D(R) = \frac{\pi^{D/2}}{\Gamma(D/2+1)} R^D$$

$$S_{D-1}(R) = \frac{2\pi^{D/2}}{\Gamma(D/2)} R^{D-1}$$

证明可以考虑超球体积和表面积之间的联系，和 D 维超球和 $D+1$ 维超球体积的联系，通过 B.1 提供的积分换元公式事实上我们可以证明半径为 R 的超球和单位超球的体积扩大了 R^D 倍，而表面积扩大了 R^{D-1} 倍，因而我们只需证明

$$V_D(1) = \frac{\pi^{D/2}}{\Gamma(D/2+1)}$$

$$S_{D-1}(1) = \frac{2\pi^{D/2}}{\Gamma(D/2)}$$

我们将超球视为由一层层球壳围成的几何体，以下求导的过程类似于给洋葱剥最外一层壳的过程

$$S_D(R) = \frac{\mathrm{d}V_{D+1}(R)}{\mathrm{d}R} \Rightarrow S_D(1) = (D+1)V_{D+1}(1)$$

从另一个角度思考 $D+1$ 维超球可以视为由一层层半径为 t 的 D 维超球在直径方向堆叠而成的几何体 [6]，以下求积分的过程类似于将洋葱切片后累积起来的过程

---

[3] https://en.wikipedia.org/wiki/Curse_of_dimensionality

[4] 课件 lecture6 pp.5

[5] https://en.wikipedia.org/wiki/N-sphere

[6] https://zhuanlan.zhihu.com/p/104715872

$$V_{D+1}(1)=\int_{-1}^{1}V_D(t)\mathrm{d}x_{D+1}=V_D(1)\int_{-1}^{1}t^D\mathrm{d}x_{D+1}$$
$$=V_D(1)\int_{-1}^{1}(1-x_{D+1}^2)^{D/2}\mathrm{d}x_{D+1}$$

令 $x_{D+1}=\sin\theta$ 得到

$$V_{D+1}(1)=V_D(1)\int_{-\pi/2}^{\pi/2}\cos^{D+1}\theta\mathrm{d}\theta$$

在数学分析中上述积分式是一个很重要的结论，利用分部积分法可以得到递推式

$$\begin{aligned}\int_{-\pi/2}^{\pi/2}\cos^{D+1}\theta\mathrm{d}\theta&=\int_{-\pi/2}^{\pi/2}\cos^{D}\theta\mathrm{d}\sin\theta\\&=\cos^{D}\theta\mathrm{d}\sin\theta\Big|_{-\pi/2}^{\pi/2}-\int_{-\pi/2}^{\pi/2}\sin\theta\mathrm{d}\cos^{D}\theta\\&=D\int_{-\pi/2}^{\pi/2}\cos^{D-1}\mathrm{d}\theta-D\int_{-\pi/2}^{\pi/2}\cos^{D+1}\mathrm{d}\theta\end{aligned}$$

记

$$I_D:=\int_{-\pi/2}^{\pi/2}\cos^{D}\theta\mathrm{d}\theta\Rightarrow I_D=\frac{D-1}{D}I_{D-2}$$

考虑

$$I_0=\int_{-\pi/2}^{\pi/2}\mathrm{d}\theta=\pi$$
$$I_1=\int_{-\pi/2}^{\pi/2}\cos\theta\mathrm{d}\theta=2$$

得到

$$I_{2D}=\pi\frac{(2D-1)!!}{(2D)!!},\ I_{2D+1}=2\frac{(2D)!!}{(2D+1)!!}$$

考虑边界条件

$$V_1(R)=2R\Rightarrow V_1(1)=2$$

考虑 $\Gamma(1/2)=\sqrt{\pi}$，从而利用 Gamma 函数和递推式 $V_{D+1}(1)=I_{D+1}V_D(1)$ 推得结论

$$V_D(1) = \frac{\pi^{D/2}}{\Gamma(D/2+1)}$$
$$S_{D-1}(1) = \frac{2\pi^{D/2}}{\Gamma(D/2)}$$

观察到取 $R=1$ 时当 $D \to +\infty$ 时 $V_D(R) \to 0$，当维数升高时，单位超球的体积和单位超立方体的体积之比以指数级的收敛于 0。这个结论告诉我们，当考虑在单位超立方体（体积为 1）均匀分布的数据时，距离中心为 1 的数据的比例趋近于 0，大部分数据分布于立方体的“角”上，角对应的数值即为边界值，这样的“角”在 D 维空间中共有 2^D 个，对应了 2^D 个长度为 D 的二进制码。有时采集到的数据维数可能比较高甚至大于数据量，因而此时每个数据有时会倾向于分散地分布在这样的一个个角（Corner）中[7]，这体现了高维数据具有稀疏性。

再考虑在 D 维单位超球内的均匀分布

$$p(\boldsymbol{x}) = \begin{cases} 1/V_D(1) & \|\boldsymbol{x}\|_2^2 \leqslant 1 \\ 0 & \text{otherwise} \end{cases}$$

计算超球上点和中心之间距离的期望，将单位超球视为一层层球壳

$$\begin{aligned} \mathrm{E}[\|\boldsymbol{x}\|_2] &= \int_0^1 r p(\boldsymbol{x}) \mathrm{d}V_D(r) = \int_0^1 r \frac{S_{D-1}(r)}{V_D(1)} \mathrm{d}r = D\int_0^1 r \frac{S_{D-1}(r)}{S_{D-1}(1)} \mathrm{d}r \\ &= D\int_0^1 r^D \mathrm{d}r = \frac{D}{D+1} \end{aligned}$$

当 $D \to +\infty$ 时，样本点到中心之间的期望趋于 1，上式直接表明在维度较高时绝大多数数据分布在超球的边缘。

以上分析均表明，在高维空间中，数据存在某种向边缘聚集的趋势。这对于 K-NN 分类器[8]来说是非常致命的，因为这表明随着半径的扩大，在邻域半径的微小变化可能决定了超球覆盖的样本点是一个也没有还是覆盖所有的样本点，此时样本之间的差异显著减小，噪声的影响显著提高，算法不稳定性上升了[7]。

更进一步地，在高维空间中从维度对应的协方差矩阵为单位矩阵的数据中采样，样本点之间的最小 Euclidean 距离 $d_m(D)$ 和最大的 Euclidean 距离 $d_M(D)$ 满足

[7] https://zhuanlan.zhihu.com/p/87577972

[8] 详见 11.2

$$\lim_{D\to\infty} \mathrm{E}[\frac{d_M(D)-d_m(D)}{d_m(D)}] = 0$$

这直接表明 Euclidean 距离实际上在高维空间的度量某种程度上是失效的，因为不同样本之间 Euclidean 距离的差异在高维空间中几乎消失了。

6.2　PCA 简介及拓展 [9]

在 Lasso 回归 [10] 中，监督信号（标签）$\boldsymbol{y}$ 辅助了 $\boldsymbol{X}$ 在 $\boldsymbol{w}$ 的作用下对特征维度进行了选择（$\boldsymbol{w}$ 实际上 mask 掉了部分在回归任务中不是特别重要的维度）。而 PCA 是考虑在没有监督信号 $\boldsymbol{w}$ 的帮助下该如何进行这一过程，这是典型的无监督学习的过程。

数据降维的第一种视角为最小化重构误差，有时也等价为写为最大化保存数据信息，从这个视角导出了线性降维中最重要的算法即主成分分析（Principal component analysis / PCA），在 PCA 中我们可以证明这两种方法是等价的；第二种视角为保证降维的等距性质，即降维后样本点之间距离是相近的，即样本之间的差异在降维前后保持基本不变。当数据降维使用的函数是线性的时候降维称为线性降维，与升维通常要求非线性不同，线性降维在理论和特定场合的实践中也能有很不错的发挥。

PCA 的求解的详细的理论推导见附录 D。在示意图 [11] 中，左图和右图都在寻找一个最大化方差的投影面。左图中二维数据（蓝色点表示）沿着投影方向投影至与灰色虚线垂直的投影面（投影后的数据点用红色点表示）；右图中三维数据沿着投影方向投向二维平面，图中两个红色箭头代表了三维数据的两个主成分，两个主成分张成了最佳的投影面。

由多元线性回归的知识，实际上 PCA 最小化降维后矩阵元素的平方和误差是基于数据的加性误差服从零均值的正态分布的假设，将正态分布更改为 Laplace 分布后对应的优化函数也随之更改，我们得到了 RPCA（Robust principal component analysis）。注意，这里矩阵的 L^1 和接下来要提到的范数指的是对矩阵的所有元素的绝对值求和。事实上，RPCA 的原问题是假定数据 $\boldsymbol{X}$ 可以被分解为低秩的和稀疏的两个部分

$$\boldsymbol{X} = \boldsymbol{L} + \boldsymbol{S},\ \mathrm{rank}(\boldsymbol{L}) \leqslant L,\ \|\boldsymbol{S}\|_0 \leqslant S$$

[9] 课件 lecture6 pp.8-9,11,15-17

[10] 详见 4.4

[11] 课件 lecture6 p.11

一种求解方法是通过凸松弛的思想，将 L^0 范数松弛为 L^1 范数[12]，将秩约束条件松弛为核范数（Nuclear norm）的约束。$\boldsymbol{L}$ 的核范数 $\|\boldsymbol{L}\|_*$ 是矩阵 $\boldsymbol{L}$ 的奇异值之和，由于奇异值是非负的，且矩阵的秩和非零的奇异值个数相同 [13]，因而矩阵的核范数，即奇异值之和就相当于此时矩阵的 L^1 范数，矩阵的秩就相当于矩阵的 L^0 范数，因而这种近似是合理的。

再有一种求解方法是通过矩阵的乘性分解，通过交替优化解决这类问题。问题可以拆分为若干个最小化 MAE 问题，而这种问题我们已经知道了两种 [14]：一种思路是转化为线性规划问题后利用单纯形法解决，还有种思路是借助 IRLS 通过再一次的交替优化解决。由于矩阵乘法事实上不会使得矩阵的秩上升，通过这种方式不仅得到了矩阵的低秩表达，还得到了使用更少的参数量对原矩阵进行存储（矩阵的有损压缩）的方法。

类似地，NMF（Non-negative matrix factorization）也考虑了矩阵进行乘性分解，拆分为两个非负的矩阵，并使得乘积和原矩阵之间的 F 范数尽可能接近。

课件中还提到了两种类似的降维方法。Subspace clustering 通过拓展 Lasso 回归来实现数据降维，得到了线性降维矩阵 $\boldsymbol{W}$。L^1 范数的约束使得 $\boldsymbol{W}$ 趋向稀疏，核范数的约束使得 $\boldsymbol{W}$ 趋向低秩。而 Dictionary learning 则进一步地将降维结果进行乘性分解，和上文中 NMF 思想类似，只不过多了一个约束 $\boldsymbol{A}$ 的 L^1 正则项使得 $\boldsymbol{A}$ 更加稀疏。

6.3　压缩感知简介 [15]

压缩感知（Compressive sensing）是 21 世纪以来数字信号处理领域的重要成果之一。我们在 D.2.4 提到了基于奇异值分解的图像压缩方法，事实上这种方法与传统的图像的有损压缩算法思想上有共通之处——我们实际上在借助谱的思想分析并提取信号的主要成分。以我们比较熟悉的 JPEG 格式为例，输入的图像经过离散余弦变换（Discrete cosine transform / DCT）或离散小波变换（Discrete wavelet transform / DWT）将图像从空域变换到相应的变换域，相应的变换域往往是稀疏的，存在很多值为 0 或相对地近似为 0 的成分，通过我们可以通过设置相应的阈值来对这些成分进行过滤，只

[12] 见 1.1 结尾部分

[13] 见 D.2.2 结尾部分

[14] 详见 4.6

[15] 课件 lecture6 pp.18-20

保留变换域中的系数较大的成分，在压缩的信息复原的过程中，仅需调用相应的逆变换，重新将变换域的信号转化为空域信号即可。

这种基于全采样并通过将信号转换到稀疏的变换域的思想给予了我们很大的启发，既然对信号全采样后还要在变换域上进行相应的舍弃，那么为什么不直接进行随机采样呢[16]？受这种思想启发压缩感知诞生了。

在示意图的左半部分，自然信号 $\boldsymbol{x}$ 在经过随机矩阵 $\boldsymbol{\Phi}$ 的采样下，得到随机采样的结果 $\boldsymbol{y}$。此时采样矩阵 $\boldsymbol{\Phi}$ 和采样结果 $\boldsymbol{y}$ 都已经被观测到了，我们现在要考虑将压缩后的数据复原。当随机采样矩阵 $\boldsymbol{\Phi}$ 对稀疏信号 $\boldsymbol{x_1}$ 和 $\boldsymbol{x_2}$ 具有有限等距的性质（Restricted isometry property / RIP）时，这实际上告诉了我们随机采样矩阵 $\boldsymbol{\Phi}$ 需要满足的原则，我们可以以高概率得到稀疏信号的复原。事实上在示意图的左侧我们看到自然信号 $\boldsymbol{x}$ 本身实际上并不是稀疏的，因而我们需要对 $\boldsymbol{x}$ 进行稀疏表示[16]（Sparse representation）。这个过程可以表示为

$$\boldsymbol{x} = \boldsymbol{\Psi\alpha},\ \|\boldsymbol{\alpha}\|_0 \leqslant S$$

我们需要一个稀疏基矩阵使得原始信号 $\boldsymbol{x}$ 能够进行稀疏表示。这个过程类似于 DCT 和 DWT 的逆变换，即将稀疏变换域中的稀疏信号 $\boldsymbol{\alpha}$ 变换为原始信号 $\boldsymbol{x}$，此时稀疏基是对应的逆变换的基。

在压缩感知中最小化稀疏系数 $\boldsymbol{\alpha}$ 的 L^0 范数和最小化 L^1 范数是等价的。以 L^1 范数代替 L^0 范数作为正则项约束，利用最小二乘思想定义设计重构误差，我们得到了示意图的右半部分

$$\boldsymbol{\alpha}^* = \arg\min_{\boldsymbol{\alpha}} \|\boldsymbol{y} - \boldsymbol{\Phi x}\|_2^2 + \|\boldsymbol{\alpha}\|_1 = \arg\min_{\boldsymbol{\alpha}} \|\boldsymbol{y} - \boldsymbol{\Phi\Psi\alpha}\|_2^2 + \|\boldsymbol{\alpha}\|_1$$

和重构信号

$$\hat{\boldsymbol{x}} = \boldsymbol{\Psi\alpha}^*$$

记传感矩阵（Sensing matrix）$\boldsymbol{\Theta} := \boldsymbol{\Phi\Psi}$，问题变为 Lasso 回归问题

$$\boldsymbol{\alpha}^* = \arg\min_{\boldsymbol{\alpha}} \|\boldsymbol{y} - \boldsymbol{\Theta\alpha}\|_2^2 + \|\boldsymbol{\alpha}\|_1$$

[16] https://zhuanlan.zhihu.com/p/22445302,

第七章　非线性降维与流形学习

7.1　流形与流形学习 [1]

对于地图的投影而言，线性的地图投影方式相当于直接利用平行光源将地球投影到投影屏中，或者理解为三视图的一个视图，这个投影事实上是一个正交投影[2]。这样得到地图很多时候是难以使用的，因为我们很难直观地利用地图获取球面的直观的信息，而要真正地绘制出实用的地图，我们常常选择非线性的投影方式，常用的投影方法为 Mercator 投影法（Mercator projection），其在地球球心处设置光源，其投影面为与地球相切于赤道的圆柱面，可以看到地图的两极被严重拉伸了，加拿大和俄罗斯领土面积在该投影法下在视觉上容易被夸大，因而对于某些国家使用该地图投影法可能隐含了一定的政治目的。在 Mercator 投影法下高纬度两点之间距离会被严重放大，因而这可能不算一个好的平面的嵌入方式。

回到这一讲中其中最重要的概念——流形（Manifold）。什么叫流形？流形定义为可局部 Euclidean 空间化的空间，具体一点的描述就是一个点到其相邻点之间的距离可以用 Euclidean 空间度量 [3]。流形是 Euclidean 空间的推广，不同于整体上均成线性的 Euclidean 空间，流形只在局部上是线性的，换而言之局部要保证是平的，在其局部可以用相对于原来的高维空间较低维 Euclidean 空间近似，如三维曲面的局部可以用二维平面近似；而从整体上看是非线性的，换而言之相对于刚直的坐标轴而言是弯曲的或是不平的。从这个角度看流形是一维的曲线和二维的曲面的推广。相邻的点之间可以用 Euclidean 距离度量，相距遥远的点就只能通过其他的样本点建立两点之间的路径，这样得到的距离称为测地线距离（Geodesic distance）即相邻近

[1] 课件 lecture7 pp.3,15

[2] 详见 D.1

[3] https://zh.wikipedia.org/wiki/%E6%B5%81%E5%BD%A2

的点可以直接用 Euclidean 距离度量而远的点则必须借助其他点建立联系[4]。经过采样后流形可以用一张图表示，数据点之间的连边表示相邻关系（通常来说这样确定的图是稀疏的），连边的两点的权重为点间的 Euclidean 距离，相邻的点之间距离可以直接由边权重给出而相邻较远的点之间的距离则可以采样最短路算法。讨论流形时我们常常以经典的流形地球表面为例，当我们考虑地球上相对于地球周长相距较近的两个点时，比如仅仅是测量一个操场的长度和宽度，我们可以将地面视为一个平面，而平面上的点显然具有线性的；但是当我们考虑测量较远的两个点时，比如测量北京到华盛顿之间的距离，地表就不能视为一个平面了。

机器学习中的流形学习（Manifold learning）的观点认为我们观测到的数据事实上是一个低维的嵌入高维空间的流形。在流形中，我们总可以找到一个比高维空间维数更低的坐标表示。这是因为流形对应的约束使得高维的坐标表示存在冗余（Reboundace）[5]，如在 3 维空间上地球表面上的点在忽略海拔的情况下我们只使用经度和纬度两个坐标就能表示，沿椭圆轨道的绕地球公转的月球我们只使用它转过的角度就能表示，这两个案例中前者的约束为点到球心的距离固定，后者对应的约束为轨道的方程。我们还可以这样理解，流形的局部可以建立到低维 Euclidean 空间的降维的映射，这样的映射可以推广至全局，从而可以建立使得流形上的点总能够从嵌入的高维空间降至低维的空间的映射。据此，如何找到一个非线性的映射，还原流形的低维本质是流形学习的目标。当我们找到这个非线性的映射时，我们恰好找到了一种从高维空间映射向低维空间的非线性的降维方法。

在流形的示意图[6]中，左图表示原来的二维的方形条带通过 S 形弯曲得到了一个三维空间中的流形，图中反映了流形经过均匀采样后生成的三维的 S 形图样。右图表示对原来的流形空间的复原。复原出的图样越接近方形，样本点越均匀，代表学习效果越好。这样构造的流形是比较简单的，因为流形是不封闭的，这使得我们可以找到一个有限的“平面”通过合适的代价最小的弯曲在“空间”中构造出相应的流形作为 ground truth。然而现实中样本形成的流形很多时候是封闭的，比如形成一个类似球面的流形，这种情况下流形是不存在边界的。我们也许可以转而将流形映射到球面上去，就像吹气球一样把封闭的流形“吹”成一个球面，对于这样的降维我们还会有其他的评价指标，如保角性和保距性等。

[4]周志华，机器学习，清华大学出版社，pp.234-235

[5]https://www.zhihu.com/question/24015486/answer/194284643

[6]课件 lecture7 p.15

7.2　流形学习的保距视角

测地线距离和流形上点之间的距离的保距性催生了流形学习中的 Classic MDS （Classic Multidimensional Scaling）。Classic MDS 的 motivation 在于找到一个数据在低维的嵌入，使得这个低维嵌入之间的 Euclidean 距离和高维空间的距离是接近的，该算法试图直接通过利用降维前后流形数据点之间的保距的性质进行流形学习。

7.2.1　从 stress 损失到 strain 损失 [7]

为了让作为低维嵌入的数据 $\boldsymbol{z_i}$ 和 $\boldsymbol{z_j}$ 的 L^p 距离（通常取 $p=1$ 或 $p=2$，这里取后者）和原先数据的距离 $d(\boldsymbol{x_i},\boldsymbol{x_j})$ 尽可能相近，我们设计的降维损失 stress 损失（Stress loss）

$$L_{stress}(\boldsymbol{Z})=\left(\sum_{n\neq m}^{N}(d_{nm}-\|\boldsymbol{z_n}-\boldsymbol{z_m}\|_p\right)^{1/2}$$

如果我们定义在低维空间中 $\boldsymbol{z_i}$ 和 $\boldsymbol{z_j}$ 两两之间的内积矩阵 $\boldsymbol{G}$，这个矩阵有些类似于之前提到的 Gram 矩阵，虽然事实上 $\{\boldsymbol{z_i}\}_{i=1}^N$ 并不一定为低维空间的一组基，因为低维空间的秩有可能是小于 N 的，但这不妨碍 $\boldsymbol{G}$ 控制低维空间的内积运算

$$\boldsymbol{G}:=\{\boldsymbol{z_n^T z_m}\}_{n,m=1}^N=\boldsymbol{Z^T Z}$$

因而我们得到了

$$g_{ij}=\langle\boldsymbol{z_i},\boldsymbol{z_j}\rangle\approx-\frac{1}{2}(d_{ij}^2-\|\boldsymbol{z_i}\|_2^2-\|\boldsymbol{z_j}\|_2^2)=-\frac{1}{2}(d_{ij}^2-g_{ii}-g_{jj})$$

为了简化问题，我们常常需要做出 $\boldsymbol{Z}$ 零均值的假设

$$\boldsymbol{Z1_N}=\boldsymbol{0_N}$$

这样的零均值假设帮助我们简洁地计算出边缘矩阵的每一行和列的均方和以及总体的均方和 [8]

[7] 课件 lecture7 pp.8-9

[8] www.stat.yale.edu/~lc436/papers/JCGS-mds.pdf

$$\begin{aligned}\bar{d}_{i\cdot}^2 &= \frac{1}{N}\sum_{j=1}^{N} d_{ij}^2 = \|\boldsymbol{z_i}\|_2^2 + \frac{1}{N}\sum_{j=1}^{N}\|\boldsymbol{z_j}\|_2^2 - \frac{2}{N}\langle \boldsymbol{z_i}, \boldsymbol{Z}\mathbf{1_N}\rangle \\ &= \|\boldsymbol{z_i}\|_2^2 + \frac{1}{N}\sum_{j=1}^{N}\|\boldsymbol{z_j}\|_2^2 = \|\boldsymbol{z_i}\|_2^2 + \frac{1}{N}\mathrm{tr}(\boldsymbol{G})\end{aligned}$$

$$\begin{aligned}\bar{d}_{\cdot j}^2 &= \frac{1}{N}\sum_{i=1}^{N} d_{ij}^2 = \|\boldsymbol{z_j}\|_2^2 + \frac{1}{N}\sum_{i=1}^{N}\|\boldsymbol{z_i}\|_2^2 - \frac{2}{N}\langle \boldsymbol{Z}\mathbf{1_N}, \boldsymbol{z_j}\rangle \\ &= \|\boldsymbol{z_j}\|_2^2 + \frac{1}{N}\sum_{i=1}^{N}\|\boldsymbol{z_i}\|_2^2 = \|\boldsymbol{z_j}\|_2^2 + \frac{1}{N}\mathrm{tr}(\boldsymbol{G})\end{aligned}$$

$$\bar{d}_{\cdot\cdot}^2 = \frac{1}{N^2}\sum_{i=1}^{N}\sum_{j=1}^{N} d_{ij}^2 = \frac{1}{N}\sum_{i=1}^{N}\bar{d}_{i\cdot}^2 = \frac{2}{N}\mathrm{tr}(\boldsymbol{G})$$

因而可以计算出 $\boldsymbol{G}$ 在对角线上的元素

$$\begin{aligned}g_{nn} = \|\boldsymbol{z_n}\|_2^2 &= \bar{d}_{n\cdot}^2 - \frac{1}{N}\mathrm{tr}(\boldsymbol{G}) = \bar{d}_{n\cdot}^2 - \frac{1}{2}\bar{d}_{\cdot\cdot}^2 \\ &= \bar{d}_{\cdot n}^2 - \frac{1}{N}\mathrm{tr}(\boldsymbol{G}) = \bar{d}_{\cdot n}^2 - \frac{1}{2}\bar{d}_{\cdot\cdot}^2\end{aligned}$$

和 $\boldsymbol{G}$ 其他的元素

$$g_{ij} = -\frac{1}{2}(d_{ij}^2 - g_{ii} - g_{jj}) = -\frac{1}{2}(d_{ij}^2 - \bar{d}_{i\cdot}^2 - \bar{d}_{\cdot j}^2 + \bar{d}_{\cdot\cdot}^2)$$

在行和列对应的均方和置为零后有

$$g_{ij} = -\frac{1}{2}d_{ij}^2$$

对于一个 $M \times N$ 的矩阵 $\boldsymbol{X}$ 而言，假设我们需要对列向量去均值化，我们通过将矩阵的每一列同时加上一个相同的偏置 $\boldsymbol{b}$ 使得矩阵的列向量的和（对矩阵的每一行求和）为零，那么我们会得到列去均值后的矩阵 $\tilde{\boldsymbol{X}}$

$$\tilde{\boldsymbol{X}} = \boldsymbol{X} - \boldsymbol{b}\mathbf{1}_{\boldsymbol{N}}^{\boldsymbol{T}}$$

对 $\tilde{\boldsymbol{X}}$ 的列向量求和得到

$$\tilde{\boldsymbol{X}}\mathbf{1_N} = (\boldsymbol{X} - \boldsymbol{b}\mathbf{1}_{\boldsymbol{N}}^{\boldsymbol{T}})\mathbf{1_N} = \boldsymbol{X}\mathbf{1_N} - N\boldsymbol{b} = \mathbf{0}_{\boldsymbol{M}} \Rightarrow \boldsymbol{b} = \frac{\boldsymbol{X}\mathbf{1_N}}{N}$$

于是我们得到

$$\tilde{\boldsymbol{X}} = \boldsymbol{X} - \frac{\boldsymbol{X}\mathbf{1}_{\boldsymbol{N}}\mathbf{1}_{\boldsymbol{N}}^{\boldsymbol{T}}}{N} = \boldsymbol{X}\left(\boldsymbol{I}_{\boldsymbol{N}} - \frac{1}{N}\mathbf{1}_{\boldsymbol{N}\times\boldsymbol{N}}\right)$$

记去中心化矩阵（Centering matrix）

$$\boldsymbol{C} := \boldsymbol{I}_{\boldsymbol{N}} - \frac{1}{N}\mathbf{1}_{\boldsymbol{N}\times\boldsymbol{N}}$$

得到对所有列向量通过 shiftting 去均值的中心化矩阵 $\boldsymbol{C}$。对于行来说是类似的，只不过从矩阵右乘变为矩阵左乘，且大小参数从 N 改为 M。因而行列去中心化均可通过矩阵 $\boldsymbol{C}$ 实现，并记距离平方矩阵行列去中心化的结果为

$$\boldsymbol{K} := -\frac{1}{2}\boldsymbol{C}(\boldsymbol{D}\odot\boldsymbol{D})\boldsymbol{C}$$

其中 $-1/2$ 推导中出现的系数。这样的去均值操作使得

$$g_{ij} = \boldsymbol{z}_{\boldsymbol{i}}^{\boldsymbol{T}}\boldsymbol{z}_{\boldsymbol{j}} \approx k_{ij}$$

因而最终我们可以设计 strain 损失（Strain loss）使得 g_{ij} 更加接近 k_{ij}

$$L_{strain}(\boldsymbol{Z}) = \left(\frac{\sum\limits_{n,m=1}^{N}(k_{nm} - \boldsymbol{z_n}^{T}\boldsymbol{z_m})^2}{\sum\limits_{n,m=1}^{N}k_{nm}^2}\right)^{1/2}$$

7.2.2　Classic MDS 的低维嵌入计算 [9]

在 Classic MDS 中我们使用距离度量为 Euclidean 距离，即

$$d_{ij} = \|\boldsymbol{x_i} - \boldsymbol{x_j}\|_2$$

平方后得到

$$d_{ij}^2 = \|\boldsymbol{x_i} - \boldsymbol{x_j}\|_2^2 = \|\boldsymbol{x_i}\|_2^2 + \|\boldsymbol{x_j}\|_2^2 - 2\langle\boldsymbol{x_i}, \boldsymbol{x_j}\rangle$$

记

$$\boldsymbol{L} = (\boldsymbol{X}\odot\boldsymbol{X})\mathbf{1}_{\boldsymbol{D}}\mathbf{1}_{\boldsymbol{N}}^{\boldsymbol{T}}$$

[9] 课件 lecture7 p.9

其第 i 行元素的 N 个元素均为 $\|\boldsymbol{x_i}\|_2^2$

$$\boldsymbol{D}\odot\boldsymbol{D}=\boldsymbol{L}+\boldsymbol{L^T}-2\boldsymbol{X^TX}$$

由于矩阵 $\boldsymbol{L}$ 的每一列实际上是一样的，以对每个列向量 shifting 的方式对列去均值后将得到 $\boldsymbol{0_{D\times D}}$，这很容易由以下式子得出

$$\begin{aligned}\boldsymbol{LC}&=(\boldsymbol{X}\odot\boldsymbol{X})\boldsymbol{1_D1_N^T}\left(\boldsymbol{I_N}-\frac{1}{N}\boldsymbol{1_{N\times N}}\right)=(\boldsymbol{X}\odot\boldsymbol{X})\boldsymbol{1_D1_N^T}\left(\boldsymbol{I_N}-\frac{1}{N}\boldsymbol{1_N1_N^T}\right)\\&=(\boldsymbol{X}\odot\boldsymbol{X})\left(\boldsymbol{1_D1_N^T}-\boldsymbol{1_D1_N^T}\right)=\boldsymbol{0_{D\times D}}\end{aligned}$$

同理得到

$$\boldsymbol{CL^T}=(\boldsymbol{LC})^T=\boldsymbol{0_{D\times D}}$$

从而

$$\begin{aligned}\boldsymbol{K}&=-\frac{1}{2}\boldsymbol{C}(\boldsymbol{D}\odot\boldsymbol{D})\boldsymbol{C}=-\frac{1}{2}(\boldsymbol{CLC}+\boldsymbol{CL^TC}-2\boldsymbol{CX^TXC})\\&=\boldsymbol{CX^TXC}=\boldsymbol{\tilde{X}^T\tilde{X}}\end{aligned}$$

这里实际上也证明了 $\boldsymbol{K}$ 的半正定性。我们定义优化目标为最小化 strain 损失，并定义秩约束

$$\begin{aligned}\min_{\boldsymbol{Z}}L_{strain}(\boldsymbol{Z})&=\min_{\boldsymbol{Z}}\sum_{n,m=1}^{N}(k_{nm}-\boldsymbol{z_n}^T\boldsymbol{z_m})^2\\&=\min_{\boldsymbol{Z}}\left\|\boldsymbol{K}-\boldsymbol{Z^TZ}\right\|_F^2,\ s.t.\ \operatorname{rank}(\boldsymbol{Z})\leqslant L\end{aligned}$$

考虑内积矩阵，我们需要保证 $\boldsymbol{G}$ 的对称性和半正定性[10]

$$\boldsymbol{G}:=\{\boldsymbol{z_n}^T\boldsymbol{z_m}\}_{n,m=1}^N=\boldsymbol{Z^TZ},\ \operatorname{rank}(\boldsymbol{G})=\operatorname{rank}(\boldsymbol{Z})$$

考虑对半正定的 $\boldsymbol{G}$ 进行特征值分解

$$\boldsymbol{G}=\boldsymbol{V\Lambda V^T}$$

得到 $\boldsymbol{Z}$ 的一种取值为

[10]秩等式的证明见 D.1.1 中间部分

$$\boldsymbol{Z} = \boldsymbol{\Lambda}^{\boldsymbol{1/2}}\boldsymbol{V}^{\boldsymbol{T}}$$

优化问题实际上已经变为了 PCA 对应的优化问题

$$\min_{\boldsymbol{G}} \|\boldsymbol{K} - \boldsymbol{G}\|_F^2,\ s.t.\ \operatorname{rank}(\boldsymbol{G}) \leqslant L,\ \boldsymbol{G}^{\boldsymbol{T}} = \boldsymbol{G},\ \boldsymbol{x}^{\boldsymbol{T}}\boldsymbol{G}\boldsymbol{x} \geqslant 0$$

上述优化问题我们实际已经求得了其闭式解 [11]，考虑 $\boldsymbol{K}$ 的特征值分解

$$\boldsymbol{K} = \boldsymbol{V}\boldsymbol{\Lambda}\boldsymbol{V}^{\boldsymbol{T}},\ \boldsymbol{V}^{\boldsymbol{T}}\boldsymbol{V} = \boldsymbol{I}_{\boldsymbol{N}}$$

得到

$$\boldsymbol{G} = \boldsymbol{V}_{\boldsymbol{L}}\boldsymbol{\Lambda}_{\boldsymbol{L}}\boldsymbol{V}_{\boldsymbol{L}}^{\boldsymbol{T}}$$

这里放宽了对称性和正定性的约束，由于 $\boldsymbol{K}$ 的对称性和半正定性实际上已经得到了满足，$\boldsymbol{G}$ 的对称性和半正定性也得到了满足，因而我们松弛后解得的解就是最优解。我们实际上已经得到了 $\boldsymbol{G}$ 的特征值分解表达式，这个表达式可以用于求解 $\boldsymbol{Z}$

$$\boldsymbol{Z} = \boldsymbol{\Lambda}^{\boldsymbol{1/2}}\boldsymbol{V}_{\boldsymbol{L}}^{\boldsymbol{T}}$$

Classic MDS 在数据空间使用的 Euclidean 距离很多情况下效果实际上发挥很有限，一方面和流形上两个之间的距离的定义有冲突，另一方面由于维度诅咒高维空间中 Euclidean 距离会有失效的问题 [12]。

7.2.3 ISOMAP [13]

为了解决 Classic MDS 未能解决的两个问题，ISOMAP 在 Classic MDS 的基础上将 Euclidean 距离替换为了测地线距离，这个距离定义为图中两个节点的最短路径。选择 Euclidean 距离为距离度量，在以数据点为节点通过 K-NN 算法确定邻接关系，通过距离度量确定边的权重的加权有向图中两个样本点之间的最短路径长度，点对之间的近似的测地线距离可以很轻易地通过 Dijkstra 或 Floyd 算法给出。

[11] 详见 D.3.2 结尾部分

[12] 详见 6.1.2

[13] 课件 lecture7 pp.10-11

示意图[14]对应的流形为“瑞士卷”，将真实的测地线距离用蓝线表示（这个图的连接关系由生成流形的方式给出，代表了原来的流形空间中样本点的真实的局部联系），将使用 K-NN（即考虑图时对于每个节点只考虑和最近的 K 个节点之间的连边）计算得到的近似的测地线距离用红线表示。B 和 C 图显示了 K-NN 稀疏的连接关系，图中存在很多小的白色的孔洞，其对应部分没有边的连接，红线绕过图中白色的空洞将两个数据点相连。如此展开图中的流形可以看到真实的测地线距离和近似的测地线距离的差异。其中选择 K-NN 的合理性是流形本身局部线性的特质赋予的。

这样定义的测地线距离有一个问题，矩阵

$$\boldsymbol{K} = -\frac{1}{2}\boldsymbol{C}(\boldsymbol{D} \odot \boldsymbol{D})\boldsymbol{C}$$

的对称性显然是满足的，但是半正定性可能不满足，此时我们使用添加数量矩阵 $k\boldsymbol{I}$ 作为扰动项的方法，强迫 $\boldsymbol{K}$ 满足正定性从而解决这个问题[15]。

7.3　流形学习的局部线性视角

流形降维前后局部线性的性质的保持催生了流形学习中的 LLE（Local Linear Embedding），该性质允许我们利用一个个的平面，即低维的 Euclidean 空间，对流形进行局部近似。LLE 的 motivation 在于通过确定数据点周围的点对数据的最佳的线性表出的系数，再将这种表出的关系迁移至低维嵌入上，利用降维前后流形局部线性关系保持基本不变进行流形学习。

7.3.1　LLE 的线性系数计算[16]

在线性系数计算中我们选取了样本点 $\boldsymbol{x_i}$ 的最近的 K 个样本点并找到了这些样本点对 $\boldsymbol{x_i}$ 的最佳的线性表示，其线性表示系数的权重经过了归一化，优化目标变为

[14]课件 lecture7 p.11

[15]见 D.1.4 中间部分

[16]课件 lecture7 p.15

$$\boldsymbol{W}^* := \arg\min_{\boldsymbol{W}} \sum_{i=1}^{N} \left\| \boldsymbol{x_i} - \sum_{j \in \mathcal{N}_K(\boldsymbol{x_i})} w_{ij} \boldsymbol{x_j} \right\|_2^2,$$
$$s.t. \sum_{j=1}^{K} w_{ij} = \sum_{j=1}^{N} w_{ij} = 1 \Rightarrow \boldsymbol{W}\boldsymbol{1_N} = \boldsymbol{1_N}$$

定义关于样本点 $\boldsymbol{x_i}$ K-NN 矩阵 $\boldsymbol{N_i}$ 和对应的权值向量 $\boldsymbol{\omega_i}$

$$\boldsymbol{N_i} := (\boldsymbol{x_j})_{j \in \mathcal{N}_K(\boldsymbol{x_i})} \in \mathbb{R}^{D \times K}$$
$$\boldsymbol{\omega_i} := (w_{ij})^T_{j \in \mathcal{N}_K(\boldsymbol{x_n})} \in \mathbb{R}^{K \times 1}$$

我们的优化目标为

$$\boldsymbol{\omega_i} := \arg\min_{\boldsymbol{\omega}} \|\boldsymbol{x_i} - \boldsymbol{N_i}\boldsymbol{\omega}\|_2^2, \ s.t.\ \boldsymbol{1_K^T}\boldsymbol{\omega} = 1, i = 1, 2, \cdots, N$$

优化目标实际上是可以化简的，由我们权值为 1 的约束条件得到

$$\boldsymbol{\omega_i} = \arg\min_{\boldsymbol{\omega}} \|\boldsymbol{x_i}1 - \boldsymbol{N_i}\boldsymbol{\omega}\|_2^2 = \arg\min_{\boldsymbol{\omega}} \left\|(\boldsymbol{x_i}\boldsymbol{1_K^T} - \boldsymbol{N_i})\boldsymbol{\omega}\right\|_2^2, \ s.t.\ \boldsymbol{1_K^T}\boldsymbol{\omega} = 1$$

令 $\boldsymbol{Y} := \boldsymbol{x_i}\boldsymbol{1_K^T} - \boldsymbol{N_i}$ 我们得到了

$$\boldsymbol{\omega_i} = \arg\min_{\boldsymbol{\omega}} \|\boldsymbol{Y}\boldsymbol{\omega}\|_2^2, \ s.t.\ \boldsymbol{1_K^T}\boldsymbol{\omega} = 1$$

采用 Lagrange 乘数法

$$L(\boldsymbol{\omega}, \lambda) := \frac{1}{2}\|\boldsymbol{Y}\boldsymbol{\omega}\|_2^2 + \lambda(1 - \boldsymbol{1_K^T}\boldsymbol{\omega})$$

求偏导得到

$$\frac{\partial L}{\partial \boldsymbol{\omega}} = \boldsymbol{Y^T}\boldsymbol{Y}\boldsymbol{\omega} - \lambda \boldsymbol{1_K}$$

假设向量 $\boldsymbol{x_i}$ 和其 K 个相邻的点都是线性无关的[17]

$$\mathrm{rank}(\boldsymbol{N_i}) = \mathrm{rank}(\boldsymbol{N_i}^T\boldsymbol{N_i}) = K$$

我们实际上可以得到 $\boldsymbol{Y}$ 的列向量事实上是线性无关的，否则

[17]秩的关系的证明见 D.1.1

$$\exists \lambda_i \neq 0,\ i = 1, 2, \cdots, K,\ \sum_{j=1}^{K} \lambda_j (\boldsymbol{x_i} - \boldsymbol{n_j}) = 0 \Rightarrow \boldsymbol{x_i} \sum_{j=1}^{K} \lambda_j - \sum_{j=1}^{K} \lambda_j \boldsymbol{n_j} = 0$$

这是矛盾的，因为这表明 $\boldsymbol{x_i}$ 和其 K 个相邻的点是线性相关的，因而我们得到了[10]

$$\text{rank}(\boldsymbol{Y}) = \text{rank}(\boldsymbol{Y}^T \boldsymbol{Y}) = K$$

如果我们很不幸地得到了 $\boldsymbol{Y}$ 实际并不是列满秩的，我们可以通过加一个小的扰动项来保证求逆运算的稳定性[15]，令 Lagrangian 函数偏导为 0 得到

$$\boldsymbol{\omega}^* = \lambda (\boldsymbol{Y^T Y})^{-1} \boldsymbol{1_K}$$

令

$$\boldsymbol{C} := \boldsymbol{Y^T Y} = (\boldsymbol{N_i} - \boldsymbol{x_i} \boldsymbol{1_K^T})^T (\boldsymbol{N_i} - \boldsymbol{x_i} \boldsymbol{1_K^T})$$

利用约束条件得到

$$\boldsymbol{1_K^T \omega^*} = \lambda \boldsymbol{1_K^T C^{-1} 1_K} = 1 \Rightarrow \boldsymbol{\omega^*} = \frac{\boldsymbol{C^{-1} 1_K}}{\boldsymbol{1_K^T C^{-1} 1_K}}$$

最优参数相当于对 $\boldsymbol{C^{-1} 1_K}$ 进行求和归一化。

利用权值向量 $\boldsymbol{\omega_i}$ 的权值去填充稀疏的权值矩阵 $\boldsymbol{W}$ 对应位置，当计算完所有样本点的权值向量，$\boldsymbol{W}$ 已经填充完了后，线性系数的计算结束。这个计算各个点的最佳线性表示的过程由于不会相互影响事实上是可以做并行处理的，这将大大加速计算效率。

7.3.2　LLE 的低维嵌入计算 [18]

在低维嵌入计算中我们通过得到的稀疏的权值矩阵来求解最合适的 $\boldsymbol{z_i}$ 的低维（L 维）嵌入，保持低维空间和原来的样本空间的局部性质。为了避免平凡解，即取所有 $\boldsymbol{z_i}$ 均为 $\boldsymbol{0}$，我们需要设计约束条件，这里 $\boldsymbol{Z}$ 是由 N 个 L 维列向量排列得到的矩阵

[18] 课件 lecture7 p.15

$$\min_{\boldsymbol{Z}} \sum_{i=1}^{N} \left\| \boldsymbol{z_i} - \sum_{j=1}^{N} w_{ij} \boldsymbol{z_j} \right\|_2^2 = \min_{\boldsymbol{Z}} \sum_{i=1}^{N} \left\| \boldsymbol{z_i} - \boldsymbol{Z} \boldsymbol{W_i^T} \right\|_2^2 = \min_{\boldsymbol{Z}} \left\| \boldsymbol{Z} - \boldsymbol{Z} \boldsymbol{W^T} \right\|_F^2$$
$$= \min_{\boldsymbol{Z}} \left\| \boldsymbol{Z^T} - \boldsymbol{W} \boldsymbol{Z^T} \right\|_F^2, \ s.t. \ \boldsymbol{Z} \boldsymbol{Z^T} = \boldsymbol{I_L}, \ \boldsymbol{Z} \boldsymbol{1_N} = \boldsymbol{0_N}$$

由于权值和为 1 的限制所有 $\boldsymbol{z_i}$ 的平移不会影响目标函数的值，这一点是容易验证的，考虑

$$\sum_{j=1}^{N} w_{ij} = 1$$

得到

$$\forall \boldsymbol{b} \in \mathbb{R}^{D \times 1}, \ \min_{\boldsymbol{Z}} \sum_{i=1}^{N} \left\| (\boldsymbol{z_i} + \boldsymbol{b}) - \sum_{i=1}^{N} w_{ij} (\boldsymbol{z_i} + \boldsymbol{b}) \right\|_2^2 = \min_{\boldsymbol{Z}} \sum_{i=1}^{N} \left\| \boldsymbol{z_i} - \sum_{i=1}^{N} w_{ij} \boldsymbol{z_i} \right\|_2^2$$

这个约束对我们避免平凡解没有帮助。在理论推导中，这个约束通常会被舍弃。令

$$\boldsymbol{\Phi} := (\boldsymbol{I} - \boldsymbol{W})^T (\boldsymbol{I} - \boldsymbol{W})$$

我们得到优化问题为

$$\min_{\boldsymbol{Z}} \operatorname{tr}(\boldsymbol{Z} \boldsymbol{\Phi} \boldsymbol{Z^T}), \ s.t. \ \boldsymbol{Z} \boldsymbol{Z^T} = \boldsymbol{I_D}$$

上述优化问题我们实际已经求得了其闭式解 [11]，只需考虑对称的半正定矩阵 $\boldsymbol{\Phi}$ 的特征值分解

$$\boldsymbol{\Phi} = \boldsymbol{V} \boldsymbol{\Lambda} \boldsymbol{V^T}, \ \boldsymbol{V^T} \boldsymbol{V} = \boldsymbol{I_N}$$

考虑矩阵 $\boldsymbol{\Phi}$ 的后 L 个特征值，注意此时记对应的特征值对角矩阵和特征向量为 $\boldsymbol{\Lambda_L}$ 和 $\boldsymbol{V_L}$，得到

$$\boldsymbol{Z^*} = \boldsymbol{V_L^T}$$

此外我们还观察到 $\boldsymbol{\Phi}$ 实际上具有一个平凡的特征值和特征向量，这是由我们给出的权值和为 1 决定的，考虑以下式子

$$\boldsymbol{\Phi} \boldsymbol{1_N} = (\boldsymbol{I} - \boldsymbol{W})^T (\boldsymbol{I} - \boldsymbol{W}) \boldsymbol{1_N} = (\boldsymbol{I} - \boldsymbol{W})^T (\boldsymbol{1_N} - \boldsymbol{W} \boldsymbol{1_N}) = 0 \boldsymbol{1_N}$$

这个特征值和特征向量实际上和我们给定的数据是无关的，如果考虑了这样的一个特征值和特征向量，低维嵌入中一个维度的值将失去特异性（而且 0 作为所有低维嵌入中某一维度唯一的值本来就是冗余的），因而我们通常转而考虑后 $L+1$ 个特征值和对应的特征向量并将最小的一个特征值和特征向量舍弃 [19]。

7.4　其他非线性降维方法 [20]

KPCA（Kernel principal component analysis）的想法非常简单，就是利用核函数将数据 ϕ 转化为表征能力更强的高维特征，在高维空间上再对特征进行处理以得到比低维空间更好的结果，实际上就是一个先升维再降维的过程，可以理解为先将低维数据在高维特征空间中弯曲成为高维特征空间中的嵌入的流形后将这个流形拍扁至低维空间。在某些情况下，这种高维空间的弯曲赋予了数据好的性质，可以使得原有的非线性的数据具有一定的线性。KPCA 和 Classic MDS 和 ISOMAP 的中间的优化问题之间是存在联系的，其核矩阵 $\boldsymbol{K}$ 在问题中被替换为了

$$\boldsymbol{K}=-\frac{1}{2}\boldsymbol{C}(\boldsymbol{D}\odot\boldsymbol{D})\boldsymbol{C}$$

在 t-SNE（T-distributed Stochastic Neighbor Embedding）中我们实际上定义了高维空间和低维空间不同的相似度度量，我们在高维空间上使用零均值正态分布定义样本之间的距离，这实际上是以 $2\sigma^2$ 为带宽的 RBF 核定义的核距离。在低维空间上，距离度量被替换为了自由度为 1 的 t 分布（或者说是 Cauchy 分布）。注意高维空间存在 Euclidean 距离失效的问题 [12]，Euclidean 空间度量可能会失效，因而更先进的手段会被用于改造高维空间的距离度量。这里我们运用核函数定义相似度的想法在很多地方还会提到，产生这种想法是因为数据点的特征在特征空间的内积某种程度上反映了数据点之间的距离。最后高维和低维相似度的分布之间的距离我们使用 KL 散度来度量，这是一种非常常见的度量分布之间互信息的“距离度量”，之所以打引号是因为这种距离度量实际上是一种不符合距离度量性质的统计距离[21]。

[19] 我们在 E.3 中也会遇到类似的情况

[20] 课件 lecture7 pp.16-26

[21] 详见 F.1.3 结尾部分

Auto-encoder 的思想是最小化数据在经过 encoder 映射至隐空间后通过 decoder 还原产生的损失，数据在隐空间的分布由我们提供的隐变量的先验对应的正则项约束。我们的目的在于通过最大化保留数据原本的信息的方式从高维的数据空间降至低维的隐空间，从而达到数据降维的目的，这个想法和 PCA 是类似的，只不过 PCA 在优化函数上没有将 encoder 和 decoder 显式地定义出来。

第四部分

密度估计与聚类分析

第八章　传统聚类方法

8.1　K-Means 聚类 [1]

K-Means 聚类（K-Means clustering）的 motivation 在于其认为属于某一类的样本点其到对应类的聚类中心的距离要远于到其他类的聚类中心的距离，基于这种知识数据将归属于距离其最近的聚类中心对应的类。我们的优化目标为寻找 k 个聚类中心使得所有样本点到这些聚类中心的距离之和最小

$$\min_{\boldsymbol{C}} \sum_{i=1}^{k} \sum_{\boldsymbol{x} \in \mathcal{C}_i} d(\boldsymbol{x}, \boldsymbol{c_i})$$

以上式子中数据点的类别变量很难显式的求解（其作为优化中的隐变量），因而我们采取交替更新的方式，先确定初始化聚类中心，将样本归属于距离其最近的聚类中心对应的类，从而将表达式转化为一个便于求解的形式

$$k = \arg \min_{i \in \{1,2,\cdots,K\}} d(\boldsymbol{x}, \boldsymbol{c_i}) \Rightarrow \boldsymbol{x} \in \mathcal{C}_k$$

取距离度量为 Euclidean 距离，为了方便求解我们采用最小二乘法，优化目标变为

$$\min_{\boldsymbol{C}} \sum_{i=1}^{k} \sum_{\boldsymbol{x} \in \mathcal{C}_i} d(\boldsymbol{x}, \boldsymbol{c_i})^2 = \min_{\boldsymbol{C}} \sum_{i=1}^{k} \sum_{\boldsymbol{x} \in \mathcal{C}_i} \|\boldsymbol{x} - \boldsymbol{c}\|_2^2$$

我们实际上只需求解

$$\boldsymbol{c_i} = \arg \min_{\boldsymbol{c}} \sum_{\boldsymbol{x} \in \mathcal{C}_i} \|\boldsymbol{x} - \boldsymbol{c}\|_2^2$$

求导得到

[1] 课件 lecture8 pp.4-6

$$\frac{\partial}{\partial \boldsymbol{c}} \sum_{\boldsymbol{x} \in \mathcal{C}_i} \|\boldsymbol{x}-\boldsymbol{c}\|_2^2 = \sum_{\boldsymbol{x} \in \mathcal{C}_i} \frac{\partial(\boldsymbol{x}-\boldsymbol{c})}{\partial \boldsymbol{c}} \frac{\partial \|\boldsymbol{x}-\boldsymbol{c}\|_2^2}{\partial(\boldsymbol{x}-\boldsymbol{c})} = -2 \sum_{\boldsymbol{x} \in \mathcal{C}_i} (\boldsymbol{x}-\boldsymbol{c})$$

令导数为 0

$$\boldsymbol{c_i} = \frac{1}{|\mathcal{C}_i|} \sum_{\boldsymbol{x} \in \mathcal{C}_i} \boldsymbol{x}$$

即使用该类对应的样本点的重心更新聚类中心。重复这两个交替的过程直至聚类中心收敛。

在 K-Means 的每一步迭代总能够使得优化函数的值下降，所以 K-Means 一直向着优化目标的方向前进，由于单调有限必有极限，我们的目标函数的值必然可以收敛。但是我们实际上可以证明样本点的类别在某些极其特殊情况下是不收敛的，这是由于 K-Means 内部实现时对于距离相等的聚类中心的归属的判断可能具有随机性 [2]。

K-Means 采用 Euclidean 距离在高维可能出现维度诅咒 [3] 的问题，而且对于线性不可分的数据表现比较无力。我们也许可以尝试其他的距离度量，此时聚类中心的更新公式需要进行相应的改写。当然我们也可以先对数据进行非线性的降维后再进行聚类，这就是谱聚类（Spectral clustering）的想法。谱聚类相关的理论推导见附录 E。

8.2　聚类的评价指标 [4]

一个常见的聚类的评价指标为纯度（Purity）。纯度实际上选取与聚类得到的类重合最大的真实类作为响应的真实类别，并取重合部分的元素个数计数作为响应的值，将 k 个类的响应值求和并除以元素个数得到纯度。这实际上会带来一个问题，当聚类的个数大于真实类别的个数时，甚至考虑极端的情况，选择 $k = N$ 作为聚类的参数，这样会导致虽然纯度很高，但是聚类得到的类别数量太多，类别过于破碎，实际上聚类的效果并不算好。

评价指标的出发点有不少时从缩小类内间距、放大类间间距的角度出发的，这也是切图聚类（Graph-cut clustering）的出发点[5]。还有一些指标结合

[2] https://stats.stackexchange.com/questions/188087/proof-of-convergence-of-k-means

[3] 详见 6.1.2

[4] 课件 lecture8 pp.12-19

[5] 详见 E.3

了信息论中熵和信息的知识，如 NMI（Normalized mutual information）[6]。有关聚类的评价指标课件已经描述得很详细了，此处不再赘述。

[6]详见 F.1.3 中间部分

第九章　聚类的参数方法与 EM 算法

9.1　生成模型与鉴别模型 [1]

这里生成模型和鉴别模型是对数据 $\boldsymbol{x}$ 分布的建模而言的。对于生成模型（Generative model），建模出发点是探索的生成方式，考虑的是对数据 $\boldsymbol{x}$ 或数据 $\boldsymbol{x}$ 和标签 y 的联合分布进行建模；对于鉴别模型（Discriminative model），建模的出发点是探索数据 $\boldsymbol{x}$ 之间的差异，考虑的是对标签在给定数据 $\boldsymbol{x}$ 的条件下得到的条件分布。一个很形象的比喻是假如你有一个通过输入一个人声音信号判断一个人性别的二分类任务，如果你是一个鉴别模型你会较为直接通过输入的声音来得到判断性别，但是如果你是生成模型你会先使用这一段声音信号重构出一个人的全貌，再通过这个全貌来判断一个人的性别。

我们之前讨论的线性回归和非线性回归模型[2]都是鉴别模型，因为在建模过程中没有用到对数据 $\boldsymbol{x}$ 分布进行建模而考虑标签 y 在给定数据 $\boldsymbol{x}$ 情况下的条件分布；而线性降维模型和非线性降维模型[3]是典型的无监督模型，我们可以认为模型考虑了对降维后数据 $\boldsymbol{x}$ 的分布进行建模（如 PCA [4] 是对降噪后的低维的数据 $\boldsymbol{x}$ 进行建模），因而属于生成模型；而经典的无监督聚类方法 K-Means [5]事实上是对数据 $\boldsymbol{x}$ 的聚类中心进行建模，因而也属于生成模型。

简单地看，输入数据 $\boldsymbol{x}$（和对应的标签 y）生成标签外的数据的是生成模型，输入数据 $\boldsymbol{x}$（和对应的标签 y）生成标签且不涉及到数据 $\boldsymbol{x}$ 生成（中途没有对数据 $\boldsymbol{x}$ 分布或数据 $\boldsymbol{x}$ 和标签 y 的联合分布进行建模）的是鉴别模

[1] 课件 lecture9 p.3

[2] 详见第二部分

[3] 详见第三部分

[4] 详见 D.3

[5] 详见 8.1

型。对于 MAP 视角下的线性回归，虽然推导涉及了数据 $\boldsymbol{x}$ 的分布，但是在求解过程中这项被马上抛弃了，我们没有用到与之相关的任何知识，一般认为也认为带正则项的线性规划模型是鉴别模型。

生成模型当然建模的难度更大，因为条件分布只覆盖了联合分布的一小部分信息，这也导致了在应用生成模型在数据 $\boldsymbol{x}$ 的鉴别任务中表现很多时候没有鉴别模型好，因为生成模型想要 cover 得太多，而这样做往往是以准确率为代价的；此外，从生成模型建立的分布中采样（如使用逆分布法采样）可以返回分布中以较高概率出现的数据 $\boldsymbol{x}$，这实际上做的就是数据 $\boldsymbol{x}$ 的生成，而鉴别模型却做不到这一点，这也是生成模型之所以被称为生成模型的原因；部分生成模型在监督信号缺失的情况下也可以进行训练，这在半监督学习和无监督学习中至关重要。

9.2　EM 算法简介 [6]

EM 算法（Expectation-maximization algorithm）的最终目的是利用 MLE 进行模型参数估计，在基于观测到数据 $\boldsymbol{X}$ 的基础上优化模型参数的似然函数，而问题中难以显式求解的隐变量给我们带来了困扰，因而考虑将似然函数写为

$$\begin{aligned}\max_{\boldsymbol{\theta}} L(\boldsymbol{X};\boldsymbol{\theta}) &= \max_{\boldsymbol{\theta}} p(\boldsymbol{X};\boldsymbol{\theta}) = \max_{\boldsymbol{\theta}} \int p(\boldsymbol{X},\boldsymbol{Z};\boldsymbol{\theta})\mathrm{d}\boldsymbol{Z} \\ &= \max_{\boldsymbol{\theta}} \int p(\boldsymbol{X} \mid \boldsymbol{Z};\boldsymbol{\theta}) p(\boldsymbol{Z};\boldsymbol{\theta})\mathrm{d}\boldsymbol{Z} = \max_{\boldsymbol{\theta}} \mathrm{E}_{\boldsymbol{Z};\boldsymbol{\theta}}[p(\boldsymbol{X} \mid \boldsymbol{Z};\boldsymbol{\theta})] \\ &= \max_{\boldsymbol{\theta}} \mathrm{E}_{\boldsymbol{Z};\boldsymbol{\theta}}[L(\boldsymbol{X} \mid \boldsymbol{Z};\boldsymbol{\theta})]\end{aligned}$$

在实际应用中会考虑改写上面这个式子，有关 EM 算法详细的推导和分析放在了 F.4.2，考虑如下交替更新的过程：

E-step　我们通过当前模型参数 $\boldsymbol{\theta}^{(t)}$ 求解 $\boldsymbol{Z}$ 的分布，结合观测数据 $\boldsymbol{X}$，我们在这一步计算条件分布 $p(\boldsymbol{Z} \mid \boldsymbol{X};\boldsymbol{\theta}^{(t)})$ 得到当前迭代轮数下的期望表达式

$$Q(\boldsymbol{\theta};\boldsymbol{\theta}^{(t)}) := \mathrm{E}_{\boldsymbol{Z} \mid \boldsymbol{X};\boldsymbol{\theta}^{(t)}}[\log L(\boldsymbol{X},\boldsymbol{Z};\boldsymbol{\theta})]$$

M-step　我们通过求解期望表达式的最大值，得到新的模型参数

[6] 课件 lecture9 pp.11, 15-16

$$\boldsymbol{\theta}^{(t+1)} = \arg\max_{\boldsymbol{\theta}} Q(\boldsymbol{\theta}; \boldsymbol{\theta}^{(t)})$$

9.3　GMM [7]

Gussian 混合模型（Gaussian mixture model / GMM）认为每个类的分布服从正态分布，即对于类别 k

$$\boldsymbol{x} \in \mathcal{C}_k \Rightarrow \boldsymbol{x} \sim \mathcal{N}(\boldsymbol{\mu_k}, \boldsymbol{\Sigma_k})$$

设类别总数为 K，对于每个类而言，其密度函数为

$$p(\boldsymbol{x}; \boldsymbol{\theta}) = \sum_{k=1}^{K} p(k; \boldsymbol{\theta}) p(\boldsymbol{x} \mid k; \boldsymbol{\theta}) = \sum_{k=1}^{K} z_k \varphi(\boldsymbol{x}; \boldsymbol{\mu_k}, \boldsymbol{\Sigma_k}),\ z_k = \mathbf{1}_{\boldsymbol{x} \in \mathcal{C}_k}$$

这里 $\varphi(\boldsymbol{x}; \boldsymbol{\mu_k}, \boldsymbol{\Sigma_k})$ 表示正态分布 $\mathcal{N}(\boldsymbol{\mu_k}, \boldsymbol{\Sigma_k})$ 的概率密度函数。数据的类别分布此时满足类别分布（Categorical distribution），$\boldsymbol{z}$ 为指示数据类别的 one-hot 变量，即

$$k \sim \mathrm{Cat}(\boldsymbol{z})$$

one-hot 的限制导致了 $\boldsymbol{z}$ 很难显式求解，其可以视为数据点所属的类别对应的分布。我们考虑用求和归一化的非负的类别参数 $\boldsymbol{w} \in [0, 1]^K$ 来代替这个类别分布。此时类别分布为

$$k \sim \mathrm{Cat}(\boldsymbol{w}),\ \boldsymbol{w} \in \Delta^{K-1}$$

其中 Δ^{K-1} 代表满足求和为 1 且每一维元素非负的向量集合 $\boldsymbol{w}$，这个集合构成 Euclidean 空间中的 $K-1$ 的标准单纯形（Standard simplex）[8]。

$$p(\boldsymbol{x}; \boldsymbol{\theta}) = \sum_{k=1}^{K} p(k; \boldsymbol{\theta}) p(\boldsymbol{x} \mid k; \boldsymbol{\theta}) = \sum_{k=1}^{K} w_k \varphi(\boldsymbol{x}; \boldsymbol{\mu_k}, \boldsymbol{\Sigma_k})$$

此时 $\boldsymbol{w}$ 成为模型参数。这样对数据 $\boldsymbol{x}$ 分布进行建模的模型显然是一个生成模型。我们确定的模型参数 $\boldsymbol{\theta}$ 为 $\{w_k, \boldsymbol{\mu_k}, \boldsymbol{\Sigma_k}\}_{k=1}^{K}$。

[7] 课件 lecture9 pp.4-10, 12

[8] https://en.wikipedia.org/wiki/Simplex

以类别变量作为隐变量（Hidden variable）$\boldsymbol{Z}$，其中 $\boldsymbol{Z}$ 的第 n 列第 k 行 z_{kn} 为数据 $\boldsymbol{x_n}$ 属于第 k 类的指示的 $0-1$ 变量，我们可以考虑 EM 算法求解模型的最优参数。

9.3.1 GMM 更新的 E-Step [9]

在 E-Step，构造期望表达式

$$
\begin{aligned}
Q(\boldsymbol{\theta};\boldsymbol{\theta^{(t)}}) &= \mathrm{E}_{\boldsymbol{Z}\mid\boldsymbol{X};\boldsymbol{\theta}^{(t)}}[\log L(\boldsymbol{X},\boldsymbol{Z};\boldsymbol{\theta})] = \mathrm{E}_{\boldsymbol{Z}\mid\boldsymbol{X};\boldsymbol{\theta}^{(t)}}[\sum_{n=1}^{N}\log p(\boldsymbol{x_n},\boldsymbol{z_n};\boldsymbol{\theta})] \\
&= \sum_{n=1}^{N}\mathrm{E}_{\boldsymbol{z_n}\mid\boldsymbol{x_n};\boldsymbol{\theta}^{(t)}}[\log p(\boldsymbol{x_n},\boldsymbol{z_n};\boldsymbol{\theta})] \\
&= \sum_{n=1}^{N}\sum_{k=1}^{K} p(z_{nk}=1\mid\boldsymbol{x_n};\boldsymbol{\theta^{(t)}})\log(p(z_{nk}=1;\boldsymbol{\theta})p(\boldsymbol{x_n}\mid z_{nk}=1;\boldsymbol{\theta})) \\
&= \sum_{n=1}^{N}\sum_{k=1}^{K} p(k\mid\boldsymbol{x_n};\boldsymbol{\theta^{(t)}})\log(p(k;\boldsymbol{\theta})p(\boldsymbol{x_n}\mid k;\boldsymbol{\theta})) \\
&= \sum_{n=1}^{N}\sum_{k=1}^{K} p(k\mid\boldsymbol{x_n};\boldsymbol{\theta^{(t)}})\log(w_k\varphi(\boldsymbol{x_n};\boldsymbol{\mu_k},\boldsymbol{\Sigma_k}))
\end{aligned}
$$

求解隐变量条件分布，即求解

$$
p(k\mid\boldsymbol{x_n};\boldsymbol{\theta^{(t)}}) = \frac{p(k\mid\boldsymbol{\theta^{(t)}})p(\boldsymbol{x_n}\mid k;\boldsymbol{\theta^{(t)}})}{\sum\limits_{i=1}^{K}p(i;\boldsymbol{\theta^{(t)}})p(\boldsymbol{x_n}\mid i;\boldsymbol{\theta^{(t)}})} = \frac{w_k^{(t)}\varphi(\boldsymbol{x};\boldsymbol{\mu_k^{(t)}},\boldsymbol{\Sigma_k^{(t)}})}{\sum\limits_{i=1}^{K}w_i^{(t)}\varphi(\boldsymbol{x};\boldsymbol{\mu_i^{(t)}},\boldsymbol{\Sigma_i^{(t)}})}
$$

为了在后续求解中表达式的简洁性，我们记

$$
p_{kn}^{(t)} := p(k\mid\boldsymbol{x_n};\boldsymbol{\theta^{(t)}})
$$

此时我们得到了 $Q(\boldsymbol{\theta};\boldsymbol{\theta^{(t)}})$。

9.3.2 GMM 更新的 M-Step [10]

取对数为自然对数，进行 M-Step。依次对如下参数进行更新：

[9] 课件 lecture9 p.10

[10] 课件 lecture9 pp.10,12

类别参数　更新参数 $\boldsymbol{w}$

$$\begin{aligned}\min_{\boldsymbol{w}\in\Delta^{K-1}} Q(\boldsymbol{\theta};\boldsymbol{\theta^{(t)}}) \Rightarrow \min_{\boldsymbol{w}\in\Delta^{K-1}} -\sum_{n=1}^{N}\sum_{k=1}^{K} p_{kn}^{(t)} \ln w_k &= \min_{\boldsymbol{w}\in\Delta^{K}} -\sum_{k=1}^{K}\left(\frac{1}{N}\sum_{n=1}^{N} p_{kn}^{(t)}\right)\ln w_k \\ = \min_{\boldsymbol{w}\in\Delta^{K-1}} -\sum_{k=1}^{K} \bar{p}_k^{(t)} \ln w_k &= \min_{\boldsymbol{w}\in\Delta^{K-1}} H(\bar{\boldsymbol{p}}^{\boldsymbol{(t)}}, \boldsymbol{w})\end{aligned}$$

由于所有的 $p_{kn}^{(t)}$ 事实上已经在 M-Step 中求出了，即 $\bar{\boldsymbol{p}}^{\boldsymbol{(t)}}$ 是完全已知的，因而我们也可以优化 KL 散度[11]

$$\begin{aligned}\min_{\boldsymbol{w}\in\Delta^{K-1}} H(\bar{\boldsymbol{p}}^{\boldsymbol{(t)}}, \boldsymbol{w}) &= \min_{\boldsymbol{w}\in\Delta^{K-1}} H(\bar{\boldsymbol{p}}^{\boldsymbol{(t)}}) + D_{KL}(\bar{\boldsymbol{p}}^{\boldsymbol{(t)}} \mid\mid \boldsymbol{w}) \\ &\Rightarrow \min_{\boldsymbol{w}\in\Delta^{K-1}} D_{KL}(\bar{\boldsymbol{p}}^{\boldsymbol{(t)}} \mid\mid \boldsymbol{w}) \geqslant 0\end{aligned}$$

我们已经知道交叉熵和 KL 散度取得最小值的条件，从而

$$\boldsymbol{w^{(t+1)}} = \bar{\boldsymbol{p}}^{\boldsymbol{(t)}} \Rightarrow w_k^{(t+1)} = \bar{p}_k^{(t)} = \frac{1}{N}\sum_{n=1}^{N} p_{kn}^{(t)},\ k = 1, 2, \cdots, K$$

考察 $\varphi(\boldsymbol{x_n}; \boldsymbol{\mu_k}, \boldsymbol{\Sigma_k})$ 的表达式

$$\varphi(\boldsymbol{x_n}; \boldsymbol{\mu_k}, \boldsymbol{\Sigma_k}) = (\det(2\pi\boldsymbol{\Sigma_k}))^{-1/2} \exp\left(-\frac{1}{2}\left\|\boldsymbol{x_n} - \boldsymbol{\mu}\right\|^2_{\boldsymbol{\Sigma_k^{-1}}}\right)$$

期望　更新参数 $\boldsymbol{w}$ 更新参数 $\boldsymbol{\mu_k}$

$$\begin{aligned}\min_{\boldsymbol{\mu}} Q(\boldsymbol{\theta};\boldsymbol{\theta^{(t)}}) &= \min_{\boldsymbol{\mu}} -\sum_{i=1}^{N} p_{kn}^{(t)} \ln \varphi(\boldsymbol{x_n}; \boldsymbol{\mu_k}, \boldsymbol{\Sigma_k}) \\ &= \min_{\boldsymbol{\mu}} -\frac{1}{2}\sum_{i=1}^{N} p_{kn}^{(t)} \left\|\boldsymbol{x_i} - \boldsymbol{\mu}\right\|^2_{\boldsymbol{\Sigma_j^{-1}}}\end{aligned}$$

求导得到

$$\frac{\partial Q}{\partial \boldsymbol{\mu}} = \boldsymbol{\Sigma_k^{-1}} \sum_{n=1}^{N} p_{kn}^{(t)} (\boldsymbol{x_n} - \boldsymbol{\mu})$$

令导数为 0 得到

[11] 详见 F.1.3 结尾部分

$$\boldsymbol{\mu_k^{(t+1)}} := \frac{\sum\limits_{n=1}^{N} p_{kn}^{(t)} \boldsymbol{x_n}}{\sum\limits_{n=1}^{N} p_{kn}^{(t)}}$$

这事实上是一个带权的平均数计算公式。

协方差　更新参数 $\boldsymbol{\Sigma_k}$

$$\begin{aligned}
\min_{\boldsymbol{\Sigma}} Q(\boldsymbol{\theta}; \boldsymbol{\theta^{(t)}}) &= \min_{\boldsymbol{\Sigma}} -\sum_{n=1}^{N} p_{kn}^{(t)} \ln \varphi(\boldsymbol{x_n}; \boldsymbol{\mu_k}, \boldsymbol{\Sigma_k}) \\
&= \min_{\boldsymbol{\Sigma}} -\frac{1}{2} \sum_{n=1}^{N} p_{kn}^{(t)} \left(\|\boldsymbol{x_n} - \boldsymbol{\mu_k}\|_{\boldsymbol{\Sigma^{-1}}}^2 + \ln \det(\boldsymbol{\Sigma}) \right) \\
&\approx \min_{\boldsymbol{\Sigma}} -\frac{1}{2} \sum_{n=1}^{N} p_{kn}^{(t)} \left(\left\| \boldsymbol{x_n} - \boldsymbol{\mu_k^{(t+1)}} \right\|_{\boldsymbol{\Sigma^{-1}}}^2 - \ln \det(\boldsymbol{\Sigma^{-1}}) \right)
\end{aligned}$$

考虑行列式求导

$$\frac{\partial \ln \det(\boldsymbol{A})}{\partial \boldsymbol{A}} = \frac{\partial \ln \det(\boldsymbol{A})}{\partial \det(\boldsymbol{A})} \frac{\partial \det(\boldsymbol{A})}{\partial \boldsymbol{A}} = \frac{1}{\det(\boldsymbol{A})} \det(\boldsymbol{A}) (\boldsymbol{A^{-1}})^T = (\boldsymbol{A^{-1}})^T$$

证明考虑行列式的代数余子式 A_{ij} 和伴随矩阵得到

$$\frac{\partial \det(\boldsymbol{A})}{\partial a_{ij}} = \frac{\partial}{\partial a_{ij}} \sum_{k=1}^{N} a_{ik} A_{ik} = A_{ij} \Rightarrow \frac{\partial \det(\boldsymbol{A})}{\partial \boldsymbol{A}} = (\boldsymbol{A^*})^T = \frac{1}{\det(\boldsymbol{A})} (\boldsymbol{A^{-1}})^T$$

对目标函数求导得到，注意这里为了求导方便对 $\boldsymbol{\Sigma^{-1}}$ 求导而不是对 $\boldsymbol{\Sigma}$ 求导

$$\frac{\partial Q}{\partial \boldsymbol{\Sigma^{-1}}} = -\frac{1}{2} \sum_{n=1}^{N} p_{kn}^{(t)} (\boldsymbol{x_n} - \boldsymbol{\mu_k^{(t+1)}}) (\boldsymbol{x_n} - \boldsymbol{\mu_k^{(t+1)}})^T - \frac{1}{2} \sum_{i=1}^{N} p_{kn}^{(t)} \boldsymbol{\Sigma}$$

令导数为 0 得到

$$\boldsymbol{\Sigma_k^{(t+1)}} := \frac{\sum\limits_{n=1}^{N} p_{kn}^{(t)} (\boldsymbol{x_n} - \boldsymbol{\mu_k^{(t+1)}}) (\boldsymbol{x_n} - \boldsymbol{\mu_k^{(t+1)}})^T}{\sum\limits_{n=1}^{N} p_{kn}^{(t)}}$$

这事实上是一个带权的协方差计算公式。

重复这个迭代过程直至参数收敛或达到迭代轮数，EM 算法结束，GMM 模型参数更新停止。

如示意图 [12] 所示，在 GMM 中，以各个类的均值（聚类中心）为中心，每一类都产生了相应的核密度。如果把类比作星团，数据点就像是星团中的小天体，聚类中心就像是星团中心虚拟的中心天体，我们考虑星团对其他数据的作用时会用这个虚拟的天体等价地代替；核密度就像是等价的中心天体释放出的的引力场，聚类中心依靠核密度将距离较近的小天体俘获，俘获的小天体又影响着虚拟的中心天体的位置和引力场。在 GMM 中，数据点的位置在聚类中不发生变动，这个星团的天体之间其实是相对静止的，考虑每个天体的引力场并让天体之间发生相对运动时，即考虑以每个数据点为中心并让数据点动起来时，我们就得到了 Mean-shift 算法 [13]。

9.4　K-Means 的 EM 算法视角 [14]

我们观察到 K-Means 的聚类中心 $\boldsymbol{c}$ 更新公式和 GMM 中期望 $\boldsymbol{\mu}$ 的更新公式是很类似的 [15]。在 K-Means 中，模型参数为 $\boldsymbol{\theta}=\{w_k,\boldsymbol{c_k}\}_{k=1}^K$。考虑 E-Step，在 K-Means 中，E-Step 由于 GMM 中正态分布方差的带来的不确定性使得类别划分中体现出了软聚类（Soft clustering）的特性，因为 GMM 中正态分布是存在交集的，因而分类时比较模糊的；而相较于 GMM 来说 K-Means 聚类是硬聚类（Hard clustering），因为其分类边界和分类结果是明确的[16]。考虑 GMM 中正态分布方差很小的情况，此时正态分布之间几乎不存在交集时，分类的模糊性消失，分类退化为 K-Means。我们选取 GMM 每一类的方差为 $\varepsilon^2\boldsymbol{I}$，令 $\varepsilon\to 0$ 得到[16]

$$p(k\mid\boldsymbol{x};\boldsymbol{\theta^{(t)}})=\frac{w_k\varphi(\boldsymbol{x};\boldsymbol{c_k^{(t)}},\varepsilon^2\boldsymbol{I})}{\sum\limits_{i=1}^{K}w_i\varphi(\boldsymbol{x};\boldsymbol{c_i^{(t)}},\varepsilon^2\boldsymbol{I})}=\begin{cases}1 & k=\arg\min\limits_{i\in\{1,2,\cdots,K\}}\|\boldsymbol{x}-\boldsymbol{c_i}\|_2\\ 0 & \text{otherwise}\end{cases}$$

上面的结果其实很容易验证，我们简单证明一下，假设 $w_k>0$ 均成立

[12] 课件 lecture9 p.6

[13] 详见 G.4

[14] 课件 lecture9 p.13

[15] 详见 8.1

[16] https://blog.csdn.net/sjyttkl/article/details/107746928

$$\varphi(\boldsymbol{x};\boldsymbol{c_k^{(t)}},\varepsilon^2\boldsymbol{I})=(2\pi\varepsilon^2)^{-D/2}\exp\left(-\frac{1}{2\varepsilon^2}\left\|\boldsymbol{x}-\boldsymbol{c_k}\right\|_{\boldsymbol{2}}^2\right)$$

取最佳的离 $\boldsymbol{x}$ 最近的聚类中心 $\boldsymbol{c}^*$ 得到

$$\begin{aligned}p(k\mid\boldsymbol{x};\boldsymbol{\theta^{(t)}})&=\frac{w_k\exp\left(-\frac{1}{2\varepsilon^2}\left\|\boldsymbol{x}-\boldsymbol{c_k}\right\|_{\boldsymbol{2}}^2\right)}{\sum\limits_{i=1}^{K}w_i\exp\left(-\frac{1}{2\varepsilon^2}\left\|\boldsymbol{x}-\boldsymbol{c_i}\right\|_{\boldsymbol{2}}^2\right)}\\&=\frac{w_k\exp\left(-\frac{1}{2\varepsilon^2}(\left\|\boldsymbol{x}-\boldsymbol{c_k}\right\|_{\boldsymbol{2}}^2-\left\|\boldsymbol{x}-\boldsymbol{c}^*\right\|_{\boldsymbol{2}}^2)\right)}{\sum\limits_{i=1}^{K}w_i\exp\left(-\frac{1}{2\varepsilon^2}(\left\|\boldsymbol{x}-\boldsymbol{c_i}\right\|_{\boldsymbol{2}}^2-\left\|\boldsymbol{x}-\boldsymbol{c}^*\right\|_{\boldsymbol{2}}^2)\right)}\end{aligned}$$

因而当且仅当 $\boldsymbol{c_k}=\boldsymbol{c}^*$ 时响应值为 1，其余当 $\varepsilon\to 0$ 时响应值均为 0。为了在后续求解中表达式的简洁性，我们记

$$r_{kn}^{(t)}:=p(k\mid\boldsymbol{x_n};\boldsymbol{\theta^{(t)}})$$

在 M-Step 中我们更新类别参数 w_k

$$w_k^{(t+1)}=\bar{p}_k^{(t)}=\frac{1}{N}\sum_{n=1}^{N}r_{kn}^{(t)}=\frac{|\mathcal{C}_k|}{N}$$

更新聚类中心 $\boldsymbol{c_k}$

$$\boldsymbol{c_k^{(t+1)}}:=\frac{\sum\limits_{n=1}^{N}r_{kn}^{(t)}\boldsymbol{x_n}}{\sum\limits_{n=1}^{N}r_{kn}^{(t)}}=\frac{1}{|\mathcal{C}_k|}\sum_{\boldsymbol{x}\in\mathcal{C}_k}\boldsymbol{x}$$

我们于是得到了一种崭新的视角审视朴素的 K-Means 聚类算法，将其视为 GMM 的一种特例。

第十章　聚类的非参数方法与 Mean-shift 算法

10.1　参数模型和非参模型 [1]

机器学习中的模型的模型参数确定了模型拟合的数据分布，参数模型（Parametric model）引入额外的参数确定这个数据分布，而非参模型（Non-parametric model）以数据本身确定了这个分布。二者在某种程度上是具有共性的，因为我们可以简单地认为非参模型其实就是把输入的数据作为参数去进行迭代更新等后续的优化操作。

这其实会导致一个问题，我们把输入数据作为参数，参数量和数据量之间是存在正比关系的，这在赋予了模型拟合更加复杂多变的数据分布的能力的同时，参数量的增加容易导致过拟合的问题，且由于参数量和数据量的关系这个问题在非参模型中比较难缓解，还有一个问题就是参数量的提升带来了时间复杂度的显著提升的问题，由于对数据分布先验知识的缺失，我们还通常还要求大量的输入数据用于拟合，这样导致了模型的训练的时间比较长。当然参数模型虽然自身在先验知识的帮助下训练快，需要的训练数据少，但是由于模型的简洁性而导致了模型在优化过程中受到的约束较大，对复杂的数据拟合能力弱，容易导致欠拟合的问题 [2]。

由于在参数模型中参数量是固定的，如果我们基于 MLE 视角认为参数是固定的，数据只是从分布得到的一个采样，则当新数据的引入时我们可以把数据直接代入我们训练好的参数模型即可得到最终的结果；在非参模型中参数量本身就是一个不固定的随着样本量的改变而改变的量，且输入数据对应着参数的初值，当新数据引入时（或者只是简单的改变输入的数据时）参数是固定的这种假设就行不通了，此时为了应对新数据的引入带来的参数的

[1] 课件 lecture5 p.13, 课件 lecture10 p.8

[2] 详见 4.2 中间部分

改变模型就需要将原有的数据和加入的新数据一起再去训练一次才能得到结果，这在数据量较大且数据需要频繁加入的场合其实是一件非常麻烦的事情。前者的模型推断是归纳式的（inductive）。什么是归纳？根据高中数学知识，从特殊到一般就是归纳。从特定任务到一般任务，在训练过程中在只观测到训练数据和标签的基础上训练模型，在测试过程输入测试数据去预测标签的过程就是归纳的过程；后者的模型推断是直推式的（transductive），这样的模式和推断的区别在于在训练阶段会使用到测试集的数据信息，换言之测试数据在训练阶段发生“泄漏”，而测试数据的真实标签依然只有在测试阶段才能观测到[3]。毫不意外地在相同条件下直推式的推断的准确率通常会比归纳式的更高，这是因为直推式更多地利用了测试集中数据分布的信息。

10.2　MCMC 算法简介[4]

MCMC 算法（Markov chain Monte Carlo algorithm）是 Bayesian 推断（Bayesian inference）在 GMM 应用。MCMC 算法和 EM 算法的本质区别，EM 算法基于概率学派的观点，它把模型参数看作是已知去求解；而 MCMC 算法则基于 Bayesian 学派的观点，把模型参数看作是未知的，是一个随机变量，服从一定的分布 [5]。对于难以显式求解的隐变量，EM 算法想的是通过已知的模型参数先把隐变量的分布估计出来，然后再对得到的表达式进行优化问题的求解去优化模型参数，在初始化模型参数确定的情况下，这样的优化方向是确定的；而 MCMC 算法想的是既然隐变量难以求解，我们就通过采样把它从已知的模型参数的分布中采样出来，采样出来后再去确定模型参数的分布，再把模型参数从分布中采样出来。MCMC 以概率的形式更新参数，大大提升了训练的不确定性，也使得通过采样能够得到的参数增多了，以往难以优化的如聚类数等参数只要能够确定合适的分布和初始化参数就能进行求解，然而这一切是以模型复杂度为代价的，从参数的采样的复杂度往往不够高效的，如何设计合理的分布，使得模型最终不仅效果好且兼顾了采样的高效性，是需要考虑的问题。

[3] https://www.zhihu.com/question/68275921

[4] 课件 lecture10 pp.4-7

[5] 详见 2.3

10.3　KDE 简介 [6]

和 GMM 模型以聚类中心建立带参的类似引力场的概率密度不同，核密度估计（Kernel density estimation / KDE）以数据点为中心利用核函数建立非参的概率密度，这个给定的概率密度我们称之为核密度，这个密度累积得到了最终的概率密度。示意图左图[7]显示在 1 维的情况下当取核密度（Kernel density）为

$$\kappa(x)=\begin{cases}1 & \lfloor x\rfloor=0\\ 0 & \text{otherwise}\end{cases}$$

如此最终每个样本的概率密度可以写为

$$p_h(x_i)=\frac{1}{Nh}\sum_{n=1}^{N}\kappa\left(\frac{x-x_n}{h}\right)=\frac{1}{Nh}\sum_{h\lfloor x_i/h\rfloor\leqslant x_n<h\lceil x_i/h\rceil}1$$

我们得到了等分的小区间长度为 h 的频率分布直方图的概率密度估计。

示意图右图显示当取核密度为一维的正态分布函数时

$$\kappa(x)=\frac{1}{\sqrt{2\pi}}\exp\left(-\frac{x^2}{2}\right)$$

如此最终每个样本的概率密度可以写为

$$p_h(x_i)=\frac{1}{Nh}\sum_{n=1}^{N}\kappa\left(\frac{x-x_n}{h}\right)=\frac{1}{N\sqrt{2\pi}h}\sum_{n=1}^{N}\exp\left(-\frac{1}{2h^2}(x_i-x_n)^2\right)$$

结合以上公式，KDE 可以视为非参的（此时权值 w_k 均取相同值，带宽 h 控制了每个类的的方差）Gaussian 成分数和样本量相同的 GMM 模型，其更新从对参数更新到对数据点的更新，体现了非参模型的特点。有关 KDE 更详细的理论推导见附录 G。

[6] 课件 lecture10 pp.9-10

[7] 课件 lecture10 p.10

10.4　RBF 核下的 Mean-shift 算法 [8]

我们进行 KDE 实际上在对样本 $\boldsymbol{x_i}$ 出现的似然函数进行估测，考虑选取核为 RBF 核[9]

$$p(\boldsymbol{x} \mid \boldsymbol{X}) = \frac{1}{N}\sum_{n=1}^{N}\kappa_h(\boldsymbol{x}-\boldsymbol{x_n})$$

在上一轮的更新后一轮的更新前输入的数据是不变的，因而此时将非参模型中模型参数视为数据并将参数固定住是可以做到的，基于这个观点我们使用 MLE 得到最优化模型参数事实上是该轮数据迭代更新的过程，考虑对数似然函数后我们有

$$\begin{aligned}\max_{\boldsymbol{X}} \log L(\boldsymbol{X} \mid \boldsymbol{X^{(t)}}) = \max_{\boldsymbol{X}} \log p(\boldsymbol{X^{(t)}} \mid \boldsymbol{X}) = \max_{\boldsymbol{X}} \sum_{m=1}^{N} \log p(\boldsymbol{x_m^{(t)}} \mid \boldsymbol{X}) \\ = \max_{\boldsymbol{X}} \sum_{m=1}^{N} \log \sum_{n=1}^{N} \kappa_h(\boldsymbol{x_m^{(t)}} - \boldsymbol{x_n})\end{aligned}$$

实际上我们只需考虑优化问题

$$\min_{\boldsymbol{x}} -\log \sum_{n=1}^{N} \kappa_h(\boldsymbol{x_m^{(t)}} - \boldsymbol{x}) = \min_{\boldsymbol{x}} -\log \sum_{n=1}^{N} \kappa_h(\boldsymbol{x} - \boldsymbol{x_m^{(t)}}) =: \min_{\boldsymbol{x}} Q(\boldsymbol{x})$$

求导后，我们得到了

$$\nabla_{\boldsymbol{x}} Q(\boldsymbol{x}) = -\frac{\partial}{\partial \boldsymbol{x}} \log \sum_{n=1}^{N} \kappa_h(\boldsymbol{x}-\boldsymbol{x_m^{(t)}}) = -\frac{1}{\sum_{n=1}^{N} \kappa_h(\boldsymbol{x} - \boldsymbol{x_m^{(t)}})} \sum_{n=1}^{N} \frac{\partial}{\partial \boldsymbol{x}} \kappa_h(\boldsymbol{x}-\boldsymbol{x_m^{(t)}})$$

由 RBF 核公式

$$\kappa_h(\boldsymbol{x}-\boldsymbol{y}) = \varphi(\boldsymbol{x}; \boldsymbol{y}, h^2\boldsymbol{I}) = (2\pi N h^2)^{-D/2} \exp\left(-\frac{1}{2h^2}\left\|\boldsymbol{x}-\boldsymbol{y}\right\|_2^2\right)$$

得到

[8] 课件 lecture10 pp.12-14

[9] 更一般的情况的见 G.4

$$\frac{\partial}{\partial \boldsymbol{x}}\kappa_h(\boldsymbol{x}-\boldsymbol{y}) = -\frac{1}{h^2}\kappa_h(\boldsymbol{x}-\boldsymbol{y})(\boldsymbol{x}-\boldsymbol{y})$$

从而

$$\begin{aligned}\nabla_{\boldsymbol{x}} Q(\boldsymbol{x}) &= -\frac{1}{h^2\sum_{n=1}^{N}\kappa_h(\boldsymbol{x}-\boldsymbol{x}_{\boldsymbol{m}}^{(\boldsymbol{t})})}\sum_{n=1}^{N}\kappa_h(\boldsymbol{x}-\boldsymbol{x}_{\boldsymbol{m}}^{(\boldsymbol{t})})(\boldsymbol{x}-\boldsymbol{x}_{\boldsymbol{m}}^{(\boldsymbol{t})}) \\ &= \frac{1}{h^2}\left(\boldsymbol{x} - \frac{\sum_{n=1}^{N}\kappa_h(\boldsymbol{x}-\boldsymbol{x}_{\boldsymbol{m}}^{(\boldsymbol{t})})\boldsymbol{x}_{\boldsymbol{m}}^{(\boldsymbol{t})}}{\sum_{n=1}^{N}\kappa_h(\boldsymbol{x}-\boldsymbol{x}_{\boldsymbol{m}}^{(\boldsymbol{t})})}\right) = \frac{1}{h^2}(\boldsymbol{x}-\hat{m}(\boldsymbol{x}))\end{aligned}$$

导数等于 0 参数实际上很难求解，考虑使用梯度下降进行参数更新

$$\boldsymbol{x}^{(\boldsymbol{t}+\mathbf{1})} := \boldsymbol{x}^{(\boldsymbol{t})} - \tau\nabla_{\boldsymbol{x}} Q(\boldsymbol{x})$$

取 $\tau = h^2$ 得到

$$\boldsymbol{x}^{(\boldsymbol{t}+\mathbf{1})} := \hat{m}(\boldsymbol{x}^{(\boldsymbol{t})})$$

当我们考虑 KNN 对梯度更新进行改进时我们利用的是 SGD 的思想取数据点的邻集（相当于一个 batch size 为 K 的 batch）对梯度进行近似。

将数据点视为一堆弹性的蹦床上的钢珠，钢珠使得蹦床产生相同程度的凹陷，KDE 的过程可以形象地描述为钢珠向着凹陷的最大处移动的过程，每次迭代的过程向凹陷对应的梯度方向进行移动，随着钢珠的移动凹陷的分布随之改变，最终钢珠将汇聚为一个个点，代表了聚类得到的各个类别。有时候为了减少计算量我们会将核密度固定下来，每次数据更新将不考虑对核矩阵的参数进行更新，此时将蹦床更换为一个带有初始的凹陷的光滑的刚性曲面，钢珠移动时不会改变凹陷的分布。

10.5　Mean-shift 的 EM 算法视角 [10]

我们已经知道了从 EM 算法视角看 K-Means 它实际上是 GMM 模型的一个特例 [11]。我们在 10.3 铺垫了 KDE 和 GMM 的联系，很自然地，从

[10] 课件 lecture10 p.15

[11] 详见 9.4

EM 算法视角看 KDE 实际上也是 GMM 模型的一个特例。考虑模型参数为 $\boldsymbol{\theta} = \{\boldsymbol{x_k}\}_{k=1}^{N}$，在非参的 KDE 中，我们以数据作为模型参数，在原有 GMM 的基础上，我们令 w_k 都取相同值，因为在 KDE 中对于每个样本考虑其核密度时的我们的权重设置恒定为 $1/N$，再令方差为 $h^2\boldsymbol{I}$ 我们得到在 E-Step 更新隐变量分布

$$p(k \mid \boldsymbol{x}; \boldsymbol{\theta^{(t)}}) = \frac{w_k \varphi(\boldsymbol{x}; \boldsymbol{x_k^{(t)}}, h^2\boldsymbol{I})}{\sum\limits_{i=1}^{K} w_i \varphi(\boldsymbol{x}; \boldsymbol{x_i^{(t)}}, h^2\boldsymbol{I})} = \frac{\kappa_h(\boldsymbol{x} - \boldsymbol{x_k^{(t)}})}{\sum\limits_{i=1}^{N} \kappa_h(\boldsymbol{x} - \boldsymbol{x_i^{(t)}})}$$

在 M-Step 更新聚类中心 $\boldsymbol{x_k}$

$$\boldsymbol{x_k^{(t+1)}} := \frac{\sum\limits_{n=1}^{N} p(k \mid \boldsymbol{x_n^{(t)}}; \boldsymbol{\theta^{(t)}}) \boldsymbol{x_n^{(t)}}}{\sum\limits_{n=1}^{N} p(k \mid \boldsymbol{x_n^{(t)}}; \boldsymbol{\theta^{(t)}})} = \frac{\sum\limits_{n=1}^{N} \kappa_h(\boldsymbol{x_k^{(t)}} - \boldsymbol{x_n^{(t)}}) \boldsymbol{x_n^{(t)}}}{\sum\limits_{n=1}^{N} \kappa_h(\boldsymbol{x_k^{(t)}} - \boldsymbol{x_n^{(t)}})} = \hat{m}(\boldsymbol{x_k^{(t)}})$$

在 EM 算法中，归属于各个类的概率是我们预设的

$$p(k; \boldsymbol{\theta^{(t)}}) = w_k = 1/N$$

而 Gaussian 成分

$$p(\boldsymbol{x} \mid k; \boldsymbol{\theta^{(t)}}) = \kappa_h(\boldsymbol{x} - \boldsymbol{x_k}) = (2\pi h^2)^{-D/2} \exp\left(-\frac{1}{2h^2} \|\boldsymbol{x} - \boldsymbol{x_k}\|_2^2\right)$$

第五部分

数据分类

第十一章　传统线性分类方法

11.1　回归问题与分类问题 [1]

回归和分类问题目的均为求解一个从样本到标签的映射，对于回归问题而言标签是连续的，而对于分类问题而言标签是离散的，这样的说法容易使得我们认为回归和分类仅仅只是标签连续和离散的不同而已，两个问题的方法应该很容易迁移才是，但是很多时候情况却没有那么简单。更本质地，对于回归问题而言标签通常是定量的，即标签具有明显的数值意义，而对于分类问题而言标签有时是定量的（Quantitative），即使有时没有明显的数值反映但是却表征了某种程度，有时是定性的（Qualitative），这个时候标签可以认为是单纯的编码，这就使得回归问题的处理方式和分类问题有很大的不同。对于回归问题，我们可能会直接学一个从样本到空间的映射函数。

从频率学派的观点，我们通常是那么解释回归模型的，对于给定样本 $\boldsymbol{x}$ 和函数 f，我们假定标签 y 是按照如下方式生成的，且我们通常假设噪声是零均值的且和 $f(\boldsymbol{x})$ 独立

$$y = f(\boldsymbol{x}) + \varepsilon$$

假定标签对应的随机变量为 Y，样本对应的随机变量为 X，我们实际上在学习一个条件期望

$$f(\boldsymbol{x}) = E[f(\boldsymbol{x}) + \varepsilon] = E[Y \mid X = \boldsymbol{x}]$$

而后者因为我们有时候没有数值意义的指导，对于给定样本 $\boldsymbol{x}$，标签 y 可能是从某个参数为 $\boldsymbol{\theta}$ 分布中采出来的

$$y \sim p(Y = y \mid X = \boldsymbol{x}; \boldsymbol{\theta})$$

[1] 课件 lecture11 pp.3-5

因而我们实际上是在学一个条件分布。而我们最终的学习到的映射为

$$f(\boldsymbol{x}) = \arg\max_y p(Y = y \mid X = \boldsymbol{x}; \boldsymbol{\theta}) = \arg\max_y p(Y = y, X = \boldsymbol{x}; \boldsymbol{\theta})$$

我们有时还会认为分类任务是在学习一个样本空间的划分，或者说决策边界。回归任务中我们学习的拟合曲线需要尽可能地穿过样本点，而分类任务中的决策边界则不同。决策边界（Decision boundary）指的是将样本空间进行划分的一系列超曲面（hypersurface），使得划分的每个部分都属于各自唯一对应的类 [2]。我们学习的决策边界需要某种程度上尽可能的不要和样本点靠的太近从而样本点能够更好地被区分。在回归问题中样本标签对和拟合曲线之间存在近似的等式关系，而在分类问题中样本标签对和决策边界之间存在近似的不等式关系。

11.2　K-NN 分类器 [3]

在 G.1 中我们已经提到过了非参统计中对概率密度估计式

$$p_{\mathcal{W}}(\boldsymbol{x}) = \frac{k}{NV}$$

在样本量固定的情况下，固定窗口 $\mathcal{W}$ （这里表现为限制带宽 h）而变动 k，这一思想引出了 KDE，限定类别 j 的情况下其最简单的表达式为（此时上下式子的 NV 都被消去了）

$$p_h(y = j \mid \boldsymbol{x}) = \frac{p_h(y = j, \boldsymbol{x})}{p_h(\boldsymbol{x})} = \frac{|\mathcal{N}_h(\boldsymbol{x}) \cap \mathcal{C}_j|}{|\mathcal{N}_h(\boldsymbol{x})|}$$

考虑一般的核函数，其实就是考虑对窗口内样本点加权

$$p_h(y = j \mid \boldsymbol{x}) = \frac{\sum_{n=1}^{N} \kappa_h(\boldsymbol{x} - \boldsymbol{x_n}) \mathbf{1}_{\boldsymbol{x_n} \in \mathcal{C}_j}}{\sum_{n=1}^{N} \kappa_h(\boldsymbol{x} - \boldsymbol{x_n})}$$

我们得到了 Nadaraya-Watson 估计器。G.3 引入的 Nadaraya-Watson 估计器的背景是回归问题，其标签变量是连续的，我们在此时多考虑了对标签的

[2] https://en.wikipedia.org/wiki/Decision_boundary

[3] 课件 lecture11 p.6

加权，并取期望得到 Nadaraya-Watson 估计器。对于回归和分类两个截然不同的问题我们最终得到的相应的期望和概率的形式居然如此的相似。

固定 k 而变动窗口 $\mathcal{W}$ 将引出 K-NN 分类器（K-nearest Neighbors classifier），限定类别 j 的情况下其表达式为

$$p_k(y=j \mid \boldsymbol{x}) = \frac{p_k(y=j,\boldsymbol{x})}{p_k(\boldsymbol{x})} = \frac{|\mathcal{N}_k(\boldsymbol{x}) \cap \mathcal{C}_j|}{k}$$

我们也可以考虑对窗口内样本点加权

$$p_k(y=j \mid \boldsymbol{x}) = \frac{\sum\limits_{\boldsymbol{x_n}\in\mathcal{N}_k(\boldsymbol{x})} \kappa_h(\boldsymbol{x}-\boldsymbol{x_n})\mathbf{1}_{\boldsymbol{x_n}\in\mathcal{C}_j}}{\sum\limits_{\boldsymbol{x_n}\in\mathcal{N}_k(\boldsymbol{x})} \kappa_h(\boldsymbol{x}-\boldsymbol{x_n})}$$

我们得到了 K-NN 优化的 Nadaraya-Watson 估计器。因而 K-NN 可以视为 Nadaraya-Watson 估计器的一个特例。

注意邻域 $\mathcal{N}_h(\boldsymbol{x})$ 的是已知的（通常取 $h/2$ 为观测窗口对应的超立方体的边长的一半，此时观测窗口体积为 h^D），而含有的样本量是未知的，而邻域 $\mathcal{N}_k(\boldsymbol{x})$ 是未知的（其对应着选用的距离张成的超球，半径是未知的），而含有的样本量是已知的（即为 k）。尽管 K-NN 和 KDE 很多时候是一同出现的，但需要知道的是两者思想的出发点是有很大不同的。

11.3　Naïve Bayes 分类器简介 [4]

Naïve Bayes 分类器（Naïve Bayes classifier）通过对数据和其带有的标签的联合分布的建模实现对标签的条件分布的建模，是经典的生成模型。模型需要做出样本特征的每个维度都是独立的假设，此后联合分布将拆分为标签分布和每个维度的数据在给定标签下的条件分布，这个假设可以通过对数据进行白化处理（Whitening）实现[5]。我们对维度独立性的让步和对条件分布的建模使得我们能够通过对联合概率（或标签在给定数据的条件概率）的计算实现分类。我们介绍如下三种常见的 Naïve Bayes 分类器：

Gaussian Naïve Bayes　模型假设数据的条件分布服从正态分布，正态分布参数可以通过 MLE 得到，对于数据分布不服从正态分布的情况，使用 KDE 是个方法，即把属于第 k 类的样本全部挑出来，利用核函数设置权重

[4] 课件 lecture11 pp.7-8

[5] 详见 D.3.4

得到估计的概率密度，即

$$p(x_d \mid y=j)=\frac{p_h(y=j,\boldsymbol{x})}{p_h(y=j)}=\frac{1}{|\mathcal{C}_j|}\sum_{n=1}^{N}\kappa_h(x_d-x_{dn})\mathbf{1}_{\boldsymbol{x_n}\in\mathcal{C}_j}$$

Multinomial Naïve Bayes　模型假设数据的条件分布服从多项分布(Multinomial distribution) [6]，我们把非负整数数据视为某种计数。想象足够多的小球，每个小球的颜色为颜色 d 的概率为 $p_{k,d}$，一次实验中我们一共抽出了颜色为 d 的小球 x_d 个，考虑对小球排列的结果去重，我们最终计算得到概率为

$$p(\boldsymbol{x}\mid y=k)=\frac{\left(\sum\limits_{c=1}^{C}x_d\right)!}{\prod\limits_{c=1}^{D}x_d!}\prod_{c=1}^{D}p_{k,d}^{x_d}$$

Bernoulli Naïve Bayes　模型假设数据的条件分布服从 Bernoulli 分布(Bernoulli distribution)。这样的假设对数据的要求就更加严格了，它干脆直接要求数据是 0-1 的。事实上我们使用多项分布一般需要基于我们采集得到的每一维数据能被解释为是某次独立重复实验的计数，而我们使用 Bernoulli 分布一般需要基于我们采集到每一维数据能被解释为一个事件是否发生的指示变量。而且维度之间的独立性是个很大的问题，这一点需要预先得到保证，这是由于非负整数对数据的数值运算有很大的限制，此时数据的白化变得异常困难，因为白化矩阵的作用会使得非负整数的限制难以保持。

11.4　LDA 及拓展 [7]

11.4.1　LDA [8]

LDA（Linear discriminant analysis）考虑计算对数似然比（Logarithm of likelihood ratio），在二分类中该值大于 0 将被归属于 $y=1$ 的一类，而小于 0 将被归属于 $y=0$ 的一类，将含有参数 $\boldsymbol{x}$ 的项整理到一起

[6] https://en.wikipedia.org/wiki/Multinomial_distribution

[7] 课件 lecture11 pp.9-14

[8] 课件 lecture11 pp.9-10

$$\begin{aligned}\log\frac{p(\boldsymbol{x}\mid y=1)}{p(\boldsymbol{x}\mid y=0)} &= \log\frac{\varphi(\boldsymbol{x};\boldsymbol{\mu_1},\boldsymbol{\Sigma_1})}{\varphi(\boldsymbol{x};\boldsymbol{\mu_0},\boldsymbol{\Sigma_0})}\\ &= \frac{1}{2}\left\|\boldsymbol{x}-\boldsymbol{\mu_0}\right\|^2_{\boldsymbol{\Sigma_0^{-1}}} - \frac{1}{2}\left\|\boldsymbol{x}-\boldsymbol{\mu_1}\right\|^2_{\boldsymbol{\Sigma_1^{-1}}} + \log\frac{\det(\boldsymbol{\Sigma_1})}{\det(\boldsymbol{\Sigma_0})}\end{aligned}$$

对于分类问题的求解，LDA 给出了另一种对条件进行松弛的方式，即使用正态分布建模给定类别时数据的条件分布，且认为两个类别的数据的协方差矩阵是相同的，区别仅在于均值的不同，这样的假设将使得对数似然比变为

$$\log\frac{p(\boldsymbol{x}\mid y=1)}{p(\boldsymbol{x}\mid y=0)} = \frac{1}{2}\left\|\boldsymbol{x}-\boldsymbol{\mu_0}\right\|^2_{\boldsymbol{\Sigma^{-1}}} - \frac{1}{2}\left\|\boldsymbol{x}-\boldsymbol{\mu_1}\right\|^2_{\boldsymbol{\Sigma^{-1}}}$$

其中 $\boldsymbol{\Sigma}$ 可以通过对每个类的训练数据去均值后再合起来计算协方差实现，这样估计得到的协方差矩阵依然是无偏的。这里利用 $\boldsymbol{\Sigma^{-1}}$ 对称性，考虑平方差公式的结构

$$(x-a)^2-(x-b)^2 = (b-a)(2x-a-b) = 2(b-a)x-(b-a)(a+b)$$

从而得到[9]

$$\begin{aligned}\log\frac{p(\boldsymbol{x}\mid y=1)}{p(\boldsymbol{x}\mid y=0)} &= (\boldsymbol{\mu_1}-\boldsymbol{\mu_0})^T\boldsymbol{\Sigma^{-1}}\left(\boldsymbol{x}-\frac{1}{2}(\boldsymbol{\mu_1}+\boldsymbol{\mu_0})\right)\\ &= (\boldsymbol{\mu_1}-\boldsymbol{\mu_0})^T\boldsymbol{\Sigma^{-1}}\boldsymbol{x} - \frac{1}{2}(\boldsymbol{\mu_1}-\boldsymbol{\mu_0})^T\boldsymbol{\Sigma^{-1}}(\boldsymbol{\mu_1}+\boldsymbol{\mu_0})\end{aligned}$$

记

$$\begin{aligned}\boldsymbol{w} &:= \boldsymbol{\Sigma^{-1}}(\boldsymbol{\mu_1}-\boldsymbol{\mu_0})\\ c &:= \langle \boldsymbol{w}, \frac{\boldsymbol{\mu_1}+\boldsymbol{\mu_0}}{2}\rangle\end{aligned}$$

得到分类对应的映射函数

$$f(\boldsymbol{x}) = \mathbf{1}_{\log\frac{p(\boldsymbol{x}\mid y=1)}{p(\boldsymbol{x}\mid y=0)}>0} = \mathbf{1}_{\langle \boldsymbol{w},\boldsymbol{x}\rangle > c}$$

此时决策边界为 $\boldsymbol{w^T x} = c$，$\boldsymbol{w}$ 充当了平面的法向量，跨过平面将发生 $\boldsymbol{w^T x}$ 和 c 的大小关系符反转，数据归属的类别的判定将发生变化。

[9]实际上我们考虑的是一个同构映射

我们也可以从投影的视角看 LDA 的过程，样本 $\boldsymbol{x}$ 首先进行了一次均值归一化操作，后沿着 $\boldsymbol{w}$ 方向进行投影（当然，由于是内积运算还多个归一化操作）得到一维数据。由这个思想产生的 LDA 降维在分类效果好时能够使得降维得到两类的一维数据以 0 为分界点被很好地分开，这是一种有监督的降维过程。下面部分我们会讲一讲当两个类协方差矩阵实际上存在差异时我们该如何寻找最优的 $\boldsymbol{w}$。

11.4.2　广义 LDA [10]

广义 LDA（Generalized linear discriminant analysis）考虑两个类协方差矩阵实际上存在差异的情况。我们尝试计算仅使用 $\boldsymbol{w}$ 进行投影得到的数据的期望和方差，这里要用到随机变量的期望和方差的运算性质。设 $\boldsymbol{x}$ 为随机变量 $(X_1, X_2, \cdots, X_D)^T$，则期望有

$$\mathrm{E}[\boldsymbol{w^T x}] = \mathrm{E}[(X_1, X_2, \cdots, X_D)\boldsymbol{w}] = \sum_{n=1}^{D} w_n \mathrm{E}[X_n] = \boldsymbol{w^T}\mathrm{E}[\boldsymbol{x}]$$

借助期望的线性性质，可以计算方差（协方差）有

$$\mathrm{Var}[\boldsymbol{w^T x}] = \mathrm{Cov}[\boldsymbol{w^T x}, \boldsymbol{w^T x}] = \sum_{n=1}^{D}\sum_{m=1}^{D} \mathrm{Cov}[X_n, X_m] w_n w_m = \boldsymbol{w^T}\mathrm{Cov}[\boldsymbol{x}, \boldsymbol{x}]\boldsymbol{w}$$

对于第 0 类，投影后的期望和方差为

$$\mathrm{E}[\boldsymbol{w^T x_0}] = \boldsymbol{w^T \mu_0}$$
$$\mathrm{Var}[\boldsymbol{w^T x_0}] = \boldsymbol{w^T \Sigma_0 w}$$

对于第 1 类，结果也是类似的

$$\mathrm{E}[\boldsymbol{w^T x_1}] = \boldsymbol{w^T \mu_1}$$
$$\mathrm{Var}[\boldsymbol{w^T x_1}] = \boldsymbol{w^T \Sigma_1 w}$$

由此定义类内散度（Within-class variance），其衡量了类内数据的方差

$$\sigma^2_{\text{within}} = \mathrm{Var}[\boldsymbol{w^T x_0}] + \mathrm{Var}[\boldsymbol{w^T x_1}] = \boldsymbol{w^T \Sigma_0 w} + \boldsymbol{w^T \Sigma_1 w} = \boldsymbol{w^T}(\boldsymbol{\Sigma_0} + \boldsymbol{\Sigma_1})\boldsymbol{w}$$

[10] 课件 lecture11 pp.11-12

其中

$$\boldsymbol{\Sigma}_{\text{within}} = \boldsymbol{\Sigma_0} + \boldsymbol{\Sigma_1}$$

为类内散度矩阵。定义类间散度（Between-class variance），其衡量了类间的期望的差异

$$\sigma^2_{\text{between}} = \frac{1}{2}\sum_{n=0}^{1}(\mathrm{E}[\boldsymbol{w^T x_n}] - \boldsymbol{\mu})^T(\mathrm{E}[\boldsymbol{w^T x_n}] - \boldsymbol{\mu}),\ \boldsymbol{\mu} = \frac{1}{2}(\boldsymbol{\mu_0} + \boldsymbol{\mu_1})$$

得到

$$\sigma^2_{\text{between}} = \frac{1}{4}\boldsymbol{w^T}(\boldsymbol{\mu_1} - \boldsymbol{\mu_0})(\boldsymbol{\mu_1} - \boldsymbol{\mu_0})^T\boldsymbol{w}$$

其中

$$\boldsymbol{\Sigma}_{\text{between}} = \frac{1}{2}\sum_{n=0}^{1}(\boldsymbol{\mu_n} - \boldsymbol{\mu})^T(\boldsymbol{\mu_n} - \boldsymbol{\mu}) = \frac{1}{4}(\boldsymbol{\mu_1} - \boldsymbol{\mu_0})(\boldsymbol{\mu_1} - \boldsymbol{\mu_0})^T$$

为类间散度矩阵。忽略系数后可以定义两个分布的 Fisher’s separation

$$S(\boldsymbol{w}) := \frac{\sigma^2_{\text{between}}}{\sigma^2_{\text{within}}} = \frac{\boldsymbol{w^T}\boldsymbol{\Sigma}_{\text{between}}\boldsymbol{w}}{\boldsymbol{w^T}\boldsymbol{\Sigma}_{\text{within}}\boldsymbol{w}} = \frac{(\boldsymbol{w^T}(\boldsymbol{\mu_1} - \boldsymbol{\mu_0}))^2}{\boldsymbol{w^T}(\boldsymbol{\Sigma_0} + \boldsymbol{\Sigma_1})\boldsymbol{w}}$$

我们的优化目标是使得 S 最大化，即最大化类间差异，最小化类内差异。这个形式其实是广义 Rayleigh 商，我们已经证明了 Rayleigh 商的值介于实对称矩阵 $\boldsymbol{A}$ 的最大特征值和最小特征值之间，且仅当 $\boldsymbol{x}$ 取最大（小）特征值对应的特征向量时 Rayleigh 商取到最大（小）值[11]。广义 Rayleigh 商（Generalized Rayleigh quotient）的形式为

$$R(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{x}) = \frac{\boldsymbol{x^T A x}}{\boldsymbol{x^T B x}}$$

这里要求 $\boldsymbol{B}$ 是对称且正定的，正定是为了保证分母不会取 0。Rayleigh 商函数的最值可以很容易地在划定约束条件 $\boldsymbol{x^T B x} = 1$ 后利用 Lagrange 乘数得到，因为 $\boldsymbol{x}$ 的 scaling 不会改变 Rayleigh 函数的取值 [12]，我们实际上可

[11] 详见 D.2.3 结尾部分

[12] https://www.zybuluo.com/w460461339/note/1261090

以任意地缩放不满足条件的 $\boldsymbol{x}$ 使得约束条件成立而不改变 Rayleigh 商。即此时问题变为

$$\max_{\boldsymbol{x}}(\min_{\boldsymbol{x}})R(\boldsymbol{A},\boldsymbol{B},\boldsymbol{x}) \Rightarrow \max_{\boldsymbol{x}^{\boldsymbol{T}}\boldsymbol{B}\boldsymbol{x}=1}(\min_{\boldsymbol{x}^{\boldsymbol{T}}\boldsymbol{B}\boldsymbol{x}=1})\boldsymbol{x}^{\boldsymbol{T}}\boldsymbol{A}\boldsymbol{x}$$

构造 Lagrangian 函数

$$L(\boldsymbol{x},\lambda) = \boldsymbol{x}^{\boldsymbol{T}}\boldsymbol{A}\boldsymbol{x} + \lambda(1 - \boldsymbol{x}^{\boldsymbol{T}}\boldsymbol{B}\boldsymbol{x})$$

求导并令导数为 0 得到

$$\frac{\partial L}{\partial \boldsymbol{x}} = 2\boldsymbol{A}\boldsymbol{x} - 2\lambda\boldsymbol{B}\boldsymbol{x} = 0 \Rightarrow \boldsymbol{A}\boldsymbol{x} = \lambda\boldsymbol{B}\boldsymbol{x} \Rightarrow \boldsymbol{B}^{-\boldsymbol{1}}\boldsymbol{A}\boldsymbol{x} = \lambda\boldsymbol{x}$$

这表明极值总是在 $\boldsymbol{B}^{-\boldsymbol{1}}\boldsymbol{A}$ 特征向量处取到，其对应的 R 的值为

$$R(\boldsymbol{A},\boldsymbol{B},\boldsymbol{x}) = \frac{\boldsymbol{x}^{\boldsymbol{T}}\boldsymbol{A}\boldsymbol{x}}{\boldsymbol{x}^{\boldsymbol{T}}\boldsymbol{B}\boldsymbol{x}} = \lambda$$

因而 R 取得最大（小）值当且仅当 $\boldsymbol{x}$ 取最大（小）特征值对应的特征向量。

回到 Fisher’s separation，我们得到

$$S(\boldsymbol{w}) = R(\boldsymbol{\Sigma}_{\text{between}}, \boldsymbol{\Sigma}_{\text{within}}, \boldsymbol{w})$$

其中 $\boldsymbol{w}$ 的最优解实际上是 $\boldsymbol{\Sigma}_{\text{between}}^{-1}\boldsymbol{\Sigma}_{\text{within}}$ 的最大特征值对应的特征向量。考虑展开 $\boldsymbol{\Sigma}_{\text{within}}$ 的表达式[12] $\boldsymbol{w}$ 的最优解满足

$$\boldsymbol{\Sigma}_{\text{between}}^{-1}\boldsymbol{\Sigma}_{\text{within}}\boldsymbol{w} = \boldsymbol{\Sigma}_{\text{between}}^{-1}(\boldsymbol{\mu_1} - \boldsymbol{\mu_0})(\boldsymbol{\mu_1} - \boldsymbol{\mu_0})^T\boldsymbol{w} = \lambda\boldsymbol{w}$$

由于 $(\boldsymbol{\mu_1} - \boldsymbol{\mu_0})^T\boldsymbol{w}$ 是一个常数，对 λ 分类讨论。当 $\lambda = 0$ 时表明 $\boldsymbol{w}$ 在 $\boldsymbol{\Sigma}_{\text{between}}^{-1}\boldsymbol{\Sigma}_{\text{within}}$ 的零空间中，否则在 $\boldsymbol{\Sigma}_{\text{between}}^{-1}(\boldsymbol{\mu_1} - \boldsymbol{\mu_0})$ 张成的特征子空间中。假若前者成立，表明

$$\boldsymbol{\Sigma}_{\text{between}}^{-1}\boldsymbol{\Sigma}_{\text{within}}\boldsymbol{w} = \boldsymbol{0} \Rightarrow \boldsymbol{\Sigma}_{\text{within}}\boldsymbol{w} = \boldsymbol{0} \Rightarrow S(\boldsymbol{w}) = R(\boldsymbol{\Sigma}_{\text{between}}, \boldsymbol{\Sigma}_{\text{within}}, \boldsymbol{w}) = 0$$

由于最大化优化问题的限制，前者是不成立的，因为任取不在 $\boldsymbol{\Sigma}_{\text{within}}$ 的零空间的向量，$S(\boldsymbol{w})$ 必然非负，从而该解不是最优解。由于 Rayleigh 商函数只在特征向量处取得最值，因而最优解 $\boldsymbol{w}$ 必然为 $\boldsymbol{\Sigma}_{\text{between}}^{-1}(\boldsymbol{\mu_1} - \boldsymbol{\mu_0})$ 张

成的特征子空间中特征向量，特征向量张成的特征子空间中的向量均为最优解。考虑 LDA 的解为

$$\boldsymbol{w} = \boldsymbol{\Sigma}^{-1}(\boldsymbol{\mu_1} - \boldsymbol{\mu_0})$$

仿照相同的形式，我们取 $\boldsymbol{w}$ 的一个解为

$$\boldsymbol{w} = \left(\frac{1}{2}\boldsymbol{\Sigma_0} + \frac{1}{2}\boldsymbol{\Sigma_1}\right)^{-1}(\boldsymbol{\mu_1} - \boldsymbol{\mu_0})$$

因而 LDA 成为广义 LDA 在取 $\boldsymbol{\Sigma_0} = \boldsymbol{\Sigma_1}$ 时的一个特例，我们从 Fisher's separation 的角度推导重新得到了投影向量 $\boldsymbol{w}$ 的表达式。

没有什么比较好的准则用于确定用于降维后区分两个类的阈值 c，但是仿照 LDA 进行类似平方差公式的展开

$$c = \frac{1}{2}(\boldsymbol{\mu_1} - \boldsymbol{\mu_0})^T\boldsymbol{\Sigma^{-1}}(\boldsymbol{\mu_1} + \boldsymbol{\mu_0}) = \frac{1}{2}\boldsymbol{\mu_1^T}\boldsymbol{\Sigma^{-1}}\boldsymbol{\mu_1} - \frac{1}{2}\boldsymbol{\mu_1^T}\boldsymbol{\Sigma^{-1}}\boldsymbol{\mu_1}$$

在依然假设两个类的分布是相近的时候，我们可以选择[13]

$$\begin{aligned} c &= \langle \boldsymbol{w}, \frac{\boldsymbol{\mu_1} + \boldsymbol{\mu_0}}{2} \rangle = \frac{1}{2}(\boldsymbol{\mu_1} + \boldsymbol{\mu_0})^T\left(\frac{1}{2}\boldsymbol{\Sigma_0} + \frac{1}{2}\boldsymbol{\Sigma_1}\right)^{-1}(\boldsymbol{\mu_1} - \boldsymbol{\mu_0}) \\ &= \frac{1}{2}\boldsymbol{\mu_1^T}\left(\frac{1}{2}\boldsymbol{\Sigma_0} + \frac{1}{2}\boldsymbol{\Sigma_1}\right)^{-1}\boldsymbol{\mu_1} - \frac{1}{2}\boldsymbol{\mu_0^T}\left(\frac{1}{2}\boldsymbol{\Sigma_0} + \frac{1}{2}\boldsymbol{\Sigma_1}\right)^{-1}\boldsymbol{\mu_0} \end{aligned}$$

我们也可以进一步地选择

$$c = \frac{1}{2}\boldsymbol{\mu_1^T}\boldsymbol{\Sigma_1^{-1}}\boldsymbol{\mu_1} - \frac{1}{2}\boldsymbol{\mu_0^T}\boldsymbol{\Sigma_0^{-1}}\boldsymbol{\mu_0}$$

11.4.3　多分类 LDA [14]

对于多分类 LDA （Multi-class LDA）我们需要拓展我们的 Fisher's separation，定义类间散度矩阵

$$\boldsymbol{\Sigma}_{\text{between}} = \frac{1}{C}\sum_{n=1}^{C}(\boldsymbol{\mu_n} - \boldsymbol{\mu})^T(\boldsymbol{\mu_n} - \boldsymbol{\mu}),\ \boldsymbol{\mu} = \frac{1}{C}\sum_{n=1}^{C}\boldsymbol{\mu_n}$$

[13] https://en.wikipedia.org/wiki/Linear_discriminant_analysis

[14] 课件 lecture11 pp.13-14

依然假设所有类别的数据的协方差是相同的，将所有类去均值合起来计算协方差矩阵，得到类间矩阵为 $\boldsymbol{\Sigma}_{\text{within}}$。此时我们的 Fisher’s separation 形式几乎完全不变

$$S(\boldsymbol{w}) := R(\boldsymbol{\Sigma}_{\text{between}}, \boldsymbol{\Sigma}_{\text{within}}, \boldsymbol{w})$$

和二分类的 LDA 类似地，多分类 LDA 可以利用投影的思想构造线性映射，用于数据的有监督降维，我们事实上需要寻找一个投影面使得在每个轴上计算得到的 S 都尽可能地大，这使得我们联想到 PCA 的过程，且和 PCA 类似地，核技巧也能应用于 LDA [15]。有关这部分有关 LDA 用于有监督降维的内容本书不进行拓展，请大家自行了解。

11.5　基于 GLM 的分类模型[16]

11.5.1　Logit 回归模型 [17]

由 GLM [18] 的范式对 Y 的条件概率进行建模，选取分布函数为简单的 Bernouli 分布，即

$$\begin{aligned} y &\sim \text{Ber}(p) \\ p(y;\theta,\tau) = p^y(1-p)^{1-y} &= \exp(y\ln p + (1-y)\ln(1-p)) \\ &= \exp\left(y\ln\frac{p}{1-p} + \ln(1-p)\right) \end{aligned}$$

从而我们确定了 GLM 的组成部分

$$\begin{aligned} \theta &= \ln\frac{p}{1-p} \\ A(\theta) &= \ln(1-p) \\ d(\tau) &= 1 \\ h(y,\tau) &= 1 \end{aligned}$$

化简得到

[15] https://zhuanlan.zhihu.com/p/92359921

[16] 课件 lecture11 pp.15-17

[17] 课件 lecture11 pp.15-16

[18] 详见 4.1

$$p = \frac{1}{1+\exp(-\theta)} =: \text{sigmoid}(\theta)$$

采用标准连接函数

$$\theta = \eta = \boldsymbol{x}^{\boldsymbol{T}}\boldsymbol{w}$$

得到

$$p(y \mid \boldsymbol{x}; \boldsymbol{w}) = p^y(1-p)^{1-y}$$

其中

$$p = \text{sigmoid}(\boldsymbol{w}^{\boldsymbol{T}}\boldsymbol{x})$$

由此我们得到了 Logit 回归（Logit regression）模型。Logit 回归显然是一个经典的鉴别模型。

考虑 MLE，计算对数似然函数得到

$$\begin{aligned}\max_{\boldsymbol{w}} \log L(\boldsymbol{X}, \boldsymbol{y}; \boldsymbol{w}) &= \max_{\boldsymbol{w}} \log p(\boldsymbol{X}, \boldsymbol{y}; \boldsymbol{w}) = \max_{\boldsymbol{w}} \log p(\boldsymbol{y}; \boldsymbol{w}) + \log p(\boldsymbol{X} \mid \boldsymbol{y}; \boldsymbol{w}) \\ &= \sum_{n=1}^{N} \log p(y_n \mid \boldsymbol{x_n}; \boldsymbol{w}) = \max_{\boldsymbol{w}} \sum_{n=1}^{N} (y_n \log p_n + (1-y_n)\log(1-p_n)) \\ &= \min_{\boldsymbol{w}} \sum_{n=1}^{N} H(\text{Ber}(y_n), \text{Ber}(p_n))\end{aligned}$$

其中

$$p_n = \text{sigmoid}(\boldsymbol{w}^{\boldsymbol{T}}\boldsymbol{x_n})$$

以上推导中出现了二分类中标签分布和拟合分布的交叉熵损失 [19]。上述公式闭式解很难求解，我们考虑梯度下降法，取对数为自然对数，设损失总和为

$$L = \sum_{n=1}^{N} H(\text{Ber}(y_n), \text{Ber}(p_n))$$

得到

[19]详见 F.3

$$\begin{aligned}\frac{\partial L}{\partial \boldsymbol{w}} &= \frac{\partial \boldsymbol{X}^{\boldsymbol{T}}\boldsymbol{w}}{\partial \boldsymbol{w}}\frac{\partial L}{\partial \boldsymbol{X}^{\boldsymbol{T}}\boldsymbol{w}} = \boldsymbol{X}\left(\frac{\partial L}{\partial \boldsymbol{x}_{\boldsymbol{i}}^{\boldsymbol{T}}\boldsymbol{w}}\right)_{N\times 1} = \boldsymbol{X}\left(\frac{\partial L}{\partial p_i}\frac{\partial p_i}{\partial \boldsymbol{x}_{\boldsymbol{i}}^{\boldsymbol{T}}\boldsymbol{w}}\right)_{N\times 1} \\ &= -\boldsymbol{X}\left(\left(\frac{y_i}{p_i}-\frac{1-y_i}{1-p_i}\right)\frac{\exp(-\boldsymbol{x}_{\boldsymbol{i}}^{\boldsymbol{T}}\boldsymbol{w})}{(1+\exp(-\boldsymbol{x}_{\boldsymbol{i}}^{\boldsymbol{T}}\boldsymbol{w}))^2}\right)_{N\times 1} \\ &= \boldsymbol{X}\left(\left(\frac{1-y_i}{1-p_i}-\frac{y_i}{p_i}\right)p_i(1-p_i)\right)_{N\times 1} = \boldsymbol{X}\left(p_i-y_i\right)_{N\times 1} \\ &= \boldsymbol{X}\left(\boldsymbol{p}-\boldsymbol{y}\right)\end{aligned}$$

得到

$$\boldsymbol{w}^{(\boldsymbol{t+1})} := \boldsymbol{w}^{(\boldsymbol{t})} - \frac{\tau}{N}\boldsymbol{X}\left(\boldsymbol{p}-\boldsymbol{y}\right)$$

从这里我们可以看出令上式为 0 并求解对应的 $\boldsymbol{w}$ 是困难的。对于二分类而言其数据到标签的映射为

$$f(\boldsymbol{x}) = \arg\max_y p(y \mid \boldsymbol{x};\boldsymbol{w}) \Rightarrow f(\boldsymbol{x}) = \mathbf{1}_{\text{sigmoid}(\boldsymbol{x}^{\boldsymbol{T}}\boldsymbol{w})>1/2} = \mathbf{1}_{\boldsymbol{x}^{\boldsymbol{T}}\boldsymbol{w}>0}$$

我们得到分类器的决策边界为

$$\boldsymbol{x}^{\boldsymbol{T}}\boldsymbol{w} = 0$$

函数

$$h(\boldsymbol{x}) = \begin{cases} 1 & \boldsymbol{x} > 0 \\ 0 & \boldsymbol{x} \leqslant 0 \end{cases}$$

是阶跃函数（Step function）[20]，该函数也可以作为 $\mathbb{R}$ 到概率的一个映射。我们也可以将学习到的映射解读为

$$f(\boldsymbol{x}) = \mathbf{1}_{h(\boldsymbol{x}^{\boldsymbol{T}}\boldsymbol{w})>1/2}$$

然而在优化中选择 sigmoid 去代替 h 的一个原因就是 sigmoid 具有可微的性质，相当于对 h 进行了一次平滑处理，并且在边界处 sigmoid 也是比较明确的，因为指数决定了其收敛得很快。

[20] https://en.wikipedia.org/wiki/Step_function

11.5.2　Softmax 回归模型 [21]

仍然考虑 GLM，将标签的条件分布从 Bernouli 分布更换为类别分布得到

$$\boldsymbol{y} \sim \mathrm{Cat}(p_1, p_2, \cdots, p_C)$$

上式隐含了归一化的约束条件

$$\sum_{c=1}^{C} p_c = 1$$

该式表明 D 维参数存在冗余，我们实际上只需使用 $\boldsymbol{y}$ 和 $\boldsymbol{p}$ 的 $C-1$ 个维度就可以表示参数的所有信息。我们会通过改写分布的形式来消除这个约束[22]

$$\begin{aligned} p(\boldsymbol{y};\boldsymbol{\theta},\tau) &= \prod_{c=1}^{C} p_c^{y_c} = \exp\left(\sum_{c=1}^{C} y_c \ln p_c\right) \\ &= \exp\left(\left(1-\sum_{c=1}^{C-1} y_c\right)\ln\left(1-\sum_{c=1}^{C-1} p_c\right) + \sum_{c=1}^{C-1} y_c \ln p_c\right) \\ &= \exp\left(\sum_{c=1}^{C-1} y_c \ln \frac{p_c}{1-\sum\limits_{i=1}^{C-1} p_i} + \ln\left(1-\sum_{c=1}^{C-1} p_c\right)\right) \end{aligned}$$

从而我们确定了 GLM 的组成部分

$$\theta_c = \ln \frac{p_c}{1-\sum\limits_{c=1}^{C-1} p_c} = \ln \frac{p_c}{p_C},\ c = 1, 2, \cdots, C-1$$

$$A(\boldsymbol{\theta}) = \ln\left(1-\sum_{c=1}^{C-1} p_c\right)$$

$$d(\tau) = 1$$

$$h(y,\tau) = 1$$

上述定义可以进行扩充，当 $d = C$ 时有

[21] 课件 lecture11 p.17

[22] https://blog.csdn.net/cdd2xd/article/details/75635688

$$\theta_C := \ln \frac{p_c}{p_C} = 0$$

化简得到

$$\sum_{i \neq c}^{C-1} p_i \exp(\theta_c) + (\exp(\theta_c) + 1) p_c = \exp(\theta_c),\ c = 1, 2, \cdots, C-1$$

这事实上是一个方程组

$$\boldsymbol{A}\boldsymbol{p} = \boldsymbol{b}$$

其中

$$\boldsymbol{A} = \begin{pmatrix} \exp(\theta_1) + 1 & \exp(\theta_1) & \cdots & \exp(\theta_1) \\ \exp(\theta_2) & \exp(\theta_2) + 1 & \cdots & \exp(\theta_2) \\ \vdots & \vdots & \ddots & \vdots \\ \exp(\theta_{C-1}) & \exp(\theta_{C-1}) & \cdots & \exp(\theta_{C-1}) + 1 \end{pmatrix}$$

$$\boldsymbol{b} = (\exp(\theta_1), \exp(\theta_2), \cdots, \exp(\theta_{C-1}))^T$$

利用 Cramer 法则解方程组，我们首先求解 $\boldsymbol{A}$ 的行列式 D'，得到

$$\begin{aligned} D' = \det(\boldsymbol{A}) &= \begin{vmatrix} 1 & 1 & 1 & \cdots & 1 \\ & \exp(\theta_1) + 1 & \exp(\theta_1) & \cdots & \exp(\theta_1) \\ & \exp(\theta_2) & \exp(\theta_2) + 1 & \cdots & \exp(\theta_2) \\ & \vdots & \vdots & \ddots & \vdots \\ & \exp(\theta_{C-1}) & \exp(\theta_{C-1}) & \cdots & \exp(\theta_{C-1}) + 1 \end{vmatrix} \\ &= \begin{vmatrix} 1 & 1 & 1 & \cdots & 1 \\ -\exp(\theta_1) & 1 & & & \\ -\exp(\theta_2) & & 1 & & \\ \vdots & & & \ddots & \\ -\exp(\theta_{C-1}) & & & & 1 \end{vmatrix} = \begin{vmatrix} 1 + \sum_{c=1}^{C-1} \exp(\theta_c) & 0 & 0 & \cdots & 0 \\ -\exp(\theta_1) & 1 & & & \\ -\exp(\theta_2) & & 1 & & \\ \vdots & & & \ddots & \\ -\exp(\theta_{C-1}) & & & & 1 \end{vmatrix} \\ &= 1 + \sum_{c=1}^{C-1} \exp(\theta_c) \end{aligned}$$

再考虑求解将 $\boldsymbol{A}$ 的第 i 列替换为 $\boldsymbol{b}$ 后得到的行列式 D_i'

$$
\begin{aligned}
D_i' &= \begin{vmatrix}
1 & 1 & 1 & \cdots & 1 & \cdots & 1 \\
 & \exp(\theta_1)+1 & \exp(\theta_1) & \cdots & \exp(\theta_1) & \cdots & \exp(\theta_1) \\
 & \exp(\theta_2) & \exp(\theta_2)+1 & \cdots & \exp(\theta_2) & \cdots & \exp(\theta_2) \\
 & \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\
 & \exp(\theta_i) & \exp(\theta_i) & \cdots & \exp(\theta_i) & \cdots & \exp(\theta_i) \\
 & \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\
 & \exp(\theta_{C-1}) & \exp(\theta_{C-1}) & \cdots & \exp(\theta_{C-1}) & \cdots & \exp(\theta_{C-1})+1
\end{vmatrix} \\
&= \begin{vmatrix}
1 & 1 & 1 & \cdots & 1 & \cdots & 1 \\
-\exp(\theta_1) & 1 & & & & & \\
-\exp(\theta_2) & & 1 & & & & \\
\vdots & & & \ddots & & & \\
-\exp(\theta_i) & & & & 0 & & \\
\vdots & & & & & \ddots & \\
-\exp(\theta_{C-1}) & & & & & & 1
\end{vmatrix}
\end{aligned}
$$

考虑对 D' 按第 $i+1$ 行展开得到

$$
\begin{vmatrix}
1 & 1 & 1 & \cdots & 1 \\
-\exp(\theta_1) & 1 & & & \\
-\exp(\theta_2) & & 1 & & \\
\vdots & & & \ddots & \\
-\exp(\theta_{C-1}) & & & & 1
\end{vmatrix}
= D_i' +
\begin{vmatrix}
1 & 1 & 1 & \cdots & 1 & 1 & \cdots & 1 \\
-\exp(\theta_1) & 1 & & & & & & \\
-\exp(\theta_2) & & 1 & & & & & \\
\vdots & & & \ddots & & & & \\
-\exp(\theta_{i-1}) & & & & 1 & & & \\
-\exp(\theta_{i+1}) & & & & & 1 & & \\
\vdots & & & & & & \ddots & \\
-\exp(\theta_{C-1}) & & & & & & & 1
\end{vmatrix}
$$

右侧第二项对应的行列式实际上就是 D' 抽取第 i 行和第 i 列后得到的行列式，这个形式和 D' 是一样的，因而考虑变量代换后我们也可以将其求解出。最终得到

$$
D_i' = 1 + \sum_{c=1}^{C-1} \exp(\theta_c) - \left(1 + \sum_{c \neq i} \exp(\theta_c)\right) = \exp(\theta_i)
$$

从而

$$p_c = \frac{D'_c}{D'} = \frac{\exp(\theta_c)}{1 + \sum\limits_{i=1}^{C-1} \exp(\theta_i)} = \frac{\exp(\theta_c)}{\sum\limits_{i=1}^{C} \exp(\theta_i)} =: \mathrm{softmax}(c, \boldsymbol{\theta})$$

联系 Nadaraya-Watson 估计器，基于 RBF 核的 Nadaraya-Watson 估计器事实上也可以写为 softmax 形式，记

$$\boldsymbol{\theta_h}(\boldsymbol{x}) = \left(-\frac{1}{2} \left\| \frac{\boldsymbol{x} - \boldsymbol{x_i}}{h} \right\|_2^2 \right)_{N \times 1}$$

则

$$\begin{aligned} \hat{y} = m_h(\boldsymbol{x}) &= \frac{\sum\limits_{n=1}^{N} \kappa_h(\boldsymbol{x} - \boldsymbol{x_n}) y_n}{\sum\limits_{n=1}^{N} \kappa_h(\boldsymbol{x} - \boldsymbol{x_n})} = \frac{\sum\limits_{n=1}^{N} \exp(-\frac{1}{2} \left\| \frac{\boldsymbol{x} - \boldsymbol{x_n}}{h} \right\|_2^2) y_n}{\sum\limits_{n=1}^{N} \exp(-\frac{1}{2} \left\| \frac{\boldsymbol{x} - \boldsymbol{x_n}}{h} \right\|_2^2)} \\ &= \sum_{n=1}^{N} \mathrm{softmax}(n, \boldsymbol{\theta_h}(\boldsymbol{x})) y_n \end{aligned}$$

特别的，基于 RBF 的 Mean-shift 也可以写为

$$\begin{aligned} \boldsymbol{x_i^{(t+1)}} := m_h(\boldsymbol{x_i^{(t)}}) &= \frac{\sum\limits_{n=1}^{N} \kappa_h(\boldsymbol{x_i^{(t)}} - \boldsymbol{x_n^{(t)}}) \boldsymbol{x_n^{(t)}}}{\sum\limits_{n=1}^{N} \kappa_h(\boldsymbol{x_i^{(t)}} - \boldsymbol{x_n^{(t)}})} \\ &= \sum_{n=1}^{N} \mathrm{softmax}(n, \boldsymbol{\theta_h}(\boldsymbol{x})) \boldsymbol{x_n^{(t)}} \end{aligned}$$

采用标准连接函数

$$\theta_c = \eta = \boldsymbol{x^T w_c},\ c = 1, 2, \cdots, C-1$$

$$\theta_D = 0 = \boldsymbol{x^T w_c} \Rightarrow \boldsymbol{w_C} = \boldsymbol{0_C}$$

立即得到

$$p_c = \mathrm{softmax}(c, \boldsymbol{W^T x}) = \frac{\exp(\boldsymbol{x^T w_c})}{\sum\limits_{i=1}^{C} \exp(\boldsymbol{x^T w_i})},\ c = 1, 2, \cdots, C-1$$

softmax 具有平移不变性，即

$$\begin{aligned}\operatorname{softmax}(c, \boldsymbol{W}^{\boldsymbol{T}}\boldsymbol{x} + b\mathbf{1}_{\boldsymbol{C}}) &= \frac{\exp(\boldsymbol{x}^{\boldsymbol{T}}\boldsymbol{w}_{\boldsymbol{c}} + b)}{\sum_{i=1}^{C}\exp(\boldsymbol{x}^{\boldsymbol{T}}\boldsymbol{w}_{\boldsymbol{i}} + b)} = \frac{\exp(\boldsymbol{x}^{\boldsymbol{T}}\boldsymbol{w}_{\boldsymbol{c}})}{\sum_{i=1}^{C}\exp(\boldsymbol{x}^{\boldsymbol{T}}\boldsymbol{w}_{\boldsymbol{i}})} \\ &= \operatorname{softmax}(c, \boldsymbol{W}^{\boldsymbol{T}}\boldsymbol{x})\end{aligned}$$

这决定了我们事实上可以将 θ_D 也用一个 $\boldsymbol{x}$ 的有效的线性组合去定义，设为 $\boldsymbol{x}^{\boldsymbol{T}}\boldsymbol{w}_{\boldsymbol{C}}$，我们这样的扩充是合理的，因为

$$\theta_D = \frac{\exp(0)}{1 + \sum_{i=1}^{C-1}\exp(\boldsymbol{x}^{\boldsymbol{T}}\boldsymbol{w}_{\boldsymbol{i}})} = \frac{\exp(\boldsymbol{x}^{\boldsymbol{T}}\boldsymbol{w}_{\boldsymbol{C}})}{\exp(\boldsymbol{x}^{\boldsymbol{T}}\boldsymbol{w}_{\boldsymbol{C}}) + \sum_{i=1}^{C-1}\exp(\boldsymbol{x}^{\boldsymbol{T}}(\boldsymbol{w}_{\boldsymbol{i}} + \boldsymbol{w}_{\boldsymbol{C}}))}$$

上面的式子依然保持各个 θ_c 和 $\boldsymbol{x}$ 的线性关系不变，这个操作使得式子的自由度增加了 1，我们通过重新定义在保持数值不变的情况下照顾了 θ_c 的对称性。得到

$$p(\boldsymbol{y} \mid \boldsymbol{X}; \boldsymbol{w}) = \prod_{c=1}^{C} p_c^{y_c}$$

其中

$$p_c = \operatorname{softmax}(c, \boldsymbol{W}^{\boldsymbol{T}}\boldsymbol{x})$$

考虑 MLE，计算对数似然函数的得到

$$\begin{aligned}\max_{\boldsymbol{W}} L(\boldsymbol{X}, \boldsymbol{Y}; \boldsymbol{W}) = \max_{\boldsymbol{W}} p(\boldsymbol{Y} \mid \boldsymbol{X}; \boldsymbol{W}) &= \max_{\boldsymbol{W}} \sum_{n=1}^{N} \log p(\boldsymbol{y}_{\boldsymbol{n}} \mid \boldsymbol{X}; \boldsymbol{W}) \\ = \max_{\boldsymbol{W}} \sum_{n=1}^{N}\sum_{c=1}^{C} y_{cn} \log p_{cn} &= \min_{\boldsymbol{W}} \sum_{n=1}^{N} H(\operatorname{Cat}(\boldsymbol{y}_{\boldsymbol{n}}), \operatorname{Cat}(\boldsymbol{p}_{\boldsymbol{n}}))\end{aligned}$$

其中

$$p_{cn} = \operatorname{softmax}(c, \boldsymbol{W}^{\boldsymbol{T}}\boldsymbol{x}_{\boldsymbol{n}})$$

以上推导中二分类的交叉熵损失变为了多分类的交叉熵损失 [19]。上述公式闭式解就更麻烦了，我们还是考虑梯度下降法，取对数为自然对数，设损失总和为

$$L = \min_{\boldsymbol{W}} \sum_{n=1}^{N} H(\mathrm{Cat}(\boldsymbol{y_n}), \mathrm{Cat}(\boldsymbol{p_n}))$$

先考虑对 $\boldsymbol{w_k}$ 进行求导

$$\begin{aligned}
\frac{\partial L}{\partial \boldsymbol{w_k}} &= \frac{\partial \boldsymbol{X^T w_k}}{\partial \boldsymbol{w_k}} \frac{\partial L}{\partial \boldsymbol{X^T w_k}} = \boldsymbol{X} \left(\frac{\partial L}{\partial \boldsymbol{x_i^T w_k}} \right)_{N\times 1} \\
&= -\boldsymbol{X} \left(\frac{\partial}{\partial \boldsymbol{x_i^T w_k}} \sum_{n=1}^{N} \sum_{c=1}^{C} y_{cn} \log p_{cn} \right)_{N\times 1} = -\boldsymbol{X} \left(\frac{\partial}{\partial \boldsymbol{x_i^T w_k}} \sum_{c=1}^{C} y_{ci} \log p_{ci} \right)_{N\times 1} \\
&= -\boldsymbol{X} \left(\sum_{c=1}^{C} y_{ci} \frac{\partial}{\partial \boldsymbol{x_i^T w_k}} \log p_{ci} \right)_{N\times 1} = -\boldsymbol{X} \left(\sum_{c=1}^{C} \frac{y_{ci}}{p_{ci}} \frac{\partial p_{ci}}{\partial \boldsymbol{x_i^T w_k}} \right)_{N\times 1}
\end{aligned}$$

当 $c = k$ 时

$$\begin{aligned}
\frac{\partial p_{ki}}{\partial \boldsymbol{x_i^T w_k}} &= \frac{\partial p_{ki}}{\partial \boldsymbol{x_i^T w_k}} \frac{\exp(\boldsymbol{x^T w_k})}{\sum\limits_{i=1}^{C} \exp(\boldsymbol{x^T w_i})} \\
&= \frac{\exp(\boldsymbol{x^T w_k}) \sum\limits_{i=1}^{C} \exp(\boldsymbol{x^T w_i}) - \exp^2(\boldsymbol{x^T w_k})}{\left(\sum\limits_{i=1}^{C} \exp(\boldsymbol{x^T w_i}) \right)^2} \\
&= \frac{\exp(\boldsymbol{x^T w_k}) \sum\limits_{i\neq k}^{C} \exp(\boldsymbol{x^T w_i})}{\left(\sum\limits_{i=1}^{C} \exp(\boldsymbol{x^T w_i}) \right)^2} = p_{ki}(1 - p_{ki})
\end{aligned}$$

当 $c \neq k$ 时

$$\frac{\partial p_{ci}}{\partial \boldsymbol{x_i^T w_k}} = \frac{\partial p_{ci}}{\partial \boldsymbol{x_i^T w_k}} \frac{\exp(\boldsymbol{x^T w_c})}{\sum\limits_{i=1}^{C} \exp(\boldsymbol{x^T w_i})} = \frac{-\exp^2(\boldsymbol{x^T w_c})}{\left(\sum\limits_{i=1}^{C} \exp(\boldsymbol{x^T w_i}) \right)^2} = -p_{ci}^2$$

代入原式中得到

$$\begin{aligned}
\frac{\partial L}{\partial \boldsymbol{w_k}} &= -\boldsymbol{X} \left(\frac{y_{ki}}{p_{ki}} p_{ki}(1 - p_{ki}) - \sum_{c\neq k} \frac{y_{ci}}{p_{ci}} p_{ci}^2 \right)_{N\times 1} = -\boldsymbol{X} \left(y_{ki} - \sum_{c=1}^{C} y_{ci} p_{ci} \right)_{N\times 1} \\
&= -\boldsymbol{X} \left(y_{ki} - \boldsymbol{p_i^T y_i} \right)_{N\times 1} = \boldsymbol{X} \left((\boldsymbol{P} \odot \boldsymbol{Y})^T \boldsymbol{1_C} - \boldsymbol{Y_k^T} \right)
\end{aligned}$$

其中 $\boldsymbol{p_i^T y_i}$ 代表了数据 $\boldsymbol{x_i}$ 真实标签对应的置信度。得到

$$\frac{\partial L}{\partial \boldsymbol{W}} = \boldsymbol{X}((\boldsymbol{P} \odot \boldsymbol{Y})^T \mathbf{1}_{\boldsymbol{C}\times\boldsymbol{C}} - \boldsymbol{Y^T})$$

得到梯度更新式

$$\boldsymbol{W^{(t+1)}} := \boldsymbol{W^{(t)}} - \frac{\tau}{N}\boldsymbol{X}((\boldsymbol{P} \odot \boldsymbol{Y})^T \mathbf{1}_{\boldsymbol{C}\times\boldsymbol{C}} - \boldsymbol{Y^T})$$

对于多分类而言其数据到标签的映射为

$$\begin{aligned} f(\boldsymbol{x}) &= \arg\max_y p(y \mid \boldsymbol{x}; \boldsymbol{w}) = \arg\max_c \operatorname{softmax}(c, \boldsymbol{W^T x}) \\ &= \arg\max_c \boldsymbol{x^T w_c} \end{aligned}$$

我们得到分类器的一系列决策边界

$$\{\boldsymbol{x^T w_c} = 0 \mid c = \arg\max_k \boldsymbol{x^T w_k}\}_{c=1}^{C}$$

第十二章　SVM

12.1　超平面 [1]

为了引入超平面的概念，首先引入 linear variety，其几何性质非常明显。n 维 Euclidean 空间 E^n 上的一个向量集合 V 被称为 linear variety 当其满足[2]

$$\forall \boldsymbol{x_1}, \boldsymbol{x_2} \in V,\ \forall \lambda \in \mathbb{R},\ \lambda \boldsymbol{x_1} + (1-\lambda)\boldsymbol{x_2} \in V$$

这个概念是从凸集中延伸出去的，因为它取消了凸集 λ 介于 0 到 1 的限制。对于凸集而言以集合内部的点为端点的连线段必在集合内，而对于 linear variety 而言则更进一步地穿过任意两个点的直线必在集合内，这样的限制使得 linear variety 将在凸集的基础上扩张为一个无限大的集合，如果在三维空间上仅仅利用两个向量去尝试构造这样一个 linear variety，我们实际上得到了一个平面。linear variety 和线性空间非常类似，线性空间满足 linear variety 的条件，是一个特殊的 linear variety。下面我们可以简单地证明，穿过原点的 linear variety V 实际上就是一个 E^n 的子空间。考虑子空间需要满足的性质：

向量数乘的封闭性　由于 $\mathbf{0}$ 在 V 内，因而取

$$\forall \boldsymbol{x_1} \in V,\ \forall \lambda \in \mathbb{R},\ \lambda \boldsymbol{x_1} + (1-\lambda)\mathbf{0} = \lambda \boldsymbol{x_1} \in V$$

向量加法的封闭性　取 $\lambda = 1/2$ 结合数乘的封闭性得到

$$\forall \boldsymbol{x_1}, \boldsymbol{x_2} \in V,\ \frac{1}{2}(\boldsymbol{x_1} + \boldsymbol{x_2}) \in V \Rightarrow \boldsymbol{x_1} + \boldsymbol{x_2} \in V$$

[1] 课件 lecture12 pp.4-7

[2] Luenberger, David G., Ye, Yinyu, *Linear and Nonlinear Programming (Fifth Edition)*, p.573

为了使得 V 穿过原点，我们可以以 V 上任意一个向量为偏置并使得 V 上所有的向量向偏置的方向进行移动，从而我们得到了 linear variety 直观的来看是一个仿射空间（Affine space）[3]，即线性空间加上一个固定的偏置。我们定义 linear variety 的维度就是这个线性空间的维度，从而我们给出了超平面的定义：n 维 Euclidean 空间 E^n 上的一个 $n-1$ 维的 linear variety 被称为超平面（Hyperplane）[2]。

超平面实际上就是除了 n 维 Euclidean 空间自身最大的一个 linear variety。我们接下来证明 linear variety 和仿射空间是等价的，尽管这看起来非常直观。对于 n 维 Euclidean 空间 E^n 上的子集 H，H 是超平面当且仅当[4]

$$\exists \boldsymbol{w}, b,\ H = \{\boldsymbol{x} \in E^n \mid \boldsymbol{x}^{\boldsymbol{T}}\boldsymbol{w} = b\}$$

先证必要性。显然 H 是一个 linear variety，因为它符合定义。任取 $\boldsymbol{x_1} \in H$ 并记

$$M := H - \boldsymbol{x_1} = \{\boldsymbol{x} - \boldsymbol{x_1} \mid \boldsymbol{x} \in H\} = \{\boldsymbol{x} \in E^n \mid \boldsymbol{x}^{\boldsymbol{T}}\boldsymbol{w} = b - \boldsymbol{x_1}^{\boldsymbol{T}}\boldsymbol{w}\}$$

且

$$\begin{aligned}\boldsymbol{x_1} - \boldsymbol{x_1} = \boldsymbol{0} \in M \Rightarrow b &= \boldsymbol{x_1}^{\boldsymbol{T}}\boldsymbol{w} \\ \Rightarrow M &= \{\boldsymbol{x} \in E^n \mid \boldsymbol{x}^{\boldsymbol{T}}\boldsymbol{w} = \langle \boldsymbol{w}, \boldsymbol{x}\rangle = 0\} = \operatorname{span}(\boldsymbol{w})^{\perp}\end{aligned}$$

得到 M 实际上是 $\boldsymbol{w}$ 生成的一维子空间的正交补，故 M 的维度为 $n-1$，H 是一个超平面。

再证充分性。对于 linear variety H 而言我们同样任取 $\boldsymbol{x_1} \in H$ 并记

$$M := H - \boldsymbol{x_1} = \{\boldsymbol{x} - \boldsymbol{x_1} \mid \boldsymbol{x} \in H\}$$

由定义 M 是一个 $n-1$ 维的线性子空间，任取其正交补的一个向量 $\boldsymbol{w}$，我们有

$$M = \{\boldsymbol{x} \in E^n \mid \boldsymbol{x}^{\boldsymbol{T}}\boldsymbol{w} = \langle \boldsymbol{w}, \boldsymbol{x}\rangle = 0\}$$

[3]详见 D.1.2

[4]Luenberger, David G., Ye, Yinyu,*Linear and Nonlinear Programming (Fifth Edition)*, p.574

记 $b := \boldsymbol{x_1^T w}$ 得到

$$H = M + \boldsymbol{x_1} = \{\boldsymbol{x} \in E^n \mid (\boldsymbol{x} - \boldsymbol{x_1})^T \boldsymbol{w} = 0\} = \{\boldsymbol{x} \in E^n \mid \boldsymbol{x^T w} = b\}$$

因而我们从几何定义得到了超平面的代数表达式。

对于超平面 $\boldsymbol{x^T w} = b$，$\boldsymbol{w}$ 和超平面对应的线性子空间正交，且方向是唯一的（不考虑方向的正负），称为超平面的法向量。从原点到超平面存在一个偏置向量，通常我们取模长最小的作为偏置向量的代表，考虑射影和 Cauchy 不等式

$$|\boldsymbol{x}|\,|\boldsymbol{w}| \geqslant \langle \boldsymbol{w}, \boldsymbol{x} \rangle \Rightarrow \|\boldsymbol{x}\|_2 \geqslant \frac{\langle \boldsymbol{w}, \boldsymbol{x} \rangle}{|\boldsymbol{w}|} = \frac{\boldsymbol{x^T w}}{|\boldsymbol{w}|} = \frac{b}{\|\boldsymbol{w}\|_2}$$

得到模长取最小时偏置 $\boldsymbol{x}$ 和 $\boldsymbol{w}$ 是共线的，此时取得的偏置向量的长度我们称为超平面到原点的距离，其代表了线性子空间沿着方向向量 $\boldsymbol{w}$ 从原点到超平面需要移动的距离，因为上式实际上计算的是超平面上的向量沿着 $\boldsymbol{w}$ 的射影。

12.2　SVM 简介 [5]

支持向量机（Support vector machine / SVM）的 motivation 在于寻找两个平行的最优的分界面使得样本点分居分界面的外侧，当两个分界面之间的距离最大时我们便找到了这个分界面。沿着 $\boldsymbol{w}$ 以两个相反的方向移动超平面 $\boldsymbol{x^T w} = 0$，我们定义切分两个类的两个平行的分界面

$$H_1 = \{\boldsymbol{x} \in E^n \mid \boldsymbol{x^T w} - b = 1\}$$

$$H_2 = \{\boldsymbol{x} \in E^n \mid \boldsymbol{x^T w} - b = -1\}$$

两个类的数据被这两个分界面分别划分至分界面外侧中的其中一侧。此时我们求解分类问题的核心在于求得最优的分界面，使得分类受噪声影响更小，分类更加稳定。很容易想到我们只需要让这两个分界面之间的距离最大即可。定义超平面的距离为两个超平面上的向量之差的模长的最小值，证明考虑射影和 Cauchy 不等式

[5] 课件 lecture12 pp.8-9

$$\forall \boldsymbol{x} \in H_1, \boldsymbol{y} \in H_2,\ |\boldsymbol{x}-\boldsymbol{y}| \geqslant \frac{\langle \boldsymbol{w}, \boldsymbol{x}-\boldsymbol{y} \rangle}{|\boldsymbol{w}|} = \frac{2}{\|\boldsymbol{w}\|_2}$$

因而我们确定了优化问题为

$$\max_{\boldsymbol{w},b} \frac{2}{\|\boldsymbol{w}\|_2},\ s.t.\ y_n(\boldsymbol{w}^{\boldsymbol{T}}\boldsymbol{x_n} - b) \geqslant 1,\ n = 1, 2, \cdots, N$$

最终判别边界为 $\boldsymbol{x}^{\boldsymbol{T}}\boldsymbol{w} = b$。在这个过程中，我们没有对数据 $\boldsymbol{x}$ 的分布进行任何建模，因而属于鉴别模型。有关 SVM 更加深入的理论推导见附录 H。

12.3　SVR 简介 [6]

支持向量回归（Support vector regression / SVR）是 SVM 分类模型在回归分析中的拓展。SVR 认为数据点不是分布在两个平行的超平面的外侧，而是超平面的内侧，其中 ε 控制了拟合的精度，所有的样本点均需落在超平面的内侧。SVM 的优化目标是最大化一对分隔样本点的超平面之间的距离，使得数据特别是离分界面比较近的极端数据在噪声干扰下分类结果仍然正确；而 SVR 最大化一对在拟合精度容忍范围内包围样本点的超平面之间的距离，使得数据特别是离分界面比较远的极端数据使得在噪声干扰下极端数据仍能满足拟合精度的要求。SVR 同样有 hard-margin 和 soft-margin 两种形式，也有相应的核技巧，其中 soft-margin 这个形式很有意思

$$\min_{\boldsymbol{w},b} \frac{1}{N}\sum_{n=1}^{N}\xi_n + \lambda\|\boldsymbol{w}\|_2^2$$

$$s.t. \begin{cases} \left|y_n - (\boldsymbol{w}^{\boldsymbol{T}}\boldsymbol{x_n} + b)\right| \leqslant \varepsilon + \xi_n \\ \xi_n \geqslant 0 \end{cases},\ n = 1, 2, \cdots, N$$

我们考虑拟合误差基于最小二乘法的岭回归[7]，此处考虑偏置和并将拟合损失除以样本量

$$\min_{\boldsymbol{w},b} \frac{1}{N}\sum_{n=1}^{N}(y_n - (\boldsymbol{w}^{\boldsymbol{T}}\boldsymbol{x_n} + b))^2 + \lambda\|\boldsymbol{w}\|_2^2$$

如果拟合误差基于的是 MAE [8]，则岭回归的优化问题被修改为

[6] 课件 lecture12 p.20

[7] 详见 4.3.1

[8] 详见 3.3

$$\min_{\boldsymbol{w},b} \frac{1}{N} \sum_{n=1}^{N} \left| y_n - (\boldsymbol{w}^{\boldsymbol{T}} \boldsymbol{x}_{\boldsymbol{n}} + b) \right| + \lambda \left\| \boldsymbol{w} \right\|_2^2$$

上式实际上就是 soft-margin 中 SVR 取拟合精度 $\varepsilon = 0$ 的特殊情况。因而相较于“刻板的”岭回归，soft-margin SVR 实际上给出了一个拟合精度的容忍上界 [9]。

SVR 的 hard-margin 和 soft-margin 两种形式求解和 SVM 非常类似，可以参考 SVM 进行相应的理论推导。

[9] https://zhuanlan.zhihu.com/p/50166358

第十三章　决策树与集成学习

13.1　决策树简介 [1]

对于传统的决策树（Decision tree）模型，其数据集分为训练集 $D = \{(\boldsymbol{x_i}, y_i)\}_{i=1}^{N} = (\boldsymbol{X}, \boldsymbol{y})$，属性集 $A = \{a_i\}_{i=1}^{K}$，代表了每个数据 $\boldsymbol{x}$ 的 K 种属性。将数据集中所有数据置于根节点，决策树穷举 A 中所有可取的属性并选取最优的划分方式对训练集进行划分。在划分后训练集中被划分的数据进入对应的子节点，属性集中对应的属性被抽出，以各个子节点数据为起点，重复这个划分过程直至满足结束条件后将节点标注为叶节点[2]。

结束条件分为三种，首先是当前包含的样本全部属于一种类别 c，此时将叶节点标注为 c；其次当前 A 集合为空，已经没有划分的标准可供选择了，或者样本对于 A 中的所有的属性取值相同，属性已经失去了划分数据的能力，此时我们将叶节点标注为 D 中样本数最多的类别，这实际上利用了数据集中标签的分布；最后时当叶节点的数据为空时，此时将叶节点标签标注为父节点的类别，这实际上利用了父节点的标签分布[2]。

ID3 是最简单的决策树模型之一，其选择使用信息增益（Information gain）/互信息来确定最佳的划分标准，对于 A 中某个属性 A，表达式如下

$$G(T, a) = I(T; a) = H(T) - H(T \mid a),\ a \in A$$

其中对于当前节点的数据 $\boldsymbol{X}$

$$H(T) = -\sum_{c=1}^{C} p(y = c) \log p(y = c)$$

表明当前节点按标签分布计算得到的熵，这实际上在划分前是一个固定的值，因而进行最优的划分选择时我们不需要关心。式子

[1] 课件 lecture13 pp.12-16

[2] 周志华，机器学习，清华大学出版社，pp.73-74

$$H(T \mid a) = -\sum_{(c,a^v)\in(\mathcal{C},a)} p(y=c, a=a^v)\log p(y=c \mid a=a^v)$$
$$= -\sum_{a^v\in a} p(a=a^v)\sum_{c=1}^{C} p(y=c \mid a=a^v)\log p(y=c \mid a=a^v)$$

表明当选择属性 a 后计算得到的标签分布的条件熵。

利用信息论视角看决策树，我们会发现决策树最佳的划分的实际上就是最小化在划分确定后子节点熵的期望，这使得我们的划分为我们提供了尽可能多的信息，这样的信息每一步为我们最大地缩减了我们所需要考虑的概率空间的大小（实际上这是一种贪心的策略），使得我们的决策树尽可能地小，算法尽可能快地结束，这是 ID3 的出发[3]。

上述简单的决策模型还面对着如何处理连续的属性值（如通过采样进行离散化），如何处理缺失值（如通过加权），如何进行预剪枝和后剪枝（如通过验证集进行实验）来提高模型的泛化能力和适用范围的问题，因而面对实际应用场景时需要在这个基础上进行改进[4]。

13.2　集成学习简介

集成学习（Ensemble learning）是利用一组个体学习器（Individual learning）（如决策树 C4.5），进行结合以获取比单一学习器更强的集成的学习器。由个体学习器的生成方式，集成学习主要分为两大类，一是个体学习器之间存在强依赖关系，必须串行生成序列的 Boosting，代表为 Adaboost；二是个体学习器之间不存在强依赖关系，可以并行地生成序列的 Bagging，代表为随机森林（Random forest）。前者更加关注模型训练的偏差，因而容易导致过拟合现象，后者更加关注模型的方差，因而容易导致欠拟合现象 [5]。前者往往效果要比后者好，但是由于串行的关系训练时间可能相对地较长[6]。

Boosting 和 Bagging 的思路都比较直接。Boosting 先从初始训练集中进行训练得到一个基学习器（Base learner），为了使得后续学习更加关注前面学习中分错的样本，我们根据基学习器的结果对样本分布通过加权的

[3] https://zhuanlan.zhihu.com/p/85731206

[4] 周志华，机器学习，清华大学出版社，pp.79-88

[5] 详见 4.2 开头部分

[6] 周志华，机器学习，清华大学出版社，pp.171-173

方式进行调整去训练下一个基学习器[7]。Bagging 想法则是利用自助采样法（Bootstrap sampling），将训练集进行切分并对切分后的训练集进行有放回的采样，以采样后的数据训练基学习器，这样做的目的以在尽量覆盖训练集的情况下训练出差异尽可能大的基学习器，最后在测试环节基学习器对最终的结果进行投票（voting），以少数服从多数的原则确定最终结果[8]。

[7]周志华，机器学习，清华大学出版社，pp.173

[8]周志华，机器学习，清华大学出版社，pp.178

第六部分

附录

附录 A　凸函数与次微分

A.1　凸函数的一般性质

对于定义域上的一个连续的函数，其中 $\mathrm{dom} f$ 代表了 f 的定义域，若

$$\forall \boldsymbol{x}, \boldsymbol{y} \in \mathrm{dom} f, f(\lambda \boldsymbol{x} + (1-\lambda)\boldsymbol{y}) \leqslant \lambda f(\boldsymbol{x}) + (1-\lambda) f(\boldsymbol{y}),\ 0 < \lambda < 1$$

则函数为凸函数（convex function）。一般地，考虑多个离散的数据点

$$\sum_{i=1}^{n} w_i = 1, w_i \geqslant 0 \Rightarrow f\left(\sum_{i=1}^{n} w_i \boldsymbol{x_i}\right) \leqslant \sum_{i=1}^{n} w_i f(\boldsymbol{x_i})$$

当且仅当 $\boldsymbol{x_1} = \boldsymbol{x_2} = \cdots = \boldsymbol{x_n}$ 时取等号。更一般地，考虑连续的情况

$$\int_{\mathcal{X}} \varphi(\boldsymbol{x})\mathrm{d}\boldsymbol{x} = 1, \varphi(\boldsymbol{x}) \geqslant 0 \Rightarrow f\left(\int_{\mathcal{X}} g(\boldsymbol{x})\varphi(\boldsymbol{x})\mathrm{d}\boldsymbol{x}\right) \leqslant \int_{\mathcal{X}} f(g(\boldsymbol{x}))\varphi(\boldsymbol{x})\mathrm{d}\boldsymbol{x}$$

当且仅当 $g(\boldsymbol{x})$ 几乎处处等于一个常数时取等号。考虑 X 为 $\boldsymbol{x}$ 对应的随机变量，则有 Jensen 不等式（Jensen inequality）的概率形式

$$f(\mathrm{E}[X]) \leqslant \mathrm{E}[f(X)]$$

这个不等式在统计学和信息论中很重要[1]。

定义实函数的 graph

$$\text{graph } f := \{(\boldsymbol{x}, y) \in \mathcal{X} \times \mathcal{Y} : y = f(\boldsymbol{x})\}$$

和其 epi graph[2]（对于二维的情况，epi graph 就是函数对应的曲线上方的区域）

[1] 我们在 F 中反复用到了它

[2] https://en.wikipedia.org/wiki/Convex_function

$$\operatorname{epi} f := \{(\boldsymbol{x}, r) \in \mathcal{X} \times \mathbb{R}, f(\boldsymbol{x}) \leqslant r\}$$

对于凸函数而言，其 epi graph 显然是一个凸集。

定义单变量函数

$$g(t; \boldsymbol{x}, \boldsymbol{y}) = f(\boldsymbol{x} + t\boldsymbol{y})$$

则函数 f 是凸函数的充要条件为对于定义域内任意的向量 $\boldsymbol{x}$ 和方向向量 $\boldsymbol{y}$ 函数 $g(t; \boldsymbol{x}, \boldsymbol{y})$ 都是凸函数，该式表明凸函数沿着各个方向均有凸性[3]。证明比较直观。如 f 是凸函数，则需证明

$$\begin{aligned} g(\lambda t_1 + (1-\lambda)t_2; \boldsymbol{x}, \boldsymbol{y}) &= f(\boldsymbol{x} + \lambda t_1 \boldsymbol{y} + (1-\lambda)t_2\boldsymbol{y}) \\ &\leqslant \lambda g(t_1; \boldsymbol{x}, \boldsymbol{y}) + (1-\lambda) g(t_2; \boldsymbol{x}, \boldsymbol{y}) \\ &= \lambda f(\boldsymbol{x} + t_1\boldsymbol{y}) + (1-\lambda) f(\boldsymbol{x} + t_2\boldsymbol{y}) \end{aligned}$$

而

$$\lambda f(\boldsymbol{x} + t_1\boldsymbol{y}) + (1-\lambda) f(\boldsymbol{x} + t_2\boldsymbol{y}) \geqslant f(\lambda(\boldsymbol{x} + t_1\boldsymbol{y}) + (1-\lambda)(\boldsymbol{x} + t_2\boldsymbol{y}))$$

展开得证。如果对于任意的向量 $\boldsymbol{x}$ 和方向向量 $\boldsymbol{y}$ 函数 $g(t; \boldsymbol{x}, \boldsymbol{y})$ 都是凸函数，仿照充分性证明，我们构造

$$\boldsymbol{x} = \boldsymbol{u} + \boldsymbol{v},\ \boldsymbol{y} = \boldsymbol{u} - \boldsymbol{v} \Rightarrow \boldsymbol{u} = \frac{1}{2}(\boldsymbol{x} + \boldsymbol{y}),\ \boldsymbol{v} = \frac{1}{2}(\boldsymbol{x} - \boldsymbol{y})$$

我们需要证明

$$\begin{aligned} f(\lambda\boldsymbol{x} + (1-\lambda)\boldsymbol{y}) &= f(\boldsymbol{u} + (2\lambda - 1)\boldsymbol{v}) = g(2\lambda - 1; \boldsymbol{u}, \boldsymbol{v}) \\ &\leqslant \lambda f(\boldsymbol{x}) + (1-\lambda) f(\boldsymbol{y}) = \lambda f(\boldsymbol{u} + \boldsymbol{v}) + (1-\lambda) f(\boldsymbol{u} - \boldsymbol{v}) \\ &= \lambda g(1; \boldsymbol{u}, \boldsymbol{v}) + (1-\lambda) g(-1; \boldsymbol{u}, \boldsymbol{v}) \end{aligned}$$

而

$$g(2\lambda - 1; \boldsymbol{u}, \boldsymbol{v}) = g(\lambda - (1-\lambda); \boldsymbol{u}, \boldsymbol{v}) \geqslant \lambda g(1; \boldsymbol{u}, \boldsymbol{v}) + (1-\lambda) g(-1; \boldsymbol{u}, \boldsymbol{v})$$

于是命题得证。

[3] https://www.princeton.edu/~aaa/Public/Teaching/ORF523/S16/ORF523_S16_Lec7_gh.pdf

A.2　一阶与二阶可微凸函数的性质

对于一阶可微的凸函数，其一阶性质为

$$\forall \boldsymbol{x}, \boldsymbol{y} \in \mathrm{dom} f, f(\boldsymbol{y}) - f(\boldsymbol{x}) \geqslant \nabla f(\boldsymbol{x})^T (\boldsymbol{y} - \boldsymbol{x})$$

证明考虑考虑单变量函数[3]

$$g(t; \boldsymbol{x}, \boldsymbol{y}) = f(\boldsymbol{x} + t\boldsymbol{y})$$

我们已经证明了函数 f 是凸函数当前仅当对于任意定义域上的向量 $\boldsymbol{x}$ 和方向向量 $\boldsymbol{y}$ 函数 $g(t; \boldsymbol{x}, \boldsymbol{y})$ 都是凸函数。对于一阶可微的函数 f 其对应的每个 $g(t; \boldsymbol{x}, \boldsymbol{y})$ 必然一阶可导，这是由于可微性保证了方向导数均存在。取向量空间中任意与 $\boldsymbol{x}$ 不同的点 $\boldsymbol{y}$ 利用凸函数性质沿 $\boldsymbol{x}$ 和 $\boldsymbol{y}$ 连线方向逼近 $\boldsymbol{x}$，得到

$$\begin{aligned} f(\boldsymbol{x} + \lambda(\boldsymbol{y} - \boldsymbol{x})) - f(\boldsymbol{x}) &= f((1-\lambda)\boldsymbol{x} + \lambda \boldsymbol{y}) - f(\boldsymbol{x}) \\ &\leqslant (1-\lambda) f(\boldsymbol{x}) + \lambda f(\boldsymbol{y}) - f(\boldsymbol{x}) = \lambda (f(\boldsymbol{y}) - f(\boldsymbol{x})) \end{aligned}$$

从而

$$f(\boldsymbol{y}) - f(\boldsymbol{x}) \geqslant \frac{f(\boldsymbol{x} + \lambda(\boldsymbol{y} - \boldsymbol{x})) - f(\boldsymbol{x})}{\lambda} = \frac{g(\lambda; \boldsymbol{x}, \boldsymbol{y} - \boldsymbol{x}) - g(0; \boldsymbol{x}, \boldsymbol{y} - \boldsymbol{x})}{\lambda}$$

利用链式法则 [4] 考虑 $g(t)$ 对 t 的导数

$$g'(0; \boldsymbol{x}, \boldsymbol{y}) = \nabla f(\boldsymbol{x})^T \boldsymbol{y}$$

取 $\lambda \to 0$ 得到

$$f(\boldsymbol{y}) - f(\boldsymbol{x}) \geqslant g'(0; \boldsymbol{x}, \boldsymbol{y} - \boldsymbol{x}) = \nabla f(\boldsymbol{x})^T (\boldsymbol{y} - \boldsymbol{x})$$

也可以证明一阶性质对于一阶可微的凸函数而言是必要的，这是因为考虑中间变量 $\boldsymbol{z}$

$$\begin{aligned} f(\boldsymbol{x}) &\geqslant f(\boldsymbol{z}) + f(\boldsymbol{z})^T (\boldsymbol{x} - \boldsymbol{z}) \\ f(\boldsymbol{y}) &\geqslant f(\boldsymbol{z}) + f(\boldsymbol{z})^T (\boldsymbol{y} - \boldsymbol{z}) \end{aligned}$$

[4] 详见 B.1 开头部分

这表明

$$\lambda f(\boldsymbol{x})+(1-\lambda)f(\boldsymbol{y})\geqslant f(\boldsymbol{z})+\nabla f(\boldsymbol{z})^T(\lambda(\boldsymbol{x}-\boldsymbol{z})+(1-\lambda)(\boldsymbol{y}-\boldsymbol{z}))$$

我们令

$$\lambda(\boldsymbol{x}-\boldsymbol{z})+(1-\lambda)(\boldsymbol{y}-\boldsymbol{z})=0\Rightarrow\boldsymbol{z}=\lambda\boldsymbol{x}+(1-\lambda)\boldsymbol{y}$$

得到

$$\lambda f(\boldsymbol{x})+(1-\lambda)f(\boldsymbol{y})\geqslant f(\boldsymbol{z})=f(\lambda\boldsymbol{x}+(1-\lambda)\boldsymbol{y})$$

将函数 $g(t;\boldsymbol{x},\boldsymbol{y})$ 简记为 $g(t)$。对于二阶可微的凸函数，任取定义域中的 $\boldsymbol{x}$，其对应位置的 Hessian 矩阵（Hessian matrix）$\nabla^2 f(\boldsymbol{x})$ 是半正定的。证明同样考虑单变量函数[3]

$$g(t;\boldsymbol{x},\boldsymbol{y})=f(\boldsymbol{x}+t\boldsymbol{y})$$

对于二阶可微的函数 f 其对应的每个 $g(t)$ 必然二阶可导。对于一维凸函数而言由其一阶性质得到

$$g(t)-g(t_0)\geqslant g'(t_0)(t-t_0)$$

$$g(t_0)-g(t)\geqslant g'(t)(t_0-t)$$

得到

$$g'(t_0)(t-t_0)+g'(t)(t_0-t)=(g'(t_0)-g'(t))(t-t_0)\leqslant 0$$

从而

$$t>t_0\Rightarrow g'(t)\geqslant g'(t_0)$$

得到在二阶可导的情况下有 $g''(t)$ 大于 0。考虑链式法则得到 $g''(t)$ 的展开式

$$g(t)''=\boldsymbol{y}^{\boldsymbol{T}}\nabla^2 f(\boldsymbol{x}+t\boldsymbol{y})\boldsymbol{y}\geqslant 0$$

对于任意 $\boldsymbol{x}$，任取方向向量 $\boldsymbol{y}$，取 $t \to 0$ 得到 $\nabla^2 f(\boldsymbol{x})$ 是非负的，从而我们证明了凸函数的二阶性质。

这个二阶性质对于二阶可微的凸函数同样是必要的。由于二阶可微的凸函数必然一阶可微，因而我们只需推导其一阶性质，由凸函数在各个方向上的凸性我们也只需验证每个 $g(t)$ 的一阶性质，对 $g(t)$ Taylor 展开得到

$$g(t) = g(t_0) + g'(t_0)(t - t_0) + g''(\xi)(t - t_0)^2 \geqslant g(t_0) + g'(t)(t - t_0)$$

因而凸函数得证。

我们利用凸函数的性质重新审视一下机器学习中的凸优化过程。考虑一阶优化，利用凸函数的一阶性质我们设置学习率 τ，得到梯度下降法（Gradient descent）

$$f(\boldsymbol{x}) \approx f(\boldsymbol{x^{(t)}}) + \nabla f(\boldsymbol{x^{(t)}})^T(\boldsymbol{x} - \boldsymbol{x^{(t)}})$$
$$f(\boldsymbol{x}) \geqslant f(\boldsymbol{x^{(t)}}) + \nabla f(\boldsymbol{x^{(t)}})^T(\boldsymbol{x} - \boldsymbol{x^{(t)}})$$

更新参数

$$\boldsymbol{x^{(t+1)}} := \boldsymbol{x^{(t)}} - \tau \nabla f(\boldsymbol{x^{(t)}})$$

得到函数值的一个下界

$$f(\boldsymbol{x^{(t+1)}}) \geqslant f(\boldsymbol{x^{(t)}}) - \tau \left\| \nabla f(\boldsymbol{x^{(t)}}) \right\|_2^2$$

当学习率越大，函数值下界越小，在收敛前优化潜力越大。

考虑二阶优化，利用凸函数的二阶性质

$$f(\boldsymbol{x}) \approx f(\boldsymbol{x^{(t)}}) + \nabla f(\boldsymbol{x^{(t)}})^T(\boldsymbol{x} - \boldsymbol{x^{(t)}}) + \frac{1}{2}(\boldsymbol{x} - \boldsymbol{x^{(t)}})^T \nabla^2 f(\boldsymbol{x^{(t)}})(\boldsymbol{x} - \boldsymbol{x^{(t)}})$$

我们利用二次函数对目标函数进行了近似，利用二次函数求最优值的方法，我们在不设置学习率的情况下完成了对参数的优化，且二阶导数保证了我们能够捕捉更多关于函数变化趋势相关的信息，此时我们得到了 Newton 法（Newton method）

$$\frac{\partial f}{\partial \boldsymbol{x}} = \nabla f(\boldsymbol{x^{(t)}}) + \nabla^2 f(\boldsymbol{x^{(t)}})(\boldsymbol{x} - \boldsymbol{x^{(t)}})$$

令导数为 0，记更新方向 $\boldsymbol{d} := \boldsymbol{x} - \boldsymbol{x}^{(t)}$ 我们实际上在求解线性方程组

$$\nabla^2 f(\boldsymbol{x}^{(t)})\boldsymbol{d} = -\nabla f(\boldsymbol{x}^{(t)})$$

从而

$$\boldsymbol{x}^{(t+1)} := \boldsymbol{x}^{(t)} + \boldsymbol{d}$$

在 $\nabla^2 f(\boldsymbol{x}^{(t)})$ 正定的情况下参数更新式可以写为

$$\boldsymbol{x}^{(t+1)} = \boldsymbol{x}^{(t)} - \nabla^2 f(\boldsymbol{x}^{(t)})^{-1}\nabla f(\boldsymbol{x}^{(t)})$$

我们可以证明参数更新的方向 $\boldsymbol{d}$ 在梯度反方向上的投影非负

$$\langle \boldsymbol{d}, \nabla f(\boldsymbol{x}^{(t)})\rangle = -\boldsymbol{d}^{\boldsymbol{T}}\nabla f(\boldsymbol{x}^{(t)}) = \boldsymbol{d}^{\boldsymbol{T}}\nabla^2 f(\boldsymbol{x}^{(t)})\boldsymbol{d} \geqslant 0$$

该式表明，凸优化中 Newton 法优化不会逆梯度进行。我们同样得到了函数后函数值的一个下界

$$f(\boldsymbol{x}^{(t+1)}) \geqslant f(\boldsymbol{x}^{(t)}) + \nabla f(\boldsymbol{x}^{(t)})^T\boldsymbol{d} = f(\boldsymbol{x}^{(t)}) - \boldsymbol{d}^{\boldsymbol{T}}\nabla^2 f(\boldsymbol{x}^{(t)})\boldsymbol{d}$$

在 $\nabla^2 f(\boldsymbol{x}^{(t)})$ 正定的情况下函数值的下界可以写为

$$f(\boldsymbol{x}^{(t+1)}) \geqslant f(\boldsymbol{x}^{(t)}) - \left\|\nabla f(\boldsymbol{x}^{(t)})\right\|^2_{\nabla^2 f(\boldsymbol{x}^{(t)})^{-1}}$$

我们把高维的问题都转化为单方向的一维的问题这种思想是重要的，我们接下来将进一步探讨在一般情况下凸函数沿着某个方向更深入的性质。

A.3　方向导数与次微分

事实上不是所有的凸函数在定义域上都具有一阶可微的性质，故我们考虑拓展梯度的定义。若对于凸函数 $f(\boldsymbol{x})$，存在 $\boldsymbol{g}$ 满足

$$\forall \boldsymbol{x}, \boldsymbol{x_0} \in I, f(\boldsymbol{x}) - f(\boldsymbol{x_0}) \geqslant \boldsymbol{g}^{\boldsymbol{T}}(\boldsymbol{x} - \boldsymbol{x_0})$$

则 $\boldsymbol{g}$ 被称为次梯度（subgradient）。我们将所有满足条件的 $\boldsymbol{g}$ 称为凸函数关于 $\boldsymbol{x}$ 的次微分（subdifferential），记为 $\partial f(\boldsymbol{x})$。当函数在该点可导时，次微分是唯一的，即 $\partial f(\boldsymbol{x}) = \{\nabla f(\boldsymbol{x})\}$。

对于凸函数而言，在每个点上次微分总是存在的，其几何意义为在几何空间中过凸集上的每个点总存在一个超平面，将凸集完全划分至超平面的一侧上去[5]。

从具有一阶可微的凸函数的相关性质看，可以从一个方向构造逼近 $\boldsymbol{x_0}$ 的序列，使得切面每一维对应的元素的值不断下降。构造方向导数（Directional derivative）[6]

$$f'(\boldsymbol{x};\boldsymbol{y}) := \inf_{\lambda>0} \frac{f((1-\lambda)\boldsymbol{x}+\lambda(\boldsymbol{x}+\boldsymbol{y}))-f(\boldsymbol{x})}{\lambda} = \inf_{\lambda>0} \frac{f(\boldsymbol{x}+\lambda\boldsymbol{y})-f(\boldsymbol{x})}{\lambda}$$

容易得到

$$f'(\boldsymbol{x};\lambda\boldsymbol{y}) = \lambda f'(\boldsymbol{x};\boldsymbol{y})$$

和

$$f(\boldsymbol{x}+\lambda\boldsymbol{y}) \geqslant f(\boldsymbol{x}) + \lambda f'(\boldsymbol{x};\boldsymbol{y}) = f(\boldsymbol{x}) + f'(\boldsymbol{x};\lambda\boldsymbol{y}),\ \lambda>0$$

定义

$$h(\boldsymbol{x};\boldsymbol{y},\lambda) := \frac{f(\boldsymbol{x}+\lambda\boldsymbol{y})-f(\boldsymbol{x})}{\lambda} = \frac{1}{\lambda}h(\boldsymbol{x};\lambda\boldsymbol{y},1),\ \lambda>0$$

由 $f(\boldsymbol{x})$ 的凸性得到

$$\begin{aligned} h(\boldsymbol{x};\boldsymbol{y},\lambda) &= \frac{f((1-\lambda)\boldsymbol{x}+\lambda(\boldsymbol{x}+\boldsymbol{y}))-f(\boldsymbol{x})}{\lambda} \\ &\leqslant f(\boldsymbol{x}+\boldsymbol{y})-f(\boldsymbol{x}) = h(\boldsymbol{x};\boldsymbol{y},1),\ 0<\lambda<1 \end{aligned}$$

立即得到

$$\begin{aligned} h(\boldsymbol{x};\boldsymbol{y},\lambda_1) &= \frac{1}{\lambda_1}h(\boldsymbol{x};\lambda_1\boldsymbol{y},1) \\ &\geqslant \frac{1}{\lambda_1}h(\boldsymbol{x};\lambda_1\boldsymbol{y},\lambda_2) = h(\boldsymbol{x};\boldsymbol{y},\lambda_1\lambda_2),\ \lambda_1>0, 1>\lambda_2>0 \end{aligned}$$

从而得到

[5] Luenberger, David G., Ye, Yinyu, *Linear and Nonlinear Programming (Fifth Edition)*, pp.575-577

[6] www.seas.ucla.edu/~vandenbe/236C/lectures/subgradients.pdf

$$\forall \alpha_1 > \alpha_2 > 0,\ h(\boldsymbol{x};\boldsymbol{y},\alpha_1) \leqslant h(\boldsymbol{x};\boldsymbol{y},\alpha_2)$$

接下来我们将通过 $f'(\boldsymbol{x};\boldsymbol{y})$ 的有界性证明方向导数总是存在的（考虑广义极限），从而利用单调有界必有极限推得

$$f'(\boldsymbol{x};\boldsymbol{y}) = \lim_{\lambda\to 0} h(\boldsymbol{x};\boldsymbol{y},\lambda) = \inf_{\lambda>0} h(\boldsymbol{x};\boldsymbol{y},\lambda)$$

由定义，任取 $\boldsymbol{g} \in \partial f(\boldsymbol{x})$ 得到

$$f'(\boldsymbol{x};\boldsymbol{y}) = \inf_{\lambda>0} \frac{f(\boldsymbol{x}+\lambda\boldsymbol{y}) - f(\boldsymbol{x})}{\lambda} \geqslant \inf_{\lambda>0} \frac{f(\boldsymbol{x}) + \boldsymbol{g}^{\boldsymbol{T}}\boldsymbol{y} - f(\boldsymbol{x})}{\lambda} = \boldsymbol{g}^{\boldsymbol{T}}\boldsymbol{y}$$

从而

$$f'(\boldsymbol{x};\boldsymbol{y}) \geqslant \sup_{\boldsymbol{g}\in\partial f(\boldsymbol{x})} \boldsymbol{g}^{\boldsymbol{T}}\boldsymbol{y}$$

这表明方向导数实际上是沿方向向量 $\boldsymbol{y}$ 方向下降最快的次梯度对应的下降的值。我们接下来将证明这个等号实际上是可以取到的。

注意到由 $f(\boldsymbol{x})$ 的凸性得到 $h(\boldsymbol{x};\boldsymbol{y},\lambda)$ 关于 $\boldsymbol{y}$ 是凸的，取极限后得到 $f'(\boldsymbol{x};\boldsymbol{y})$ 关于 $\boldsymbol{y}$ 也是凸的。取 $f'(\boldsymbol{x};\boldsymbol{y})$ 关于 $\boldsymbol{y}$ 的一个次梯度 $\hat{\boldsymbol{g}}$，任取空间中的一个点 $\boldsymbol{v}$ 和 $\lambda > 0$ 得到[6]

$$f'(\boldsymbol{x};\lambda\boldsymbol{v}) = \lambda f'(\boldsymbol{x};\boldsymbol{v}) \geqslant f'(\boldsymbol{x};\boldsymbol{y}) + \hat{\boldsymbol{g}}^{\boldsymbol{T}}(\lambda\boldsymbol{v} - \boldsymbol{y})$$

变形得到

$$f'(\boldsymbol{x};\boldsymbol{v}) \geqslant \frac{1}{\lambda}(f'(\boldsymbol{x};\boldsymbol{y}) - \hat{\boldsymbol{g}}^{\boldsymbol{T}}\boldsymbol{y}) + \hat{\boldsymbol{g}}^{\boldsymbol{T}}\boldsymbol{v}$$
$$f'(\boldsymbol{x};\boldsymbol{y}) \leqslant \lambda f'(\boldsymbol{x};\boldsymbol{v}) - \lambda\hat{\boldsymbol{g}}^{\boldsymbol{T}}\boldsymbol{v} + \hat{\boldsymbol{g}}^{\boldsymbol{T}}\boldsymbol{y}$$

取 $\lambda \to 0$ 得到

$$f'(\boldsymbol{x};\boldsymbol{y}) \leqslant \hat{\boldsymbol{g}}^{\boldsymbol{T}}\boldsymbol{y}$$

取 $\lambda \to +\infty$ 得到

$$f'(\boldsymbol{x};\boldsymbol{v}) \geqslant \hat{\boldsymbol{g}}^{\boldsymbol{T}}\boldsymbol{v}$$

只需证明 $\hat{\boldsymbol{g}} \in \partial f(\boldsymbol{x})$ 从而得到

$$f'(\boldsymbol{x};\boldsymbol{y}) = \hat{\boldsymbol{g}}^{\boldsymbol{T}}\boldsymbol{y} = \sup_{\boldsymbol{g}\in\partial f(\boldsymbol{x})} \boldsymbol{g}^{\boldsymbol{T}}\boldsymbol{y}$$

考虑

$$f(\boldsymbol{x}+\boldsymbol{y}) \geqslant f(\boldsymbol{x}) + f'(\boldsymbol{x};\boldsymbol{y}) \geqslant f(\boldsymbol{x}) + \hat{\boldsymbol{g}}^{\boldsymbol{T}}\boldsymbol{y}$$

由定义得到 $\hat{\boldsymbol{g}} \in \partial f(\boldsymbol{x})$。若函数在 $\boldsymbol{x}$ 处可导，则可以立即得到

$$f'(\boldsymbol{x};\boldsymbol{y}) = \nabla^T f(\boldsymbol{x})\boldsymbol{y}$$

由于沿各个方向极限的值虽然存在但不一定相等，故凸函数的一阶导数不一定存在。有时函数的次微分可以通过方向导数得出（至少是一个子集），而有时这样做是没有必要的，因为我们只需要计算得到一个次梯度即可。

有关次微分我们有以下重要结论[6]

$$\boldsymbol{x}^* = \arg\min_{\boldsymbol{x}} f(\boldsymbol{x}) \Leftrightarrow \mathbf{0} \in \partial f(\boldsymbol{x})$$

这是因为由次微分的定义

$$f(\boldsymbol{x}) - f(\boldsymbol{x}^*) \geqslant \mathbf{0}(\boldsymbol{x}-\boldsymbol{x}^*) \Leftrightarrow f(\boldsymbol{x}) \geqslant f(\boldsymbol{x}^*)$$

这为我们提供了存在不可导点函数的凸函数的最值提供了方法。

附录 B　Jacobian 矩阵

B.1　Jacobian 矩阵与行列式

当 $M = N$ 时 Jacobian 矩阵是一个方阵，其对应的行列式对于多元函数的微积分而言具有重要意义。当 $N = 1$ 时，Jacobian 矩阵为函数 $\boldsymbol{f}$ 的梯度的转置，当 $M = N = 1$ 时，Jacobian 矩阵就对应着函数的导数，从这个角度思考，Jacobian 矩阵实际上是导数在多元函数上的推广。

Jacobian 矩阵对于复合函数 $\boldsymbol{h}(\boldsymbol{x}) = \boldsymbol{f}(\boldsymbol{g}(\boldsymbol{x}))$ 而言，存在链式法则 (Chain rule)

$$\boldsymbol{J_h}(\boldsymbol{x}) = \boldsymbol{J_f}(\boldsymbol{g}(\boldsymbol{x}))\boldsymbol{J_g}(\boldsymbol{x})$$

该法则是偏导的链式法则赋予的。当 $M = N = 1$ 时，该法则就是一元函数求导的链式法则。取 $M = N$ 时, 当 $\boldsymbol{f}$ 可逆时

$$\boldsymbol{J_I}(\boldsymbol{x}) = \boldsymbol{I} = \boldsymbol{J_{f^{-1}}}(\boldsymbol{f}(\boldsymbol{x}))\boldsymbol{J_f}(\boldsymbol{x})$$

即[1]

$$\boldsymbol{J_{f^{-1}}}(\boldsymbol{f}(\boldsymbol{x})) = \boldsymbol{J_f^{-1}}(\boldsymbol{x})$$

对于函数 $\boldsymbol{f}$ 的可导的一点 $\boldsymbol{p}$，Jacobian 矩阵可以用以表示对点 $\boldsymbol{p}$ 临近点的函数值最佳的线性估计（Linear approximation）[1]，即

$$\boldsymbol{f}(\boldsymbol{x}) - \boldsymbol{f}(\boldsymbol{p}) \approx \boldsymbol{J_f}(\boldsymbol{p})(\boldsymbol{x} - \boldsymbol{p})$$

当 $M = N = 1$ 时上式为函数的一次的 Taylor 展开。上式也可以表示为

[1] https://en.wikipedia.org/wiki/Jacobian_matrix_and_determinant

$$\boldsymbol{\Delta f}(\boldsymbol{x}) \approx \boldsymbol{J_f}(\boldsymbol{p})\boldsymbol{\Delta x}$$

从变量微分的角度看，可以得到

$$\mathrm{d}\boldsymbol{f}(\boldsymbol{x}) = \mathrm{d}\boldsymbol{y} = \boldsymbol{J_f}(\boldsymbol{x})\mathrm{d}\boldsymbol{x}$$

上式是由全微分公式赋予的，当 $N = 1$ 时上式为全微分公式

$$\mathrm{d}f = \sum_{n=1}^{N} \frac{\partial f}{\partial x_n}\mathrm{d}x_n = \nabla_{\boldsymbol{x}}^{T} f \mathrm{d}\boldsymbol{x}$$

Jacobian 矩阵是利用线性变换对多元函数的估计，而行列式是线性变换对轴的伸缩因子的乘积或是线性变换过程中单位“体积”变化的度量，这也可以用奇异值分解去理解，由于正交变换实际上并不会改变体积度量，体积度量的变化由奇异值对角阵的伸缩因子决定 [2]。事实上 Jacobian 矩阵也存在着类似的关系[3]

$$\mathrm{d}V' = \left|\det(\boldsymbol{J_f}(\boldsymbol{x}))\right| \mathrm{d}V$$

很直观地，我们有

$$\prod_{i=1}^{n} \mathrm{d}y_i = \left|\det(\boldsymbol{J_f}(\boldsymbol{x}))\right| \prod_{i=1}^{n} \mathrm{d}x_i$$

该式是多元函数积分的坐标变换和概率密度函数的换元公式的关键。

对于向量映射到向量的可逆映射 $\boldsymbol{f}$

$$f : \mathbb{R}^N \mapsto \mathbb{R}^N$$

对于关于随机变量 X 的概率密度函数 $p_X(\boldsymbol{x})$ 而言，取概率微元[4]

$$p_X(\boldsymbol{x}) \prod_{i=1}^{n} \mathrm{d}x_i$$

进行变量代换 $Y = \boldsymbol{f}(X)$ 后概率微元变为

[2] 详见 D.2.2

[3] 严格证明见：张筑生，数学分析新讲第二册，北京大学出版社，pp.363-370

[4] https://zhuanlan.zhihu.com/p/59615785

$$p_Y(\boldsymbol{y})\prod_{i=1}^{n}\mathrm{d}y_i = p_X(\boldsymbol{x})\prod_{i=1}^{n}\mathrm{d}x_i = p_X(\boldsymbol{f^{-1}}(\boldsymbol{y}))\left|\det(\boldsymbol{J_{f^{-1}}}(\boldsymbol{y}))\right|\prod_{i=1}^{n}\mathrm{d}y_i$$

得到概率密度换元公式

$$p_Y(\boldsymbol{y}) = p_X(\boldsymbol{f^{-1}}(\boldsymbol{y}))\left|\det(\boldsymbol{J_{f^{-1}}}(\boldsymbol{y}))\right| = p_X(\boldsymbol{x})\left|\det(\boldsymbol{J_f}(\boldsymbol{x}))\right|^{-1}$$

B.2　标准化流简介

标准化流（Normalization flow）是机器学习领域中和 Jacobian 矩阵相关的一个重要的生成模型。标准化流是利用简单的正态分布通过一系列变换拟合复杂分布。设第 K 层变量为 $\boldsymbol{z_K}$，其中 $\boldsymbol{z_0}$ 代表服从初始分布即正态分布的变量，则 $\boldsymbol{z_K}$ 可以表示为[5]

$$\boldsymbol{z_K} = \boldsymbol{f_K}\circ\cdots\circ\boldsymbol{f_2}\circ\boldsymbol{f_1}(\boldsymbol{z_0})$$

对概率密度换元公式取对数得到

$$\begin{aligned}\log p(\boldsymbol{z_K}) &= \log p(\boldsymbol{z_{K-1}}) + \log\left|\det(\boldsymbol{J_{f_K}}(\boldsymbol{z_{K-1}}))\right|^{-1}\\ &= \cdots = \log p(\boldsymbol{z_0}) + \sum_{k=1}^{K}\left|\det(\boldsymbol{J_{f_k}}(\boldsymbol{z_{k-1}}))\right|^{-1}\end{aligned}$$

我们不显示地计算样本的对数似然函数，而是通过一系列变换通过初始分布和 Jacobian 行列式将其计算出，最终借助 MLE 采用梯度下降法更新模型参数。标准化流在早期由于计算量大等原因在实际中难以得到使用，随着硬件设施的发展近年来不断得到重视。在标准化流中，如何设计一个可逆的便于计算 Jacobian 矩阵的一系列映射使得模型计算量下降并保证模型的表征能力是一个核心的问题[4]。标准化流在变分推断（Variational inference）中的应用我们在 F.4.1 中再介绍。

[5] Danilo Jimenez Rezende, Shakir Mohamed, *Variational Inference with Normalizing Flows*, p.4

附录 C　RKHS

与 5.2 所述从 RKHS 出发构造 K 不同，我们考虑设计 K 构造 RKHS 的过程，这个过程更加贴近核技巧的思考方式。

C.1　向量空间回顾

在讨论核技巧之前我们回顾一下向量空间中的一些概念，由于实际问题中通常只涉及实数，我们只讨论实向量。首先回顾一下实对称矩阵的特征值分解，对于任意的实对称矩阵 $\boldsymbol{A}$，即 $a_{ij} = a_{ji}$，任意的实对称矩阵 $\boldsymbol{A}$ 均存在如下形式的特征值分解，该定理被称为谱定理（Spectral Theorem）[1]，通常规定特征值需要降序排序

$$\boldsymbol{A} = \boldsymbol{V}\boldsymbol{\Lambda}\boldsymbol{V}^{\boldsymbol{T}} = \sum_{i=1}^{N} \lambda_i \boldsymbol{v_i}\boldsymbol{v_i^T},\ \boldsymbol{V}\boldsymbol{V}^{\boldsymbol{T}} = \boldsymbol{I}$$

即矩阵 $\boldsymbol{V}$ 的列向量 $\{\boldsymbol{v_i}\}_{i=1}^{N}$（是 $\boldsymbol{\Lambda}$ 对应位置的特征值的特征向量）构成空间 $\mathbb{R}^n$ 的一组单位正交基。若

$$\forall \boldsymbol{x},\ \boldsymbol{x}^{\boldsymbol{T}}\boldsymbol{A}\boldsymbol{x} = \sum_{i=1}^{N}\sum_{j=1}^{N} x_i a_{ij} x_j \geqslant 0$$

则矩阵 $\boldsymbol{A}$ 是半正定的，特别的，若严格有

$$\forall \boldsymbol{x} \neq \boldsymbol{0},\ \boldsymbol{x}^{\boldsymbol{T}}\boldsymbol{A}\boldsymbol{x} > 0$$

则矩阵 $\boldsymbol{A}$ 是正定的。特别地，对于半正定矩阵，其特征值时非负的，因而可以记

$$\boldsymbol{Z} = \boldsymbol{\Lambda}^{\boldsymbol{1/2}}\boldsymbol{V}^{\boldsymbol{T}}$$

[1]详见 D.2.1 开头部分

从而存在分解式

$$\boldsymbol{A} = \boldsymbol{Z}^{\boldsymbol{T}}\boldsymbol{Z}$$

然后回顾一下内积运算和 Gram 矩阵。向量空间中满足正定性和对称性的双线性函数 $\langle\cdot,\cdot\rangle$ 称为向量空间的内积，即满足：

双线性

$$\langle a\boldsymbol{x_1} + b\boldsymbol{x_2}, \boldsymbol{y}\rangle = a\langle \boldsymbol{x_1}, \boldsymbol{y}\rangle + b\langle \boldsymbol{x_2}, \boldsymbol{y}\rangle$$

$$\langle \boldsymbol{x}, a\boldsymbol{y_1} + b\boldsymbol{y_2}\rangle = a\langle \boldsymbol{x}, \boldsymbol{y_1}\rangle + b\langle \boldsymbol{x}, \boldsymbol{y_2}\rangle$$

对称性

$$\langle \boldsymbol{x}, \boldsymbol{y}\rangle = \langle \boldsymbol{y}, \boldsymbol{x}\rangle$$

正定性

$$|\boldsymbol{x}| = \langle \boldsymbol{x}, \boldsymbol{x}\rangle > 0$$

$$|\boldsymbol{x}| = \langle \boldsymbol{x}, \boldsymbol{x}\rangle = 0 \Rightarrow \boldsymbol{x} = 0$$

设 N 维向量空间的一组基为 $\{\boldsymbol{b_i}\}_{i=1}^N$，将该组基排列为矩阵 $\boldsymbol{B}$，定义其 Gram 矩阵 $\boldsymbol{G}$，设 $\boldsymbol{b_i}$ 同样表自身在空间中自然基下的坐标，此时内积运算为标准内积运算，即满足

$$\boldsymbol{G} = (\langle \boldsymbol{b_i}, \boldsymbol{b_j}\rangle)_{N\times N} = (\boldsymbol{b_i^T b_j})_{N\times N} = \boldsymbol{B^T B}$$

在基 $\{\boldsymbol{b_i}\}_{i=1}^N$ 下对于 Euclidean 空间中两个向量 $\boldsymbol{x}$ 和 $\boldsymbol{y}$，两者之间的内积有

$$\langle \boldsymbol{x}, \boldsymbol{y}\rangle = \boldsymbol{x^T G y}$$

当 $\boldsymbol{G}$ 为单位矩阵时，即基为标准正交基时，内积为标准内积

$$\langle \boldsymbol{x}, \boldsymbol{y}\rangle = \boldsymbol{x^T y}$$

对于给定的 Gram 矩阵 $\boldsymbol{G}$（内积的运算性质要求必须其必须是正定的），设其特征值分解为

$$\boldsymbol{G} = \boldsymbol{V}\boldsymbol{\Lambda}\boldsymbol{V}^{\boldsymbol{T}}$$

取 $\boldsymbol{Z} = \boldsymbol{\Lambda}^{\boldsymbol{1/2}}\boldsymbol{V}^{\boldsymbol{T}}$ 则

$$\boldsymbol{G} = \boldsymbol{Z}^{\boldsymbol{T}}\boldsymbol{Z}$$

$\boldsymbol{Z}$ 可以视为在给定 Gram 矩阵的情况下自然基下空间的一组基的坐标排列得到的矩阵。尽管 $\boldsymbol{Z}$ 不是唯一的，因为将其左乘任意一个正交矩阵其 Gram 矩阵不变，但是如果我们只考虑向量空间中的内积运算我们可以完全忽略 $\boldsymbol{Z}$ 具有的形式，因为 Gram 矩阵事实上已经控制了空间中向量的所有的内积运算。

C.2 从向量空间到函数空间

在代数教材中经常有意地淡化向量作为数字序列的概念，这是因为数字序列仅仅是向量的一种特殊的通常而言是直观的一种表述方式，事实上向量可以是很多别的事物，比如一段文字、一张图片、一段语音等等，只要在这之上定义了某种运算，满足向量空间的某些性质，向量空间的结论就能自然而然地迁移过来。

集合论相关知识告诉我们，由于 N 维的实向量实际上是 $\{1, 2, \cdots, N\}$ 到 $\mathbb{R}$ 的映射，实际上 N 维的实向量是对定义在 $\{1, 2, \cdots, N\}$ 的一个实函数的表示方式，很自然地想到我们是否能够将我们一般讨论的向量空间中向量的概念推广至一般的函数，从而套用向量空间的某些结论呢？定义域 $\mathcal{X}$ 上的实函数 f 实际上是定义域上的每个点到实数的一对多映射。因而仿照有限维的实向量（是有限个整数到实数域上的一对多映射）任意的 f 均可视为一个无穷维的向量 $\boldsymbol{f}$[2]。我们可以定义向量的标准内积运算，从而定义一个新的与定义域上所有满足要求的函数有关的 Euclidean 空间，这部分在数学分析课程傅里叶级数相关内容中有所应用

$$\langle f, g\rangle := \langle \boldsymbol{f}, \boldsymbol{g}\rangle = \boldsymbol{f}^{\boldsymbol{T}}\boldsymbol{g} = \int_{\mathcal{X}} f(\boldsymbol{x})g(\boldsymbol{x})\mathrm{d}\boldsymbol{x}$$

这里我们通常要求定义域 $\mathcal{X}$ 上的函数 f 和 g 是平方可积的，因为

$$\|f\|_2^2 := \langle f, f\rangle = \|\boldsymbol{f}\|_2^2 = \boldsymbol{f}^{\boldsymbol{T}}\boldsymbol{f} = \int_{\mathcal{X}} f(\boldsymbol{x})^2\mathrm{d}\boldsymbol{x} < \infty$$

[2] http://songcy.net/posts/story-of-basis-and-kernel-part-2/

我们于是类似地定义了函数的 L^2 范数，我们把定义域 $\mathcal{X}$ 上 L^2 范数存在，即平方可积的函数的集合定义为 $L_2(\mathcal{X})$。可以用向量空间的定义证明 $L_2(\mathcal{X})$ 实际上也是一个向量空间，我们将 $L_2(\mathcal{X})$ 上的函数 f 和 g 的标准内积运算记为 $\langle f,g\rangle_{L_2}$ [3]。

对于二元函数 K

$$K:\mathcal{X}^2\mapsto\mathbb{R}$$

其可类似地视为一个无穷维的矩阵[2]。当 $K(\boldsymbol{x},\boldsymbol{y})=K(\boldsymbol{y},\boldsymbol{x})$ 时矩阵 $\boldsymbol{K}$ 为实对称的，定义线性算子

$$T_K:L_2(\mathcal{X})\mapsto L_2(\mathcal{X}),\ [T_Kf](\cdot)=\int_{\mathcal{X}}K(\cdot,\boldsymbol{x})f(\boldsymbol{x})\mathrm{d}\boldsymbol{x}$$

类似地我们可以定义矩阵的特征值和特征向量（特征函数）

$$\boldsymbol{K}\boldsymbol{\psi}=\lambda\boldsymbol{\psi}\Rightarrow[T_K\psi]=\lambda\psi\Rightarrow\forall y\in\mathbb{R},\ \int_{\mathcal{X}}K(\boldsymbol{x},\boldsymbol{y})\psi(\boldsymbol{x})\mathrm{d}\boldsymbol{x}=\lambda\psi(\boldsymbol{y})$$

类比实对称矩阵属于不同特征向量相互正交的证明，我们也可以证明属于不同特征值的特征函数之间也是相互正交的

$$\begin{aligned}\lambda_1\langle\boldsymbol{\psi_1},\boldsymbol{\psi_2}\rangle&=\int_{\mathcal{X}}\lambda_1\psi_1(\boldsymbol{x})\psi_2(\boldsymbol{x})\mathrm{d}\boldsymbol{x}=\int_{\mathcal{X}}\int_{\mathcal{X}}K(\boldsymbol{y},\boldsymbol{x})\psi_1(\boldsymbol{y})\mathrm{d}\boldsymbol{y}\psi_2(\boldsymbol{x})\mathrm{d}\boldsymbol{x}\\&=\int_{\mathcal{X}}\int_{\mathcal{X}}K(\boldsymbol{y},\boldsymbol{x})\psi_2(\boldsymbol{x})\mathrm{d}\boldsymbol{x}\psi_1(\boldsymbol{y})\mathrm{d}\boldsymbol{y}=\int_{\mathcal{X}}\lambda_2\psi_2(\boldsymbol{x})\psi_2(\boldsymbol{x})\mathrm{d}\boldsymbol{x}\\&=\lambda_2\langle\boldsymbol{\psi_1},\boldsymbol{\psi_2}\rangle\end{aligned}$$

得到

$$\lambda_1\neq\lambda_2\Rightarrow\langle\boldsymbol{\psi_1},\boldsymbol{\psi_2}\rangle=\int_{\mathcal{X}}\psi_1(\boldsymbol{x})\psi_2(\boldsymbol{x})\mathrm{d}\boldsymbol{x}=0$$

可以类似地定义（半）正定的概念，并将这个概念推广至二元的对称函数 K

$$\forall f\in L_2(\mathcal{X}),\ \boldsymbol{f^T}\boldsymbol{K}\boldsymbol{f}=\int_{\mathcal{X}}\int_{\mathcal{X}}f(\boldsymbol{x})K(\boldsymbol{x},\boldsymbol{y})f(\boldsymbol{y})\mathrm{d}\boldsymbol{x}\mathrm{d}\boldsymbol{y}\geqslant 0$$

由 $\boldsymbol{K}$ 的特征值分解，也可以类似地定义特征值分解，下面需要保证函数是正定的

[3] https://en.wikipedia.org/wiki/Reproducing_kernel_Hilbert_space

$$\boldsymbol{K} = \boldsymbol{\psi}\boldsymbol{\Lambda}\boldsymbol{\psi}^{\boldsymbol{T}} = (\boldsymbol{\psi}\boldsymbol{\Lambda}^{\boldsymbol{1/2}})(\boldsymbol{\Lambda}^{\boldsymbol{1/2}}\boldsymbol{\psi}^{\boldsymbol{T}})$$
$$\Rightarrow \forall \boldsymbol{x}, \boldsymbol{y} \in \mathcal{X}, K(\boldsymbol{x}, \boldsymbol{y}) = \sum_{i=1}^{D} \lambda_i \psi_i(\boldsymbol{x}) \psi_i(\boldsymbol{y}) = \sum_{i=1}^{D} (\sqrt{\lambda_i}\psi_i(\boldsymbol{x}))(\sqrt{\lambda_i}\psi_i(\boldsymbol{y}))$$

如此得到的特征值和特征函数可能是无穷维的。其中 $\lambda_i \geqslant 0$ 且 $\{\psi_i\}_{i=1}^{D}$ 构成向量空间的一组标准正交基。上述与谱定理相对的定理称为 Mercer 定理（Mercer's theorem）[4]，通常需要规定特征值是降序排序的，且显然无需考虑为 0 的特征值和对应的特征函数，这里我们没有要求二元函数的正定条件和矩阵的正定条件一样严格是因为我们并不要求 $\{\psi_i\}_{i=1}^{D}$ 的元素数量维度受 $\boldsymbol{K}$ 大小的约束而为无穷多个。

C.3 构造 Hilbert 空间

以基 $\{\psi_i\}_{i=1}^{D}$ 生成 Hilbert 空间 $\mathcal{H}_K$[3]。其中正定函数 K 被称为空间的核函数，我们实际上通过一个正定函数 K 曲折地确定了 $\mathcal{H}_K$ 的一组基。

在 $\mathcal{H}_K$ 我们不再使用加粗符号表示函数对应的向量，因为其内积等运算的迁移是自然的，且用数字序列表示向量实际上也只是函数的一种表示方法而已，上面的推导仅是为了方便理解使用了向量和矩阵的相关记号。
任意 $\mathcal{H}_K$ 上的函数均可被这些标准正交基的线性组合表出

$$\forall f \in \mathcal{H}_K,\ f = \sum_{i=1}^{D} f_i \psi_i \Rightarrow f = (f_1, f_2, \cdots)_{\mathcal{H}_K}^{T}$$

特别地

$$K(\boldsymbol{x}, \boldsymbol{y}) = \sum_{i=1}^{D} \lambda_i \psi_i(\boldsymbol{x}) \psi_i(\boldsymbol{y})$$
$$\Rightarrow K_{\boldsymbol{x}} = K(\boldsymbol{x}, \cdot) = K(\cdot, \boldsymbol{x}) = \sum_{i=1}^{D} \lambda_i \psi_i(\boldsymbol{x}) \psi_i$$
$$\Rightarrow K_{\boldsymbol{x}} = (\lambda_1\psi_1(\boldsymbol{x}), \lambda_2\psi_2(\boldsymbol{x}), \cdots)_{\mathcal{H}_K}^{T}$$

对于另一个函数 $g \in \mathcal{H}_K$，其和 f 之间的内积运算如果按照标准内积计算其结果为

[4] https://en.wikipedia.org/wiki/Mercer%27s_theorem

$$\langle f, g\rangle_{L_2} = \sum_{i=1}^{D} f_i g_i$$

我们实际上可以很自然地以 $L_2(\mathcal{X})$ 上的标准内积运算为基础定义 $\mathcal{H}_K$ 上新的内积运算，这样的内积运算可以保证我们接下来的两个重要的运算性质

$$\langle f, g\rangle_{\mathcal{H}_K} = \sum_{i=1}^{D} \frac{f_i g_i}{\lambda_i} = \sum_{i=1}^{D} \frac{\langle f, \psi_i\rangle_{L_2}\langle g, \psi_i\rangle_{L_2}}{\lambda_i}$$

实际上这也可以通过定义 $\mathcal{H}_K$ 的基之间的内积运算实现

$$\langle \psi_i, \psi_j\rangle_{\mathcal{H}_K} = \frac{\langle \psi_i, \psi_j\rangle}{\sqrt{\lambda_i\lambda_j}} = \begin{cases} 1/\lambda_i & i = j \\ 0 & i \neq j \end{cases}$$

其中 K 的正定性保证了我们定义的内积运算 $\langle\cdot,\cdot\rangle_{\mathcal{H}_K}$ 符合内积的定义中的正定性，其余两条内积性质的证明是显然的。完善我们对 $\mathcal{H}_K$ 的定义[3]

$$\mathcal{H}_K = \left\{ f \in L_2(\mathcal{X}) \;\middle|\; \|f\|^2_{\mathcal{H}_K} := \langle f, f\rangle_{\mathcal{H}_K} = \sum_{i=1}^{D} \frac{\langle f, \psi_i\rangle^2_{L_2}}{\lambda_i} < \infty \right\}$$

边缘函数 $K_{\boldsymbol{x}}$ 实际上代表了 K 对应矩阵的列向量，其内积运算具有如下两个重要性质[2]：

性质一

$$\langle K_{\boldsymbol{x}}, K_{\boldsymbol{y}}\rangle_{\mathcal{H}_K} = \sum_{i=1}^{D} \lambda_i \psi_i(\boldsymbol{x})\psi_i(\boldsymbol{y}) = K(\boldsymbol{x}, \boldsymbol{y})$$

该性质指出当考虑以 $\{K_{\boldsymbol{x}}\}_{\boldsymbol{x}\in\mathcal{X}}$ 生成的子空间 $\mathcal{F}$ 即

$$\mathcal{F} = \mathrm{span}(\{K_{\boldsymbol{x}}\}_{\boldsymbol{x}\in\mathcal{X}})$$

K 实际上充当了 $\mathcal{F}$ 中类似于 Gram 矩阵的作用（之所以说是类似的是因为 $\{K_{\boldsymbol{x}}\}_{\boldsymbol{x}\in\mathcal{X}}$ 之间可能存在线性相关关系使得这样的一组向量不能成为特征空间的一组基，这样的冗余使得特征空间的维度可能远小于 $|\mathcal{X}|$，甚至对于势为无穷的样本空间 $\mathcal{X}$ 特征空间可能还是有限维的），其选取的原则和我们之前讨论的向量空间中的 Gram 矩阵的选取原则是类似的，即必须保证正定性。

性质二

$$\forall f \in \mathcal{H}_K,\ \langle f, K_{\boldsymbol{x}} \rangle_{\mathcal{H}_K} = \sum_{i=1}^{D} f_i \psi_i(\boldsymbol{x}) = f(\boldsymbol{x})$$

该性质被称为再生性质（Reproducing property），由于整个 Hilbert 空间是由核函数的特征函数生成的，这些特征函数通过线性组合构造出了其他的函数，且通过与边缘函数 $K_{\boldsymbol{x}}$ 的内积运算可以“再生”出对应的函数值，因而 $\mathcal{H}_K$ 也被称为再生核 Hilbert 空间（Reproducing kernel Hilbert space / RKHS）。通过定义新的内积运算我们也得到了 RKHS 的性质，并且我们指明了 K 设计的原则，即正定性。

将边缘函数记为

$$\phi(\boldsymbol{x}) = K_{\boldsymbol{x}}$$

函数 ϕ 实际上定义了一种从低维空间的样本点到高维空间 $\mathcal{F}$ 的映射，我们也把这个高维空间 $\mathcal{F}$ 称为特征空间（Feature space）

$$\phi : \mathcal{X} \mapsto \mathcal{F},\ \mathcal{F} = \mathrm{span}(\{K_{\boldsymbol{x}}\}_{\boldsymbol{x} \in \mathcal{X}}) \subset \mathcal{H}_K$$

其中 $\mathcal{H}_K$ 的子空间 $\mathcal{F}$ 也是一个 Hilbert 空间。

和一开始通过给定相关的映射 ϕ 生成特征空间 $\mathcal{F}$ 不同，我们通过给定正定的对称函数 K 分解得到的 $L_2(\mathcal{X})$ 上的单位正交基生成 $\mathcal{H}_K$，并通过其边缘函数得到低维空间到特征空间 $\mathcal{F}$ 的映射 ϕ 。这个过程在核技巧中是自然的，因为我们在很多时候会给定一个对称矩阵 K 来考虑与之关联的映射 ϕ，而不是给出 ϕ 的显示表达。

函数 ϕ 通常是很难求解的，且数据通过 ϕ 映射到的特征空间的维度甚至有可能为无穷维，因而我们在很多情况下不会通过求解 ϕ 来解决特征空间中相关的问题。我们可以类比向量空间，尽管我们完全不知道基所具有的形式，但是在很多情况下如果我们只考虑特征空间 $\mathcal{F}$ 中的内积运算的情况下考虑核函数 K 足以，因为 K 实质上已经控制了 $\mathcal{F}$ 中所有的内积运算。

附录 D　奇异值分解与 PCA

D.1　正交投影

现实生活中的数据常常是种类丰富且结构复杂的。实际采集的数据不可避免地出现维数较高的问题，如一套大小为 256×256 的图片，一本长达百万字的小说集，一些采样率为 22050Hz 的语音等等。高维数据的处理常常是我们需要面对的重大课题。不幸的是，高维数据分析和可视化难度远高于地位数据，但是令人可喜的是，通常来讲，高维数据中仅有少之又少的维度保留了主要的信息，这表明，利用高维向量来表示数据通常存在大量的冗余。能否通过建立高维空间到子空间的映射，来实现数据在低维空间的保留信息尽可能多的表示呢?

D.1.1　向量在线性空间上的投影

我们在下面讨论的 Euclidean 空间如不加说明均为 $(\mathbb{R}^n, \langle\cdot,\cdot\rangle)$，其中 $\langle\cdot,\cdot\rangle$ 为标准内积:

$$\langle \boldsymbol{x}, \boldsymbol{y} \rangle = \boldsymbol{x}^T \boldsymbol{y}$$

讨论向量空间到子空间的一种特殊的线性映射。向量空间 V 到其子空间 U 上的投影（Projection）是满足如下条件的线性映射 π[1]

$$\pi : V \mapsto U, \pi^2 = \pi \circ \pi = \pi$$

显然，线性空间 U 是 π 的不变子空间。设 π 对应的线性映射矩阵为 $\boldsymbol{P}_\pi$，对应地 $\boldsymbol{P_\pi}$ 应当满足

[1]Marc Peter Deisenroth, A Aldo Faisal, Cheng Soon Ong, *Mathematics for Machine Learning*, p. 82

$$\boldsymbol{P}_{\boldsymbol{\pi}}^{\boldsymbol{2}} = \boldsymbol{P}_{\boldsymbol{\pi}}$$

为了衡量向量 $\boldsymbol{x}$ 因投影产生的损失，即投影到向量自身的距离，定义映射 π 损失为

$$L(\pi) := \|\boldsymbol{x} - \pi(\boldsymbol{x})\|_2$$

设 U 到 V 的全体投影的集合为 P，即

$$P := \{\pi : V \mapsto U \mid \pi^2 = \pi\}$$

寻找损失最小的投影等价于求解以下优化问题

$$\pi_U := \arg\min_{\pi \in P} L(\pi) = \arg\min_{\pi \in P} L(\pi)^2 = \arg\min_{\pi \in P} \|\boldsymbol{x} - \pi(\boldsymbol{x})\|_2^2$$

当 $\dim(V) = 2,\ \dim(U) = 1$ 时，该损失具有鲜明的几何解释，即过原点的直线外的一点（这里假定 $\boldsymbol{x}$ 不在 U 中）到直线上的一点的距离。显然当连线与直线垂直时该距离最短。对于维度更高的空间，这个结论在某种意义上也是成立的。设 U 的正交补为 $U^\perp$，可以简单证明当且仅当 $\boldsymbol{x} - \pi(\boldsymbol{x}) \in U^\perp$ 时 $L(\pi)$ 取得最小值，即

$$\boldsymbol{x} - \pi_U(\boldsymbol{x}) \in U^\perp$$

对 $\boldsymbol{x}$ 和 $\pi(\boldsymbol{x})$ 分别进行关于 U 和 $U^\perp$ 的正交分解得到

$$\boldsymbol{x} = \boldsymbol{y_1} + \boldsymbol{y_2},\ \boldsymbol{y_1} \in U,\ \boldsymbol{y_2} \in U^\perp$$

和

$$\pi(\boldsymbol{x}) = \boldsymbol{y_3} + \boldsymbol{y_4} = \boldsymbol{y_3} \in U$$

得到 $\boldsymbol{x}$ 的正交分解

$$\boldsymbol{x} - \pi(\boldsymbol{x}) = (\boldsymbol{y_1} - \boldsymbol{y_3}) + \boldsymbol{y_2},\ \boldsymbol{y_1} - \boldsymbol{y_3} \in U,\ \boldsymbol{y_2} \in U^\perp$$

即有

$$\|\boldsymbol{x}-\pi(\boldsymbol{x})\|_2^2=(\boldsymbol{x}-\pi(\boldsymbol{x}))^T(\boldsymbol{x}-\pi(\boldsymbol{x}))=(\boldsymbol{y_1}-\boldsymbol{y_3})^2+\boldsymbol{y_2}^2\geqslant\boldsymbol{y_2}^2$$

等号成立当且仅当

$$\pi(\boldsymbol{x})=\boldsymbol{y_1}$$

这样的 π 显然满足投影的条件，即 $\pi^2=\pi$，因为对于任意的 $\boldsymbol{x}$

$$\pi^2(\boldsymbol{x})=\pi(\boldsymbol{y_1})=\boldsymbol{y_1}=\pi(\boldsymbol{x})$$

此时

$$\boldsymbol{x}-\pi(\boldsymbol{x})\in U^{\perp}$$

我们同时也得到了

$$\boldsymbol{x}=\pi_U(\boldsymbol{x})+\pi_{U^{\perp}}(\boldsymbol{x})$$

特别地，选择 V 的一组单位正交基 $\{\boldsymbol{b_1},\boldsymbol{b_2},\cdots,\boldsymbol{b_n}\}$，则

$$\boldsymbol{x}=\sum_{i=1}^{n}\pi_{\boldsymbol{b_i}}(\boldsymbol{x})$$

该推论揭示了 $\boldsymbol{x}$ 正交分解的本质，即 $\boldsymbol{x}$ 向两个相互垂直的方向的投影。其满足

$$\|\boldsymbol{x}\|_2^2=\|\pi_U(\boldsymbol{x})+\pi_{U^{\perp}}(\boldsymbol{x})\|_2^2=\|\pi_U(\boldsymbol{x})\|_2^2+\left\|\pi_{U^{\perp}}(\boldsymbol{x})^2\right\|_2^2$$

其在物理学的意义为总能量等于各个分量能量之和。

由于推导没有用到任何除了 $\pi(\boldsymbol{x})\in U$ 的条件，故在保证 $\pi:V\mapsto U$ 的情况下对 π 作为投影的约束的松弛不会影响最终的结论，从而

$$\pi_U=\arg\min_{\pi\in P}\|\boldsymbol{x}-\pi(\boldsymbol{x})\|_2=\arg\min_{\pi\in V^U}\|\boldsymbol{x}-\pi(\boldsymbol{x})\|_2$$

或表述为

$$\pi_U(\boldsymbol{x})=\arg\min_{\boldsymbol{y}\in U}\|\boldsymbol{x}-\boldsymbol{y}\|_2$$

在推导满足条件的 π 之前，我们再证明一个常见的结论，即对于任意矩阵 $\boldsymbol{A}$，其秩存在如下等式

$$\operatorname{rank}(\boldsymbol{A}) = \operatorname{rank}(\boldsymbol{A^T A}) = \operatorname{rank}(\boldsymbol{A^T}) = \operatorname{rank}(\boldsymbol{A A}^T)$$

设 $\boldsymbol{A}$ 为 $m \times n$ 矩阵，由于矩阵的行秩等于列秩，故

$$\operatorname{rank}(\boldsymbol{A}) = \operatorname{rank}(\boldsymbol{A^T})$$

下面证明 $\operatorname{rank}(\boldsymbol{A}) = \operatorname{rank}(\boldsymbol{A^T A})$，从而得到 $\operatorname{rank}(\boldsymbol{A^T}) = \operatorname{rank}(\boldsymbol{A A}^T)$。只需证明方程组 $\boldsymbol{Ax} = \boldsymbol{0}$ 和 $\boldsymbol{A^T Ax} = \boldsymbol{0}$ 是同解的，从而由 $\operatorname{rank}(\boldsymbol{A}) = n - \dim(\operatorname{Null}(\boldsymbol{A}))$ 得到两个矩阵的秩相等。容易得到

$$\boldsymbol{Ax} = \boldsymbol{0} \Rightarrow \boldsymbol{A^T Ax} = \boldsymbol{0}$$

借助二次型的思想

$$\boldsymbol{A^T Ax} = \boldsymbol{0} \Rightarrow \boldsymbol{x}^T \boldsymbol{A^T Ax} = \boldsymbol{0} \Rightarrow \|\boldsymbol{Ax}\|_2 = \boldsymbol{0} \Rightarrow \boldsymbol{Ax} = \boldsymbol{0}$$

从而结论完全得证。

由此我们选择 U 的一组合适的基，进行 $\boldsymbol{P_\pi}$ 的求解[2]。设 $\dim(V) = n$, $\dim(U) = r$,取 U 的一组基为 $\{\boldsymbol{a_1}, \boldsymbol{a}_2, \cdots, \boldsymbol{a_r}\}$,设 $\pi_U(\boldsymbol{x})$ 在基 $\{\boldsymbol{a_1}, \boldsymbol{a_2}, \cdots, \boldsymbol{a_r}\}$ 下的坐标 $\boldsymbol{\lambda}$，则

$$\pi_U(\boldsymbol{x}) = \boldsymbol{P_\pi x} = \sum_{i=1}^{r} \lambda_i \boldsymbol{a_i} = \boldsymbol{A\lambda}$$

由我们上面证明的取得最优解的正交条件得到

$$\langle \boldsymbol{a_i}, \boldsymbol{x} - \pi_U(\boldsymbol{x}) \rangle = \boldsymbol{a_i^T}(\boldsymbol{x} - \pi_U(\boldsymbol{x})) = \boldsymbol{a_i^T}(\boldsymbol{x} - \pi_U(\boldsymbol{x})) = 0,\ i = 1, 2, \cdots, r$$

即有

$$\boldsymbol{A^T}(\boldsymbol{x} - \pi_U(\boldsymbol{x})) = \boldsymbol{A^T}(\boldsymbol{x} - \boldsymbol{A\lambda}) = \boldsymbol{0}$$

移项得到

[2]Marc Peter Deisenroth, A Aldo Faisal, Cheng Soon Ong, *Mathematics for Machine Learning*, pp.85-87

$$\boldsymbol{A^T A\lambda = A^T x}$$

由于 $\{\boldsymbol{a_1}, \boldsymbol{a}_2, \cdots, \boldsymbol{a_r}\}$ 是线性无关的，因此 $\text{rank}(A) = r$，由 **Lemma 1.3** 得到 $\text{rank}(A) = \text{rank}(\boldsymbol{A^T A}) = r$ 从而 $r \times r$ 方阵 $\text{rank}(\boldsymbol{A^T A})$ 是可逆的。于是

$$\boldsymbol{\lambda} = (\boldsymbol{A^T A})^{-1}\boldsymbol{A^T x}$$

$$\pi_U(\boldsymbol{x}) = \boldsymbol{P_\pi x} = \boldsymbol{A\lambda} = \boldsymbol{A}(\boldsymbol{A^T A})^{-1}\boldsymbol{A^T x}$$

此时

$$\boldsymbol{P_\pi} = \boldsymbol{A}(\boldsymbol{A^T A})^{-1}\boldsymbol{A^T}$$

特别地，取 $\{\boldsymbol{a_1}, \boldsymbol{a_2}, \cdots, \boldsymbol{a_r}\}$ 为标准正交基得到

$$\pi_U(\boldsymbol{x}) = \boldsymbol{AA^T x}$$

$$\boldsymbol{\lambda} = \boldsymbol{A^T x}$$

最优解 π_U 被称为正交投影（Orthogonal Projections)。在通常情况下，如果不加说明，则投影一般指正交投影。

于是结合前面得到的推论，我们得到当矩阵 $\boldsymbol{A}$ 列满秩时，以下优化问题存在唯一解

$$\begin{aligned}\min_{\boldsymbol{y}\in U} \|\boldsymbol{x} - \boldsymbol{y}\|_2 &= \min_{\boldsymbol{z}} \|\boldsymbol{x} - \boldsymbol{Az}\|_2 \\ &= \pi_{\text{span}(\boldsymbol{a_1}, \boldsymbol{a_2}, \cdots, \boldsymbol{a_r})}(\boldsymbol{x}) = \boldsymbol{A}(\boldsymbol{A^T A})^{-1}\boldsymbol{A^T x}\end{aligned}$$

且取到最优解时坐标 $\boldsymbol{z}^*$ 满足

$$\boldsymbol{z}^* = (\boldsymbol{A^T A})^{-1}\boldsymbol{A^T b}$$

特别地，取 $\{\boldsymbol{a_1}, \boldsymbol{a_2}, \cdots, \boldsymbol{a_r}\}$ 为标准正交基

$$\boldsymbol{z}^* = \boldsymbol{A^T b}$$

D.1.2　向量在仿射空间上的投影

当 $n = 2$ 时，正交投影将向量映射为向量在原点的直线上的投影，很自然地想到将这个定义推广到一般直线的情况，于是我们将正交投影的定义进行如下推广。先定义关于向量 $\boldsymbol{x}_0$ 和子空间 U 的仿射空间（Affine Space）L[3]

$$L := \{\boldsymbol{x} + \boldsymbol{x}_0 \mid \boldsymbol{x} \in U\},\ \boldsymbol{x}_0 \in U$$

其中子空间 U 被称为仿射空间的方向空间（Direction space），向量 $\boldsymbol{x_0}$ 被称为仿射空间 L 的支撑点（Support point）。向量空间 V 到其仿射空间 L 上的正交投影是满足如下条件的映射：

$$\pi_L : V \mapsto L,\ \pi_L(\boldsymbol{x}) = \boldsymbol{x_0} + \pi_U(\boldsymbol{x} - \boldsymbol{x_0})$$

由于仿射函数 $\boldsymbol{Ax} + \boldsymbol{b}$ 的偏置 $\boldsymbol{b}$ 实际上确定了一个仿射函数集合的等价类，当前仅当一个仿射函数在这个等价类上时该仿射函数才能将空间上的其他向量映到支撑点为 $\boldsymbol{b}$ 的仿射空间上。因而寻找最佳的仿射函数实际上和寻找最佳的线性映射在某种意义上是等价的，因为两个问题之间可以通过加减偏置 $\boldsymbol{b}$ 相互转化。以下部分我们将探讨正交投影的几个具体的应用。

D.1.3　Gram-Schmidt 正交化

Gram-Schmidt 正交化（Gram -Schmidt orthogonalization）在许多和正交性相关的定理证明中都有所应用，也是我们接下来部分证明的关键。正交化使得对于 n 维空间中任意一组基 $\{\boldsymbol{a_1}, \boldsymbol{a_2}, \cdots, \boldsymbol{a_n}\}$ 可通过如下算法将其转化为标准正交基：

初始化

$$\boldsymbol{b_1} := \boldsymbol{a_1}$$

迭代

$$\boldsymbol{b_{i+1}} := \boldsymbol{a_{i+1}} - \pi_{\text{span}(\boldsymbol{b_1}, \cdots, \boldsymbol{b_i})}(\boldsymbol{a_{i+1}}),\ i = 1, 2, \cdots, n-1$$

[3]Marc Peter Deisenroth, A Aldo Faisal, Cheng Soon Ong, *Mathematics for Machine Learning*, pp.61-62

归一化

$$\boldsymbol{c_i} := \boldsymbol{b_i} / \langle \boldsymbol{b_i}, \boldsymbol{b_i} \rangle = \boldsymbol{b_i} / \left\| \boldsymbol{b_i} \right\|_2^2$$

最终得到的 $\{\boldsymbol{c_1}, \boldsymbol{c_2}, \cdots, \boldsymbol{c_n}\}$ 即为所求的一组标准正交基。这是由于

$$\boldsymbol{a_{i+1}} - \pi_{\mathrm{span}(\boldsymbol{b_1}, \cdots, \boldsymbol{b_i})}(\boldsymbol{a_{i+1}}) \in \mathrm{span}(\boldsymbol{b_1}, \cdots, \boldsymbol{b_i})^{\perp}$$

因此

$$\boldsymbol{b_{i+1}} \perp \boldsymbol{b_j},\ j = 1, 2, \cdots, i$$

成立。由于 scaling 不会改变向量之间的正交性，且

$$\langle \boldsymbol{c_i}, \boldsymbol{c_i} \rangle = \left\| \boldsymbol{b_i} \right\|_2^2 / \left\| \boldsymbol{b_i} \right\|_2^2 = 1$$

因此 $\{\boldsymbol{c_1}, \boldsymbol{c_2}, \cdots, \boldsymbol{c_n}\}$ 为一组标准正交基。

D.1.4　最小二乘思想与矩阵伪逆

正交投影还与最小二乘法之间存在紧密联系。对于线性映射矩阵 $\boldsymbol{A} \in \mathbb{R}^{n \times m}$ 和 m 维向量 $\boldsymbol{b}$ 而言，线性方程组

$$\boldsymbol{Ax} = \boldsymbol{b}$$

不一定有非零解，即 $\boldsymbol{b}$ 不在 $\boldsymbol{A}$ 的列空间 $\mathrm{Col}(\boldsymbol{A})$ 上。考虑以下两种特殊情况：

情况一　对于未知数个数等于方程的个数，此时系数矩阵为方阵。假定 $\boldsymbol{A}$ 是满秩的，则方程组由 Cramer 法则是有唯一解的，容易解得

$$\boldsymbol{x} = \boldsymbol{A^{-1}b}$$

情况二　对于超定的线性方程组，即 $n > m$ 且 $\boldsymbol{A}$ 列满秩的情况下，方程组无解。此时为了得到一个近似程度较好的解，我们通常求解相距 $\boldsymbol{A}$ 的列空间 Euclidean 距离 $\boldsymbol{b}$ 最近的一个向量，即向量 $\boldsymbol{Ax}$ 满足损失

$$\boldsymbol{x}^* := \arg\min_{\boldsymbol{x}} \left\| \boldsymbol{Ax} - \boldsymbol{b} \right\|_2 = \arg\min_{\boldsymbol{x}} \left\| \boldsymbol{Ax} - \boldsymbol{b} \right\|_2^2$$

最小。该解被称为最小二乘解（Leastsquare solution），这种求解近似解的方法被称为最小二乘法（Leastsquare method）。平方损失具有非常优秀的数学性质，我们由列满秩的条件可得 $\boldsymbol{A}$ 的列向量为 $\mathrm{Col}(\boldsymbol{A})$ 的一组基，可以推得方程 $\boldsymbol{Ax}=\boldsymbol{b}$ 的最小二乘解 $\boldsymbol{x}^*$ 存在唯一的闭式解[4]

$$\boldsymbol{x}^*=(\boldsymbol{A}^{\boldsymbol{T}}\boldsymbol{A})^{-1}\boldsymbol{A}^{\boldsymbol{T}}\boldsymbol{b}$$

仿照系数矩阵为方阵的情况，我们可以类似地定义列满秩矩阵 $\boldsymbol{A}$ 的左逆，尽管该矩阵不一定是方阵。定义矩阵 $\boldsymbol{A}$ 的伪逆（Peudo-inverse）为

$$\boldsymbol{A}^{+}:=(\boldsymbol{A}^{\boldsymbol{T}}\boldsymbol{A})^{-1}\boldsymbol{A}^{\boldsymbol{T}}$$

伪逆存在当且仅当 $\boldsymbol{A}$ 是列满秩的。这是因为当 $\boldsymbol{A}$ 列满秩时，设 $\boldsymbol{A}\in\mathbb{R}^{m\times n}$，则 [5]

$$\mathrm{rank}(\boldsymbol{A})=\mathrm{rank}(\boldsymbol{A}^{\boldsymbol{T}}\boldsymbol{A})=m$$

因而 $\boldsymbol{A}^{\boldsymbol{T}}\boldsymbol{A}$ 可逆，故伪逆存在；同理，当伪逆存在时，$\boldsymbol{A}^{\boldsymbol{T}}\boldsymbol{A}$ 可逆，也能同样地推出 $\boldsymbol{A}$ 是列满秩的。由于

$$\boldsymbol{x}^{\boldsymbol{T}}(\boldsymbol{A}^{\boldsymbol{T}}\boldsymbol{A})\boldsymbol{x}=\|\boldsymbol{Ax}\|_2^2\geqslant 0$$

方阵 $\boldsymbol{A}^{\boldsymbol{T}}\boldsymbol{A}$ 必然是半正定的。但实际上，在 $\boldsymbol{A}$ 列满秩的情况下对称方阵 $\boldsymbol{A}^{\boldsymbol{T}}\boldsymbol{A}$ 是正定的，因为对于任意的非 $\boldsymbol{0}$ 的 $\boldsymbol{x}$，此时 $\boldsymbol{A}$ 的列满秩限制了 $\boldsymbol{A}$ 的零空间为 $\{\boldsymbol{0}\}$，故 $\boldsymbol{Ax}\neq 0$，因而二次型

$$\boldsymbol{x}^{\boldsymbol{T}}(\boldsymbol{A}^{\boldsymbol{T}}\boldsymbol{A})\boldsymbol{x}=\|\boldsymbol{Ax}\|_2^2>0$$

即 $\boldsymbol{A}^{\boldsymbol{T}}\boldsymbol{A}$ 是正定的。

联系前文，矩阵的伪逆显然满足如下几条性质：

性质一　当矩阵 $\boldsymbol{A}$ 的伪逆存在时，有

$$\boldsymbol{A}^{+}\boldsymbol{A}=\boldsymbol{I}$$

特别地，当矩阵为方阵时，伪逆和矩阵的逆相同，即

[4] 见 D.1.1 结尾部分提供的结论

[5] 见 D.1.1 中间部分提供的秩等式

$$\boldsymbol{A}^{+}=\boldsymbol{A}^{-1}$$

更特别地，当矩阵为正交矩阵时

$$\boldsymbol{A}^{+}=\boldsymbol{A}^{\boldsymbol{T}}$$

性质二　正交投影可以用矩阵的伪逆表示

$$\pi_{\mathrm{span}(\boldsymbol{a_1},\boldsymbol{a_2},\cdots,\boldsymbol{a_n})}(\boldsymbol{x})=\boldsymbol{A}\boldsymbol{A}^{+}\boldsymbol{x}$$

因而对于超定方程组 $\boldsymbol{Ax}=\boldsymbol{b}$，其最小二乘解可以表示为

$$\boldsymbol{x}^{*}=\boldsymbol{A}^{+}\boldsymbol{x}$$

在实际应用场景中，$\boldsymbol{A}$ 列满秩的条件不总是能够满足，因而在求取 $\boldsymbol{A}^{\boldsymbol{T}}\boldsymbol{A}$ 的逆前常常会加上一个微小的扰动项（Jitter term），我们可以基于以下简单的定理对扰动项进行设计，即对于任意实对称矩阵 $\boldsymbol{A}$ 而言，总存在 $k>0$ 使得矩阵

$$\boldsymbol{A}(\boldsymbol{k})=\boldsymbol{A}+k\boldsymbol{I}$$

是正定的。特别地，对于半正定矩阵而言，k 可以任意小。我们只需要考虑对实对称矩阵 $\boldsymbol{A}$ 进行特征值分解

$$\boldsymbol{A}=\boldsymbol{V}\boldsymbol{\Lambda}\boldsymbol{V}^{\boldsymbol{T}}$$

考虑函数

$$\boldsymbol{A}(\boldsymbol{k})=\boldsymbol{V}\boldsymbol{\Lambda}\boldsymbol{V}^{\boldsymbol{T}}+k\boldsymbol{V}\boldsymbol{V}^{\boldsymbol{T}}=\boldsymbol{V}(\boldsymbol{\Lambda}+k\boldsymbol{I})\boldsymbol{V}^{\boldsymbol{T}}$$

上式实际上是矩阵 $\boldsymbol{A}(\boldsymbol{k})$ 的特征值分解。我们只需要找到 k，使得 $\boldsymbol{\Lambda}+k\boldsymbol{I}$ 所有对角线上的特征值非负，从而可以通过线性代换证明 $\boldsymbol{A}(\boldsymbol{k})$ 的正定性。设 $\boldsymbol{A}$ 最小特征值为 λ_m，则 k 只需满足 $k+\lambda_m>0$，即 $k>-\lambda_m$。因而 $k>0$ 总是存在的。对于半正定矩阵而言 $\lambda_m=0$，因而对所有的 k 条件均成立。由矩阵秩的性质可得保证 $\boldsymbol{A}(\boldsymbol{k})$ 的正定性实际上也保证了 $\boldsymbol{A}(\boldsymbol{k})$ 的可逆性，因为此时特征值分解中对角阵是满秩的。

我们已经知道了 $\boldsymbol{A^T A}$ 是半正定的了，因而我们可以向我们需要求逆的矩阵 $\boldsymbol{A^T A}$ 添加扰动项 $k\boldsymbol{I}$ 使得 $\boldsymbol{A^T A}$ 可逆，其中只需 $k > 0$ 即可。同时注意到

$$\boldsymbol{A(k)^{-1}} = \boldsymbol{V}(\boldsymbol{\Lambda} + k\boldsymbol{I})^{-1}\boldsymbol{V^T}$$

当 $k \to +\infty$ 时，由于 $\boldsymbol{\Lambda} + k\boldsymbol{I}$ 对角线的元素趋向 $+\infty$，因而其逆趋向于 $\boldsymbol{O}$。为了防止对后续运算的干扰，我们设计的 k 不应过大，通常是一个小于 1 的小项。

扰动项的添加不仅使得 $\boldsymbol{A^T A}$ 可逆且正定的，同时大大增强了数值计算的稳定性。这种技巧在机器学习中经常用到。

最小二乘法为解决多元线性回归问题提供了思路。对于线性相关程度较高的数据对 $\{\boldsymbol{x_n}, y_n\}_n^N$ 可以采用线性函数对其进行拟合。采用最小二乘思想，定义误差为

$$L(\boldsymbol{A}, \boldsymbol{b}) := \sum_{n=1}^{N}(y_n - \boldsymbol{A x_n} - \boldsymbol{b})^2 = \|\boldsymbol{y} - \boldsymbol{AX} - \boldsymbol{b}\|_2^2$$

求解拟合度最好的系数就是在求解以下问题

$$\boldsymbol{A}^*, \boldsymbol{t}^* := \arg\min_{\boldsymbol{A},\boldsymbol{b}} \|\boldsymbol{y} - \boldsymbol{AX} - \boldsymbol{b}\|_2^2$$

由最小二乘解的求解过程，利用在机器学习领域中我们常使用合并的思想处理偏置项

$$\boldsymbol{A'} = (\boldsymbol{A}, \boldsymbol{b})$$
$$\boldsymbol{X'} = \begin{pmatrix} \boldsymbol{X} \\ \boldsymbol{1_N^T} \end{pmatrix}$$

由于扩增一行不会减少行秩，进而不会减少列秩，故 $\boldsymbol{X'}$ 依然是列满秩的。借助无偏置的多元线性回归的结论可得最终结果。于是我们得到了有关数据集 $\{\boldsymbol{x_n}, y_n\}_n^N$ 多元线性回归最佳的最小二乘的拟合结果为

$$(\boldsymbol{A}^*, \boldsymbol{b}^*) = (\boldsymbol{X'^T X'})^{-1}\boldsymbol{X'y}$$

D.1.5　向量组在线性空间上的投影

当由单个向量拓展至向量的集合时，如果我们定义的损失为降维后的向量到原向量 Euclidean 距离的平方和，由于正交投影是单个向量损失的最优

解，因而也是多个向量损失之和的最优解。设 $\dim(V)=n$，$\dim(U)=r$，取 U 上的一组正交基为 $\{\boldsymbol{b_1},\boldsymbol{b_2},\cdots,\boldsymbol{b_r}\}$，对于 V 中给定的含有 m 个列向量的矩阵

$$\boldsymbol{X}=(\boldsymbol{x_1},\boldsymbol{x_2},\cdots,\boldsymbol{x_m})\in\mathbb{R}^{n\times m}$$

定义 U 中的含有 m 个列向量的矩阵 $\tilde{\boldsymbol{X}}$ 到 $\boldsymbol{X}$ 的损失为向量之间 Euclidean 距离的平方和，即

$$L(\tilde{\boldsymbol{X}})=\sum_{i=1}^{m}\|\boldsymbol{x_i}-\tilde{\boldsymbol{x}}_{\boldsymbol{i}}\|_2^2=\left\|\boldsymbol{X}-\tilde{\boldsymbol{X}}\right\|_F^2$$

设 $\tilde{\boldsymbol{x}}_{\boldsymbol{j}}$ 在基 $\{\boldsymbol{b_1},\boldsymbol{b_2},\cdots,\boldsymbol{b_m}\}$ 下的坐标 $\boldsymbol{z_j}$，即

$$\tilde{\boldsymbol{x}}_{\boldsymbol{j}}=\sum_{i=1}^{m}z_{ij}\boldsymbol{b_i}=\boldsymbol{B}\boldsymbol{z_j}$$
$$\tilde{\boldsymbol{X}}=\boldsymbol{B}\boldsymbol{Z}$$

由于

$$\begin{aligned}\boldsymbol{X}^*&=\arg\min_{\tilde{\boldsymbol{X}}}\sum_{i=1}^{m}\|\boldsymbol{x_i}-\tilde{\boldsymbol{x}}_{\boldsymbol{i}}\|_2^2=\sum_{i=1}^{m}\arg\min_{\tilde{\boldsymbol{x}}\in U}\|\boldsymbol{x_i}-\tilde{\boldsymbol{x}}\|_2^2\\&=\sum_{i=1}^{m}\arg\min_{\boldsymbol{z}}\|\boldsymbol{x_i}-\boldsymbol{B}\boldsymbol{z}\|_2^2\end{aligned}$$

结合 D.1.1 的最后的结论得到损失最小化问题的最优解为

$$\boldsymbol{X}^*=\{\pi_U(\boldsymbol{x_i})\}_{i=1}^m=\{\pi_{\mathrm{span}(\boldsymbol{b_1},\boldsymbol{b_2},\cdots,\boldsymbol{b_r})}(\boldsymbol{x_i})\}_{i=1}^m=\boldsymbol{B}\boldsymbol{B}^{\boldsymbol{T}}\boldsymbol{X}=:\pi_U(\boldsymbol{X})$$

坐标满足

$$\boldsymbol{Z}^*=\{\boldsymbol{B}\boldsymbol{x_i}\}_{i=1}^m=\boldsymbol{B}^{\boldsymbol{T}}\boldsymbol{X}$$

对于高维数据，我们通常需要借助映射来将数据映射至低维，而这种由高维到低维的映射通常是不可逆的（在某些特殊的情况下，我们可以建立近似的逆映射，我们由此定义了伪逆），因此这种降维伴随着信息不可逆转的损失。当我们考虑最小化这种信息的损失时，无论是对于单个向量还是向量的集合，正交投影都是最小化损失的最优解。在下一节我们将继续从映射的角度出发，结合矩阵的分解，通过对分解后矩阵的分析，得出矩阵的一些重要性质。

D.2　奇异值分解

如果将矩阵对向量的作用理解为一种线性映射，那么矩阵乘法就可以理解为线性映射的叠加。在线性映射对向量作用的过程中，向量有时被拉伸或是收缩（Scaling），有时维度增加（Augmentation）或者是降低（Reduction），有时向某个方向转过了一个角度（Rotation）。为了对线性映射更加深入地研究，将叠加在一起的作用分离开，线性映射的分解常常是我们关心的问题。线性映射的分解等价于矩阵的乘性分解，如何对于任意的线性映射矩阵进行分解呢？以及通过这个分解我们能获得线性映射的哪些信息呢？是我们这节所关注的内容。

D.2.1　方阵的特征值分解

在本节的开始，我们引入谱定理（Spectral theorem）。我们接下来无论是在附录的补充材料还是在正文中还会提到很多和“谱”有关的名词，我们实际上对这个词不陌生。在高中化学我们对于化合物的分析会用到红外光谱和核磁共振氢谱，在高中物理中我们在分析发光物体时提到了物体的发射光谱，在人工智能导论中我们在对语音特征进行提取时提到了语音的频谱图，那么究竟什么是谱？我们可以简单地谱的思想就是将信号的一种简单元素线性组合从而更好地反映信号的某种特征[6]。这样的简单元素通常要满足线性无关的性质（有时我们要求他们是正交的），因而这些分解出的简单元素可以视为信号的基，谱对应的变换实际上是一种空间变换，将信号从时域或空域到对应的变换域（例如变换是傅里叶变换，则变换域是频域）。对于一段语言信号而言，这种表示方式可以是信号的傅里叶变换和小波变换；对于一个实对称矩阵而言，这种表示方式可以是矩阵的特征值分解，这就是我们接下来讨论的谱定理就是基于实对称矩阵的特征值分解产生的。

对于矩阵 $\boldsymbol{A}$ 而言，若存在可逆矩阵 $\boldsymbol{V}$ 和对角阵 $\boldsymbol{\Lambda}$，使得 $\boldsymbol{A}$ 可以表示为

$$\boldsymbol{A} = \boldsymbol{V}\boldsymbol{\Lambda}\boldsymbol{V}^{-1}$$

则称这种形式的分解为特征值分解（Eigenvalue decomposition）。

由于实对称矩阵各个特征值对应的特征空间相互正交，且对于实对称矩阵的 k 重特征值，其特征空间的维数为 k，即特征值几何重数和代数重数总

[6] https://zhuanlan.zhihu.com/p/81502804

是相等的。故可以在实对称矩阵的每个特征空间选择一组单位正交的特征向量 $\{\boldsymbol{v}_1, \boldsymbol{v}_2, \cdots, \boldsymbol{v}_n\}$ 并将这些特征向量组成特征向量矩阵 $\boldsymbol{V}$，则 $\boldsymbol{V}$ 为正交矩阵。取特征值 λ_i

$$\boldsymbol{A}\boldsymbol{v_i} = \lambda_i \boldsymbol{v_i} = \boldsymbol{v_i}\lambda_i$$

得到

$$\boldsymbol{AV} = \boldsymbol{V\Lambda} \Rightarrow \boldsymbol{A} = \boldsymbol{V\Lambda V^{-1}} = \boldsymbol{V\Lambda V^T} = \sum_{i=1}^{n} \lambda_i \boldsymbol{v_i v_i^T}$$

因而任意实对称矩阵 $\boldsymbol{A}$ 均可进行特征值分解，且可以规定特征向量矩阵 $\boldsymbol{V}$ 为正交矩阵，这一定理被称为谱定理。我们可以证明特征值分解在某种程度上具有唯一性，即若 $\boldsymbol{A}$ 可进行特征值分解，则分解得到的特征值降序序列是唯一的。先将矩阵 $\boldsymbol{A}$ 进行特征值分解

$$\boldsymbol{A} = \boldsymbol{V\Lambda V^{-1}}$$

假设存在对角阵 $\boldsymbol{\Sigma}$ 和可逆矩阵 $\boldsymbol{U}$

$$\boldsymbol{A} = \boldsymbol{U\Sigma U^{-1}}$$

则有

$$\boldsymbol{AU} = \boldsymbol{U\Sigma}$$

设对角阵的第 i 行元素为 σ_i，$\boldsymbol{U}$ 第 i 列对应的向量为 $\boldsymbol{u}_i$ 得到

$$\boldsymbol{A}\boldsymbol{u}_i = \sigma_i \boldsymbol{u}_i$$

这表明 σ_i 是 $\boldsymbol{A}$ 的特征值，而 $\boldsymbol{A}$ 的特征值降序序列是由 $\boldsymbol{A}$ 对应的特征多项式唯一确定的，因此得到的 $\boldsymbol{\Sigma} = \boldsymbol{\Lambda}$。我们还得到了一旦找到了符合矩阵特征值分解形式的式子，我们事实上已经确定了矩阵的特征值和相应的特征向量。

为了讨论问题的方便，我们需要使得特征值分解在某种程度上具有唯一性。今后若不加说明，默认实对称矩阵的特征值分解分解出的特征向量组成的矩阵为正交矩阵。对于任意特征值分解，我们将不加说明的使用 λ_k 代替矩阵的第 k 大的特征值，使用 $\boldsymbol{v_k}$ 代替与之对应的特征向量。

对于一般的矩阵而言，我们还得到了由于矩阵和其转置特征多项式是相等的

$$f_{\boldsymbol{A}}(\lambda) = |\boldsymbol{A} - \lambda \boldsymbol{I}| = \left|(\boldsymbol{A} - \lambda \boldsymbol{I})^T\right| = \left|\boldsymbol{A}^{\boldsymbol{T}} - \lambda \boldsymbol{I}\right| = f_{\boldsymbol{A}^{\boldsymbol{T}}}(\lambda)$$

因而矩阵的特征值降序序列不受转置的影响。

对于矩阵的特征值分解，由于特征向量矩阵 $\boldsymbol{V}$ 可逆（满秩），故其可分解为有限个初等变换矩阵的乘积，由于初等变换不会改变矩阵的秩，故对于任意特征值分解

$$\operatorname{rank}(\boldsymbol{A}) = \operatorname{rank}(\boldsymbol{V}\boldsymbol{\Lambda}\boldsymbol{V}^{-\boldsymbol{1}}) = \operatorname{rank}(\boldsymbol{\Lambda}) = \|\boldsymbol{\Lambda}\|_0$$

即矩阵的秩等于非零特征值的个数。从特征值对角阵我们可以看出矩阵对应的线性映射对向量的拉伸程度，对于拉伸量为 0 的方向的映射是冗余的，因此非零特征值的个数反映了矩阵的冗余程度。

D.2.2　矩阵的奇异值分解

矩阵 $\boldsymbol{A}$ 的特征值正交分解将其对应的线性映射分解为两次 n 维空间的正交变换和一次伸缩变换，且可以通过特征值对角阵的秩或者说非零特征值的个数来确定特征值矩阵的秩。

很自然地想到，对于一般的矩阵 $\boldsymbol{A}$ 是否也存在这样的分解。假设 $\boldsymbol{A}$ 不是方阵，这个结论显然是不成立的；当 $\boldsymbol{A}$ 为方阵的时候，结论会成立吗？很可惜答案是否定的。这是因为一般的矩阵的 k 重特征值，其特征空间维度小于或等于 k，且等号不总是能够取到。我们退而求其次，猜想一种与实对称矩阵特征值分解类似的分解形式

$$\boldsymbol{A} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^{\boldsymbol{T}}$$

其中 $\boldsymbol{U}$ 和 $\boldsymbol{V}$ 为正交矩阵，相当于两次正交变换，而 $\boldsymbol{\Sigma}$ 的非 0 元素仅出现在行标和列标相同的位置，即含有一个对角子矩阵，其余元素均为 0，相当于一次伸缩变换和维度变换。这个想法太大胆了，但是我们观察形式，很自然地联想到，若这种分解存在，则

$$\boldsymbol{A}\boldsymbol{A}^{\boldsymbol{T}} = \boldsymbol{U}(\boldsymbol{\Sigma}\boldsymbol{\Sigma}^{\boldsymbol{T}})\boldsymbol{U}^{\boldsymbol{T}}$$

$$\boldsymbol{A}^{\boldsymbol{T}}\boldsymbol{A} = \boldsymbol{V}(\boldsymbol{\Sigma}^{\boldsymbol{T}}\boldsymbol{\Sigma})\boldsymbol{V}^{\boldsymbol{T}}$$

我们联想到的对称矩阵的特征值分解。由于

$$(\boldsymbol{A}\boldsymbol{A}^{\boldsymbol{T}})^T = \boldsymbol{A}\boldsymbol{A}^{\boldsymbol{T}}$$
$$(\boldsymbol{A}^{\boldsymbol{T}}\boldsymbol{A})^T = \boldsymbol{A}^{\boldsymbol{T}}\boldsymbol{A}$$

矩阵的对称性得证，因而我们实际上利用我们的猜测构造出了对称矩阵的特征值分解式。相较于我们的猜想，我们还需要进一步验证 $\boldsymbol{A}\boldsymbol{A}^{\boldsymbol{T}}$ 和 $\boldsymbol{A}^{\boldsymbol{T}}\boldsymbol{A}$ 中间的特征值对角阵是非负的，且非零元素相等。考虑二次型简单验证对称矩阵满足半正定性

$$f(\boldsymbol{x}) := \boldsymbol{x}^T\boldsymbol{A}\boldsymbol{A}^{\boldsymbol{T}}\boldsymbol{x} = \left\|\boldsymbol{A}^{\boldsymbol{T}}\boldsymbol{x}\right\|_2^2 \geqslant 0$$
$$g(\boldsymbol{x}) := \boldsymbol{x}^T\boldsymbol{A}^{\boldsymbol{T}}\boldsymbol{A}\boldsymbol{x} = \|\boldsymbol{A}\boldsymbol{x}\|_2^2 \geqslant 0$$

为了验证两个对称方阵非零元素相等，再简单证明一个常见的小引理。对于任意的 $m \times n$ 的矩阵 $\boldsymbol{A}$ 和 $n \times m$ 的矩阵 $\boldsymbol{B}$，$\boldsymbol{A}\boldsymbol{B}$ 和 $\boldsymbol{B}\boldsymbol{A}$ 具有相同的非零特征值，且其代数重数也相同，即

$$f_{\boldsymbol{A}\boldsymbol{B}}(\lambda) := |\boldsymbol{A}\boldsymbol{B} - \lambda\boldsymbol{I}| \propto f_{\boldsymbol{B}\boldsymbol{A}}(\lambda) := |\boldsymbol{B}\boldsymbol{A} - \lambda\boldsymbol{I}|$$

证明利用线性代数中常见的套路，考虑两种方式的分块矩阵的初等变换，其中 $\sigma \neq 0$

$$\begin{pmatrix} \sigma\boldsymbol{I}_n & \boldsymbol{B} \\ \boldsymbol{A} & \sigma\boldsymbol{I}_m \end{pmatrix} \to \begin{pmatrix} \sigma\boldsymbol{I}_n & \boldsymbol{B} \\ \boldsymbol{O} & \sigma\boldsymbol{I}_m - \frac{1}{\sigma}\boldsymbol{A}\boldsymbol{B} \end{pmatrix}$$
$$\begin{pmatrix} \sigma\boldsymbol{I}_n & \boldsymbol{B} \\ \boldsymbol{A} & \sigma\boldsymbol{I}_m \end{pmatrix} \to \begin{pmatrix} \sigma\boldsymbol{I}_n - \frac{1}{\sigma}\boldsymbol{B}\boldsymbol{A} & \boldsymbol{O} \\ \boldsymbol{A} & \sigma\boldsymbol{I}_m \end{pmatrix}$$

由于初等行列变换不改变行列式的值，取行列式后得到

$$\sigma^n \left|\sigma\boldsymbol{I}_m - \frac{1}{\sigma}\boldsymbol{A}\boldsymbol{B}\right| = \sigma^m \left|\sigma\boldsymbol{I}_n - \frac{1}{\sigma}\boldsymbol{B}\boldsymbol{A}\right|$$

得到

$$(-1)^m\sigma^{n-m}\left|\boldsymbol{A}\boldsymbol{B} - \sigma^2\boldsymbol{I}\right| = (-1)^n\sigma^{m-n}\left|\boldsymbol{B}\boldsymbol{A} - \sigma^2\boldsymbol{I}\right|$$

取 $\lambda = \sigma^2$，得到 $f_{\boldsymbol{A}\boldsymbol{B}}(\lambda)$ 和 $f_{\boldsymbol{B}\boldsymbol{A}}(\lambda)$ 成比例。对于负特征值这里取 σ 为相应的复数即可，这里我们在机器学习基础这门课程里破天荒地考虑了复数（大

家也不需要对复数有抗拒的心理，其实复数的结论很多时候只是实数的简单拓展而已）。

特别地，当 $n = m$ 时由于特征多项式 $f_{\boldsymbol{AB}}(\lambda)$ 和 $f_{\boldsymbol{BA}}(\lambda)$ 的次数是相等的，这迫使 $\boldsymbol{AB}$ 和 $\boldsymbol{BA}$ 也具有相同的 0 特征值和代数重数。

我们现在已经对我们对任意矩阵分解的想法已经很有信心了，现在我们把我们的想法用数学语言正式地表达出来。对任意矩阵 $\boldsymbol{A}$ 均存在分解

$$\boldsymbol{A} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^T$$

其中 $\boldsymbol{U}$ 和 $\boldsymbol{V}$ 为正交矩阵，而 $\boldsymbol{\Sigma}$ 的非 0 元素仅出现在行标和列标相同的位置。$\boldsymbol{U}$ 的列向量被称为左奇异向量（Left-sigular vector），$\boldsymbol{V}^T$ 被称为右奇异向量（Right-sigular vector），$\boldsymbol{\Sigma}$ 对角线上的值被称为奇异值（Sigular value）。这种矩阵的分解方式被称为奇异值分解（Sigular value decomposition / SVD）。

要验证这个想法，只需证明

$$\boldsymbol{A}\boldsymbol{V} = \boldsymbol{U}\boldsymbol{\Sigma}$$

对 $\boldsymbol{A}^{\boldsymbol{T}}\boldsymbol{A}$ 进行特征值分解

$$\boldsymbol{A}^{\boldsymbol{T}}\boldsymbol{A} = \boldsymbol{V}\boldsymbol{\Lambda}_2\boldsymbol{V}^{\boldsymbol{T}}$$

由于 $\boldsymbol{A}^{\boldsymbol{T}}\boldsymbol{A}$ 的特征值非负，设其非零特征值为

$$\lambda_1 \geqslant \lambda_2 \geqslant \cdots \geqslant \lambda_r > 0$$

并将其非零的特征值开平方根作为奇异值，设奇异值为

$$\sigma_1 = \sqrt{\lambda_1} \geqslant \sigma_2 = \sqrt{\lambda_2} \geqslant \cdots \geqslant \sigma_r = \sqrt{\lambda_r} > 0$$

取非零奇异值对应的 $\boldsymbol{V}$ 的列向量为右奇异向量，此时 $\boldsymbol{V}$ 即为分解对应的右侧的正交矩阵 $\{\boldsymbol{v}_1, \boldsymbol{v_2}, \cdots, \boldsymbol{v_r}\}$，于是只需证明：

$$\boldsymbol{A}\boldsymbol{v_i} = \sigma_i \boldsymbol{u_i}$$

构造左奇异向量

$$\boldsymbol{u_i} := \frac{1}{\sigma_i}\boldsymbol{A}\boldsymbol{v_i}$$

下面证明我们构造的左奇异向量满足正交条件

$$\langle \boldsymbol{u_i}, \boldsymbol{u_j} \rangle = {\boldsymbol{u_i}}^T \boldsymbol{u_j} = \frac{1}{\sigma_i \sigma_j} \boldsymbol{v_i^T} (\boldsymbol{A^T A v_j}) = \frac{\lambda_j}{\sigma_i \sigma_j} \boldsymbol{v_i^T v_j} = \begin{cases} 1 & i = j \\ 0 & i \neq j \end{cases}$$

通过 Gram-Schmidt 正交化 [7] 将左奇异向量扩充为 n 个单位正交向量，组成左侧的正交矩阵 $\boldsymbol{U}$，于是奇异值分解的存在性得证。[8]

而我们发现 $\boldsymbol{U}$ 中左奇异向量 $\boldsymbol{u_i}$，$i = 1, 2, \cdots, n$ 不是由 $\boldsymbol{AA^T}$ 分解得到的列向量定义的，我们接下来证明左奇异向量可以为 $\boldsymbol{AA^T}$ 特征值分解得到的正交矩阵 $\boldsymbol{U}$ 的列向量

$$\boldsymbol{AA^T} = \boldsymbol{U} \boldsymbol{\Lambda}_1 \boldsymbol{U^T}$$

其对应的的特征值为 λ_i。设 $1 \leqslant i \leqslant r$，由于

$$\boldsymbol{AA^T u_i} = \frac{1}{\sigma_i} \boldsymbol{A} (\boldsymbol{A^T A v_i}) = \frac{\lambda_i}{\sigma_i} (\boldsymbol{A v_i}) = \sigma_i (\sigma_i \boldsymbol{u_i}) = \lambda_i \boldsymbol{u_i}$$

得到 $\boldsymbol{u_i}$ 为 $\boldsymbol{U}$ 特征值为 λ_i 的列向量。对于 $i > r$ 的部分，由于 $\boldsymbol{U}$ 保证了正交性，特征向量 $\boldsymbol{u_i}$ 也符合要求左奇异矩阵的要求。

进一步地，我们证明奇异值分解同样具有特征值分解的很多性质。在完成存在性的证明后，我们证明奇异值分解在某种程度上也具有唯一性。考虑 $\boldsymbol{A^T A}$ 的特征值分解

$$\boldsymbol{A^T A} = \boldsymbol{V' \Lambda V'^T}$$

再结合 $\boldsymbol{A}$ 的奇异值分解

$$\boldsymbol{A^T A} = \boldsymbol{V} (\boldsymbol{\Sigma^T \Sigma}) \boldsymbol{V}^T$$

矩阵 $\boldsymbol{A^T A}$ 的特征值分解得到的特征值降序序列是唯一的 [9]，由此得到

$$\boldsymbol{\Sigma^T \Sigma} = \boldsymbol{\Lambda}$$

[7] 详见 D.1.3

[8] Marc Peter Deisenroth, A Aldo Faisal, Cheng Soon Ong, *Mathematics for Machine Learning*, pp.122-125

[9] 详见 D.2.1

因而对于非零的奇异值 σ_i，其对应着特征值 λ_i，在对角线上有

$$\sigma_i^2 = \lambda_i$$

得到矩阵 $\boldsymbol{A}$ 的奇异值降序序列是唯一的。我们还可以类似地利用 D.2.1 证明一旦我们构造得到了奇异值分解式，我们事实上已经找到了 $\boldsymbol{AA^T}$ 和 $\boldsymbol{A^TA}$ 的特征值和特征向量。

为了讨论的方便，今后我们将不加说明的使用 σ_k 代替矩阵的第 k 大的奇异值，使用 $\boldsymbol{u_k}$ 和 $\boldsymbol{v_k}$ 分别代替与之对应的左右奇异向量。

结合奇异值降序序列的唯一性，考虑 $\boldsymbol{A}$ 的奇异值分解

$$\boldsymbol{A} = \boldsymbol{U\Sigma V^T}$$

$$\boldsymbol{A^T} = \boldsymbol{V\Sigma^T U^T}$$

得到矩阵 $\boldsymbol{A}$ 和 $\boldsymbol{A^T}$ 的奇异值降序序列的非零值相等。

由于可逆的正交变换不会改变矩阵的秩，因而有

$$\operatorname{rank}(\boldsymbol{A}) = \operatorname{rank}(\boldsymbol{U\Sigma V^T}) = \operatorname{rank}(\boldsymbol{\Sigma}) = \|\boldsymbol{\Sigma}\|_0$$

矩阵 $\boldsymbol{A}$ 的秩与非零奇异值的个数相同。

最后考察方阵的特征值和奇异值分解之间的联系。对 $\boldsymbol{A}$ 进行特征值分解得到

$$\boldsymbol{A} = \boldsymbol{P\Lambda P^T} \Rightarrow \boldsymbol{A^TA} = \boldsymbol{P\Lambda^2 P^T}$$

由 D.2.1 得到这种分解得到的特征值降序序列是唯一的，即有 $\boldsymbol{A^TA}$ 分解得到的对角阵为 $\boldsymbol{\Lambda^2}$，考虑 $\boldsymbol{A}$ 奇异值和 $\boldsymbol{A^TA}$ 的特征值的关系得到奇异值矩阵和 $\boldsymbol{\Lambda}$ 对应位置的绝对值对应相等。因而事实上对于半正定的对称矩阵而言，方阵的特征值分解是矩阵的奇异值分解的一个特例。

D.2.3　奇异值分解的映射视角

奇异值分解为我们提供了一种全新的看待矩阵对应的映射的方式。由映射的复合与矩阵乘法的联系，可以将矩阵 $\boldsymbol{A}$ 的左乘视为线性映射 $\mathcal{A}$

$$\mathcal{A} : V \mapsto U,\ \mathcal{A}(\boldsymbol{x}) = \boldsymbol{Ax}$$

奇异值分解将任意映射分解为两个空间上的正交变换 $\mathcal{V}$ 和 $\mathcal{U}$ 和夹在两次正交变换中间的维度变换和伸缩变换 $\mathcal{S}$

$$\mathcal{V}: V \mapsto V,\ \mathcal{V}(\boldsymbol{x}) = \boldsymbol{V^T x}$$

$$\mathcal{S}: V \mapsto U,\ \mathcal{S}(\boldsymbol{x}) = \boldsymbol{\Sigma x}$$

$$\mathcal{U}: U \mapsto U,\ \mathcal{U}(\boldsymbol{x}) = \boldsymbol{U x}$$

映射 $\mathcal{A}$ 被分解为

$$\mathcal{A} = \mathcal{U} \circ \mathcal{S} \circ \mathcal{V}$$

映射 $\mathcal{V}$ 和 $\mathcal{U}$ 都是可逆的，对 $\boldsymbol{\Sigma}$ 分两种情况讨论：

情况一　当 $\boldsymbol{\Sigma}$ 的列不满秩时，映射 $\mathcal{S}$ 的作用将使得 $\boldsymbol{x}$ 的某些维度置零，这样的映射对 $\boldsymbol{x}$ 造成的信息缺失是不可逆的，因而映射 $\mathcal{A}$ 具有不可逆的性质。同样地还可以联想到如果是对于向量的集合 $\boldsymbol{X}$，在映射 $\mathcal{A}$ 分解得到的 $\mathcal{U}$ 和 $\mathcal{V}$ 的作用下行之间的线性相关关系不会改变，但在 $\mathcal{S}$ 的作用下，某些行被伸缩而某些行被置为 0，因而 $\boldsymbol{X}$ 行秩必然不会上升。由于 $\mathcal{A}$ 的作用等价于 $\boldsymbol{A}$ 的左乘，从这个角度看，矩阵的乘法使得 $\boldsymbol{X}$ 行秩不升，因而矩阵的乘法不会使得 $\boldsymbol{X}$ 的秩上升，这为我们理解矩阵乘法不会使得矩阵的秩上升提供了一种新的视角。

情况二　当 $\boldsymbol{\Sigma}$ 的列满秩时映射 $\mathcal{S}$ 的作用保留了 $\boldsymbol{x}$ 的信息，这是因为我们显然可以通过与映射对应的降维和伸缩变换还原矩阵的信息。此时，我们尝试恢复映射 $\mathcal{A}$ 的作用，得到映射 $\mathcal{A}$ 的左逆 $\mathcal{A}^+$，其对应的分解为

$$\mathcal{A}^+ = \mathcal{V}^{-1} \circ \mathcal{S}^{-1} \circ \mathcal{U}^{-1}$$

将线性映射 $\mathcal{S}$ 对应的矩阵 $\boldsymbol{\Sigma}$ 表示为

$$\boldsymbol{\Sigma} = \begin{pmatrix} \boldsymbol{\Sigma}_r \\ \boldsymbol{O} \end{pmatrix}$$

联系该映射的含义，$\mathcal{S}^{-1}$ 对应的矩阵可以表示为

$$\boldsymbol{\Sigma}^+ = \begin{pmatrix} \boldsymbol{\Sigma}_r^{-1} & \boldsymbol{O} \end{pmatrix}$$

两个正交变换的逆对应的矩阵只需在原矩阵的基础上取转置即可，得到 $\mathcal{A}^+$ 对应的矩阵为 $\boldsymbol{V\Sigma^+U^T}$，此时

$$\mathcal{A}^+\mathcal{A}=\mathcal{I}$$

因而我们构造得到了矩阵的一个左逆。

根据情形二我们构造得到了列满秩矩阵的一个左逆。在 D.1.4 我们得到的矩阵伪逆也是矩阵的左逆。而列满秩矩阵的左逆是唯一的，因为假设 $\boldsymbol{B}$ 存在左逆 $\boldsymbol{A_1}$ 和 $\boldsymbol{A_2}$ ，我们假设 $\boldsymbol{B}$ 的列数为 n，则

$$\boldsymbol{A_1B}=\boldsymbol{A_2B}=\boldsymbol{I}\Rightarrow(\boldsymbol{A_1}-\boldsymbol{A_2})\boldsymbol{B}=\boldsymbol{O}\Rightarrow\{\boldsymbol{b_i}\}_{i=1}^n\subset\mathrm{Null}(\boldsymbol{A_1}-\boldsymbol{A_2})$$

从而 $\boldsymbol{B}$ 的列向量都在 $\boldsymbol{A_1}-\boldsymbol{A_2}$ 的零空间中，而 $\boldsymbol{B}$ 是列满秩的，故得到

$$\dim(\mathrm{Null}(\boldsymbol{A_1}-\boldsymbol{A_2}))\geqslant\mathrm{rank}(\{\boldsymbol{b_i}\}_{i=1}^n)=n$$

而零空间维度和矩阵的秩之间存在关系

$$\dim(\mathrm{Null}(\boldsymbol{A_1}-\boldsymbol{A_2}))=n-\mathrm{rank}(\boldsymbol{A_1}-\boldsymbol{A_2})\leqslant n$$

这迫使

$$\mathrm{rank}(\boldsymbol{A_1}-\boldsymbol{A_2})=0\Rightarrow\boldsymbol{A_1}-\boldsymbol{A_2}=\boldsymbol{O}\Rightarrow\boldsymbol{A_1}=\boldsymbol{A_2}$$

从而得到左逆唯一。故我们通过这个方式推导的左逆事实上就是矩阵的伪逆，我们得到了矩阵伪逆的奇异值分解

$$\boldsymbol{A^+}=\boldsymbol{V\Sigma^+U^T}$$

不难观察到 $\boldsymbol{A^T}$ 和 $\boldsymbol{A^+}$ 的奇异值分解在形式上是相似的，唯一的区别在于对于中间的奇异值方阵中非零奇异值而言，$\boldsymbol{A^+}$ 取了个倒数。

从映射的视角看，矩阵 $\boldsymbol{A}$ 对向量的伸缩作用主要反映在其奇异值矩阵 $\boldsymbol{\Sigma}$ 上。矩阵对向量的伸缩程度可以视为映射的一种特征，反映了在空间的不同基下 $\boldsymbol{A}$ 实际发挥的作用。这有些类似特征值的几何意义，对于任意可进行特征值分解的方阵 $\boldsymbol{A}$，考虑其对向量 $\boldsymbol{x}$ 的作用。设其特征值分解为选取其线性无关的特征向量作为空间的一组基，利用这组正交基表示 $\boldsymbol{x}$

$$\boldsymbol{x}=\sum_{i=1}^{N} z_i \boldsymbol{v_i}=\boldsymbol{V}\boldsymbol{z} \Rightarrow \boldsymbol{A}\boldsymbol{x}=\sum_{i=1}^{N} z_i \boldsymbol{A}\boldsymbol{v_i}=\sum_{i=1}^{N} \lambda_i z_i \boldsymbol{v_i}$$

矩阵 $\boldsymbol{A}$ 对 $\boldsymbol{x}$ 的作用使得 $\boldsymbol{x}$ 在 $\boldsymbol{A}$ 特征向量方向上的分量进行伸缩，伸缩的比例为对应的特征值。

对奇异值分解而言，若我们更关注伸缩对向量长度最终的效果，我们可以淡化正交变换对向量方向的影响。为了更加直观地衡量矩阵对向量的这种伸缩作用，我们需要定义一种新的度量，类比向量的 L^2 范数，矩阵的谱范数（Spectual norm）定义为

$$\|\boldsymbol{A}\|_2 := \max_{\boldsymbol{x}} \frac{\|\boldsymbol{A}\boldsymbol{x}\|_2}{\|\boldsymbol{x}\|_2}$$

结合范数的定义，我们可以将表达式写为

$$\|\boldsymbol{A}\|_2 = \max_{\boldsymbol{x}} \left\| \boldsymbol{A}\frac{\boldsymbol{x}}{\|\boldsymbol{x}\|_2} \right\|_2 = \max_{\|\boldsymbol{x}\|_2=1} \|\boldsymbol{A}\boldsymbol{x}\|_2$$

矩阵的谱范数的几何意义很直观，其表示矩阵 $\boldsymbol{A}$ 将单位超球上的向量伸缩至离原点最远的距离。考虑转化优化函数为

$$\max_{\boldsymbol{x}} \frac{\|\boldsymbol{A}\boldsymbol{x}\|_2}{\|\boldsymbol{x}\|_2} = \max_{\boldsymbol{x}} \sqrt{\frac{\|\boldsymbol{A}\boldsymbol{x}\|_2^2}{\|\boldsymbol{x}\|_2^2}} = \max_{\boldsymbol{x}} \sqrt{\frac{\boldsymbol{x^T}\boldsymbol{A^T}\boldsymbol{A}\boldsymbol{x}}{\boldsymbol{x^T}\boldsymbol{x}}}$$

取 $\boldsymbol{A}$ 的奇异值分解得到

$$\boldsymbol{A^T}\boldsymbol{A} = \boldsymbol{V}(\boldsymbol{\Sigma^T}\boldsymbol{\Sigma})\boldsymbol{V^T} = \boldsymbol{V}\boldsymbol{\Lambda}\boldsymbol{V^T}$$

为了化简目标函数，取 $\boldsymbol{A^T}\boldsymbol{A}$ 的单位正交的特征向量作为空间的一组基，利用这组正交基表示 $\boldsymbol{x}$

$$\boldsymbol{x}=\sum_{i=1}^{n} z_i \boldsymbol{v_i}=\boldsymbol{V}\boldsymbol{z}$$
$$\boldsymbol{z}=\boldsymbol{V^T}\boldsymbol{x}$$

得到

$$\begin{aligned}\boldsymbol{x^T}\boldsymbol{A^T}\boldsymbol{A}\boldsymbol{x} &= \left(\sum_{i=1}^{n} z_i \boldsymbol{v_i}\right)\boldsymbol{A^T}\boldsymbol{A}\left(\sum_{j=1}^{n} z_j \boldsymbol{v_j}\right) = \sum_{i=1}^{n}\sum_{j=1}^{n} z_i z_j \boldsymbol{v_i^T}\boldsymbol{A^T}\boldsymbol{A}\boldsymbol{v_j} \\ &= \sum_{i=1}^{n}\sum_{j=1}^{n} \sigma_j^2 z_i z_j \boldsymbol{v_i^T}\boldsymbol{v_j} = \sum_{i=1}^{n} \sigma_j^2 z_i^2 \leqslant \sigma_1^2 \boldsymbol{z^T}\boldsymbol{z} = \sigma_1^2 \boldsymbol{x^T}\boldsymbol{x}\end{aligned}$$

该式表明，单位超球上的向量沿单位正交特征向量分解得到的各个分量决定了向量长度最终被伸缩的程度。上式当 $\boldsymbol{x} = \boldsymbol{v_1}$ 取等号。得到谱范数的值为

$$\|\boldsymbol{A}\|_2 = \sqrt{\max_{\boldsymbol{x}} \frac{\boldsymbol{x^T A^T A x}}{\boldsymbol{x^T x}}} = \sigma_1$$

即矩阵 $\boldsymbol{A}$ 的谱范数等于矩阵 $\boldsymbol{A}$ 的最大奇异值 σ_1。

这里我们引出了 Rayleigh 商（Rayleigh quotient）的概念，当 $\boldsymbol{A}$ 为实对称矩阵，$\boldsymbol{x}$ 取值为实向量时，函数

$$R(\boldsymbol{A}, \boldsymbol{x}) = \frac{\boldsymbol{x^T A x}}{\boldsymbol{x^T x}}$$

被称为 Rayleigh 商[10]，该函数在实向量空间中连续。由于对向量 $\boldsymbol{x}$ 的 scaling 实际上不会影响 Rayleigh 商的取值，因而我们考虑 Rayleigh 商取最值对应的优化问题时我们可以添加对 $\boldsymbol{x}$ 模长为 1 的约束

$$\max_{\|\boldsymbol{x}\|_2=1} (\min_{\|\boldsymbol{x}\|_2=1}) \boldsymbol{x^T A x}$$

构造 Lagrangian 函数

$$L(\boldsymbol{x}, \lambda) = \boldsymbol{x^T A x} + \lambda(1 - \boldsymbol{x^T x})$$

求导得到

$$\frac{\partial L}{\partial \boldsymbol{x}} = 2(\boldsymbol{A x} - \lambda \boldsymbol{x}) = 0 \Rightarrow \boldsymbol{A x} = \lambda \boldsymbol{x}$$

这表明极值总是在特征向量处取到，其对应的值为

$$R(\boldsymbol{A}, \boldsymbol{x}) = \frac{\boldsymbol{x^T A x}}{\boldsymbol{x^T x}} = \lambda$$

因而 Rayleigh 商取值介于最大特征值和最小特征值之间，当且仅当取到对应的特征向量时取到最值。

D.2.4　矩阵的低秩估计

对于矩阵的分析，奇异值分解也为我们提供了一种新的视角，取

$$\boldsymbol{U_r} := (\boldsymbol{u}_1, \boldsymbol{u}_2, \cdots, \boldsymbol{u}_r) \in \mathbb{R}^{m \times r}$$

[10] https://en.wikipedia.org/wiki/Rayleigh_quotient

$$\boldsymbol{\Sigma_r} := \mathrm{diag}(\sigma_1, \sigma_2, \cdots, \sigma_r) \in \mathbb{R}^{r\times r}$$

$$\boldsymbol{V_r} := (\boldsymbol{v}_1, \boldsymbol{v}_2, \cdots, \boldsymbol{v}_r) \in \mathbb{R}^{n\times r}$$

$$\boldsymbol{A_i} = \sigma_i \boldsymbol{u_i} \boldsymbol{v_i^T},\ i = 1, 2, \cdots, n$$

由矩阵分块的思想，奇异值分解也可以写成

$$\boldsymbol{A} = \sum_{i=1}^{n} \sigma_i \boldsymbol{u_i} \boldsymbol{v_i^T} = \sum_{i=1}^{r} \sigma_i \boldsymbol{u_i} \boldsymbol{v_i^T} = \sum_{i=1}^{r} \boldsymbol{A_i} = \boldsymbol{U_r} \boldsymbol{\Sigma_r} \boldsymbol{V_r^T}$$

这种形式的优点之一在于中间的对角矩阵 $\boldsymbol{\Sigma_r}$ 的大小直接反映的矩阵 $\boldsymbol{A}$ 的秩 r 的大小。矩阵 $\boldsymbol{A_i}$ 实际上代表了矩阵 $\boldsymbol{A}$ 的奇异值对应的矩阵成分，我们可以定义这种成分的累加

$$\hat{\boldsymbol{A}}(k) := \sum_{i=1}^{k} \boldsymbol{A_i}$$

由于 $k > r$ 后对应奇异值为 0，故我们只讨论 $k \leqslant r$ 的情况。考虑 $\hat{\boldsymbol{A}}(k)$ 的奇异值分解

$$\hat{\boldsymbol{A}}(k) = \sum_{i=1}^{k} \boldsymbol{A_i} = \sum_{i=1}^{k} \sigma_i \boldsymbol{u_i} \boldsymbol{v_i^T} = \boldsymbol{U_k} \boldsymbol{\Sigma_k} \boldsymbol{V_k^T} = \boldsymbol{U} \begin{pmatrix} \boldsymbol{\Sigma_k} & \\ & \boldsymbol{O} \end{pmatrix} \boldsymbol{V^T}$$

得到 $\hat{\boldsymbol{A}}(k)$ 的秩满足

$$\mathrm{rank}(\hat{\boldsymbol{A}}(k)) = k$$

不难想象 $\hat{\boldsymbol{A}}(k)$ 可以作为 $\boldsymbol{A}$ 的近似，称为 k 秩近似（k-rank approximation）[11]，这种近似实际上将矩阵以另一种参数量更少的方式进行表示，因而在某种程度上可以视为数据在低维度的表示

$$\hat{\boldsymbol{A}}(k) = \sum_{i=1}^{k} \sigma_i \boldsymbol{u_i} \boldsymbol{v_i^T}$$

这种近似表示的求解兼顾时间和空间的效率。我们实际上仅使用了 k 组左奇异向量 $\boldsymbol{u_i}$，右奇异向量 $\boldsymbol{v_i}$ 和奇异值 σ_i 就可以利用 $\hat{\boldsymbol{A}}(k)$ 近似地表示矩阵 $\boldsymbol{A}$。相较于矩阵的元素个数而言，当取 k 的值较小时，参数量大大减少

[11] Marc Peter Deisenroth, A Aldo Faisal, Cheng Soon Ong, *Mathematics for Machine Learning*, pp.129-131

了，且由 在 k 比较小时，计算前 k 个特征值和相应的特征向量的速度时比较快的。这样的近似表示可以用于图像的有损压缩，但是由于从压缩信号到真实信号所需要用到的矩阵需要单独存储，因而在实际应用中我们不会考虑这种做法[12]。

这种近似在映射方面的某种程度上的最优性由 Eckart-Young 定理(Eckart-Young Theorem) [13] 给出，即

$$\hat{\boldsymbol{A}}(k) = \arg\min_{\boldsymbol{B}} \|\boldsymbol{A}-\boldsymbol{B}\|_2,\ s.t.\ \operatorname{rank}(\boldsymbol{B}) \leqslant k$$

对于未接触矩阵论的同学而言，证明具有一定技巧性。考虑 $\boldsymbol{A}-\hat{\boldsymbol{A}}(k)$ 的奇异值分解

$$\boldsymbol{A}-\hat{\boldsymbol{A}}(k) = \boldsymbol{U}\left(\boldsymbol{\Sigma} - \begin{pmatrix} \boldsymbol{\Sigma_k} & \\ & \boldsymbol{O} \end{pmatrix}\right)\boldsymbol{V^T}$$

由于谱范数等于最大奇异值 [14]，我们得到

$$\left\|\boldsymbol{A}-\hat{\boldsymbol{A}}(k)\right\|_2 = \sigma_{k+1}$$

假设存在另一个符合条件的矩阵 $\boldsymbol{B}$，满足

$$\|\boldsymbol{A}-\boldsymbol{B}\|_2 < \left\|\boldsymbol{A}-\hat{\boldsymbol{A}}(k)\right\|_2 = \sigma_{k+1},\ \operatorname{rank}(\boldsymbol{B}) \leqslant k$$

则由谱范数的定义得到

$$\forall \boldsymbol{x},\ \|(\boldsymbol{A}-\boldsymbol{B})\boldsymbol{x}\|_2^2 < \sigma_{k+1}\|\boldsymbol{x}\|_2$$

为了简化问题，推出矛盾，取 $\boldsymbol{x} \in \operatorname{Null}(\boldsymbol{B})$，此时

$$\dim(\operatorname{Null}(\boldsymbol{B})) = n - \operatorname{rank}(\boldsymbol{B}) \geqslant n-k$$

$$\|(\boldsymbol{A}-\boldsymbol{B})\boldsymbol{x}\|_2^2 = \|\boldsymbol{A}\boldsymbol{x}\|_2^2 < \sigma_{k+1}^2\|\boldsymbol{x}\|_2^2$$

考虑 $\operatorname{Null}(\boldsymbol{B})$ 的补空间，考虑 $\boldsymbol{A}$ 前 $k+1$ 个奇异值对应的右奇异向量张成的子空间，设子空间中的向量 $\boldsymbol{y}$ 坐标表示为

[12] 现实中经典的图像有损压缩算法为 JPEG 图像压缩算法，我们在 6.3 提到了它

[13] Marc Peter Deisenroth, A Aldo Faisal, Cheng Soon Ong, *Mathematics for Machine Learning*, pp.131-132

[14] 详见 D.2.3

$$\forall \boldsymbol{y} \in \operatorname{span}(\boldsymbol{v_1}, \boldsymbol{v_2}, \cdots, \boldsymbol{v_{k+1}}),\ \boldsymbol{y} = \sum_{i=1}^{k+1} z_i \boldsymbol{v_i}$$

$$\|\boldsymbol{Ay}\|_2^2 = \boldsymbol{y^T A^T Ay} = \sum_{i=1}^{k+1} \sigma_{k+1}^2 z_i^2 \geqslant \sigma_{k+1}^2 \|\boldsymbol{z}\|_2^2$$

这证明了两个空间确实是互补的。设

$$V_1 = \operatorname{Null}(\boldsymbol{B}),\ \dim(V_1) \geqslant n - k$$

$$V_2 = \operatorname{span}(\boldsymbol{v_1}, \boldsymbol{v_2}, \cdots, \boldsymbol{v_{k+1}}),\ \dim(V_2) = k + 1$$

则

$$V_1 \cap V_2 = \{\boldsymbol{0}\}$$

由维度公式得到

$$\dim(V_1 + V_2) = \dim(V_1) + \dim(V_2) - \dim(V_1 \cap V_2) \geqslant n + 1$$

由于 $V_1 + V_2$ 是 $\mathbb{R}^n$ 的子空间，因而上式推出了矛盾。从而

$$\hat{\boldsymbol{A}}(k) = \arg\min_{\boldsymbol{B}} \|\boldsymbol{A} - \boldsymbol{B}\|_2,\ s.t.\ \operatorname{rank}(\boldsymbol{B}) \leqslant k$$

D.2.5　奇异值分解的现代计算方法

我们在 D.2.3 探讨了 $\boldsymbol{A}$ 对向量单次作用的性质。很自然地想到可以沿着这个思路，研究 $\boldsymbol{A}$ 迭代作用于一个向量 $\boldsymbol{x}$ 最终的作用效果（为了防止向量被过度伸缩，在每轮迭代过程中我们可能需要对其进行归一化）。上述以特征向量为基的思想对可以进行特征值分解的 $\boldsymbol{A}$ 的研究可能会有所启发。对于任意可对角化的方阵 $\boldsymbol{A}$，设其特征值分解为

$$\boldsymbol{A} = \boldsymbol{V\Lambda V^{-1}}$$

对于任意向量 $\boldsymbol{x_0} \notin \operatorname{Null}(\boldsymbol{A})$，定义向量序列

$$\boldsymbol{x_n} := \frac{\boldsymbol{Ax_{n-1}}}{\|\boldsymbol{Ax_{n-1}}\|_2},\ n = 1, 2, \cdots$$

我们研究 $n \to \infty$ 时，$\boldsymbol{x_n}$ 的收敛性质。设 $\boldsymbol{A} \in \mathbb{R}^{N\times N}$ 展开序列表达式，由范数的定义得到

$$\begin{aligned}\boldsymbol{x_n} &= \frac{\boldsymbol{Ax_{n-1}}}{\|\boldsymbol{Ax_{n-1}}\|_2} = \frac{\|\boldsymbol{Ax_{n-2}}\|_2}{\|\boldsymbol{A^2x_{n-2}}\|_2}\frac{\boldsymbol{A^2x_{n-2}}}{\|\boldsymbol{Ax_{n-2}}\|_2} \\ &= \frac{\boldsymbol{A^2x_{n-2}}}{\|\boldsymbol{A^2x_{n-2}}\|_2} = \cdots = \frac{\boldsymbol{A^nx_0}}{\|\boldsymbol{A^nx_0}\|_2}\end{aligned}$$

为了化简表达式，由于 $\boldsymbol{V}$ 是可逆的，选取其线性无关的特征向量作为空间的一组基，利用这组正交基表示 $\boldsymbol{x_0}$

$$\boldsymbol{x_0} = \sum_{i=1}^{N} z_i\boldsymbol{v_i} = \boldsymbol{Vz}$$

因此

$$\begin{aligned}\boldsymbol{A^nx_0} &= \sum_{i=1}^{N} z_i\boldsymbol{A^nv_i} = \sum_{i=1}^{N} \lambda_i z_i\boldsymbol{A^{n-1}v_i} \\ &= \cdots = \sum_{i=1}^{N} \lambda_i^n z_i\boldsymbol{v_i} = \lambda_1^n \sum_{i=1}^{N} \left(\frac{\lambda_i}{\lambda_1}\right)^n z_i\boldsymbol{v_i}\end{aligned}$$

设特征值 λ_1 对应特征子空间维数为 D，即存在 D 个特征值等于 λ_1 的线性无关特征向量，此时记

$$\boldsymbol{v} = \sum_{i=1}^{D} \lambda_i z_i\boldsymbol{v_i}$$

则

$$\boldsymbol{A^nx_0} = \lambda_1^n\boldsymbol{v} + \sum_{i=D+1}^{N} \left(\frac{\lambda_i}{\lambda_1}\right)^n z_i\boldsymbol{v_i} = \lambda_1^n\boldsymbol{v} + o$$

得到

$$\lim_{n\to\infty} \boldsymbol{x_n} = \lim_{n\to\infty} \frac{\boldsymbol{A^nx_0}}{\|\boldsymbol{A^nx_0}\|_2} = \lim_{n\to\infty} \frac{\boldsymbol{v}+o}{\|\boldsymbol{v}+o\|_2} = \frac{\boldsymbol{v}}{\|\boldsymbol{v}\|_2}$$

通过迭代地将可特征值分解的方阵 $\boldsymbol{A}$ 右乘任意一个不在 $\boldsymbol{A}$ 的零空间的向量 $\boldsymbol{x_0}$ 并进行归一化，我们实际得到了 $\boldsymbol{x_0}$ 在最大特征值对应的特征子空间的归一化后的分量 $\boldsymbol{x}$，借助 Reileigh 商的思想得到

$$\lambda_1 := \frac{\boldsymbol{x^TAx}}{\boldsymbol{x^Tx}}$$

最大特征值和次大特征值分别被称为主特征值（Dominant eigenvalue）和次特征值（Second dominant eigenvalue），两者的比值实际上控制了收敛的速度，当主特征值和次特征值的比值大较时，由于收敛是指数级别的，收敛是非常快的。

传统的求取方阵特征值和特征向量的方法是解方阵的特征多项式的根求取方阵的特征值后，通过解方程得到对应的特征向量。然而 Abel-Ruffini 定理（Abel-Ruffini Theorem）告诉我们五次及五次以上的多项式是不存在通项公式的，这为我们求解高阶方阵的特征值和特征向量带来了巨大的挑战。然而通过以上性质我们可以很方便地求取矩阵的主特征值对应的特征向量，从而求得矩阵的主特征值，这与传统方法恰好是反过来的。这种简单的方法被称为幂迭代法（Power iteration），该方法及其拓展 [15] 被广泛应用于各大计算机线性代数相关的工具包中。幂迭代法通过如下方法求取任意矩阵 $\boldsymbol{A}$ 的主特征值和特征向量：

初始化　随机选取不在 $\boldsymbol{A}$ 的零空间的向量 $\boldsymbol{x}$。

迭代　采用矩阵右乘与归一化的方式更新 $\boldsymbol{x}$ 直至收敛

$$\boldsymbol{x} := \frac{\boldsymbol{A}\boldsymbol{x}}{\|\boldsymbol{A}\boldsymbol{x}\|_2}$$

特征值计算　计算得到矩阵的主特征值

$$\lambda := \frac{\boldsymbol{x}^{\boldsymbol{T}}\boldsymbol{A}\boldsymbol{x}}{\boldsymbol{x}^{\boldsymbol{T}}\boldsymbol{x}}$$

由向量序列的收敛速度的决定因素我们得到幂迭代法的效率很大程度上取决于主特征值和次特征值的比值，该比值越大，收敛越快，效率越高。

我们不太满意这个简单的算法，因为我们如果要对矩阵进行奇异值分解，只能求出主特征值是不够的。在对矩阵 $\boldsymbol{A}$ 进行奇异值分解时，我们首先考虑计算 $\boldsymbol{A}^{\boldsymbol{T}}\boldsymbol{A}$ 的主特征值对应的特征向量（即最大奇异值对应的右奇异向量）$\boldsymbol{v_1}$ 和主特征值 λ_1，开根得到对应的奇异值 σ_1，D.2.2 为我们提供了构造左奇异向量的方法

$$\boldsymbol{u_1} := \frac{1}{\sigma_1}\boldsymbol{A}\boldsymbol{v_1}$$

[15] https://www.cs.cornell.edu/~bindel/class/cs6210-f09/lec26.pdf

当然我们也可以考虑计算 $\boldsymbol{AA^T}$ 主特征值对应的特征向量 $\boldsymbol{u_1}$ 和主特征值 σ_1，类似地我们也可以构造左奇异向量，证明思路是类似的

$$\boldsymbol{v_1} := \frac{1}{\sigma_1}\boldsymbol{A^T u_1}$$

这两种方式究竟选择哪一种是由 $\boldsymbol{A^T A}$ 和 $\boldsymbol{AA^T}$ 的大小决定的，为了减少计算量，我们选择两个对称矩阵中较小的矩阵。我们实际上计算得到的是 $\boldsymbol{A}$ 的 1 秩近似 $\hat{\boldsymbol{A}}(1)$，即矩阵最主要的成分，如果我们想要进一步地求取矩阵的其他成分，受 D.2.4 的启发我们可以考虑从 $\boldsymbol{A}$ 中减去它来更新 $\boldsymbol{A}$，这样做的合理性可以构造 $\boldsymbol{A} - \hat{\boldsymbol{A}}(1)$ 的奇异值分解式得到

$$\boldsymbol{A} - \hat{\boldsymbol{A}}(1) = \boldsymbol{U}\left(\boldsymbol{\Sigma} - \begin{pmatrix} \sigma_1 & \\ & \boldsymbol{O} \end{pmatrix}\right)\boldsymbol{V^T}$$

可以看到更新后 $\boldsymbol{A}$ 的最大的奇异值已经被删去了，而其余奇异值保持不变。由此我们可以依次计算出所有的奇异值和奇异向量，从而构造奇异值矩阵和奇异向量矩阵。注意到当奇异值存在相等的情况时奇异向量需要借助前面输出的与之属于相同奇异值的正交向量进行正交化，利用 Gram-Schmidt 正交化 [16] 中迭代和归一化步骤即可。

在特征值分解或奇异值分解的过程中即使是没有对特征值或奇异值进行排序的要求，如果采用这种方法对特征值或奇异值进行求解，最终解得的结果会很自然地呈现降序的状态。另外，如果我们没有必要求得所有的特征值和特征向量，则在我们得到我们想要的特征值和特征向量后，算法可以提前结束。

特别地当 $\boldsymbol{A}$ 是半正定的对称矩阵时，我们可以直接考虑计算 $\boldsymbol{A}$ 的主特征值对应的特征向量和主特征值，此时主特征值就对应着最大奇异值，相应的特征向量同时对应着左右奇异向量，奇异值分解为我们提供了计算 $\boldsymbol{A}$ 的特征值分解的方法，这是因为对于半正定的对称矩阵，奇异值分解和特征值分解结果是可以完全相同的 [17]。

奇异值分解在线性代数的意义是基础性的，由于奇异值分解代表了普适的的矩阵的乘性分解，实际上也代表了一切线性映射的分解。对这种映射分解的分析从另一种角度导出了列满秩矩阵的伪逆，并从矩阵对向量的伸缩作

[16] 详见 D.1.3

[17] 详见 D.2.2 末尾部分

用出发，定义了矩阵的谱范数和矩阵的低秩估计。在下一节，我们将联系上一节的内容，从另外的两个角度再度论证这种低秩估计的合理性。

D.3　PCA

数据在低维空间的投影我们在第一节已经讲了个大概，但是还留下了一点令人感到不满的小尾巴。有关向量组到特定子空间上的最佳的表示我们已经有了结论[18]，但是自然而然地我们就会发问如何选取合适的子空间使得降维的损失最小呢？我们接下来将解决这个问题。

D.3.1　PCA 的最大投影方差视角

上一节我们从映射角度通过 Eckart-Young 定理得到了矩阵的一个最优的低秩表示，这一节的这一个部分我们将转换视角，从方差的角度入手。

在 D.1.1 我们证明了 D 维向量 $\boldsymbol{x}$ 可以经过正交投影沿空间的一组正交基被分解为向 D 个正交向量的投影（即该方向上的分量）之和的形式。对于 N 个向量的组合而成的数据 $\boldsymbol{X}$ 而言，其向每个正交向量的投影组成直线上的一组点集，这样的 D 组点集构成 $\boldsymbol{X}$ 的 D 个独立的成分（Components），其中点集方差最大的（由于方差是不确定性的度量，也是某种意义上保留信息最多的 [19]）成分被成为 $\boldsymbol{X}$ 的第一主成分（First principal component），从 $\boldsymbol{X}$ 中去除第一主成分的分量后得到的与第一主成分方向相互正交的点集方差最大的成分被称为 $\boldsymbol{X}$ 的第二主成分（Second principal component），由此不断地去除点集方差最大的成分，从而依次得到 $\boldsymbol{X}$ 的第 k 主成分（k-th principal component）。我们在进行数据降维的过程实际上就是寻找数据在更少的且最具有代表性的 L 个成分下的表示，由这种思想得到的代表数据的前 L 个主成分的过程被称为主成分分析（Principal component analysis / PCA）。对向量组 $\boldsymbol{X}$ 的 PCA 可表示为如下迭代的过程[20]：

初始化

$$\boldsymbol{X}^{(0)} := \boldsymbol{X}$$

[18] 详见 D.1.5

[19] 详见 F.1.2 末尾部分

[20] Marc Peter Deisenroth, A Aldo Faisal, Cheng Soon Ong, *Mathematics for Machine Learning*, pp.321-325

迭代

$$\boldsymbol{b_t} := \arg\max_{\boldsymbol{b}} \mathrm{Var}[\pi_{\boldsymbol{b}}(\boldsymbol{X^{(t-1)}})] = \arg\max_{\boldsymbol{b}} \frac{1}{N-1}\sum_{i=1}^{N}\left\|\pi_{\boldsymbol{b}}(\boldsymbol{x_i^{(t-1)}}) - \mu_{\boldsymbol{b}}\right\|_2^2$$

$$s.t.\ \ \|\boldsymbol{b}\|_2^2 = 1,\ \boldsymbol{bb_j} = 0, j = 1, 2, \cdots, t-1$$

更新数据

$$\boldsymbol{X^{(t)}} := \boldsymbol{X^{(t-1)}} - \pi_{\boldsymbol{b_t}}(\boldsymbol{X^{(t-1)}})$$

回到初始化部分直至输出特定的前 l 个的主成分后停止。

借助正交投影表达式 [18]，对迭代式

$$\begin{aligned}\boldsymbol{b_t} &:= \arg\max_{\boldsymbol{b}} \frac{1}{N-1}\sum_{i=1}^{N}\|\pi_{\boldsymbol{b}}(\boldsymbol{x_i})\|_2^2 \\ &= \arg\max_{\boldsymbol{b}} \|\pi_{\boldsymbol{b}}(\boldsymbol{X})\|_F^2 = \arg\max_{\boldsymbol{b}} \left\|\boldsymbol{bb^T X}\right\|_F^2 = \arg\max_{\boldsymbol{b}} \mathrm{tr}(\boldsymbol{X^T bb^T X}) \\ &= \arg\max_{\boldsymbol{b}} \boldsymbol{b^T X X^T b},\ s.t.\ \|\boldsymbol{b}\|_2^2 = 1,\ \boldsymbol{b^T b_j} = 0, j = 1, 2, \cdots, t-1\end{aligned}$$

上式是 Rayleigh 商对应的优化问题的形式 [21]，因而第一次迭代过程 $\boldsymbol{b_1}$ 就是 $\boldsymbol{XX^T}$ 最大的特征值对应的特征向量 $\boldsymbol{u_1}$，即 $\boldsymbol{X}$ 最大奇异值对应的左奇异向量。对更新式

$$\boldsymbol{X^{(t)}} := \boldsymbol{X^{(t-1)}} - \boldsymbol{b_t b_t^T X^{(t-1)}}$$

对 $\boldsymbol{X}$ 进行奇异值分解得到

$$\boldsymbol{X} = \boldsymbol{U\Sigma V^T}$$

$$\boldsymbol{b_1 b_1^T X} = (\boldsymbol{u_1}, 0, \cdots, 0)\boldsymbol{\Sigma V^T} = \sigma_1 \boldsymbol{u_1 v_1^T} = \boldsymbol{U}\begin{pmatrix}\sigma_1 & \\ & \boldsymbol{O}\end{pmatrix}\boldsymbol{V^T}$$

于是得到 $\boldsymbol{X^{(1)}}$ 的奇异值分解

$$\boldsymbol{X^{(1)}} = \boldsymbol{U}\left(\boldsymbol{\Sigma} - \begin{pmatrix}\sigma_1 & \\ & \boldsymbol{O}\end{pmatrix}\right)\boldsymbol{V^T}$$

[21] 详见 D.2.3 末尾部分

可以看到更新后 $\boldsymbol{A}$ 的最大的奇异值已经被删去了，而其余奇异值保持不变，更新数据的过程实际上将向量组转化到正交补空间上。从而 PCA 过程的输出输出的正交向量为 $\boldsymbol{XX^T}$ 按照对应特征值降序排序的特征向量序列，和对 $\boldsymbol{XX^T}$ 利用幂迭代法[22]进行特征值分解输出向量的结果是相同的，流程也比较类似。

我们最终得到了主成分对应的一组正交基，将正交基依次排列为矩阵得到的是前 L 个左奇异向量排列而成的矩阵 $\boldsymbol{U_L}$，于是降维后的数据即对应主成分之和可以表示为[18]

$$\begin{aligned}\boldsymbol{X}^* &= \sum_{i=1}^{L} \pi_{\boldsymbol{u_i}}(\boldsymbol{X}) = \pi_{\mathrm{span}(\boldsymbol{u_1},\boldsymbol{u_2},\cdots,\boldsymbol{u_L})}(\boldsymbol{X}) = \boldsymbol{U_L U_L^T X} \\ &= \boldsymbol{U_L \Sigma_L V_L^T} = \hat{\boldsymbol{A}}(\boldsymbol{L})\end{aligned}$$

于是我们从方差的角度我们得到了上一节 k 秩估计的合理性。

实际上在第一节的引言中我们提到了降维的最终目的是获得参数量更少的表示，我们实际上倾向于用点在各个主成分上的坐标表示降维后数据的结果，由正交投影得到子空间的坐标表示公式 [18]

$$\boldsymbol{X_p} = \sum_{i=1}^{L} \boldsymbol{u_i^T X} = \boldsymbol{U_L^T X} = \boldsymbol{\Sigma_L V_L^T}$$

因而正交基 $\boldsymbol{U_L^T}$ 可以视为编码器（Rncoder）将原空间上的数据映向低维空间，这个低维空间被称为主成分空间（Principal space），得到数据的在低维上的表示 $\boldsymbol{X_p}$，再将低维空间上的向量通过解码器（Decoder）映射回原空间，得到高维空间上还原后的保留了 $\boldsymbol{X}$ 大部分信息的数据 $\boldsymbol{X}^*$。

D.3.2　PCA 的最小重构误差视角

主成分的最大方差视角虽然容易求解，但并不是完全符合人的直觉，一种比较直观的定义 $\boldsymbol{X}$ 的主成分的方式是从误差估计的角度出发的，即 $\boldsymbol{X}$ 的主成分和 $\boldsymbol{X}$ 的误差应当尽量小。我们通常使用最小二乘思想来对这个误差进行近似的估计。对于 n 个向量的 d 维组合 $\boldsymbol{X}$，我们希望寻找一个低维的空间，使得这个空间上存在一个和 $\boldsymbol{X}$ 最相近的向量组，我们将这个向量组作为 $\boldsymbol{X}$ 的低维表示。即

[22]详见 D.2.5

$$\begin{aligned}\boldsymbol{X}^* &= \arg\min_{\tilde{\boldsymbol{X}}\in U}\sum_{n=1}^{N}\|\boldsymbol{x_n}-\tilde{\boldsymbol{x}}_{\boldsymbol{n}}\|_2^2 \\ &= \arg\min_{\tilde{\boldsymbol{X}}\in U}\left\|\boldsymbol{X}-\tilde{\boldsymbol{X}}\right\|_F^2,\ s.t.\ \dim(U)\leqslant L\end{aligned}$$

或表示为矩阵的形式

$$\boldsymbol{X}^* = \arg\min_{\tilde{\boldsymbol{X}}}\left\|\boldsymbol{X}-\tilde{\boldsymbol{X}}\right\|_F^2,\ s.t.\ \operatorname{rank}(\tilde{\boldsymbol{X}})\leqslant L$$

选择低秩空间 U 上的一组标准正交基，并利用这组标准正交基表示向量组中的向量对应的近似表示

$$\begin{aligned}\tilde{\boldsymbol{x}}_{\boldsymbol{n}} &:= \sum_{i=1}^{L} z_{in}\boldsymbol{b_i} = \boldsymbol{B}\boldsymbol{z_n} \\ \tilde{\boldsymbol{X}} &= \boldsymbol{B}\boldsymbol{Z}\end{aligned}$$

问题转化为[23]

$$\begin{aligned}\min_{\boldsymbol{B}}\sum_{n=1}^{N}\|\boldsymbol{x_i}-\boldsymbol{B}\boldsymbol{z_n}\|_2^2 = \min_{\boldsymbol{B}}\|\boldsymbol{X}-\boldsymbol{B}\boldsymbol{Z}\|_F^2 &= \min_{\boldsymbol{B}}\operatorname{tr}\left((\boldsymbol{X}-\boldsymbol{B}\boldsymbol{Z})^T(\boldsymbol{X}-\boldsymbol{B}\boldsymbol{Z})\right) \\ &\Rightarrow \min_{\boldsymbol{B}}\operatorname{tr}(\boldsymbol{Z}^{\boldsymbol{T}}\boldsymbol{Z})-2\operatorname{tr}(\boldsymbol{Z}^{\boldsymbol{T}}\boldsymbol{B}^{\boldsymbol{T}}\boldsymbol{X}),\ s.t.\ \boldsymbol{B}^{\boldsymbol{T}}\boldsymbol{B}=\boldsymbol{I_L}\end{aligned}$$

对于固定的空间 U，向量组到一个特定的子空间的最佳的降维结果就是向量组到该子空间的正交投影[18]，即

$$\begin{aligned}\tilde{\boldsymbol{X}} &= \pi_{\operatorname{span}(\boldsymbol{b_1},\boldsymbol{b_2},\cdots,\boldsymbol{b_L})}(\boldsymbol{X}) = \boldsymbol{B}\boldsymbol{B}^{\boldsymbol{T}}\boldsymbol{X} \\ \boldsymbol{Z} &= \boldsymbol{B}^{\boldsymbol{T}}\boldsymbol{X}\end{aligned}$$

代入得到优化问题为

$$\begin{aligned}\min_{\boldsymbol{B}}-\operatorname{tr}(\boldsymbol{Z}^{\boldsymbol{T}}\boldsymbol{Z}) &= \max_{\boldsymbol{B}}\operatorname{tr}(\boldsymbol{Z}^{\boldsymbol{T}}\boldsymbol{Z}) = \max_{\boldsymbol{B}}\operatorname{tr}(\boldsymbol{Z}\boldsymbol{Z}^{\boldsymbol{T}}) \\ &= \max_{\boldsymbol{B}}\operatorname{tr}(\boldsymbol{B}^{\boldsymbol{T}}\boldsymbol{X}\boldsymbol{X}^{\boldsymbol{T}}\boldsymbol{B}),\ s.t.\ \boldsymbol{B}^{\boldsymbol{T}}\boldsymbol{B}=\boldsymbol{I_L}\end{aligned}$$

得到

$$\boldsymbol{X}^* = \boldsymbol{B}^*\boldsymbol{B}^{*\boldsymbol{T}}\boldsymbol{X}$$

[23]周志华，机器学习，清华大学出版社，pp.229-230

其 U 上的坐标表示为

$$\boldsymbol{X_p} := \boldsymbol{Z^*} = \boldsymbol{B^{*T}X}$$

考虑 PCA 对应的以下形式的优化问题

$$\max_{\boldsymbol{B}} \operatorname{tr}(\boldsymbol{B^T\Phi B}),\ s.t.\ \boldsymbol{B^TB} = \boldsymbol{I_L}$$

其中 $\boldsymbol{\Phi}$ 是一个半正定矩阵，这里取的是 $\boldsymbol{XX^T}$。这是一个在数据降维中重要的问题，在流形学习中很大一类问题最终都能归结为这种形式，下面我们尝试对这个问题进行求解。由于相互正交的条件太严格了，并不是很好处理，我们尝试对其进行松弛得到优化问题为[24]

$$\max_{\boldsymbol{B}} \operatorname{tr}(\boldsymbol{B^T\Phi B}) = \max_{\boldsymbol{B}} \sum_{n=1}^{L} \boldsymbol{b_n^T\Phi b_n},\ s.t.\ \boldsymbol{b_n^Tb_n} = 1,\ n = 1, 2, \cdots, L$$

利用 Lagrange 乘数法得到最优解的必要条件，考虑 Lagrangian 函数

$$L(\boldsymbol{B}, \boldsymbol{\lambda}) = \sum_{n=1}^{L} \boldsymbol{b_n^T\Phi b_n} + \sum_{n=1}^{L} \lambda_n(1 - \boldsymbol{b_n^Tb_n})$$

求导并令导函数为 0 得到

$$\frac{\partial L}{\partial \boldsymbol{b_n}} = 2(\boldsymbol{\Phi b_n} - \lambda_n \boldsymbol{b_n}) = 0 \Rightarrow \boldsymbol{\Phi b_n} = \lambda_n \boldsymbol{b_n}$$

从而得到最优解的一个必要条件为 λ_n 为 $\boldsymbol{\Phi}$ 的一个特征值，$\boldsymbol{b_n}$ 是对应的特征向量。考虑到原来的约束条件确定了向量是单位模长的，因而优化函数取值为

$$\sum_{n=1}^{L} \boldsymbol{b_n^T\Phi b_n} = \sum_{n=1}^{L} \lambda_n$$

再加上单位正交条件，于是特征值取相互正交的特征子空间对应的特征值。因而要使得目标函数最大，我们对 $\boldsymbol{\Phi}$ 进行特征值分解后取前 L 个特征值和其对应的特征向量即可。此时 $\boldsymbol{B}$ 的取值为最大的 L 个特征值对应的特征向量组成的矩阵。

[24] https://www.cnblogs.com/gyhhaha/p/11794257.html

同样的方法可以证明将最大化目标函数更换为最小化目标函数，目标函数的最优值变为后 L 个特征值之和，此时 $\boldsymbol{B}$ 的取值为最小的 L 个特征值对应的特征向量组成的矩阵。

对 $\boldsymbol{X}$ 进行奇异值分解，得到

$$\boldsymbol{X} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^{\boldsymbol{T}}$$

考虑特征值分解与奇异值分解的联系，代入优化问题后解得

$$\boldsymbol{B}^* = \boldsymbol{U}_{\boldsymbol{L}}$$

$$\boldsymbol{X}^* = \boldsymbol{B}^*\boldsymbol{B}^{*\boldsymbol{T}}\boldsymbol{X} = \boldsymbol{U}_{\boldsymbol{L}}\boldsymbol{\Sigma}_{\boldsymbol{L}}\boldsymbol{V}_{\boldsymbol{L}}^{\boldsymbol{T}} = \hat{\boldsymbol{A}}(\boldsymbol{L})$$

$$\boldsymbol{X}_{\boldsymbol{p}} = \boldsymbol{B}^{*\boldsymbol{T}}\boldsymbol{X} = \boldsymbol{U}_{\boldsymbol{L}}^{\boldsymbol{T}}\boldsymbol{X} = \boldsymbol{\Sigma}_{\boldsymbol{L}}\boldsymbol{V}_{\boldsymbol{L}}^{\boldsymbol{T}}$$

和最大方差视角得出的结果是一致的，因而从误差的角度我们得到了上一节 k 秩估计的合理性。可以理解为最大方差视角下的 PCA 的优化目标是最大化主成分中保存的信息（能量），而最小误差视角是最小化主成分以外的成分以外的信息（能量），我们上面的这些过程实际上在证明两者的等价性。

D.3.3　PCA 的实现细节

虽然我们以上的证明没有要求 $\boldsymbol{X}$ 需要进行去均值化和归一化操作，但是为了数值计算的稳定性，我们经常在运算前对数据进行去均值操作，有时也会在去均值后对数据的每个维度进行方差归一化。我们也可以在奇异值分解时将计算 $\boldsymbol{X}\boldsymbol{X}^{\boldsymbol{T}}$ 的特征值更换为计算协方差矩阵

$$\boldsymbol{\Gamma}_{\boldsymbol{X}} = \frac{1}{N-1}(\boldsymbol{X} - \boldsymbol{\mu}_{\boldsymbol{X}}\boldsymbol{1}_{\boldsymbol{N}}^{\boldsymbol{T}})(\boldsymbol{X} - \boldsymbol{\mu}_{\boldsymbol{X}}\boldsymbol{1}_{\boldsymbol{N}}^{\boldsymbol{T}})^T$$

的特征值（协方差矩阵在后面细节中会被提及），这是由于从最大方差视角看偏置的添加不会影响方差的大小，且在特征值分解中系数的存在只会影响特征值的大小，不会影响特征向量的求解（这可以通过添加系数后构造矩阵的特征值分解得到），因而最终的结果是不变的。

在 D.2.5 的说明中有提及在 $\boldsymbol{X}$ 的奇异值分解中对于计算 $\boldsymbol{X}\boldsymbol{X}^{\boldsymbol{T}}$ 的特征值还是计算 $\boldsymbol{X}^{\boldsymbol{T}}\boldsymbol{X}$ 的特征值的问题。事实上在 PCA 中，由于我们最终要计算低维空间的坐标表示存在两种等价的表示

$$\boldsymbol{X}_{\boldsymbol{p}} = \boldsymbol{U}_{\boldsymbol{L}}^{\boldsymbol{T}}\boldsymbol{X} = \boldsymbol{\Sigma}_{\boldsymbol{L}}\boldsymbol{V}_{\boldsymbol{L}}^{\boldsymbol{T}}$$

我们也按照类似的方法来决定是计算 $\boldsymbol{X}\boldsymbol{X}^{\boldsymbol{T}}$ 的特征值还是 $\boldsymbol{X}^{\boldsymbol{T}}\boldsymbol{X}$ 的特征值[25]。利用 D.2.5 提供的特征值分解方法，若 $N > D$，则 $\boldsymbol{X}\boldsymbol{X}^{\boldsymbol{T}}$ 计算方便一些，此时我们对 $\boldsymbol{X}^{\boldsymbol{T}}\boldsymbol{X}$ 输出前 L 个特征向量，采用公式

$$\boldsymbol{X_p} = \boldsymbol{U_L^T X}$$

计算低维坐标表示。若 $D > N$，则 $\boldsymbol{X}^{\boldsymbol{T}}\boldsymbol{X}$ 计算方便一些，此时我们对 $\boldsymbol{X}^{\boldsymbol{T}}\boldsymbol{X}$ 输出前 L 个特征向量和特征值，特征值开平方得到奇异值，采用公式

$$\boldsymbol{X_p} = \boldsymbol{\Sigma_L V_L^T}$$

计算低维坐标表示。

由 PCA 的最大方差视角，PCA 中主成分保留的信息的比例实际上可以定义为主成分的方差之和占数据的总方差的比例。对数据进行两次迭代的 PCA，前一次选择 L 个主成分而后一次选择所有的主成分，由前文的相关推导可知这个比例为[26]

$$p(L) = \frac{\sum\limits_{n=1}^{L} \lambda_n}{\sum\limits_{n=1}^{D} \lambda_n}$$

在对降维后的数据维度没有严格要求的场景下，我们可以在得到特征值降序序列后划定好合适的比例（如 0.9）后选择 $p(L)$ 恰好超过该比例的主成分数 L 即可。

D.3.4　PCA 与数据白化

将数据 $\boldsymbol{X}$ 第 d 维的对应的行向量 $\boldsymbol{\chi_d}$ 的每个元素视为从总体抽得的 N 个样本，则其协方差被定义为

$$\begin{aligned}
\mathrm{Cov}[\boldsymbol{\chi_n}, \boldsymbol{\chi_m}] &:= \mathrm{E}[(\boldsymbol{\chi_n} - \mathrm{E}[\boldsymbol{\chi_n}])(\boldsymbol{\chi_m} - \mathrm{E}[\boldsymbol{\chi_n}])] \\
&= \frac{1}{N-1}\sum_{i=1}^{N}(x_{ni} - \bar{x}_n)(x_{mi} - \bar{x}_m) \\
&= \frac{1}{N-1}(\boldsymbol{\chi_n} - \boldsymbol{\mu_{\chi_n}})^T(\boldsymbol{\chi_m} - \boldsymbol{\mu_{\chi_m}})
\end{aligned}$$

[25] Marc Peter Deisenroth, A Aldo Faisal, Cheng Soon Ong, *Mathematics for Machine Learning*, pp.322-325

[26] https://en.wikipedia.org/wiki/Principal_component_analysis

$\boldsymbol{X}$ 的协方差矩阵被定义为

$$\begin{aligned}\mathrm{Cov}[\boldsymbol{X},\boldsymbol{X}] = \boldsymbol{\Gamma_X} &= \{\mathrm{Cov}[\boldsymbol{\chi_n},\boldsymbol{\chi_m}]\}_{n,m=1}^{D} \\ &= \frac{1}{N-1}\{(\boldsymbol{\chi_n}-\boldsymbol{\mu_{\chi_n}})^T(\boldsymbol{\chi_m}-\boldsymbol{\mu_{\chi_m}})\}_{n,m=1}^{D} \\ &= \frac{1}{N-1}(\boldsymbol{X}-\boldsymbol{\mu_X}\boldsymbol{1_N^T})(\boldsymbol{X}-\boldsymbol{\mu_X}\boldsymbol{1_N^T})^T\end{aligned}$$

在机器学习的理论推导中时有时需要假定数据的每个维度是不相关的，然而对于大多数情况，这个假设通常不成立。针对这种情况，我们通常需要对数据进行预处理，以消除数据之间的相关关系。一种简单的相关关系是线性相关关系，且当数据服从正态分布时，数据之间是相关的当前仅当数据之间是线性相关的。我们通常对数据进行线性变换以消除数据的线性相关性，从而某种程度上降低数据之间相关性，这里我们引出了数据白化（Data whitening）的概念。即去均值化的列满秩矩阵 $\boldsymbol{X}$ 的白化变换（Whitening transformation）是一种线性变换，设其线性变换矩阵为 $\boldsymbol{W}$，则

$$\mathrm{Cov}[\boldsymbol{WX},\boldsymbol{WX}] = \boldsymbol{\Gamma_{WX}} = \boldsymbol{I_D}$$

白化使得 $\boldsymbol{X}$ 的每一行线性无关（此时每一行数据的协方差为 0，每一维数据线性相关系数为 0），且方差均为 1。在实际操作过程中，未经过去均值化操作的数据在经过白化前需要进行去均值化操作。为了求解白化矩阵，把 $\boldsymbol{X}$ 视为随机变量 $(X_1, X_2, \cdots, X_D)^T$，其每一行对应一个随机变量，考虑期望的线性性质得到

$$\begin{aligned}\mathrm{E}[\boldsymbol{WX}] &= \mathrm{E}[\boldsymbol{W}(X_1, X_2, \cdots, X_D)^T] \\ &= \sum_{n=1}^{N} \boldsymbol{W_n}\mathrm{E}[(X_1, X_2, \cdots, X_D)^T] = \boldsymbol{W}\mathrm{E}[\boldsymbol{X}] = \boldsymbol{0_D}\end{aligned}$$

从而

$$\boldsymbol{\Gamma_{WX}} = \frac{(\boldsymbol{WX})(\boldsymbol{WX})^T}{N-1} = \boldsymbol{W\Gamma_X W^T} = \boldsymbol{I_D}$$

由于列满秩矩阵 $\boldsymbol{X}$ 的协方差矩阵是一个对称且正定的矩阵 [27]，$\boldsymbol{W^T}$ 实际上是合同变换中将 $\boldsymbol{\Gamma_X}$ 转化为合同规范型的线性变换矩阵。将上式写为

[27] 详见 D.2.2 开头部分

$$\boldsymbol{\Gamma_X} = (\boldsymbol{W^T W})^{-1} \Rightarrow \boldsymbol{W^T W} = \boldsymbol{\Gamma_X^{-1}}$$

有时也将白化矩阵统一记为

$$\boldsymbol{W} = \boldsymbol{\Gamma_X^{-1/2}}$$

记号 $\boldsymbol{\Gamma_X^{-1/2}}$ 并不代表某个特定的矩阵，因为 $\boldsymbol{W}$ 的解实际上并不唯一，对 $\boldsymbol{W}$ 左乘或任何一个正交变换矩阵均可满足条件。

为了求解 $\boldsymbol{W}$ 我们对 $\boldsymbol{\Gamma_X}$ 进行特征值分解得到

$$\boldsymbol{\Gamma_X} = \boldsymbol{U\Lambda U^T} \Rightarrow \boldsymbol{\Gamma_X^{-1}} = \boldsymbol{U\Lambda^{-1}U^T}$$

取

$$\boldsymbol{W_P} = \boldsymbol{\Lambda^{-1/2}U^T}$$

则白化被称为 PCA 白化（PCA whitening）。白化操作实际上相当于考虑了所有主成分的 $\boldsymbol{X}$ 在的低维空间的坐标表示（即利用作为 Encoder 的正交矩阵 $\boldsymbol{U^T}$ 将整个数据映向低维空间），进行方差归一化。

$$\boldsymbol{W_P X} = \boldsymbol{\Lambda^{-1/2}U^T X} = \boldsymbol{\Lambda^{-1/2}X_p}$$

取

$$\boldsymbol{W_Z} = \boldsymbol{UW_P} = \boldsymbol{U\Lambda^{-1/2}U^T}$$

则白化被称为 ZCA 白化（ZCA whitening）[28]。白化操作实际上相当于将 PCA 白化得到的低维空间的数据再利用作为 Decoder 的正交矩阵 $\boldsymbol{U}$ 映射回原空间。

D.3.5　KPCA

核技巧的本质是将低维数据利用核函数 ϕ 将低维数据点转化为表征能力更强的高维特征，在高维空间上再对特征进行处理以得到比低维空间更好的结果。在使用核技巧的过程中不一定总是能得到 ϕ 的显式表达，有时只能利用核空间的 Gram 矩阵 $\boldsymbol{K}$ 对数据进行分析和处理[29]

[28] https://en.wikipedia.org/wiki/Whitening_transformation

[29] 详见 C.3 结尾部分

$$\boldsymbol{K} = \{\phi(\boldsymbol{x_n})^T\phi(\boldsymbol{x_m})\}_{n,m=1}^N = \{\kappa(\boldsymbol{x_n}, \boldsymbol{x_m})\}_{n,m=1}^N = \phi(\boldsymbol{X})^T\phi(\boldsymbol{X})$$

对于核空间上的 PCA，即核主成分分析（Kernel principal component analysis / KPCA），我们只需将 $\boldsymbol{X}$ 代换为 $\phi(\boldsymbol{X})$ 即可，其余结论不变。此时对 $\phi(\boldsymbol{X})$ 进行奇异值分解

$$\phi(\boldsymbol{X}) = \boldsymbol{U\Sigma V^T}$$

得到

$$\boldsymbol{X^*} = \boldsymbol{U_L U_L^T}\phi(\boldsymbol{X})$$

$$\boldsymbol{X_p} = \boldsymbol{U_L^T}\phi(\boldsymbol{X}) = \boldsymbol{\Sigma_L V_L^T}$$

在很多情况下我们只关心数据在低维空间上的坐标表示 $\boldsymbol{X_p}$，由于 $\phi(\boldsymbol{X})$ 有时是未知的，故我们只能利用另一种方法求解 $\boldsymbol{X_p}$。而由奇异值分解和特征值分解的联系，利用 $\boldsymbol{K}$ 恰好能够求解表达式 $\boldsymbol{\Sigma_L V_L^T}$，只需对 $\boldsymbol{K}$ 进行特征值分解

$$\boldsymbol{K} = \boldsymbol{V\Lambda V^T}$$

求得 $\boldsymbol{V_L}$ 和 $\boldsymbol{\Lambda}$ 后利用求得的特征值开平方求出前 L 个奇异值即可。虽然此时我们甚至没有求得高维上的数据 $\phi(\boldsymbol{X})$ 的显示表达，但我们已经求得了该数据在低维空间上的坐标表示。

从正交投影的视角看待 PCA，则 PCA 是寻找一个合适的子空间进行投影的过程，从 SVD 的视角看待 PCA 则 PCA 是一个利用 SVD 提供的正交矩阵进行旋转，从而将数据中方差较大的几个轴对齐，把方差较小的轴舍弃的过程。在我们的这几节的学习中，我们得知了就保存矩阵信息（能量）这方面而言，PCA 从好几个角度都具有无可挑剔的结果。然而我们对降维的探索之路就止步于此了吗？答案显然是否定的。在 PCA 的过程中，虽然数据的信息在某种意义上得到了保存，但是数据中向量之间的距离有时却难以保证在降维前后是近乎一致的。这为流形学习提供了突破口。

附录 E　Laplacian 矩阵与谱聚类

E.1　Laplace 算子

定义向量微分算子 nabla 算子（Nabla operator），其定义为

$$\nabla = \left(\frac{\partial}{\partial x_1}, \frac{\partial}{\partial x_2}, \cdots, \frac{\partial}{\partial x_n}\right)^T = \left(\frac{\partial}{\partial x_i}\right)_{n\times 1}$$

以下写法只是一种方便记忆的表达，利用 ∇ 算子函数梯度可以写为

$$\text{grad } f = \nabla f$$

定义向量散度（Divergence）用以表征向量场在各点的发散程度，散度大于 0 代表该点为源（Source）（正源），小于 0 代表该点为汇（Sink）（负源），等于零代表该点无源。如果向量场处处散度为 0，该向量场称为无源场（Solenoidal field）[1]。散度算子建立了向量场到标量场的映射

$$\text{div } \boldsymbol{v} = \nabla \cdot \boldsymbol{v}$$

定义向量旋度（Curl）用以表征向量场在各点的旋转程度，其旋转以旋度对应的向量为转轴按右手法则（Right-hand rule）决定旋转的方向。如果向量场处处旋度为 0，该向量场称为无旋场（Irrotational field）[2]。旋度算子建立的向量场到向量场的映射

$$\text{curl } \boldsymbol{v} = \nabla \times \boldsymbol{v}$$

定义（连续）Laplace 算子（Laplace operator）Δ [3]，其中 $\nabla^2 f(\boldsymbol{x})$ 是 Hessian 矩阵

[1] https://en.wikipedia.org/wiki/Divergence

[2] https://en.wikipedia.org/wiki/Curl_(mathematics)

[3] https://en.wikipedia.org/wiki/Laplace_operator

$$\Delta f = \operatorname{div}(\operatorname{grad} f) = \nabla \cdot \nabla f = \sum_{i=1}^{n} \frac{\partial^2 f}{\partial x_i^2} = \operatorname{tr}(\nabla^2 f(\boldsymbol{x}))$$

由数学分析知识可知对于二阶导数

$$f''(x) = \lim_{h \to 0} \frac{f'(x+h) - f'(x)}{h} = \lim_{h \to 0} \frac{f(x+h) + f(x-h) - 2f(x)}{h^2}$$

这可以由 L'Hôpital 法则验证。取 $h = 1$ 得到近似公式，这实际上是二阶差分的计算式

$$f''(x) \approx f(x+1) + f(x-1) - 2f(x)$$

考虑二元函数 f 的 Laplace 算子，由以上二阶差分近似得到的 Laplace 算子称为离散 Laplace 算子（Discrete Laplace operator）

$$\Delta f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2} = f(x+1) + f(x-1) + f(y+1) + f(y-1) - 4f(x)$$

将数字图像视为一张间距为 1 的由像素点构成的点阵，像素 p 对应的像素值为 $f(p)$，其上下左右的像素点称为像素点的 4 邻域（4-neighborhood），记为 $N_4(p)$ [4]。我们考虑对图像上的数据点进行微小扰动，在扰动后数据点随机跳转至其邻域上的另一个点，其像素值值也发生相应的变化，Laplace 算子实际上反映的是这种扰动带来的所有可能的像素值的变化的和[5]，也许这在某种程度上给出了为什么用变化量符号 Δ 代表 Laplace 算子的解释。在图像的卷积中，这对应着 Laplacian 滤波器（Laplacian filter）

$$\begin{pmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{pmatrix}$$

其对图像的处理相当于对图像进行边缘检测（Edge detection）或者说是锐化处理（Sharpening），像素点加权后的值体现了该像素点周围像素值变化的快慢[6]。

[4] Gonzalez, Rafael C. Woods,Richard E., *Digital Image Processing (Fourth Edition)*, p.94

[5] https://zhuanlan.zhihu.com/p/85287578

[6] Gonzalez, Rafael C. Woods,Richard E., *Digital Image Processing (Fourth Edition)*, p.256-257

事实上，以每个像素点为节点，和 4 邻域的点之间进行连边，这种连接关系构成的图 G 是一个网格，此时边的权重一般都会视为单位权重。这样的图是相当规整的，在这个网络里，当像素点不是边缘点时，我们用 p_{ij} 代表坐标为 (i,j) 的像素点，f_{ij} 代表对应的像素值，则 Laplace 算子可以写为

$$\Delta f_{ij} = f_{i+1,j} + f_{i-1,j} + f_{i,j+1} + f_{i,j-1} - 4f_{ij} = \sum_{(n,m)\in\mathcal{N}_{ij}} (f_{nm} - f_{ij})$$

即为我们刚才定义的离散 Laplace 算子对应的表达式。

E.2　图的 Laplacian 矩阵

现实中我们抽象出的节点的连接关系却往往没有网格那么单纯（或具有如此好的对称性），这启发我们将图像处理的算法或者说概念推广到一般图上去 [5]，这也是从 CNN 到 GCN 的思路。考虑一般的 N 个节点组成的图 G，我们将邻域从图像的 4 邻域推广至一般图节点的邻域，以节点序号 i 代替节点，以 f_i 代表节点的函数值。

$$\Delta f_i := \sum_{j\in\mathcal{N}_i} (f_j - f_i)$$

考虑邻接矩阵，以 w_{ij} 代表边 (i,j) 的权重，这里通常考虑权重非负，且当边不存在时权重为 0，得到

$$\Delta f_i = \sum_{j\in V(G)} w_{ij}(f_j - f_i) = \sum_{j\in V(G)} w_{ij} f_j - f_i \sum_{j\in V(G)} w_{ij} = \boldsymbol{W_i f} - f_i \boldsymbol{W_i 1_N}$$

上式实际上代表了由节点 i 转移至其他节点的函数值的加权变化量，w_{ij} 可以理解为这种转移的难易程度或者连接关系的强弱，如果 w_{ij} 是归一化的，这个值实际代表了函数值变化量的期望。得到图节点的离散 Laplace 算子

$$\Delta \boldsymbol{f} := (\Delta f_i)_{N\times 1} = (\boldsymbol{W_i f} - f_i \boldsymbol{W_i 1_N})_{N\times 1} = (\boldsymbol{W} - \mathrm{diag}(\boldsymbol{W 1_N}))\boldsymbol{f}$$

因而图的 Laplace 算子实际上是一个和函数值序列 $\boldsymbol{f}$ 有关的线性算子，其对图的作用可以通过矩阵乘法的形式体现。由此设图的 Laplacian 矩阵（Laplacian matrix）为 $\boldsymbol{L}$，其定义如下

$$\boldsymbol{L} := \boldsymbol{W} - \mathrm{diag}(\boldsymbol{W}\mathbf{1}_{\boldsymbol{N}})$$

为了满足 $\boldsymbol{L}$ 的半正定性的代数性质，我们对上式取一个负号

$$\boldsymbol{L} := \mathrm{diag}(\boldsymbol{W}\mathbf{1}_{\boldsymbol{N}}) - \boldsymbol{W}$$

当取 $\boldsymbol{W}$ 为邻接矩阵 $\boldsymbol{A}$ 时，$\boldsymbol{W}\mathbf{1}_{\boldsymbol{N}}$ 实际上代表了节点按照节点序号排列而成的节点度的对角阵，即度矩阵 $\boldsymbol{D}$（Degree matrix）。在通常情况下有时我们也会直接直接记矩阵 $\mathrm{diag}(\boldsymbol{W}\mathbf{1}_{\boldsymbol{N}})$ 为度矩阵 $\boldsymbol{D}$，将节点的度定义为从节点流出的边的权重之和。我们在后续将沿用这种做法，此时上式可以写为

$$\boldsymbol{L} := \boldsymbol{D} - \boldsymbol{W}$$

通常考虑无向图对应的 Laplacian 矩阵，且无向图需要满足权值非负和没有孤立节点的条件（因为我们后续要对度矩阵求逆），此时 $\boldsymbol{W}$ 是实对称的，$\boldsymbol{L}$ 具有以下重要的性质[7]：

性质一　$\boldsymbol{L}$ 是半正定的。

考虑定义

$$\begin{aligned}\boldsymbol{x}^{\boldsymbol{T}}\boldsymbol{L}\boldsymbol{x} &= \boldsymbol{x}^{\boldsymbol{T}}\mathrm{diag}(\boldsymbol{W}\mathbf{1}_{\boldsymbol{N}})\boldsymbol{x} - \boldsymbol{x}^{\boldsymbol{T}}\boldsymbol{W}\boldsymbol{x} = \sum_{i=1}^{N}\sum_{j=1}^{N} w_{ij}x_i^2 - \sum_{i=1}^{N}\sum_{j=1}^{N} w_{ij}x_i x_j \\ &= \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} w_{ij}(x_i - x_j)^2 \geqslant 0\end{aligned}$$

上述 $\boldsymbol{L}$ 对应的二次型表达式十分重要，该式实际上反映了数据点之间加权的距离的平方和。我们接下来有关谱聚类的设计将围绕这个表达式进行。取 $w_{ij} = 1/N$ 我们顺带得到了中心化矩阵 $\boldsymbol{C}$ 的一个性质

$$\boldsymbol{x}^{\boldsymbol{T}}\boldsymbol{C}\boldsymbol{x} = \boldsymbol{x}^{\boldsymbol{T}}\left(\boldsymbol{I} - \frac{1}{N}\mathbf{1}_{\boldsymbol{N}\times\boldsymbol{N}}\right)\boldsymbol{x} = \frac{1}{2N}\sum_{i=1}^{N}\sum_{j=1}^{N}(x_i - x_j)^2$$

性质二　$\boldsymbol{L}$ 必然存在一个 0 特征值和全 1 的特征向量，且这个 0 特征值对应的特征子空间维数为图的连通分支的数量。

[7] http://www.cs.yale.edu/homes/spielman/eigs/lect2.pdf

考虑

$$(\mathrm{diag}(\boldsymbol{W}\mathbf{1}_{\boldsymbol{N}})-\boldsymbol{W})\mathbf{1}_{\boldsymbol{N}}=\boldsymbol{W}\mathbf{1}_{\boldsymbol{N}}-\boldsymbol{W}\mathbf{1}_{\boldsymbol{N}}=0=0\mathbf{1}_{\boldsymbol{N}}$$

我们实际上已经得到了 $\boldsymbol{L}$ 必然存在一个 0 特征值全 1 的特征向量，接下来我们将进一步地分析问题。由于 $\boldsymbol{L}$ 是实对称且半正定的，因而 $\boldsymbol{L}$ 的特征值均非负，我们只需考虑 $\boldsymbol{L}$ 中 0 特征值对应的的特征向量（即 Null($\boldsymbol{L}$) 中的向量）即可，设这样的向量为 $\boldsymbol{x}$，则

$$\boldsymbol{x}^{\boldsymbol{T}}\boldsymbol{L}\boldsymbol{x}=0=\frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}w_{ij}(x_i-x_j)^2$$

对于 G 的一个连通分支 C 而言

$$\forall i_0,i_l\in V(C),\ \exists i_1,i_2,\cdots,i_{l-1}\in V(G),\ w_{i_j,i_{j+1}}>0,\ j=0,1,\cdots,l-1$$

由 $\boldsymbol{x}^{\boldsymbol{T}}\boldsymbol{L}\boldsymbol{x}=0$ 迫使

$$x_{i_0}=x_{i_1}=\cdots=x_{i_l}\Rightarrow\forall i,j\in V(C),\ x_i=x_j$$

设 G 所有的连通分量为 $C_1,C_2,\cdots,C_K$ 得到

$$\boldsymbol{x}^{\boldsymbol{T}}\boldsymbol{L}\boldsymbol{x}=\frac{1}{2}\sum_{k=1}^{K}\sum_{(i,j)\in E(C_k)}w_{ij}(x_i-x_j)^2$$

对于每个连通分支内的所有节点，其序号对应的 $\boldsymbol{x}$ 分量的值必须相等，而不同连通分支内的节点其序号对应的 $\boldsymbol{x}$ 分量却未必相等，因而能且仅能构造出与连通分支个数相等的相互线性无关的一组 0 特征值对应的特征向量，将连通分支对应的分量取 1 其余分支对应的分量取 0 即可。

上述该定理表明，$\boldsymbol{L}$ 的次小特征值是否为 0 直接反映了图的连通性。图的连通性和连通分支个数从权重矩阵（邻接矩阵）的角度看并不容易得出，然而我们却通过 Laplacian 矩阵和特征值分解将两者建立起了联系。由该定理我们也可以通过零空间的维度和 $\boldsymbol{L}$ 的秩的关系立即得出 $\boldsymbol{L}$ 的秩为图中节点数减去强连通分量的个数。

考虑矩阵论中 Courant-Fischer 定理（Courant-Fischer Theorem），这里将实对称矩阵 $\boldsymbol{A}$ 的特征值从小到大排序

$$0 = \lambda_1 \leqslant \lambda_2 \leqslant \cdots \leqslant \lambda_N$$

则对于矩阵的 Rayleigh 商满足

$$\begin{aligned}\lambda_k &= \min_{\dim(S)=k} \max_{\boldsymbol{x}\in S} R(\boldsymbol{A},\boldsymbol{x}) = \min_{\dim(S)=k} \max_{\boldsymbol{x}\in S} \frac{\boldsymbol{x}^{\boldsymbol{T}}\boldsymbol{A}\boldsymbol{x}}{\boldsymbol{x}^{\boldsymbol{T}}\boldsymbol{x}} \\ &= \max_{\dim(S)=n-k-1} \min_{\boldsymbol{x}\in S} R(\boldsymbol{A},\boldsymbol{x}) = \max_{\dim(S)=n-k-1} \min_{\boldsymbol{x}\in S} \frac{\boldsymbol{x}^{\boldsymbol{T}}\boldsymbol{A}\boldsymbol{x}}{\boldsymbol{x}^{\boldsymbol{T}}\boldsymbol{x}}\end{aligned}$$

在 $\boldsymbol{x}^{\boldsymbol{T}}\boldsymbol{L}\boldsymbol{x}$ 中通过适当选取变量 $\boldsymbol{x}$ 可以估测 $\boldsymbol{L}$ 特征值的范围[7]。

E.3　切图聚类

事实上将图划分为连通分量 $C_1, C_2, \cdots, C_K$ 是图的一种特殊的也是常用的划分，因为划分实际上对应着一种等价关系，以连通分量内所有的节点确定的等价类具有很好的可解释性（以节点之间能够互达作为等价关系）。考虑一般的划分，将 G 划分为子图 $G_1, G_2, \cdots, G_K$，仿照关于连通分支得到的结论我们得到

$$\boldsymbol{x}^{\boldsymbol{T}}\boldsymbol{L}\boldsymbol{x} = \frac{1}{2}\sum_{k=1}^{K}\sum_{(i,j)\in E(G_k)} w_{ij}(x_i - x_j)^2 + \frac{1}{2}\sum_{k\neq w}\sum_{i\in V(G_k), j\in V(G_w)} w_{ij}(x_i - x_j)^2$$

我们事实上将边集划分为了两个部分，一部分是子图内部的边的集合，另一部分是子图之间的割集的边；当我们考虑把图划分为连通分支时，子图之间其实是没有边的，于是也就没有了后一项。

对于给定的数据点，如果我们将数据点用一个图表示出来，如果图恰好是一个完全图，我们对这些数据点的聚类事实在将图切分为和聚类数目相等的子图。图的切分是存在代价的，因为我们要切断连接子图的边，如果将图中边的权重视为节点相似度的度量，则图的边的权重某种程度上反映了图节点联系的强弱，我们当然希望切断的边的权值和越小越好，以保持子图内节点之间较强的相似程度，由此定义子图的切图权重作为切图的损失

$$W(A,B) = \frac{1}{2}\sum_{i\in V(A), j\in V(B)} w_{ij}$$

其中 1/2 是为了照顾形式。这样的聚类方法称为切图聚类（Graph-cut clustering）。

观察 $\boldsymbol{x}^{\boldsymbol{T}}\boldsymbol{L}\boldsymbol{x}$ 的形式，我们尝试通过控制 $\boldsymbol{x}$ 从 $\boldsymbol{x}^{\boldsymbol{T}}\boldsymbol{L}\boldsymbol{x}$ 表达式中消去某个子图 G_k 内部节点的权值而将外部节点的权值置为 G_k 和其他子图的切图权重，为了达到这个目的我们必须使得在 G_k 内的节点的 $\boldsymbol{x}$ 分量的值相等而在 G_k 外节点的 $\boldsymbol{x}$ 分量的值不相等，只需考虑指示变量 $\boldsymbol{h_k}$

$$h_{ki} = \begin{cases} 1 & i \in G_k \\ 0 & i \notin G_k \end{cases}$$

得到

$$\boldsymbol{h_k^T L h_k} = \frac{1}{2} \sum_{i \in V(G_k), j \notin V(G_k)} w_{ij}(1-0)^2 = W(G_k, \overline{G_k})$$

则总的切图损失为

$$L(G_1, G_2, \cdots, G_K) := \sum_{k=1}^{K} W(G_k, \overline{G_k}) = \sum_{k=1}^{K} \boldsymbol{h_k^T L h_k} = \mathrm{tr}(\boldsymbol{H^T L H})$$

此时

$$\|\boldsymbol{h_k}\|_2^2 = \boldsymbol{h_k^T h_k} = \sum_{i \in V(G_k)} 1 = |V(G_k)|$$

以上优化问题比较接近于 D.3.2 对应的优化问题。但是问题是尽管 $\boldsymbol{h_k}$ 之间的满足正交条件，其单位模长条件却不满足，其 $0-1$ 的限制条件也导致了问题的求解非常困难。

事实上考虑这种损失效果也未必很好，因为算法实际上很容易将图切分为孤立的节点以使得切图损失最小，我们也许需要考虑子图内部的性质对子图划分进行约束[8]。

考虑指示变量的变形我们还能得到更多结论

$$h_{ki} = \begin{cases} f(k) & i \in G_k \\ 0 & i \notin G_k \end{cases}$$

得到

[8] https://www.cnblogs.com/pinard/p/6221564.html

$$\boldsymbol{h_k^T L h_k} = \frac{1}{2} \sum_{i \in V(G_k), j \notin V(G_k)} w_{ij}(f(k) - 0)^2 = W(G_k, \overline{G_k}) f(k)^2$$

这实际上为我们设计约束提供了很大的启发，$f(k)$ 即可作为我们对切分得到的子图 G_k 的限制。例如我们可以设计如下 f 用于限制每个子图的节点数不应过大，考虑

$$f(k)^2 = 1/\left|V(G_k)\right|$$

这样定义出的切图称为 RatioCut[8]，其损失为

$$L(G_1, G_2, \cdots, G_K) := \mathrm{tr}(\boldsymbol{H^T L H}) = \sum_{k=1}^{K} \frac{W(G_k, \overline{G_k})}{|V(G_k)|}$$

此时

$$\|\boldsymbol{h_k}\|_2^2 = \sum_{i \in V(G)} f(k)^2 = \sum_{i \in V(G)} (1/\left|V(G_k)\right|) = 1$$

通过改进损失我们还将 $\boldsymbol{h_k}$ 的模长限制为 1，从而使得正交条件得到了满足。这依然是一个很难求解的问题，因为 $\boldsymbol{h_k}$ 属于该子图的分量为 $1/\sqrt{|V(G_k)|}$ 其余分量为 0 的形式过于特殊，考虑其一个近似解，我们放弃对 $\boldsymbol{h_k}$ 形式的约束，从而使得优化问题转为

$$\min_{\boldsymbol{H}} \mathrm{tr}(\boldsymbol{H^T L H}),\ \boldsymbol{H^T H = I_K}$$

其求解细节我们放在本小节的最后。

我们在 RatioCut 设置的约束实际上是通过限制子图的规模来防止不均衡的划分，然而事实上这个均衡程度的衡量不单单是依靠划分得到的子图内节点的数量。我们进行切图聚类的目的是想让子图内部节点之间相似程度高而子图之间节点相似程度低，换言之我们实际更关心的是子图内节点的度之和（这里忽略了子图内节点和子图外节点的连边的影响）而不是子图内节点的数量（这相当于将节点的度均置为 1），因而我们可以考虑按照节点度加权后的子图节点的计数，以得到从子图节点度之和视角下更均匀的划分

$$f(k)^2 = 1/\sum_{i \in V(G_k)} d_i = 1/\sum_{i \in V(G_k)} \sum_{j \in V(G)} w_{ij}$$

或记

$$\mathrm{vol}(G_k)=\sum_{i\in V(G_k)}\sum_{j\in V(G)}w_{ij}=\sum_{i\in V(G_k)}\boldsymbol{W_j}\mathbf{1_N}\Rightarrow f(k)^2=1/\mathrm{vol}(G_k)$$

这样设计出的切图称为 Ncut（Normalized cut）[8]，其损失如下

$$L(G_1,G_2,\cdots,G_K):=\mathrm{tr}(\boldsymbol{H^TLH})=\sum_{k=1}^{K}\frac{W(G_k,\overline{G_k})}{\mathrm{vol}(G_k)}$$

此时

$$\|\boldsymbol{h_k}\|_2^2=\sum_{i\in V(G_k)}f(k)^2=\sum_{i\in V(G_k)}(1/\mathrm{vol}(G_k))=|V(G_k)|/\mathrm{vol}(G_k)$$

对应的优化问题稍有不同，因为此时 $\boldsymbol{h_k}$ 的模长不为 1，此时我们可以考虑对角阵来修正这个问题，这一步依赖我们对问题的观察

$$\boldsymbol{h_k^TDh_k}=\sum_{i\in V(G_k)}d_ih_{ki}^2=\frac{1}{\mathrm{vol}(G_k)}\sum_{i\in V(G_k)}\sum_{j\in V(G)}w_{ij}=1$$

于是优化问题为

$$\min_{\boldsymbol{H}}\mathrm{tr}(\boldsymbol{H^TLH}),\ \boldsymbol{H^TDH=I_K}$$

记

$$\boldsymbol{Z=D^{1/2}H}\Rightarrow\boldsymbol{H=D^{-1/2}Z}$$

问题转变为

$$\min_{\boldsymbol{Z}}\mathrm{tr}(\boldsymbol{Z^TD^{-1/2}LD^{-1/2}Z}),\ \boldsymbol{Z^TZ=I_K}$$

其中

$$\boldsymbol{\hat{L}=D^{-1/2}LD^{-1/2}=I-D^{-1/2}WD^{-1/2}=I-\hat{W}}$$

称为标准化 Laplacian 矩阵（Normalized Laplacian），这个矩阵依然是半正定的，因为半正定矩阵可以通过特征值分解拆分为一个矩阵和其转置矩阵的

乘积，两边乘以对称矩阵后可以使用半正定矩阵的性质证明矩阵仍然保持半正定的性质。优化问题变为

$$\min_{\boldsymbol{Z}} \operatorname{tr}(\boldsymbol{Z}^{\boldsymbol{T}} \hat{\boldsymbol{L}} \boldsymbol{Z}) \Rightarrow \max_{\boldsymbol{Z}} \operatorname{tr}(\boldsymbol{Z}^{\boldsymbol{T}} \hat{\boldsymbol{W}} \boldsymbol{Z}),\ \boldsymbol{Z}^{\boldsymbol{T}} \boldsymbol{Z} = \boldsymbol{I}_{\boldsymbol{K}}$$

对于优化问题

$$\min_{\boldsymbol{H}} \operatorname{tr}(\boldsymbol{H}^{\boldsymbol{T}} \boldsymbol{L} \boldsymbol{H}),\ s.t.\ \boldsymbol{H}^{\boldsymbol{T}} \boldsymbol{H} = \boldsymbol{I}_{\boldsymbol{D}}$$

我们已经知道的问题的闭式解[9]，只需考虑对称的半正定矩阵 $\boldsymbol{L}$ 的特征值分解

$$\boldsymbol{L} = \boldsymbol{V} \boldsymbol{\Lambda} \boldsymbol{V}^{\boldsymbol{T}}$$

考虑矩阵 $\boldsymbol{L}$ 的后 L 个特征值，注意此时记对应的特征值对角矩阵和特征向量为 $\boldsymbol{\Lambda}_{\boldsymbol{L}}$ 和 $\boldsymbol{V}_{\boldsymbol{L}}$，得到

$$\boldsymbol{H}^* = \boldsymbol{V}_{\boldsymbol{L}}$$

$\boldsymbol{L}$ 实际上具有一个平凡的为 0 的特征值和特征向量[10]，因而在 RatioCut 切图中我们需要特别注意我们实际上需要考虑的是后 $L+1$ 个特征值和对应的特征向量并将小的一个特征值和特征向量舍弃。对于 Ncut 则不需要做这进一步的考虑。

此时算法没有结束，因为我们对 $\boldsymbol{h}_{\boldsymbol{k}}$ 的形式的松弛不能保证最后得出的 $\boldsymbol{h}_{\boldsymbol{k}}$ 能够很好地揭示节点的类别信息。在经过松弛前我们理想的最优解 $\boldsymbol{H} \in \mathbb{R}^{N \times K}$ 的每一行的行向量类似于 one-hot 向量，其唯一的非零元素反映了对应序号节点的类别信息，这也可以视为节点的类别的概率分布，或者说反映节点类别信息的 K 维特征向量。这样 one-hot 的形式和我们最终解得的 $\boldsymbol{H}^*$ 其实是不一定相符的，如果我们仍把 $\boldsymbol{H}^*$ 的行向量视为某种类别分布的话，我们实际上需要做的是对这样的类别分布在原有基础上进行调整使之某种程度上成为一个个 one-hot 向量。

我们可以首先对 $\boldsymbol{H}^*$ 的行向量进行归一化为了满足归一化的特性（为了数值计算稳定需要添加一个小项防止出现全 0 的行）。上述过程就是传统的聚类的过程，通过数据点的高维特征进行聚类，赋予其对应的标签。我们采用的聚类方法可以是传统的 K-Means 算法，当我们这样做的时候，切图聚类事实上缓解了对传统的 K-Means 聚类在高维上由于维度诅咒表现不佳且对线性不可分数据表现较差的劣势。由于在切图聚类过程中考虑了了特

[9] 详见 D.3.2 末尾部分

[10] 我们已经在 E.2 中验证过了

征值分解这一和谱有关的方法，因而整个聚类方法被称为谱聚类（Spectral clustering）。

下一节我们会讲到，谱聚类可以理解为先对数据进行非线性降维后进行传统的聚类。

E.4　Laplacian Eigenmap

考虑流形上与数据点相对的低维嵌入 $\{\boldsymbol{z_i}\}_{i=1}^N$，我们希望相似的数据点对应的低维嵌入之间的距离较近而不相似的节点能够尽可能远（这某种程度上还是流形的保距性），我们定义相似度度量

$$a : \mathcal{X} \times \mathcal{X} \mapsto [0,1],\ a(\boldsymbol{x_n}, \boldsymbol{x_m}) = a_{nm}$$

相似度度量可以借助归一化后的核实现[11]。记相似度矩阵为（相似度越接近 1，两个节点越相似）

$$\boldsymbol{A} = \{a(\boldsymbol{x_n}, \boldsymbol{x_m})\}_{n,m=1}^N$$

最终的优化问题可以写为

$$\begin{aligned}\min_{\boldsymbol{Z}} \sum_{i=1}^N \sum_{j=1}^N \|\boldsymbol{z_i} - \boldsymbol{z_j}\|_2^2 a_{ij} &= \min_{\boldsymbol{Z}} \sum_{i=1}^N \sum_{j=1}^N \sum_{k=1}^D (z_{ki} - z_{kj})^2 a_{ij} \\ &= \min_{\boldsymbol{Z}} \sum_{k=1}^D \sum_{i=1}^N \sum_{j=1}^N (z_{ki} - z_{kj})^2 a_{ij}\end{aligned}$$

这个优化问题实际上存在平凡解，只需取所有低维嵌入均为 0 向量即可，因而我们可以加上正交条件的约束（也是一种秩约束）以阻止这种情况的产生，这里 $\boldsymbol{Z}$ 是由 N 个 D 维列向量排列得到的矩阵

$$\min_{\boldsymbol{Z}} \sum_{k=1}^D \sum_{i=1}^N \sum_{j=1}^N (z_{ki} - z_{kj})^2 a_{ij},\ s.t.\ \boldsymbol{Z}\boldsymbol{Z}^{\boldsymbol{T}} = \boldsymbol{I_D}$$

如果考虑相似度矩阵确定的带权图，其对应的 Laplacian 矩阵为

$$\boldsymbol{L} = \mathrm{diag}(\boldsymbol{A}\boldsymbol{1_N}) - \boldsymbol{A}$$

优化问题的形式非常接近 Laplacian 矩阵对应的二次型的表达式

[11] 这种思想在 7.4 中有所提及

$$\boldsymbol{x}^{\boldsymbol{T}}\boldsymbol{L}\boldsymbol{x} = \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} a_{ij}(x_i - x_j)^2$$

因而我们可以将优化问题写为

$$\min_{\boldsymbol{Z}} 2\sum_{k=1}^{D} \boldsymbol{Z_k}\boldsymbol{L}\boldsymbol{Z_k^T} \Rightarrow \min_{\boldsymbol{Z}} \mathrm{tr}(\boldsymbol{Z}\boldsymbol{L}\boldsymbol{Z^T}),\ s.t.\ \boldsymbol{Z}\boldsymbol{Z^T} = \boldsymbol{I_D}$$

这实际上就是 RatioCut 切图对应的优化问题。因而流形学习是除了切图外谱聚类在第一步提取特征的另一种解读方式。这种流形学习的方法称为 Laplacian Eigenmap。

附录 F　信息论与 EM 算法

F.1　信息论中熵与信息的概念

F.1.1　从物理学的熵到信息熵

熵（Entropy）在物理学中是一个衡量一个系统内在的混乱程度、不确定性和随机性的度量。在 Carnot 循环（Carnot cycle）的启发下，熵的概念在 1862 年由 Clausius 提出[1]，其定义为在可逆过程中输入热量相对于温度的变化率

$$\mathrm{d}S = \mathrm{d}Q/T$$

Clausius 发现在孤立的热力学系统中热量总是从高温物体流向低温物体，系统总是倾向于从有序走向无序，对于孤立的热力学系统而言，这样的熵增的过程是不可逆的。这是熵的宏观的定义。

在 Clausius 之后，人们提出了热力学系统的熵在统计热力学中的等价的解释为[2]

$$S = -k_B \sum_i p_i \ln p_i$$

其中 k_B 为 Boltzmann 常数，p_i 为微观态 i 出现的概率。上述 Gibbs 熵的等式事实上建立起了宏观态的热力学系统的熵和其微观态之间的联系。而对于一个孤立的热力学系统而言，在统计热力学的基本假设下，微观态是等概率出现的，此时设微状态数为 W，则概率有 $p_i = 1/W$，上式退化为 Boltzmann 熵，这是 Boltzmann 在 1872 年提出的著名的 Boltzmann 公式[3]

[1] https://en.wikipedia.org/wiki/Entropy

[2] https://en.wikipedia.org/wiki/Entropy_(statistical_thermodynamics)

[3] https://en.wikipedia.org/wiki/Boltzmann%27s_entropy_formula

$$S = k_B \ln W$$

因而向一个孤立的热力学系统传递热量将增加系统的微观态的数量，从而使得整个热力学系统的熵增加，这是向孤立系统中传递热量导致熵增的微观解释。系统微观态的数量越多，系统的混乱程度和不确定性越大，从而系统的熵越大。

在热力学的熵出现后约 80 年后，Shannon 在其 1948 年的论文 *A Mathematical Theory of Communication* 将热力学的熵引入通信领域，建立的信息熵（Information entropy）（Shannon 熵）的概念。在一年后其论文被重命名为 *The Mathematical Theory of Communication*[4]，一字之差暗示了现代信息论研究的开端，成为了科学史的一大美谈，Shannon 也因此被称为信息论之父。对于离散型随机变量 X，其信息熵被定义为

$$H(X) := -\sum_{\boldsymbol{x}\in\mathcal{X}} p_X(\boldsymbol{x}) \log p_X(\boldsymbol{x}) = \mathrm{E}_{\boldsymbol{x}\sim p_X}[-\log p_X(\boldsymbol{x})]$$

可见信息熵的定义和 Gibbs 熵事实上仅相差了一个常数的倍数。

熵的表达式该如何去理解？$-\log p(\boldsymbol{x})$ 究竟该如何去解读？一种解读信息熵的方式是如果我们把不确定的事件“丢失”的信息看作是微观态的话，所有的丢失的信息组成了信息的宏观的熵[5]。事件“丢失”的信息越多，其混乱程度越大和不确定性越大，这种解读体现了信息熵和统计热力学的熵之间的联系[6]。

F.1.2　一元的熵与信息

这里需要引入信息量（Information content）的概念。什么是信息？Shannon 认为信息是用来消除随机的、不确定的东西[7]。随机事件发生前后从不确定到确定的这一过程中产生了信息。对于以概率 p 发生的随机事件 E 而言，我们可以定义事件的信息量 $I(E) = f(P(E))$。事件的信息量需要满足以下基本的条件[1]：

[4] https://en.wikipedia.org/wiki/A_Mathematical_Theory_of_Communication

[5] https://en.wikipedia.org/wiki/Entropy_in_thermodynamics_and_information_theory

[6] 这种解读需要在学习 F.1.2 后才能更好的理解

[7] Shannon, Claude E., *A Mathematical Theory of Communication*,July 1948

单调性　函数 $f(p)$ 应当是单调递减的。

对于一个事件而言，事件的信息量（Information content）由事件未发生到事件已经发生这一从不确定到确定的过程赋予，从未知到已知的过程中我们获取到了事件的信息。事件发生的概率越低，其不确定性越大，在事件发生前我们对于事件缺失的信息也就越多，因而事件在发生后我们获取到的信息量越大。

边界条件　函数 $f(p)$ 满足 $f(1)=0$。

对于一个必然发生的事件，我们对这个事件确定得不能再确定了，在事件发生前后我们从中获取不到信息，因为我们对这件事情的发生已经完全知晓了，对于这个随机事件我们没有缺失任何信息，因而信息量为 0。该性质结合单调性可得到信息量的非负性。

可加性　对于相互独立的随机事件 E_2，两个事件的积事件满足 $I(E_1E_2)=I(E_1)+I(E_2)$。

对于两个独立的事件，其同时观测得到的信息量等于先后观测两个事件获取到的信息量之和，这很符合我们对于信息量直观的理解。

事实上信息量还需满足连续的条件，这一条件使得确定信息量的函数 f 可以被比较容易地解出来。考虑独立事件的概率性质

$$f(P(E_1E_2))=f(P(E_1)P(E_2))=f(P(E_1))+f(P(E_2))$$

由上式得到

$$0\leqslant p_1,p_2\leqslant 1,\ f(p_1p_2)=f(p_1)+f(p_2)$$

取 $p_1=e^{-x_1}$，$p_2=e^{-x_2}$，令

$$f(p_1p_2)=f(e^{-x_1-x_2})=f(e^{-x_1})+f(e^{-x_2}),\ x_1,x_2\geqslant 0$$

令 $g(x)=f(e^{-x})$ 上式变为 Cauchy 函数方程（Cauchy functional equation）[8]

$$g(x_1+x_2)=g(x_1)+g(x_2)$$

[8] https://en.wikipedia.org/wiki/Cauchy%27s_functional_equation

这个形式的函数方程可能有不少同学在数学分析的习题集解过了。从整数到有理数再到实数，这样一步步地思考问题是数学分析证明中比较常用的技巧:

整数　首先考虑整数，取 $n \in \mathbb{Z}$，得到

$$\forall x,\ g(nx) = g(x + (n-1)x) = g(x) + g((n-1)x) = \cdots = ng(x)$$

我们事实上取 $x = 1$ 就得到了

$$g(n) = ng(1)$$

这是在整数上的线性方程的形式，我们接着验证在有理数上 g 也满足这个形式。

有理数　取 $n \neq 0$ 且 $x = y/n$ 得到

$$\forall y,\ g(ny/n) = g(y) = ng(y/n) \Rightarrow g(y)/n = g(y/n)$$

再取 $m \in \mathbb{Z}$ 并记 $q = m/n \in \mathbb{Q}$ 得到

$$mg(y)/n = mg(y/n) = g(my/n) \Rightarrow qg(y) = g(qy)$$

取 $y = 1$ 我们得到了

$$g(q) = qg(1)$$

实数　对于实数 r 的情况，我们可以构造满足 $\lim\limits_{n\to\infty} q_n = r$ 的有理数列 $\{q_n\}_{n=1}^{\infty}$ 来逼近它，由于 $q_n \in \mathbb{Q}$ 我们得到

$$g(q_n x) = q_n g(x)$$

由 g 连续的条件得到，考虑做差并取极限

$$g(rx) - rg(x) = g(rx) - \lim_{n\to\infty} q_n g(x) = g(rx) - \lim_{n\to\infty} g(q_n x) = 0$$

取 $x = 1$ 我们得到了

$$g(r) = rg(1)$$

令 $g(1) = c, x > 0$，有

$$g(x) = f(e^{-x}) = cx \Rightarrow f(x) = -c \ln x$$

函数 f 事实上已经满足了边界条件。由单调性得到 f 单调递减，因而 $c > 0$，f 最终满足以下形式

$$f(x) = -\log_a x,\ a > 1$$

由于信息量实际上是没有单位的，就像我国际计量大会要对七个基本物理量的基本单位进行定义一样，我们必须选择一个基准点作为单位的的信息量以对信息量进行度量。根据我们选择的 f 的底数 a 的不同，我们的基准点也会发生相应的变化。当取 $a = 2$ 时，信息量单位为 bit，抛掷一枚均匀硬币其结果为正面这一事件的信息量为 1 bit；当取 $a = e$ 时，信息量为 nat，选择这个单位的目的通常是为了方便理论推导；当取 $a = 10$ 时，信息量为 hartley，一个随机的等概率取到 1 到 10 中任何一个数字的十进制数码取到 1 这一事件的信息量为 1 hartley[1]。

对于一个随机变量（信号）X 而言，其在 X 取值为 $\boldsymbol{x}$ 这一事件的信息量被定义为

$$I_X(\boldsymbol{x}) := -\log p_X(\boldsymbol{x})$$

我们发现随机变量的信息熵等于其取样本空间每个值对应的事件的信息量的期望，因而信息熵公式实际上可以改写为

$$H(X) = \mathrm{E}[I_X(X)]$$

取 $a = 2$，信息熵的单位自然而然的为 bit，这样信息熵就和物理学中的熵一样具有了单位。如果随机变量指示一枚均匀硬币抛掷的结果，则随机变量的熵为 1 bit。我们事实上对 bit 这个单位并不陌生，因为 bit 也是计算机最基本的存储单位。计算机利用二进制存储数据，我们将一个二进制的一位称为 1 bit。这里注意不要和计算机存储单位中 bit 弄混了，因为计算机存储的数据是确定的，本身在信息论不具有信息量，当从分布的角度解读存储数据时，存储的数据才具有信息量。如果我们考虑存储数据长度相同的随机

的二进制数的值作为随机变量时，随机变量的信息熵对应的 bit 数和存储数据所用到的 bit 数相同。我们事实上可以通过二进制来存储信息，而这种存储是通过编码来实现的，这两者更深入的联系我们将在 F.2.1 展示，在那里我们将架起信息和存储信息的数据之间的桥梁。

上面提到了我们需要特别注意一点，信息熵的大小取决于我们对信息的分布的解读方式。对于一段英文按照字母组成的系统、词根组成的系统和单词组成的系统去解读信息熵很可能都是不一样，尽管两者表示的信息是一样的。

我们在入信息量时提到了 Shannon 对于信息的定义。当取 $a = 2$ 时，从这种视角看，信息量衡量的是我们在观测到事件时观测的前后概率空间以折半地方式缩小了多少次（因为事件发生后事件不发生的情况被排除了），信息熵衡量的当我们对一个随机变量进行观测时我们在观测前后期望能将概率空间缩小了多少次[9]。这种解释我比较喜欢，它真实地反映了我们的观测使得结果更加确定了。

离散型随机变量的信息熵的表达式事实上具有对称性，即将 $p_X(\boldsymbol{x})$ 的取值打乱，其熵仍然能保持不变。注意到此时就算有某个 $p_X(\boldsymbol{x}) = 0$ 我们也可以由 L'Hospital 法则得到对应的项为 0，这保证了我们可以将随机变量取值延拓至整个样本空间

$$\lim_{p\to 0^+} p\log p = \lim_{p\to 0^+} \frac{\log p}{1/p} = -\lim_{p\to 0^+} \frac{p}{\ln 2} = 0$$

上式表明样本空间中概率取零的项不会影响信息熵的值。显然 $H(X)$ 必然不小于 0，当且仅当某个 $p_X(\boldsymbol{x})$ 取 1 其余取 0 时等号成立。

离散型随机变量的熵当且仅当 $p_X(\boldsymbol{x})$ 取值相同时熵达到最大，这个性质直观上非常好理解，即当随机变量均匀地取值时其不确定性达到最大，即

$$H(X) = -\sum_{\boldsymbol{x}\in\mathcal{X}} p_X(\boldsymbol{x})\log p_X(\boldsymbol{x}) \leqslant \log|\mathcal{X}|$$

证明考虑 Jensen 不等式[10]

$$\sum_{\boldsymbol{x}\in\mathcal{X}} p_X(\boldsymbol{x})\log\frac{1}{p_X(\boldsymbol{x})} \leqslant \log\sum_{\boldsymbol{x}\in\mathcal{X}} p_X(\boldsymbol{x})\frac{1}{p_X(\boldsymbol{x})} = \log|\mathcal{X}|$$

[9] http://b23.tv/Pi7V6Jy

[10] 详见 A.1 的开头部分

该式还表明，当随机变量均匀地从样本空间取值时，样本空间中样本点越多，对应的熵越大，随机变量越不确定。通过以上推导我们得到了离散型随机变量的熵的具有非负性和有界性。

我们常常听到一种说法是中文的信息熵相较于英文而言要大得多。这实际上比较的是中文汉字和英文字母的信息量，中文汉字的常用字有 2000 字以上而英文字母总共只有 26 个，按照信息熵的公式，不仅仅是对均匀分布这种特殊情况，一般而言，样本空间中可取的值越多，信息熵的上界越大，信息熵也会随之更大。中文汉字的平均信息量某种程度上是要比英文字母的平均信息量大的，因而对于同一段信息而言，将其翻译为英文和中文后，我们常常能够发现中文的译文要比英文的短很多。对于信息量相同的信息，相较于英文而言，我们使用更短的中文有就能传达，即用更短的中文去消除等量的不确定性。

信息熵的定义很容易推广至连续型随机变量

$$H(X) := -\int_{\mathcal{X}} p_X(\boldsymbol{x}) \log p_X(\boldsymbol{x}) \mathrm{d}\boldsymbol{x}$$

此时注意 $H(X)$ 事实上是可以小于 0 的。考虑一维的服从均匀分布的随机变量 X，取 $X \sim \mathcal{U}(0, 1/2)$ 得到

$$H(X) = -\int_0^{1/2} 2 \log 2 \mathrm{d}x = -\log 2$$

在偏差-方差分析中我们分析得到方差事实上是导致模型的不确定性的因素 [11]，我们可以通过正态分布简单验证一下熵和方差之间的联系，为了计算方便我们取信息熵对应的底数为 e，$X \sim \mathcal{N}(0, \sigma^2)$ 我们得到

$$\begin{aligned} H(X) &= -\int_{-\infty}^{+\infty} p_X(x) \ln p_X(x) \mathrm{d}x = -\frac{\ln(\sqrt{2\pi}\sigma)}{2\sqrt{2\pi}\sigma} \int_{-\infty}^{+\infty} x \mathrm{d}e^{-\frac{x^2}{2\sigma^2}} \\ &= \frac{\ln(\sqrt{2\pi}\sigma)}{2\sqrt{2\pi}\sigma} \left(\int_{-\infty}^{+\infty} e^{-\frac{x^2}{2\sigma^2}} \mathrm{d}x - x e^{-\frac{x^2}{2\sigma^2}} \Big|_{-\infty}^{+\infty} \right) \\ &= \frac{\ln(\sqrt{2\pi}\sigma)}{2} \left(\int_{-\infty}^{+\infty} \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{x^2}{2\sigma^2}} \mathrm{d}x \right) = \frac{\ln(\sqrt{2\pi}\sigma)}{2} \end{aligned}$$

从而我们验证了正态分布的方差越大，其对应的信息熵也越大。

[11] 详见 4.2 中间部分

F.1.3　二元的熵与信息

一个随机变量的熵和信息的概念很容易推广至两个随机变量的情况。当考虑随机变量 X 和 Y 的联合分布时，定义二者的联合的信息量为

$$I_{X,Y}(\boldsymbol{x},\boldsymbol{y}) = -\log(p_{X,Y}(\boldsymbol{x},\boldsymbol{y}))$$

和条件的信息量为

$$I_{X\mid Y}(\boldsymbol{x}\mid\boldsymbol{y}) = -\log(p_{X\mid Y}(\boldsymbol{x}\mid\boldsymbol{y}))$$

由信息量的可加性，当 X 和 Y 独立时，其信息量满足

$$I_{X,Y}(\boldsymbol{x},\boldsymbol{y}) = -\log(p_{X,Y}(\boldsymbol{x},\boldsymbol{y})) = -\log(p_X(\boldsymbol{x}))-\log(p_Y(\boldsymbol{y})) = I_X(\boldsymbol{x})+I_Y(\boldsymbol{y})$$

在一般情况，其信息量满足

$$\begin{aligned} I_{X,Y}(\boldsymbol{x},\boldsymbol{y}) &= -\log(p_{X,Y}(\boldsymbol{x},\boldsymbol{y})) \\ &= -\log(p_{X\mid Y}(\boldsymbol{x}\mid\boldsymbol{y})) - \log(p_Y(\boldsymbol{y})) = I_{X\mid Y}(\boldsymbol{x}\mid\boldsymbol{y}) + I_Y(\boldsymbol{y}) \\ &= -\log(p_{Y\mid X}(\boldsymbol{y}\mid\boldsymbol{x})) - \log(p_X(\boldsymbol{x})) = I_{Y\mid X}(\boldsymbol{y}\mid\boldsymbol{x}) + I_X(\boldsymbol{x}) \end{aligned}$$

我们发现了可加性赋予了信息量很多类似概率的运算性质，只不过把乘法运算换成了加法运算，除法运算换成了减法运算，更进一步地联系推广条件概率公式

$$I_{X_1,X_2,\cdots,X_N}(\boldsymbol{x_1},\boldsymbol{x_2},\cdots,\boldsymbol{x_N}) = \sum_{n=1}^{N} I_{X_n\mid X_1,\cdots,X_{n-1}}(\boldsymbol{x_n}\mid\boldsymbol{x_1},\cdots,\boldsymbol{x_{n-1}})$$

我们得到了对于离散型随机变量 X 和 Y 的联合熵（Joint entropy）

$$\begin{aligned} H(X,Y) &:= -\sum_{(\boldsymbol{x},\boldsymbol{y})\in\mathcal{X}\times\mathcal{Y}} p(\boldsymbol{x},\boldsymbol{y})\log p(\boldsymbol{x},\boldsymbol{y}) \\ &= \mathrm{E}_{(\boldsymbol{x},\boldsymbol{y})\sim p_{X,Y}}[-\log(p_{X,Y}(\boldsymbol{x},\boldsymbol{y}))] = \mathrm{E}[I_{X,Y}(X,Y)] \end{aligned}$$

上述定义容易推广至连续型随机变量

$$H(X,Y) := \int_{\mathcal{X}\times\mathcal{Y}} p_{X,Y}(\boldsymbol{x},\boldsymbol{y})\log p_{X,Y}(\boldsymbol{x},\boldsymbol{y})\mathrm{d}\boldsymbol{x}\mathrm{d}\boldsymbol{y}$$

当 X 和 Y 独立时，联合熵和二者的信息量一样也具有可加性

$$\begin{aligned} H(X,Y) &= \mathrm{E}[I_{X,Y}(X,Y)] = \mathrm{E}[I_X(X) + I_Y(Y)] \\ &= \mathrm{E}[I_X(X)] + \mathrm{E}[I_Y(Y)] = H(X) + H(Y) \end{aligned}$$

我们之前提到了熵对应的求和事实上是可以交换顺序的，因而我们有

$$H(X,Y) = H(Y,X)$$

条件熵（Conditional entropy）和联合熵略有区别，其计算信息量时考虑的是条件分布

$$\begin{aligned} H(X \mid Y) &:= -\sum_{(\boldsymbol{x},\boldsymbol{y})\in\mathcal{X}\times\mathcal{Y}} p_{X,Y}(\boldsymbol{x},\boldsymbol{y}) \log(p_{X \mid Y}(\boldsymbol{x} \mid \boldsymbol{y})) \\ &= \mathrm{E}_{(\boldsymbol{x},\boldsymbol{y})\sim p_{X,Y}}[-\log(p_{X \mid Y}(\boldsymbol{x} \mid \boldsymbol{y}))] = \mathrm{E}[I_{X \mid Y}(X \mid Y)] \end{aligned}$$

上述定义也容易推广至连续型随机变量

$$H(X \mid Y) := -\int_{\mathcal{X}\times\mathcal{Y}} p_{X,Y}(\boldsymbol{x},\boldsymbol{y}) \log(p_{X \mid Y}(\boldsymbol{x} \mid \boldsymbol{y}))\mathrm{d}\boldsymbol{x}\mathrm{d}\boldsymbol{y}$$

考虑信息量的性质，我们同样有

$$H(X,Y) = H(X \mid Y) + H(Y) = H(Y \mid X) + H(X)$$

和

$$H(X_1, X_2, \cdots, X_N) = \sum_{n=1}^{N} H(X_n \mid X_1, \cdots, X_{n-1})$$

对于条件熵 $H(X \mid Y)$ 而言，由 Y 的约束使得其熵实际上会小于 $H(X)$，该性质被称为熵的条件减少性，这里以离散型随机变量为例，以下用到的不等关系由 Jensen 不等式[10]得到

$$\begin{aligned}
H(X)-H(X\mid Y) &= \mathrm{E}[I_X(X)]-\mathrm{E}[I_{X\mid Y}(X\mid Y)]\\
&= \mathrm{E}[I_X(X)+I_Y(Y)-I_{X,Y}(X,Y)]\\
&= -\sum_{(\boldsymbol{x},\boldsymbol{y})\in\mathcal{X}\times\mathcal{Y}} p(\boldsymbol{x},\boldsymbol{y})\log\frac{p(\boldsymbol{x})p(\boldsymbol{y})}{p(\boldsymbol{x},\boldsymbol{y})}\\
&\geqslant -\log\sum_{(\boldsymbol{x},\boldsymbol{y})\in\mathcal{X}\times\mathcal{Y}} p(\boldsymbol{x},\boldsymbol{y})\frac{p(\boldsymbol{x})p(\boldsymbol{y})}{p(\boldsymbol{x},\boldsymbol{y})}\\
&= -\log\sum_{(\boldsymbol{x},\boldsymbol{y})\in\mathcal{X}\times\mathcal{Y}} p(\boldsymbol{x})p(\boldsymbol{y}) = 0
\end{aligned}$$

我们把上面这个差值记为互信息（Mutual information）$I(X;Y)$，其代表了随机变量 X 包含随机变量 Y 的信息量，这也就不难理解为什么当 X 和 Y 无关时二者的互信息为 0。上面推导显示了其与信息熵之间的联系和作为信息量非负的性质

$$\begin{aligned}
I(X;Y) &:= -\sum_{(\boldsymbol{x},\boldsymbol{y})\in\mathcal{X}\times\mathcal{Y}} p(\boldsymbol{x},\boldsymbol{y})\log\frac{p(\boldsymbol{x})p(\boldsymbol{y})}{p(\boldsymbol{x},\boldsymbol{y})}\\
&= H(X)-H(X\mid Y) = H(X)+H(Y)-H(X,Y)\\
&= H(Y)-H(Y\mid X) = I(Y;X)\geqslant 0
\end{aligned}$$

如果把 $H(X),H(Y)$ 视为集合 X,Y 的势（随机变量的信息量），$H(X,Y)$ 视为集合交集 $X\cup Y$ 的势（随机变量联合的信息量），$H(X\mid Y)$ 和 $H(Y\mid X)$ 视为集合差集的势（由于已知信息的限制信息量，因而已知的信息量从原有信息中剔除），则互信息事实上是集合交集 $X\cap Y$ 的势（随机变量之间重合的信息量）。熵和信息之间的运算和集合运算具有高度的相似性，我们定义的联合熵、条件熵、互信息事实上就对应着集合的交运算、差运算和并运算三种基本的二元运算。

在离散的情况下熵值是非负的，因而对于离散型随机变量有归一化互信息（Normalized mutual information / NMI），这个指标在集合中的含义也非常明显，代表了重合的信息量占总信息量的比例

$$\begin{aligned}
\mathrm{NMI}(X;Y) &= \frac{I(X;Y)+I(Y;X)}{H(X)+H(Y)} = \frac{2I(X;Y)}{H(X)+H(Y)}\\
&= 1-\frac{H(X\mid Y)+H(Y\mid X)}{H(X)+H(Y)}
\end{aligned}$$

显然 NMI 指标也具有对称性，有时被用作聚类的评价指标。随机变量 Ω 以真实类别 $\Omega = \{w_k\}_{k=1}^{K_1}$ 的数据比例取到对应类别的序号，随机变量 $\mathcal{C}$ 以聚类得到的类别 $\mathcal{C} = \{c_k\}_{k=1}^{K_2}$ 的数据比例取到对应类别的序号（这里我们用相同的字母表示随机变量与对应的集合，赋予了对应的概率的集合这一形式实际上是随机变量的另一种表示方法），即

$$p_{\Omega}(k) = \frac{|w_k|}{N},\ p_{\mathcal{C}}(j) = \frac{|c_j|}{N}$$

其联合分布 $p_{\Omega,\mathcal{C}}(k,j)$ 表示取到一个元素同时属于 w_k 和 c_j 的概率，得到

$$p_{\Omega,\mathcal{C}}(k,j) = \frac{|w_k \cap c_j|}{N}$$

得到 NMI 指标为

$$\mathrm{NMI}(\Omega,\mathcal{C}) = \frac{2I(\Omega;\mathcal{C})}{I(\Omega)+I(\mathcal{C})}$$

我们很多时候也会用信息熵和信息量来度量一个分布对应的信息，这与随机变量的熵和信息实际上也是等价的，因为这也只是两种不同的表示方法而已。如果结果越趋向于模糊，分布就会越平坦，其对应的熵值也会越高，因而分布的信息熵实际上显示了分布的“平坦”程度。让我们把目光转向概率分布，利用 Jensen 不等式[10]考虑离散形式的 Gibbs 不等式（Gibbs' inequality）

$$-\sum_{n=1}^{N} p_n \log\frac{q_n}{p_n} \geqslant -\log\sum_{n=1}^{N} p_n\frac{q_n}{p_n} = \log\sum_{n=1}^{N} q_n = 0$$
$$\Rightarrow \sum_{n=1}^{N} p_n \log q_n \leqslant \sum_{n=1}^{N} p_n \log p_n$$

通过连续形式的 Jensen 不等式 Gibbs 不等式也容易推广到连续的情况

$$-\int_{\mathcal{X}} p(\boldsymbol{x})\log\frac{q(\boldsymbol{x})}{p(\boldsymbol{x})}\mathrm{d}\boldsymbol{x} \geqslant 0 \Rightarrow \int_{\mathcal{X}} p(\boldsymbol{x})\log q(\boldsymbol{x})\mathrm{d}\boldsymbol{x} \leqslant \int_{\mathcal{X}} p(\boldsymbol{x})\log p(\boldsymbol{x})\mathrm{d}\boldsymbol{x}$$

我们将这个非负的和分布相关的统计量记为 p 和 q 的 KL 散度（Kullback–Leibler divergence），其又被称为相对熵（Relative entropy）或信息增益（Information gain），其反映了分布 p 和分布 q 之间的距离。KL 散度之所以能够反映分布之间距离的原因我们放在 F.2.1 讲述

$$D_{KL}(p \mid\mid q) := -\sum_{\boldsymbol{x}\in\mathcal{X}} p(\boldsymbol{x}) \log \frac{q(\boldsymbol{x})}{p(\boldsymbol{x})}$$

考虑分布的信息熵，KL 散度和信息熵之间存在以下联系

$$\begin{aligned} D_{KL}(p \mid\mid q) &= -\sum_{\boldsymbol{x}\in\mathcal{X}} p(\boldsymbol{x}) \log \frac{q(\boldsymbol{x})}{p(\boldsymbol{x})} = \sum_{\boldsymbol{x}\in\mathcal{X}} p(\boldsymbol{x}) \log p(\boldsymbol{x}) - \sum_{\boldsymbol{x}\in\mathcal{X}} p(\boldsymbol{x}) \log q(\boldsymbol{x}) \\ &= -H(p) - \sum_{\boldsymbol{x}\in\mathcal{X}} p(\boldsymbol{x}) \log q(\boldsymbol{x}) \end{aligned}$$

我们将这个多出来的积分式定义为分布 p 和 q 交叉熵（Cross entropy）

$$H(p,q) := -\sum_{\boldsymbol{x}\in\mathcal{X}} p(\boldsymbol{x}) \log q(\boldsymbol{x}) = -\mathrm{E}_{\boldsymbol{x}\sim p}[\log q(\boldsymbol{x})] = D_{\mathrm{KL}}(p \mid\mid q) + H(p)$$

其反映了我们使用 q 在每个样本点的信息量来替换 p 对应的信息量得到信息的信息熵，在 F.2.1 我们会进一步地认识到这就是用分布 q 确定的编码方案去编码分布 p 对应的信息的过程[12]。

以上有关 KL 散度和交叉熵的定义很容易拓展至连续的情况

$$\begin{aligned} D_{KL}(p \mid\mid q) &:= -\int_{\mathcal{X}} p(\boldsymbol{x}) \log \frac{q(\boldsymbol{x})}{p(\boldsymbol{x})} \mathrm{d}\boldsymbol{x} \\ H(p,q) &:= -\int_{\mathcal{X}} p(\boldsymbol{x}) \log q(\boldsymbol{x}) \mathrm{d}\boldsymbol{x} \end{aligned}$$

我们发现互信息事实上是特殊的 KL 散度

$$I(X;Y) = D_{KL}(p_{X,Y}(\cdot,\cdot) \mid\mid p_X(\cdot)p_Y(\cdot))$$

其代表了联合分布和边缘分布乘积之间的距离，当 X 和 Y 无关时，联合分布和边缘分布乘积重合，互信息取到最小值 0。我们还发现交叉熵在两个分布相等时退化为分布的信息熵

$$H(p,p) = H(p)$$

Gibbs 不等式告诉我们交叉熵和信息熵之间满足关系

$$H(p,q) \geqslant H(p)$$

[12] https://en.wikipedia.org/wiki/Cross_entropy

$$H(p,q) \geqslant H(q)$$

或表述为

$$H(p,q) \geqslant \max\{H(p), H(q)\}$$

等号当前仅当 p 和 q 几乎处处相等时取到。

因为对于给定的真实分布 p 而言其信息熵是确定的，因而交叉熵损失函数事实上就是在优化真实分布 p 和拟合分布 q 之间的 KL 散度。当且仅当 p 和 q 几乎重合时交叉熵损失达到最小，此时 $H(p,q)$ 将退化为 $H(p)$。

KL 散度虽然常常用来衡量分布之间的差异，但它不是距离度量，虽然它满足非负性且取零的条件为两个分布几乎处处相等，但是其不满足最基本的对称性。同样注意交叉熵也不满足对称性。

F.2　Shannon 编码定理及应用

F.2.1　前缀编码与无损压缩

信号是信息的载体，信息是信号的内容。信息不止有一种表达方式，作为其载体的信号可以不断地进行更换，一首抒情的中文诗，其载体可以是中文文字，也可以是翻译得到的英文文字，还可以是计算机中对应的存储内容的二进制编码；信号有不同的解读方式，就像有一千个读者就有一千个哈姆雷特一样，一篇作品经由不同人的解读可以获取不同的信息，一个文件可以通过修改为不同的后缀名来得到不一样的显示内容。

编码的本质是信号之间的映射，是信息形式的转换。我们将各个国家的文字通过 Unicode 格式编为二进制码显示在屏幕上当然是一种编码，我们将话筒接受到的声音信息转化为电磁信号是一种编码，更广义地说我们在表述自己内心所想表达出来的过程、我们将外文著作翻译为中文的过程也是一个编码的过程。如果需要更加贴近我们之前学习的内容的例子，我们将数据进行线性降维和非线性降维的过程当然也是一个编码的过程，且在这个编码过程中我们实现了信息的压缩。在编码的过程中，如果信息能被完整地复原而不损失任何原始数据则这种编码是无损的，反之是有损的。

通过无损编码对信息压缩是具有极限的，这一极限和信息的熵密切相关，我们举一个大家熟悉的案例来帮助大家认识到这一点。

考虑将字符进行编码，实际上是建立字符到其编码的可逆的映射，当我们考虑变长的编码时，在不引入分隔符的情况下为了使得我们的编码能够被

正确解码，我们通常会考虑前缀码（Prefix code），即要求任意字符的编码均不为其他字符编码的前缀来使得我们解码时不会出现歧义。考虑二进制前缀码，如此编码实际上构造出了一颗以每个字符为叶节点的二叉树，对与父节点相连的每条边从 0 到 1 进行标号后，从树的根节点到字符对应的叶节点的路径对应的标号序列事实上就对应着字符的前缀码。对于二进制前缀码，考虑需要编码的 N 个字符，设字符集为

$$\mathcal{C} = \{c_1, c_2, \cdots, c_N\}$$

假设其码长为 $\{l_n\}_{n=1}^N$，设码长的最大值为 l_m，则有

$$\sum_{n=1}^{N} 2^{l_m - l_n} \leqslant 2^{l_m}$$

考虑深度为 l_m 的满二叉树，这颗树一共有 2^{l_m} 个叶节点，假设我们取了一个字符的码长为 l_i，则以这个字符为祖先节点，其下的 $2^{l_m - l_i}$ 个叶节点都要被删去，因为这违反了前缀编码的规则。被删去的叶节点数量必然不能超过满二叉树的叶节点的数量，因而我们得到了以上不等关系。将上式化简后得到 Kraft 不等式（Kraft's inequality）[13]（在 $D = 2$ 的特殊情况）

$$\sum_{n=1}^{N} 2^{-l_n} \leqslant 1$$

我们事实上可以验证对于满足 Kraft 不等式的码长集合 $\{l_n\}_{n=1}^N$ 必然可以构造出其对应的编码树，从而使得 Kraft 不等式成立对一个可行的二进制前缀编码来说是个充要条件。还是考虑深度为 l_m 的满二叉树，我们通过修剪这颗树使得这棵树符合前缀码的条件。不妨设 $\{l_n\}_{n=1}^N$ 按照序号升序排序，随机选择深度为 l_1 的节点作为字符对应的叶节点并删去其所有的子节点，然后随机选择深度为 l_2 的没有在上一次被选择的节点作为字符对应的叶节点并删去其所有的子节点，重复这个过程直至所有字符都被安插在编码树上，此后舍弃不在字符对应的叶节点到根节点的路径上所有剩下的节点。

这样的构造产生了一个问题，在第 i 次选择是否有深度为 l_i 的节点可供我们选择呢？在第 i 次选择时，深度为 l_i 的节点还剩下

$$2^{l_i} - \sum_{n=1}^{i-1} 2^{l_i - l_n} = 2^{l_i} - 2^{l_i} \sum_{n=1}^{i-1} 2^{-l_n} > 2^{l_i} - 2^{l_i} = 0$$

[13] https://en.wikipedia.org/wiki/Kraft%E2%80%93McMillan_inequality

Kraft 不等式保证了我们总是有剩下的节点可以选。由于我们事实上没有取完所有的项，因此等号不会取到。上述构造编码树的算法是构造 Shannon 编码（Shannon coding）的基础[14]。

Kraft 不等式可以从概率的角度去解读。假设我们有一段随机的二进制码，我们尝试解读其第一个字符，按照编码树去解码。编码从树根节点开始输入，按照 1/2 的概率选择向左还是向右，得到字符 c_n 出现的概率为 2^{-l_n}，当我们考虑部分编码为无效的编码时，每个字符为第一个出现的字符的概率之和

$$C = \sum_{n=1}^{N} 2^{-l_n}$$

必然小于 1，当概率之和小于 1 时，代表编码树不是完全二叉树，部分父节点仅有一个子节点，使得部分二进制码无法被解码，编码存在冗余。当对随机的二进制码加上可以被正确解码的限制时，考虑一段足够长的编码，我们得到了字符 c_n 出现在编码中的频率为

$$q_n = \frac{2^{-l_n}}{C}$$

因而我们可以在编码中出现字符 c_n 对应的事件 E_n 的信息量

$$I(E_n) = -\log q_n = l_n + \log C$$

我们把 $\log C$ 解读为编码的冗余长度 Δ，当 $C = 1$ 时编码的冗余长度最小为 0，这符合我们前文对 Kraft 不等式的解读。因而在不考虑编码冗余的情况下编码的信息量实际上就对应着编码的存储空间的大小。

设固定的字符集 $\mathcal{C}$ 内 N 个字符在一段文字信息中出现频率为 $\{p_n\}_{n=1}^{N}$，设编码对应的可逆的映射为 f，将字符映向一个变长的二进制码，考虑字符的分布为 p，字符对应的码长为随机变量 S，寻找使得码长最短的编码事实上在求解码长期望最小化的优化问题

$$L(\mathcal{C}) = \min_{l_1, l_2, \cdots, l_N} \sum_{n=1}^{N} p_n l_n = \min_{f} \mathrm{E}[S]$$

[14] https://wlsdzyzl.top/2018/11/14/%E4%BF%A1%E6%81%AF%E8%AE%BA%E2%80%94%E2%80%94Kraft%E4%B8%8D%E7%AD%89%E5%BC%8F%E4%BB%A5%E5%8F%8A%E5%8F%98%E9%95%BF%E7%BC%96%E7%A0%81%E5%AE%9A%E7%90%86/

求和式对应着叶节点到根节点的加权路径长度和（Weighted path length / WPL），寻找最短的编码方式也可以认为是在求解使得 WPL 最小的最优树。在数据结构和算法课上我们接触到了 Huffman 编码（Huffman coding），Huffman 编码对应的树为完全二叉树，这代表了 Huffman 编码是一种没有冗余的编码，对应编码的算法以 $O(n\log n)$ 为时间代价给出了最优的编码方式，得到了上面的期望的最小值，记为 $L_H(\mathcal{C})$。下面我们从信息熵的视角来分析一下这种编码方式[15]。

考虑字符对应编码出现的频率 $\{q_n\}_{n=1}^N$，则期望表达式可以写为

$$\begin{aligned}\mathrm{E}[S] &= -\sum_{n=1}^{N} p_n \log 2^{-l_n} = -\sum_{n=1}^{N} p_n \log q_n + \log C \sum_{n=1}^{N} p_n \\ &= -\sum_{n=1}^{N} p_n \log q_n + \Delta\end{aligned}$$

考虑分布的交叉熵，上式被我们改写为

$$\mathrm{E}[S] = H(p,q) + \Delta$$

该式表明当我们使用字符分布为 q 的编码去编码字符分布为 p 的文字信息时码长的平均值，即存储字符所花费的平均空间为两个分布的交叉熵和编码冗余长度的和，当编码的字符分布和真实的分布匹配度较低时，或者编码的冗余长度较高时，都会使得编码的期望长度较大。

从公式上看编码长度是没有上界的，因为我们事实上可以在编码时添加很多冗余，如我们可以向每一段编码前加上 n 个 0，如此我们的冗余长度增加了 n。就像我们写文章一样，我们可以在文章中写很多正确的废话来增加我们干货就不多的文章的字数（这实际上也是个编码的过程，虽然不一定满足前缀码的条件，而且一般不会有计算机编码那么机械）。然而通过灌水的方式增加文章字数是没有上限的，但是我们想使得文章变得凝练，通过删除不必要的废话来削减文章字数，这一操作是有上限的。

编码的过程是使用信息量不同的编码去替换原有字符的过程，由交叉熵的性质得到这种替换会使得信息熵变大

$$\mathrm{E}[S] \geqslant H(p)$$

[15] https://en.wikipedia.org/wiki/Shannon%27s_source_coding_theorem

这一性质也可以理解为用中文翻译一段俄文的《战争与和平》，我们得到的译文中很多在中文看来很不常见的人名会让人看起来非常头大。从而我们得到了期望编码长度的一个下界，上式表明期望编码长度不会小于信息熵，信息熵是前缀编码存储空间的下界，一切基于前缀编码的无损压缩最后的存储空间都不能小于这个下界。

下一步我们证明这个下界能被最优的 f 多大程度上逼近，即考虑这个下界有多紧。很容易想到我们可以考察边界情况，即尽量去匹配编码的字符分布和真实的字符分布

$$\begin{aligned} l'_n = \lceil -\log p_n \rceil \Rightarrow -\log p_n \leqslant l'_n < -\log p_n + 1 \\ \Rightarrow 2^{-l'_n} \leqslant p_n \Rightarrow \sum_{n=1}^{N} 2^{-l'_n} \leqslant \sum_{n=1}^{N} p_n = 1 \end{aligned}$$

因而由 Kraft 不等式存在一种编码方式使得对应编码 f' 满足

$$\mathrm{E}[S'] = \sum_{n=1}^{N} p_n l'_n < \sum_{n=1}^{N} p_n(-\log p_n + 1) = H(p) + \sum_{n=1}^{N} p_n = H(p) + 1$$

由 Kraft 不等式提供的构造方式我们构造出的满足以上不等关系的编码称为 Shannon 编码，这种码不一定是最优的，它可能离最优的前缀编码 Huffman 编码相差很远。我们将这种编码得到的期望值记为 $L_S(\mathcal{C})$。于是我们完成了 Shannon 编码定理（Shannon source coding theorem）[16] 在二进制前缀编码这一特殊前提下的证明，即对于最优的二进制前缀编码 $L_H(\mathcal{C})$，其满足

$$H(p) \leqslant L_H(\mathcal{C}) \leqslant L_S(\mathcal{C}) < H(p) + 1$$

对于 D 进制前缀编码，Kraft 不等式的条件可以写为

$$\sum_{n=1}^{N} D^{-l_n} \leqslant 1$$

Huffman 编码和 Shannon 编码都可以很自然地拓展到 D 进制，而且我们可以证明 D 进制的 Huffman 编码依然是最优的，我们在不改变信息熵的单位 bit 的情况下，Shannon 编码定理可以写为

[16] https://en.wikipedia.org/wiki/Shannon%27s_source_coding_theorem

$$\frac{H(p)}{\log D} \leqslant L_H(\mathcal{C}) \leqslant L_S(\mathcal{C}) < \frac{H(p)}{\log D} + 1$$

推导过程非常类似，此处略。

信息熵在这里确定的界是比较紧的。这恰好体现了信息熵反映了在一个符号系统下文字真实的信息量，即为文字通过前缀码进行无损压缩的极限，当突破信息熵的极限时，原始数据的信息将会发生亏损。类似于 Huffman 编码和 Shannon 编码这样的通过对信息的编码使得信息的存储空间逼近信息熵确定的极限的编码方式被称为熵编码。

我们在上一节谈到了我们可以使用空间来存储信息，如果我们考虑以存储数据长度相同的随机的二进制数的值作为随机变量时，随机变量的信息熵对应的 bit 数和存储数据所用到的 bit 数相同，这样看我们的存储看似是没有冗余的，实则不然。Shannon 编码定理揭示了字符平均存储空间的 bit 数事实上代表了我们利用这段空间能够存储信息的信息熵的上界，而我们考虑文字信息量时我们实际上考虑的是文字系统中字符分布的信息熵而不是随机的二进制的值对应的分布的信息熵，因而实际上按照以字符方式解读文字计算得到的文字信息的信息熵可能远远没有达到存储空间所能存储的信息的上限，即我们存储的信息在空间上存在冗余。

由于之前我们提到了信息熵是针对一个符号系统而言的，因而换种方式去解读一段待压缩的文字是有可能使得信息熵下降的，因而一个符号系统下我们计算得到的信息熵并不代表了这段文字信息实际上的压缩极限。比如我们都使用 Huffman 编码，将以单词为单位进行英文的压缩更换为以字母为单位进行英文的压缩，我们压缩的效果一般都能有所提高。针对每种语言固定的语法特性，我们构造不同的符号系统，可以使得文字信息实现更进一步的压缩。

我们可以衡量编码对应的期望长度和信息熵确定的理论上的紧的下界之间的距离

$$E[S] - H(p) = H(p, q) + \Delta - H(p) = D_{KL}(p \,||\, q) + \Delta$$

因而交叉熵反映的是在排除了编码冗余的情况下利用分布 q 的确定的编码方案去编码信息对应的真实分布 p 得到的编码的平均空间代价。当编码方案并不完美，即编码确定的分布和真实分布存在一定距离时，我们使用信息量不同的编码替换字符将产生的额外的平均空间开支，而 KL 散度衡量的正是这个额外的代价。这更进一步地解释了交叉熵和 KL 散度的本质含义。

信息熵反映了信息的信息量，揭示了无损压缩在某种程度上是具有极限的，这种极限使得我们不得不考虑另一种压缩思路，即能不能通过尽可能小的损失达到尽可能高的压缩率[17]。此外我们在第六章和第七章中学习到的以最小化信息损失为目的降维方法实际上就是在做有损压缩这件事情。一个典型的案例就是 Auto-encoder，我们的目的在于通过最大化保留数据原本的信息的方式从高维的数据空间降至低维的隐空间，这个最大化的方式是通过最小化压缩到隐空间的数据复原出的数据和原数据之间的距离实现的。

F.2.2　比较算法的复杂度下界

我们在算法课上还到了比较排序实际上可以抽象为决策树（Decision tree）。我们考虑序列中每个元素都不相等的情况，把每次判断都抽象为一个节点，每次判断时如果排序算法没有结束将根据判断结果是不大于还是小于产生两个分支，将每个输出的排好序结果都抽象为叶节点（其对应的是序列指标的置换），这样构造出的决策树实际上也是二叉树，只不过和编码树不同，这样的二叉树每个节点必然有两个分支，故这样的树实际上是一颗完全二叉树。如果按照比较结果为大于还是小于为两个分支附上 0 和 1 两个值，把输入的序列的每种排列情况的分布视为词频的分布（考虑所有排列情况时词的数量即叶节点的数量为 $N!$），则决策树可以视为特殊的二进制的编码树。我们实际上在对每种决策结果赋予一段由 0 和 1 编码的决策序列，这样的决策序列是由结果唯一确定的，且一旦决策的过程和结果对应的决策序列发生重合将立即终止，决策树是完全二叉树代表了整个决策过程不存在无法处理的决策序列。

我们求解理论最快的排序算法实际上是在找符合完全二叉树的最优树，其 WPL 就是对应的平均比较次数。而我们实际上知道了理论上最优的编码树 Huffman 树是一颗完全二叉树，因而理论上最快的算法对应的就是一颗 Huffman 树。因而我们最终得到了排序算法理论上运行时间的紧的下界，这里我们设每次比较的时间代价为 1 且忽略其他的除了比较外的时间代价，设输入的 N 元序列对应分布为 p，则理论最优的排序算法对应的平均时间复杂度为

$$H(p) \leqslant T^*(N) < H(p) + 1$$

假设输入的分布是均匀的，我们有

[17]我们在 6.3 中提到的 JPEG 是一种实用的图像有损压缩算法

$$H(p) = -\sum_{n=1}^{N!} \frac{1}{N!} \log \frac{1}{N!} = \log N!$$

考虑 Stirling 公式

$$n! = \sqrt{2\pi n}(n/e)^n(1 + O(1/n))$$

得到

$$H(p) = N \log N - N \log e + \frac{1}{2} \log N + \log \sqrt{2\pi} + O(1/N)$$

从而得到了理论最优的排序算法对应的平均时间复杂度的一个比较好的估计。因而基于比较的排序算法时间复杂度以 $\Theta(n \log n)$ 为紧的下界，而对于快速排序，我们可以简单地估计其平均时间的复杂度为[18]

$$T(N) = 2N \ln N + (2\gamma - 4)N + o(N)$$

其中 γ 为 Euler 常数。快速排序已经比较接近这个理论下界了。

我们在算法课还学到了在序列中利用和关键字比较方法对序列的元素进行搜索可以得到一颗二叉搜索树。我们考虑序列中每个元素都不相等，且查询时查询关键字不在序列的情况，我们实际上考虑的是查询失败时元素的插入位置。把每次判断都抽象为一个节点，每次判断如果关键字和元素值不相等将根据产生两个分支，将每个输出的元素在序列中的索引都抽象为叶节点，由此我们同样构造出了一颗决策树，这样构造出的决策树同样是一颗完全二叉树。我们也可以用编码的视角来解读它，同样地按照比较结果为大于还是小于为两个分支附上 0 和 1 两个值，把输入的查询的键值插入位置的分布视为词频的分布，于是我们同样得到了一颗编码树。同样地我们可以得到这样的基于比较的搜索算法紧的下界，只考虑比较带来的时间开销，同样地设输入的键值对应对应分布为 p，则理论最优的搜索算法对应的平均时间复杂度为

$$H(p) \leqslant T^*(N) < H(p) + 1$$

假设对于插入位置而言的输入分布是均匀的，我们有

[18] Uwe Roesler, *A Limit Theorem for "Quicksort"*, July 3, 2003, p.6

$$H(p) = -\sum_{n=1}^{N+1} \frac{1}{N+1} \log \frac{1}{N+1} = \log(N+1)$$

对于序列是有序的情况，我们有二分搜索算法，可以证明（在查找失败的情况下）其时间复杂度为[19]

$$T(N) = \lfloor \log N \rfloor + 2 - 2^{(\lfloor \log N \rfloor + 1)}/(N+1)$$

在序列有序的情况二分已经逼近这个理论下界。

F.3　MLE 与交叉熵损失

第二节我们讲到了由于优化真实分布和拟合分布的交叉熵损失事实上在优化真实分布和拟合分布之间的 KL 散度，第三节我们讲到了交叉熵的本质为利用拟合分布 q 确定的编码方案去编码 p 时所带来的空间开支，分布的差异将带来额外的空间开销，这个开销即为 KL 散度，这赋予了我们选择交叉熵作为真实分布和拟合分布之间的损失函数的合理性。

我们事实上还可以从信息论视角重新审视 MLE，从而从另一个视角得到我们选择交叉熵作为损失函数的原因 [12]。对于 MLE，我们考虑随机变量 X，假设我们进行 N 次独立重复实验后对于样本空间中每个可能出现的结果 $\boldsymbol{x}$ 我们一共观测到 $N(\boldsymbol{x})$ 次，设模型参数为 $\boldsymbol{\phi}$，我们得到了出现这样的结果的似然函数为

$$L(\boldsymbol{X};\boldsymbol{\phi}) = \prod_{\boldsymbol{x}\in\mathcal{X}} q(\boldsymbol{x};\boldsymbol{\phi})^{N(\boldsymbol{x})}$$

考虑采样得到数据的过程，在样本收集的过程中我们实际上已经对样本空间进行了一次观测，重复实验次数为样本量 N，观测的结果就对应着 $\boldsymbol{X}$ 矩阵，于是似然函数可以写为

$$L(\boldsymbol{X};\boldsymbol{\phi}) = q(\boldsymbol{X};\boldsymbol{\phi}) = \prod_{n=1}^{N} q(\boldsymbol{x_n};\boldsymbol{\phi})$$

在 MLE 中通常考虑对数似然函数，如果我们将目光不止局限于样本收集这一次重复实验次数为 N 的观测，而考虑一般情况

[19] https://en.wikipedia.org/wiki/Binary_search_algorithm

$$\max_{\boldsymbol{\phi}} L(\boldsymbol{X};\boldsymbol{\phi}) \Rightarrow \min_{\boldsymbol{\phi}} -\frac{1}{N}\log L(\boldsymbol{X};\boldsymbol{\phi}) = \min_{\boldsymbol{\phi}} -\sum_{\boldsymbol{x}\in\mathcal{X}} \frac{N(\boldsymbol{x})}{N}\log q(\boldsymbol{x};\boldsymbol{\phi})$$

我们已经配凑出通过观测得到了样本出现的频率，由大数定律这个频率在数据量足够大的时候将趋近于样本出现的真实概率，于是我们取 $N \to \infty$ 得到

$$\frac{N(\boldsymbol{x})}{N} \to p(\boldsymbol{x};\boldsymbol{\theta})$$

样本的真实分布是无从得知的，我们也只能依靠采样数据对其进行近似。依托采样得到的数据分布 $\hat{p}$，我们实际上得到了 MLE 的另一种形式

$$\max_{\boldsymbol{\phi}} L(\boldsymbol{X};\boldsymbol{\phi}) \Rightarrow \min_{\boldsymbol{\phi}} H(p_{\boldsymbol{\theta}}, q_{\boldsymbol{\phi}}) \approx \min_{\boldsymbol{\phi}} H(\hat{p}, q_{\boldsymbol{\phi}})$$

其中上面除以观测次数的目的是为了设计损失时消除样本量的影响。我们在使用 SGD 对参数进行更新时，考虑一个 batch 的数据并计算其损失对应的梯度时实际上计算的是梯度的平均值用以近似损失的期望 [20]，即样本的损失在求和后还需除以样本量，因而我们在考虑 MLE 并消去样本量的影响后设计出的损失实际上就是原分布（或近似原分布的采样得到的分布）和拟合分布之间的交叉熵损失。这就揭示了最小化交叉熵损失的本质就是 MLE。由于 $p_{\boldsymbol{\theta}}$ 是固定的，因而最小化 $p_{\boldsymbol{\theta}}$ 和 $q_{\boldsymbol{\phi}}$ 的交叉熵损失也等价于最小化 $p_{\boldsymbol{\theta}}$ 和 $q_{\boldsymbol{\phi}}$ 之间的 KL 散度。

F.4　变分推断简介与 EM 算法

F.4.1　变分推断简介与 ELBO

隐变量的引入有时候可以用来解决复杂分布的问题。考虑带隐变量的优化问题，为了获取隐变量的分布，Bayesian 模型一般需要基于观测数据 $\boldsymbol{X}$ 利用 Bayesian 公式考虑计算如下后验分布

$$p(\boldsymbol{Z} \mid \boldsymbol{X}) = \frac{p(\boldsymbol{X},\boldsymbol{Z})}{p(\boldsymbol{X})}$$

其中 $p(\boldsymbol{X})$ 称为证据（Evidence）。由于证据的计算需要考虑积分，这使得很多情况下后验分布 $p(\boldsymbol{Z} \mid \boldsymbol{X})$ 的解析解的获取异常困难，因而对于真实

[20] 详见 3.2

分布 $p(\boldsymbol{Z} \mid \boldsymbol{X};\boldsymbol{\theta})$ 考虑利用一个用于拟合的变分分布 $q(\boldsymbol{Z};\boldsymbol{\phi})$（Variational distribution）来对后验分布进行近似，这样的对后验分布进行近似的方法被称为变分推断（Variational inference）[21]。最经典的度量分布之间差异的选择是 KL 散度（其合理性我们在 F.2.1 和 F.3 已经进行了论述），问题转化为最小化 KL 散度的问题。考虑 KL 散度的表达式

$$D_{KL}(q_{\boldsymbol{\phi}} \,||\, p_{\boldsymbol{\theta}}(\cdot \mid \boldsymbol{X})) = -\int q(\boldsymbol{Z};\boldsymbol{\phi}) \log \frac{p(\boldsymbol{Z} \mid \boldsymbol{X};\boldsymbol{\theta})}{q(\boldsymbol{Z};\boldsymbol{\theta})} \mathrm{d}\boldsymbol{Z}$$

对 $p(\boldsymbol{Z} \mid \boldsymbol{X})$ 考虑 Bayesian 公式

$$\begin{aligned}
&-\int q(\boldsymbol{Z};\boldsymbol{\phi}) \log \frac{p(\boldsymbol{X},\boldsymbol{Z};\boldsymbol{\theta})}{p(\boldsymbol{X};\boldsymbol{\theta})q(\boldsymbol{Z};\boldsymbol{\phi})} \mathrm{d}\boldsymbol{Z} \\
&\quad = \int q(\boldsymbol{Z};\boldsymbol{\phi}) \log \frac{q(\boldsymbol{Z};\boldsymbol{\phi})}{p(\boldsymbol{X},\boldsymbol{Z};\boldsymbol{\theta})} \mathrm{d}\boldsymbol{Z} + \int q(\boldsymbol{Z};\boldsymbol{\phi}) \log p(\boldsymbol{X};\boldsymbol{\theta}) \mathrm{d}\boldsymbol{Z} \\
&\quad = -\int q(\boldsymbol{Z};\boldsymbol{\phi}) \log \frac{p(\boldsymbol{X},\boldsymbol{Z};\boldsymbol{\theta})}{q(\boldsymbol{Z};\boldsymbol{\phi})} \mathrm{d}\boldsymbol{Z} + \log p(\boldsymbol{X};\boldsymbol{\theta})
\end{aligned}$$

从而得到对数似然函数的分解式

$$\log p(\boldsymbol{X};\boldsymbol{\theta}) = \int q(\boldsymbol{Z};\boldsymbol{\phi}) \log \frac{p(\boldsymbol{X},\boldsymbol{Z};\boldsymbol{\theta})}{q(\boldsymbol{Z};\boldsymbol{\phi})} \mathrm{d}\boldsymbol{Z} + D_{KL}(q_{\boldsymbol{\phi}} \,||\, p_{\boldsymbol{\theta}}(\cdot \mid \boldsymbol{X}))$$

由 KL 散度的非负性 [22] 得到

$$\log p(\boldsymbol{X};\boldsymbol{\theta}) \geqslant \int q(\boldsymbol{Z};\boldsymbol{\phi}) \log \frac{p(\boldsymbol{X},\boldsymbol{Z};\boldsymbol{\theta})}{q(\boldsymbol{Z};\boldsymbol{\phi})} \mathrm{d}\boldsymbol{Z} =: \mathrm{ELBO}(\boldsymbol{X};\boldsymbol{\phi},\boldsymbol{\theta})$$

右式被称为证据下界（Evidence Lower Bound / ELBO）。对于真实分布而言，由于 $p(\boldsymbol{X};\boldsymbol{\theta})$ 是固定的，因而优化问题最终可以写为

$$\min_{\boldsymbol{\phi}} D_{KL}(q_{\boldsymbol{\phi}} \,||\, p_{\boldsymbol{\theta}}(\cdot \mid \boldsymbol{X})) = \max_{\boldsymbol{\phi}} \mathrm{ELBO}(\boldsymbol{X};\boldsymbol{\phi},\boldsymbol{\theta})$$

这里我们看出为什么隐变量的引入可以用来解决复杂分布的问题。我们并不直接通过对数据分布的建模来进行 MLE 估计模型的参数得到数据的真实分布，而是通过对相对简单的先验分布 q 的建模来优化 ELBO 来对真实分布进行拟合。从联合分布拆解出先验分布后得到

[21] https://zhuanlan.zhihu.com/p/88336614

[22] 详见 F.1.3 结尾部分

$$\int q(\boldsymbol{Z};\boldsymbol{\phi})\log p(\boldsymbol{X}\mid\boldsymbol{Z};\boldsymbol{\theta})\mathrm{d}\boldsymbol{Z}+\int q(\boldsymbol{Z};\boldsymbol{\phi})\log\frac{p(\boldsymbol{Z};\boldsymbol{\theta})}{q(\boldsymbol{Z};\boldsymbol{\phi})}\mathrm{d}\boldsymbol{Z}$$
$$=\mathrm{E}_{\boldsymbol{Z}\sim q_{\boldsymbol{\phi}}}[L(\boldsymbol{X}\mid\boldsymbol{Z};\boldsymbol{\theta})]-D_{KL}(q_{\boldsymbol{\phi}}\,||\,p_{\boldsymbol{\theta}})$$

前一项是由 $q_{\boldsymbol{\phi}}$ 重建的似然函数，第二项是 $q_{\boldsymbol{\phi}}$ 和 $p_{\boldsymbol{\theta}}$ 的 KL 散度。

在变分推断中如何设计变分分布和先验分布以及引入合适的分布距离的度量是核心的问题[21]。我们在前文介绍的标准化流[23]可以用于设计变分后验分布，其思想的出发点为数据的真实分布往往很复杂，因而我们可以利用简单分布在对变量进行一系列满足要求的可逆的变换最终实现对复杂分布的拟合，据此设计变分分布。

F.4.2　EM 算法

考虑结合对数似然函数 $p(\boldsymbol{X};\boldsymbol{\theta})$ 关于 ELBO 的分解式进行 MLE，我们得到了 EM 算法（Expectation–maximization algorithm）。为了使得优化 ELBO 和优化对数似然函数是等价的，我们选择让 $q_{\boldsymbol{\phi}}$ 和 $p_{\boldsymbol{\theta}}$ 相等，此时我们实际上直接对数据分布进行建模。在 EM 算法的下 ELBO 被改写为

$$\mathrm{ELBO}(\boldsymbol{X};\boldsymbol{\theta})=\int p(\boldsymbol{Z}\mid\boldsymbol{X};\boldsymbol{\theta})\log\frac{p(\boldsymbol{X},\boldsymbol{Z};\boldsymbol{\theta})}{p(\boldsymbol{Z}\mid\boldsymbol{X};\boldsymbol{\theta})}\mathrm{d}\boldsymbol{Z}$$

更常见的推导是利用 Jensen 不等式[10]。由于隐变量的引入，MLE 中的似然函数可以改写为与隐变量分布相关的条件分布的形式，考虑对数似然函数，得到

$$\log L(\boldsymbol{X};\boldsymbol{\theta})=\log p(\boldsymbol{X};\boldsymbol{\theta})=\log\int p(\boldsymbol{X},\boldsymbol{Z};\boldsymbol{\theta})\mathrm{d}\boldsymbol{Z}$$

上述对数和形式会给求导带来很大麻烦，我们引入隐变量的先验分布 $q(\boldsymbol{Z};\boldsymbol{\phi})$ 得到

$$\log\int q(\boldsymbol{Z};\boldsymbol{\phi})\frac{p(\boldsymbol{X},\boldsymbol{Z};\boldsymbol{\theta})}{q(\boldsymbol{Z};\boldsymbol{\phi})}\mathrm{d}\boldsymbol{Z}$$

由于对数函数取负以后是凸函数，考虑 Jensen 不等式得到上式的一个下界

$$\log\int q(\boldsymbol{Z};\boldsymbol{\phi})\frac{p(\boldsymbol{X},\boldsymbol{Z};\boldsymbol{\theta})}{q(\boldsymbol{Z};\boldsymbol{\phi})}\mathrm{d}\boldsymbol{Z}\geqslant\int q(\boldsymbol{Z};\boldsymbol{\phi})\log\frac{p(\boldsymbol{X},\boldsymbol{Z};\boldsymbol{\theta})}{q(\boldsymbol{Z};\boldsymbol{\phi})}\mathrm{d}\boldsymbol{Z}=:\mathcal{L}(q_{\boldsymbol{\phi}};\boldsymbol{\theta})$$

[23] 详见 B.2

当且仅当存在一个常数 c

$$\frac{p(\boldsymbol{X},\boldsymbol{Z};\boldsymbol{\theta})}{q(\boldsymbol{Z};\boldsymbol{\phi})}=c$$

使得上式几乎处处成立时 Jensen 不等式取等号，考虑概率密度归一化的特性

$$\int q(\boldsymbol{Z};\boldsymbol{\phi})\mathrm{d}\boldsymbol{Z}=\frac{1}{c}\int p(\boldsymbol{X},\boldsymbol{Z};\boldsymbol{\theta})\mathrm{d}\boldsymbol{Z}=\frac{1}{c}p(\boldsymbol{X};\boldsymbol{\theta})=1$$

得到

$$q(\boldsymbol{Z};\phi)=\frac{p(\boldsymbol{X},\boldsymbol{Z};\boldsymbol{\theta})}{c}=\frac{p(\boldsymbol{X},\boldsymbol{Z};\boldsymbol{\theta})}{p(\boldsymbol{X};\boldsymbol{\theta})}=p(\boldsymbol{Z};\boldsymbol{\theta})$$

即 $q_{\boldsymbol{\phi}}$ 和 $p_{\boldsymbol{\theta}}$ 几乎处处相等时取等号。通过这种方式我们同样推出了 ELBO。

进一步地考虑信息熵，则

$$\begin{aligned}&\int p(\boldsymbol{Z}\mid\boldsymbol{X};\boldsymbol{\theta})\log p(\boldsymbol{X},\boldsymbol{Z};\boldsymbol{\theta})p(\boldsymbol{Z}\mid\boldsymbol{X};\boldsymbol{\theta})\mathrm{d}\boldsymbol{Z}-\int p(\boldsymbol{Z}\mid\boldsymbol{X};\boldsymbol{\theta})\log p(\boldsymbol{Z}\mid\boldsymbol{X};\boldsymbol{\theta})\mathrm{d}\boldsymbol{Z}\\&\quad=\mathrm{E}_{\boldsymbol{Z}\mid\boldsymbol{X};\boldsymbol{\theta}}[\log p(\boldsymbol{X},\boldsymbol{Z};\boldsymbol{\theta})]+H(p_{\boldsymbol{\theta}}(\cdot\mid\boldsymbol{X}))\\&\quad=\mathrm{E}_{\boldsymbol{Z}\mid\boldsymbol{X};\boldsymbol{\theta}}[\log L(\boldsymbol{X},\boldsymbol{Z};\boldsymbol{\theta})]+H(p_{\boldsymbol{\theta}}(\cdot\mid\boldsymbol{X}))\end{aligned}$$

对数似然函数分解为了 ELBO 和 KL 散度，在 EM 算法中，ELBO 再次被分解为了期望和熵。这样的形式上就比较漂亮了，第一项是隐变量条件分布下的对数似然函数的期望，第二项是对这个分布的熵约束。

这个优化目标比较难求解，但是采用交替更新的思路也许对解决我们的问题会有所帮助，以下两个步骤请大家关注参数的符号：

E-step　我们通过当前模型参数 $\boldsymbol{\theta}^{(t)}$ 求解 $\boldsymbol{Z}$ 的分布，结合观测数据 $\boldsymbol{X}$，我们在这一步计算条件分布 $p(\boldsymbol{Z}\mid\boldsymbol{X};\boldsymbol{\theta}^{(t)})$ 得到当前迭代轮数下的期望表达式

$$Q(\boldsymbol{\theta};\boldsymbol{\theta}^{(t)}):=\mathrm{E}_{\boldsymbol{Z}\mid\boldsymbol{X};\boldsymbol{\theta}^{(t)}}[\log L(\boldsymbol{X},\boldsymbol{Z};\boldsymbol{\theta})]$$

如此在分布确定后，分布的熵实际上是一个常数，因而可以在下一步最大化期望中被舍弃。

M-step　我们通过求解期望表达式的最大值，得到新的模型参数

$$\boldsymbol{\theta}^{(t+1)}:=\arg\max_{\boldsymbol{\theta}}Q(\boldsymbol{\theta};\boldsymbol{\theta}^{(t)})$$

在初始化模型参数后如此交替地进行两个步骤直至收敛。这样看 EM 算法还是比较好理解的，既然隐变量难以显式的求解，那么我们就将隐变量用概率的形式进行度量，既然这个最大化期望的优化还是难以求解，那么我们就通过确定模型参数来间接地确定隐变量的分布，然后再去最大化隐变量分布对应的期望表达式。这个过程实际上是在不断寻找局部的似然概率的最大值。这种求解局部最优值的想法和 SGD 某种程度上是相通的。

我们使用优化 $Q(\boldsymbol{\theta};\boldsymbol{\theta}^{(t)})$ 优化 ELBO，代替直接优化 $\log p(\boldsymbol{X};\boldsymbol{\theta})$，三者之间存在如下关系

$$\log p(\boldsymbol{X};\boldsymbol{\theta}) = \mathrm{ELBO}(\boldsymbol{X};\boldsymbol{\theta}) = Q(\boldsymbol{\theta};\boldsymbol{\theta}) + H(p_{\boldsymbol{\theta}}(\cdot \mid \boldsymbol{X}))$$

$$\log p(\boldsymbol{X};\boldsymbol{\theta}^{(t)}) = \mathrm{ELBO}(\boldsymbol{X};\boldsymbol{\theta}^{(t)}) = Q(\boldsymbol{\theta}^{(t)};\boldsymbol{\theta}^{(t)}) + H(p_{\boldsymbol{\theta}^{(t)}}(\cdot \mid \boldsymbol{X}))$$

为了证明这种替代是有道理的，我们对每轮优化进行分析。考虑我们最初要优化的对数函数并将其拆分为条件概率，其中 $\boldsymbol{\theta}$ 是 $Q(\boldsymbol{\theta};\boldsymbol{\theta}^{(t)})$ 在该轮优化中 M-Step 解得的最优参数，$\boldsymbol{\theta}^{(t)}$ 是 $\boldsymbol{\theta}$ 上一轮参数的估计值，在这里都被视为固定的参数[24]

$$\log p(\boldsymbol{X};\boldsymbol{\theta}) = \log p(\boldsymbol{X},\boldsymbol{Z};\boldsymbol{\theta}) - \log p(\boldsymbol{Z} \mid \boldsymbol{X};\boldsymbol{\theta})$$

为了配凑 $Q(\boldsymbol{\theta};\boldsymbol{\theta}^{(t)})$ 两边进行乘以 $p(\boldsymbol{X},\boldsymbol{Z};\boldsymbol{\theta})$ 并积分，注意到左侧实际上是一个关于 $\boldsymbol{Z}$ 的常量

$$\begin{aligned}\log p(\boldsymbol{X};\boldsymbol{\theta}) &= \int p(\boldsymbol{Z} \mid \boldsymbol{X};\boldsymbol{\theta}^{(t)}) \log p(\boldsymbol{X};\boldsymbol{\theta}) \mathrm{d}\boldsymbol{Z} \\ &= \int p(\boldsymbol{Z} \mid \boldsymbol{X};\boldsymbol{\theta}^{(t)}) \log p(\boldsymbol{X},\boldsymbol{Z};\boldsymbol{\theta}) \mathrm{d}\boldsymbol{Z} - \int p(\boldsymbol{Z} \mid \boldsymbol{X};\boldsymbol{\theta}^{(t)}) \log p(\boldsymbol{Z} \mid \boldsymbol{X};\boldsymbol{\theta}) \mathrm{d}\boldsymbol{Z}\end{aligned}$$

第一项即为 $Q(\boldsymbol{\theta};\boldsymbol{\theta}^{(t)})$，第二项是一个熵

$$\log p(\boldsymbol{X};\boldsymbol{\theta}) = Q(\boldsymbol{\theta};\boldsymbol{\theta}^{(t)}) + H(p_{\boldsymbol{\theta}^{(t)}}(\cdot \mid \boldsymbol{X}), p_{\boldsymbol{\theta}}(\cdot \mid \boldsymbol{X}))$$

类似地我们可以通过变量代换得到

$$\log p(\boldsymbol{X};\boldsymbol{\theta}^{(t)}) = Q(\boldsymbol{\theta}^{(t)};\boldsymbol{\theta}^{(t)}) + H(p_{\boldsymbol{\theta}^{(t)}}(\cdot \mid \boldsymbol{X}))$$

从而

[24] https://en.wikipedia.org/wiki/Expectation%E2%80%93maximization_algorithm

$$\begin{aligned}&\log p(\boldsymbol{X} \mid \boldsymbol{\theta}) - \log p(\boldsymbol{X} \mid \boldsymbol{\theta}^{(t)}) \\ &\quad = Q(\boldsymbol{\theta}; \boldsymbol{\theta}^{(t)}) - Q(\boldsymbol{\theta}^{(t)}; \boldsymbol{\theta}^{(t)}) + D_{KL}\left(p_{\boldsymbol{\theta}^{(t)}}(\cdot \mid \boldsymbol{X}) \,||\, p_{\boldsymbol{\theta}}(\cdot \mid \boldsymbol{X})\right)\end{aligned}$$

KL 散度非负[25]，于是得到

$$\log p(\boldsymbol{X}; \boldsymbol{\theta}) - \log p(\boldsymbol{X}; \boldsymbol{\theta}^{(t)}) \geqslant Q(\boldsymbol{\theta}; \boldsymbol{\theta}^{(t)}) - Q(\boldsymbol{\theta}^{(t)}; \boldsymbol{\theta}^{(t)})$$

上式表明在该轮优化中优化 $\boldsymbol{\theta}$ 至少能够使得目标函数 $\log p(\boldsymbol{X}; \boldsymbol{\theta})$ 提升的值和 $Q(\boldsymbol{\theta}; \boldsymbol{\theta}^{(t)})$ 的值提升得一样大，从而优化 $Q(\boldsymbol{\theta}; \boldsymbol{\theta}^{(t)})$ 是合理的。

由我们上面推得的结论，我们事实上还可以得到 $Q(\boldsymbol{\theta}; \boldsymbol{\theta}^{(t)})$ 和目标函数 $\log p(\boldsymbol{X}; \boldsymbol{\theta})$ 的差距，这个差距实际上就是该轮优化前后的隐变量条件分布的交叉熵

$$\begin{aligned}\log p(\boldsymbol{X}; \boldsymbol{\theta}) - Q(\boldsymbol{\theta}; \boldsymbol{\theta}^{(t)}) &= H(p_{\boldsymbol{\theta}^{(t)}}(\cdot \mid \boldsymbol{X})) + D_{\mathrm{KL}}\left(p_{\boldsymbol{\theta}^{(t)}}(\cdot \mid \boldsymbol{X}) \,||\, p_{\boldsymbol{\theta}}(\cdot \mid \boldsymbol{X})\right) \\ &= H\left(p_{\boldsymbol{\theta}^{(t)}}(\cdot \mid \boldsymbol{X}), p_{\boldsymbol{\theta}}(\cdot \mid \boldsymbol{X})\right)\end{aligned}$$

当参数 $\boldsymbol{\theta}$ 收敛时，此时交叉熵退化为了熵，我们得到了

$$\log p(\boldsymbol{X}; \boldsymbol{\theta}) - Q(\boldsymbol{\theta}; \boldsymbol{\theta}^{(t)}) \approx H(p_{\boldsymbol{\theta}}(\cdot \mid \boldsymbol{X}))$$

对于刚刚接触到 EM 算法的同学而言一个很有意思的问题是，在推导 EM 算法时为什么不直接在一开始利用权值和为 1 的 $\{w_k\}_{k=1}^K$ 套用 Jensen 不等式而要去配凑 $\{p^{(t)}(k \mid \boldsymbol{x})\}_{k=1}^K$，我们配凑条件分布的操作看起来不是多此一举吗？事实上如果直接在一开始套用 Jensen 不等式相当于我们在上面 EM 算法目标求解的过程中进行

$$\begin{aligned}\min_{\boldsymbol{\theta}} -\int \log p(\boldsymbol{X}, \boldsymbol{Z}; \boldsymbol{\theta}) \mathrm{d}\boldsymbol{Z} &= \min_{\boldsymbol{\theta}} -\log \int p(\boldsymbol{Z}; \boldsymbol{\theta}) p(\boldsymbol{X} \mid \boldsymbol{Z}; \boldsymbol{\theta}) \mathrm{d}\boldsymbol{Z} \\ &\leqslant \min_{\boldsymbol{\theta}} -\int p(\boldsymbol{Z}; \boldsymbol{\theta}) \log p(\boldsymbol{X} \mid \boldsymbol{Z}; \boldsymbol{\theta}) \mathrm{d}\boldsymbol{Z}\end{aligned}$$

这个等号在通常情况下是不会取到的，要想等号取到必须满足存在一个常数 c 使得下式对隐变量 $\boldsymbol{Z}$ 几乎处处成立，此时其余参数被固定了

$$p(\boldsymbol{X} \mid \boldsymbol{Z}; \boldsymbol{\theta}) = c \propto p(\boldsymbol{X}; \boldsymbol{\theta}) \Rightarrow \exists a > 0,\ p(\boldsymbol{X} \mid \boldsymbol{Z}; \boldsymbol{\theta}) = a p(\boldsymbol{X}; \boldsymbol{\theta})$$

[25]详见 F.1.3 结尾部分

两边对 $\boldsymbol{X}$ 积分得到

$$\int p(\boldsymbol{X} \mid \boldsymbol{Z};\boldsymbol{\theta})\mathrm{d}\boldsymbol{X} = 1 = a\int p(\boldsymbol{X};\boldsymbol{\theta})\mathrm{d}\boldsymbol{X} = a \Rightarrow p(\boldsymbol{X} \mid \boldsymbol{Z};\boldsymbol{\theta}) = p(\boldsymbol{X};\boldsymbol{\theta})$$

此时在固定模型参数后无论隐变量取何值均不会改变 $\boldsymbol{X}$ 出现的似然概率，更进一步地说隐变量和数据之间是独立的，我们从似然函数提取出隐变量这个操作事实上已经失去了它的意义。更恶劣的是，当任务为聚类任务时，隐变量通常为数据的类别，类别与数据之间是不相关的，这在聚类中是一个难以被接受的假设，因为我们在聚类任务中必须借助数据信息来推断数据类别信息。我们 EM 算法推导中我们的配凑保证了 Jensen 不等式实际上最终是取等号的，而上面这种套用 Jensen 不等式的方法等号在某些的场景下是绝对不能让它取到的。

在 EM 算法中我们直接使 $q_{\boldsymbol{\phi}}$ 和 $p_{\boldsymbol{\theta}}$ 相等，而当数据的分布比较复杂或者当隐空间维度过高时 $p(\boldsymbol{Z};\boldsymbol{\theta})$ 的计算可能是很难实现的。两种算法可用于解决这个问题，一种是借助带有随机性的采样算法，如 MCMC 算法[26]，而另一种是确定性的算法，如变分推断[27]。

[26]详见 10.2

[27]详见 F.4.1

附录 G　非参统计与 KDE

G.1　直方图估计

一维情况下的直方图估计（Histogram estimation）是最简单的非参统计（Nonparametric statistics）中的概率密度的估计方法，将采集到的数据分布的区间划分为一个个宽度为 h 的 bin，统计和样本 x 落入相同的 bin 的样本点的数量 $N_h(x)$，得到样本 x 的在直方图中的概率密度的估计值

$$p_H(x) = \frac{1}{Nh} N_h(x)$$

我们可以用概率密度的定义来理解这个式子，一维随机变量的分布函数为 F 和概率密度函数为 f 之间存在关系[1]

$$f(x) = \lim_{\Delta x \to 0} \frac{F(x + \Delta x)}{x} = \lim_{h \to 0} \frac{F(x + h/2) - F(x - h/2)}{h}$$

从而对于 N 次观测得到的样本 $\{\boldsymbol{x_n}\}_{n=1}^{N}$，得到估计式

$$f_h(x) = \frac{1}{Nh} \sum_{x-h/2<x_n<x+h/2} 1$$

当数据恰好落在 bin 的中心时，$f_h(x)$ 和 $p_H(x)$ 相等。

我们还可以理解为从统计的角度理解直方图估计，从概率密度函数为 $p(\boldsymbol{x})$ 的分布中采样，$\boldsymbol{x}$ 落入区域 $\mathcal{W}$ 的概率为

$$p := \int_{\mathcal{W}} p(\boldsymbol{u}) \mathrm{d}\boldsymbol{u}$$

重复 N 次，样本落入区域 $\mathcal{W}$ 的次数 X 服从二项分布 $B(N, p)$，即

[1] https://www.zhihu.com/question/27301358

$$P(X=k)=\binom{N}{k}p^k(1-p)^{N-k},\ k=0,1,\cdots,N$$

由大数定律得到

$$\lim_{n\to\infty}P\left(a<\frac{X-np}{\sqrt{np(1-p)}}<b\right)=\int_a^b\frac{1}{\sqrt{2\pi}}e^{-\frac{t^2}{2}}\mathrm{d}t$$

取频率 $\hat{p}=X/n$，当满足大样本条件，即当 $p<1-p$ 时，满足 $np\geqslant 15$。设置信度为 $1-\alpha$，由假设检验相关知识得到

$$\hat{p}=p\pm\sqrt{\frac{p(1-p)}{n}}z_{\alpha/2}$$

其中对于误差需满足

$$\sqrt{\frac{p(1-p)}{n}}z_{\alpha/2}\leqslant\delta$$

取

$$V=\int_{\mathcal{W}}\mathrm{d}\boldsymbol{u}$$

令 $\boldsymbol{x}\in\mathcal{W}$，设在一次实验中样本落入区域的频率为 k，我们得到了近似的概率密度估计[2]

$$\frac{k}{n}\approx p=\int_{\mathcal{W}}p(\boldsymbol{y})\mathrm{d}\boldsymbol{y}\approx p(\boldsymbol{x})\int_{\mathcal{W}}\mathrm{d}\boldsymbol{u}=p(\boldsymbol{x})V$$

区域 $\mathcal{W}$ 像是一个在数据点 $\boldsymbol{x}$ 周围设置的用于观测的窗口（Window），由于样本空间一般而言一般是连续取值的或者取某个特定值对应的概率较低，一次观测中几乎不可能观测到取到给定的数据点 $\boldsymbol{x}$ 这一事件，因而我们会考虑将观测的范围扩大至 $\boldsymbol{x}$ 的邻近的点。

这里要求窗口 $\mathcal{W}$ 要足够大使得在确定样本量 N 后区域能够包含一定数量的样本点（满足大样本条件）且使得误差尽可能小，同时也应足够小使得在窗口 $\mathcal{W}$ 内概率密度可以近似为一个常量，这在某种程度上反映了最佳的窗口 $\mathcal{W}$ 的存在性，这点我们下面会探讨。最终得到

$$p_{\mathcal{W}}(\boldsymbol{x})=\frac{k}{NV}$$

[2] https://zhuanlan.zhihu.com/p/39962383

在样本量固定的情况下，固定窗口 $\mathcal{W}$ 而变动 k，这一思想引出了核密度估计（Kernel density estimation / KDE），而固定 k 而变动窗口 $\mathcal{W}$ 将引出 K-NN 分类器（K-Nearest Neighbor classifier）[2]。

定义由带宽（Bandwidth）h 控制大小的窗口

$$\mathcal{N}_h(\boldsymbol{x}) = \{\boldsymbol{u} \mid |x_d - u_d| / h < 1/2,\ d = 1, 2, \cdots, D\}$$

得到

$$V = \int_{\mathcal{N}_h(\boldsymbol{x})} \mathrm{d}\boldsymbol{u} = h^D$$

从而对于 N 次观测得到的样本 $\{\boldsymbol{x_n}\}_{n=1}^N$，估计得到的概率密度为

$$p_h(\boldsymbol{x}) = \frac{1}{Nh^D} \sum_{n=1}^{N} \mathbf{1}_{\boldsymbol{x_n} \in \mathcal{N}_h(\boldsymbol{x})}$$

或者将上式写为给样本点加权的形式，定义权重函数

$$\kappa(\boldsymbol{x}) = \begin{cases} 1 & -1/2 \preccurlyeq \boldsymbol{x} \preccurlyeq 1/2 \\ 0 & \text{otherwise} \end{cases}$$

从而估计得到的概率密度为

$$p_h(\boldsymbol{x}) = \frac{1}{Nh^D} \sum_{n=1}^{N} \kappa\left(\frac{\boldsymbol{x} - \boldsymbol{x_n}}{h}\right)$$

取 $D = 1$，这对应着我们最初提到的一维情况下的直方图估计，只不过窗口的选择上，直方图估计多了一个取整的操作。

这样估计得到的 p_h 满足概率的归一性，考虑积分换元 [3]

$$\begin{aligned} \int_{\mathcal{X}} p_h(\boldsymbol{x}) \mathrm{d}\boldsymbol{x} &= \frac{1}{Nh^D} \sum_{n=1}^{N} \int_{\mathcal{X}} \kappa\left(\frac{\boldsymbol{x} - \boldsymbol{x_n}}{h}\right) \mathrm{d}\boldsymbol{x} \\ &= \frac{1}{N} \sum_{n=1}^{N} \int_{\mathcal{X}} \kappa\left(\frac{\boldsymbol{x} - \boldsymbol{x_n}}{h}\right) \mathrm{d}\frac{\boldsymbol{x} - \boldsymbol{x_n}}{h} = 1 \end{aligned}$$

对于直方图估计，在高维的情况下为了覆盖采集到的数据分布的区间我们需要使用很多的 bins，且由于高维数据的稀疏性很多 bins 实际上是没有数据的，因而导致为了得到合理的样本的概率密度的估计值我们需要大量的

[3] 详见B.1结尾部分

数据量，维度诅咒在非参统计中的体现 [4]。因而非参统计中对数据的大量需求来源于此。

G.2　非参统计中的核函数

我们所使用的加权函数 κ 是不平滑的，这导致了我们估计得到的 $p_h(\boldsymbol{x})$ 也是不平滑的，且在窗口 $\mathcal{N}_h(\boldsymbol{x})$ 内样本的权重是均匀的，和参考点 $\boldsymbol{x}$ 的距离几乎没有什么关系。此时我们可以考虑使用更加平滑的 κ 来代替它，此时窗口的大小可能会扩大至整个空间，只不过在边缘处趋于 0 限制了保证了只有一小部分区域的取值会比较显著地影响估计结果。这样的 κ 需要满足条件：

非负性　这是为了满足概率密度的非负性

$$\forall \boldsymbol{x} \in \mathcal{X},\ \kappa(\boldsymbol{x}) \geqslant 0$$

归一性　这是为了满足概率密度的归一性

$$\int_{\mathcal{X}} \kappa(\boldsymbol{x}) \mathrm{d}\boldsymbol{x} = 1$$

对称性　此时任意两个样本点自身对另一个样本点的概率密度的估计值的影响是相同的

$$\forall \boldsymbol{x} \in \mathcal{X},\ \kappa(\boldsymbol{x}) = \kappa(-\boldsymbol{x})$$

如此我们在保证概率归一化和权重满足对称性的情况下，得到了一个推广形式的概率密度估计。其中权重函数被称为核函数（Kernel function），这样的概率密度估计称为 KDE。KDE 并没有什么高深的，本质就是带宽 h 的限制下对所有样本的加权，因为 KDE 认为每个样本的权重并不相等。

$$p_h(\boldsymbol{x}) = \frac{1}{Nh^D} \sum_{n=1}^{N} \kappa\left(\frac{\boldsymbol{x} - \boldsymbol{x_n}}{h}\right)$$

或记为

[4] 详见6.1

$$p_h(\boldsymbol{x}) = \frac{1}{N}\sum_{n=1}^{N}\kappa_h(\boldsymbol{x}-\boldsymbol{x_n})$$

此时函数 κ_h 满足核函数的性质，可视为由带宽 h 控制的核函数，归一性的证明我们其实已经在上文归一化性质的证明里间接证明过了，我们只需关注到

$$\int_{\mathcal{X}}\kappa_h\left(\boldsymbol{x}-\boldsymbol{x_n}\right)\mathrm{d}\boldsymbol{x} = \frac{1}{h^D}\int_{\mathcal{X}}\kappa\left(\frac{\boldsymbol{x}-\boldsymbol{x_n}}{h}\right)\mathrm{d}\boldsymbol{x} = \int_{\mathcal{X}}\kappa\left(\frac{\boldsymbol{x}-\boldsymbol{x_n}}{h}\right)\mathrm{d}\frac{\boldsymbol{x}-\boldsymbol{x_n}}{h} = 1$$

为什么要叫核函数而不叫窗函数（Window function）呢？也许是在信号处理里窗函数这个名号已经被占用了，窗函数用于对固定的区间内信号进行加权，其目的为减少信号非周期截断带来的频率泄漏，并增加帧左右两段的连续性 [5]，我们在 MFCC 语言特征提取中对信号加窗的一步中使用到了它。也许是因为核技巧中出现的二元核函数 K 形式上和我们定义的一元的 κ 比较像。甚至卷积核都能和核函数 κ 攀上亲戚，因为利用卷积核对图像处理本质上也是一个加权操作，后面我们将看到 κ 某种程度上和卷积核一样，也能视为一个滤波器。

由核函数的定义，一维直方图估计选用的核函数可以表示为

$$\kappa(\boldsymbol{x}) = \begin{cases} 1 & \lfloor x_d \rfloor = 0,\ d = 1, 2, \cdots, D \\ 0 & \text{otherwise} \end{cases}$$

当 5.2 中定义的二维核函数具有形式

$$K(\boldsymbol{x}, \boldsymbol{y}) = \kappa(\boldsymbol{x}-\boldsymbol{y})$$

且积分有限时，我们可以利用进行过归一化的 K 改造为符合要求的 κ。对于 5.2 提及的几个二维的核函数我们可以通过简单地改造得到其对应的 κ 用于 KDE 。注意我们在非参统计里提到的核函数和核技巧中提到的核函数是有点区别的，尽管有时候它们在形式上非常相似。

一般对于高维的核函数可以选用多个一维的核函数的乘积，若数据每个维度没有进行过归一化，数据在每个维度上分布范围的不同使得每个维度对应的一维的核的带宽的选择也应该不同。

[5] https://zhuanlan.zhihu.com/p/24318554

这样的加权机制实际上是注意力机制（Attention mechanism）的思想的体现，带宽 h 控制了“感受野（Receptive field）”的大小，通常来说它使得我们把关注的数据的重心放在参考点 $\boldsymbol{x}$ 附近的数据上。当带宽 $h \to 0$ 时，窗口收缩至数据点，此时我们估计的概率密度退化为在训练集数据上的一个个尖峰（脉冲）；当 $h \to \infty$ 时窗口拓展至整个样本空间，此时我们估计的概率密度均退化为 $\kappa_h(\mathbf{0})$。

考虑 KDE 的期望，这里我们固定变量 $\boldsymbol{x}$ 而从分布中随机抽样得到的样本 $\{\boldsymbol{x_n}\}_{n=1}^N$，设抽样得到的样本为随机变量 X，计算得到的核密度为 $p_h(\boldsymbol{x}; X)$，得到

$$\mathrm{E}_X[p_h(\boldsymbol{x}; X)] = \frac{1}{N}\sum_{n=1}^{N} \mathrm{E}_{\boldsymbol{x_n}}\left[\kappa\left(\frac{\boldsymbol{x}-\boldsymbol{x_n}}{h}\right)\right] = \int_{\mathcal{X}} p(\boldsymbol{\tau})\kappa_h(\boldsymbol{x}-\boldsymbol{\tau})\mathrm{d}\boldsymbol{\tau}$$

考虑卷积公式得到

$$\mathrm{E}_X[p_h(\boldsymbol{x}; X)] = (p * \kappa_h)(\boldsymbol{x}) \Rightarrow \mathrm{E}[p_h] = p * \kappa_h$$

当取 $h \to 0$ 时，此时函数 κ_h 退化为 Dirac 函数

$$\delta(\boldsymbol{x}) = \begin{cases} +\infty & \boldsymbol{x} = \mathbf{0} \\ 0 & \text{otherwise} \end{cases}$$

且积分满足

$$\int_{\mathcal{X}} \delta(\boldsymbol{x})\mathrm{d}\boldsymbol{x} = 1$$

这并不是一个在数学上严格意义的函数，而是广义的函数，其构造方式可以通过函数列来逼近。在数字信号处理中该函数用于对连续信号采样得到离散信号，此时离散信号可被表示为连续信号的形式[6]

$$\int_{-\infty}^{+\infty} x(t)\delta(t-t_0)\mathrm{d}t = (x * \delta)(t_0) = x(t_0)$$

此时核密度函数的期望退化为

$$\mathrm{E}_X[p_h(\boldsymbol{x}; X)] = p(\boldsymbol{x}) \Rightarrow \mathrm{E}[p_h] = p$$

[6] Gonzalez, Rafael C. Woods,Richard E., *Digital Image Processing (Fourth Edition)*, p.303

从期望的角度看核函数的本质是对概率密度函数的卷积平滑处理[2]，当取 $h \to \infty$ 时期望退化为

$$\mathrm{E}_X[p_h(\boldsymbol{x}; X)] = \kappa_h(\mathbf{0})$$

这是一个常数函数。

在 4.2 我们学习到了在线性回归问题中模型的偏差和方差之间是存在 trade-off 的，在 KDE 中也是如此。当 h 选择较小时得到的 p_h 比较尖锐，虽然准确率较高，但方差较大，存在过拟合的风险；而当 h 选择较大时得到的 p_h 比较平滑，虽然方差较小但是准确率较低，存在欠拟合的风险。

事实上对于某些特殊情况最佳的 h 是可以被大致地计算出来的，选取指标为分布的均方误差（用积分形式表示）。RBF 核（Radial basis function kernel）是我们最常用的核之一，其表达式为

$$\kappa(\boldsymbol{x}) = (2\pi)^{-D/2} \exp\left(-\frac{1}{2}\|\boldsymbol{x}\|_2^2\right)$$

RBF 核作为核函数而言其具有很优秀的数学性质，如径向对称性（Radial symmetry）等。对于一维数据选用 RBF 核，当选取损失为积分形式的均方损失时，对样本量 N，最佳的 h 与 $N^{-0.2}$ 成正比[7]。

G.3　Nadaraya-Watson 估计器

对于非线性回归问题，考虑以下方式生成的数据对 $(\boldsymbol{x}, y)$

$$y = f(\boldsymbol{x}) + \varepsilon$$

此处噪声的期望为 0。设样本 $\boldsymbol{x}$ 对应的随机变量为 X，标签 y 对应的随机变量为 Y。由于生成数据的 f 是已知的，两边取期望得到

$$f(\boldsymbol{x}) = \mathrm{E}[f(\boldsymbol{x}) + \varepsilon] = \mathrm{E}[Y \mid X = \boldsymbol{x}]$$

我们的目标是求解如上期望表达式实现对 f 的估测，记[8]

$$f(\boldsymbol{x}) = m(\boldsymbol{x}) := \mathrm{E}[Y \mid X = \boldsymbol{x}] = \int_{\mathcal{Y}} y p_{Y \mid X}(y \mid \boldsymbol{x}) \mathrm{d}y = \int_{\mathcal{Y}} y \frac{p_{X,Y}(\boldsymbol{x}, y)}{p_X(\boldsymbol{x})} \mathrm{d}y$$

[7] https://en.wikipedia.org/wiki/Kernel_density_estimation

[8] https://en.wikipedia.org/wiki/Kernel_regression

考虑联合的核密度，这对应着对样本和标签分别加权

$$p_h(\boldsymbol{x}, y) = \frac{1}{N}\sum_{n=1}^{N} \kappa_h(\boldsymbol{x} - \boldsymbol{x_n})\kappa_{h'}(y - y_n) \approx p_{X,Y}(\boldsymbol{x}, y)$$

这样定义的核密度同样满足归一性。考虑数据点的核密度

$$p_h(\boldsymbol{x}) = \frac{1}{N}\sum_{n=1}^{N} \kappa_h(\boldsymbol{x} - \boldsymbol{x_n}) \approx p_X(\boldsymbol{x})$$

从而得到

$$\begin{aligned} m_h(\boldsymbol{x}) &= \int_{\mathcal{Y}} y\frac{p_h(\boldsymbol{x}, y)}{p_h(\boldsymbol{x})}\mathrm{d}y = \int_{\mathcal{Y}} y\frac{\sum\limits_{n=1}^{N} \kappa_h(\boldsymbol{x} - \boldsymbol{x_n})\kappa_{h'}(y - y_n)}{\sum\limits_{n=1}^{N} \kappa_h(\boldsymbol{x} - \boldsymbol{x_n})}\mathrm{d}y \\ &= \frac{1}{\sum\limits_{n=1}^{N} \kappa_h(\boldsymbol{x} - \boldsymbol{x_n})}\sum_{n=1}^{N} \kappa_h(\boldsymbol{x} - \boldsymbol{x_n})\left(\int_{\mathcal{Y}} y\kappa_{h'}(y - y_n)\mathrm{d}y\right) \end{aligned}$$

考虑

$$\int_{\mathcal{Y}} y\kappa_{h'}(y - y_n)\mathrm{d}y = \int_{\mathcal{Y}} (y - y_n)\kappa_{h'}(y - y_n)\mathrm{d}(y - y_n) + y_n\int_{\mathcal{Y}} \kappa_{h'}(y - y_n)\mathrm{d}y$$

由于 $\kappa_{h'}$ 是对称函数，故函数 $y\kappa_{h'}(y)$ 是奇函数，在对称空间上其积分值为 0，从而积分式的值为 y_n（这里我们一般认为标签空间 $\mathcal{Y}$ 是实数域），考虑恒等变换 I，以上性质可以用卷积的形式表示

$$(I * \kappa_{h'})(y_n) = y_n \Rightarrow I * \kappa_{h'} = I$$

最终我们得到 Nadaraya-Watson 估计器（Nadaraya-Watson estimator / NWE），这事实上是在核密度加权下的标签的期望

$$\hat{y} = m_h(\boldsymbol{x}) = \frac{\sum\limits_{n=1}^{N} \kappa_h(\boldsymbol{x} - \boldsymbol{x_n})y_n}{\sum\limits_{n=1}^{N} \kappa_h(\boldsymbol{x} - \boldsymbol{x_n})}$$

当我们将更新对象从标签更换为新的数据点时，Nadaraya-Watson 估计器变为

$$\boldsymbol{x}_i^{(t+1)} := m_h(\boldsymbol{x}_i^{(t)}) = \frac{\sum_{n=1}^{N} \kappa_h(\boldsymbol{x}_i^{(t)} - \boldsymbol{x}_n^{(t)})\boldsymbol{x}_n^{(t)}}{\sum_{n=1}^{N} \kappa_h(\boldsymbol{x}_i^{(t)} - \boldsymbol{x}_n^{(t)})}$$

上式计算量太大了，所以我们考虑使用 K-NN 对期望的计算进行近似，仅考虑参数附近的数据点即可，注意这一操作使得窗口不再是固定的，而窗口内的样本量是固定的

$$\boldsymbol{x}_i^{(t+1)} := m_h(\boldsymbol{x}_i^{(t)}) = \frac{\sum_{\boldsymbol{x_n} \in \mathcal{N}(\boldsymbol{x}_i)} \kappa_h(\boldsymbol{x}_i^{(t)} - \boldsymbol{x}_n^{(t)})\boldsymbol{x}_n^{(t)}}{\sum_{\boldsymbol{x_n} \in \mathcal{N}(\boldsymbol{x}_i)} \kappa_h(\boldsymbol{x}_i^{(t)} - \boldsymbol{x}_n^{(t)})}$$

由于分式中消去了归一化系数，κ 只需要考虑除了系数项以外的部分即可。

G.4　Mean-shift 算法

考虑具有以下形式的核函数

$$\kappa(\boldsymbol{x}) = cf(\|\boldsymbol{x}\|_2^2)$$

其中 c 为归一化系数，f 为 $(0, +\infty)$ 上的单调不增且分段连续的函数 [9]。为了保证归一化是可以进行的，考虑 D 维超球的表面积 $S_{D-1}(r)$ 得到

$$\int_{\mathcal{X}} \kappa(\boldsymbol{x})\mathrm{d}\boldsymbol{x} = c\int_0^{+\infty} S_{D-1}(r)f(r)\mathrm{d}r = 1 \propto \int_0^{+\infty} r^{D-1}f(r)\mathrm{d}r$$

因而最右侧的反常积分需存在。由于对样本点的正交变换（包含任意旋转和镜面变换操作）不会改变 κ 的取值，因而 κ 是一个径向对称的核函数，常常选用 RBF 核作为相应的核函数。

我们的目的在于使得样本点朝向核密度的最大值方向移动，在离散情况下样本的“众数”对应的方向，在连续的情况下对应着样本的密度最大的区域移动。考虑数据点的核密度函数，为了推导方便，我们讨论 f 可导的情况[10]

[9] https://en.wikipedia.org/wiki/Mean_shift

[10] https://www.zhihu.com/question/67943169

$$p_h(\boldsymbol{x}) = \frac{1}{N}\sum_{n=1}^{N}\kappa_h\left(\boldsymbol{x}-\boldsymbol{x_n}\right) = \frac{1}{Nh^D}\sum_{n=1}^{N} cf\left(\left\|\frac{\boldsymbol{x}-\boldsymbol{x_n}}{h}\right\|_2^2\right)$$

求导后得到

$$\begin{aligned}\nabla_{\boldsymbol{x}} p_h(\boldsymbol{x}) = \frac{\partial p_h}{\partial \boldsymbol{x}} &= \frac{2c}{Nh^{D+2}}\sum_{n=1}^{N} f'\left(\left\|\frac{\boldsymbol{x}-\boldsymbol{x_n}}{h}\right\|_2^2\right)\left(\boldsymbol{x}-\boldsymbol{x_n}\right)\\ &= \frac{2c}{Nh^{D+2}}\left(\boldsymbol{x}\sum_{n=1}^{N} f'\left(\left\|\frac{\boldsymbol{x}-\boldsymbol{x_n}}{h}\right\|_2^2\right) - \sum_{n=1}^{N} f'\left(\left\|\frac{\boldsymbol{x}-\boldsymbol{x_n}}{h}\right\|_2^2\right)\boldsymbol{x_n}\right)\\ &= \frac{2c}{Nh^{D+2}}\sum_{n=1}^{N} f'\left(\left\|\frac{\boldsymbol{x}-\boldsymbol{x_n}}{h}\right\|_2^2\right)\left(\boldsymbol{x} - \frac{\sum\limits_{n=1}^{N} f'\left(\left\|\frac{\boldsymbol{x}-\boldsymbol{x_n}}{h}\right\|_2^2\right)\boldsymbol{x_n}}{\sum\limits_{n=1}^{N} f'\left(\left\|\frac{\boldsymbol{x}-\boldsymbol{x_n}}{h}\right\|_2^2\right)}\right)\end{aligned}$$

记 $g=-f'$，此时 g 满足非负条件，则 g 实际上也可以定义一个核函数

$$\kappa'(\boldsymbol{x}) = c'g(\|\boldsymbol{x}\|_2^2)$$

选择合适的 c' 使得这个核函数满足归一性，由 g 的非负性这个 c' 当然是非负的。这是可以做到的，只需验证以下积分存在即可

$$\int_0^{+\infty} r^{D-1}f'(r)\mathrm{d}r = \int_0^{+\infty} r^{D-1}\mathrm{d}f(r) = r^{D-1}f(r)\Big|_0^{+\infty} - (D-1)\int_0^{+\infty} r^{D-2}f(r)\mathrm{d}r$$

前一项极限存在且为 0，否则可以使用极限定义推得反常积分

$$\int_0^{+\infty} r^{D-1}f(r)\mathrm{d}r$$

不存在，对于后一项积分式由于 $(0,1)$ 上函数值有限，因而区间的积分有限，故讨论区间 $(1,+\infty)$，由 f 非负得到

$$\int_1^{+\infty} r^{D-2}f(r)\mathrm{d}r < \int_1^{+\infty} r^{D-1}f(r)\mathrm{d}r < +\infty$$

因而我们的归一化是可以做到的。记

$$m_h'(\boldsymbol{x}) = \frac{\sum\limits_{n=1}^{N}\kappa_h'\left(\boldsymbol{x}-\boldsymbol{x_n}\right)\boldsymbol{x_n}}{\sum\limits_{n=1}^{N}\kappa_h'\left(\boldsymbol{x}-\boldsymbol{x_n}\right)}$$

考虑核密度

$$p'_h(\boldsymbol{x}) = \frac{1}{Nh^D}\sum_{n=1}^{N}\kappa'_h\left(\boldsymbol{x}-\boldsymbol{x_n}\right)$$

得到

$$\nabla_{\boldsymbol{x}} p_h(\boldsymbol{x}) = \frac{2c}{c'h^2}p'_h(\boldsymbol{x})(m'_h(\boldsymbol{x})-\boldsymbol{x})$$

由于这里是求解最大值，利用梯度上升法进行参数更新，得到

$$\boldsymbol{x^{(t+1)}} := \boldsymbol{x^{(t)}} + \tau\nabla_{\boldsymbol{x}} p_h(\boldsymbol{x^{(t)}})$$

取

$$\tau = \frac{c'h^2}{2c}\frac{1}{p'_h(\boldsymbol{x})} > 0$$

得到

$$\boldsymbol{x^{(t+1)}} := m'_h(\boldsymbol{x})$$

在取 RBF 时 Mean-shift 的结果和 Nadaraya-Watson 估计器的结果是相同的。这里利用到了另一个核函数 κ'，有时候我们根本不关心由 κ 确定的核密度，Mean-shift 虽然优化的不是 κ' 确定的核密度而是 κ 确定的核密度，但是我们近似地认为两者都作为 KDE 的核函数优化的方向应该是类似的，因而在结果上和 Nadaraya-Watson 估计器是一样的。对于 κ' 确定的核密度，学习率随着核密度的上升而下降，在样本密度较小时步长大而在样本密度较大时步长小，这很符合我们对于变步长的优化的需求 [10]。

如果为了兼顾运算效率在计算 $m'_h(\boldsymbol{x})$ 时同样考虑 K-NN 的话，我们实际上是利用 SGD 的思想对梯度进行近似，或者我们可以理解为通过改写核函数，决定样本对应的权重是否取 0 来选定对样本梯度更新有效的数据点，使得样本向局部的样本密度最大的方向移动。

附录 H　KKT 条件与 SVM

H.1　Lagrange 乘数法的推广

对于 $\mathbb{R}^D$ 上的连续可微函数 f，无约束优化问题

$$\min_{\boldsymbol{w}} f(\boldsymbol{w})$$

由数学分析和函数极值点相关的知识最优解必然满足

$$\frac{\partial f}{\partial \boldsymbol{w}} = \nabla_{\boldsymbol{w}} f(\boldsymbol{w}) = 0$$

因而优化问题的求解可以通过对 f 求导并令导数等于 0 筛选出可能的极值点得到。

添加和可微函数 $h_i,\ i = 1, 2, \cdots, m$ 有关的等式约束

$$s.t.\ h_i(\boldsymbol{w}) = 0 \quad i = 1, 2, \cdots, m$$

在等式约束下优化问题的求解就没有那么容易了，我们考虑构造 Lagrangian 函数，其中 $\boldsymbol{\beta}$ 被称为 Lagrange 乘数（Lagrange multiplier）

$$L(\boldsymbol{w}, \boldsymbol{\beta}) = f(\boldsymbol{w}) + \sum_{i=1}^{m} \beta_i h_i(\boldsymbol{w})$$

由 Lagrange 乘数法上述优化问题的最优解必然满足 Lagrangian 函数对自变量的偏导为 0

$$\frac{\partial L}{\partial \boldsymbol{w}} = \nabla_{\boldsymbol{w}} f(\boldsymbol{w}) + \sum_{i=1}^{m} \beta_i \nabla_{\boldsymbol{w}} h_i(\boldsymbol{w}) = 0$$

和解在可行域内

$$h_i(\boldsymbol{w}) = 0$$

有时候约束还会以不等式的形式出现,添加和可微函数 $g_i,\ i=1,2,\cdots,n$ 有关的不等式约束，如此约束条件可以和写为

$$s.t.\ \begin{cases} g_i(\boldsymbol{w}) \leqslant 0 & i=1,2,\cdots,n \\ h_i(\boldsymbol{w}) = 0 & i=1,2,\cdots,m \end{cases}$$

Lagrangian 函数变为

$$L(\boldsymbol{w},\boldsymbol{\alpha},\boldsymbol{\beta}) = f(\boldsymbol{w}) + \sum_{i=1}^{n}\alpha_i g_i(\boldsymbol{w}) + \sum_{i=1}^{m}\beta_i h_i(\boldsymbol{w})$$

我们尝试一种全新的思路，这个思路借助了 minmax 问题的一些神奇的性质，记

$$\theta_{\mathcal{P}}(\boldsymbol{w}) := \max_{\boldsymbol{\alpha}\succcurlyeq\mathbf{0},\boldsymbol{\beta}} L(\boldsymbol{w},\boldsymbol{\alpha},\boldsymbol{\beta})$$

其中 $\mathcal{P}$ 表示原始的优化问题（primal problem)。可以证明使得 $\theta_{\mathcal{P}}(\boldsymbol{w})$ 取得最小值的解必然满足原始问题的约束。这其实是由最大值的性质限制的。设等式条件限制下的可行域为 $\mathcal{F}$，假设解 $\boldsymbol{w}\notin\mathcal{F}$，我们来证明[1]

$$\theta_{\mathcal{P}}(\boldsymbol{w}) = +\infty$$

假设最优解不满足某一项不等式约束

$$\begin{aligned} g_k(\boldsymbol{w}) > 0 \Rightarrow \theta_{\mathcal{P}}(\boldsymbol{w}) &\geqslant f(\boldsymbol{w}) + \sum_{i\neq k}\alpha_i g_i(\boldsymbol{w}) + \sum_{i=1}^{m}\beta_i h_i(\boldsymbol{w}) + \alpha_k g_k(\boldsymbol{w}) \\ &\Rightarrow \theta_{\mathcal{P}}(\boldsymbol{w}) \geqslant f(\boldsymbol{w}) + \sum_{i\neq k}\alpha_i g_i(\boldsymbol{w}) + \sum_{i=1}^{m}\beta_i h_i(\boldsymbol{w}) + \lim_{\alpha_k\to\infty}\alpha_k g_k(\boldsymbol{w}) = +\infty \end{aligned}$$

假设最优解不满足某一项等式约束，这里不妨假设 $h_i(\boldsymbol{w})>0$，对于 $h_i(\boldsymbol{w})<0$ 的情况由于对 β_i 符号没有约束，类似地取 $\beta_i\to-\infty$ 即可

$$\begin{aligned} h_k(\boldsymbol{w}) > 0 \Rightarrow \theta_{\mathcal{P}}(\boldsymbol{w}) &\geqslant f(\boldsymbol{w}) + \sum_{i=k}^{n}\alpha_i g_i(\boldsymbol{w}) + \sum_{i\neq k}\beta_i h_i(\boldsymbol{w}) + \beta_k h_k(\boldsymbol{w}) \\ &\Rightarrow \theta_{\mathcal{P}}(\boldsymbol{w}) \geqslant f(\boldsymbol{w}) + \sum_{i=k}^{n}\alpha_i g_i(\boldsymbol{w}) + \sum_{i\neq k}\beta_i h_i(\boldsymbol{w}) + \lim_{\beta_k\to\infty}\beta_k h_k(\boldsymbol{w}) = +\infty \end{aligned}$$

[1] https://zhuanlan.zhihu.com/p/32501517

我们得到了

$$\theta_{\mathcal{P}}(\boldsymbol{w}) = +\infty,\ \boldsymbol{w} \notin \mathcal{F}$$

如此我们也可以看出令 $\boldsymbol{\alpha} \succcurlyeq \mathbf{0}$ 的原因，如果这个条件不成立，在满足约束时也可以通过取极限的方式令 $\theta_{\mathcal{P}}(\boldsymbol{w})$ 趋于 $+\infty$，该条件限制了只有当不等式约束不满足时该性质才会成立。
对于 $\boldsymbol{w} \in \mathcal{F}$ 的情况

$$\theta_{\mathcal{P}}(\boldsymbol{w}) = \max_{\boldsymbol{\alpha} \succcurlyeq \mathbf{0}, \boldsymbol{\beta}} L(\boldsymbol{w}, \boldsymbol{\alpha}, \boldsymbol{\beta}) = f(\boldsymbol{w}) + \max_{\boldsymbol{\alpha} \succcurlyeq \mathbf{0}} \sum_{i=1}^{n} \alpha_i g_i(\boldsymbol{w}) = f(\boldsymbol{w})$$

取得最优解时 Lagrange 乘数 $\boldsymbol{\alpha}$ 取值满足

$$\alpha_i g_i(\boldsymbol{w}) = 0, i = 1, 2, \cdots, n$$

我们至少可以通过取 $\alpha_i = 0$ 得到上述结果。这一性质被称为互补松弛（Complementary slackness）。

因而我们最终得到

$$\theta_{\mathcal{P}}(\boldsymbol{w}) = \begin{cases} f(\boldsymbol{w}) & \boldsymbol{w} \in \mathcal{F} \\ +\infty & \boldsymbol{w} \notin \mathcal{F} \end{cases}$$

$\theta_{\mathcal{P}}(\boldsymbol{w})$ 的取值仅在可行域 $\mathcal{F}$ 中与 $f(\boldsymbol{w})$ 相等，这是非常好的性质。当考虑最小化问题时，由于在 $\mathcal{F}$ 外的点 $\theta_{\mathcal{P}}$ 取值为 $+\infty$，因而在可行域存在时可行域内 $\theta_{\mathcal{P}}$ 取值有限，在可行域外的点不可能成为 $\theta_{\mathcal{P}}$ 的最小值点，从而求约束条件下 $f(\boldsymbol{w})$ 的最小值问题可以转化为无约束条件下求解 $\theta_{\mathcal{P}}(\boldsymbol{w})$ 的最小值的问题。这是原始的优化问题，我们将原始问题的最优值记为 p^*

$$p^* = \min_{\boldsymbol{w} \in \mathcal{F}} f(\boldsymbol{w}) = \min_{\boldsymbol{w}} \theta_{\mathcal{P}}(\boldsymbol{w}) = \min_{\boldsymbol{w}} \max_{\boldsymbol{\alpha} \succcurlyeq \mathbf{0}, \boldsymbol{\beta}} L(\boldsymbol{w}, \boldsymbol{\alpha}, \boldsymbol{\beta})$$

且两者最优解对应的解集相同。

如果没有不等式约束，则问题就转化为了等式约束下的问题，此时由于没有 $\boldsymbol{\alpha}$ 的非负约束直接对 $\theta_{\mathcal{P}}$ Lagrangian 函数求两次导就好了，这就对应着我们之前提到的 Lagrange 乘数法。

我们发现了加上不等式约束后最优解起码满足如下条件：

条件一　解在原问题可行域内

$$\boldsymbol{w} \in \mathcal{F} \Rightarrow g_i(\boldsymbol{w}) \leqslant 0,\ h_j(\boldsymbol{w}) = 0,\ i = 1, 2, \cdots, n,\ j = 1, 2, \cdots, m$$

条件二　不等式约束对应的 Lagrange 乘数为正

$$\boldsymbol{\alpha} \succcurlyeq \mathbf{0}$$

条件三　互补松弛性

$$\alpha_i g_i(\boldsymbol{w}) = 0, i = 1, 2, \cdots, n$$

为什么没有 L 对 $\boldsymbol{w}$ 导数为 0 的条件呢？我们在 H.3.1 再谈。

H.2　Lagrange 对偶问题

H.2.1　对偶问题的引入

$\boldsymbol{\alpha}$ 的非负约束导致了最里层的最大化问题难以求解，而如果将 min 和 max 调换一下顺序最里层的求解会方便很多，这样转化后的问题称为（Lagrange）对偶问题（Dual problem），我们通常通过对这个问题的研究间接地得到原优化问题的一些性质

$$\max_{\boldsymbol{\alpha} \succcurlyeq \mathbf{0}, \boldsymbol{\beta}} \min_{\boldsymbol{w}} L(\boldsymbol{w}, \boldsymbol{\alpha}, \boldsymbol{\beta})$$

记

$$\theta_{\mathcal{D}}(\boldsymbol{\alpha}, \boldsymbol{\beta}) := \min_{\boldsymbol{w}} L(\boldsymbol{w}, \boldsymbol{\alpha}, \boldsymbol{\beta})$$

其中 $\mathcal{D}$ 表示对偶问题。对偶问题在求解如下的优化问题，我们将对偶问题的最优值记为 d^*

$$d^* = \max_{\boldsymbol{\alpha} \succcurlyeq \mathbf{0}, \boldsymbol{\beta}} \theta_{\mathcal{D}}(\boldsymbol{\alpha}, \boldsymbol{\beta}) = \max_{\boldsymbol{\alpha} \succcurlyeq \mathbf{0}, \boldsymbol{\beta}} \min_{\boldsymbol{w}} L(\boldsymbol{w}, \boldsymbol{\alpha}, \boldsymbol{\beta})$$

在原问题中，约束以 Lagrangian 函数中的等式和不等式约束对应的线性组合的形式出现，因而 $\theta_{\mathcal{P}}$ 对应的优化参数 $\boldsymbol{\alpha}$ 和 $\boldsymbol{\beta}$ 的个数可以视为约束的数量。容易发现原问题的约束的数量和对偶问题的变量个数是相同的，如果把 $\theta_{\mathcal{D}}$ 对应的优化参数 $\boldsymbol{w}$ 视为对偶问题中约束的数量，则对偶问题中约

束的数量和原问题的变量个数是相同的，这反映了原问题和对偶问题具有很强的对称性。

由于对于任意的 $\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{w}$ 满足

$$\begin{aligned} d = \theta_{\mathcal{D}}(\boldsymbol{\alpha}, \boldsymbol{\beta}) = \min_{\boldsymbol{w}} L(\boldsymbol{w}, \boldsymbol{\alpha_d}, \boldsymbol{\beta}) \leqslant L(\boldsymbol{w}, \boldsymbol{\alpha}, \boldsymbol{\beta}) \\ \leqslant \max_{\boldsymbol{\alpha} \succcurlyeq \mathbf{0}, \boldsymbol{\beta}} L(\boldsymbol{w}, \boldsymbol{\alpha}, \boldsymbol{\beta}) = \theta_{\mathcal{P}}(\boldsymbol{w}) = p \end{aligned}$$

得到对偶问题和原问题之间必然满足弱对偶定理（Weak duality theorem），即

$$d^* = \max_{\boldsymbol{\alpha} \succcurlyeq \mathbf{0}, \boldsymbol{\beta}} \theta_{\mathcal{D}}(\boldsymbol{\alpha}, \boldsymbol{\beta}) \leqslant \min_{\boldsymbol{w}} \theta_{\mathcal{P}}(\boldsymbol{w}) = p^*$$

弱对偶定理表明对偶问题和原问题最优值之间存在一个 duality gap。这是一个非常重要的性质，该性质有一个很重要的推论：若存在原问题和对偶问题的一组解 $\boldsymbol{w}^*$ 和 $\boldsymbol{\alpha}^*, \boldsymbol{\beta}^*$ 使得

$$\theta_{\mathcal{P}}(\boldsymbol{w}^*) = \theta_{\mathcal{D}}(\boldsymbol{\alpha}^*, \boldsymbol{\beta}^*)$$

成立，则 $\boldsymbol{w}^*$ 和 $\boldsymbol{\alpha}^*, \boldsymbol{\beta}^*$ 分别为原问题和对偶问题的一组最优解，此时 gap 消失，不等号对应的等号可以取得，强对偶（Strong duality）成立了。此时原问题和对偶问题满足互补松弛，即对于原问题的最优解 $\boldsymbol{w}^*$

$$p^* = f(\boldsymbol{w}^*) = \theta_{\mathcal{P}}(\boldsymbol{w}^*)$$

对于对偶问题的最优解 $\boldsymbol{\alpha}^*$ 和 $\boldsymbol{\beta}^*$

$$d^* = \theta_{\mathcal{D}}(\boldsymbol{\alpha}^*, \boldsymbol{\beta}^*)$$

满足

$$\alpha_i^* g_i(\boldsymbol{w}^*) = 0, \ i = 1, 2, \cdots, n$$

证明我们放在 H.3.2。

H.2.2　线性规划中的对偶问题

对于线性规划（Linear programming）问题的标准型（Standard form）[2]，这里设 $\boldsymbol{x}$ 为 n 维向量，$\boldsymbol{A}$ 为 $m \times n$ 矩阵

$$-p^* = \max_{\boldsymbol{x}} \boldsymbol{c^T x} \Rightarrow p^* = \min_{\boldsymbol{x}} -\boldsymbol{c^T x}$$

$$s.t. \begin{cases} \boldsymbol{Ax} \preccurlyeq \boldsymbol{b} \\ \boldsymbol{x} \succcurlyeq \boldsymbol{0} \end{cases}$$

考虑松弛变量（Slack variable）后，记

$$\boldsymbol{x} := \begin{pmatrix} \boldsymbol{x_n} \\ \boldsymbol{x_s} \end{pmatrix},\ \boldsymbol{x_s^T} \succcurlyeq \boldsymbol{0}$$

问题转为

$$\min_{\boldsymbol{x}} \boldsymbol{c^T x_n}$$

$$s.t. \begin{cases} (\boldsymbol{A}, \boldsymbol{I})\boldsymbol{x} = \boldsymbol{b} \\ \boldsymbol{x} \succcurlyeq \boldsymbol{0} \end{cases}$$

考虑 Lagrangian 函数

$$L(\boldsymbol{x}, \boldsymbol{z}, \boldsymbol{y}) = -\boldsymbol{c^T x_n} - \boldsymbol{z^T x} + \boldsymbol{y^T}(\boldsymbol{A}, \boldsymbol{I})\boldsymbol{x} - \boldsymbol{y^T b}$$

得到

$$\theta_{\mathcal{D}}(\boldsymbol{z}, \boldsymbol{y}) := \min_{\boldsymbol{x}} L(\boldsymbol{x}, \boldsymbol{z}, \boldsymbol{y}) = \min_{\boldsymbol{x}}(-(\boldsymbol{c^T}, \boldsymbol{0_s^T}) + (\boldsymbol{y^T A}, \boldsymbol{y^T}) - \boldsymbol{z^T})\boldsymbol{x} - \boldsymbol{y^T b}$$

当 $\boldsymbol{c^T} + \boldsymbol{y^T A} - \boldsymbol{z^T}$ 或 $\boldsymbol{y^T} - \boldsymbol{z^T}$ 存在一个维度的元素小于 0 时，我们可以取 $\boldsymbol{x}$ 对应的维度趋向 $+\infty$ 使得 $\theta_{\mathcal{D}}$ 趋向 $-\infty$，否则我们取 $\boldsymbol{x} = \boldsymbol{0}$ 使得 $\theta_{\mathcal{D}}$ 取得最小值为 $-\boldsymbol{y^T b}$，即

$$\theta_{\mathcal{D}}(\boldsymbol{z}, \boldsymbol{y}) = \begin{cases} -\boldsymbol{y^T b} & \boldsymbol{y^T A} - \boldsymbol{z^T} \succcurlyeq \boldsymbol{c^T},\ \boldsymbol{y^T} \succcurlyeq \boldsymbol{z^T} \\ -\infty & \text{otherwise} \end{cases}$$

[2] Frederick S. Hillier, Gerald J. Lieberman, *Introduction to Operations Research (Tenth Edition)*, p.34

我们设

$$\mathcal{F}=\{(\boldsymbol{z},\boldsymbol{y})\mid \boldsymbol{y}^{\boldsymbol{T}}\boldsymbol{A}-\boldsymbol{z}^{\boldsymbol{T}}\succcurlyeq \boldsymbol{c}^{\boldsymbol{T}},\ \boldsymbol{y}\succcurlyeq \boldsymbol{z}\succcurlyeq \boldsymbol{0}\}$$

假设 $\mathcal{F}$ 非空，我们得到 Lagrange 对偶问题为

$$d^*=\max_{\boldsymbol{z}\succcurlyeq \boldsymbol{0},\boldsymbol{y}}\min_{\boldsymbol{x}}\theta_{\mathcal{D}}(\boldsymbol{z},\boldsymbol{y})=\max_{(\boldsymbol{z},\boldsymbol{y})\in\mathcal{F}}-\boldsymbol{y}^{\boldsymbol{T}}\boldsymbol{b}\Rightarrow -d^*=\min_{(\boldsymbol{z},\boldsymbol{y})\in\mathcal{F}}\boldsymbol{y}^{\boldsymbol{T}}\boldsymbol{b}$$

我们注意到 $\boldsymbol{z}$ 不在目标函数内，考虑

$$\mathcal{F}(\boldsymbol{z})=\{\boldsymbol{y}\mid \boldsymbol{y}^{\boldsymbol{T}}\boldsymbol{A}-\boldsymbol{z}^{\boldsymbol{T}}\succcurlyeq \boldsymbol{c}^{\boldsymbol{T}},\ \boldsymbol{y}\succcurlyeq \boldsymbol{z}\},\ \boldsymbol{z}\succcurlyeq \boldsymbol{0}$$

则优化问题为

$$\min_{\boldsymbol{y}\in\mathcal{F}(\boldsymbol{z})}\boldsymbol{y}^{\boldsymbol{T}}\boldsymbol{b}$$

取 $\boldsymbol{z}=\boldsymbol{0}$ 可行域包含了所有的 $\mathcal{F}(\boldsymbol{z})$，即包含了问题所有可能出现的解

$$\forall \boldsymbol{z}\succcurlyeq \boldsymbol{0},\ \mathcal{F}(\boldsymbol{z})\subset\mathcal{F}(\boldsymbol{0})=\{\boldsymbol{y}\mid \boldsymbol{y}^{\boldsymbol{T}}\boldsymbol{A}\succcurlyeq \boldsymbol{c}^{\boldsymbol{T}},\ \boldsymbol{y}\succcurlyeq \boldsymbol{0}\}=:\mathcal{F}_y\Rightarrow \mathcal{F}_y=\bigcup_{\boldsymbol{z}\succcurlyeq \boldsymbol{0}}\mathcal{F}(\boldsymbol{z})$$

于是我们得到了对偶问题

$$-d^*=\min_{\boldsymbol{y}\in\mathcal{F}_y}\boldsymbol{y}^{\boldsymbol{T}}\boldsymbol{b}$$

即

$$\min_{\boldsymbol{y}}\boldsymbol{y}^{\boldsymbol{T}}\boldsymbol{b}$$

$$s.t.\ \begin{cases}\boldsymbol{y}^{\boldsymbol{T}}\boldsymbol{A}\succcurlyeq \boldsymbol{c}^{\boldsymbol{T}}\\ \boldsymbol{y}\succcurlyeq \boldsymbol{0}\end{cases}$$

这是我们所熟悉的线性规划的对偶问题。此时弱对偶成立

$$d^*\leqslant p^*\Rightarrow \min_{\boldsymbol{y}}\boldsymbol{y}^{\boldsymbol{T}}\boldsymbol{b}\geqslant\max_{\boldsymbol{x}}\boldsymbol{c}^{\boldsymbol{T}}\boldsymbol{x}$$

事实上我们还学习到了线性规划中强对偶也是成立，即

$$d^*=p^*\Rightarrow \min_{\boldsymbol{y}}\boldsymbol{y}^{\boldsymbol{T}}\boldsymbol{b}=\max_{\boldsymbol{x}}\boldsymbol{c}^{\boldsymbol{T}}\boldsymbol{x}$$

此时考虑松弛变量

$$(\boldsymbol{A},\boldsymbol{I})\boldsymbol{x}^* = \boldsymbol{A}\boldsymbol{x}_n^* + \boldsymbol{x}_s^* = \boldsymbol{b}$$

对对偶问题考虑剩余变量（Surplus variable），记

$$\boldsymbol{y} := \begin{pmatrix} \boldsymbol{y}_m \\ \boldsymbol{y}_s \end{pmatrix},\ \boldsymbol{y}_s \succcurlyeq \boldsymbol{0}$$

当对偶问题取最优解向对偶问题添加剩余变量

$$\boldsymbol{y}_m^{*\,\boldsymbol{T}}\boldsymbol{A} - \boldsymbol{y}_s^{*\boldsymbol{T}} = \boldsymbol{y}^{*\boldsymbol{T}}(\boldsymbol{A},\boldsymbol{I}) = \boldsymbol{c}^{\boldsymbol{T}}$$

得到

$$\boldsymbol{y}_m^{*\,\boldsymbol{T}}\boldsymbol{b} = \boldsymbol{c}^{\boldsymbol{T}}\boldsymbol{x}_n^* \Rightarrow \boldsymbol{y}_m^{*\,\boldsymbol{T}}(\boldsymbol{A}\boldsymbol{x}_n^*+\boldsymbol{x}_s^*) = (\boldsymbol{y}_m^{*\,\boldsymbol{T}}\boldsymbol{A}-\boldsymbol{y}_s^*)^T\boldsymbol{x}_n^* \Rightarrow \boldsymbol{y}_m^{*\,\boldsymbol{T}}\boldsymbol{x}_s^*+\boldsymbol{y}_s^{*\boldsymbol{T}}\boldsymbol{x}_n^* = 0$$

由 $\boldsymbol{x}^*, \boldsymbol{y}^*$ 的非负性立即得到

$$\boldsymbol{y}_m^{*\,\boldsymbol{T}}\boldsymbol{x}_s^* = \boldsymbol{y}_s^{*\boldsymbol{T}}\boldsymbol{x}_n^* = 0$$

这是线性规划中的互补松弛定理，在原问题和对偶问题分别达到最优时，原问题的松弛变量和对偶问题原变量其中至少有一个为 0。对于线性规划而言事实上强互补松弛定理（Strict complementary slackness）也是成立的，即原问题和对偶问题存在一个最优解，使得原问题的松弛变量和对偶问题原变量有且仅有一个为 0 [3]。无论是线性规划中的强对偶、互补松弛和进一步的强互补松弛，即使是不借助凸优化的进阶的理论，在学习过最优化理论这门课程后，我们借助对偶思想，通过单纯形法（Simplex method）确定的 simplex 表（Simplex tableau）也能将这些结论直观地证明，从中可以看出单纯形法对于线性规划理论的重要性。

从线性规划问题中我们可以明显地感受到而原问题的约束越多，对偶问题整理后得到的变量也越多，而原问题的变量越多，对偶问题整理后得到的对应的约束项也越多。某些问题的原问题的变量的数量是十分惊人的，且约束条件相对而言比较复杂，这有时候会问题的求解带来很多的困扰，

[3] Luenberger, David G., Ye, Yinyu, *Linear and Nonlinear Programming (Fifth Edition)*, p.54

因而在这个时候考虑对偶问题、根据对偶问题设计优化算法也许会有所帮助。例如我们学过的在最优化理论中非常重要的最优运输问题（Optimal transportation problem）和接下来要重点推导的 SVM 中的优化问题。

对于一般的最优运输问题，考虑 n 个供应端和 m 个需求端，第 i 个供应端需要向需求端供应的物资的总和为 s_i，第 j 个需求端需要接收的物资的总和为 d_j，两者之间的运输的成本为 c_{ij}，运送量为 x_{ij}，则最小化运输成本的问题可以写为如下线性规划问题

$$\min_{\boldsymbol{X}} \sum_{i=1}^{n}\sum_{j=1}^{m} c_{ij}x_{ij}$$

$$s.t. \begin{cases} \sum\limits_{j=1}^{m} x_{ij} = s_i & i = 1,2,\cdots,n \\ \sum\limits_{i=1}^{n} x_{ij} = d_j & j = 1,2,\cdots,m \\ x_{ij} \geqslant 0 & i = 1,2,\cdots,n,\ j = 1,2,\cdots,m \end{cases}$$

在统计学中最小化运输的成本称为分布 $(s_1, s_2, \cdots, s_n)$ 和 $(d_1, d_2, \cdots, d_m)$ 的 Wasserstein 距离（Wasserstein distance），它度量了两个分布之间的最小转移代价[4]。该问题的参数量为 mn，不考虑 x_{ij} 的非负约束时约束的数量为 $m+n$。上式实际上暗含还了总供应量和需求量相等的条件

$$\sum_{i=1}^{n} s_i = \sum_{j=1}^{m} d_j$$

将上式转化为对偶问题，则对偶问题为

$$\max_{\boldsymbol{u},\boldsymbol{v}} \sum_{i=1}^{n} s_i u_i + \sum_{j=1}^{m} d_i v_i$$

$$s.t.\ u_i + v_j \leqslant c_{ij},\ i = 1,2,\cdots,n,\ j = 1,2,\cdots,m$$

此时参数量被削减为 $m+n$ 且约束的数量为 mn，问题在形式上大大简化了。事实上将原问题转化为对偶问题是使用运输单纯形法（Transportation simplex method）求解最优运输问题求解的关键[5]。

[4] Luenberger, David G., Ye, Yinyu, *Linear and Nonlinear Programming (Fifth Edition)*, p.18

[5] Frederick S. Hillier, Gerald J. Lieberman, *Introduction to Operations Research (Tenth Edition)*, pp.333-347

H.3　KKT 条件

H.3.1　KKT 条件与 CQ

$\boldsymbol{\alpha}$ 的非负约束最内层的最大值问题的求解有时候比较困难，不等式约束和等式约束相比通常要复杂得多。在第一节我们求得了解所需要满足的三个必要的条件，联系 Lagrange 乘数法我们很想加上令 Lagrangian 函数对参数 $\boldsymbol{w}$ 求导得到的导函数为 0 的条件。但是在有些情况下这样做是不行的，因为在某些极端情况下我们得到的导函数在可行域内没有零点。举一个简单的例子[6]

$$\min_x x$$
$$s.t.\ x^2 \leqslant 0$$

以上例子中可行域为 $\{0\}$，显然最优解为 0，但是对其 Lagrangian 函数求导得到

$$\frac{\partial}{\partial x}(x - \lambda x^2) = 1 - 2\lambda x = 1 \neq 0$$

我们发现导函数在可行域内没有零点了。

令人惊喜的是当函数 f，g_i 和 h_i 满足某些特殊的约束时，最优解满足导函数为 0 的条件，这样的条件称为约束的规范性条件（Constraint Qualification / CQ），此时我们得到了在优化问题中非常重要的 KKT 条件（Karush–Kuhn–Tucker conditions）。在可行域内的解满足 CQ 的条件下，带约束的优化问题

$$\min_{\boldsymbol{w}} f(\boldsymbol{w})$$
$$s.t. \begin{cases} g_i(\boldsymbol{w}) \leqslant 0 & i = 1, 2, \cdots, n \\ h_i(\boldsymbol{w}) = 0 & i = 1, 2, \cdots, m \end{cases}$$

的最优解的必要条件可以写为：

平稳性　对自变量 $\boldsymbol{w}$ 偏导数为 0，该性质被称为平稳性（Stationarity）

$$\frac{\partial L}{\partial \boldsymbol{w}} = \nabla_{\boldsymbol{w}} f(\boldsymbol{w}) + \sum_{i=1}^{n} \alpha_i \nabla_{\boldsymbol{w}} g_i(\boldsymbol{w}) + \sum_{i=1}^{m} \beta_i \nabla_{\boldsymbol{w}} h_i(\boldsymbol{w}) = 0$$

[6] https://www.zhihu.com/question/49754458

原问题可行性　解在原问题可行域内，这一性质被称为原问题的可行性（Primal feasible）

$$\boldsymbol{w} \in \mathcal{F} \Rightarrow g_i(\boldsymbol{w}) \leqslant 0,\ h_j(\boldsymbol{w}) = 0,\ i = 1, 2, \cdots, n,\ j = 1, 2, \cdots, m$$

对偶问题可行性　不等式约束对应的 Lagrange 乘数为正

$$\boldsymbol{\alpha} \succcurlyeq \mathbf{0}$$

这一性质也被称为对偶问题的可行性（Dual feasible），原因可以在接下来的阅读过程中感受到。

互补松弛性　互补松弛性质成立

$$\alpha_i g_i(\boldsymbol{w}) = 0, i = 1, 2, \cdots, n$$

相较于我们前面证明已经的三个必要条件外，还多出了令人欣喜的导函数为 0 的条件，因为这个条件一般在求解的时候比较好用。

以下列出了几个常用的 CQ [7]：

LCQ　若 g_i 和 h_i 均为仿射函数，即约束是线性的，则 LCQ（linearity constraint qualification）成立。

LICQ　若“起作用的”(active) g_i（即对应的取等的情况）和所有的 h_i 在解对应的梯度线性无关，则 LICQ（Linear independence constraint qualification）成立。

SCQ / SC　若 f 和 g_i 均为凸函数，且 h_i 为仿射函数，即优化问题为凸优化问题，且可行域存在 $\boldsymbol{x}$ 使得所有的 $g_i(\boldsymbol{x}) < 0$ 和 $h_i(\boldsymbol{x}) = 0$ 均成立，即可行域是严格的，则 SCQ（Slater’s constraint qualication）/ SC（Slater’s condition）成立。

H.3.2　KKT 条件与强对偶

可行域内的解满足 CQ 是强对偶成立的必要条件，即当优化问题的强对偶成立且最优解都能取到时，CQ 都会得到满足。我们利用对偶问题解得

[7] https://en.wikipedia.org/wiki/Karush%E2%80%93Kuhn%E2%80%93Tucker_conditions

的一组解来构造原问题对应的 Lagrange 乘数[8]。

对于带约束的优化问题

$$\min_{\boldsymbol{w}} f(\boldsymbol{w})$$

$$s.t. \begin{cases} g_i(\boldsymbol{w}) \leqslant 0 & i = 1, 2, \cdots, n \\ h_i(\boldsymbol{w}) = 0 & i = 1, 2, \cdots, m \end{cases}$$

对于原问题的最优解 $(\boldsymbol{w}^*, \boldsymbol{\alpha_0}, \boldsymbol{\beta_0})$

$$p^* = f(\boldsymbol{w}^*) = \theta_{\mathcal{P}}(\boldsymbol{w}^*)$$

其满足原问题可行性

$$g_i(\boldsymbol{w}^*) \leqslant 0,\ h_j(\boldsymbol{w}^*) = 0,\ i = 1, 2, \cdots, n,\ j = 1, 2, \cdots, m$$

且由于

$$\theta_{\mathcal{P}}(\boldsymbol{w}^*) = \max_{\boldsymbol{\alpha} \succcurlyeq \mathbf{0}, \boldsymbol{\beta}} L(\boldsymbol{w}, \boldsymbol{\alpha}, \boldsymbol{\beta}) = L(\boldsymbol{w}^*, \boldsymbol{\alpha_0}, \boldsymbol{\beta_0})$$

满足 $\boldsymbol{\alpha}$ 与 $\boldsymbol{\beta}$ 的最优性

$$\forall \boldsymbol{\alpha} \succcurlyeq \mathbf{0}, \boldsymbol{\beta},\ p^* = L(\boldsymbol{w}^*, \boldsymbol{\alpha_0}, \boldsymbol{\beta_0}) \geqslant L(\boldsymbol{w}^*, \boldsymbol{\alpha}, \boldsymbol{\beta})$$

和互补松弛性

$$\alpha_{0i} g_i(\boldsymbol{w}^*) = 0,\ i = 1, 2, \cdots, n$$

对于对偶问题的最优解 $(\boldsymbol{w_0}, \boldsymbol{\alpha}^*, \boldsymbol{\beta}^*)$

$$d^* = \theta_{\mathcal{D}}(\boldsymbol{\alpha}^*, \boldsymbol{\beta}^*) = f(\boldsymbol{w_0}) + \sum_{i=1}^{n} \alpha_i^* g_i(\boldsymbol{w_0}) + \sum_{i=1}^{m} \beta_i^* h_i(\boldsymbol{w_0})$$

满足对偶问题可行性

$$\boldsymbol{\alpha}^* \succcurlyeq \mathbf{0}$$

且由于

[8] http://www.stat.cmu.edu/~ryantibs/convexopt/lectures/kkt.pdf

$$\theta_{\mathcal{D}}(\boldsymbol{\alpha}^*, \boldsymbol{\beta}^*) = \min_{\boldsymbol{w}} L(\boldsymbol{w}, \boldsymbol{\alpha}^*, \boldsymbol{\beta}^*) = L(\boldsymbol{w_0}, \boldsymbol{\alpha}^*, \boldsymbol{\beta}^*)$$

满足 $\boldsymbol{w}$ 的最优性，故

$$\frac{\partial L}{\partial \boldsymbol{w}} = \nabla_{\boldsymbol{w}} f(\boldsymbol{w_0}) + \sum_{i=1}^{n} \alpha_i^* \nabla_{\boldsymbol{w}} g_i(\boldsymbol{w_0}) + \sum_{i=1}^{m} \beta_i^* \nabla_{\boldsymbol{w}} h_i(\boldsymbol{w_0}) = 0$$

且

$$\forall \boldsymbol{w},\ L(\boldsymbol{w}, \boldsymbol{\alpha}^*, \boldsymbol{\beta}^*) \geqslant L(\boldsymbol{w_0}, \boldsymbol{\alpha}^*, \boldsymbol{\beta}^*) = d^*$$

我们尝试利用强对偶将这些变量的信息联系起来

$$p^* = L(\boldsymbol{w}^*, \boldsymbol{\alpha_0}, \boldsymbol{\beta_0}) \geqslant L(\boldsymbol{w}^*, \boldsymbol{\alpha}^*, \boldsymbol{\beta}^*) \geqslant L(\boldsymbol{w_0}, \boldsymbol{\alpha}^*, \boldsymbol{\beta}^*) = d^*$$

由于 $p^* = d^*$ 这迫使

$$p^* = L(\boldsymbol{w}^*, \boldsymbol{\alpha_0}, \boldsymbol{\beta_0}) = L(\boldsymbol{w}^*, \boldsymbol{\alpha}^*, \boldsymbol{\beta}^*) = L(\boldsymbol{w_0}, \boldsymbol{\alpha}^*, \boldsymbol{\beta}^*) = d^*$$

由弱对偶的推论得到 $(\boldsymbol{w}^*, \boldsymbol{\alpha}^*, \boldsymbol{\beta}^*)$ 就是原问题和对偶问题的一个公共的最优解。其满足上述最优解给出的所有性质，即 $\boldsymbol{w}^*$ 的平稳性，原问题可行性，对偶问题可行性，互补松弛性质。我们也因而证明了上一节没有证明的强对偶问题满足互补松弛性。

我们刚刚实际上证明了在强对偶条件成立的情况下满足 KKT 条件对于原问题和对偶问题而言解的最优性判定而言是必要的，这实际上暗示了一点，在强对偶条件成立的情况下，解满足 KKT 条件不仅是原问题解为最优的必要条件，还是对偶问题的解为最优解的必要条件。回顾 H.2.1 的开头部分我们因为原问题不好解而将原问题转化为对偶问题，强对偶保证了我们求解对偶问题也能得到原问题的最优解，而强对偶推得的 KKT 条件保证了我们能以一种比较简单和系统的方式去将最优解筛出来，因而求解对偶问题的合理性和可行性已经得到了保证。

事实上满足 KKT 条件的解 $(\boldsymbol{w}^*, \boldsymbol{\alpha}^*, \boldsymbol{\beta}^*)$ 在特定情况下可以构造出原问题的解 $\boldsymbol{w}^*$ 和对偶问题的解 $\boldsymbol{\alpha}^*$，$\boldsymbol{\beta}^*$。此时满足 KKT 条件对于解的最优性判定而言是充分的[8]。

在问题为凸优化问题（Convex problem），即 f 和 g_i 均为凸函数，且 h_i 为仿射函数时，由于仿射函数为凸函数，仿射函数的线性组合仍为仿射函数即凸函数，由凸函数的非负线性组合仍是凸函数，因而得到 L 也是凸函数（g_i 的线性组合的非负性由对偶问题可行性保证）。此时我们能够保证是解最优的充分条件。我们以前可能听说过凸优化问题，更广义的凸优化问题指的是目标函数为凸函数，可行域为凸集时的优化问题，我们是怎么通过对函数的约束推出约束后的问题为凸优化问题的呢？对于凸函数而言，其划定的可行域

$$\mathcal{F}_f := \{\boldsymbol{x} \mid f(\boldsymbol{x}) \leqslant 0\}$$

是一个凸集，凸集的交也是一个凸集，下面我们讨论等式约束的情况，等式约束很容易转换为两个不等式约束

$$h(\boldsymbol{x}) = 0 \Rightarrow \begin{cases} h(\boldsymbol{x}) \leqslant 0 \\ -h(\boldsymbol{x}) \leqslant 0 \end{cases}$$

当前仅当 $h(\boldsymbol{x})$ 是一个仿射函数时，函数在取负前后仍能保持凸性。于是我们证明了我们对函数的约束确实使得问题变成了一个凸优化问题。

在给定条件下，由原问题可行性得到

$$f(\boldsymbol{w}^*) = \theta_{\mathcal{P}}(\boldsymbol{w}^*)$$

通过极值点条件（平稳性）和 L 的凸性我们得到

$$L(\boldsymbol{w}^*, \boldsymbol{\alpha}^*, \boldsymbol{\beta}^*) = \min_{\boldsymbol{w}} L(\boldsymbol{w}, \boldsymbol{\alpha}^*, \boldsymbol{\beta}^*) = \theta_{\mathcal{D}}(\boldsymbol{\alpha}^*, \boldsymbol{\beta}^*)$$

考虑互补松弛条件得到

$$L(\boldsymbol{w}^*, \boldsymbol{\alpha}^*, \boldsymbol{\beta}^*) = f(\boldsymbol{w}^*)$$

于是我们又构造出了连接原问题和对偶问题的桥梁

$$\theta_{\mathcal{P}}(\boldsymbol{w}^*) = \theta_{\mathcal{D}}(\boldsymbol{\alpha}^*, \boldsymbol{\beta}^*)$$

弱对偶的推论使得 $f(\boldsymbol{w}^*)$ 和 $L(\boldsymbol{w}^*, \boldsymbol{\alpha}^*, \boldsymbol{\beta}^*)$ 分别成为原问题和对偶问题的最优解，强对偶成立了。因而在凸优化问题的限制下可行域内的解满足 CQ 条件对于强对偶而言不仅是必要的，还是充分的。

在可微优化中，强对偶（或者更弱一点的 CQ 条件）赋予了 KKT 条件在解的最优性判定中的必要性，而凸优化问题的条件赋予了 KKT 条件在原问题和对偶问题解的最优性判定中的充分性，在强对偶和凸优化问题的条件同时成立时，KKT 条件在解的最优性判定中上升为充要条件[9]。借助作为必要条件的 KKT 条件我们可以排除大部分非最优解的情况，借助作为充要条件的 KKT 条件我们可以对解的最优性进行判定，即 optimal test。这两个重要的条件都可以由 SCQ 条件保证，因而 SCQ 是很常用的 CQ 条件。

线性规划问题是一类非常特殊的凸优化问题，相较于一般的凸优化问题其性质更加优秀，我们在上一小节和最优化课程的学习中已经能比较直观地感受到了这一点。我们可以使用 LCQ 得到线性规划满足 CQ 条件，由于线性规划是凸优化问题，因而线性规划又满足强对偶和对应的互补松弛定理。

H.4　Dual SVM

H.4.1　Hard-margin SVM

回顾最基础的 SVM（hard-margin SVM），SVM 的 motivation 在于我们要在样本点之间建立一个两个平行的分界面（决策边界），使得样本点分布在分界面之外且相同类别的样本分布在同一侧，要使得我们的分类达到最优，模型更加 robust，我们应当使得两个分界面之间的距离最大，因而 SVM 最初的优化问题可以表述为

$$\max_{\boldsymbol{w},b} \frac{2}{\|\boldsymbol{w}\|_2}$$
$$s.t.\ y_n(\boldsymbol{w}^{\boldsymbol{T}}\boldsymbol{x_n} - b) \geqslant 1,\ n = 1, 2, \cdots, N$$

将问题转化为易于求解的二次规划的形式

$$\min_{\boldsymbol{w},b} \frac{1}{2}\|\boldsymbol{w}\|_2^2$$
$$s.t.\ y_n(\boldsymbol{w}^{\boldsymbol{T}}\boldsymbol{x_n} - b) \geqslant 1,\ n = 1, 2, \cdots, N$$

这是一个满足 LCQ 条件的凸优化问题，因而考虑构造 Lagrangian 函数

$$L(\boldsymbol{w}, b, \boldsymbol{c}) = \frac{1}{2}\|\boldsymbol{w}\|_2^2 - \sum_{n=1}^{N} c_n(y_n(\boldsymbol{w}^{\boldsymbol{T}}\boldsymbol{x_n} - b) - 1)$$

[9] https://www.zhihu.com/question/23311674/answer/2439141067

考虑 KKT 条件（这里由于 SCQ 条件的保证 KKT 条件对最优解而言是充要的），我们有

平稳性，这里取 $\boldsymbol{X}$ 为列向量排列得到的 $D \times N$ 矩阵

$$\begin{aligned}\frac{\partial}{\partial \boldsymbol{w}} L &= \frac{1}{2}\nabla_{\boldsymbol{w}} \|\boldsymbol{w}\|_2^2 - \sum_{n=1}^{N} c_n y_n \nabla_{\boldsymbol{w}}(\boldsymbol{w}^{\boldsymbol{T}} \boldsymbol{x}_{\boldsymbol{n}}) \\ &= \boldsymbol{w} - \sum_{n=1}^{N} c_n y_n \boldsymbol{x}_{\boldsymbol{n}} = \boldsymbol{w} - \boldsymbol{X}(\boldsymbol{c} \odot \boldsymbol{y}) = 0 \\ \frac{\partial}{\partial b} L &= \sum_{n=1}^{N} c_n y_n \nabla_b b = \sum_{n=1}^{N} c_n y_n = \boldsymbol{c}^{\boldsymbol{T}} \boldsymbol{y} = 0\end{aligned}$$

原问题可行性

$$y_n(\boldsymbol{w}^{\boldsymbol{T}} \boldsymbol{x}_{\boldsymbol{n}} - b) \geqslant 1,\ n = 1, 2, \cdots, N$$

对偶问题可行性

$$\boldsymbol{c} \succcurlyeq \boldsymbol{0}$$

互补松弛性

$$c_n(y_n(\boldsymbol{w}^{\boldsymbol{T}} \boldsymbol{x}_{\boldsymbol{n}} - b) - 1) = 0,\ n = 1, 2, \cdots, N$$

KKT 对解的最优性判定而言是至少是必要条件，因而对于其提供的对我们解决问题具有帮助的约束我们可以直接将其添加进问题中而使得问题发生等价转换，我们需要考虑的是，这样的约束对于问题的求解是否有所帮助，是否会使得问题变得更加复杂。例如不少同学都看出了我们可以直接把互补松弛条件添加进约束从而使得目标函数一下子消去了很多项，从而使得变量的数量大大削减了。这个操作当然是被允许的，然而这样的操作以引入我们比较难求解的互补松弛的约束条件为代价（互补松弛条件中变量之间相互耦合的关系是我们很不希望看到的），因而我们需要尽量去多观察、多尝试。考虑展开 Lagrangian 函数

$$\begin{aligned} L(\boldsymbol{w}, b, \boldsymbol{c}) &= \frac{1}{2}\|\boldsymbol{w}\|_2^2 - \sum_{n=1}^{N} c_n(y_n(\boldsymbol{w}^{\boldsymbol{T}}\boldsymbol{x_n} - b) - 1) \\ &= \frac{1}{2}\boldsymbol{w}^{\boldsymbol{T}}\boldsymbol{w} - \boldsymbol{w}^{\boldsymbol{T}}\sum_{n=1}^{N} c_n y_n \boldsymbol{x_n} + b\sum_{n=1}^{N} c_n y_n + \sum_{n=1}^{N} c_n \\ &= \frac{1}{2}\boldsymbol{w}^{\boldsymbol{T}}\boldsymbol{w} - \boldsymbol{w}^{\boldsymbol{T}}\boldsymbol{X}(\boldsymbol{c}\odot\boldsymbol{y}) + b\boldsymbol{c}^{\boldsymbol{T}}\boldsymbol{y} + \mathbf{1}_{\boldsymbol{N}}^{\boldsymbol{T}}\boldsymbol{c} \end{aligned}$$

引入约束条件

$$\boldsymbol{w} - \boldsymbol{X}(\boldsymbol{c}\odot\boldsymbol{y}) = 0$$

$$\boldsymbol{c}^{\boldsymbol{T}}\boldsymbol{y} = 0$$

考虑最优解具有的形式，使用 KKT 条件对上式进行化简

$$L(\boldsymbol{w}, b, \boldsymbol{c}) = \mathbf{1}_{\boldsymbol{N}}^{\boldsymbol{T}}\boldsymbol{c} - \frac{1}{2}(\boldsymbol{c}\odot\boldsymbol{y})^T\boldsymbol{X}^{\boldsymbol{T}}\boldsymbol{X}(\boldsymbol{c}\odot\boldsymbol{y})$$

我们通过两个约束把原问题对应的变量全部消除了，因而我们可以考虑对偶问题

$$\max_{\boldsymbol{c}\succcurlyeq\mathbf{0}}\min_{\boldsymbol{w},b} L(\boldsymbol{w}, b, \boldsymbol{c}) = \max_{\boldsymbol{c}\succcurlyeq\mathbf{0}} \mathbf{1}_{\boldsymbol{N}}^{\boldsymbol{T}}\boldsymbol{c} - \frac{1}{2}(\boldsymbol{c}\odot\boldsymbol{y})^T\boldsymbol{X}^{\boldsymbol{T}}\boldsymbol{X}(\boldsymbol{c}\odot\boldsymbol{y})$$

因而我们的 Lagrange 对偶问题实际上可以转化为

$$\max_{\boldsymbol{c}} \mathbf{1}_{\boldsymbol{N}}^{\boldsymbol{T}}\boldsymbol{c} - \frac{1}{2}(\boldsymbol{c}\odot\boldsymbol{y})^T\boldsymbol{X}^{\boldsymbol{T}}\boldsymbol{X}(\boldsymbol{c}\odot\boldsymbol{y})$$

$$s.t. \begin{cases} \boldsymbol{w} - \boldsymbol{X}(\boldsymbol{c}\odot\boldsymbol{y}) = 0 \\ \boldsymbol{c}^{\boldsymbol{T}}\boldsymbol{y} = 0 \\ \boldsymbol{c}\succcurlyeq\mathbf{0} \end{cases}$$

我们总是可以通过直接取已经被剔除优化问题的 $\boldsymbol{w} = \boldsymbol{X}(\boldsymbol{c}\odot\boldsymbol{y})$ 来保证第一个条件是可行的，从而原问题转化为经典的二次规划（Quadratic programming）问题

$$\max_{\boldsymbol{c}} \mathbf{1}_{\boldsymbol{N}}^{\boldsymbol{T}}\boldsymbol{c} - \frac{1}{2}(\boldsymbol{c}\odot\boldsymbol{y})^T\boldsymbol{X}^{\boldsymbol{T}}\boldsymbol{X}(\boldsymbol{c}\odot\boldsymbol{y})$$

$$s.t. \begin{cases} \boldsymbol{c}^{\boldsymbol{T}}\boldsymbol{y} = 0 \\ \boldsymbol{c}\succcurlyeq\mathbf{0} \end{cases}$$

观察到 $\boldsymbol{X^T X}$ 实际上是一个 Euclidean 空间的内积矩阵。我们考虑使用核技巧将数据转至特征空间得到[10]

$$\max_{\boldsymbol{c}} \mathbf{1}_{\boldsymbol{N}}^{\boldsymbol{T}}\boldsymbol{c} - \frac{1}{2}(\boldsymbol{c}\odot\boldsymbol{y})^T\phi(\boldsymbol{X})^T\phi(\boldsymbol{X})(\boldsymbol{c}\odot\boldsymbol{y})$$

考虑特征空间上的内积矩阵，即核矩阵

$$\boldsymbol{K} = \phi(\boldsymbol{X})^T\phi(\boldsymbol{X})$$

得到

$$\max_{\boldsymbol{c}} \mathbf{1}_{\boldsymbol{N}}^{\boldsymbol{T}}\boldsymbol{c} - \frac{1}{2}(\boldsymbol{c}\odot\boldsymbol{y})^T\boldsymbol{K}(\boldsymbol{c}\odot\boldsymbol{y})$$

$$s.t.\ \begin{cases}\boldsymbol{c^T y} = 0\\ \boldsymbol{c} \succcurlyeq \mathbf{0}\end{cases}$$

这个问题的求解我们在这里不讨论，请感兴趣的同学自行查找相关资料。

H.4.2　Soft-margin SVM

SVM 的分类要求不等式约束严格成立，即我们确定的决策边界是严格的，或者说硬的，然而当噪声存在时这样的条件有时候很难做到，我们可以使用核技巧来缓解一下这个问题，当然也可以把分类边界放得宽松一点，即考虑软边界。

定义越界惩罚项

$$\xi_n = \max\{0, 1 - y_n(\boldsymbol{w^T x_n} - b)\}$$

当不等式约束不成立时，我们设计的惩罚项将大于 0。考虑将越界损失加入原问题

$$\min_{\boldsymbol{w},b} \frac{1}{N}\sum_{n=1}^{N}\xi_n + \lambda\|\boldsymbol{w}\|_2^2$$

此时目标函数中分界面距离成为 L2 正则项，前一项惩罚项的和称为 hinge 损失（Hinge loss），代表了数据的总的越界损失。由惩罚项确定的约束代替原始的硬的不等式约束得到

[10]详见 C.3 结尾部分

$$\begin{aligned}\xi_n &= \max\{0, 1 - y_n(\boldsymbol{w}^{\boldsymbol{T}}\boldsymbol{x_n} - b)\} \\ &\Rightarrow \xi_n \geqslant 0,\ \xi_n \geqslant 1 - y_n(\boldsymbol{w}^{\boldsymbol{T}}\boldsymbol{x_n} - b),\ n = 1, 2, \cdots, N\end{aligned}$$

我们事实上将 ξ_n 松弛为 $\max\{0, 1 - y_n(\boldsymbol{w}^{\boldsymbol{T}}\boldsymbol{x_n} - b)\}$ 的一个上界，最小化迫使这个上界被取到，因而我们将优化问题最终写为

$$\min_{\boldsymbol{w},b} \frac{1}{N}\sum_{n=1}^{N}\xi_n + \lambda \left\|\boldsymbol{w}\right\|_2^2$$

$$s.t. \begin{cases} y_n(\boldsymbol{w}^{\boldsymbol{T}}\boldsymbol{x_n} - b) \geqslant 1 - \xi_n \\ \xi_n \geqslant 0 \end{cases},\ n = 1, 2, \cdots, N$$

该问题同样是一个满足 LCQ 条件的凸优化问题，因而考虑构造 Lagrangian 函数，其形式和 Hard-margin SVM 相比只多了几项，推导流程完全相同

$$L(\boldsymbol{w}, b, \boldsymbol{\xi}, \boldsymbol{c}, \boldsymbol{\tau}) = \frac{1}{N}\sum_{n=1}^{N}\xi_n + \lambda \left\|\boldsymbol{w}\right\|_2^2 - \sum_{n=1}^{N} c_n(y_n(\boldsymbol{w}^{\boldsymbol{T}}\boldsymbol{x_n} - b) - 1 + \xi_n) + \sum_{n=1}^{N}\tau_n\xi_n$$

考虑 KKT 条件

平稳性

$$\begin{aligned}\frac{\partial}{\partial \boldsymbol{w}}L &= \lambda\nabla_{\boldsymbol{w}}\left\|\boldsymbol{w}\right\|_2^2 - \sum_{n=1}^{N} c_n y_n \nabla_{\boldsymbol{w}}(\boldsymbol{w}^{\boldsymbol{T}}\boldsymbol{x_n}) \\ &= 2\lambda\boldsymbol{w} - \sum_{n=1}^{N} c_n y_n \boldsymbol{x_n} = 2\lambda\boldsymbol{w} - \boldsymbol{X}(\boldsymbol{c} \odot \boldsymbol{y}) = 0 \\ \frac{\partial}{\partial b}L &= \sum_{n=1}^{N} c_n y_n \nabla_b b = \sum_{n=1}^{N} c_n y_n = \boldsymbol{c}^{\boldsymbol{T}}\boldsymbol{y} = 0 \\ \frac{\partial}{\partial \boldsymbol{\xi}}L &= \frac{1}{N}\sum_{n=1}^{N}\nabla_{\boldsymbol{\xi}}\xi_n - \sum_{n=1}^{N} c_n \nabla_{\boldsymbol{\xi}}\xi_n + \sum_{n=1}^{N}\tau_n\nabla_{\boldsymbol{\xi}}\xi_n = \frac{1}{N}\boldsymbol{1_N} - \boldsymbol{c} + \boldsymbol{\tau}\end{aligned}$$

原问题可行性

$$y_n(\boldsymbol{w}^{\boldsymbol{T}}\boldsymbol{x_n} - b) \geqslant 1 - \xi_n,\ \xi_n \geqslant 0,\ n = 1, 2, \cdots, N$$

对偶问题可行性

$$\boldsymbol{c} \succcurlyeq \mathbf{0_N}$$

$$\boldsymbol{\tau} \succcurlyeq \mathbf{0_N}$$

互补松弛性

$$c_n(y_n(\boldsymbol{w^T x_n} - b) - 1 + \xi_n) = 0,\ \tau_n \xi_n = 0,\ n = 1, 2, \cdots, N$$

注意到 $\boldsymbol{\xi}$ 对应的项是线性的，我们在线性规划的对偶问题的求解中看到原问题的变量 $\boldsymbol{\xi}$ 可以转化为和变量个数相等的线性约束条件。考虑对偶问题，记 L 中不含 $\boldsymbol{\xi}$（当然也不含 $\boldsymbol{\tau}$）的项为 $R(\boldsymbol{w}, b, \boldsymbol{c})$

$$\max_{\boldsymbol{c},\boldsymbol{\tau}\succcurlyeq\mathbf{0}} \min_{\boldsymbol{w},b} L(\boldsymbol{w}, b, \boldsymbol{\xi}, \boldsymbol{c}, \boldsymbol{\tau}) = \max_{\boldsymbol{c}\succcurlyeq\mathbf{0}} \min_{\boldsymbol{w},b} R(\boldsymbol{w}, b, \boldsymbol{c}) + \max_{\boldsymbol{c},\boldsymbol{\tau}\succcurlyeq\mathbf{0}} \min_{\boldsymbol{\xi}} \left(\frac{1}{N}\mathbf{1_N} - \boldsymbol{c} + \boldsymbol{\tau}\right)^T \boldsymbol{\xi}$$

此时

$$R(\boldsymbol{w}, b, \boldsymbol{c}) = \lambda \left\|\boldsymbol{w}\right\|_2^2 - \sum_{n=1}^{N} c_n(y_n(\boldsymbol{w^T x_n} - b) - 1)$$

这非常类似我们之前求解的 hard-margin SVM 的 Lagrangian 函数的形式。

和之前的分析类似，对于 $\boldsymbol{\xi}$ 出现的项，假设 $\frac{1}{N}\mathbf{1_N} - \boldsymbol{c} + \boldsymbol{\tau}$ 有一个维度的元素小于 0，则取 min 后可以令 $\boldsymbol{\xi}$ 对应维度的元素趋于 $+\infty$ 得到最优值为 $-\infty$，反之最优值为 0，即

$$\max_{\boldsymbol{c},\boldsymbol{\tau}\succcurlyeq\mathbf{0}} \min_{\boldsymbol{w},b} L(\boldsymbol{w}, b, \boldsymbol{\xi}, \boldsymbol{c}, \boldsymbol{\tau}) = \begin{cases} \max_{\boldsymbol{c}\succcurlyeq\mathbf{0}} \min_{\boldsymbol{w},b} R(\boldsymbol{w}, b, \boldsymbol{c}) & \boldsymbol{c} - \frac{1}{N}\mathbf{1_N} \succcurlyeq \boldsymbol{\tau} \succcurlyeq \mathbf{0} \\ -\infty & \text{otherwise} \end{cases}$$

当问题存在可行解时得到

$$\max_{\boldsymbol{c},\boldsymbol{\tau}\succcurlyeq\mathbf{0}} \min_{\boldsymbol{w},b} L(\boldsymbol{w}, b, \boldsymbol{\xi}, \boldsymbol{c}, \boldsymbol{\tau}) = \max_{\boldsymbol{c}\succcurlyeq\mathbf{0}} \min_{\boldsymbol{w},b} R(\boldsymbol{w}, b, \boldsymbol{c}),\ s.t.\ \boldsymbol{c} - \frac{1}{N}\mathbf{1_N} \succcurlyeq \boldsymbol{\tau} \succcurlyeq \mathbf{0}$$

当 $\boldsymbol{\tau}$ 取 $\mathbf{0}$ 时可行域中包含了问题所有可能出现的解，因而问题实际上转化为

$$\max_{\boldsymbol{c}} \min_{\boldsymbol{w},b} R(\boldsymbol{w}, b, \boldsymbol{c}),\ s.t.\ \frac{1}{N}\mathbf{1_N} \succcurlyeq \boldsymbol{c} \succcurlyeq \mathbf{0}$$

如此我们剔除了原问题的变量 $\boldsymbol{\xi}$ 和对偶问题的变量 $\boldsymbol{\tau}$。类比 hard-margin SVM 展开 R 的表达式

$$R(\boldsymbol{w},b,\boldsymbol{c})=\lambda\boldsymbol{w}^{\boldsymbol{T}}\boldsymbol{w}-\boldsymbol{w}^{\boldsymbol{T}}\boldsymbol{X}(\boldsymbol{c}\odot\boldsymbol{y})+b\boldsymbol{c}^{\boldsymbol{T}}\boldsymbol{y}+\mathbf{1}_{\boldsymbol{N}}^{\boldsymbol{T}}\boldsymbol{c},\ s.t.\ \frac{1}{N}\mathbf{1}_{\boldsymbol{N}}\succcurlyeq\boldsymbol{c}\succcurlyeq\mathbf{0}$$

为了方便上式的求解，考虑到 λ 是常量，对 $\boldsymbol{c}$ 和函数分别进行代换不会影响最优解的求解

$$\boldsymbol{c}:=\frac{\boldsymbol{c}}{2\lambda}$$
$$R(\boldsymbol{w},b,\boldsymbol{c}):=\frac{R(\boldsymbol{w},b,\boldsymbol{c})}{2\lambda}$$

类比 hard-margin SVM 引入约束条件

$$\boldsymbol{w}-\boldsymbol{X}(\boldsymbol{c}\odot\boldsymbol{y})=0$$
$$\boldsymbol{c}^{\boldsymbol{T}}\boldsymbol{y}=0$$

得到

$$R(\boldsymbol{w},b,\boldsymbol{c})=\mathbf{1}_{\boldsymbol{N}}^{\boldsymbol{T}}\boldsymbol{c}-\frac{1}{2}(\boldsymbol{c}\odot\boldsymbol{y})^{T}\boldsymbol{X}^{\boldsymbol{T}}\boldsymbol{X}(\boldsymbol{c}\odot\boldsymbol{y}),\ s.t.\ \frac{1}{2\lambda N}\mathbf{1}_{\boldsymbol{N}}\succcurlyeq\boldsymbol{c}\succcurlyeq\mathbf{0}$$

和 hard-margin SVM 类似地优化问题被等价地转化为

$$\max_{\boldsymbol{c}}\mathbf{1}_{\boldsymbol{N}}^{\boldsymbol{T}}\boldsymbol{c}-\frac{1}{2}(\boldsymbol{c}\odot\boldsymbol{y})^{T}\boldsymbol{X}^{\boldsymbol{T}}\boldsymbol{X}(\boldsymbol{c}\odot\boldsymbol{y})$$
$$s.t.\ \begin{cases}\boldsymbol{c}^{\boldsymbol{T}}\boldsymbol{y}=0\\ \frac{1}{2\lambda N}\mathbf{1}_{\boldsymbol{N}}\succcurlyeq\boldsymbol{c}\succcurlyeq\mathbf{0}\end{cases}$$

同样考虑核技巧得到

$$\max_{\boldsymbol{c}}\mathbf{1}_{\boldsymbol{N}}^{\boldsymbol{T}}\boldsymbol{c}-\frac{1}{2}(\boldsymbol{c}\odot\boldsymbol{y})^{T}\boldsymbol{K}(\boldsymbol{c}\odot\boldsymbol{y})$$
$$s.t.\ \begin{cases}\boldsymbol{c}^{\boldsymbol{T}}\boldsymbol{y}=0\\ \frac{1}{2\lambda N}\mathbf{1}_{\boldsymbol{N}}\succcurlyeq\boldsymbol{c}\succcurlyeq\mathbf{0}\end{cases}$$

相较于 hard-margin SVM，soft-margin SVM 仅添加了更严格的对变量 $\boldsymbol{c}$ 的上界的约束。取 $\lambda \to 0$ 消去 L2 正则系数，soft-margin 退化为 hard-margin。

这点看起来有点出乎人的意料的，因为在原问题中更松弛的约束在对偶问题中却反而变得更加严格了，但是联想到对偶问题的性质，这一切又解释得通了。为了达到 soft-margin 我们向问题中添加了用于松弛的变量 $\boldsymbol{\xi}$，这样的操作使得对偶问题中约束的数量上升了，因而导致对偶问题的约束更加紧了。

更进一步地说，从 hard-margin 到 soft-margin 问题看上去变得复杂了很多，事实上我们只在问题中引入了新的变量 $\boldsymbol{\xi}$ 和相关的线性约束项。线性约束项具有优秀的性质，我们在线性规划的对偶问题的求解中已经学习过了原问题中新的变量 $\boldsymbol{\xi}$ 的加入仅仅会在对偶问题中添加与之数量相同的线性约束项，且在偏置的存在时会向优化目标引入新的线性项，由于我们新引入的变量和线性约束在 Lagrangian 函数中没有引入任何偏置，因而最终转化为的对偶问题仅仅添加了与变量个数相同的约束项。因而从 hard-margin 到 soft-margin 我们多做的仅仅是只是求解了一次线性规划的对偶问题而已。

SVM 不仅仅是表现优异的分类模型，其思想理论的简洁性使我们作为机器学习初学者也能够轻松地明白模型基本的思想出发点并从中获得启发，其理论推导也很容易作为优化理论初学者深化对 KKT 条件、对核技巧理解的经典案例，是机器学习教材中绕不开的、不可多得的优秀模型。